EXTENDED ABSTRACTS

SEVENTH TRIENNIAL SYMPOSIUM ON TRANSPORTATION ANALYSIS

NTNU – Trondheim
Norwegian University of
Science and Technology

The Research Council
of Norway

SINTEF

Høgskolen i Molde
Vitenskapelig høgskole i logistikk

TROMSØ - NORWAY
JUNE 20-25 2010

TRISTAN VII

TRISTAN VII - Committees

Organizing committee

Post. doc. Henrik Andersson, NTNU
Professor Michel Bierlaire, EPFL
Professor Marielle Christiansen, NTNU
Chief Scientist Geir Hasle, SINTEF (chair)
Associate Professor Arild Hoff, Høgskolen i Molde
Professor Arne Løkketangen, Høgskolen i Molde

Scientific committee

Vinícius A. Armentano,
    Universidade Estadual de Campinas
Mike Ball, University of Maryland
Jaime Barcelo, Universitat Politècnica de Catalunya
Cynthia Barnhart, Massachusetts Institute of Technology
David Boyce, Northwestern University
Olli Bräysy, University of Jyväskylä
Svein Bråthen, Høgskolen i Molde
Angel Corberán, University of Valencia
Jean-Francois Cordeau, HEC Montreal
Teodor Gabriel Crainic, Université du Québec à Montréal
Andrew Daly, University of Leeds
Guy Desaulniers, Ecole Polytechnique de Montreal
Jacques Desrosiers, HEC Montreal
Richard Eglese, University of Lancaster
Kjetil Fagerholt,
    Norwegian University of Science and Technology
Michael Florian, Université de Montréal
Michel Gendreau, Université de Montréal
Richard Hartl, University of Vienna
Martin Hazelton, Massey University
Stephane Hess, University of Leeds
Lars Magnus Hvattum,
    Norwegian University of Science and Technology
Anton Kleywegt, Georgia Institute of Technology
Martine Labbé, Université Libre de Bruxelles
Gilbert Laporte, HEC Montreal
Jesper Larsen, Technical University of Denmark
Odd Larsen, Molde University College
Jean-Patrick Lebacque,
    Institut National de Recherche sur les
    Transports et leur Sécurité
Der-Horng Lee, National University of Singapore

Jean-Baptiste Lesort,
    Institut National de Recherche sur les
    Transports et leur Sécurité
Janny Leung, The Chinese University of Hong Kong
Henry Liu, University of Minnesota
Hong K. Lo,
    Hong Kong University of Science and Technology
Jens Lysgaard, Aarhus University
Oli B. G. Madsen, Technical University of Denmark
Pitu Mirchandani, University of Arizona
Otto Anker Nielsen, Technical University of Denmark
Bjørn Nygreen,
    Norwegian University of Science and Technology
Amedeo Odoni, Massachusetts Institute of Technology
Markos Papageorgiou, Technical University of Crete
Michael Patriksson, Chalmers University of Technology
Warren B. Powell, Princeton University
Christian Prins, Université de Technologie de Troyes
Harilaos Psaraftis, National Technical University of Athens
Mikael Rönnquist,
    Norwegian School of Economics and
    Business Administration
Martin Savelsbergh, Georgia Institute of Technology
Frédéric Semet, École Centrale de Lille
Marius Solomon, Northeastern University
M. Grazia Speranza, Università degli Studi di Brescia
Philippe Toint,
    Facultés Universitaires Notre-Dame de la Paix
Paolo Toth, Università di Bologna
Daniele Vigo, Università di Bologna
Stefan Voss, University of Hamburg
Stein Wallace, Lancaster University

TRISTAN VII – Abstracts

Passenger Improver - A Second Phase Method for Integrated Aircraft-Passenger Recovery Systems

Rodrigo Acuna-Agost

ENAC, Ecole Nationale de l'Aviation Civile, Toulouse, France

Operations Research Division, Amadeus S.A.S., Sophia Antipolis, France

Email: rodrigo.acuna-agost-ext@enac.fr

Mourad Boudia

Operations Research Division, Amadeus S.A.S., Sophia Antipolis, France

Nicolas Jozefowiez

LAAS-CNRS, INSA Université de Toulouse, Toulouse, France

Catherine Mancel, Félix Mora-Camino

ENAC, Ecole Nationale de l'Aviation Civile, Toulouse, France

1   Introduction

Airlines are permanently confronted to disruptions caused by external or internal factors like extreme weather conditions, unavailability of crew members, unexpected breakdowns of aircraft, or airport capacity shortages. These disruptions prevent the planned execution of the schedule, which either becomes suboptimal or infeasible. In this paper, a solution method to solve "*the simultaneous aircraft and passenger recovery problem*" is developed. This approach minimizes the impact of disruptions by taking into consideration the flight schedule, the fleet and maintenance management requirements and the impact on passengers, all simultaneously. This viewpoint is contrasted with the classical approach found in the literature that reallocates resources according to a common hierarchy: aircraft, crew, and finally passengers.

The literature provides some references to approaches where resources are coordinated by a meta-process. Though, there are very few references proposing integrated approaches where a single solution process addresses the problem globally. In this respect, Bratu and Barnhart have proposed an approach in which the objective is to find the optimal trade-off between airline operating costs and passenger delay costs [2]. More recently, the French Operational Research and Decision

Analysis Society (ROADEF) and AMADEUS S.A.S. organized the ROADEF challenge 2009 on the topic: "Disruption Management for Commercial Aviation" [4]. Several research teams participated with original solution methods solving large-scale instances generated by the organization. The approaches presented by the finalist teams include MIP-based methods, minimum-cost flow models, decomposition techniques, and heuristic approaches based on the use of shortest path methods.

In this paper, a post-optimization method that improves the solutions obtained by current approaches is developed. The method, called *Passenger Improver* (PI), provides an optimal reaccommodation for the still disrupted passengers. The term "reaccommodation" refers to a modified assignment of passengers to flight-cabins such that passenger are re-routed to their destinations in the best possible conditions. The remainder of the paper is organized as follows. Section 2 describes two different approaches from the ROADEF challenge used as a first phase of this method. Section 3 presents PI and a mathematical formulation of the problem. Finally, the conclusions and main results are discussed in Section 4.

2   Integrated Aircraft-Passenger Recovery Systems

The authors of this paper have worked developing two solution methods. These approaches are:

NCF: *New Connection Flights* [3]. This is a heuristic approach that tries to recover a perturbed schedule by adding flight legs to connect infeasible aircraft routings while passengers are reaccommodated at each step of the method. These adjustments are locally optimized by applying an adapted version of Dijkstra's algorithm.

SAPI: *Statistical Analysis of Propagation of Incidents* [1]. This method is based on a mixed-integer programming (MIP) formulation that is solved using a statistical analysis of the propagation of disruptions. This approach provides the third best overall results of the competition and it outperformed NCF on small and middle size instances. Nevertheless, on large instances, the MIP becomes too large to perform in a reasonable amount of time.

NCF and SAPI are based on important approximations. In particular, the schedule changes provide unused passenger reaccomodation opportunities.

3   Passenger Improver (PI)

Passengers are grouped in an so-called *itinerary*. Two passengers of the same "itinerary" have to share at least these common characteristics: flights, type of cabin class, and type of trip. The type of the cabin class can be first, business or economy; and the type of the trip can be inbound or outbound trips. Let K be the set of itineraries indexed by k where an *itinerary* k is composed of n_k passengers. After the execution of one of the recovery procedures described in the previous section, itineraries are classified in two groups: non-disrupted (K_F) and disrupted (K_D). The passengers

belonging to K_F are considered fixed and cannot be reaccommodated. On the other hand, K_D contains passengers that need to be reaccommodated because of missed connections, flight leg cancelations or delays. The problem can be formulated as a variation of the classical minimum-cost multi-commodity flow problem. Let $G = (V, E)$ be a directed and connected network, where V is the set of nodes, indexed by i, and E is the set of directed arcs connecting them. To reduce the size of the problem, let V_k be the set of nodes compatible with commodity k. Every itinerary in K_D is considered as a commodity having only one supply node and one sink node. All the remaining nodes, are transshipment nodes representing a pair flight-cabin labeled with R_i, the remaining capacity of the flight-cabin. In general, an arc represents a valid connection for passengers between two flight-cabins. Some extra arcs are added to model cancelation of itineraries by connecting origins to destinations directly. The cost of the flow through each arc is proportional to the amount of that flow, representing delay, cancelation or downgrading costs. These last costs model the inconvenience to passengers when they are reaccommodated to a lower service cabin on all or part of the trip. The objective is then to minimize the total cost of transporting all disrupted passengers to their destinations including the expensive possibility of canceling their trips. Let x_{ij}^k be the decision variables that represent the quantity of flow of commodity (itinerary) k from node i to node j. Thus, the mathematical formulation of this problem is:

$$\text{Minimize} : \sum_{k \in K_D} \sum_{i \in V_k} \sum_{j \in V_k} c_{ij}^k x_{ij}^k$$

Subject to:

$$\sum_{j \in V_k} x_{ij}^k = n_k \qquad \forall k \in K_D, i \in O_k \qquad (1)$$

$$\sum_{j \in V_k} x_{ji}^k - \sum_{j \in V_k} x_{ij}^k = 0 \qquad \forall k \in K_D, i \in V_k \setminus O_k \setminus D_k \qquad (2)$$

$$\sum_{i \in V_k} x_{ij}^k = n_k \qquad \forall k \in K_D, j \in D_k \qquad (3)$$

$$\sum_{k \in I_D} \sum_{j \in V} x_{ij}^k \leq R_i \qquad \forall i \in V \setminus O_k \setminus D_k \qquad (4)$$

$$x_{ij}^k \in \mathbb{N}_0 \qquad \forall k \in K_D, i \in V_k, j \in V_k \qquad (5)$$

The classical multi-commodity flow problem is known to be NP-complete for integer flows. Nevertheless, from numerical tests, it seems that the coefficient matrix is totally unimodular and a formal proof is under study. This property implies that the solution to the linear relaxation of the IP model would be integer and the problem would be solved in polynomial time.

The construction of V_k is a key aspect of this method because it permits to reduce the size of the model. An additional function based on the Floyd - Warshall algorithm is used to limit the nodes compatibles with the passengers by allowing a maximal number of connections per passenger.

4   Results, Conclusion, and Perspectives

Two reference solution methods, NCF and SAPI, have been presented to solve *the simultaneous aircraft and passenger recovery problem.* They have their respective strengths, but also show a common weakness, as they do not guarantee an optimal reaccommodation of disrupted passengers. To complement these two approaches, a new method called Passenger Improver (PI) is presented to reaccommodate disrupted passengers based on a particular case of the minimum-cost multicommodity flow problem. The algorithm was tested on a PC Intel Core Duo T5500 1.66 GHz 2 GB RAM over Windows Vista Operation System using ILOG CPLEX 11.1 as a standard MIP solver. The results evidence important improvements to the final solutions of NCF and SAPI over the 32 instances of the ROADEF challenge 2009, some of them with more than 700000 passenger and 6000 flight-legs. Actually, PI decreases the total cost by 18% (NCF) and 5.92% (SAPI) with an average CPU time of 120 [s]. Additionally, to the best of our knowledge, 71.8% of the current best known solutions for these instances have been calculated using NCF, SAPI and PI.

A future research direction is the development of a fully integrated system primarily based on SAPI. First, an adaptation of NCF will be used to calculate an initial solution. The main part of the algorithm will perform improvements by the study of propagation of disruptions (SAPI). Finally, the post-optimization function will be replaced by an adaptation of PI to have a better reaccomodation for still disrupted passengers. We expect that this integration will help to improve the CPU time rather than the quality of solutions.

References

[1] R. Acuna-Agost, D. Feillet, P. Michelon, and S. Gueye; "Rescheduling Flights, Aircraft, and Passengers Simultaneously under Disrupted Operations - A Mathematical Programming Approach based on Statistical Analysis", *AGIFORS*. Submitted to Anna Valicek Medal 2009, (2009)

[2] S. Bratu and C. Barnhart, "Flight operations recovery: New approaches considering passenger recovery", *Journal of Scheduling* 9, 279298 (2006)

[3] N. Jozefowiez, C. Mancel, and F. Mora-Camino, "Approche heuristique pour la gestion de pertubation dans le domaine aérien (in French)", http://challenge.roadef.org/2009/methodes.en.htm, ROADEF (2009)

[4] M. Palpant, M. Boudia, C.A. Robelin, S. Gabteni, and F. Laburthe. "ROADEF 2009 challenge: Disruption Management for commercial aviation", *Working paper, AMADEUS S.A.S.*, Operations Research Division, Sophia Antipolis, France (2009).

Dynamic Ride-Sharing in Metro Atlanta

Niels Agatz

Rotterdam School of Management

Erasmus University

Email: nagatz@rsm.nl

Alan Erera

School of Industrial and Systems Engineering

Georgia Institute of Technology

Martin Savelsbergh

School of Industrial and Systems Engineering

Georgia Institute of Technology

Xing Wang

School of Industrial and Systems Engineering

Georgia Institute of Technology

1 Introduction

Rising gas prices, traffic congestion, and environmental concerns have increased the interest in services that allow people to use their cars more wisely. While ride-sharing is not new, the ubiquity of Internet-enabled cell phones has enabled practical *dynamic* ride-sharing. By dynamic ride-sharing we refer to a system where an automated process provided by a ride-share provider matches up drivers and riders for a ride on very short notice or even en-route. Recently, many new companies have emerged with ride-sharing concepts similar to those we present in this paper. For example, providers like Carticipate, EnergeticX/Zebigo, Avego, and Piggyback recently started offering mobile phone applications that allow drivers with spare seats to connect to people wanting to share a ride. To ease the fear of sharing a ride with a stranger these services can use reputation systems (see e.g. PickupPal) or can be linked with social network tools like Facebook (see e.g. GoLoco and Zimride).

The ability of a dynamic ride-share provider to successfully establish ride-shares on short notice depends on the characteristics of the environment in terms of participation density, traffic patterns,

and type of infrastructure. Hall and Qureshi [1] analyze the likelihood that a person will be successful in finding a ride-match, given a pool size of potential ride matches. Based on a simple probabilistic analysis, they conclude that in theory ride-sharing is viable since a congested freeway corridor should offer sufficient potential ride-matches. The authors also observe that there are many obstacles, primarily in terms of communication, so that the chance of finding a ride-match in practice may in fact be small. Fortunately, the recent advances in mobile communications give rise to new opportunities for matching up people for rides in real-time.

Even though the enabling technology is there, ride-sharing success stories are still lacking. The development of algorithms for optimally matching drivers and riders in real-time may only play a small role in the ultimate success of ride-sharing, it is at the heart of the ride-sharing concept, and the transportation community has largely ignored it.

2   Dynamic Ride-Share Problem Setting

Both riders and drivers must provide information on their time schedule preferences. Many of the currently available and proposed dynamic ride-share applications simply let each potential participant specify a desired departure time. The provider then attempts to find an assignment with a departure time that is as close as possible to this desired departure time. This approach minimizes the information that participants must supply, but, at the same time, provides only limited information regarding a participant's time preferences and flexibility. A time window representation may capture a participant's time preferences more accurately. One could, for example, let a participant specify an earliest possible departure time and latest possible arrival time. Furthermore, it may be beneficial to allow limits on the actual time that users may spend traveling on a given trip, for example by allowing each participant to specify the maximum excess travel time (over the direct travel time for his origin to destination) he is willing to accept.

Ride-sharing allows people to save on travel-related expenses by sharing trip costs. A ride-share provider, either private or public, helps people to establish ride-shares on short-notice by automatically matching up drivers and riders. If the system is private and operated for profit, the added value of the ride-share provider is reducing the total costs of all participants by the largest amount possible; by enabling this economy, the provider can take a cut. Private ride-share providers typically charge a commission per successful ride-share, either a fixed fee or proportional to the trip cost. As a result, the objective of the provider is mostly in line with the goals of the participants. This is also true for a public system with a societal objective, such as the reduction of pollution and congestion.

Here, we focus on the establishment of ride-shares that minimize the system-wide vehicle-miles. The system-wide vehicle-miles represent the total vehicle-miles driven by all participants

traveling to their destinations, either in a ride-share or driving alone. This objective is important from a societal point of view as its helps to reduce pollution (emissions) and congestion. This objective is compatible with *minimizing total travel costs*, which is an important consideration for the participating drivers and riders and directly related to the revenues of the ride-share provider.

In any practical dynamic ride-share implementation, new riders and drivers continuously enter and leave the system. A driver enters the system by announcing a planned trip and offering a ride and a rider enters the system by announcing a planned trip and requesting a ride. Drivers and riders leave the system when a match has been found for them or when their planned trips "expire," i.e., when the latest possible departure time of their planned trip occurs without a match being found.

Since new drivers and riders continuously arrive, not all relevant offers and requests may be known at the time the ride-share provider plans the ride-shares. A common approach for dealing with these types of planning uncertainties is to use a rolling horizon approach. In this paper, we consider several different re-optimization strategies and frequencies. Notably, we chose to re-optimize each time a new trip is announced or to re-optimize at fixed time intervals. The best choice is not obvious. Moreover, we develop heuristics that are fast enough to handle realistic-size instances in a matching engine of an actual ride-share system, i.e. thousands of riders and drivers that travel between thousands of origins and destinations at the same time.

The optimization problem that needs to be solved involves all the offered rides (drivers) and requested rides (riders) that are known at the time of the optimization and that have not yet been matched. The ride-share provider may decide to immediately notify drivers and riders of the matches identified by the optimization or may decide to hold off on notifying drivers and riders in the hope of improved matches the next time the optimization is executed. For example, if the next optimization is scheduled at time t, then a match between driver d and rider r with $\min\{b(d), b(r)\} \geq t$ can be postponed without negatively impacting cost savings.

Although the issues of re-optimization frequency and solution commitment are not unique to ride-share systems, ride-share systems offer a highly dynamic environment in which to study them.

3 Simulation Experiments

For this study, we have developed a simulation environment based on the 2009 travel demand model for the metropolitan Atlanta region, developed by the Atlanta Regional Commission (ARC). The ARC is the regional planning and intergovernmental coordination agency for the 10-county Atlanta area, a sprawling region with a population of approximately 5 million people occupying 6,500 square miles. The travel demand model for the region is used in this study to generate daily vehicle trips by purpose between all pairs of travel analysis zones within the region.

1. Participation rate: the percentage of the total system-wide vehicle-trips that is participating in dynamic ride-sharing.

2. Announcement lead-time: the time before the latest departure time that a participant announces his trip. This can be an absolute value or a relative value i.e., relative to the duration of the trip.

3. Ride-share time-flexibility: The difference between the earliest and latest departure time minus the direct trip length. This can be an absolute value or a relative value, i.e., relative to the duration of the trip.

4. Driver-rider ratio: The ratio between the number of drivers and the number of riders.

Ongoing experimentation with different matching optimization approaches within the simulation environment has begun to generate results that demonstrate the potential additional value to ride-sharing systems of using more sophisticated matching algorithms rather than simple heuristics. For example, for the simplest problem setting where each driver may be matched with at most one rider, a solution approach that uses a network-flow formulation to solve a bipartite matching model optimally systematically outperforms (by 10-20 %) a greedy matching heuristic in terms of success-rate, vehicle-miles and corresponding cost savings. Furthermore, comparison to a best-possible *a posteriori* bound has provided evidence that the rolling horizon approach is nearly optimal for practical instances. The numerical experiments also show that increasing the percentage of the total vehicle-trips that announce to potentially participate in ride-sharing results in relatively more ride-share arrangements, and also increases the average per-trip cost-savings for the participants. The simulation environment enables clear study of the economy of participant density for such systems, an important determinant of potential success.

References

[1] Hall, R. W. and Qureshi, A., "Dynamic ride-sharing: Theory and practice", *Journal of Transportation Engineering* 123, 4, 308-315 (1997).

Discrete time models for maritime inventory routing problems

Agostinho Agra

Department of Mathematics

University of Aveiro, Portugal

Email: aagra@ua.pt

Henrik Andersson

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Email: henrik.andersson@iot.ntnu.no

Marielle Christiansen

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Email: marielle.christiansen@iot.ntnu.no

The maritime inventory routing problem (IRP) combines a ship routing problem with an inventory management problem. The goal is to find routes and schedules for the fleet and load/unload quantities at each port in order to minimize the transportation costs while satisfying the inventory lower and upper bound levels at each port.

During the last decade optimization problems in maritime transportation have received a clearly increased interest compared to air and land-based transportation problems that have been intensively studied for several decades. This increase of interest has mainly been motivated by the large economic importance and by the high complexity of real maritime transportation problems. These two factors created the need to provide good support systems to maritime transportation (MT) industrial planners. As a consequence, most of the literature on maritime inventory routing is usually concerned with real applications (e.g. [1, 5, 6, 7, 8, 9, 11, 13, 14, 15]). Christiansen et al. [4] present a recent review on maritime transportation, while [3] is devoted to maritime IRPs. In a work closely related to the one we consider, Christiansen [2] studies a supply chain for ammonia consisting of several locations that either produce or consume ammonia and the transportation network between these locations. Ammonia is produced and stored in inventories at given loading ports and transported at sea to inventories at unloading or consumption ports. Inventory capacities are defined in all ports. Here, the production and consumption rates are given and fixed during the planning horizon in all

ports. To transport the product between the given production and consumption ports, the planners control a heterogeneous fleet of ships. The planning problem is to design routes and schedules for the fleet that minimize the transportation costs without interrupting production or consumption at the inventories. The overall problem is solved by a branch-and-price method in [5, 6] and by a heuristic in [7].

Most of the common approaches for solving the maritime IRP are based on heuristics or decomposition techniques (e.g. [2, 5, 6, 8, 11]). The choice of these approaches might be explained by the high complexity of real applications. However, the constant hardware development combined with the theoretical advances in optimization techniques has produced optimization solvers capable of handling increasingly larger instances. Currently, it is possible to obtain optimal or near optimal solutions, in acceptable computational time, to small real instances occurring in MT problems using commercial solvers. It is well-known that to solve a problem efficiently, the formulation of it is crucial [10]. This makes the study of the mathematical formulation a key issue to solve larger maritime IRPs. Although the study of valid inequalities for mixed-integer sets and the derivation of extended formulations is currently receiving large attention with several applications to other mixed-integer problems, little work has been done on applying these techniques to maritime transportation problems. However, a few contributions already exist within maritime transportation. Sherali et al., [15], included valid inequalities in order to enhance the proposed formulations to an oil products transportation problem, and Persson and Göthe-Lundgren, [11], developed valid inequalities within a column generation approach for a maritime IRP with production considerations. Recently, Grønhaug et al. [8], include valid inequalities to improve the path-flow formulation presented for the liquefied natural gas inventory routing problem.

A maritime IRP may be characterized in different ways, such as the number of products (single item vs multi-item); inventory management at all ports vs inventory management only at production or consumption ports; constant, varying or variable production and consumption rates, etc. When considering production and consumption rates the underlying models may be quite different. If it is assumed that the production and consumption rates are fixed during the planning horizon, then a mathematical model based on a continuous time can be used, see for example [1] and [2]. When the production and/or consumption rate is varying or variable during the planning horizon a discrete time model has been used, see [8, 13]. The case of variable production and consumption rates is, of course, the most general one. In practice, the production and consumption rates are most often variable, although, in some applications, the simplification made by assuming a constant rate is acceptable.

We consider a single product problem with varying production and consumption rates with inventory management at all ports. The product is produced and stored at production (loading) ports and transported by a heterogeneous fleet of ships to the consumption ports (unloading ports). Inventory capacities and inventory safety stocks are considered at the production and consumption ports. The problem is similar to the one given in [2]. The major differences are related to the varying rates of

production/consumption which implies that the mathematical model is based on a discrete time horizon. We present and discuss different mathematical formulations for this maritime IRP and outline approaches to strengthen the proposed formulations. These approaches include the study of extended formulations and the inclusion of valid inequalities [10]. We focus on deriving valid inequalities for this particular type of problems, such as, inequalities that impose visits to ports during an interval of time, based on the demands, capacity of ports and safety stocks.

Since the maritime IRP is a general problem that incorporates characteristics from other well-known intensively studied problems such as the vehicle routing problem [16] and the capacitated lot-sizing problem [12], the approaches followed explore also the connections between the maritime IRP and these other related problems.

Computational results based on real data will be reported.

References

[1] F. Al-Khayyal and S.J. Hwang, "Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk, Part I: Applications and model", *European Journal of Operational Research* 176, 106-130, 2007.

[2] M. Christiansen, "Decomposition of a combined inventory and time constrained ship routing problem", *Transportation Science* 33, 3-16, 1999.

[3] M. Christiansen and K. Fagerholt, "Maritime inventory routing problems", in *A chapter for Encyclopedia of Optimization*, second edition, Springer, 2009.

[4] M. Christiansen, K. Fagerholt, B. Nygreen and D. Ronen, "Maritime transportation", in *Handbook in Operations Research and Management Sciences* 14, Transportation. C. Barnhart and G. Laporte (eds.), 189-284, North-Holland, Amsterdam., 2007.

[5] M. Christiansen and B. Nygreen "A method for solving ship routing problems with inventory constraints", *Annals of Operations Research* 81, 357-378, 1998.

[6] M. Christiansen and B. Nygreen "Modeling path flows for a combined ship routing and inventory management problem", *Annals of Operations Research* 82, 391-412, 1998.

[7] T. Flatberg, H. Haavardtun, O. Kloster and A. Løkketangen, "Combining exact and heuristic methods for solving a vessel routing problem with inventory constraints and time windows", *Ricerca Operativa* 29, 55-68, 2000.

[8] R. Grønhaug, M. Christiansen, G. Desaulniers, and J. Desrosiers, "A branch-and-price method for a liquefied natural gas inventory routing problem", Fortcoming in *Transportation Science,* 2010.

[9] S.J. Hwang, "Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk", Ph.D. Thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology, GA, 2005.

[10] G. Nemhauser and L. Wolsey, "Integer and Combinatorial Optimization", *Wiley*, New York, 1988.

[11] J. A. Persson and M. Göthe-Lundgren, "Shipment planning at oil refineries using column generation and valid inequalities", *European Journal of Operational Research* 163, 631-652, 2005.

[12] Y. Pochet and L. Wolsey, "Production planning by mixed integer programming", *Springer*, New York, 2005.

[13] D. Ronen "Marine inventory routing: Shipments planning", *Journal of the Operational Research Society* 53, 108-114, 2002.

[14] M. Stålhane, J. G. Rakke, C. R. Moe, H. Andersson, M. Christiansen and K. Fagerholt, "A construction and improvement heuristic for a liquefied natural gas inventory routing problem", 47 pages, Working paper, NTNU, Trondheim, Norway, 2009.

[15] H. Sherali, S. Al-Yakoob and M. Hassan, "Fleet management models and algorithms for an oiltanker routing and scheduling problem", *IIE Transactions* 31, 395-406, 1999.

[16] H. Yaman "Formulations and valid inequalities for the heterogeneous vehicle routing problem", *Mathematical Programming* 106, 2, 365-390, 2006.

Strong mixed integer formulations for a short sea fuel oil distribution problem

Agostinho Agra

Department of Mathematics

University of Aveiro, Portugal

Email: aagra@ua.pt

Marielle Christiansen

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Email: marielle.christiansen@iot.ntnu.no

Alexandrino Delgado

Department of Mathematics

University of Cape Verde

Email: alexandrino.delgado@unicv.edu.cv

We consider a short sea fuel oil distribution (SSD) problem occurring in the archipelago at Cape Verde. Here, an oil company is responsible for the routing and scheduling of ships between the islands, as well as for the management of the inventory of various fuel oil products.

During the last decade optimization problems in maritime transportation have received a clearly increased interest compared to air and land-based transportation problems that have been intensively studied for several decades. Christiansen et al. [4] present a recent review on maritime transportation, while [3] is devoted to maritime inventory routing problems. Combined routing and inventory management within maritime transportation have been present in the literature the last decade only. See [1, 2, 6, 7, 8, 9 10, 13, 14, 15].

The inter-islands distribution of fuel oil is a real problem of Cape Verde, an archipelago with nine inhabited islands. Fuel oil products are imported and delivered to specific islands and stored in large supply storage tanks. From these islands, fuel oil products are distributed among all the inhabited islands using a small heterogeneous fleet of ships. These products are stored in consumption storage tanks. Some ports have both supply and consumption tanks. The inter island distribution plan consists of designing routes and schedules for the fleet of ships including the loading and unloading quantity of each product at each port. This plan must satisfy the demand of each product at each island per period, time window constraints on the operations, and the capacities of the ships, ports and depots. The

objective is to minimize the overall cost including sailing costs, a fixed cost for each operation and a penalty cost for violation of time windows.

We consider a short-term distribution problem with a planning horizon of twelve days. The input to this problem is the output of a medium-term plan for several weeks (few months). The demands correspond to the quantities to be delivered at each port per day determined in the medium-term plan. Hence, usually the demands at each port follow a pattern where the demands are zero for most periods and relatively large in the rest of the periods. By coordinating the distribution of all products in all ports during the planning horizon, it might be efficient to deliver the demand in periods prior to the specified period by the medium-term plan or in other quantities. This means that we need to keep track of the inventory level at the consumption storage tanks for all products in all ports. Storage capacities of supply and consumption tanks are taken into account in the medium-term planning. In the short-term plan considered here capacities of supply tanks can be ignored, since the total consumption of each product from all consumption tanks during the time horizon, is much smaller than the capacity of the supply tanks. However, the capacity of the consumption tanks for a particular product can be less than the total demand of the product over the planning horizon.

It is assumed that only one ship can visit a port simultaneously. During a port call for a ship, it is possible to load and unload different products. We assume that there is a fixed (un)loading time per unit of product (un)loaded. This time may vary for different products and different ports. In addition, there is a considerable set up time between (un)loading different products due to coupling and decoupling of pipes between tanks in the ship and tanks in port.

Most of the ports are closed during night and some ports have operational restrictions during certain periods of the day. This means that in each period (day), there may be a time window for (un)loading. These time windows may vary from port to port. A ship cannot start to operate before the beginning of the time window. However, if the operation has begun inside the time window, it can be finished outside that time window. In this case, an extra man-power cost is incurred.

To transport the fuel products between the islands, the planners control a small, heterogeneous fleet of ships. Each ship has a specified load capacity, fixed speed and cost structure. The set up time and the (un)loading time is independent of the ship. The cargo hold of each ship is separated into several cargo tanks. However, we do not consider the allocation of the different fuel products into different cargo tanks.

We propose a mixed integer formulation for the SSD problem. The formulation is related to some previous works; see [1, 2, 9, 14]. In [1] and [2] a time continuous model is presented and an index indicating the visit number to a particular port is introduced. For both these underlying models it is assumed that the production/consumption rate is given and fixed during the planning horizon. In [9] and [14] time discrete models are developed to overcome the complicating factors with variable production and consumption rates. In the SSD problem, the inventory at the production side is not considered, but we have variable consumption rates during the planning horizon. Hence the

mathematical model includes both a continuous and discrete time horizon due to the multiple time windows and a daily varying consumption rate of the various products in the different ports.

It is known that deriving a good model is essential for efficiently solving a mixed integer program to optimality [10]. In order to improve the original formulation we derive stronger formulations, that is, formulations whose linear relaxation is tighter than the original one. We consider three different approaches to strengthen the original formulation based on standard techniques. The first one is to tighten the bounds on constraints linking continuous and binary variables. The second one is based on a reformulation of the model with the inclusion of additional variables, the arc-load flow variables instead of considering the amount of fuel of each product onboard the ship when it is leaving a port. The new variables also indicate the next port in the ship route. The inclusion of these variables increases the size of the model but it also allows us to derive a tighter model. The last improvement is related to the inclusion of valid inequalities. In order to derive valid inequalities we consider simpler substructures arising from relaxations of the set of feasible solutions of the model. For instance, we identify the fixed charge single node flow set, when aggregating the demand for a set of products during a subset of periods in a given subset of islands, and identify other (mixed-) integer sets resulting from considering the routing constraints, the inventory constrains, the arc-load flow constraints (from the arc-load flow extended formulation) alone. These substructures have been intensively studied, from the point of view of the theory of valid inequalities, during the last years. From well-known inequalities for these simple (mixed-) integer sets we derive inequalities for our model. Since the amount of families of known inequalities for these substructures is very large, an extensive computational study of their relevance in solving our instances would be impractical. We decide to focus on few well-known families of inequalities that proved to be efficient in solving related problems, such as cover type inequalities, see [16], mixed integer rounding inequalities, see [11], etc..

Finally, an extensive computational study, based on real data, is reported in order to compare different ways of combining the approaches. Based on this computational study we propose a final improved formulation which can solve all the tested instances to optimality within a reasonable computational time.

References

[1] F. Al-Khayyal and S.-J. Hwang, "Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk, Part I: Applications and model", *European Journal of Operational Research* 176, 106-130, 2007.

[2] M. Christiansen, "Decomposition of a combined inventory and time constrained ship routing problem", *Transportation Science* 33, 3-16, 1999.

[3] M. Christiansen and K. Fagerholt, "Maritime inventory routing problems", in *A chapter for Encyclopedia of Optimization*, second edition, Springer, pp. 1947-1955, 2009.

[4] M. Christiansen, K. Fagerholt, B. Nygreen and D. Ronen, "Maritime transportation", in *Handbook in Operations Research and Management Sciences* 14, Transportation. C. Barnhart and G. Laporte (eds.), 189-284, North-Holland, 2007.

[5] M. Christiansen, K. Fagerholt, and D. Ronen, "Ship routing and scheduling: Status and perspectives", *Transportation Science* 38, 1-18, 2004.

[6] M. Christiansen, and B. Nygreen, "A method for solving ship routing problems with inventory constraints", *Annals of Operations Research* 81, 357-378, 1998.

[7] M. Christiansen, and B. Nygreen, "Modeling path flows for a combined ship routing and inventory management problem", *Annals of Operations Research* 82, 391-412, 1998.

[8] T. Flatberg, H. Haavardtun, O. Kloster and A. Løkketangen, "Combining exact and heuristic methods for solving a vessel routing problem with inventory constraints and time windows", *Ricerca Operativa* 29, 55-68, 2000.

[9] R. Grønhaug, M. Christiansen, G. Desaulniers, and J. Desrosiers, "A branch-and-price method for a liquefied natural gas inventory routing problem", Forthcoming in *Transportation Science*, 2010. [10]

[10] S.-J. Hwang, "Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk", Ph.D. Thesis, School of Industrial and Systems Engineering, Georgia Institute of Technology, GA, 2005.

[11] H. Marchand and L. Wolsey, "Aggregation and mixed integer rounding to solve MIP's", *Operations Research* 49, 363-371, 2001.

[12] G. Nemhauser and L. Wolsey, "Integer and Combinatorial Optimization", *Wiley*, 1988.

[13] J. A. Persson and M. Göthe-Lundgren "Shipment planning at oil refineries using column generation and valid inequalities", *European Journal of Operational Research* 163, 631-652, 2005.

[14] D. Ronen "Marine inventory routing: Shipments planning", *Journal of the Operational Research Society* 53, 108-114, 2002.

[15] M. Stålhane, J. G. Rakke, C. R. Moe, H. Andersson, M. Christiansen and K. Fagerholt, "A construction and improvement heuristic for a liquefied natural gas inventory routing problem", 47 pages, Working paper, NTNU, Trondheim, Norway, 2009.

[16] H. Yaman, "Formulations and valid inequalities for the heterogeneous vehicle routing problem", *Mathematical Programming* 106, 2, 365-390, 2006.

A Methodology to Assess Vessel Berthing and Speed Optimization Policies

J Fernando Alvarez

Research & Innovation

Det Norske Veritas AS

Tore Longva

Business Development

Det Norske Veritas AS

Erna Engebrethsen

Asset Risk Management

Det Norske Veritas AS

Email: jose.fernando.alvarez@dnv.com

1   Extended Abstract

Port congestion has been a serious problem for the maritime shipping sector for many decades [1]. Major causes for port congestion include insufficient terminal equipment, inadequate hinterland infrastructure, labor shortages or conflicts, and poor managerial practices. The widespread utilization of antiquated shipping contracts and berthing priority policies also constitutes a major driver of port congestion.

Standard ocean shipping contracts[1] require a chartered vessel to proceed at "utmost despatch" to its destination, even when it is almost certain that the vessel will have to wait for several days before being admitted to a berth. The berthing policies at many major ports, which admit vessels on a first-come, first-served (FCFS) basis, represent an additional incentive for the master to sail at full speed. The widespread utilization of these legacy contracts and berthing policies constitutes a major, and arguably unreasonable, driver of harbor congestion and marine fuel consumption.

As was observed during the grounding of the *Pasha Bulker* at Newcastle, Australia, in 2007 and the grounding of the *Full City* outside Langesund in 2009, port congestion can constitute a serious safety issue [2, 3].

Fuel expenses represent up to 50% of voyage costs, and are a major concern to fleet operators. Given that the rate of fuel consumption of a vessel increases approximately as the cube of the

---

[1] Ocean shipping contracts are known as charter party forms, and establish the conditions of carriage, including dates for loading and unloading, cost of carriage, and penalties for delays. The charter party form is typically a contract between the vessel owner/operator and the shipper.

vessel's speed [4], the aforementioned contracts and policies result in unnecessarily high bunker fuel consumption.

The maritime transport sector is currently placing much emphasis on the reduction of operational expenses and carbon emissions. The issue of safety is also paramount amongst responsible vessel and terminal operators. This motivates us to consider alternative vessel berthing policies and contractual mechanisms that allow the maritime transport industry as a whole to operate in a more safe, environmentally responsible, and economical manner. Some of the largest actors in maritime shipping have expressed strong support for such alternative contractual and coordination mechanisms. For instance, the *virtual arrival* initiative – a simple mechanism for reducing vessel speed and distributing the economic benefits amongst vessel owners and charterers – is backed by prominent industry associations such as INTERTANKO and OCIMF [5]. Another example is that of the port of Newcastle, which has recently implemented a new berth allocation system to replace the traditional first-come-first-served discipline.

The contribution of this paper consists of a methodology to evaluate the benefits that can be derived from innovative berthing priority policies and ocean shipping contracts. Given the importance of stochasticity on the performance of maritime transport systems, as well as the need to represent the scheduling and allocation decisions made by the terminal planner, we propose a discrete event simulation model with an embedded optimization routine as the appropriate framework to assess new policies.

The simulation model includes seven event types, representing the departure of vessels from a remote port; a communication event where a vessel may receive sailing instructions from the terminal planner; arrival of the vessels at the focal harbor; berthing and commencement of loading or unloading; allocation of land-side equipment (LSE) to the vessel; end of land-side operations and departure from the focal port; and reoptimization by the terminal planner. A reoptimization event is triggered whenever a vessel communicates an updated position to the planner, when a vessel arrives at the harbor, or when a vessel leaves the terminal.

In order to represent the role of the terminal planner, we formulate a fairly detailed mixed integer program (MIP) that finds feasible assignments of berths and land-side equipment to each vessel. Our MIP formulation is a substantial extension on the traditional berth assignment problem (BAP, see e.g. [6, 7]), as we include many issues that are important in a practical setting. For instance, our model includes the fuel consumption of different vessel types at different speeds, the compatibility between LSE and different merchandise types, the spatial configuration of the terminal, and the transfer of demurrage and despatch fees. By modifying a small number of parameters and sets in the MIP formulation, we can represent different berthing priority and speed optimization policies.

An interesting feature of the proposed MIP is that it represents the planner's decisions at

different levels of precision. The most immediate decisions, those that will be implemented within the current work shift, are represented using hourly time intervals. The following shift is represented using two-hour blocks, and the third shift is represented using four-hour blocks. This technique results in a considerable reduction of the size of the MIP instances, and resembles the rolling planning horizon often employed in terminal planning activities.

We present a case study where we compare the performance of the simulated transport system under three policies, namely FCFS, standardized estimated time of arrival (SETA), and global optimization of speed, berth, and equipment allocation (GOSBEA). Under policy GOSBEA, the terminal planner dictates the target sailing speed of each vessel, and assigns bething priorities accordingly. We compare the performance of the policies in terms of average harbor dwell time, terminal throughput, total fuel consumption, and the balance of despatch credits and demurrage fees. Furthermore, we compare the three policies under two different market scenarios, resembling the positive industry outlook of 2008, and the depressed circumstances observed in 2009. We conduct 120 replications, twenty for each of three policies in each of the two market scenarios.

We conclude that policy GOSBEA is substantially better than SETA and FCFS. In particular, it appears that policy GOSBEA would lead to significant reductions in fuel consumption, while simultaneously reducing harbor dwell times, vessel call cancellations, and contractual penalties.

While GOSBEA is clearly beneficial when all parties are considered jointly, we can anticipate that not every individual vessel operator would benefit from the scheme. For instance, the terminal planner may request a vessel to sail faster than initially planned, even when the vessel is not at risk of missing its contractual window for land-side operations. The vessel operator may not be willing to expend additional fuel for the benefit of other parties. Such cases may well represent the biggest obstacle to the implementation of GOSBEA or similar policies. Additional contractual mechanisms must be implemented in order to compensate the ocassional losses that would otherwise be imposed on individual vessel operators. The precise specification of such mechanisms constitutes an interesting direction for further research.

References

[1] P. Alderton, *Port Management and Operations*. Informa, 3rd ed., 2008.

[2] A. Corbett, "Safety push to cut traffic at Newcastle," *Tradewinds*, p. 58, 16 October 2009.

[3] Norwegian Coastal Administration, "Preliminary report, investigation of the grounding of MV Full City, IMO no 9073672 at Sastein july 31st 2009," tech. rep., 2009.

[4] P. Alderton, *Reeds Sea Transport Operations and Economics*. Thomas Reed Publications, 5th ed., 2005.

[5] INTERTANKO and OCIMF, "Virtual arrival – a best practice guide," London, March 2010.

[6] L. Moccia, *New Optimization Models and Algorithms for the Management of Maritime Container Terminals.* PhD thesis, Universita degli studi della Calabria, 2004.

[7] J. Cordeau, G. Laporte, P. Legato, and L. Moccia, "Models and tabu search heuristics for the berth allocation problem," *Transportation Science*, vol. 39 (4), pp. 526–538, 2005.

Train load planning in seaport container terminals

Daniela Ambrosino

Department of Economics and Quantitative Methods, University of Genova, Italy

Simona Sacone, Silvia Siri

Department of Communication, Computer and Systems Science, University of Genova, Italy

1 Introduction

Seaport container terminals are nowadays very complex systems that require the development of quantitative methods to support the relevant decisions. Referring to the surveys [1] and [2] on operations research methods applied to container terminals, the developed optimization approaches can be divided according to the different processes in a seaport terminal: ship planning (i.e. berth allocation, stowage planning and crane split), storage and stacking planning, and transport optimization (divided in quayside, landside, and crane movements). With respect to this classification, this work is devoted to landside transport optimization and presents an optimization approach for the definition of loading plans for trains. As highlighted in [1], a loading plan indicates on which wagon a container has to be placed; this decision generally depends on the destination, type and weight of the container, the maximum load of the wagon and the wagon's position in the train. Also the container location in the storage area can influence the loading plan. We consider the case in which this loading plan is performed by the terminal operator with the aim of optimizing both the pick-up operations in the storage area and the load of each train. In the literature, few research studies are devoted to the load planning problem and they are referred to landside intermodal terminals rather than to seaports. In [3] the authors propose some models and heuristic methods for container allocation problems on trains, referring to rail-rail terminals with rapid transfer yards, whereas in [4] the authors consider a terminal where containers are transferred to and from trucks on a platform adjacent to the rail tracks provided with a short-term storage area.

The present work addresses the train load planning problem in a container terminal in which the railway yard works as sketched in Fig. 1. The transfer of containers from the stocking area (where they are stacked up until the forth tier) to the train is realized as follows: a reach stacker takes a container from the stocking area and puts it on a tractor; then, the tractor moves it near the railway tracks where it is taken by an overhead travelling crane that loads it on the train. We assume to plan the train loading operations over a given planning horizon (e.g. one or more days).
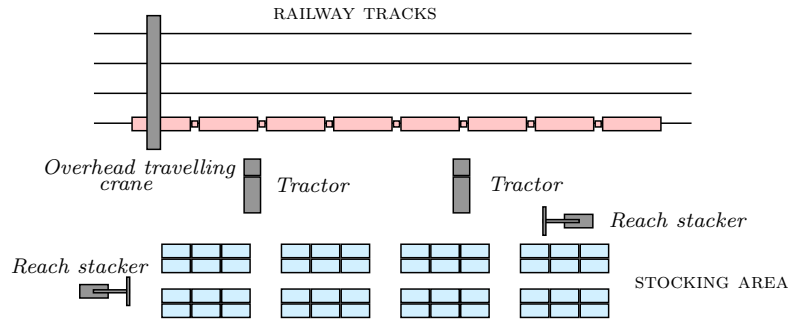
Figure 1: A sketch of the railway yard.

We are interested in studying the load planning problem either in a seaport terminal or an inland terminal (such as a dry port) directly connected with the seaport. In the former case, import containers must be loaded on trains, whereas in the latter case we focus on export containers directed to ships. Therefore, four different scenarios (of increasing difficulty) could be considered:

1. *train load planning of import containers on one track* - in the stocking area containers are stocked following a very simple policy (depending on their weight and length); the destination of containers is not taken into account in the load planning problem; the overhead travelling crane is assumed to load containers sequentially, i.e. from the first wagon to the last one;

2. *train load planning of import containers on more tracks* - the stocking policy is very simple and the destination of containers is not considered, as in scenario 1; in this case, the loading sequence of the overhead travelling crane is matter of decision (trains on parallel tracks can be loaded either sequentially or in parallel);

3. *train load planning of export containers on one track* - containers are supposed to be stored according to a specific policy that takes into account more factors as the ship destination, the weight, size and destination of containers (these aspects will be taken into account in the planning problem); the overhead travelling crane is assumed to load containers sequentially;

4. *train load planning of export containers on more tacks* - as in scenario 3, the ship destination of containers influences both the stocking policy in the storage area and the train load planning; instead, the loading sequence of the overhead travelling crane is matter of decision.

2   Problem definition

We now consider scenario 1, i.e. train load planning of import containers on one track. In the optimization problem, we consider C containers, R trains and W wagons; wagons are ordered such that $w = 1, \ldots, W_1$, are wagons of train $r = 1$ (leaving as first), $w = W_1 + 1, \ldots, W_2$, are

wagons of train $r = 2$ (leaving as second), and so on, until the set of wagons of the last train, i.e. $w = W_{R-1} + 1, \ldots, W_R$; for notational purposes, we set $W_0 = 0$ and $W_R = W$. Moreover, we suppose that the position of a container in the stocking area is given by the relevant slot and tier. To this end, let us denote with S the number of slots and with T the maximum number of tiers for slot. The data relative to each container $c = 1, \ldots, C$, are the length λ_c, the weight ω_c and the cost of not loading it, i.e. π_c (that takes into account the urgency and importance of the container). For each wagon $w = 1, \ldots, W$, the input data are the length $\overline{\lambda}_w$ and the weight capacity $\overline{\omega}_w$. Moreover, $\overline{\Omega}_r$ is the weight capacity of train $r = 1, \ldots R$, and γ is a cost term associated with rehandling one container. The problem decision variables are:

- $x_{c,s,t,w} \in \{0,1\}$, equal to 1 if container c placed in slot s, tier t is assigned to wagon w;

- $y_{c,s,t,w} \in \{0,1\}$, equal to 1 if container c is placed in slot s, tier t when wagon w is assigned (quantities $y_{c,s,t,0}$, equal to 1 if container c is initially placed at tier t of slot s, are known);

- $b_{s,t,w} \in \mathbb{N}$, number of occupied slots above slot s, tier t when wagon w is assigned;

- $v_{s,t,w} \in \mathbb{N}$, number of occupied slots above slot s, tier t when the container in slot s, tier t is assigned to wagon w;

- $u_{s,w} \in \mathbb{N}$, total number of rehandling operations needed when containers in slot s are assigned to wagon w;

- $z_c \in \{0,1\}$, equal to 1 if container c is not assigned (i.e. it is not loaded on a train).

The problem is formalized as follows.

$$\min \sum_{s=1}^{S} \sum_{w=1}^{W} \gamma \cdot u_{s,w} + \sum_{c=1}^{C} \pi_c \cdot z_c \tag{1}$$

s.t.

$$\sum_{s=1}^{S} \sum_{t=1}^{T} \sum_{w=1}^{W} x_{c,s,t,w} + z_c = 1 \qquad c = 1, \ldots, C \tag{2}$$

$$\sum_{c=1}^{C} \lambda_c \cdot \sum_{s=1}^{S} \sum_{t=1}^{T} x_{c,s,t,w} \leq \overline{\lambda}_w \qquad w = 1, \ldots, W \tag{3}$$

$$\sum_{c=1}^{C} \omega_c \cdot \sum_{s=1}^{S} \sum_{t=1}^{T} x_{c,s,t,w} \leq \overline{\omega}_w \qquad w = 1, \ldots, W \tag{4}$$

$$\sum_{c=1}^{C} \sum_{w=W_{r-1}+1}^{W_r} \omega_c \cdot \sum_{s=1}^{S} \sum_{t=1}^{T} x_{c,s,t,w} \leq \overline{\Omega}_r \qquad r = 1, \ldots, R \tag{5}$$

$$y_{c,s,t,w} = y_{c,s,t,w-1} - x_{c,s,t,w} \qquad c = 1, \ldots C, \ s = 1, \ldots, S, \ t = 1, \ldots, T, \ w = 1, \ldots, W \tag{6}$$

$$x_{c,s,t,w} \leq y_{c,s,t,w-1} \qquad c = 1, \ldots C, \ s = 1, \ldots, S, \ t = 1, \ldots, T, \ w = 1, \ldots, W \tag{7}$$

$$b_{s,t,w} = \sum_{c=1}^{C} \sum_{h=t+1}^{T} y_{c,s,h,w-1} \qquad s = 1, \ldots, S, \ t = 1, \ldots, T-1, \ w = 1, \ldots, W \tag{8}$$

$$M\left(1 - \sum_{c=1}^{C} x_{c,s,t,w}\right) + v_{s,t,w} \geq b_{s,t,w} \qquad s = 1, \ldots, S, \ t = 1, \ldots, T, \ w = 1, \ldots, W \tag{9}$$

$$u_{s,w} = \max_{t=1,\ldots,T} v_{s,t,w} \qquad s = 1, \ldots, S, \ w = 1, \ldots, W \tag{10}$$

$$x_{c,s,t,w} \in \{0,1\}, \ y_{c,s,t,w} \in \{0,1\} \qquad c = 1, \ldots C, \ s = 1, \ldots, S, \ t = 1, \ldots, T, \ w = 1, \ldots, W \tag{11}$$

$$b_{s,t,w} \in \mathbb{N}, \ v_{s,t,w} \in \mathbb{N} \qquad s = 1, \ldots, S, \ t = 1, \ldots, T, \ w = 1, \ldots, W \tag{12}$$

$$u_{s,w} \in \mathbb{N} \qquad s = 1, \ldots, S, \ w = 1, \ldots, W \tag{13}$$

$$z_c \in \{0,1\} \qquad c = 1, \ldots, C \tag{14}$$

The cost function (1) includes rehandling costs and penalty costs. Constraints (2) impose that, if container c is not assigned, the corresponding variable z_c is equal to 1. Constraints (3), (4) and (5) concern the limitations due to wagon length and wagon/train weight capacity. Constraints (6) and (7) assure the relations between $x_{c,s,t,w}$ and $y_{c,s,t,w}$ variables. Constraints (8) make variables $b_{s,t,w}$ represent the number of occupied cells above the considered position in the stocking area, whereas constraints (9) assure that, if a container in a certain position s, t is assigned to wagon w, the number of rehandled containers is equal to number of above occupied cells. Finally, the total number of rehandled containers associated with slot s when wagon w is assigned is given by $u_{s,w}$ thanks to constraints (10). Thus, we deal with an integer non linear formulation that can be solved with mathematical programming solvers for instances of small-medium size. We are now investigating both new mathematical formulations and solution procedures (such as ad-hoc heuristic techniques) for solving the train load planning problem.

References

[1] D. Steenken, S. Voss and R. Stahlbock, "Container terminal operation and operations research - a classification and literature review", *OR Spectrum*, 26, 3-49 (2004).

[2] R. Stahlbock and S. Voss, "Operations research at container terminals: a literature update", *OR Spectrum*, 30, 1-52 (2008).

[3] N. Bostel and P. Dejax, "Models and algorithms for container allocation problems on trains in a rapid transshipment shunting yard", *Transportation Science*, 32, 370-379 (1998).

[4] P. Corry and E. Kozan, "An assignment model for dynamic load planning of intermodal trains", *Computers and Operations Research*, 33, 1-17 (2006).

A Path Relinking based Evolutionary Algorithm for Vehicle Routing Problems with Time Windows and Return Products

Anagnostopoulou A.K.

Center for Operations Research & Decision Systems

Department of Management Science & Technology

Athens University of Economics & Business

Repoussis P.P.

Center for Operations Research & Decision Systems

Department of Management Science & Technology

Athens University of Economics & Business

Tarantilis C.D.

Center for Operations Research & Decision Systems

Department of Management Science & Technology

Athens University of Economics & Business

Email: tarantil@aueb.gr

1 Introduction

Managing the flows of spent products has become a crucial concern for companies seeking to explore and integrate reverse logistics as a viable business option. To this end, a major concern is the design of the distribution and collection system for both new and return products respectively. Obviously, inefficient transportation activities can limit the economic success of reprocessing products, while several issues and operational constraints emerge for the collection of used-returned products.

Significant developments has been made towards the design of models and optimization methods to address pick and delivery problems [1, 2]. In this study, the main focus is given on combined distribution-collection systems, where the same vehicle can be used for both deliveries and collection services. Clearly, the utilization of vehicles increases significantly when merging products brought to the customers as well as products brought back to the depot, while the vehicle routing and

scheduling plans are getting more effective. Several models appear in the literature that embodies the essence of dealing with both linehaul and backhaul customers on the same vehicle routes. Among those problem variants with time window restrictions the most well-studied are the so-called Vehicle Routing Problem with Time Windows and Backhauls (VRPBTW) [3] and the Mixed Vehicle Routing Problem with Backhauls and Time Windows (MVRPBTW) [4].

Given a homogeneous fleet of depot-returning capacitated vehicles the goal of the VRPBTW is to design a set of vehicle routes in order to satisfy the delivery and collection requirements of a set of geographically scattered customers. Each customer has a known demand for delivery (linehaul) or collection (backhaul) and it must be serviced within a predefined time window that models the earliest and the latest times during the day that service can take place. Furthermore, each customer must be visited only once by exactly one vehicle, while all linehaul customers of a route must be serviced before the vehicle starts visiting backhaul customers. Contrary, the MVRPBTW assumes that the backhaul customers are not prohibited to be visited before linehaul customers. The primary objective of both problems is to minimize the total number of vehicles required for the service of all customers, while the secondary objective is to minimize the total distance traveled.

Due to their wide applicability and high complexity, the solution aspects of the VRPBTW and the MVRPBTW have generated substantial research [7, 8, 5, 9]. In this paper the main focus is given on the design and development of a novel unified Path-Relinking based Evolutionary Algorithm to address both problems. Computational experiments on benchmark data sets of the literature demonstrate the efficiency and effectiveness of the proposed solution method compared to the current state-of-the-art. Furthermore, new benchmark data sets are generated for the problem variant with simultaneous pick ups and deliveries and time windows, hereafter abbreviated as VRPSPDTW. The latter is first introduced by Angelelli et al. [6].

2   Solution Framework

The proposed solution framework adopts an (μ,λ)-evolution strategy to evolve monotonically a population of individuals [12, 13, 14]. Initiating from a population of μ adequately diversified individuals, at each generation a new intermediate population of λ individuals is produced via a novel Path Relinking (PR)[10] recombination mechanism. The μ survivors from the union of parents and offspring are selected deterministically to form the next generation population.

PR is used to generate trajectories-paths between between two parent solutions from the current population. For this purpose, several semi-probabilistic strategies are used to select the initial and guiding solutions, while the mechanism for generating the trajectories adopts simple edge-exchange neighborhood structures. The main effort is to introduce gradually features and attributes appearing in the guiding the solution. As such, the trajectories followed are always heading towards

solutions with reduced Hamming distance w.r.t the guiding solution. To this end, controlled tunneling is also allowed, assuming that capacity and time window constraints are relaxed.

On the other hand, a subset of promising intermediate solutions (offspring) is selected and further improved via a route elimination procedure and a memory-based trajectory local search algorithm. Both algorithms utilize the basic Tabu Search (TS) framework along with long term memory structures to drive the search process . The primary objective of the former is to reduce the total number of vehicles using a tailored lexicographic ordering evaluation function for the exploration of the neighboring space [12], while the latter explores the solution space on the basis of a Guided Local Search (GLS) [11] algorithm in an effort to reduce the total distance traveled.

3 Computational Experiments

The benchmark data sets found in [3], [5], [4] and [6] are used as the baseline for evaluating the proposed solution approach, abbreviated hereafter as PREA. Furthermore, new benchmark data sets are also developed for the VRPSPDTW. Table 1 summarizes the average results obtained (MNV, MTD and MCT stands for mean number of vehicles, mean traveling distance and mean computational time respectively). Compared to the current state-of-the-art, PREA proved to be highly efficient and competitive, producing several new best solutions and improving the best reported average results with reasonable computational time requirements.

Table 1: Computational Results

Problem Variant	Author / Data Set	Best Known		PREA		
		MNV	MTD	MNV	MTD	MTC (sec)
VRPBTW	Gelinas et al. [3] / (R1-100)	17.07	1559.01	17.20	1569.95	165
VRPBTW	Thangiah et al. [5] / (250)	36.92	4555.33	36.92	4549.23	510
VRPBTW	Thangiah et al. [5] / (500)	56.08	6907.38	55.75	6896.62	1890
MVRPBTW	Kontoravdis and Bard [4] (MR-100)	4.00	902.43	4.00	901.75	500
MVRPBTW	Kontoravdis and Bard [4] (MRC-100)	4.13	1122.83	4.13	1103.50	500
VRPSPDTW	New (R1)	12.33	1276.46	12.33	1276.46	570
VRPSPDTW	New (RC1)	12.00	1425.09	12.00	1425.09	597
VRPSPDTW	New (R2)	2.82	1040.78	2.82	1040.78	605
VRPSPDTW	New (RC2)	3.25	1361.97	3.25	1361.97	622
VRPSPDTW	Angelelli and Mansini [6] (R1)	4.5	3863.75	3.83	3976.83	30
VRPSPDTW	Angelelli and Mansini [6] (RC1)	4.33	2807.56	4.33	2806.11	75
VRPSPDTW	Angelelli and Mansini [6] (C1)	5.00	4338.37	5.00	4338.38	60

References

[1] S. Parragh, K.F. Doerner and R.F. Hartl, "A survey on Pickup and Delivery Problems Part I: Transportation Between Customers and Depot", *Journal für Betriebswirtschaft* 58(1), 21-51

(2008a).

[2] S. Parragh, K.F. Doerner and R.F. Hartl, "A survey on Pickup and Delivery Problems Part II: Transportation Between Pickup and Delivery Locations", *Journal für Betriebswirtschaft* 58(2), 81-117 (2008b).

[3] S. Gelinas, M. Desrochers, J. Desrosiers and M. Solomon, "A New Branching Strategy for Time Constrained Routing Problems with Application to Backhauling", *Annals of Operations Research* 61, 91-100 (1995).

[4] G. Kontoravdis and J. F. Bard, "A GRASP for the vehicle routing problem with time windows", *ORSA Journal on Computing* 7, 10-23 (1995).

[5] S. Thangiah, J. Potvin and T. Sun, "Heuristic approaches to Vehicle Routing with Backhauls and Time Windows", *Computers and Operations Research* 23, 1043-105 (1996).

[6] E. Angelelli and R. Mansini, "The Vehicle Routing Problem with Time Windows and Simultaneous Pick-up and Delivery", In *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, Klose A., Speranza G. and Van Wassenhove L.V. (eds), 249-267, Springer-Verlag, 2002.

[7] S. Ropke and D. Pisinger, "A unified heuristic for a large class of Vehicle Routing Problems with Backhauls", *European Journal of Operation Research* 171, 750-775 (2006).

[8] Y. Zhong and M. Cole, "A Vehicle Routing Problem with Backhauls and Time Windows: a guided local search solution", *Transportation Research Part E* 41, 131-144 (2005).

[9] M. Reimann, K. Doerner, R. Harlt, "Insertion based ants for Vehicle Routing Problems with Backhauls and Time Windows", *Lecture Notes in Computer Science* 2463, 135-148 (2002).

[10] F. Glover, "Scatter Search and Path Relinking", In *New Ideas in Optimisation*, D. Corne, M. Dorigo and F. Glover (eds.), Wiley, (1999).

[11] C. Voudouris and E. Tsang,"Guided Local Search", *European Journal of Operational Research* 113, 80-119 (1998).

[12] P.P. Repoussis, C. D. Tarantilis, and G. Ioannou, "An Arc-Guided Evolutionary Algorithm for the Vehicle Routing Problem with Time Windows". *IEEE Transactions on Evolutionary Computation* 13(3), 624-647 (2009)

[13] C.D. Tarantilis, "Solving the vehicle routing problem with adaptive memory programming methodology", *Computers and Operations Research* 32, 2309-2327 (2005).

[14] C.D. Tarantilis and C.T. Kiranoudis, "Boneroute: An adaptive memory-based method for effective fleet management", *Annals of Operations Research* 115, 227-241 (2002).

A New Decomposition Approach for a Liquefied Natural Gas Inventory Routing Problem

Henrik Andersson, **Marielle Christiansen**

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, Trondheim, Norway

Email: henrik.andersson@iot.ntnu.no

Guy Desaulniers

Department of Mathematics and Industrial Engineering

École Polytechnique, Montréal, Canada

1   Introduction

The problem analyzed in this paper is a combined inventory management, ship routing and scheduling problem describing the distribution of LNG from the liquefaction plants where the natural gas is cooled down to its liquid state, via the LNG carriers to the regasification terminals where the LNG is stored, reheated, and fed into pipeline systems to serve the market. The problem can be classified as a liquefied natural gas inventory routing problem (LNG-IRP), a class of problems that has not been much studied earlier. The LNG-IRP was defined in [2], where two different formulations are described and compared. A branch-and-price-and-cut algorithm for the problem is developed in [3]. The algorithm uses a path based formulation where movements and operations of the ships are described by schedules covering the whole planning horizon. Valid inequalities derived from the minimum and maximum number of visits to a port are used to strengthen the continuous relaxation.

The purpose of this paper is to describe a new decomposition approach for an LNG-IRP. Instead of describing the movements and operations through schedules, duties are used. A duty covers a smaller part of the planning horizon, and a sequence of duties now represents the movements and operations. A branch-and-price-and-cut algorithm has been developed. Advanced branching schemes and stronger valid inequalities have been implemented.

The outline of the paper is as follows. In Section 2 the problem is presented and the assumptions made are discussed. The mathematical model is given in Section 3, while the solution approach is described in Section 4. Section 5 covers computational results.

2 Problem description

The problem considered is the routing and scheduling of LNG ships between liquefaction plants and regasification terminals, as well as the production planning and inventory management at the plants and sales planning and inventory management at the terminals. Each liquefaction plant has a limited inventory capacity given by an upper and a lower bound. The production rate is variable and can vary between upper and lower limits. There is a per unit cost of producing LNG, but no holding costs are incurred. By each plant there is a port with a limited berth capacity where the LNG ships can load. The loading operation is assumed to take one day, and the ships are always fully loaded. The fleet of ships is heterogeneous, both considering the capacities and the initial positions. All ships contain a number of tanks and, when a tank is unloaded, it must be fully unloaded. This means that the ship can do partial unloading at a regasification terminal, but an integer number of tanks must be unloaded. A fixed amount of LNG evaporates from the tanks each day. In order to avoid costly operations of recooling the tanks of the ships, the tanks are not allowed to be completely empty. This means that a fraction of LNG must be left in the tanks after unloading, so that the ships have time to sail to a liquefaction plant and start loading before the tanks are empty. Only operating costs related to the number of sailing days are considered, other costs are assumed fixed and cannot be affected by the decisions made. Each regasification plant also has a limited inventory capacity, and the sales rate is regulated by upper and lower limits. Revenues from selling LNG at the regasification terminals are part of the model, but no holding costs are incurred. The objective is to maximize the total profit consisting of revenue from selling LNG and costs of operating the ships and producing LNG.

3 Mathematical formulation

Define the time horizon $\mathcal{T}$, the set of ships $\mathcal{V}$, the set of liquefaction plants $\mathcal{P}$, and the set of regasification terminals $\mathcal{D}$. For each ship v, a set of duties $\mathcal{R}_v$ is defined. A duty consists of a geographical route specifying the ports visited, a schedule specifying when each port is visited and an unloading plan specifying the number of tanks and the amount of LNG unloaded at each visit at a regasification terminal. All duties start at a liquefaction plant where the ship loads, then one or two regasification terminals are visited before the ship sails to a liquefaction plant to reload. In the model, Z_{vrpt}^P is 1 if duty r for ship v starts in time period t at production port p and 0 otherwise, $Z_{vrd\tau t}^D$ is 1 if duty r for ship v starting in time period τ unloads at delivery port d in time period t, and $Z_{vrp\tau t}^P$ 1 if duty r for ship v starting in time period τ ends with a visit at production port p in time period t.

The loading quantities corresponding to Z_{vrpt}^P and the unloading quantities and number of tanks corresponding to $Z_{vrd\tau t}^D$ are denoted Q_{vrpt}^P, $Q_{vrd\tau t}^D$, and $H_{vrd\tau t}^D$ respectively.

For each liquefaction plant p and time period t upper and lower bounds on production $\overline{P}_{pt}^P$, $\underline{P}_{pt}^P$ and inventory $\overline{I}_{pt}^P$, $\underline{I}_{pt}^P$ as well as berth capacities $\overline{B}_{pt}^P$ are defined. The corresponding parameters for a regasification terminal d are: $\overline{S}_{dt}^D$, $\underline{S}_{dt}^D$ for the sales limits, $\overline{I}_d^D$, $\underline{I}_d^D$ for the inventory limits and $\overline{B}_{dt}^D$ for the berth capacities.

The unit cost and revenue for producing and selling LNG are location and time dependent. The cost of producing one unit of LNG at liquefaction plant p in time period t is C_{pt}^P, while the revenue of selling one unit of LNG at regasification terminal d in time period t is R_{dt}^D. The operating cost of duty r sailed by ship v is C_{vr}^R. For each ship v, F_{vpt}^V indicates the initial position and the time when the ship is available and is 1 if ship v is first available at liquefaction plant p in time period t. $\overline{H}_v$ denotes the number of tanks of ship v.

For liquefaction plant p, let i_{pt}^P and p_{pt}^P denote the inventory level and the production in time period t, respectively. i_{dt}^D and s_{dt}^D are used for the inventory level and sales at regasification terminal d in time period t, and x_{vrt} is the fraction of duty r starting in time period t that ship v sails.

$$\max \sum_{d \in \mathcal{D}} \sum_{t \in \mathcal{T}} R_{dt}^D s_{dt}^D - \sum_{p \in \mathcal{P}} \sum_{t \in \mathcal{T}} C_{pt}^P p_{pt}^P - \sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} \sum_{t \in \mathcal{T}} C_{vr}^R x_{vrt} \tag{1}$$

$$i_{pt}^P - p_{pt}^P - i_{pt-1}^P + \sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} Q_{vrpt}^P x_{vrt} = 0 \qquad p \in \mathcal{P}, t \in \mathcal{T} \tag{2}$$

$$i_{dt}^D + s_{dt}^D - i_{dt-1}^D - \sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} \sum_{\tau \in \mathcal{T}} Q_{vrd\tau t}^D x_{vr\tau} = 0 \qquad d \in \mathcal{D}, t \in \mathcal{T} \tag{3}$$

$$\sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} Z_{vrpt}^P x_{vrt} \le \overline{B}_{pt}^P \qquad p \in \mathcal{P}, t \in \mathcal{T} \tag{4}$$

$$\sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} \sum_{\tau \in \mathcal{T}} Z_{rd\tau t}^D x_{vr\tau} \le \overline{B}_{dt}^D \qquad d \in \mathcal{D}, t \in \mathcal{T} \tag{5}$$

$$\sum_{r \in \mathcal{R}_v} Z_{vrpt}^P x_{vrt} - \sum_{r \in \mathcal{R}_v} \sum_{\tau \in \mathcal{T}} Z_{vrp\tau t}^P x_{vr\tau} = F_{vpt}^V \qquad v \in \mathcal{V}, p \in \mathcal{P}, t \in \mathcal{T} \tag{6}$$

$$\sum_{r \in \mathcal{R}_v} \sum_{\tau \in \mathcal{T}} H_{vrd\tau t}^D x_{vr\tau} \in \{0, \ldots, \overline{H}_v\} \qquad v \in \mathcal{V}, d \in \mathcal{D}, t \in \mathcal{T} \tag{7}$$

$$\sum_{r \in \mathcal{R}_v} Z_{vrpt}^P x_{vrt} \in \{0, 1\} \qquad v \in \mathcal{V}, p \in \mathcal{P}, t \in \mathcal{T} \tag{8}$$

$$\sum_{r \in \mathcal{R}_v} \sum_{\tau \in \mathcal{T}} Z_{rd\tau t}^D x_{vr\tau} \in \{0, 1\} \qquad v \in \mathcal{V}, d \in \mathcal{D}, t \in \mathcal{T} \tag{9}$$

$$\underline{I}_p^P \le i_{pt}^P \le \overline{I}_p^P \qquad p \in \mathcal{P}, t \in \mathcal{T} \tag{10}$$

$$\underline{I}_d^D \le i_{dt}^D \le \overline{I}_d^D \qquad d \in \mathcal{D}, t \in \mathcal{T} \tag{11}$$

$$\underline{P}_{pt}^P \le p_{pt}^P \le \overline{P}_{pt}^P \qquad p \in \mathcal{P}, t \in \mathcal{T} \tag{12}$$

$$\underline{S}_{dt}^D \le s_{dt}^D \le \overline{S}_{dt}^D \qquad d \in \mathcal{D}, t \in \mathcal{T} \tag{13}$$

$$x_{vrt} \ge 0 \qquad v \in \mathcal{V}, r \in \mathcal{R}_v, t \in \mathcal{T} \tag{14}$$

The objective function (1) maximizes the total profit consisting of the revenue from selling

and cost of producing LNG and the operating costs of the ships. Constraints (2) and (3) are the inventory balance constraints at the liquefaction plants and regasification terminals, respectively. The number of ships that can load and unload at the same time at a plant or terminal is restricted by constraints (4) and (5). Constraints (6) are the duty balance constraints. They state that the number of duties starting at a liquefaction plant a given day is equal to the number of duties ending at the same plant the same day. The right hand side is 1 for the liquefaction plant and time period where the ship is first available, making sure that the ship starts a duty from that plant the given day. The integrality restrictions on the number of tanks unloaded are stated in constraints (7), and the binary restrictions on the number of visits are imposed by (8) and (9). The upper and lower bounds on the continuous variables are given in constraints (10) to (13) and (14) are the non-negativity constraints.

4 Solution approach

A branch-and-price-and-cut algorithm has been developed for the problem. The subproblems are fairly easy compared to formulations where the movements and operations of the ships are described by schedules covering the whole planning horizon, and are solved using an enumeration algorithm. Many different branching entities are used, and the connection between the branching entities and the possibility to generate strong valid inequalities is utilized. Known valid inequalities have been generalized to better exploit the structure of the problem and are added during the branch-and-bound search.

5 Computational results

Computational results comparing this decomposition approach to earlier formulations will be presented.

References

[1] EIA, "International energy outlook 2009", Tech. rep., U.S. Department of Energy, Energy Information Administration U.S. Department of Energy Washington, DC 20585, 2009.

[2] R. Grønhaug, M. Christiansen, "Supply chain optimization for the liquefied natural gas business", in: *Innovations in Distribution Logistics*, L. Bertazzi, J. v. Nunen, M. Speranza (eds), 195-218, Springer Berlin Heidelberg, 2009.

[3] R. Grønhaug, M. Christiansen, G. Desaulniers, and J. Desrosiers, "A branch-and-price-and-cut method for a liquefied natural gas inventory routing problem", Working paper G-2008-49, GERAD, Montréal, 2008.

The Dynamic Traveling Purchaser Problem
with Deterministic Quantity:
A Branch and Cut Approach

E. Angelelli

Dept. of Quantitative Methods,

University of Brescia, Italy

M. Gendreau

CIRRELT and MAGI

École Polytechnique de Montréal, Canada

R. Mansini

Dept. of Electronics for Automation

University of Brescia, Italy

M. Vindigni

Dept. of Quantitative Methods,

University of Brescia, Italy

1   Introduction

Let us consider a set of products $K := \{1, \cdots, n\}$ and a set of markets $M := \{1, \cdots, m\}$ plus a depot indexed as 0. For each product k a positive discrete demand d_k is specified. Each product k can be purchased in a subset of markets $M_k \subseteq M$ at a unit price f_{ki} depending on the market i. For each market i, $i \in M_k$, q_{ki} units of product k are offered such that $\sum_{i \in M_k} q_{ki} \geq d_k$ for all $k \in K$. For each pair i, j of markets and for each market and the depot, a traveling cost c_{ij} is known. The Traveling Purchaser Problem (TPP) can be defined on an undirected graph $G = (V, E)$, where $V := M \cup \{0\}$ is the vertex set and $E := \{(i, j) : i, j \in V, \ i < j\}$ is the edge set. The problem looks for a simple tour in G starting at and ending to the depot by visiting a set of markets so that the demand for all products is satisfied at the minimum routing and purchasing costs. The problem is known to be NP-hard in the strong sense and several solution algorithms, both heuristic and exact, have been proposed to solve it (see [6] and [7] and references therein). Moreover, the problem find relevant applications either in routing or in scheduling and warehousing contexts (see [3], [4], [5]).

In practical contexts, the markets will receive requests from several different purchasers operating in the system, so that the quantity one might expect to find in a market is not constant but depends on the time the market will be visited. To better evaluate how this would affect the complexity of the problem, we assume here a simple model for the consumption process, where the quantity made available for each product in each market q_{ki}, $k \in K$ and $i \in M_k$, may linearly reduce over time according to a consumption rate $\alpha_{ki} \geq 0$ depending on the market and on the

product.

Thus this problem is dynamic, since data change over time, and is deterministic, as the changes can be foreseen. The underlying assumption is that the decision maker has information about the future since he exactly knows how quantities will decrease, i.e. the consumption function is a known deterministic function. We call this problem the Dynamic Traveling Purchaser Problem with Deterministic Quantity (DTPPDQ). To the best of our knowledge, this version of the TPP has never been addressed in the literature before now. We propose a Branch and Cut approach to solve it.

The TPP has been analyzed in a more general dynamic context in the introductory paper [1] where heuristic approaches are proposed. Finally, the dynamic TPP with stochastic quantities can be found in [2] where different operating scenarios characterized by the presence of a planner who makes decisions and an executor (the traveling purchaser) who is involved with the service in practice are considered. Each scenario is faced assuming two different type of planners, a stochastic planner and a deterministic one. The stochastic planner does not know future products availability, but he does have probabilistic information regarding future consumption events (stochastic model). The deterministic planner approximates the stochastic model by using a deterministic consumption function (deterministic model).

Our exact approach provides a tool to compute the optimal solutions for the deterministic model used in [2].

2   The mathematical formulation

Let x_{ij} be a binary variable equal to 1 if edge $(i,j) \in E$ is selected in the optimal tour and 0 otherwise. We define as y_i a binary variable equal to 1 if market $i, i \in M$, is selected and zero otherwise, while z_{ki} is the integer variable representing the quantity of product $k, k \in K$, purchased at market $i, i \in M_k$. Variable $t_i, i \in M$, represents the time at which market i will be visited. Finally, variables $\lambda_{ki}, k \in K, i \in M$ model products exhaustion in the markets. More precisely, λ_{ki} takes value 1 if the product k is depleted in market i and 0 otherwise.

The DTPPDQ can be formulated as an integer linear-programming problem where $t_{ij}, (i,j) \in E$, represents the traveling time associated to edge (i,j):

$$\min \sum_{(ij)\in E} c_{ij}x_{ij} + \sum_{k\in K}\sum_{i\in M_k} f_{ki}z_{ki} \tag{1}$$

$$\sum_{j\in V\setminus\{i\}} x_{ij} = y_i \qquad i \in V, \tag{2}$$

$$\sum_{i\in V\setminus\{j\}} x_{ij} = y_j \qquad j \in V, \tag{3}$$

$$\sum_{i \in M_k} z_{ki} = d_k \qquad k \in K, \tag{4}$$

$$z_{ki} \leq q_{ki} y_i \qquad k \in K, \ i \in M_k, \tag{5}$$

$$t_i + t_{ij} \leq t_j + A(1 - x_{ij}) \quad i \in V, j \in M \tag{6}$$

$$z_{ki} \leq q_{ki}(1 - \lambda_{ki}) \quad k \in K, i \in M_k \tag{7}$$

$$z_{ki} \leq q_{ki} - \alpha_{ki} t_i + \alpha_{ki} A \lambda_{ki} \quad k \in K, i \in M_k \tag{8}$$

$$t_0 = 0, y_0 = 1 \tag{9}$$

$$x_{ij} \in \{0,1\} \quad (i,j) \in E \tag{10}$$

$$y_i \in \{0,1\} \quad i \in M \tag{11}$$

$$t_i \geq 0 \quad i \in M \tag{12}$$

$$z_{ki} \geq 0 \text{ integer} \quad k \in K, i \in M_k \tag{13}$$

$$\lambda_{ki} \in \{0,1\} \quad k \in K, i \in M_k \tag{14}$$

Objective function represents the minimization of the total traveling and purchasing costs. Constraints (2) and (3) are assignment constraints and impose that we enter and leave each visited market exactly once. Equalities (4) guarantee that an amount equal to d_k is purchased for each product $k \in K$ and inequalities (5) mean that a product k can be purchased at market i only if the market is visited in the optimal solution and for an amount not larger than q_{ki}. Set of constraints (6) models arrival times and imposes that if we travel from market i to market j (i.e. $x_{ij} = 1$) then $t_j \geq t_i + t_{ij}$ whereas if $x_{ij} = 0$ the constraint is always satisfied since A is a sufficiently large constant measuring an upper bound on the optimal total traveling time. For each market i and each product k, constraint (7) forces z_{ki} to 0 if the supply is depleted, whereas constraints (8) model how quantities decrease. More precisely, constraints (8) force λ_{ki} to 1 if t_i is too large; otherwise, λ_{ki} can be 0 and then z_{ki} is bounded by $q_{ki} - \alpha_{ki} t_i$. Finally, equations (9) impose that the depot is always visited and its visiting time is equal to 0. Constraints (10)–(14) are non negativity, integer and binary conditions.

3 Branch and Cut algorithm

We propose a branch and cut algorithm to solve the problem. Different valid inequalities and branching rules have already been worked out.

Experimental analysis is in progress. Since problem is new we provide a large set of benchmark instances to test the efficiency of the proposed exact approach. At present, we have already run some tests to evaluate the effectiveness of alternative branching rules and of different cuts when directly added to the initial formulation. In these preliminary results bare our approach

outperforms bare CPLEX, which already runs out of memory with instances with more than 10 markets.

References

[1] E. Angelelli, R. Mansini and M. Vindigni (2009) "Exploring greedy criteria for the dynamic traveling purchaser problem", *CEJOR* 17, 141–158 (2009).

[2] E. Angelelli, R. Mansini and M. Vindigni, "The Dynamic Traveling Purchaser Problem with Stochastic Quantity", Working Paper Department of Electronics for Automation, University of Brescia, 2009.

[3] R. M. Burstall, "A Heuristic Method for a Job Sequencing Problem", *Operational Research Quarterly* 17, 291–304 (1966).

[4] J. A. Buzacott and S.K. Dutta, "Sequencing Many Jobs on a Multipurpose Facility", *Naval Research Logistics Quarterly* 18, 75–82 (1971).

[5] B. Golden B., L. Levy, and R. Dahl, Two Generalizations of the Traveling Salesman Problem, *Omega* 9, 439–455 (1981).

[6] G. Laporte, J. Riera-Ledesma, J. J. Salazar-Gonzalez, "A Branch-and-Cut Algorithm for the Undirected Traveling Purchaser Problem", *Operations Research* 6, 940-951 (2003).

[7] J. Riera-Ledesma and J. J. Salazar-Gonzalez, "A heuristic approach for the Traveling Purchaser Problem", *European Journal of Operational Research* 162, 142–152 (2005).

Aggregate planning in general freight intermodal transportation networks

Davide Anghinolfi, Massimo Paolucci, Simona Sacone, Silvia Siri

Department of Communication, Computer and Systems Science

University of Genova, Italy

1 Introduction

Freight intermodal transportation may be defined as the transportation of goods involving at least two modes of transport, where long distances are generally covered by rail, inland waterway or vessels, and short distances (initial and final journeys) are covered by road [1]. As highlighted in the survey [2], the different problems faced in intermodal freight transport can be classified according to the division among drayage operators (planning and scheduling trucks), terminal operators (planning transhipment operations at terminals), network operators (planning the infrastructure and the transport organization) and, finally, intermodal operators (planning the route for shipments in the intermodal network). The present work can be classified in the forth category, i.e. the planning of shipment transportation over a certain intermodal network. This is the problem that each single freight forwarder has to face whenever he has to organize the transportation for his customers buying the services offered by drayage, terminal and network operators.

The objective of this work is making an aggregate medium-term planning in a general freight intermodal network involving different transportation modes, such as road, rail and sea. The transportation demand (made of customer orders or forecasts) is known in terms of origin, destination, quantity (in TEU) and time conditions. The planning problem consists in determining how to satisfy this transportation demand over the considered network and with the available transportation resources, in order to minimize the overall transportation cost and the delay in delivery of orders. The decisions will concern the number of products moved by a specific transportation resource on each network link in each time period. This kind of problem could be considered as an intermediate approach between a service network design problem (SNDP) and an assignment problem. A generic SNDP considers a transportation demand expressed as a set of commodities with different origins and destinations and the decision variables indicate the type of services to be used and the flows on the network [3]. Our problem is similar as it determines the flows in the network, even

though our objective is not the design of the network but the determination of how to optimally use the network and the transportation resources provided. In this sense, our problem is not an assignment problem since the decisions concern flow variables (and not the specific assignment of each container of an order to a transportation resource). Moreover, with respect to SNDP, we consider some time constrains for orders, i.e. release times and deadlines, as well as availability constraints for transportation resources, i.e. fixed schedules. A similar approach can be found in [4] where only rail-road services are considered and a time-space network is used to represent the intermodal operations over the time horizon. Moreover, in [5] an intermodal network is considered (involving road, rail and maritime links) in which a heuristic multi-objective approach is presented for both the tactical and the operational planning phase.

The proposed planning procedure is composed of two phases:

1. *Path evaluation phase*: the objective is to determine all the possible paths (and also the associated transport resources) available to serve a given order.

2. *Planning phase*: the objective is to determine the quantities to be transported on each arc of the network by a specific transportation resource in each time instant (note that split delivery is allowed for serving orders).

The *path evaluation phase* consists of two steps. *(a)* The subset $\mathcal{AP}(o)$ of available paths is chosen to serve a given order o (between the corresponding origin and destination, respectively denoted as s_o and e_o) on the basis of topological criteria. Specifically, this can be done in two ways: the available paths in $\mathcal{AP}(o)$ are chosen directly by the freight forwarder or they are computed as the k-shortest paths from s_o to e_o for a given value k, having assigned fixed traveling times for the network arcs. *(b)* In the second step, for each order o the temporal feasibility of the available paths in $\mathcal{AP}(o)$ is evaluated assuming that infinite transportation resources are available, but taking into account the schedules for the involved transportation modes, with a procedure similar to the well-known one for the critical path computation in project management. The output of this step is the subset $\mathcal{P}(o)$ of paths which are both temporally and topologically feasible for order o, since they are those paths in $\mathcal{AP}(o)$ for which the earliest delivery time is not greater than the order deadline. This step also selects, for each link in a feasible path associated with a transportation schedule, the subset of feasible departures in the schedule. Then, the *planning phase* corresponds to the solution of an integer programming problem described in the following.

2 Problem formulation

We consider a general intermodal transportation network modeled as a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ where $\mathcal{V}$ is a set of locations and the arcs $a \in \mathcal{A}$ correspond to transportation operations allowing

to move freight between two locations by a specific mode of transport. Locations can be associated with origins and destinations of orders, exchange nodes, where freight can change the mean of transport, or finally intermodal nodes, where freight can change the mode of transport. The set $\mathcal{A}$ can include two subsets of arcs, i.e. $\mathcal{A}^C$ gathering capacity constrained arcs and $\mathcal{A}^S$ gathering capacity and schedule constrained arcs. The arcs in $\mathcal{A}^C$ are associated with road transportation, whereas the ones in $\mathcal{A}^S$ with rail or ship transportation.

For each order $o \in \mathcal{O}$ we assume to know the origin and destination nodes, respectively denoted s_o and e_o, the number of containers q_o (expressed in TEU) to be transported, the release time r_o, i.e., the earliest time from which the containers of o can depart from s_o, the deadline dl_o, i.e., the maximum allowed time by which all the containers of o must reach e_o, and the due date dd_o, i.e., the time by which the containers should reach e_o without incurring in penalty. We consider a planning horizon of T time steps and we associate the planning decisions with each time step in $t = 0, \ldots, T$. The set $\mathcal{A}^C$ can be further partitioned into subsets denoted as $\mathcal{A}^C_g$, with $g = 1, \ldots, G$, depending on the way the transportation resources are associated with arcs $a \in \mathcal{A}^C$; for example, we can group together in $\mathcal{A}^C_g$ the arcs in $\mathcal{A}^C$ that are served by a same pool of vehicles of a given transportation company. Then, we assign a transportation capacity $R_{g,t}$ (expressed in number of TEU) to each set of arcs $\mathcal{A}^C_g$ which represents the maximum number of TEU that can be transported over all the arcs $a \in \mathcal{A}^C_g$ in the time step t. The arcs in $\mathcal{A}^S$ represent transportation modes for which both a departure schedule and an available capacity is given (this is usually the case of rail or ship transportation). Therefore, for each $a \in \mathcal{A}^S$ a departure schedule $\mathcal{S}_a$ is given, consisting of a set of pairs $(D_{a,v}, R_{a,v})$ representing, for each scheduled vehicle $v = 1, \ldots, V_a$ in the time horizon, the departure time $D_{a,v}$ and the transportation capacity in TEU $R_{a,v}$.

As already described, the outputs of the *path evaluation phase* are the feasible temporal and topological transportation alternatives for each order o which are defined in a set of paths $\mathcal{P}(o)$. A path p included in $\mathcal{P}(o)$ consists of a sequence of n_p arcs, each denoted with the pair (p,l), $l = 0, \ldots n_p - 1$. We use the mapping $\alpha(p,l) = a \in \mathcal{A}$ to identify the l-th arc in path p. For each arc $\alpha(p,l) \in \mathcal{A}^S$ such that $(p,l) \in \mathcal{P}(o)$, the *path evaluation phase* also determines the non empty subset of suitable departure schedule denoted as $\mathcal{S}_{(p,l)} \subseteq \mathcal{S}_a$. Moreover, $TT(p,l)$ indicates the transportation time needed to cover the arc $\alpha(p,l) \in \mathcal{A}$ (note that these time values account also for loading and unloading operations at nodes).

The planning model can be formulated as an integer programming (IP) problem introducing the integer variables $x_{o,(p,l),t}$ for each order o, $(p,l) \in \mathcal{P}(o)$, $t = 0, \ldots, T$, which indicate the number of TEU of order o transported on arc $a = \alpha(p,l)$ and departed from the tail of a in time step t. The objective function to be minimized considers transportation costs and tardiness costs, being γ_o the tardiness cost for order o and $\delta_{(p,l)}$ the unitary transportation cost on arc (p,l). The planning problem is formulated as follows.

$$\min \sum_{o \in \mathcal{O}} \sum_{t=dd_o+1}^{T} \frac{\gamma_o}{q_o} \cdot x_{o,(p,n_p-1),t-TT(p,n_p-1)} \cdot (t - dd_o) + \sum_{o \in \mathcal{O}} \sum_{(p,l) \in \mathcal{P}(o)} \sum_{t=0}^{T} \delta_{(p,l)} \cdot x_{o,(p,l),t} \quad (1)$$

s.t.

$$\sum_{(p,0) \in \mathcal{P}(o)} \sum_{t=r_o}^{dl_o} x_{o,(p,0),t} = q_o \qquad o \in \mathcal{O} \quad (2)$$

$$x_{o,(p,l),t} = x_{o,(p,l-1),t-TT(p,l-1)}$$
$$o \in \mathcal{O} \quad (p,l) \in \mathcal{P}(o) \quad l = 1,\dots,n_p-1 \quad \alpha(p,l) \in \mathcal{A}^C \quad t = r_o+1,\dots,T \quad (3)$$

$$x_{o,(p,l),D_{a,v}} = \sum_{z=D_{a,v-1}+1}^{D_{a,v}} x_{o,(p,l-1),z-TT(p,l-1)}$$
$$o \in \mathcal{O} \quad (p,l) \in \mathcal{P}(o) \quad l = 1,\dots,n_p-1 \quad \alpha(p,l) \in \mathcal{A}^S \quad v = 1,\dots,V_{\alpha(p,l)} \quad (4)$$

$$\sum_{o \in \mathcal{O}} \sum_{(p,l) \in \mathcal{P}(o):\alpha(p,l) \in \mathcal{A}_g^C} x_{o,(p,l),t} \le R_{g,t} \qquad g = 1,\dots,G \quad t = r_o+1,\dots,T \quad (5)$$

$$\sum_{o \in \mathcal{O}} \sum_{(p,l) \in \mathcal{P}(o):\alpha(p,l)=a} x_{o,(p,l),D_{a,v}} \le R_{a,v} \qquad a \in \mathcal{A}^S \quad v = 1,\dots,V_a \quad (6)$$

Constraints (2) assure the demand satisfaction, whereas (3) and (4) impose the flow conservation, for arcs in $\mathcal{A}^C$ and in $\mathcal{A}^S$ (matching schedule), respectively. Finally, (5) and (6) are capacity constraints for arcs in $\mathcal{A}^C$ and in $\mathcal{A}^S$, respectively. This IP model can be optimally solved only when the instance dimensions (i.e. the number of orders, paths, the length of paths and time horizon) are limited; otherwise, a good sub-optimal solution can be generated by means of an approximation procedure that iteratively solves a sequence of linearly relaxed problems with additional constraints.

References

[1] T.G. Crainic and K.H. Kim, "Intermodal transportation", in *Transportation*, C. Barnhart and G. Laporte (eds), 467-537, North-Holland, 2007.

[2] C. Macharis and Y.M. Bontekoning, "Opportunities for OR in intermodal freight transport research: A review", *European Journal of Operational Research* 153, 400-416 (2004).

[3] N. Wieberneit, "Service network design for freight transportation: a review", *OR Spectrum*, 30, 77-112 (2008).

[4] L.K. Nozick and E.K. Morlok, "A model for medum-term operations planning in an intermodal rail-truck service", *Transportation Research A* 31, 91-107 (1997).

[5] M. Caramia and F. Guerriero, "A heuristic approach to long-haul freight transportation with multiple objective functions", *OR Spectrum*, 30, 77-112 (2008).

The Uncapacitated Dial-a-Ride Problem on a Tree

Shoshana Anily

Email: anily@post.tau.ac.il

and

Aharona Pfeffer

Faculty of Management

The Leon Recanati Graduate School of Business Administration

Tel-Aviv University, Israel

March 4, 2010

1   Introduction and Literature Review

The uncapacitated *Dial-a-Ride Problem* (DARP) on a tree, is defined by a single vehicle of unlimited capacity, a set of object types $O = \{0, 1, \ldots, m\}$ where 0 is a null object, and a tree $T = (V, E)$, where $V = \{1, \ldots, n\}$ is the set of vertices and E is the set of edges. Each vertex v is associated with a pair of object types $(s(v), d(v))$, where $s(v)$ $(d(v))$ is the object type supplied (required) by v. The total supply (demand) of each object type $i \neq 0$ is one unit. In addition, each edge $(u, v) \in E$ is associated with a cost $c((u, v)) \geq 0$. The objective is to design a minimum cost feasible route that starts and ends at the root of the tree, so that the vehicle while following the route satisfies the requirements of all the vertices. This problem is known to be NP-hard, see [7].

The DARP is a special case of the *Swapping Problem* (SP) that was first introduced by Anily and Hassin [1]. In the SP it is possible to have several units of the same object type, where in the DARP it is limited to one unit. From that aspect, the DARP is simpler as the destination of each object type is well defined. Both, the SP and DARP were studied extensively, especially for capacity one. In the capacitated versions of the problems there is a distinction between *preemptive* objects that can be unloaded temporarily at intermediate vertices and *non-preemptive* objects that must be shipped directly to their destination. This distinction is not necessary in the uncapacitated case as preemption is used only for freeing space in the vehicle. Applications of variants of the SP arise in the optimization of robot arm movements, see [5], and printed circuit board assembly, see [6]. The *Stacker Crane Problem* (SCP) proposed by Frederickson [10], is equivalent to the DARP with capacity one and all objects being non-preemptive.

It is easy to show that the unit capacity SP and DARP on general graphs are NP-hard. In our literature review we focus on graphs that are trees and carriers of unit or unlimited capacity: Attalah and Kosaraju [5], and Ball and Magazine [6], provide low polynomial algorithms for solving the preemptive or non-preemptive unit capacity DARP. Anily, Gendreau and Laporte provide in [2] an $O(n^2)$ algorithm for solving the unit capacity SP on a line. Anily and Pfeffer provide in [4] an $O(n)$ algorithm for the uncapacitated SP on a line. These two papers consider any mix of objects. The unit capacity SCP (and therefore, the non-preemptive SP) on a tree was shown to be NP-hard by Frederickson and Guan [9]. The preemptive version of the same problem was shown to be polynomially solvable by Frederickson and Guan in [8], who designed an $O(nm)$ algorithm. The unit capacity preemptive SP on a tree was shown to be NP-hard by Anily, Gendreau and Laporte [3]. The uncapacitated SP and DARP on a tree were shown to be NP-hard by de Paepe, Lenstra et al. in [7], by a reduction of the Minimum Vertex Feedback Problem, see Karp [11].

We prove some structural properties that any optimal solution for the uncapacitated DARP on a tree satisfies. Tight lower and upper bounds on the optimal cost are provided, as well as necessary and sufficient conditions under which the optimal solution coincides with the lower bound. We develop a dynamic programming formulation that computes the optimal solution in small problems. Finally, a heuristic whose effectiveness is tested via a computational study is designed.

2 Notations and Preliminary Results

Given the tree $T = (V, E)$ let vertex 1 be its root. For any vertex $v \in V$ let $CH(v)$ its set of children ($CH(v) = \emptyset$ if v is a leaf), $p(v)$ its parent vertex, where $p(1) = 0$, and $T(v) = (V_v, E_v)$ the maximal subtree rooted at v. For any leaf v assume that $s(v) \neq d(v)$, as otherwise, the leaf and the edge incident to it can be removed from T. The cost of a subset of edges $E' \subseteq E$, denoted by $cost(E')$, is the sum of the costs of the edges in E'. Similarly, we use $cost(T) = cost(E)$ for the total length of the edges of T. As T is a tree, there exists a single path connecting each pair of vertices. For $u, v \in V$, $u \neq v$, denote by $[u, v]$ the path connecting u and v. I.e., $[u, v]$ is the minimal sequence of edges $(u_1, u_2), (u_2, u_3), \ldots, (u_{k-1}, u_k)$ that covers $[u, v]$, where $(u_\ell, u_{\ell+1}) \in E$, for $\ell = 1, \ldots, k - 1$. The path $[u, v]$ is said to traverse any of the vertices in $\{u_1, u_2, \ldots, u_k\}$. For any object $i \neq 0$ let $srce(i)$ ($dest(i)$) be the vertex that supplies (demands) i. For any $v \in V$ define the subset of objects $O(v) = \{i : i \in O \backslash \{0\}, \{srce(i), dest(i)\} \subset V_v \backslash \{v\}$ and $[srce(i), dest(i)]$ traverses $v\}$. Thus a shipment of an object in $O(v)$ must traverse vertex v. The set $O(0) = O \backslash \cup_{v \in V} O(v)$ contains the null object and all objects that both their source and destination appear on one path from the root to one of the leaves. These objects are easy to serve as they can be shipped without any extra cost by any tour that visits all vertices. Let OPT denote the optimal solution's cost. The first lemma provides a lower and an upper bound on OPT.

Lemma 2.1 $2cost(T) \leq OPT \leq 4cost(T)$. *These two bounds are tight.*

Two traversals of each edge is the minimum required for visiting all vertices. For the upper bound note that two complete traversals of T - while in the first all objects are picked up, and in the second they are unloaded - is a feasible solution. The upper bound does not imply that each edge must be traversed at most 4 times by an optimal solution. We have an example where in any optimal solution and any vertex v with $k \geq 2$ children each having a supply and a demand for non-null object2, then the edge $(p(v), v)$ is traversed exactly $2(k + 1)$ times.

Next, in order to solve the problem we use auxiliary directed graphs (*digraphs*). Thus, we define an *arc* (u, v) to be a directed edge pointing from u to v. A *loop* is a directed cycle. A digraph is said to be *simple* or *a-cyclic* if it has no loops. For each v with $O(v) \neq \emptyset$ construct a digraph $G(v) = (CH(v), A(v))$ where arc $(u, w) \in A(v)$ if and only if there exists $i \in O(v)$ such that $srce(i) \in V_u$ and $dest(i) \in V_w$. The next theorem identifies cases where the problem is simple.

Theorem 2.1 $OPT = 2cost(T)$ *if and only if for all vertices $v \in V$ for which $O(v) \neq \emptyset$, all digraphs $G(v) = (CH(v), A(v))$ are simple.*

Under the theorem's condition we design an optimal solution for DARP by a depth first scanning of T, where the order the vertices $CH(V)$ are visited preserves the partial order defined by $G(v)$. For general problems we design a heuristic H in which each edge of T is traversed at most 4 times. A computational test compares the heuristic to the optimal solution, which we compute by either using an exact dynamic programming that we developed or by an exact integer linear programming formulation. Currently we are testing the effectiveness of these two formulations while improving them. We conclude the abstract by describing the heuristic.

The Heuristic H: Initially let $\Delta^0 = 0$, and $c^0(u, w) = c(u, w)$ for any $(u, w) \in E$. Identify all leaves v of T where either their supply or demand is in $O(0)$. The edge $(p(v), v)$ is traversed exactly twice in any optimal solution, therefore set $c^0(p(v), v) := 0$. Similarly, generalize this procedure to any $v \in V$ for which $s(u) \in O(0)$ ($d(u) \in O(0)$) for all $u \in V_v$, by setting $c^0(u', w') := 0$ for any $(u', w') \in E_v$, and $c^0(p(v), v) := 0$. Scan T by a depth first search where the decision of which child of $v \in V$ to visit, while at v,is determined by using digraph $G(v)$, which is dynamically updated once one of its vertices is visited. Initially, $v := 1$. If $G(v)$ contains a vertex $u \in CH(v)$ whose in-degree is 0, then remove u and all arcs exiting u from $G(v)$, and proceed to $v := u$. Otherwise, all the remaining vertices in $G(v)$ have an in-degree greater than 1, implying that $G(v)$ contains at least one loop. Let $LOOP(G(v))$ be the collection of vertices that form the loops in $G(v)$. We propose to solve a *a minimum weighted feedback vertex set* problem on the subgraph induced by $G(v)$ on the vertices $LOOP(G(V))$. For that sake assign *weights* to the vertices in $LOOP(G(V))$: for any $u \in LOOP(G(v))$ let the set of edges $ST(v, u)$ to consist of (v, u) and the minimal subtree of $T(u)$ that spans the vertices $w \in V_v$ for which $d(w) \in O(v)$. Let $\alpha(v, u)$ be the total length of

the edges in $ST(v,u)$. Let $out(u)$ be the out-degree of vertex u in $G(v)$, and set $price(u) = \frac{\alpha(v,u)}{out(u)}$ to be the weight of u. Let $u* = argmin\{price(u) : u \in LOOP(G(v))\}$. In case of ties give priority to a vertex whose out value is the greatest. In the heuristic H each edge $(p(w),w) \in ST(v,u)$ needs to be traversed twice from $p(w)$ to w. Therefore, set $\Delta^0 := \Delta^0 + \alpha(v,u*)$ and $c^0(u',w') := 0$ for any edge $(u',w') \in ST(v,u)$. Remove from $G(v)$ vertex $u*$ and all the arcs incident to $u*$. Tag $u*$ and set $v =: u*$ and repeat. If all vertices of $CH(v)$ have been tagged or if v is a leaf then set $v := p(v)$. If $p(v) = 0$ stop. The cost of the heuristic H is $cost(H) = 2cost(T) + 2\Delta^0$.

References

[1] Anily, S. and R. Hassin, "The Swapping Problem", *Networks* 22, 419-433 (1992).

[2] Anily, S., M. Gendreau and G. Laporte, "The Swapping Problem on a Line", *SIAM Journal on Computing* 29, 327-335 (1999).

[3] Anily, S., M. Gendreau and G. Laporte, "The Preemptive Swapping Problem on a Tree", Working Paper, Center for Research in Transportation, Montreal, Canada (2006).

[4] Anily, S., and A. Pfeffer, "The Uncapacitated Swapping Problem on a Line and on a Circle", Working Paper, Israeli Institute for Business Research, Faculty of Management, Tel-Aviv University, Israel (2009).

[5] Attalah, M.J. and S.R. Kosaraju, "Efficient Solutions to Some Transportation Problems with Applications to Minimizing Robot Arm Travel", *SIAM Journal on Computing* 17, 849-869 (1988).

[6] Ball, M.O. and M.J. Magazine, "Sequencing of Insertions in Printed Circuit Board Assembly", *Operations Research* 36, 192–201 (1988).

[7] de Paepe, W. J.K. Lenstra, J. Sgall, R. Sitters and L. Stougie, "Computer-Aided Complexity Classification of Dial-a-Ride Problems", *INFORMS Journal on Computing* 16, 120–132 (2004).

[8] Frederickson, G.N. and D.J. Guan, "Preemptive Ensemble Motion Planning on a Tree", *SIAM Journal on Computing* 22, 1130-1152 (1992).

[9] Frederickson, G.N. and D.J. Guan, "Non-preemptive Ensemble Motion Planning on a Tree", *Journal of Algorithms* 15, 29-60 (1993).

[10] Frederickson, G.N., M.S. Hecht and C.E. Kim, "Approximation Algorithms for some Routing Problems", *SIAM Journal on Computing* 7, 178-193 (1978).

[11] Karp, R.M., "Reducibility among Combinatorial Problems", in R.E. Miller and J.W Thatcher (eds.), *Complexity of Computer Computations* Plenum Press, New York, 85-103 (1972).

The undirected capacitated arc routing problem with profits

Dominique Feillet

École des Mines de Saint-Étienne

Gardanne, France

Alain Hertz

École Polytechnique and GERAD

Montréal, Canada

M. Grazia Speranza

Department of Quantitative Methods

University of Brescia, Italy

Claudia Archetti

Department of Quantitative Methods

University of Brescia, Contrada Santa Chiara 50, Brescia, Italy

Email: archetti@eco.unibs.it

1   Text

In this paper we consider the problem where a profit and a demand are associated with each edge of a set of profitable edges, representing customers, of a given graph. A travel time is associated with each edge of the graph. A fleet of capacitated vehicles is given to serve the profitable edges. A maximum duration of the route of each vehicle is also given. The profit of an edge can be collected by one vehicle at most. If the profit is collected, the vehicle also serves the demand of the edge. The objective of this problem, that is called the Undirected Capacitated Arc Routing Problem with Profits (UCARPP), is to find a set of routes that satisfy the constraints on the duration of the route and on the capacity of the vehicle and maximize the total collected profit.

The routing problems with profits have very interesting applications in the context of auctions in transportation, as support tools to the decisions of the carriers. We focus here only on the applications of the UCARPP. Consider the case of a truck load carrier that has a fleet of vehicles. Each customer is represented by an edge of a graph. Serving a customer means traversing the edge from the origin of the trip to the destination. The same vehicle can serve several customers but after a maximum time limit has to return to the depot, due for example to constraints on the driving time of the driver. The carrier has got a set of customers to be served but these customers

do not completely use the time available for the service of one or more vehicles. Thus, the carrier is interested in finding new customers that, together with the others, fill the time capacity of the vehicles. Several databases are available nowadays that contain demands of customers. The carriers can submit bids to these potential customers. To do so, they must be able to evaluate the impact of serving one or more of these customers on their fleet. The problem for a carrier becomes the problem of identifying the subset of the potential customers that maximize the profit collected by the carrier while satisfying the constraint on the duration of each route. This problem can be modeled with the UCARPP. In this case there is no demand of the customers and the vehicles are uncapacitated. The customers that need to be served can be modeled as customers with a very large profit. The capacitated case arises when sets of potential customers are located along streets, represented by edges, and can be served by traversing the edge. The collected freight is taken to the depot where it is then re-organized to be delivered to the destinations.

A formal description of the UCARPP is the following. Consider an undirected graph $G = (V, E)$ where $V = \{1, \ldots, n\}$ is the set of vertices and E is the set of edges. Vertex 1, the depot, is the starting and ending point of each tour. A travel time t_e is associated with each edge $e \in E$. The travel times satisfy the triangle inequality. Let $L \subset E$ be a subset of the edges of the graph, called *profitable edges*. A positive profit p_e and a demand d_e are associated with each edge $e \in L$, in addition to the travel time t_e. A set of m vehicles is available. Each vehicle has a capacity Q and is subject to a time limit T_{max} on the duration of the route. The profit of each edge $e \in L$ can be collected by one vehicle at most. If the profit is collected by a vehicle, the corresponding demand has to be served by the same vehicle. The objective of the UCARPP is to find a route for each of the m vehicles such that the total profit is maximized and each route satisfies the capacity and time limit constraints on the vehicles.

We first propose a branch and price algorithm to solve the UCARPP exactly. The master problem reduces to a set packing problem where the objective is to select the m most profitable routes. The subproblem is instead a variant of the elementary shortest path problem with resource constraints (ESPPRC) where the resources are related to the capacity and time constraints. A label-setting algorithm is proposed to solve the subproblem and different acceleration techniques are introduced to speed-up the process. However, the exact approach is able to solve only relatively small size instances. Thus, we then propose also three different metaheuristic algorithms to deal with large size instances: a variable neighborhood search and two tabu search heuristics, one that explores feasible solutions only and one that allows infeasibility. The main idea of the three heuristic schemes is that all customers are organized in routes (possibly having a number of routes greater than m) and only the first m routes contribute to the calculation of the objective function. As a consequence, at each move the set of 'profitable' routes can change, and this proved to be helpful in diversifying the search. The difference between Tabu Search (TS) and Variable Neighborhood

Search (VNS) schemes is that VNS explores different neighborhoods while TS focuses on a single neighborhood. For all the metaheuristic algorithms we propose two variants: a slow variant, with a careful but time-consuming evaluation of the neighborhood, and a fast variant, with an approximate but fast evaluation of the neighborhood.

Computational tests are made to show the effectiveness of the methods proposed on a set 34 benchmark instances for the arc routing problem proposed by Belenguer and Benavent (2003). For each instance we consider a fleet of 2, 3 and 4 vehicles, respectively, thus obtaining a first set of 102 instances. This first set of instances proved to be very difficult to be solved by the branch and price algorithm. Thus, we consider a second set of 102 instances obtained from the instances of the previous set where the value of the capacity and time limit of each route is decreased. This allow to obtain shorter routes and thus, makes the instances easier for the branch and price.

The results show that the exact approach can solve instances with up to 97 profitable edges in few minutes, when the capacity of the vehicles and the duration of the routes do not generate 'long' routes, while, when the routes are 'long', one hour is not sufficient to solve exactly instances with 39 profitable edges although some instances with 66 profitable edges were solved. The hardness of the instances depends on the length of the routes in the optimal solution much more than on the number of given profitable edges. The variable neighborhood search heuristic is the most effective of the tested heuristics, with average errors below 1% with respect to the best solution obtained, either optimal or heuristic. The results also showed that a careful, though time consuming, evaluation of the quality of the solutions is beneficial with respect to a rough evaluation of the quality of the solutions in favor of a larger number of iterations.

References

[1] C. Archetti, D. Feillet, A. Hertz and M.G. Speranza, "The capacitated team orienteering and profitable tour problem", *Journal of the Operational Research Society*, doi: 10.157/palgrave.jors.2602603, (2008).

[2] C. Archetti, A. Hertz and M.G. Speranza, "Metaheuristics for the team orienteering problem", *Journal of Heuristics* 13, 49-76 (2007).

[3] J.M. Belenguer and E. Benavent, "A cutting plane algorithm for the capacitated arc routing problem", *Computers and Operations Research* 30, 705-728 (2003).

[4] S. Boussier, D. Feillet and M. Gendreau, "An exact algorithm for team orienteering problems", *4OR* 5, 211-230 (2007).

[5] I-M. Chao, B. Golden and E.A. Wasil, "The team orienteering problem", *European Journal of Operational Research* 88, 464-474 (1996).

[6] M. Dror, *Arc routing: Theory, solutions and applications*, Kluwer Academic Publishers (2000).

[7] D. Feillet, P. Dejax and M. Gendreau, "Traveling salesman problems with profits", *Transportation Science* 39, 188-205 (2005).

[8] D. Feillet, P. Dejax and M. Gendreau, "The profitable arc tour problem: solution with a branch-and-price algorithm", *Transportation Science* 39, 539-552 (2005).

[9] H. Tang and E. Miller-Hooks, "A tabu search heuristic for the team orienteering problem", *Computers and Operations Research* 32, 1379-1407 (2005).

The Free Newspaper Delivery Problem

Claudia Archetti

Department of Quantitative Methods

University of Brescia, Brescia, Italy

Karl F. Doerner, Fabien Tricoire

Department of Business Administration

University of Vienna, Brünner Straße 72, Vienna, Austria

Email: karl.doerner@univie.ac.at

1   Motivation and Problem Description

In the past years free newspapers available at subway and tramway stations become more and more popular. These newspapers are displayed in freely accessible news racks mainly in front of the entrance of subway stations or in the stopping area of the tramway or the public bus. People on their way to work, school or university using public transportation have access to these newspapers already early in the morning.

The production and delivery costs of these newspapers are financed only by the advertising revenues for publishing advertisements in these newspapers. The prices for the advertisements depend mainly from the estimated number of readers of the newspapers. For the regular newspapers sold at newspapers stores the number of readers can easily be determined. The number of sold newspapers is known and this number is usually multiplied by a certain factor. For the free newspaper no exact information on the number of readers exist. The produced newspapers have to be delivered to the news racks in such a way that after the end of the morning rush hour (almost) all the news racks are empty. The number of produced newspapers is then multiplied by the same factor for the number of readers and this is the used approximation of the number of readers. To justify the number of readers the newspapers have to be distributed in that way that all the news racks are empty after the morning rush hour (at about 9.00 a.m.). To have a high number of readers is important for the revenues generated by the advertisements.

During the operation start of the public transportation system and the end of the morning rush hour (about 9.00 a.m.) the passengers should have access to the free newspaper. At the end of the morning rush hours all the newspaper racks have to be empty. The consumption rate of

the newspapers is different at every news rack. Also the sizes of the news racks located at the stations are different. Most of the news racks are accessible from 5 a.m. in the morning - before 5 a.m. the public transportation system does not operate and the subway stations are closed. The production of the newspaper starts at 1.00 a.m. and ends at 7.00 a.m. 40.000 newspapers per hour are produced. The batch of the 40.000 newspapers is always available at the end of the hour. In practice, since the delivery horizon starts at 4.00 am, we begin with an available stock of 120.000 newspapers. In total 240.000 newspapers are produced and the production quantity is fixed.

In order to provide free newspapers to underground and tramway stations, a delivery company has to perform vehicle routes. The used vehicle fleet is homogeneous. Each of these routes start from a central depot - the production plant, provides a set of stations with certain amounts of newspapers, and ends at the same depot. Each vehicle can perform several routes. - which means that the vehicle can return to the central depot to pickup new batches of newspapers.

Each news rack at the station has a given capacity, and the inventory level at a station should never exceed its capacity and also no stock-out should occur. In a similar way, each vehicle has a given capacity. Each station has also a given consumption rate, and the depot or production plant has a production rate.

The goal of the company is (i) to consume all produced newspapers by distributing them to the stations, where they are actually consumed, and (ii) to use as few trips as possible. The newspaper delivery company is paid by the number of trips performed.

As the number of pages of the newspaper can vary and therefore the capacity requirements of each newspaper varies, therefore also the capacity requirements of the newspapers in the news rack is different. For different capacity requirements of the newspapers different delivery plans have to be created.

The planning horizon starts at 4.00 am and ends at 9.00 am. In our solution approach the planning horizon is divided into periods. The length of each period is 30 minutes. The consumption of newspapers at the stations starts at 5.00 am. Hence, all the stations should be visited at least once during 4.00 a.m. and 5.00 a.m., otherwise some consumption is lost. Around 9.00 am all the newspapers have to be distributed and all the news racks should be empty.

Some stations need to be visited only once. These are typically small stations, mainly far distant from the depot. Their capacity allows them to handle their total consumption on the whole horizon. Other stations require two to three visit. Some stations require even up to four visits.

The objective is to minimise a linear combination of two distinct quantities:

1. The time period at which the latest stockout occurs.

2. The total number of trips.

The considered problem is closely related to the inventory routing problem [2]. In inventory routing problems the vendor manages the resupply. Vendor managed inventory is an example of value creating logistics. Vendors save on distribution cost by being able to better coordinate deliveries to different customers, and customers do not have to dedicate resources to inventory management. The differences to the classical inventory routing problem lie in the following aspects: i) at the end of the planning horizon in all the news racks a stock-out should occur. ii) Furthermore, all the newspapers are not available when the distribution starts. The production rate and the availability of the newspapers have to be taken into account. This aspect is related to the integrated production and transportation scheduling problem with capacity constraints [3]. iv) The main difference to the classical inventory routing problem is the time aspect. In classical inventory routing problems in every period delivery routes for the customers are created (see e.g. [4]). In our problem - as the periods are quite short consisting of intervals of 30 minutes - the routes can last more than one period. This leads to the following effects: The available capacity of a news rack can differ when the customer is visited at the beginning of the route or at the end of the route. As in [3] we consider also a short shelf life product. There is no inventory of the product in process. The newspaper is produced and immediately delivered.

2 Solution Approach

For approaching this specific inventory routing problem different hybrid solution methods of heuristics and exact solution techniques are developed. We present a solution approach that integrates heuristic search with optimization by using an integer program to explore promising parts of the search space. Exact algorithms are guaranteed to find an optimal solution and prove its optimality; the run-time, however, often increases dramatically with a problem instance's size, and only small or moderately sized instances can be practically solved to provable optimality. For larger instances the only possibility is usually to turn to heuristic algorithms that trade optimality for run-time; i.e., they are designed to obtain good but not necessarily optimal solutions in acceptable time. The combination of heuristics and integer programming leads to many different design decisions. Successful combinations of heuristic search with integer programming are reported in [1].

We propose different decomposition approaches for the considered problem. We decompose our problem in two phases.

In the first phase, the stations are allocated quantities to be delivered at given periods. We call this phase the *demand fixing* phase. The demand fixing phase can easily be solved by using an exact solver. In this phase different strategies and different objectives for the subproblem are tested. By using different strategies different problems for the second phase emerge. The different strategies in the demand fixing phase produce subproblems for the routing phase with different

batch sizes of newspapers to be delivered at the public transportation stations. The second phase is called the *routing phase*.

Within the demand fixing phase vehicle routing problems with time windows are generated. In the routing phase the vehicle routing problems with time windows, in which each time window corresponds to a given period, have to be solved. If a given station has to be visited several times during the horizon, then it will be associated to several different nodes in the generated vehicle routing problem with time windows, one node per delivery is created. In the second phase, the vehicle routing problem with time windows is solved with a classical local search based heuristic solution approaches. One should note that the generated vehicle routing problem with time windows is more complex than the usual case, since the production schedule has to be taken also into account: all routes that already left the depot should not use more product than what has been already produced.

References

[1] C. Archetti, M. G. Speranza, M. W. P. Savelsbergh (2009),"An Optimization-based Heuristic for the Split Delivery Vehicle Routing Problem", *Transportation Science* 42 (1), pp. 22-31.

[2] A. Campbell, L. Clarke, A. Kleywegt, M. Savelsbergh (1998), "The Inventory Routing Problem". In. Crainic, T., Laporte, G. (eds.), Fleet Management and Logistics, Kluwer Academic Publishers, Dordrecht, The Netherlands, pp. 95–112.

[3] H. N. Geismar, G. Laporte, L. Lei, C. Sriskandarajah (2009), "The Integrated Production and Transportation Scheduling Problem for a Product with a Short Lifespan", *INFORMS Journal on Computing* 20 (1), pp. 21-33.

[4] V. Hemmelmayr, Doerner, K. F., Hartl, R. F., M. Savelsbergh (2008), "Delivery strategies for periodic blood products supplies", *OR Spectrum*, 31 (4)., pp. 707-725.

Tabu Search for Coordinating Production and Distribution Routing Problems

Vinícius Amaral Armentano

Faculdade de Engenharia Elétrica e de Computação

Universidade Estadual de Campinas, CP 6101, Campinas, SP, 13083-970, Brazil

Email: vinicius@densis.fee.unicamp.br

André Luís Shiguemoto

Departamento de Estatística e Matemática Aplicada

Universidade Federal do Ceará

1 Introduction

This work addresses the problem of optimally coordinating a production-distribution system over a horizon with T periods, where a facility production with no capacity constraints produces J items which are distributed to a set of N customers by a fleet of V homogeneous vehicles in order to meet customers' demands. Each customer defines its minimum and maximum inventory levels. In each period, the production problem involves determining how much to produce of each item, while the distribution planning defines when customers should be visited, the amount of each item that should be delivered to customers and the vehicle routes. A production fixed cost is incurred every period that an item is produced, and a transportation fixed cost is incurred if a vehicle is used at least once in the planning horizon. The objective is to minimize the sum of production and inventory costs at the facility, inventory costs at the customers and distribution costs. An inventory-routing problem is also addressed, where known quantities of items are produced or made available at a supplier, which has to plan the distribution to customers as described above. In this case, there is no production planning and consequently no production costs, but inventory costs are still taken into account at the supplier. For reviews on such problems, see [1], [2], [3], [4] and [5].

Bertazzi et al. [6] deal with the single item version of the production-distribution problem and analyze two Vendor-Managed Inventory (VMI) policies: the order-up-to level policy, in which the amount of an item that is delivered to a customer is such that it reaches its maximum inventory (VMIR-OU) and the fill-fill dump policy, in which the order-up-to level amount is shipped to all but the last retailer on each delivery route, while the quantity delivered to the last retailer is the minimum

between the order-up-to level quantity and the residual vehicle capacity. A heuristic that decomposes the problem in a production subproblem and a distribution subproblem is proposed to minimize the total cost subject to the above inventory policies.

Bertazzi et al. [7] propose a heuristic for solving a single item, single vehicle inventory-routing problem subject to the VMIR-OU policy. Archetti et al. [8] develop a branch-and-cut algorithm for solving the same problem, subject to three vendor-managed inventory policies. The first policy is the VMIR-OU and the second policy allows the delivery of any quantity between the minimum and maximum inventory levels (VMIR-ML) The third policy disregards the maximum inventory level and any amount can be delivered as long as the inventory does not fall under the minimum inventory level (VMIR).

In this work, we use the tabu search methodology [9] to solve the production-distribution and the inventory-routing problems. Our objective is threefold. First, we show that there is a significant cost reduction on the single item instances generated by Bertazzi [6] by applying the VMIR-ML strategy as opposed to the VMIR-OU policy. Second, we generate multiple item instances and show that the integrated VMIR-ML approach leads to a significant cost reduction relative to the decoupled VMIR-ML approach. The third objective is to show that tabu search is able to obtain high quality solutions when applied to the single item, single vehicle instances generated and solved to optimality by Archetti et al. [8].

2 Tabu search

The proposed tabu search procedure consists of three phases: construction of an initial solution, short term memory and diversification. A solution s is evaluated by a function $c(s) + \alpha \, g(s)$, where $c(s)$ denotes the total cost, $g(s)$ represents the total violation of the capacities of the vehicles and α is a positive parameter that is adjusted during the search in order to facilitate the exploration of the search space.

An initial solution is constructed in three steps: i) the quantity of each item and each customer to be delivered in a period is the demand of this period; ii) the delivery routes are constructed by the parallel version of the Clarke and Wright [10] algorithm; iii) the production plan for each item is determined by the Evans [11] efficient implementation of the Wagner and Whitin [12] algorithm. Such a solution is likely to be infeasible with respect to the capacity of the vehicles.

The move that defines a neighborhood in the short term memory consists of the following three components:
i) for each customer k determine the maximum quantity of an item j that can be transferred from period t to a period t', $t' \neq t$, without violating the minimum and maximum inventory levels for item j;
ii) if customer k is visited in period t', then we add this quantity to be delivered to customer k and maintain the same route, regardless of the vehicle capacity. If customer k is not visited in period t', we insert customer k in all positions of the routes in period t' and select the cheapest insertion. If there are

no routes in period t', then a new route is created for customer k. If the insertion results in a feasible capacity vehicle, we then apply the 2-Opt move to the route from which the customer is removed and to the route where the customer is inserted. The implementation of this move follows the strategy of a bounded neighbor list of the nearest 20 customers proposed by Johnson and McGeoch [13].

iii) determine the new production plan for item j by applying the efficient implementation of the Wagner and Whitin algorithm.

The composite move is examined for each customer k, all periods t, t' and each item j, and the move that results in the least total cost is executed. The pair (j, t') associated with such a move is stored in a matrix to indicate that the shift of any quantity of item j from period t' is tabu for γ iterations, where γ is selected from an interval [a, b] with uniform distribution. As aspiration criterion we adopt the most commonly used: the tabu status of the move is revoked whenever the move leads to a solution that is better than the best solution recorded during the search so far. The search procedure terminates when it reaches $\sigma NJTV$ iterations or when $\eta NJTV$ iterations have elapsed without updating the incumbent solution, where σ and η are parameters.

3 Computational results

The tabu search procedure is tested on a set of instances generated according to the scheme proposed Bertazzi et al. (2005). The number of customers, periods and items is {30, 50, 100}, {12, 24}, {5, 10}, respectively. For each combination of customer, period and item, nine instances were created, totalling 108 generated instances. In order to evaluate the cost that results from the integrated production-distribution approach, we compare it with the cost of the classical decoupled approach, in which first the production planning is obtained by the application of the efficient implementation of the Wagner and Whitin algorithm, and then the distribution planning is solved by the above tabu search procedure. For 30, 50 and 100 customers the integrated production-distribution approach presents a mean reduction cost over 108 instances of 22.76%, 33.01% and 58.97% with respect to the decoupled approach. For the 96 single-item instances of Bertazzi et al. [6] with 50 customers and planning horizon of 30 periods, the cost reduction of the VMIR-ML strategy relative to the VMIR-OU is nearly 49%. With relation to the 160 instances generated by Archetti et al. [8] with number of customers varying from 5 to 50 and number of periods, 3 and 6, the tabu search obtained a mean cost gap below 1.6% with respect to the optimal solution cost.

References

[1] J.D. Thomas and P.M. Griffin, "Coordinated Supply Chain Management", *European Journal of Operational Research* 94, 1-15 (1996).

[2] A.M. Sarmiento and R. Nagi, "A Review of Integrated Analysis of Production-Distribution", *IIE Transactions*, 31, 1061-1074 (1999).

[3] Z-L. Chen, "Integrated Production and Distribution Operations: Taxonomy, Models, and Review", in *Handbook of Quantitative Supply Chain Analysis: Modelling in the E-Business Era*, D. Simchi-Levi, D. Wu and Z.-L. Chen (eds), 711-746, Kluwer, Boston, MA (2004).

[4] A.M. Campbell, L. Clarke, A. Kleywegt and M.W.P. Savelsbergh, "The Inventory Routing Problem". In: *Fleet Management and Logistics*, T.G. Crainic and G. Laporte, (eds), 95-113, Kluwer, Boston, MA (1998).

[5] L. Bertazzi, M.W.P. Savelsbergh and M.G. Speranza, "Inventory Routing", in *The Vehicle Routing Problem: Latest Advances and New Challenges*, B. Golden, R. Raghavan, and E. Wasil (eds), 49-72, Springer, New York (2007).

[6] L. Bertazzi, G. Paletta and M.G. Speranza, "Minimizing the Total Cost in an Integrated Vendor-Managed Inventory System", *Journal of Heuristics* 11, 393-419 (2005).

[7] L. Bertazzi, G. Paletta and M.G. Speranza, "Deterministic Order-up-to Level Policies in an Inventory Routing Problem", *Transportation Science* 36, 119-132 (2002).

[8] C. Archetti, L. Bertazzi, G. Laporte and M.G. Speranza, "A Branch-and-Cut Algorithm for a Vendor-Managed Inventory-Routing problem", *Transportation Science* 31, 381-391 (2007).

[9] F. Glover and M. Laguna,. *Tabu Search*, Kluwer, Boston, MA (1997).

[10] G. Clarke and J.V. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", *Operations Research* 12, 568-581 (1964).

[11] J.R. Evans, "An Efficient Implementation of the Wagner-Whitin Algorithm for Dynamic Lot-Sizing", *Journal of Operational Managemen,* 5, 229-235 (1985).

[12] H.M. Wagner and T.M. Whitin, "Dynamic Version of the Economic Lot Size Model", *Management Science* 5, 89-96 (1958).

[13] D.S. Johnson and L.A. McGeoch, "The Traveling Salesman Problem: A Case Study in Local Optimization", in *Local Search in Combinatorial Optimization*, E.H.L. Aarts and J.K. Lenstra (eds), 215-310, Wiley, Chichester (1997).

Analysis and Simulation of a Port Container Terminal

Pietro Averaimo, Giuseppe Bruno, Francesco Gargano,

Andrea Genovese, Gennaro Improta, Cinzia Vinti

Department of Engineering Management

University of Naples "Federico II"

Email: improta@unina.it

Introduction

A port container terminal, as known, is a complex logistic system in which several, material and immaterial, flows interact. In this context, operations management requires an appropriate use of Information and Communication Technologies as well as Decisions Support Systems. This problem becomes more and more crucial for container terminals using limited resources in terms of yards, quays and equipments.

This is the case of the port container terminal located in Naples harbor (Italy) that we analyzed and whose operations and processes we reproduced through a discrete event simulation model built on the base of real data, on the field, collected.

The performed experiments indicated the suitability of the simulation model to effectively represent the current scenario. On this basis, we have implemented and compared different management and optimization policies in order to evaluate how they could affect the overall terminal performances.

Container Terminals: Generalities

A container terminal can be represented as a system with four main physical components: (a) *Quay Side*, the marine side interface for berthing/unberthing operations; (b) *Container Storage System*, hosting the yard and the relative cranes for the container stacking and delivery operations; (c) *Transfer System*, including vehicles for moving containers from/to quay side to/from yard; (d) *Gate*, the land side interface for trucks and trains.

In a maritime terminal it is possible to distinguish three different container flows: *Import flow*, formed by containers unloaded from a vessel through gantry cranes, transferred to the yard and then picked up by road or rail carriers; *Export flow*, formed by containers entering the terminal through the gate, transferred to the yard and then loaded on a ship; *Transhipment flow*, formed by containers unloaded from a vessel, temporarily stored in the yard and then loaded on another vessel. Figure 1 schematically shows the components of the system and container flows among them.
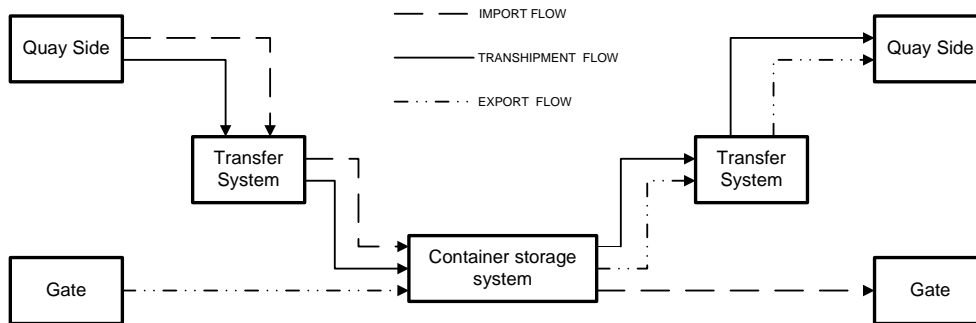
Fig 1. Port container terminal system.

The presence of the physical components and container flows makes a port terminal a complex logistic system in which several operations could be optimized in order to achieve desired efficiency and effectiveness. Various decision making problems can be associated with the four terminal components. As regards to *Quay Side*, each berthing vessel has to be assigned to one of the available quays. Moreover, gantry cranes have to be assigned to berthed vessels and their loading/unloading plans and operations management should be defined. With reference to the *Transfer system* the problem of assigning vehicles to gantry cranes (or to berthed vessels) occurs. Then in the *Container Storage System*, areas have to be allocated to containers belonging to different flows as well as yard cranes have to be assigned to the several areas and, moreover, container storage strategies should be defined. Finally, at the *Gate*, admission and scheduling policies should be arranged.

The complexity of the occurring decision problems has stimulated a significant recent literature about modeling and solving management operational problems. For a detailed review see [3] and [4].

Simulation and Container Terminals

Though there is a wide availability of optimization models and techniques to deal with specific operational problems, as the number of the involved parameters is extremely high, and they are tied up each other very often according to non-linear patterns, simulation represents a suitable approach for analyzing container terminals.

We thoroughly analyzed international literature, from 1988 to 2008, dealing with container terminals' operations through simulation methods. Many papers (29) regarding specific simulation-based studies have been retrieved. The most of them (72%) reproduces the container terminal as a whole, including, in such a way, some of the above mentioned decision problems, while 78% of papers deals with real case studies. In particular, 28% of papers embed optimization approaches within the general simulation framework, in order to cope with specific sub-problems. Parola and Sciomachen [2], as an example, developed a simulation-based environment to estimate the impact of railway capacity expansion on Genova (Italy) terminal performances; Legato and Mazza [1] coped with the

berth resources planning problem, including what-if approaches in a simulation environment. Some attempts of general purpose port simulation models have been also developed. Among them the UNCTAD port model, PORTSIM, and the MIT port simulator [5]. Further indications can be derived from [3] and [4].

A critical analysis of the literature has pointed out that it is quite hard to provide general purpose mathematical and/or simulation models for reproducing port container terminals. This is due to the fact that each specific case study presents several singularities (i.e., equipments, procedures, layout, operational constraints, local factors). For this reason it is very difficult, in general, to adapt available proposals to specific case studies.

Some information about Co.Na.Te.Co. Container Terminal

The Co.Na.Te.Co. container terminal, located in Naples harbor (Italy), has been studied. The terminal, one of the top-twenty in the Mediterranean sea, extends over an area of 145.671 square meters and with a maximum static storage capacity of 12.000 TEUs. The terminal is characterized by 3 quays, 6 gantry cranes and 26 yard cranes. The throughput in 2008, before the current economic crisis, reached about 420.000 TEUs, while the number of berthed vessels exceeded 570.

Due to the limited available resources, the terminal often works under congested conditions and, consequently, its performances should be optimized. As a consequence the terminal management was interested: (i) in individuating system bottlenecks mainly affecting the performances and (ii) in developing possible future scenarios in terms of capacity expansion projects.

The proposed model

To this aim we have designed a discrete event simulation model in order to reproduce current operations and understating critical issues. In Figure 2 the overall scheme of the proposed simulation model is shown. Its functioning is based on the interaction between the *Vessels to Berth Matrix* and the *Quays State Matrix* and on *five main procedures* regarding: (i) the arrival of a vessel j, (ii) the assignment of a quay b to a vessel k, (iii) the loading/unloading operations, (iv) the container storage system and (v) the gate management.

In order to measure the provided performances we have used as principal key performance indicators: the *Average roadstead waiting time*, i.e. the time a vessel has to wait before its berthing; the *Quays utilization rate*, i.e. the vessels berthed on a quay out of the total number of vessels; the *Average amount of containers in the storage system*.

The model has been implemented in Rockwell ARENA 11.0.0 environment and was firstly used to simulate, on the basis of historical data, current situation. The obtained results, in terms of key performance values, confirmed the reliability of the implemented model to reproduce the real functioning of the system. As a second step, we have compared optimizing rules in critical points of the decisional process (as the berth assignment phase). In this way, interesting indications have been

obtained to improve the current performances of the system and, in a long term strategy, to provide insights for a better design of terminal operations.
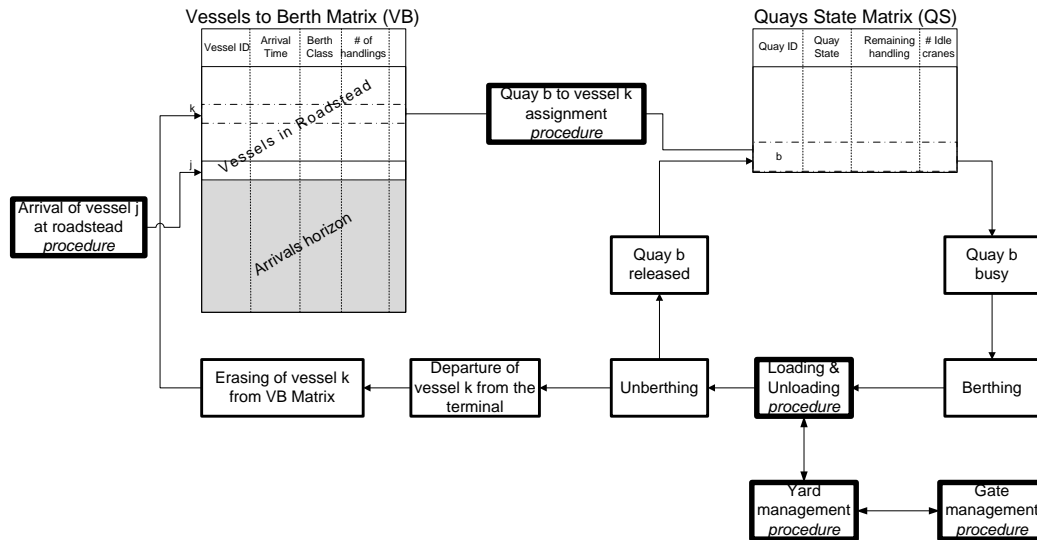
Fig 2. Overall scheme of the proposed simulation model.

Conclusions and open research questions

We developed a simulation model to reproduce logistic activities related to Co.Na.Te.Co. port container terminal operating in Naples harbor. The considered case is particularly interesting as it regards a situation of work under congested conditions (low yard space, few quays, limited number of cranes). In this case the optimization of key performance indicators appears to be the only opportunity to increase the level and the quality of the terminal activities.

As validation tests were satisfactory and the experiments performed indicated a good correspondence of the simulation model results with the current scenario, we have implemented different management and optimization policies in order to evaluate how they could affect the overall terminal performances.

References

[1] Legato P., Mazza R.M. (2001). Berth planning and resources optimization at a container terminal via discrete event simulation. *European Journal of Operational Research*; 133; pp. 537-547.
[2] Parola F., Sciomachen A. (2005). Intermodal container flows in a port system network:: Analysis of possible growths via simulation models. *International Journal of Production Economics*; 97(1); pp. 75-88.
[3] Steenken D., Voß S., Stahlbock R. (2004). Container terminal operation and operations research – a classification and literature review. *OR Spectrum*; 26; pp.3-49.
[4] Stahlbock, R. and S. Voß. 2008. Operations research at container terminals: a literature update. *OR Spectrum*; 30; pp.1–52.
[5] Yun W.Y., Choi Y.S. (1999). A simulation model for container-terminal operation analysis using an object-oriented approach. *International Journal of Production Economics*; 59; pp. 221-230.

Stochastic Integer Programming Models for Air Traffic Flow Management Problems

Michael O. Ball

Robert H Smith School of Business & Institute for Systems Research

University of Maryland

College Park, MD 20742, USA

mball@rhsmith.umd.edu

Moein Ganji

Department of Civil and Environmental Engineering & Institute for Systems Research

University of Maryland

Charles Glover

Applied Mathematics and Scientific Computing Program & Institute for Systems Research

University of Maryland

David Lovell

Department of Civil and Environmental Engineering & Institute for Systems Research

University of Maryland

1   Introduction

In this paper we address a class of stochastic air traffic flow management problems. We focus on problems that arise when airspace congestion is predicted, usually because of a weather disturbance, so that the number of flights passing through a volume of airspace (flow constrained area - FCA) must be reduced. We formulate an optimization model for the assignment of dispositions to flights whose preferred flight plans pass through an FCA. For each flight, the disposition can be either to depart as scheduled but via a secondary route, or to use the originally intended route but to depart with a controlled (adjusted) departure time and accompanying ground delay. We model the possibility that the capacity of the FCA may increase at some future time once the weather activity clears. The model is a two-stage stochastic program that represents the time of this capacity windfall as a random variable, and determines expected costs given a second-stage

decision, conditioning on that time. Our model allows the initial reroutes to vary from pessimistic (initial trajectory avoids weather entirely) to optimistic (initial trajectory assumes weather not present); included is the possibility of routes that "hedge" between the optimistic and pessimistic strategies. We conduct experiments allowing a range of such trajectories and draw conclusions regarding appropriate strategies. While other research [2], considers larger scale (deterministic) problems with many capacitated elements, this work considers a single capacitated element but includes models of uncertainty and system dynamics. Our work aligns closely with current air traffic flow management practice, which defines specific FCA's and associated traffic management initiatives.

We model the problem studied as a multi-stage stochastic program that includes a sequence of decision points over time. A very special structure is employed that leads to a compact scenario space. Specifically, the required solution should specify an initial plan that assumes a particular sample path (that a weather disturbance has a particular duration and particular impact on capacity). Then, there is a sequence of points in time when the weather may clear and a revised plan put in place. There is a probability associated with each such clearance time. This simple structure allows the problem to be modeled as a two stage stochastic program where the random variable realized in the second stage is the clearance time. This model was inspired by Ground Delay Program (GDP) planning problems [1], where this structure is an accurate representation of reality. In our case, it is less accurate; nonetheless, we feel it represents a good approximation and computational studies show it leads to much improved plans when compared to deterministic models.

We now present an integer programming formulation for the case where rerouting of flights is not allowed, i.e. the only control action employed is ground delay. This is a complete model for the GDP planning problem but for airspace problem we must also add variables that represent reroute options. In our optimization model, we represent the airspace capacity using *slots*. A slot is a time window in which a reservation for a flight arrival may be made. A capacity reduction reduces the number available slots and a capacity increase, increases the number of slots. x variables represent the initial plan and y variables the plan under each weather clearance scenario. Each clearance scenario has a larger number of slots (than the base case associated with the initial plan) with earlier clearance times having more slots than later clearance times.

$$k, i/j, t :$$ flight, slot and scenario indexes (respectively)

$$q_t :$$ probability of scenario (clearance time) t

$$a_k :$$ earliest time (slot) that flight k can reach the FCA or airport

$$last(k, j, t) :$$ the latest stage 1 (k-to-i) assignment that can be reallocated to slot j in scenario t

$$x_{k,i} = 1$$ if flight k is assigned to slot i under initial plan

0 if not.

$$y_{k,j,t} \quad = \quad \text{1 if flight } k \text{ is assigned to slot } j \text{ under scenario } t$$

0 if not.

$$\min f(y) \quad = \quad \sum_k \sum_j \sum_t (q_t(j - a_k)y_{k,j,t})$$

subject to

$$\sum_i x_{k,i} \quad = \quad 1 \text{ for each flight } k \tag{1}$$

$$\sum_k x_{k,i} \quad \leq \quad 1 \text{ for each slot } i \tag{2}$$

$$\sum_j y_{k,j,t} \quad = \quad 1 \text{ for each } k, t \tag{3}$$

$$\sum_k y_{k,j,t} \quad \leq \quad 1 \text{ for each } j, t \tag{4}$$

$$\sum_{i|i>last(k,j-1,t);i\leq last(k,j,t);i\geq a_k} x_{k,i} \quad \geq \quad \sum_{j'|last(k,j',t)=last(k,j,t)} y_{k,j',t} \text{ for each } k, j, t \tag{5}$$

$$x_{k,i}, y_{k,j,t} \quad \in \{0,1\} \quad \text{for all } k, j, t \tag{6}$$

Note that the constraints for the initial plan: (1), (2), or for the plan under a given scenario (t): (3), (4), represent a simple assignment problem. However, the connecting constraints (5) result in a more complex integer program. Nonetheless we are able to show that the LP relaxation of this problem always has an integer solution. The underlying polyhedron does have non-integer extreme points, which implies that this result depends on the structure of the objective function.

Our integer programming model for the more general problem with reroute variables is more complex. Figure 1 gives a simplified view of the geometry of routing options. The most direct (zero-angle) route corresponds to the flight proceeding along the horizontal axis at the bottom of the Figure. Alternatively, a flight might proceed around the weather disturbance (maximum angle route) proceeding along the outer edge of the triangle. We call the zero-angle route the optimistic route because a flight might proceed along this route even if the weather had not yet cleared. If the weather does clear before the flight reaches the FCA then the flight can stay on this route. Otherwise, it would fly along the edge of the storm and possibly then turn directly to the destination, e.g. at point 2, if the weather clears. Alternatively, a flight might start on the maximum angle (pessimistic) route and, if the weather clears early, then the flight can turn directly to the destination, e.g. at point 5. There are also conditions under which "hedging" represents the best strategy; this is illustrated by the middle angle route. These various route options represent input data for the more complex integer program. It then determines the best options for each flight taking into account the geometry illustrated in the Figure, the appropriate costs and also the weather clearance probabilities.
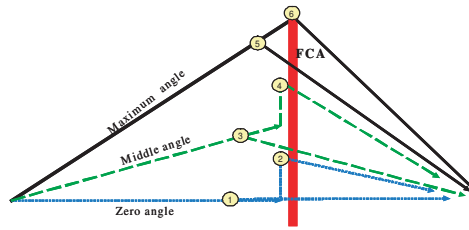
Figure 1: Route Options

We have been able to solve problems of realistic size, e.g. with up to 500 flights and 6 clearance time scenarios, using commercial integer programming solvers. Further, in a variety of simulations we have shown that our models produce substantially improved decisions that lead to significant delay reductions.

References

[1] Ball, M.O., R. Hoffman and A. Mukherjee, 2010, "Ground Delay Program Planning under Uncertainty Based on the Ration-By-Distance Principle", *Transportation Science*, 44, 1-14.

[2] Bertsimas, D. and S. Stock Paterson, 2000, "The Traffic Flow Management Rerouting Problem in Air Traffic Control: a Dynamic Network Flow Approach. *Transportation Science* 34, 239-255.

EXPLORING THE USE OF TRAFFIC DATA COLLECTED FROM NEW ICT BASED SENSORS TO ESTIMATE TIME DEPENDENT OD MATRICES

L. Marqués

CENIT (Center for Innovation in Transport) Technical University of Catalonia

L. Montero

Department of Statistics and Operations Research, Technical University of Catalonia

C. Carmona

CENIT (Center for Innovation in Transport) Technical University of Catalonia

J. Barceló

Department of Statistics and Operations Research and

CENIT (Center for Innovation in Transport) Technical University of Catalonia

Email: jaume.barcelo@upc.edu

Time dependent origin to destination, OD, matrices are the key input to dynamic traffic models, mainly to simulation models, micro as well as mesoscopic. Dynamic Traffic Models, DTM, are one of the major components of the Advanced Traffic Management Systems and Advanced Traffic Information Systems. DTM play a crucial role in estimating the current traffic state and forecasting its short term evolution. The quality of the results that they provide depends, not only on the quality of the models, but also on the accuracy and reliability of the inputs and, therefore, on the quality of the time dependent OD matrices as part of that input. These matrices have usually been estimated by procedures exogenous to the traffic simulation model, based typically on heuristic procedures adapted from static matrix adjustments from link flow counts. Recently dynamic approach based on Kalman Filtering, [1], [2], have been proposed, they explicitly assume that a dynamic assignment procedure is available. The quality and reliability of the measurements produced by inductive loop detectors, is not usually the one required by real-time applications, therefore one wonders what could be expected from the new ICT technologies as for example Automatic Vehicle Location, License Plate Recognition, detection of mobile devices and so on, in particular those equipped with Bluetooth technology, that are becoming pervasive data sources. Once the privacy concerns are overcome, tracking mobile devices associated uniquely to vehicles becomes a rich source of new traffic related data from which infer time dependent mobility patterns. Better results should be expected when V2I technologies are taken into account making possible paths reconstructions and from them the estimation of origin to destination matrices for each time period. The research reported in this paper explores two complementary issues for

estimating OD matrices: the exploitation of travel time measurements provided by sensors detecting Bluetooth devices equipping vehicles (Tom-Tom, Parrot, hands free...) combined with input-output flow measurement at entry and exit ramps on a motorway; and the use of data supplied by V2I technologies (i.e. positions and speeds) that allow tracking vehicles and estimate direct samples which combined with a path reconstruction process allow to estimate the OD. The first approach, suited only to Motorways or Freeways, is based in an ad hoc adaptation of Kalman-Filter, combining elements from [5] and [6]; while the second, more appropriate for networks, uses elements from [3] generalizing methods for OD estimation based on license plate recognition.

In the first case a simulation experiment has been conducted, prior to the deployment of the technology in a forthcoming pilot project. The simulation emulates the logging and time stamping of a sample of equipped vehicles. Since data from equipped vehicles constitute a random sample of traffic data of significant size, the measured travel times can be used as real-time estimates of travel times for the whole population of vehicles. The availability of real-time travel time estimates makes possible a more efficient use of Kalman Filtering for OD estimates, simplifying the equations and replacing state variables by real-time measurements. We focused our attention on dynamic OD estimation in linear congested corridors where no route choice strategy is considered since there exists a unique path connecting each OD pair, but the travel time between each OD pair is considered and affected by congestion. We propose a space-state formulation for dynamic OD matrix estimation in corridors considering congestion that combines elements of Chang and Wu [6] and Van Der Zijpp and Hamerslag [5] proposals. A linear Kalman-based filter approach is implemented for recursive state variables estimation. Tracking the vehicles is assumed by processing Bluetoooth and WiFi signals by sensors located at the entry ramps (mandatory), in the main section (as many as possible) and the off-ramps (as many as possible). Traffic counts for every sensor and OD travel time from each entry ramp to the other sensors (main section and ramps) are available for any selected interval of length higher than 1 second. Then travel time delays between OD pairs or between each entry and sensor location are directly provided by the detection layout and should no longer be state variables but measurements simplifying the approach and making it more reliable. A basic hypothesis that requires a statistic contrast for real test site applications is that equipped and non equipped vehicles follow a common OD pattern. The state variables $b_{ij}(k)$, defined in terms of proportions of trips between OD pairs (i,j), are assumed to be stochastic in nature and evolve according with an independent random walk process whose state equation is: $\mathbf{b}(k+1) = \mathbf{Db}(k) + \mathbf{w}(k)$ $b(k)$ is the column vector of all feasible OD pairs (i,j), ordered by entry ramp, and $w_{ij}(k)$'s are independent Gaussian white noise sequences with zero mean and covariance matrix $\mathbf{Q}$. The state variables should additionally satisfy the structural

constraints
$$b_{ij}(k) \geq 0 \quad i = 1\ldots I, \quad j = 1\ldots J$$
$$\sum_{j=1}^{J} b_{ij}(k) = 1 \quad i = 1\ldots I$$

Let's denote by: $q_i(k)$ number of equipped vehicles entering the freeway from on-ramp i during interval k and i=1,...I; $s_j(k)$ number of equipped vehicles leaving the freeway by off-ramp j during interval k and j=1,...,J; $y_p(k)$ number of equipped vehicles crossing main section sensor p and p=1,...,P; $G_{ij}(k)$ number of vehicles entering the freeway at on-ramp i during interval k with destination to off-ramp j; $g_{ij}(k)$ number of equipped vehicles entering the freeway from ramp i during interval k that are headed towards off-ramp j; IJ = IxJ, number of feasible OD pairs depending on entry/exit ramp topology in the corridor; $t_{ij}(k)$ average measured travel time for equipped vehicles entering from entry i and leaving by off-ramp j during interval k; $t_{ip}(k)$ average measured travel time for equipped vehicles entering from entry i and crossing sensor p during interval k; $b_{ij}(k)=g_{ij}(k)/q_i(k)$ the proportion of equipped vehicles entering the freeway from ramp i during interval k that are destined to off-ramp j; e(k)=e a colum vector of dimension I containing ones; z(k) vector of observation variables during interval k; i.e. a column vector of dimension I+J+P, whose structure is $z(k)^T=[s(k), y(k), e(k)]^T$. Let's define $U_{ijq}^{h}(k)=1$ if the average measured time-varying travel time during interval k to traverse the freeway section from entry i to sensor q takes h time intervals, h = 1,...,M and q = 1,...Q and Q = J+P (the total number of main section and off-ramp sensors), and M the maximum number of time intervals required by vehicles to traverse the entire freeway section considering a high congestion scenario; the value is 0 otherwise. Let's also define:

E	:	Matrix of row dimension I containing 0 for columns related to state variables in time intervals $k-1,\ldots,k-M$ and **B** for time interval k.
B	:	Matrix of dimension I x IJ defining equality constraints (sum to 1 in OD proportions for each entry) for state variable in time interval k.
F(k)	:	Matrix of dimensions (1+M)IJ x (1+M)IJ consisting on diagonal matrices $f(k),\ldots,f(k-M)$ containing input on-ramp volumes. This applies to each OD pair and time interval. Each $f(.)$ is a squared diagonal matrix of dimension IJ.
g(k)	:	*Column vector* of OD flows of equipped vehicles for time intervals $k,k-1,\ldots,k-M$
A	:	Matrix of dimensions (J+P) x (1+M)(J+P) that adds up for a given sensor q (main section or off-ramp) traffic flows from any previous on-ramps arriving to sensor at interval k assuming their travel times are $t_{iq}(k)$

Defining the measurement equation as $z(k) = \begin{pmatrix} \mathbf{H(k)} \\ \mathbf{E} \end{pmatrix}\mathbf{b(k)} + \begin{pmatrix} v'(k) \\ 0 \end{pmatrix} = \mathbf{R(k)b(k)} + v(k)$ where

$v'_{ij}(k)$'s are independent Gaussian white noise sequences with zero mean and covariance matrix $\mathbf{R'}$.

The Kalman-Filter algorithm for the dynamic estimation of OD matrices in Motorway is:

KF Algorithm	:	Let K be the total number of time intervals for estimation purposes and M maximum number of time intervals for the longest trip
Initialization	:	$\mathbf{b}_k^k = \mathbf{b(0)}$ k=0; Build constant matrices and vectors: e, **A, B, D, E, R, W** where each time interval and each row is set to the maximum indetermination proportion $1/J_i$

Prediction Step	:	$$\mathbf{P_k^k} = \mathbf{V}[\mathbf{b(0)}]$$ $$\mathbf{b_{k+1}^k} = \mathbf{Db_k^k}$$ $$\mathbf{P_{k+1}^k} = \mathbf{DP_k^kD^T} + \mathbf{W}$$
Kalman gain computation	:	Get observations of counts and travel times: $q(k+1)$, $s(k+1)$, $y(k+1)$, $t_{ij}(k+1)$ $t_{ip}(k+1)$.
Filtering	:	Build $z(k+1)$, $\mathbf{F(k+1)}$, $\mathbf{U(k+1)}$. Build $\mathbf{R_{k+1}} = \mathbf{R(k+1)}$. Compute $\mathbf{G_{k+1}} = \mathbf{P_{k+1}^kR_{k+1}^T}\left(\mathbf{R_{k+1}P_{k+1}^kR_{k+1}^T} + \mathbf{R}\right)^-$ (where $(.)^-$ denotes the pseudoinverse) Compute $\mathbf{d_{k+1}} = \mathbf{G_{k+1}}\left(z(k+1) - \mathbf{R_{k+1}b_{k+1}^k}\right)$ filter for state variables and errors $\mathbf{\varepsilon_{k+1}} = \left(z(k+1) - \mathbf{R_{k+1}b_{k+1}^k}\right)$ Search maximum step length $0 \le \alpha \le 1$ such that $\mathbf{b_{k+1}^{k+1}} = \mathbf{b_{k+1}^k} + \alpha\,\mathbf{d_{k+1}} \ge 0$ $\mathbf{P_{k+1}^{k+1}} = \left(\mathbf{I} - \mathbf{G_{k+1}R_{k+1}}\right)\mathbf{P_{k+1}^k}$
Iteration	:	*k=k*+1 if *k*=K **EXIT** otherwise **GOTO Prediction Step**
Exit	:	*Print results*

In a benchmark conducted at a Toll Plaza of the Motorway Site the number of Bluetooth devices associated with vehicles was in average the 27.67%, that determined the significance of the sample used in the experiments. The increasing penetration of the technologiy guarantees larger samples in the near future. A pending task planned for the near future will be to determine the influence of the sample size in the accuracy of the results. A Set of computational experiments has been conducted with time sliced OD flows with time horizon split in four time intervals of 15 minutes and the demand accordingly distributed to account for the 15%, 25%, 35% and 25% of the total demand in each interval. The results can be summarized as follows: for time intervals where traffic flow varies from free flow to dense but not yet saturation conditions the filtering approach works as expected and its performance seems not affected as traffic flows become congested. RMSE values are of a similar order of magnitude, ranging in the interval [0.63, 6.35] ($\times 10^{-2}$). The convergence to the true values is quite satisfactory as prove the computational results in the full paper.

The second approach, intended for more general networks, has been based on the use of disaggregated flows, as in the case of the license plate recognition, [3]. The procedure works as follows: given a sample of equipped vehicles in the V2I scenario, their positions are tagged along their paths, there will be various classes, trips crossing the scenario at entry and exit tagged points as well as at intermediate positions, trips starting outside by a tagged entry point and ending inside, trips starting inside, leaving by a tagged exit point and ending a a destination outside the scenario, and finally trips starting and ending within the scenario, whose pats are tagged at intermediate points.

In all cases we have assumed that vehicles interact with the infrastructure at V2I sensors located in the border tagging all entries and exits and that there is a sensor layout in the network tagging the

vehicles at intermediate points in their routes. From the point of view of the observability, defined in terms of identifying if a set of available measurements is sufficient to estimate the state of a system [2], in the first approach the detection layout has been set up in such way that intercepts flows for all OD pairs in the Motorway section, and therefore satisfies the observability conditions. While in the second case it is guaranteed by a suitable design of the sensor layout [7], determined by the network topology and the identification of the most likely used paths between origins and destinations. This layout allows to collect a sample of the OD matrix, for each time interval, that can be expanded to the whole population as a function of a initial OD matrix and the rate of penetration of the technology. Computational results will be included in the full version.

References

[1] M. Ben-Akiva, M. Bierlaire, D. Burton, H.N. Kotsopoulos, R. Mishalani, (2001), Network State Estimation and Prediction for Real-Time Traffic Management, Networks and Spatial Economics, 1, pp. 293-318.

[2] E. Castillo, A. J. Conejo J. M. Menéndez, P. Jiménez, The Observability Problem in Traffic Network Models, Computer-Aided Civil and Infrastructure Engineering **23** (2008) 208–222

[3] E. Castillo, J.M. Menéndez, P. Jiménez, (2008), Trip matrix and path flow reconstruction and estimation based on plate scanning and link observations, Transportation Research Part B 42 455–481

[4] Mahmassani, H.S., and Zhou, X., (2005), Transportation System Intelligence: Performance Measurement and Real-Time Traffic Estimation and Prediction in a Day-to-Day Learning Framework, Chapter 16 in Advances in Control, Communication Networks, and Transportation Systems, In Honor of Pravin Varaiya, edited by E. Abed, Birkhauser,

[5] Van Der Zijpp, N.J. and Hamerslag, R., (1994). An improved Kalman filtering approach to estimate Origin—Destination matrices for freeway corridors. In: *Transportation Research Records 1443*, National Research Council, pp. 54–64.

[6] Chang, G.L. and Wu, J., (1994). Recursive estimation of time-varying Origin—Destination flows from traffic counts in freeway corridors. Transportation Research B Volume 28B No.2 pp. 141-160

[7] Gillieron, F. and Barceló, (2008) J. Optimization of Road Traffic Counts Location in a Network, Conference Proceedings (Paper 2728), ITS Europe 2008.

A Branch-and-Cut algorithm for the Multi Depot Multiple TSP

Enrique Benavent

Departamento de Estadística e Investigación Operativa

Universitat de València

Antonio Martínez

Departamento de Estadística e Investigación Operativa

Universitat de València

Enrique Benavent

Departamento de Estadística e Investigación Operativa

Universitat de València

C/ Dr. Moliner, 50, Burjassot, 46100, Valencia, Spain

Email: enrique.benavent@uv.es

1 Introduction

In this work, we study a variant of the very well known Traveling Salesman Problem (TSP), the Multi Depot Multiple Traveling Salesman Problem (MDMTSP) in which an unlimited number of salesmen have to visit a set of customers using routes that can be based in one of several available depots.

We study the polyhedron associated to the MDMTSP and present new valid inequalities that could be useful for other multi-depot problems. We introduce in Section 3 an integer formulation of the problem, the associated polyhedron and some results that allow obtaining facet inducing inequalities for the MDMTSP from certain facet inducing inequalities for the TSP polyhedron. Section 4 contains the main facet inducing inequalities for the MDMTSP, including two new families of inequalities that are specific for multi depot problems and have shown to be very effective in a cutting plane algorithm. This partial knowledge of the polyhedron is used to implement a Branch-and-Cut algorithm, presented in Section 5, which is able to solve instances with up to 279 customers and 25 depots.

2. The MDMTSP and related problems

Let J be a set of vertices representing customers and I a set of vertices representing possible depots. For each pair of vertices i, j such that $i \in I \cup J, j \in J$, we are given a travel cost (e.g. distance) c_{ij}. Distances are symmetric and are assumed to satisfy the triangular inequality. The MDMTSP consists of finding a set of routes such that each route contains exactly one depot, each customer belongs to one route and the total cost of the routes is minimized. A consequence of assuming the triangular inequality is that, although solutions with more than one route per depot are allowed, each depot will be contained in at most one route in any optimal solution. Therefore, the MDMTSP with only one depot is equivalent to the TSP, thus proving that the MDMTSP is NP-hard. Kara and Betkas [1] and Yadlapalli et al. [3] study some variants of the MDTSP but, as far as we know, no polyhedral study has been proposed for this problem.

Other multi depot problems include the Multi Depot Vehicle Routing Problem (MDVRP) and the Location Routing Problem (LRP), a much more general problem than the MDMTSP, which includes opening costs for the depots, and capacitated vehicles. A vast literature exists on the LRP, a recent survey is [2].

3. Formulation and first results on the polyhedron of the MDMTSP

The MDMTSP is formulated as an integer program using the following variables: for any $i, j \in J$, $x_{ij} = 1$ if edge (i, j) is used in the solution and $x_{ij} = 0$ otherwise; for any $i \in I, j \in J$, $x_{ij} \leq 2$ is the number of times that the edge (i, j) is used in the solution, where the value $x_{ij} = 2$ represents the case where a route based on depot i contains only customer j. Then the MDMTSP can be formulated as:

Min $\sum_{(i,j)\in E} c_{ij} x_{ij}$

s.t.

$$x(\delta(j)) = 2 \ \forall j \in J \tag{1}$$

$$x(\gamma(S)) \leq |S| - 1 \ \ \forall S \subseteq J \tag{2}$$

$$\sum_{i\in I'} x_{ij} + 2x(\gamma(S \cup \{j, l\})) + \sum_{k\in I\setminus I'} x_{kl} \leq 2|S| + 3 \ \text{ with } j, l \in J$$
$$S \subseteq J\setminus\{j, l\}, S \neq \emptyset, I' \subset I \tag{3}$$

$$\sum_{i\in I'} x_{ij} + 3x_{jl} + \sum_{k\in I\setminus I'} x_{kl} \leq 4 \ \ \ \forall j, l \in J, \ I' \subset I \tag{4}$$

$$x_{ij} \in \{0, 1, 2\} \ \ \forall i \in I \ \forall j \in J \tag{5}$$

$$x_{ij} \in \{0, 1\} \ \ \ \forall i \in J \ \forall j \in J \tag{6}$$

As usual, given a subset $S \subseteq J$, $\gamma(S)$ denotes the set of edges with both extremes in S, and $\delta(S)$ denotes the set of edges with only one extreme in S. Then, (2) are the degree constraints for the customers, (3) are the subtour elimination constraints, (3) and (4) avoid the existence of a path starting at one depot and ending at a different depot and are called path elimination constraints, and finally (5) and (6) are the integrality constraints.

The polyhedron associated to the MDMTSP, denoted P_{MDMTSP}, is defined as the convex hull of the vectors x that represent feasible solutions of the MDMTSP. We have shown that the only equalities satisfied by all the points of P_{MDMTSP} are the degree constraints (1), thus determining the dimension of the polyhedron. All the trivial inequalities, with the exception of inequalities $x_{ij} \leq 2$ for $i \in I, j \in J$, induce facets of P_{MDMTSP}.

Given a valid constraint for the TSP satisfying a mild condition, an *extended constraint* for the MDMTSP can be generated. Roughly speaking the extension consists of substituting a node by the set of depots. We show that if the TSP constraint is written in Tight Triangular form then the extended constraint is valid for the MDMTSP and that, if the TSP constraint is facet inducing for the TSP polyhedron, then the extended constraint is also facet inducing for P_{MDMTSP}, given that it is valid. This result provides a large number of facet inducing constraints for P_{MDMTSP} that come from the very well studied polyhedron of the TSP. Nevertheless, it should be remarked that all the depots are essentially treated as a single vertex in these constraints.

We also present a depot lifting theorem that allows to substitute a single depot by a set of depots in a facet inducing inequality for the P_{MDMTSP}, under certain conditions.

4. Facets of the P_{MDMTSP}

Using the extended constraints mentioned in the preceding Section, it can be shown that the subtour elimination constraints (2) and the very well known comb constraints for the TSP are facet inducing for P_{MDMTSP}. Comb constraints, in particular, lead to several families of facet inducing inequalities for P_{MDMTSP} depending on where the set of depots are located: outside the comb, in the intersection of a tooth and the handle, in a tooth outside the handle, or inside the handle but not in any tooth.

Path elimination constraints (3) and (4) are also facet inducing of P_{MDMTSP}. Finally, we have introduced two new families of constraints for P_{MDMTSP} that can be called multi-depot comb constraints because their structure is similar to that of the TSP combs but are specific for multi-depot problems. In these constraints, not all the depots are located in the same part of the comb structure. Thus, in the first family, some depots are inside the handle but outside any tooth, and some depots are outside the comb. In the second family, the handle contains no depot but each tooth contains at least one depot. The number of teeth can be even in these last inequalities. These multi-depot combs have been shown to be facet inducing for the P_{MDMTSP} and they have been very useful in the Branch-and-Cut algorithm described below. They cut some undesirable fractional solutions where some depots have very small positive degree (note that, if a depot is used, its degree must be at least equal to two).

4. Branch-and Cut

We have implemented a Branch-and-Cut algorithm to solve the MDMTSP based on the linear relaxation of formulation of the problem and the families of valid constraints presented in the preceding Section. The separation procedures used to identify constraints violated by the current fractional solution are as follows. Subtour and TSP comb constraints are separated using some common procedures also used in the TSP; for the path elimination constraints, we have used a very simple heuristic; finally, some heuristic procedures have been implemented to separate multi-depot comb constraints. We have used the LP solver Cplex 9.0 and the *strong branching* strategy to explore the branch-and-cut tree.

The Branch-and-Cut algorithm was tried on two sets of instances. The first set contains 30 instances taken from the LRP literature discarding the capacities and opening costs of the depots; this set contains instances with up to 200 customers and 10 depots. All the instances in this set were solved to optimality in less than one minute except for the instances with 200 customers, which were solved in a few minutes. It is also remarkable that 18 instances were solved at the root node. The second set of eight instances was generated from two large TSP instances, with 127 and 280 customers respectively, by substituting by depots some randomly selected customers. From each TSP instance, four MDMTSP instances were generated with 1, 5, 10 and 25 depots, respectively. The results obtained with these instances were similar to the preceding ones but one instance with 255 customers and 25 depots could not be solved with a time limit of 30 minutes.

References

[1] I. Kara and T. Betkas, "Integer programming formulations of multiple salesmen problems and its variants", *European Journal of Operational Research* 174, 1449-1458 (2006).

[2] G. Nagy and S. Salhi, "Location-routing: Issues, models and methods", *European Journal of Operational Research* 177, 649-672 (2007).

[3] S. Yadlapalli, W.A. Malik, S. Darbha and M. Pachter, "A lagrangean-based algorithm for a Multiple Depot, Multiple Traveling Salesman Problem", *Nonlinear Analysis: Real World Applications* 10, 1990-1999 (2009).

On Negative Correlations and the Consistency of GEV-based Discrete Choice Models

Eran Ben-Elia*

Centre for Transport and Society
Faculty of Environment and Technology
University of the West of England
Email: eran.ben-elia@uwe.ac.uk

Tomer Toledo and Joseph N. Prashker

Transportation Research Institute,
Faculty of Civil and Environmental Engineering,
Technion - Israel Institute of Technology

*corresponding author

Extended Abstract

Introduction

The science of choice modelling has flourished in the last years as more and more studies are made in order to better understand human decision making. Discrete choice models form a major part in econometric studies in general and travel based studies in particular. Econometric random utility based discrete-choice models are still regarded as the main workhorse for most travel related behavioural modelling. Consequently, random utility models (RUM) have been developed considerably in the past three decades [1]. The wide RUM family includes three main sub-family types: The Multinomial Logit Model (MNL) and its applications, The GEV (General Extreme Value) models and the different mixed models. Despite the improvements in more sophisticated modelling specifications, MNL and the family of GEV models are still those most frequently applied in practical applications involving planning, forecasting and feasibility assessments. However, GEV models are based on a set of specific mathematical properties one of which is non-negativity in unobserved correlations. In reality, there is no fundamental reason why non-positive correlations should not occur.

The generalized extreme value (GEV) theory was, developed by [2] to accommodate these deficiencies of MNL. This general theorem consists of a large family of specifications that includes in addition to MNL itself, also the different nest-based logit models: nested logit (NL), pair combinatorial logit (PCL), cross-nested logit (CNL) and generalized nested logit (GNL) models. GEV models are derived under a set of several restricting assumptions. These conditions are sufficient to observe a continuous multivariate extreme value distribution function. However, as noted by [3], these constraints also imply that the correlations in unobserved factors (or error terms) reproduced by a GEV model are necessarily always positive.

The inherited assumption of non-negative correlations is brought about by mathematical necessities. However, from a behavioural perspective, within elaborate nested structures, there is no apparent reason why this assumption must always hold. Therefore, we decided to put this to the test by creating artificial correlation

structures by generating synthetic data and estimating GEV models – NL (Experiment I) and CNL (Experiment II) to measure the obtained bias between estimated parameters and true parameters. In this context it is appropriate to use synthetic data generated with a postulated model, since the true parameter values are known in advance. As explained in the next subsection, in order to validate the results, the same models were also estimated using a Multinomial Probit (MNP) specification.

Experiment Design

In Experiment 1, a sample of 100 files (runs) each with 10,000 synthetic choice observations was created using MATLAB. The sample was created separately for two choice problems: a choice between three alternative and a choice between four alternatives. Each file contained the deterministic utility for each alternative and the error components. The synthetic utilities – both the deterministic and stochastic parts were computed using a random normal distribution. 21 artificial 'true' correlation values (ρ_k) were assumed ranging from -0.95 to +0.95. For each correlation (ρ_k), a variance-covariance matrix was computed. For the three-alternative case the covariance matrix is:

$$\text{cov}_k = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & \rho_k \\ 0 & \rho_k & 1 \end{bmatrix}, k = 1,...,21$$

In the case of the four-alternative choice set the variance-covariance matrix was defined separately for positive and negative correlations.

$$\text{cov}_k = \left\{ \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & \rho_k & \rho_k \\ 0 & \rho_k & 1 & \rho_k \\ 0 & \rho_k & \rho_k & 1 \end{bmatrix} \text{if } \rho_k \geq 0, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & -\rho_k & \rho_k \\ 0 & -\rho_k & 1 & \rho_k \\ 0 & \rho_k & \rho_k & 1 \end{bmatrix} \text{if } \rho_k < 0 \right\}$$

To create the correlation between alternatives the vector of errors was multiplied by the Cholesky factor of each of the 21 covariance matrices. This procedure created the vectors of correlated error terms. The chosen alternative was the one with the maximum utility. In this way, a corresponding vector of choices was matched for each 'true' correlation value and in total 21 choice vectors in each file.

NL models were estimated with BIOGEME 1.4 [4] for each of the 21 choice vectors in each of the 100 data sets (in total - 2100 models). The NL model had a common nest which included all alternatives apart for one. The estimated correlations of the NL models were compared to the estimated values of an equivalent MNP model. The MNP model was estimated with GAUSS software using numeric integration procedures. The restrictions imposed on MNP model included setting the variance of all the alternatives equal to 1.

In experiment 2 the choice was only between three alternatives. The artificial correlations were derived from the combinations of the values (0.75, 0.25, -0.25, 0.75) in groups of three. In total *k=20* combinations were created. For example the combination (0.75, 0.75, 0.75) is the first, (0.75, 0.75, 0.25) the second, etc. The covariance matrix was defined as:

$$\text{cov}_k = \begin{bmatrix} 1 & \rho_k^{12} & \rho_k^{13} \\ \rho_k^{12} & 1 & \rho_k^{23} \\ \rho_k^{13} & \rho_k^{23} & 1 \end{bmatrix}, \quad k = 1,...,20$$

Since In five out of the 20 combinations the matrix is not semi positive-definite, the Cholesky factorization is invalid. This fact reduced the number of combinations from 20 to 15. The choice vectors were then obtained in the same manner as experiment 1. A CNL model was estimated with BIOGEME 1.4 [4] for each of the 15 choice vectors in each of the 100 data sets (in total – 1,500 models). The CNL model had a PCL specification of three alternatives, whereby each alternative has a shared nest with each of the other two alternatives. In order to facilitate the computations, the estimated correlations of the CNL model were computed using Papola's approximation [5]. We note the correct computation is provided by [6] but its estimation requires simulation. However, Papola's approximation provides a conservative estimate. The estimated correlations of the CNL model were also compared to the estimates of an equivalent MNP model.

Results

In both experiments the estimated correlations for the MNP model were basically identical to the true values. In experiment 1, for three-alternatives - no real difference could be seen between the estimate (average of 100 data sets) and the true value. A t-test for difference confirms these results. For four-alternatives - a different picture can be seen compared to the results of the three-alternative model. For positive correlations there appears no real difference between the results of the NL model and the true values. However, for negative correlations there is a gap between the true values and the estimation. This difference is statistically significant.

In experiment 2, the PCL model did not provide comparable results to the MNP. We can differentiate between several cases. First, when the true correlations are equal and positive the PCL model is identified and the correlations are also equal. However the estimates are biased. Second, when the true correlations are unequal and positive, the PCL model is identified. Although the estimated correlations are still biased the relative sizes are comparable. Third, when the true correlation is negative but the correlation of utility differences remains positive the PCL is identified. However, the model over estimates the positive correlation, while the negative correlation always has a positive estimate. Fourth, when the correlations are equal and negative, the correlation of utility differences is also positive. However, the PCL model estimates the correlations as positive and the estimates are similar to the first case. Fifth, when the correlation of utility differences has a negative true value, the PCL model is not identified.

Remarkably despite the bias in the estimated correlations, in both experiments no bias was observed for the coefficients of the utility functions and the alternative specific constants. A fact also corroborated by the MNP estimates which were almost identical.

Conclusions

These results are not surprising but the fact that most modelers are ambiguous or unaware to the severity of the problem requires rethinking of the almost automatic adoption of GEV-based specifications to choice modeling problems. Caution should be used and if possible it is recommended that more flexible models specifications should be applied, particularly when the level of uncertainty regarding the used data is high. In practical rather then research endeavors, there is still limited use of flexible modeling specifications like MNP or the family of Mixed Models like Mixed Logit [7-10]. Hopefully this limited study which elucidates some of the problematic features in commonly used discrete choice models, will motivate more research into this seemingly overlooked issue in the choice modeling community.

References

1. Ben Akiva, M. and S.R. Lerman, *Discrete Choice Analysis: Theory and Application in Travel Demand*. 1985, Cambridge, MA: MIT Press.
2. McFadden, D., *Modeling the choice of residential location*, in *Spatial Interaction Theory and Residential Location*, A. Karlqvist, et al., eds. 1978: Amsterdam. p. 75-96.
3. Dagsvik, J.K., "How large is the class of generalized extreme value random utility models?" *Journal of Mathematical Psychology*, 1995. 39: p. 90-98.
4. Bierlaire, M. *BIOGEME: a free package for the estimation of discrete choice models*. in *3rd Swiss Transportation Research Conference*. 2003. Ascona, Switzerland.
5. Papola, A., "Some developments of the Cross- Nested Logit model". *Tranportation Research B*, 2004. 38: p. 833-851.
6. Abbe, E., M. Bierlaire, and T. Toledo, "Normalization and correlation of Cross-Nested Logit models". *Tranportation Research B*, 2007. 41: p. 795-808.
7. Ben-Akiva, M. and D. Bolduc, *Multinomial probit with a Logit kernel and a general parametric specification of the covariance structure*. 1996, Massachusetts Institute of Technology.
8. McFadden, D. and K. Train, "Mixed MNL models of discrete response". *Journal of Applied Econometrics*, 2000. 15: p. 447-470.
9. Train, K., *Mixed logit models for recreation demand*, in *Valuing Recreation and the Environment.*, J. Herriges and C. Kling, eds. 1999, Edward Elgar: Northampton, MA.
10. Walker, J.L., *Extended Discrete Choice Models: Integrated Framework, Flexible Error Structures, and Latent Variables*, in *Department of Civil and Environmental Engineering*. 2001, Massachusetts Institute of Technology.

Optimization of an Order–Up–To Level Policy in an Inventory Routing Problem with Stock–Outs

Adamo Bosco

DEIS, University of Calabria

Francesca Guerriero

DEIS, University of Calabria

Demetrio Laganà

DEIS, University of Calabria

Luca Bertazzi

Department of Quantitative Methods

University of Brescia, Contrada Santa Chiara 50, Brescia, Italy

Email: bertazzi@eco.unibs.it

Since the pioneering paper by Harris (1913) [8], several optimization models facing the integration of different types of logistic costs have been proposed. One of the most common problems is the one in which both transportation and inventory costs are taken into account. The aim is to determine shipping policies that allow to minimize the sum of these two costs. This is an interesting problem as the transportation cost raises and the inventory cost drops when shipments are performed frequently, while the contrary happens when shipments are rare over time.

From a mathematical modeling point of view, the integration of vehicle routing with inventory control problems has led to the development of the so-called inventory routing models, that have been proposed and analyzed in different papers. For an in-depth overview of this area of research, the reader is referred to [7], [5], [6], [12] and [4]. Few papers have been devoted to inventory routing problems with stochastic demand. [9] study the repeated distribution of a commodity over a long time horizon to several customers. Several satellite facilities can be visited by the drivers to refill their vehicles. In case of stockout, a direct delivery is made and a penalty cost is incurred. An incremental cost approximation in a rolling horizon framework is proposed to minimize the total expected delivery costs. [10] and [11] formulate the inventory routing problem as a Markov decision process and propose approximation methods to find good solutions with reasonable computational effort. [1] propose a new approach that approximates the future costs of current actions using optimal dual prices of a linear program.

In this work, we study an inventory routing problem, in which each retailer defines a maximum inventory level and has to satisfy either a deterministic or a stochastic demand over a given time horizon. An order-up-to level policy is applied, i.e. the quantity sent to each retailer is such that its inventory level reaches the maximum level, whenever the retailer is served. This policy has been introduced for the case with deterministic demand by [3]. An inventory cost is charged if the inventory level is positive. Instead, whenever the inventory level is negative, a penalty cost is charged and the excess demand is not backlogged. Shipments from the supplier to the retailers can be performed by one vehicle of given capacity at each discrete time instant. The problem is to determine a shipping policy that minimizes the total cost, given by the sum of the total inventory cost at the retailers and of the total routing cost.

We first study the deterministic version of this problem, which is a generalization of the problem studied in [2], as no stock–out at the retailers is allowed and a deterministic supply is assumed at the supplier in that paper. We formulate a mixed–integer linear programming model and implement a branch–and–cut algorithm. Since for each time instant not only the set of retailers to serve, but also the set of retailers with stock–out have to be selected, the model is very different from the one in [2] and more difficult to solve. We propose a set of new valid inequalities to solve instances of comparable dimension.

This deterministic version of the problem is used to compute a benchmark policy for the stochastic version of the problem, by setting the demand equal to the average demand. Then, a dynamic programming formulation of the problem with stochastic demand and a rollout algorithm for the solution of this problem are proposed. Rollout algorithms are a class of heuristic algorithms that can be used to solve deterministic and stochastic dynamic programming problems. The basic idea is to use the cost obtained by applying a heuristic, called base policy, to approximate the value of the optimal cost-to-go in a one-step lookahead policy. These algorithms are very appealing from the practical point of view, as they are easy to be implemented and guarantee a no worse, and usually much better, performance than the corresponding base policy. The rollout algorithm we propose is based on an approximate cost–to–go, an approximate Q-factor and an approximate set of controls. The approximate cost–to–go is obtained as follows: For each state at each time, we fix the future demands of each retailer to the corresponding average value. Then, we compute the approximate cost–to–go by solving the deterministic problem from the next time instant to the end of the time horizon. The approximate Q-factor is computed by generating a given number of different trajectories of demand. Each trajectory is obtained by generating the current demand of each retailer from the corresponding probability distribution and by setting the demand of each retailer from the next time instant to the end of the time horizon equal to the corresponding average value. For each different trajectory, we compute the total cost and the frequency at which it occurs and we set the probability of the trajectory equal to the corresponding relative frequency. Finally,

we normalize the probabilities and compute the approximate Q–factor by taking the expectation over the different trajectories. The approximate set of controls associated to each state at each time is built by an iterative procedure in which the deterministic problem is solved.

The performance of the branch-and–cut and the rollout algorithms is evaluated on the basis of a large set of problem instances. The computational results show that, although the problem is more difficult than the one in [2], the branch–and–cut algorithm is effective and that the rollout algorithm significantly outperforms its base policy.

References

[1] Adelman, D. (2004), A Price-directed Approach to Stochastic Inventory/Routing, *Operations Research* 52, 499-514.

[2] Archetti, C., Bertazzi, L., Laporte, G. and Speranza, M.G. (2007), A branch-and-cut algorithm for a vendor managed inventory routing problem, *Transportation Science* 41, 382–391.

[3] Bertazzi, L., Paletta, G., Speranza, M.G. (2002), Deterministic Order-up-to Level Policies in an Inventory Routing Problem, *Transportation Science* 36, 119-132.

[4] Bertazzi, L., Savelsbergh, M. and Speranza, M.G. (2008), Inventory Routing, in *The Vehicle Routing Problem, Latest Advances and New Challenges*, Golden, B., Raghavan, S. and Wasil, E., (eds.), 49–72, Springer.

[5] Campbell, A.M., Clarke, L., Kleywegt, A. and Savelsbergh, M.W.P. (1998), The Inventory Routing Problem, in: *Fleet Management and Logistics*, T.G. Crainic and G. Laporte (eds), 95–113, Kluwer, Boston.

[6] Cordeau, J.-F., Laporte, G., Savelsbergh, M.W.P., Vigo, D. (2007), Short-Haul Routing, in *Handbooks in Operations Research and Management Science: Transportation* G. Laporte and C. Barnhart (eds.), vol. 14, 367–428.

[7] Federgruen, A. and Simchi–Levi, D. (1995), Analysis of Vehicle Routing and Inventory–Routing Problems, in: *Handbooks in Operations Research and Management Science* (Ball, M.O., Magnanti, T.L., Monma, C.L. and Nemhauser, G.L., eds.), Vol. 8, 297–373, North–Holland.

[8] Harris, F.W. (1913), How Many Parts to Make at Once, *Factory, The Magazine of Management* 10, 135–136, reprinted in *Operations Research* 38, 947–950, 1990.

[9] Jaillet, P., Bard, J.F., Huang, L. and Dror, M. (2002), Delivery Cost Approximations for Inventory Routing Problems in a Rolling Horizon Framework, *Transportation Science* 36, 292–300.

[10] Kleywegt, A.J., Nori, V.S. and Savelsbergh, M.W.P (2002), The Stochastic Inventory Routing Problem with Direct Deliveries, *Transportation Science* 36, 94-118.

[11] Kleywegt, A.J., Nori, V.S. and Savelsbergh, M.W.P (2004), Dynamic Programming Approximations for a Stochastic Inventory Routing Problem, *Transportation Science* 38, 42–70.

[12] Moin, N.H. and Salhi, S. (2007), Inventory Routing Problems: A Logistical Overview, *Journal of Operational Research Society* 58, 1185–1194.

The Split Delivery Capacitated Team Orienteering Problem

C. Archetti

Dipartimento Metodi Quantitativi

Università degli Studi di Brescia, Brescia, Italy

A. Hertz

Département de Mathématiques et de génie industriel

Ecole Polytechnique and GERAD, Montréal, Canada

M.G. Speranza

Dipartimento Metodi Quantitativi

Università degli Studi di Brescia, Brescia, Italy

N. Bianchessi

Dipartimento Metodi Quantitativi

Università degli Studi di Brescia, C.da S. Chiara, 50, I-25122 Brescia, Italy

Email: bianche@eco.unibs.it

1 Extended abstract

In most of the routing problems addressed in the literature, the set of customers to serve is known in advance [9]. Only recently, among the researchers, it is growing up the interest for routing problems where a non-negative value is associated with each customer and gained when the customer is visited. In these problems, called routing problems with profits, it is necessary to select the subset of customers to visit which allows us to optimize the objective function while satisfying the given constraints. In general, two conflicting objectives should be taken into account: the maximization of the collected profit and the minimization of the traveled distance. Nevertheless, a very few works in the literature consider the true bicriteria objective of the problems, whereas, in the greatest part of the studies, these problems are addressed in a single-criterion version. See [8] for a recent survey on routing problems with profits.

In this work, we consider the Capacitated Team Orienteering Problem (CTOP). In particular,

here we investigate a variant of the problem arising when split deliveries, that is possible multiple visits to customers, are allowed. Split deliveries have been first taken into account in [6, 7] while addressing the classical Vehicle Routing Problem (VRP). Then, in these very last years, the Split Delivery Vehicle Routing Problem (SDVRP) has received a lot of attention, mainly due to the savings that can be achieved performing split deliveries, savings that can reach 50% of the cost [3]. Allowing splitting the deliveries could have an impact also on the CTOP, an interesting issue to study both from a theoretical and a computational point of view.

Let us first recall the Capacitated Team Orienteering Problem. The problem can be defined over a complete directed graph $G = (V, A)$, where $V = \{0, \ldots, n\}$ is the set of vertices and A is the set of arcs. Vertex 0 is the starting and ending point of each route and vertices $i \in V' = V \setminus \{0\} = \{1, \ldots, n\}$ represent potential customers. An arc $(i, j) \in A$ represents the possibility to travel from vertex $i \in V$ to vertex $j \in V$. With each customer $i \in V'$ a non-negative profit p_i and a non-negative demand d_i are associated. A travel time t_{ij} is given for each arc $(i, j) \in A$. The travel time matrix (t_{ij}) is supposed to satisfy the triangular inequality. A fleet of m identical vehicles is available to serve the customers. Each vehicle has a limited capacity Q, starts and ends its route at vertex 0 and can visit any subset of the customers within a given time limit T_{max}. The profit of each customer $i \in V'$ can be collected by one vehicle at most. The objective is to maximize the total collected profit while satisfying the time limit and the capacity constraints for each vehicle. This problem has been recently studied in [2] where the authors proposed an exact method as well as effective heuristic procedures.

The Split Delivery Capacitated Team Orienteering Problem (SDCTOP) is a relaxation of the CTOP where a customer may be served by one or several vehicles. At each visit to a customer $i \in V'$, an amount $\bar{d}_i \leq d_i$ can be delivered to the customer and a proportional part of the profit $(p_i \bar{d}_i)$ is collected (with respect to the notation previously introduced, here p_i has to be interpreted as a unitary profit). The sum of the partial amounts delivered to a customer cannot be greater than the demand.

Based on the concept of *k-split cycle* defined in [6], some important properties can be proved to hold for an optimal solution to the SDCTOP. In particular, if the travel time matrix satisfies the triangular inequality, then it can be shown there exists an optimal solution where no two routes have more than one customer in common. In turn, this implies that there exists an optimal solution such that at most one arc is traversed for each pair of reverse arcs between two customers. Moreover, a property can be proved which allows us to relate the number of splits to the number of routes in an optimal solution. Let n_i be the number of routes that visit customer $i \in V'$. We say that customer $i \in V'$ receives a split delivery if $n_i > 1$, with $n_i - 1$ equal to the number of splits at customer i. If the cost matrix satisfies the triangular inequality, then it can be shown there exists an optimal solution to the SDCTOP where the total number of splits $(\sum_{i \in V'} n_i - 1)$

is less than the number of routes.

Concerning the gain that can be achieved by allowing split deliveries, let z(P) denote the optimal value of problem P. It can be seen how the gain results to be unbounded when $d_i > Q$ for some $i \in V'$, whereas, given the properties previously outlined, if $d_i \leq Q$ for all $i \in V'$, then $\frac{z(\text{SDCTOP})}{z(\text{CTOP})} \leq 2$ can be proved, and the bound can be shown to be tight.

In order to solve the problem and to experimentally evaluate the impact of allowing split deliveries, an exact approach, based on column generation, is devised. The approach is similar to the one proposed in [1] for the SDVRP. Differences come firstly from the presence of global constraints which impose an upper bound on the quantity deliverable to each customer. Thus, when a branching rule is applied enforcing the visit of a customer, attention must be paid in order to constraint the customer to be fully served. Then, the capacity cut constraints considered in [1] to separate an optimal fractional solution, are no more valid here and the exact approach reduces to a pure branch-and-price algorithm. Finally, the speed of convergence of the column generation process has been accelerated improving the heuristic methods used to address the pricing subproblem.

The algorithm is tested on a new set of benchmark instances derived from some VRP instances presented in [5], the same considered in [2]. The technique adopted to define the new instances is the one used in [4] to derive benchmark instances for the SDVRP. For each initial VRP instance we generate six new instances in which the coordinates, the fleet size and the vehicle capacity are kept unchanged, whereas the customer demand is generated according to six scenarios ($[0.01-0.1]$, $[0.1-0.3]$, $[0.1-0.5]$, $[0.1-0.9]$, $[0.3-0.7]$, $[0.7-0.9]$). The demand of a customer in scenario $[\alpha - \beta]$ is randomly generated from a uniform distribution on the interval $[\alpha Q, \beta Q]$. Bounding and non-bounding values for T_{max} are considered. Preliminary computational results show the effectiveness of the solution approach and give an empirically evidence that splitting the deliveries can yield substantial gain.

References

[1] C. Archetti, N. Bianchessi and M. G. Speranza, "A Column Generation Approach for the Split Delivery Vehicle Routing Problem", Quaderno di Ricerca n. 329, Dipartimento Metodi Quantitativi, Università degli Sudi di Brescia, Italy, 2009.

[2] C. Archetti, D. Feillet, A. Hertz and M. G. Speranza, "The capacitated team orienteering and profitable tour problem", *Journal of the Operational Research Society* doi:10.157/palgrave.jors.2602603 (2008).

[3] C. Archetti, M. W. P. Savelsbergh and M. G. Speranza, "Worst-case analysis for split delivery vehicle routing problems", *Transportation Science* 40, 226–234 (2006).

[4] J. M. Belenguer, M. C. Martinez and E. Mota, "A Lower Bound for the Split Delivery Vehicle Routing Problem", *Operations Research* 48, 801–810 (2000).

[5] N. Christofides, A. Mingozzi and P. Toth "The Vehicle Routing Problem", in *Combinatorial Optimization*, N. Christofides, A. Mingozzi, C. Sandi and P. Toth (eds), 315–338, Wiley, Chichester, 1988.

[6] M. Dror and P. Trudeau, "Savings by split delivery routing", *Transportation Science* 23, 141–145 (1989).

[7] M. Dror and P. Trudeau, "Split delivery routing", *Naval Research Logistics* 37, 383–402 (1990).

[8] D. Feillet, P. Dejax and M. Gendreau, "Traveling Salesman Problems with Profits", *Transportation Science* 39, 188–205 (2005).

[9] P. Toth and D. Vigo, *The Vehicle Routing Problem*, SIAM Monographs on Discrete Mathematics and Applications , Philadelphia (2002).

Regulating HazMat Transportation by Toll-Setting: a Game Theory Approach

Lucio Bianco   **Massimiliano Caramia**   **Stefano Giordani**

Veronica Piccialli

Dipartimento di Ingegneria dell'Impresa

Università di Roma "Tor Vergata", Viale del Politecnico 1, 00133 Rome, Italy

Email: giordani@disp.uniroma2.it

1   Introduction

Hazardous materials (hazmats) transportation presents extremely typical characteristics due to the risk associated with its accidental release during a trip. Due to the potential destructive effect of accidents related to this kind of transportation, the public opinion is very sensitive to the possible dangers of hazmat shipments. Consequently, the governments in the different countries recently have put much more attention to this problem than in the past and have encouraged research on this field.

In particular, one of the main stream of research in hazmat transportation focus on shipments planning ([4]). A main aspect of this problem is the routing of hazmat shipments, that involves a selection among the alternative paths between origin-destination pairs. From the carrier's perspective, shipment contracts can be considered independently and a routing decision needs to be made for each shipment, on the basis of the cost associated to the transportation. At the macro level, hazmat routing is a "many to many" routing problem with multiple origins and destinations: the global route planning must be under the jurisdiction of a government authority, that keeps into account the risk induced by hazmat transportation over the population and the environment. For example, an authority should aim at the minimization of the total risk and/or promote equity in the spatial distribution of risk. The latter becomes crucial when certain populated zones are exposed to intolerable levels of risk as a result of the carriers' routing decisions.

Typically, a government authority does not have the right to impose specific routes to individual carriers. In literature, two main approaches have been proposed in order to control the hazmat flows. The first stream of research assumes that the government has the authority to close certain road segments to hazmat vehicles or to limit the amount of hazmat traffic flow on the links, and

this leads to the study of hazmat transportation network design (see, e.g., [5, 2, 3, 8, 1]).

Recently, in [6] a different approach has been proposed. The authors assume that the authority controls the hazmat flows on the network by introducing tolls on the network links in such a way to force the carriers to choose routes that minimize the total risk on the network. In this work, as in [6], we use the toll setting as an instrument to control the behavior of the carriers. However, our model not only aims to minimize the total risk on the network but also to descourage congestion on some links, in such a way to keep into account risk equity.

To achieve this result, we assume that the toll on a certain link depends on the total risk on that link and this implies an influence of the choices of all the carriers on the cost of the single carrier, and this naturally leads to a Nash game. Next, we give a formal description of the proposed model to implement the previous idea.

2 Model description

Given a road transportation network where hazmat is shipped, let it be represented by a directed graph $G = (V, A)$ where V is the set of n nodes and A is the set of m arcs (links) of the network. We consider the following scenario: there are N carriers and each carrier $k = 1, \ldots, N$ transports one hazardous material $h(k) \in H$, with H being the set of hazardous materials shipped by the carriers. We assume that each carrier has to satisfy a single shipment order, so that carrier k has a single commodity and a single origin-destination pair (s^k, t^k), and has variables x_a^k equal to the amount of material sent on the arc $a \in A$ of the network. The total amount of material that carrier k has to ship is denoted by r^k. Given the transportation cost c_a^k per unit amount of hazmat shipped by carrier k on arc a, each carrier aims at minimizing his own transportation cost $\sum_{a \in A} c_a^k x_a^k$. Therefore, given the network, the subproblem of each carrier k is a minimum cost single commodity flow problem that reduces to find the minimum cost route from s^k to t^k. If the risk was not considered each carrier could choose therefore independently his own (minimum cost) route. However, shipping an amount x_a^k of hazmat $h(k)$ induces a risk on arc a that we assume equal to $\rho_a^{h(k)} x_a^k$, where $\rho_a^{h(k)}$ is the risk induced on arc a by a single unit of material $h(k)$. Therefore the total risk on arc a is equal to $\sum_{l=1}^N \rho_a^{h(l)} x_a^l$. We assume that the authority imposes to carrier k a toll for shipping hazmat $h(k)$ along arc a, which is proportional to the risk $\rho_a^{h(k)} x_a^k$ induced on a by the amount x_a^k of hazmat $h(k)$ that carrier k transports along that arc. The unitary toll imposed by the authority on arc $a \in A$ depends on the hazmat $h(k)$ shipped by carrier k, and is a function $\tau_a^{h(k)} \left(\sum_{l=1}^N \rho_a^{h(l)} x_a^l \right)$ of the amount of the total risk on that arc: in particular, we assume that $\tau_a^{h(k)}$ is a linear function of the total risk on the arch, i.e., $\tau_a^{h(k)} \left(\sum_{l=1}^N \rho_a^{h(l)} x_a^l \right) = t_a^{h(k)} + d_a^{h(k)} \sum_{l=1}^N \rho_a^{h(l)} x_a^l$, where $t_a^{h(k)}$ and $d_a^{h(k)}$ are the coefficients of the linear function, and are controlled by the authority. Hence, the additional toll cost for carrier k on arc a depends also on the amount of hazmat shipped

by the other carriers along that arc, since the toll depends on the arc total risk. Therefore, the carriers' problem constitutes a Nash game (see, e.g., [7]) where each carrier k is a player and his subproblem has the following form

$$\min \quad \sum_{a \in A} c_a^k x_a^k + \sum_{a \in A} t_a^{h(k)} \rho_a^{h(k)} x_a^k + \sum_{a \in A} d_a^{h(k)} \left(\sum_{l=1}^{N} \rho_a^{h(l)} x_a^l \right) \rho_a^{h(k)} x_a^k$$

$$\sum_{a \in A^+(i)} x_a^k - \sum_{a \in A^-(i)} x_a^k = \begin{cases} r^k & i = s^k \\ 0 & \forall i \in V \setminus \{s^k, t^k\} \\ -r^k & i = t^k \end{cases} \tag{1}$$

$$x_a^k \geq 0 \quad \forall a \in A,$$

where $A^+(i) = \{(i,j) \in A : j \in V\}$ and $A^-(i) = \{(j,i) \in A : j \in V\}$ are the set of outgoing arcs and the set of incoming arcs, respectively, of node i.

We study theoretical properties of this model, i.e., the existence of an equilibrium, and the conditions on the unitary toll functions $\tau_a^{h(k)} \left(\sum_{l=1}^{N} \rho_a^{h(l)} x_a^l \right)$ ensuring the uniqueness of the equilibrium. Moreover, we propose a technique to evaluate the minimum toll level that the authority has to impose to the carriers in order to achieve a given target level of the network total risk and enforcing also risk equity.

The complete model is a bi-level optimization problem, where we assume that there is a leader (the authority) that chooses the coefficients $t_a^{h(k)}$ and $d_a^{h(k)}$ of the unitary toll functions on each arc in order to minimize a combination of risk magnitude and carrier travel cost, and the followers (the carriers) are the players of the game, where each player k (with $k = 1, \ldots, N$) aims to solve subproblem (1). In particular the leader aims to minimize the risk magnitude by minimizing the total risk of the network, and the maximum total link risk ϕ on the arcs of the network. In this situation, we get the following mathematical programming with equilibrium constraint (MPEC) problem:

$$\min_{t_a^h, d_a^h, \phi} \left\{ \alpha \sum_{k=1}^{N} \sum_{a \in A} \rho_a^{h(k)} x_a^k + \beta \cdot \phi + \right.$$

$$\gamma \left(\sum_{k=1}^{N} \sum_{a \in A} (c_a^k + t_a^{h(k)} \rho_a^{h(k)}) x_a^k + \sum_{k=1}^{N} \sum_{a \in A} d_a^{h(k)} \left(\sum_{l=1}^{N} \rho_a^{h(l)} x_a^l \right) \rho_a^{h(k)} x_a^k \right) \right\}$$

$$\sum_{l=1}^{N} \rho_a^{h(l)} x_a^l \leq \phi, \quad \forall a \in A$$

$$t_a^h \geq 0, d_a^h \geq 0, \quad \forall a \in A, \forall h \in H \tag{2}$$

$$\text{s.t. } \boldsymbol{x} \text{ is a Nash Equilibrium of (1)}$$

where α, β and γ are weights measuring the relative importance of total network risk, maximum total link risk and total (carriers') cost.

3   Experimenting the model

We study the properties of this model, reformulate it as a single optimization problem by using the optimality conditions for problem (1), and define a heuristic algorithm for solving it. We test the proposed model on some real instances, and, in order to evaluate the obtained results, we compare our solution with the one found by the method defined in [6], The aim of the comparison is to emphasize the ability of our model to optimize risk equity, although guaranteeing a comparable level of total risk on the network.

References

[1] L. Bianco, M. Caramia and S. Giordani, "A Bilevel Flow Model for HazMat Transportation Network Design", *Transportation Research Part C* 17, 175–196 (2009).

[2] E. Erkut and O. Alp, "Designing a road network for hazardous materials shipments", *Computers and Operations Research* 34, 1389–1405 (2007).

[3] E. Erkut and F. Gzara, "Solving the HazMat Transport Network Design Problem", *Computers and Operations Research* 35, 2234–2247 (2008).

[4] E. Erkut, S.A. Tjandra and V. Verter, "Hazardous Materials Transportation", in *Operations Research Handbook on Transportation*, C. Barnhart and G. Laporte (eds.), 539–621, Elsevier, 2007.

[5] B.Y. Kara and V. Verter, "Designing a Road Network for Hazardous Materials Transportation", *Transportation Science* 38, 188–196 (2004).

[6] P. Marcotte, A, Mercier. G. Savard, and V. Verter. "Toll Policies for Mitigating Hazardous Materials Transport Risk", Transportation Science, *Transportation Science*, 43, 228–243 (2009).

[7] N. Nisan, T. Roughgarden, E. Tardos and V.V. Vazirani, *Algorithmic Game Theory*, Cambridge University Press, New York, NY, USA (2007).

[8] V. Verter and B.Y. Kara, "A Path-Based Approach for Hazmat Transport Network Design", *Management Science* 54, 29–40 (2008).

Integrated Urban Hierarchy and Transportation Network Planning

João F. Bigotte

Department of Civil Engineering

University of Coimbra

Email: jbigotte@dec.uc.pt

António P. Antunes

Department of Civil Engineering

University of Coimbra, Portugal

1 Introduction

The improvement of accessibility to services of general interest, such as education, health care, public safety, and justice, is a typical objective of regional development plans. Two types of actions may be used to tackle this objective. The first is to redefine the level of hierarchy assigned to the urban centers of the region under study. This can be done through the location of facilities where these services are offered to the population, with a class of facilities associated with each level of hierarchy. For instance, in a three-level hierarchy, facilities such as elementary schools or basic health care units could be associated with first-level centers, whereas high schools and local hospitals could be associated with second-level centers, and universities and central hospitals with third-level centers. The second is to redesign the transportation network of the region. In this case, the aim is to select which links of the transportation network should be improved (or which new links should be built). The transportation network is also a hierarchical system because different levels of links are characterized with different travel conditions (in terms of speed, safety, comfort, etc.).

With very few exceptions (e.g. [1] and [2]), the two subjects – urban hierarchy and transportation network planning – have been addressed separately in the optimization literature. In this article, we present an optimization model for integrated urban hierarchy and transportation network planning. The aim of the model is to answer the following question: which urban centers and which transportation links of a given region should be promoted to a new level of hierarchy so as to maximize the accessibility of population to the services of all levels available in the region?

2 Optimization model

The optimization model developed to represent integrated urban hierarchy and transportation network planning problems is as follows:

$$\min F = \sum_{k \in N} \sum_{(i,j) \in I} \sum_{l \in L} \sum_{m \in M} t_{ijm}^k x_{ijlm}^k \tag{1}$$

subject to

$$\sum_{i \in N} \sum_{m \in M} x_{iSlm}^k = 1, \forall k \in N, l \in L \tag{2}$$

$$\sum_{i \in N} \sum_{m \in M} x_{ijlm}^k = \sum_{m \in M} x_{jSlm}^k + \sum_{i \in N} \sum_{m \in M} x_{jilm}^k, \forall j \neq k \in N, l \in L \tag{3}$$

$$x_{ijlm}^k \leq r_{ijm}, \forall k \in N, \{i,j\} \in I_1, l \in L, m \in M \tag{4a}$$

$$x_{jilm}^k \leq r_{ijm}, \forall k \in N, \{i,j\} \in I_1, l \in L, m \in M \tag{4b}$$

$$x_{iSlm}^k \leq y_{il}, \forall k, i \in N, l \in L, m \in M \tag{5}$$

$$\sum_{i \in N} y_{il} \leq Y_l, \forall l \in L \tag{6}$$

$$y_{i(l-1)} \geq y_{il}, \forall i \in N, l \in \{L \mid (l-1) \geq 2\} \tag{7}$$

$$\sum_{m \in \{M \mid m > 1\}} r_{ijm} \leq 1, \forall \{i,j\} \in I_1 \tag{8}$$

$$\sum_{\{i,j\} \in I_1} \sum_{m \in M} w_{ijm} r_{ijm} \leq W \tag{9}$$

$$x_{ijlm} \geq 0, \forall (i,j) \in I, l \in L, m \in M \tag{10}$$

$$y_{il} \in \{0,1\}, \forall i \in N, l \in L; r_{ijm} \in \{0,1\}, \forall \{i,j\} \in I_1, m \in M \tag{11}$$

where

Sets: N - set of urban centers, $N = \{1, \ldots, NC\}$; NC - number of urban centers in the region; N_S - set of nodes (urban centers + a sink node, S), $N_S = \{1, \ldots, NC+1\}$; L - set of higher-level services, $L = \{2, \ldots, NL\}$; NL - number of levels of service (or urban hierarchy levels); M - set of link levels, $M = \{1, \ldots, NM\}$, NM - number of link levels; I_1 - set of links (i,j), $I_1 = \{1, \ldots, NI\}$, NI - number of undirected links connecting the centers; I_2 – set of links (j,i), $I_2 = \{NI+1, \ldots, 2 \times NI\}$; I_3 - set of links (i,S), $I_3 = \{2 \times NI+1, \ldots, 2 \times NI+NC\}$; I - set of all links, $I = I_1 + I_2 + I_3$.

Parameters: u_i - population residing in center I; Y_l - number of centers to be promoted to level-l; W - maximum length of transportation network improvements (expressed in length reference units); t_{ijm}^k - aggregate travel time of demand at center k in level-m link (i,j).

Decision variables: $y_{il} = 1$ if center i is designated as level-l center, $y_{il} = 0$ otherwise; $r_{ijm} = 1$ if the undirected link $\{i,j\}$ is designated as level-m link, $r_{ijm} = 0$ otherwise; x_{ijlm}^k - fraction of flow from center k using level-m link (i,j) en route to obtaining level-l service.

The objective-function (1) of this mixed-integer optimization model expresses the minimization of the total (or demand-weighted) travel time. This automatically implies that demand for

level-*l* service is satisfied at the closest level-*l* center. Constraints (2) guarantee that the flow from any center *k* reaches the sink node (which is equivalent to saying that the demand from *k* must be satisfied at some center *i*). Constraints (3) state that the total flow into node *j* must equal the total flow out of node *j*, either to the sink node (first term on the right-hand side) or to any center *i* (last term on the right-hand side). Please note that the demand from any center *k* travels to the closest center offering level-*l* service. Unless there are ties, the values of the variables x_{ijlm}^k will be either 0 or 1. Constraints (4a) force the fraction of flow from any demand center *k* in level-*m* link (*i*,*j*) to be zero if level-*m* link {*i*,*j*} is not part of the network. Constraints (14b) are identical but with respect to flow (*j*,*i*). Constraints (15) state that demand for level-*l* service from center *k* can only be satisfied at center *i* if this center is designated as level-*l* center. Constraints (6) ensure that a maximum of Y_l urban centers are designated as level-*l* centers. Constraints (7) enforce that if center *i* provides level-*l* service, then it provides all lower-level services as well. Constraints (8) state that the link between center *i* and center *j* can be improved to one level only. Constraint (9) guarantees that the maximum length for link improvements is not exceeded. Expressions (10) and (11) specify the domain of the decision variables.

3 Model application

The optimization model presented in the previous section was used to analyze possible decisions regarding the evolution of the urban hierarchy and the transportation network of the Centro Region of Portugal (Figure 1). The region has an area of approximately 28,000 km2 and a population of 2.3 million. The territory is organized in two levels – municipality (lower-level) and district (higher-level). There are 100 municipality main towns in the Centro Region, six of which are district capitals. The road network shown in the figure consists of the 2000 Road Network Plan (Decree-Law 222/98) and the additional projects that have been announced by the Portuguese government since the plan was adopted (MOPTC, 2008). This road network is expected to be completed by 2015. It is classified in three hierarchical levels according to design speed – 60, 90, and 120 kph.

In the analysis, we considered three scenarios, A, B, and C, and favored the less developed off-coast areas of the Centro Region (the Interior) from the more developed coastal areas (the "Litoral"). The scenarios were as follows: A - upgrade of six lower-level centers to a new, sub-regional level (that is, $Y_2 = 12$) and improvement of roads for an amount of 200 length-equivalent units ($W = 200$); B - $Y_2 = 12$, $W = 400$; C - $Y_2 = 18$, $W = 200$. For favoring the Interior, we considered $h_{Lit} = 0.33$ and $h_{Int} = 0.66$ in the objective-function:

$$\min F = \sum_{k \in N} \sum_{(i,j) \in I} \sum_{l \in L} \sum_{m \in M} \alpha^k h_{Lit} t_{ijm}^k x_{ijlm}^k + (1 - \alpha^k) h_{Int} t_{ijm}^k x_{ijlm}^k \tag{18}$$

where $\alpha^k = 1$ when center *k* is located in the Litoral and $\alpha^k = 0$ otherwise; h_{Lit} is the accessibility weight for the Litoral and h_{Int} is accessibility weight for the Interior..

The results obtained through the application of the model are illustrated in Figure 2 (for Scenario A) and Table 1 (for the three scenarios). They provide a clear picture of the implications for

the Centro Region of alternative decisions regarding the urban hierarchy and the transportation network.
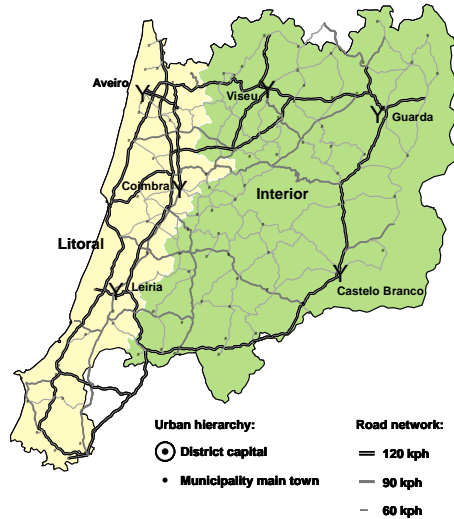
Figure 1: Current urban hierarchy and transportation network of the Centro Region
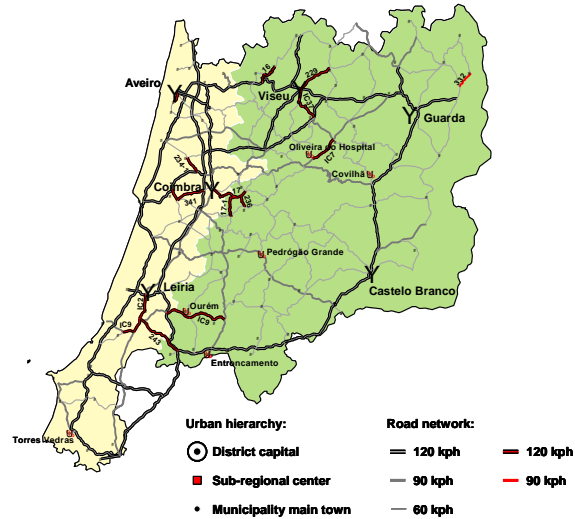
Figure 2: Optimum urban hierarchy and transportation network of the Centro Region for Scenario A

Table 1: Summary of results for the three scenarios

Scenario	Area	Number of centers		Road improvements (% of total)		Travel time (comparison with the base scenario)			
		Sub-regional (level-2)	District cap. (level-3)	90 kph	120 kph	to level-2 services	to level-3 services	to level-4 services	to all levels of service
A	Litoral	1	3	0%	27%	-	-	-	-
	Interior	5	3	4%	69%	-	-	-	-
	Region	6	6	4%	96%	-	-	-	-
B	Litoral	1	3	0%	27%	-4%	-4%	-	-4%
	Interior	5	3	3%	70%	-8%	-5%	-	-6%
	Region	6	6	3%	97%	-5%	-5%	-	-5%
C	Litoral	5	3	0%	16%	-41%	2%	-	-14%
	Interior	7	3	0%	84%	-18%	-1%	-	-7%
	Region	12	6	0%	100%	-32%	0%	-	-11%

References

[1] S. Melkote and M. S. Daskin, "An Integrated Model of Facility Location and Transportation Network Design", *Transportation Research Part A* 35(6), 515-538 (2001).

[2] S. Melkote and M. S. Daskin, "Capacitated Facility Location-Network Design Problems", *European Journal of Operational Research* 129(3), 481-495 (2001).

Cluster Based Fleet Assignment Problem

Mourad Boudia, Semi Gabteni

Amadeus s.a.s, Operational Research and Innovation Division

485 Route du Pin Montard. BP 69, 06902 Sophia Antipolis, France

Email: mourad.boudia@amadeus.com

1   Introduction

The Fleet Assignment Problem consists in assigning a particular fleet type and therefore a seat capacity to each flight leg in an airline schedule. The objective is to minimize the assignment costs and maximize the revenue subject to the available number of aircraft and different operational limitations. This is an upstream step of the airline planning process, that takes place a year prior to operations.

Different solution approaches have been proposed to solve this problem. The "Basic or Leg Based Fleet Assignment Model" [1] assumes a leg or flight based demand, independent to the demand for the other legs. However, demands are based on itineraries, or combinations of legs, rather than legs. This is the foundation of the alternative "Itinerary Based Fleet Assignment Models or IFAM" [2].

The basic models do not capture the network effects of the fleeting decisions. For a major US carrier, researchers [3] report that fleet assignments using network-based models with passenger itineraries enhance fleeting solutions significantly, with estimated savings ranging from 30 million to over 100 million dollars annually. Despite the benefits of the itinerary based models, these are subject to the excessive granularity of itinerary based demand and the difficulty of forecasting them one year before the day of operations. These weaknesses are even more significant for airlines with large networks and high connectivity.

This paper proposes to consolidate itinerary demands into clusters to take advantage of the fact that airlines are better able to make forecasts on itinerary aggregates [4], such as region to region traffic rather than origin to destination traffic. Through this demand consolidation, a more robust fleet assignment solution approach is achieved and the airline planning process is less affected by itinerary demand volatility. The proposed model and the clustering approach are presented, followed by comparison approach and computational results. Conclusion and future directions close this paper.

2 Cluster based fleet assignment model (CFAM)

The present proposal is based on the assumption of a pure Hub and Spoke network. All flights take-off or land at the hub, and the airline has one single hub.

Notations:

Let A be the set of airports, F the set of fleets, L the set of legs indexed by l or adt where $a, d \in A$ represent the origin and the destination and t the departure time, C the set of itinerary clusters, and $C(l)$ the set of itinerary clusters using leg l. For the time horizon, T is the set of all events at all airports, where t_0 is the chosen moment of the day to check the fleet size and $O(f)$ the set of flight arcs of the fleet f. N the set of nodes of the network enumerated by $\{fat\}$ with $f \in F, a \in A$ and t a time of takeoff or landing at a.

Parameters

D_c: the unconstrained demand of cluster c

$c_{f,l}$: assignment cost of fleet f to leg l

$\widetilde{fare}_c$: the average fare for demand cluster c

$SEATS(f)$: seat capacity on aircraft of fleet type f

$Size(f)$: the size of fleet type f

Variables

$X_{f,l}$: binary variable equal to one if fleet f is assigned to leg l, 0 otherwise

Y_{f,a,t^+}, Y_{f,a,t^-}: the number of aircraft of fleet type $f \in F$ that remain on the ground at airport $a \in A$ within the time interval $[t, t^+]$ or $[t^-, t]$

s_c: the spill or the part of the not stisfied demand of the cluster c

$$Min \sum_{l \in L, f \in F} c_{f,l} \, X_{f,l} + \sum_{c \in C} \widetilde{fare}_c \, s_c \tag{1}$$

$$\sum_{f \in F} X_{f,l} = 1 \qquad \forall l \in L \tag{2}$$

$$\sum_{d \in A} X_{f,dat} + Y_{f,a,t^-} = \sum_{d \in A} X_{f,adt} + Y_{f,a,t^+} \qquad \forall fat \in N, \forall f \in F, \forall a \in A \tag{3}$$

$$\sum_{l \in O(f)} X_{f,l} + \sum_{a \in A} Y_{f,a,t_0} \leq Size(f) \qquad \forall f \in F \tag{4}$$

$$\sum_{c \in C(l)} D_c - \sum_{c \in C(l)} s_c \leq \sum_{f \in F} X_{f,l}.SEATS(f) \qquad \forall l \in M \tag{5}$$

$$s_c \leq D_c \qquad \forall c \in C \tag{6}$$

The objective function (1) minimizes the total operations cost plus the cost related to cluster spilled passengers. The latter corresponds to the revenue maximization. Constraints (2) are the classical constraints of legs coverage. The flow conservation at each airport of the network is ensured

thanks to constraints (3). The limited size of each fleet f must be respected, constraints (4).

The key difference with the Itinerary Based FAM is that the leg capacity constraints (5) are based on itinerary cluster demands rather than itineraries. As such, the approach makes seat capacity allocation and spill decisions based on itinerary clusters, for which the airline is in a position to make reliable demand forecasts. We should underline here that each local leg demand (one-leg itinerary) is considered as a specific cluster demand. For each leg l, the sum of the unconstrained cluster demands using that leg minus the sum of the spill for each cluster using leg l cannot exceed the capacity of this leg. Constraints (6) state that for each cluster, the spill for that cluster cannot exceed its demand.

3 Cluster building

The model should be fed by unconstrained cluster demands, which do not exist. These have to be built based on the available itinerary demand information, and a clustering method. Such a clustering method relies on a metric that measures itineraries against one another, and drives the decision to have two itineraries in the same cluster or not.

Two metrics were evaluated, based on two distinct clustering approaches, that were both driven by the need to achieve reduced demand volatility

First an intuitive geographic metric is proposed, based on the distance between airports. With such a metric, itineraries originating within a defined geographical region are grouped together. A distance matrix is computed providing the distance between each airport pair. A distance threshold determines airports which are grouped together. Such that the itineraries originating at those airports are then found in the same itinerary cluster. The appropriate threshold is found with extensive simulations.

The second metric reflects, for two origin stations, their similarity in terms of destinations served by their respective itineraries. The resulting clusters can contain airports that are not necessarily close considering the geographical distance, but from where passengers travel to the same destinations. For a couple of airports their distance is calculated based on their respective list of destinations that are compared. The more similar the destinations, the closer to zero is the distance, conversely, if the lists are completely different the distance is 1. Here again, several simulations were necessary to find a good threshold.

4 Comparison approach

Once the Itinerary Clustering settled, the estimated cluster demands built by the demand forecaster are fed to the CFAM model which provides a CFAM based fleeted schedule. In parallel, the original estimated itineraries demands are fed to the traditional IFAM model [2], which provides an IFAM

based fleeted schedule. Several sets of demand realizations scenarii are generated based on varying itinerary demand volatility assumptions. For each set, and for each scenario within the set, the CFAM based fleeted schedule and the IFAM based fleeted schedule are compared, by computing the assignment cost and the revenue generated by each fleeted schedule.

5   Computational results

The fleet assignment models were implemented in C++ and solved using COIN-OR solver. A mid size European carrier's network has been used for this study.

Overall, the simulations show that the cluster based approach is more robust and gives better results compared to the itinerary based approach. However, in some cases with low demand volatility, the itinerary based approach is unsurprisingly better. The metrics used were also compared. The results of the destination metric are better than the geographical metric. This is due to the fact that the destination metric better takes into account the passenger flows.

6   Conclusion and future directions

In this paper, a new model for the fleet assignement problem is proposed. The introduction of the itinerary clusters improves the robustness of the fleet assignment model compared to itinerary based fleet assignment. The future directions involve the evaluation of alternative metrics, the extension to non pure hub and spoke networks, and the feed of the model by pure cluster based forecasts.

References

[1] J. Abara, "Applying Integer Linear Programming to the Fleet Assignment Problem", *Interfaces* 19, 20-28 (1989).

[2] C. Barnhart, T.S. Kniker and M. Lohatepanont, "Itinerary-Based Airline Fleet Assignment", *Transportation Science* 36, 199-217 (2002).

[3] C. Banhart and Amy Cohn, "Airline schedule planning: Accomplishments and opportunities", *Manufacturing & Service Operations Management* Vol. 6, No. 1, 3-22 (2004).

[4] J.C. Culioli, "Modèles et problèmes de revenue management pour une compagnie aérienne", 5ème Rencontre Math-Industrie: Aéronautique et Espace, Toulouse, 2006.

Network Contraction for Rapid Equilibrium Assessment

Stephen D. Boyles

Department of Civil and Architectural Engineering

University of Wyoming, Laramie, WY, USA

Email: sboyles@uwyo.edu

1 Introduction

Static user equilibrium traffic assignment lies at the core of a variety of transportation planning and operations models. When a large number of alternatives are considered, a frequent practice is to use a *sketch network* containing a specific region of interest, while assuming unchanged conditions elsewhere. This allows a very large number of alternatives to be compared in short order, but taking the *ceteris paribus* assumption loses important global diversion/attraction effects which occur due to local changes, and makes it difficult to evaluate the large-scale impacts of the alternatives.

Network contraction provides an alternate approach for rapid equilibrium sensitivity analysis. The idea is to replace a large transportation network with a smaller one which responds similarly to changes in input flow. A simple example is compressing two links which are in series into a single link by simply adding their cost functions. It is often useful to keep a portion of the network uncontracted (to "zoom in" on an area of particular interest) while contracting the remainder of the network to reduce computation time.

These transformations, derived formally in the next section, closely resemble techniques used in the analysis of electric circuits and, indeed, the formulas bear much similarity although derived from first principles here. Two important distinctions between these two domains are that (1) the difference in electric potential between two nodes must be the same among *all* paths, whereas the travel cost between two nodes must be equal only among *used* paths; and (2) in transportation networks, travelers departing different origins are distinct and may see different "potentials." This distinction vanishes if we consider a single origin at a time, and restrict attention to the set of minimum-cost paths rooted at that origin. More precisely, given an origin u, we only consider the subgraph $B^u = (N, A^u)$ where $A^u = \{(i,j) \in A : t_{ij} - (\pi_j - \pi_i) = 0$ and $\boldsymbol{\pi}$ is a set of shortest-path

node potentials from u. Following Dial (2006), we term this an *equilibrium bush*.

Within an origin's equilibrium bush, all paths between two nodes have equal travel cost, and we seek to analyze how the equilibrium travel cost will change between each origin and destination either as a function of the input flows, or as a function of changes in link parameters in uncontracted regions of the graph. We assume that (1) the equilibrium bush remains unchanged throughout the range of input demands, and (2) the network is planar.

2 Network Transformations

The following notation is used: let $t_{ij}(x_{ij})$ represent the travel time on link (i,j) when the demand for travel on this link is x_{ij}, and let $T_{ij}(X_{ij})$ represent the equilibrium travel cost between any two nodes among *all* paths connecting these nodes, as a
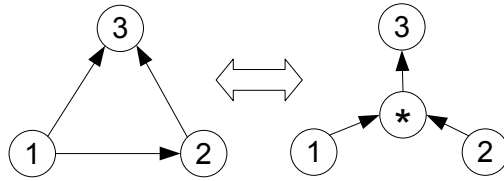
Figure 1: Schematic of the delta-Y and Y-delta transforms.

function of the total demand for travel between these nodes (regardless of destination). In particular, we are interested in the derivatives $T'_{ij}(X_{ij})$; if we have a "base" equilibrium solution, we know X_{ij} and T_{ij}, and can make the approximation $T_{ij}(X) \approx T_{ij}(X_{ij}) + T'_{ij}(X_{ij})(X - X_{ij})$. Higher-order approximations naturally follow if further derivatives are known. Our goal, therefore, is to derive T'_{ij} for a variety of network configurations. This necessarily involves calculating the derivatives in equilibrium link flows $\alpha_{k\ell} \equiv dx_{k\ell}/dX_{ij}$ as demand between i and j varies.

Series Consider two links (i,j) and (j,k) in series, with cost functions $t_{ij}(x_{ij})$ and $t_{jk}(x_{jk})$, and let x be the demand for travel between i and k. Then $x_{ij} = x_{jk} = x$, and the equilibrium travel time T_{ik} between i and k is clearly $(t_{ij} + t_{jk})(X_{ik})$, so $T'_{ik} = (t'_{ij} + t'_{jk})(X_{ik})$. From this point on, dependence of link travel times on flows is suppressed for brevity.

Parallel Consider two links $(i,j)^1$ and $(i,j)^2$ in parallel, with cost functions t^1_{ij} and t^2_{ij} Because the bush is in equilibrium we have $t^1_{ij} = t^2_{ij}$. Furthermore, as X_{ij} changes, x^1_{ij} and x^2_{ij} will change such that the equilibrium is preserved, that is,

$$\frac{dT_{ij}}{dX_{ij}} = \frac{dt^1_{ij}}{dX_{ij}} = \frac{dt^2_{ij}}{dX_{ij}} \iff \frac{dt^1_{ij}}{dx^1_{ij}}\alpha^1_{ij} = \frac{dt^2_{ij}}{dx^1_{ij}}\alpha^2_{ij} \qquad (1)$$

Furthermore, by conservation of flow we have $\alpha^1_{ij} + \alpha^2_{ij} = 1$, so by substitution $\alpha^1_{ij} = (t^2_{ij})'/[(t^1_{ij})' + (t^2_{ij})']$ and

$$\frac{dT_{ij}}{dx} = \frac{(t^1_{ij})'(t^2_{ij})'}{(t^1_{ij})' + (t^2_{ij})'} \qquad (2)$$

Delta-Y Consider an undirected cycle of three nodes with an empty interior (Figure 1). Without loss of generality let these be nodes 1, 2, and 3, and let their node potentials satisfy $\pi_1 \leq \pi_2 \leq \pi_3$. The arc orientations must then be as indicated as in the figure. For brevity, let Δ_1, Δ_2, and Δ_3 represent the marginal inflow to the cycle; that is, $\alpha_{12} + \alpha_{13} = \Delta_1$, $-\alpha_{12} + \alpha_{23} = \Delta_2$, and $-\alpha_{13} - \alpha_{32} = \Delta_3$. These equations are linearly dependent, as flow conservation demands. However, the requirement that the bush remain at equilibrium also requires $\alpha_{13} t'_{13} = \alpha_{12} t'_{12} + \alpha_{23} t'_{23}$. This provides a third, linearly independent equation to solve for α_{12}, α_{13}, and α_{23}. Omitting the details, we have

$$\alpha_{12} = \frac{t'_{13}\Delta_1 - t'_{23}\Delta_2}{t'_{12} + t'_{13} + t'_{23}} \qquad \alpha_{13} = \frac{t'_{12}\Delta_1 - t'_{23}\Delta_3}{t'_{12} + t'_{13} + t'_{23}} \qquad \alpha_{23} = \frac{t'_{12}\Delta_2 - t'_{13}\Delta_3}{t'_{12} + t'_{13} + t'_{23}}$$

with a pleasing symmetry. Thus, the differential change in equilibrium costs are given by

$$dT_{12} = t'_{12}\frac{t'_{13}\Delta_1 - t'_{23}\Delta_2}{t'_{12} + t'_{13} + t'_{23}} \qquad dT_{13} = t'_{13}\frac{t'_{12}\Delta_1 - t'_{23}\Delta_3}{t'_{12} + t'_{13} + t'_{23}} \qquad dT_{23} = t'_{23}\frac{t'_{12}\Delta_2 - t'_{13}\Delta_3}{t'_{12} + t'_{13} + t'_{23}}$$

Now, consider the Y junction in Figure 1. The geometry of this junction forces $\alpha_{1*} = \Delta_1$, $\alpha_{2*} = \Delta_2$, and $\alpha_{*3} = -\Delta_3$. If we choose delay functions t_{1*}, t_{2*}, and t_{*3} such that

$$t'_{1*} = \frac{t'_{12}t'_{13}}{t'_{12} + t'_{13} + t'_{23}} \qquad t'_{2*} = \frac{t'_{12}t'_{23}}{t'_{12} + t'_{13} + t'_{23}} \qquad t'_{*3} = \frac{t'_{13}t'_{23}}{t'_{12} + t'_{13} + t'_{23}}$$

it is easily seen that the change in equilibrium costs is identical to that in the delta.

Y-Delta The delta-Y equations can be inverted to reverse the previous transformation, allowing one to replace a three-pronged intersection with an equivalent triangular component. The reader can verify that the following equations indeed accomplish this inversion.

$$t'_{12} = \frac{t'_{1*}t'_{2*} + t'_{1*}t'_{*3} + t'_{2*}t'_{*3}}{t'_{*3}} \qquad t'_{13} = \frac{t'_{1*}t'_{2*} + t'_{1*}t'_{*3} + t'_{2*}t'_{*3}}{t'_{*2}} \qquad t'_{23} = \frac{t'_{1*}t'_{2*} + t'_{1*}t'_{*3} + t'_{2*}t'_{*3}}{t'_{*1}}$$

3 Demonstration

This section demonstrates the above procedure on the well-known Braess network as portrayed in Sheffi (1985). Figure 2 shows iteratively how the contraction is performed: panel (a) shows the equilibrium solution on the initial network with a travel demand of 6 between nodes 1 and 4, along with link travel times and travel time derivatives. Panel (b) shows the link travel time derivatives alone; panel (c) shows the derivatives after applying a delta-Y transform to nodes 2, 3, and 4; panel (d) shows the result of series transformations to nodes 1, 2, and * and 1, 3, and *; panel (e) shows the result of a parallel transformation among nodes 1 and *; and panel (f) shows the result after a final series contraction (note that Feo and Provan (1993) show that reduction to a single link is always possible, as long as there is just one origin and one destination), demonstrating that the derivative of the equilibrium travel time between nodes 1 and 4, with respect to travel demand, is 31/13. To verify this, Figure 3 plots the equilibrium travel time for a variety of travel demand
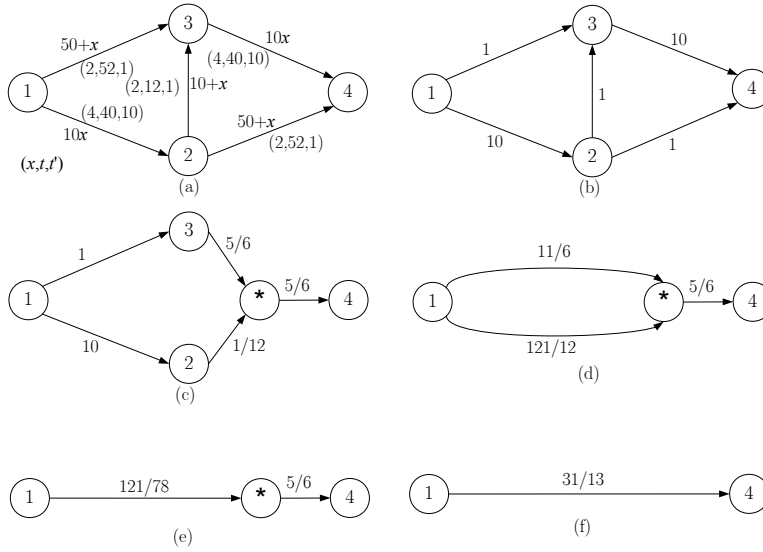
Figure 2: Application to the Braess network.

values; the resulting graph is piecewise linear due to the cost functions used in this network. When travel demand lies in the interval between 40/11 and 100/9, the approximation is exact. More comprehensive demonstrations, including multiple origins and destinations, is included in the full paper.

References

Dial, R. B. (2006). A path-based user-equilibrium traffic assignment algorithm that obviates path storage and enumeration. *Transportation Research Part B 40*, 917–936.

Feo, T. A. and J. S. Provan (1993). Delta-wye transformations and the efficient reduction of two-terminal planar graphs. *Operations Research 41*(3), 572–582.

Sheffi, Y. (1985). *Urban Transportation Networks*. Englewood Cliffs, NJ: Prentice-Hall.
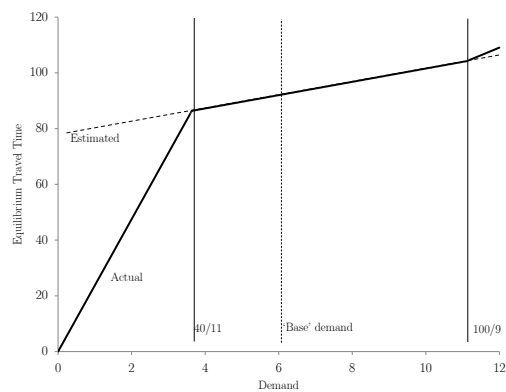
Figure 3: Application to the Braess network.

Approximate Hill Climbing Approach for the Fleet Size and Mix Vehicle Routing Problem with Time Windows

Olli Bräysy

Agora Innoroad Laboratory

University of Jyväskylä, Jyväskylä, Finland

Email: olli.braysy@jyu.fi

1 Introduction

The Vehicle Routing Problem (VRP) is at the heart of distribution problems as it addresses how the demand of customers can be satisfied at minimal cost by homogeneous vehicles located at the intermediate storage facility. This paper addresses two of the most common extensions of the VRP occurring in practice: the presence of service time windows at customers and the use of heterogeneous vehicles. The objective of the so-called Fleet Size and Mix Vehicle Routing Problem with Time Windows (FSMVRPTW) is therefore to find a fleet composition and a corresponding routing plan that minimizes the sum of routing and vehicle costs. For recent literature, see [1], [2], [3], [4]. Here we describe a new approximate search strategy that is combined with recent threshold accepting metaheuristic by [4].

2 Solution Method

The applied heuristic solution method consists of three phases and a pre-processing step. The pre-processing step is used to define a limiting value for each customer point, specifying a radius (distance) in which the c closest customers are located.

In the first phase, a single initial solution is generated with a modification of the savings heuristic [5]. To save computation time, the search is restricted at two levels. First, the mergers are limited to the p closest routes only. The geographical proximity of the routes is based on the Euclidean distance of the average X and Y coordinates of the customers in the routes. Second, only the c customers from route R_2 that are closest to the endpoints of route R_1 are considered for insertion of merging points.

In the second phase, the route elimination procedure (ELIM) is used to reduce the number of routes in the initial solution. ELIM considers all routes of the incumbent solution for depletion in random order, until no more improvements can be found with respect to the total cost objective. For a given route, ELIM removes all customers, and tries to insert them one by one in the remaining routes. Removed customers are reinserted in the p geographically closest neighbouring routes. For a given

customer v and geographically close route R_2, only insertion positions adjacent to one of the c closest customers with regard to customer v are considered.

In the third phase the search is based on the Threshold Accepting (TA) metaheuristic [6] . The basic idea of TA is to allow also local search moves that worsen the objective value, as long as the worsening is within the current value of the threshold limit. The TA procedure starts with threshold T = 0 (no worsening allowed) and is repeated with that value until a local minimum is reached. If no more improvements have been found for a given number of iterations, the threshold is set to a maximum value and the search is restarted from the current best solution. At each non-improving iteration, the threshold is reduced by ΔT units until zero is reached again.

In each restart, the size of the maximum threshold is controlled depending on the search phase. More precisely, in the beginning of the search we allow for potentially large worsenings of the objective functions, but this maximum threshold is reduced as the search progresses. After resetting T ten times, moves resulting in large worsenings of the objective function are again attempted. In addition to the threshold value that controls the acceptance of individual moves, we also control the total relative worsening of the solution quality with respect to the current best solution to avoid allowing for too many worsening moves.

To avoid cycling and to speed up the search, we included a simple Tabu Search (TS) method that is juxtapositioned with the TA. Within the TS scheme, we record as tabu the arcs connecting the first nodes of the route segments of improving local search moves. The tabu status of these arcs forces keeping them for a given number of iterations, even if changing them would enable further improvements (i.e. no aspiration criteria are used).

The applied local search operators include a route splitting operator called SPLIT, new variants of ICROSS and IOPT operators and ELIM method described above. The SPLIT neighbourhood consists of all solutions that result from splitting a single route in the current solution into two parts at any point. We employ the method here in a greedy first-accept fashion. ICROSS relocates or exchanges segments of consecutive customers between two separate routes and IOPT intra-tour operator is a generalization of Or-opt.

To speed up ICROSS and IOPT, only geographically close routes and only segments that involve the geographically closest customer pairs in the two routes are considered. Instead of considering a fixed number of close routes, p, for a given route R_1, an analysis step is made in the beginning of Phase 3 and after each successful SPLIT move. In the analysis step, ICROSS is applied to the p closest routes of each route, starting from the closest and we record to each route the information that how many of its closest routes should be considered to find improvements.

Instead of the traditional first- or best-accept rules based on the objective function only, we keep track of the arc frequencies in the obtained solutions during the search. Every time when evaluating a new move, we calculate both the total cost value and the sum of the frequencies of the new arcs created by the move. This is repeated for all feasible moves of a given neighbourhood. Then, the feasible move that improves the total cost value and has the lowest total arc frequency is selected.

3 Approximate Search Strategy

As in most local search based hill-climbing metaheuristics, in the above described method the most time-consuming part is the evaluation of the local search moves. To obtain good solutions it often requires hundreds or even thousands of iterations. In most of these iterations, worsening of the objective function value is allowed. The goal of these worsening moves is to diversify the search and escape local minima.

The suggested approximate search strategy is based on the claim that there is no need to evaluate exactly the objective function value of the hill-climbing moves. It even does not matter if some improving move is evaluated occasionally as a worsening move or vice versa. It is enough to calculate the approximate impact of each move on the objective function value. However, in the improvement stage, when $T=0$, one must evaluate the moves exactly.

The basic idea of the suggested approximate search strategy is that when $T=0$, the moves are evaluated exactly. Just before setting T to its new maximum value, all local search moves and their feasibility are evaluated and the information is stored to memory using a priority queue data structure. The objective function values are picked from the queue during the entire hill-climbing phase. To avoid moving points that are already relocated, each point is allowed to be moved to another route only once during the hill-climbing phase. The feasibility checks are calculated exactly also during the hill-climbing phase in case the associated routes have changed.

4 Computational Experiments

The computational experiments are carried out with the benchmark problems suggested by [2]. In contrast to Liu and Shen [1], who minimized the sum of all vehicle costs and en route time, we consider the sum of all vehicle costs and total distance as optimization objective. The benchmark set consists of 768 problems with varying fleet costs, time windows, spatial structure etc. The key issue is to demonstrate the impact of the suggested approximation strategy on speed and solution quality. The preliminary results indicate that the suggested approach outperforms previous methods with respect to both time and solution quality. This indicates that the suggested approximation strategy may well be combined also with other metaheuristics and applied to other problems.

References

[1] F.H. Liu and S.Y. Shen, "The Fleet Size and Mix Vehicle Routing Problem with Time Windows", *Journal of the Operational Research Society* 50, 721–732 (1999).

[2] O. Bräysy, W. Dullaert, G. Hasle, D. Mester and M. Gendreau, "An Effective Multi-Restart Deterministic Annealing Metaheuristic for the Fleet Size and Mix Vehicle Routing Problem with Time Windows", *Transportation Science* 42, 371–386 (2008).

[3] O. Bräysy, P. Porkka, W. Dullaert, P. Repoussis and C. Tarantilis, "A Well-scalable Metaheuristic for the Fleet Size and Mix Vehicle Routing Problem with Time Windows", *Expert Systems with Applications* 36, 8460–8475 (2009).

[4] O. Bräysy, P. Hotokka, W. Dullaert and Y. Nagata, An Adaptive Diversification Heuristic for the Fleet Size and Mix Vehicle Routing Problem with Time Windows, Working paper, University of Jyväskylä, Finland (2009).

[5] G. Clarke and J. Wright, "Scheduling of Vehicles From a Central Depot to a Number of Delivery Points", *Operations Research* 12, 568–581 (1964).

[6] G. Dueck and T. Scheurer, "Threshold Accepting: A General Purpose Optimization Algorithm", *Journal of Computational Physics* 90, 161–175 (1990).

Exact and Metaheuristic Approaches for Bi-Objective Winner Determination in Transportation Procurement Auctions

Tobias Buer

Department of Information Systems

University of Hagen, Profilstr. 8, 58084 Hagen, Germany

Email: tobias.buer@fernuni-hagen.de

Giselher Pankratz

Department of Information Systems

University of Hagen

Shippers, such as industrial or trading companies, regularly use framework agreements in order to contract out their transportation tasks to motor carriers. In a framework agreement the shipper arranges with the freight carrier which transportation services he is to take over on what level of service and at what cost they are to be carried out. In this case, a framework agreement (denoted as contract in the remainder) comprises the transportation (typically repeated) of a volume of goods from a pickup point to a delivery point.

The placing of transport contracts by a shipper typically is carried out within a tender. In practice, such tenders are used to contract out up to 5,000 transport contracts valued up to 700 millon US-$ [3, p. 543]. The process for tendering transportation contracts is carried out, as a rule, in three steps [3, p. 542]: In the qualification stage (pre-auction stage) the shipper chooses the carriers that can provide transportation services at a minimum specified quality level. Regularly applied quality criteria are, for example, solid financial ratios, suitable IT systems for a smooth exchange of data, a vehicle fleet that is suitable for transporting the goods, or the reliable adherence to delivery deadlines ([3], [2]). Freight carriers that were able to pass the qualification stage are allowed to take part in the bidding stage (auction stage). In this stage the qualified carriers submit bids for the tendered transportation contracts. After that, in the allocation stage (post auction

stage) the shipper assigns the contracts to carriers based on the bids received and in accordance with previously established criteria.

In our contribution, we focus on the allocation stage. The outlined scenario inhibits two special features. Firstly, from the point of view of the carriers participating in the auction, there are valuation interdependencies between transport contracts. The costs of a contract for a bidder strictly depend on which other contracts he is awarded in the allocation stage. Complementary and substitutional interdependencies can be distinguished. If two contracts are complementary, e.g. because they can be combined to make a busy route, then the costs of the combined execution of both contracts are less than the sum of the costs that result for each of the contracts when executed separately (cost subadditivity). As a second feature, the shipper faces a multi-criteria decision making situation when determining the winning bids. While some of these criteria can be formulated in a tolerable approximation of practical custom as side constraints, other criteria are to be considered explicitly as minimization or maximization objectives in order to adequately reflect practical requirements. Along with the common objective of minimizing total costs, this contains the goal of achieving the highest possible overall performance quality in carrying out the contracts. Beyond assuring a minimium level of service within the qualification stage, an improvement of transportation quality (perhaps at the disadvantage of higher total procurement costs) can be considered in the allocation stage. The joint consideration of both features in a simultaneous approach is up to now missing in the literature. A review of existing approaches for modelling and solving winner determination problems is given by [1]. Among others, [4] as well as [3] especially deal with winner determination in the transportation domain.

This contribution addresses this gap. In doing so, it proposes a winner determination model which simultaneously considers interdependent valuations of contracts and multiple decision criteria. This problem, called "bi-objective winner determination problem of a combinatorial procurement auction based on a set covering formulation" (2WDP-SC), is based on the well-known NP-hard set covering problem. To express cost-interdependent valuations between contracts, carriers are allowed to submit so-called bundle-bids. A bundle-bid comprises one or more transport contracts. A bundle-bid is indivisible, i.e., a bidder is awarded all of the contracts in a bundle-bid or none of them. From the set of all submitted bundle-bids, the shipper has to select a subset of bundle-bids, in a way that every contract is part of at least one selected bundle-bid. The total procurement costs are to be minimized (first objective function) and the total service quality level of all contracts is to be maximized (second objective function). With respect to the latter objective it is assumed, that for each combination of a contract and a carrier a service quality value is given. This value denotes, how well the respective carrier can execute the according contract.

To solve the 2WDP-SC, a GRASP-based Pareto-solver is introduced. The proposed algorithm uses the Pareto dominance principle to determine a set of non dominated solutions. For this reason, a transformation of the two objective 2WDP-SC into a single objective optimization problem, e.g., through weighting the objective functions, intendedly needs not be considered. Thus, the algorithm does not require any individual preference information of the shipper. This appears especially advantageous in the light of the transportation scenario at hand, since eliciting the preferences of the shipper is one of the most challenging and most time-consuming activities in the tendering process [4, p. 249]. To conclude the allocation phase, the shipper finally needs to select a best solution according to his subjective preferences – however this issue is not treated in the current contribution. Nevertheless, solution approaches to this problem are already known in the literature (e.g. [5]).

To assess the performance of the proposed GRASP-based Pareto solver numerical benchmark tests are performed with the aid of specifically generated test instances. An extension of test instances known in the literature, e.g., for the set covering problem, seemed inappropriate, as these instances do not reflect some major economic features of the scenario. The GRASP-based Pareto solver is compared to a genetic algorithm and an exact branch-and-bound approach.

References

[1] J. Abrache, T. Crainic, M. Gendreau, and M. Rekik. Combinatorial auctions. *Annals of Operations Research*, 153(34):131–164, 2007.

[2] C. Caplice and Y. Sheffi. Optimization-based procurement for transportation services. *Journal of Business Logistics*, 24(2):109–128, 2003.

[3] C. Caplice and Y. Sheffi. Combinatorial auctions for truckload transportation. In P. Cramton, Y. Shoaham, and R. Steinberg, editors, *Combinatorial Auctions*, pages 539–571. MIT Press, Cambridge, MA, 2006.

[4] Y. Sheffi. Combinatorial auctions in the procurement of transportation services. *Interfaces*, 34(4):245–252, 2004.

[5] P. Vincke. *Multicriteria Decision-Aid*. John Wiley & Sons, Chichester, 1992.

Managing Debris Collection and Disposal Operations

José Antonio Carbajal, Kael Stilp, Monica Villarreal

Özlem Ergun and Pinar Keskinocak

Center for Health and Humanitarian Logistics

The Supply Chain and Logistics Institute

School of Industrial and Systems Engineering

Georgia Institute of Technology

Atlanta, GA USA

Email: oergun@isye.gatech.edu

March 1, 2010

1   Introduction

Debris is the waste generated after the strike of a natural or man-made disaster, such as hurricanes, earthquakes, terrorist attacks, etc. The amount of debris generated by some large-scale disasters is equivalent in volume to years of normal solid waste production in the affected areas [7]. Hurricane Katrina (2005) generated the greatest amount of debris than any other disaster in the United States, more than 100 million cubic yards (CY) (previously, Hurricane Andrew (1992) had generated 43 million CY) [2]. More recently, Hurricane Ike (2008) generated 19 million CY, enough to fill a football field stacked 2 miles high [5].

Debris removal is *costly* and it is often a *long and complicated process* requiring the careful consideration of both *short term and long term effects* on people's health and safety, and the environment. In the short term, the main consideration is the clearance of debris to allow for the transportation of relief resources (health care workers, personnel from relief agencies, relief items, etc.) and access to disaster areas or critical facilities for lifesaving activities [2]. Given that the debris may contain toxic or hazardous waste, one needs to weigh the benefits of rapid clearing with the long term impact "to ensure that their management (e.g., landfilling) would not pose a future threat to human health or the environment" [2].

With the goal of enabling the rapid deployment of relief efforts and at the same time diminishing

the long term impacts of post-disaster generated debris, a *debris management plan* that considers debris *clearance*, *collection*, and *disposal* operations should be developed. In the aftermath of Hurricane Katrina, the Federal Emergency Management Agency (FEMA) recognized the need for developing new strategies and plans for debris removal and management [4] and made available the main guidelines for such a plan through its Public Assistance Debris Management Guide [3]. However, FEMA Debris Management Guide focuses on 'what' to do, rather than on 'how' to do it. Hence, models and decision support tools are needed to assist agencies and communities in planning for and executing debris clearance missions.

Most of the literature debris management focuses on high level process-related aspects. [7] provides general guidelines, similar to those from FEMA, including allocation of responsibilities, public policy definitions, administrative procedures, etc. [6] discusses the development of strategic management for earthquake debris in the city of Tehran, carries out a SWOT (Strengths, Weaknesses, Opportunities, and Threats) analysis to assess the actual and potential debris management capacity, and prioritizes strategies for debris forecast estimation, proper design of construction, recycling and reuse of debris, etc. [1] provides a survey of disaster related research and identifies debris cleanup as a major logistics recovery problem and an important future research direction.

Debris related operations is a complex problem with multiple interrelated sub-problems. FEMA classifies debris operations as follows: debris forecasting, sourcing strategy, debris management sites (DMS) planning and operation, debris collection during disaster response and recovery phases, and debris reduce/recycle and final disposal. FEMA develops guidelines for how to handle a disaster situation, but it is the responsibility of the local governments to carry out those guidelines into strategic plans and operations. Agencies such as the United States Army Corps of Engineers (USACE) have developed tools for debris forecasting for natural disasters, implemented some common sourcing contracts, and designed DMS layouts [3]. Local authorities often contract with private firms to ensure the availability of adequate resources for the timely clearance and removal of debris in their areas. They also maintain monitors in the field as eyes and ears for the operation and make tactical and operational decisions, for example, by assigning regions to different contractors, choosing the DMS sites, and providing road priorities for clearance.

Our research in this area has three main components: (i) debris clearance, (ii) debris collection, and (iii) debris disposal, which correspond to the response, recovery, and the post-recovery phases of the disaster timeline. We also study the interactions between these three components. Next, we provide a summary of our current models and results.

2 Debris Clearance - Response

The response operations during a disaster comprise all the activities which take place during, or immediately after, a catastrophe occurs and which are aimed to provide relief or prevent additional damage. Such operations include the mobilization of search and rescue teams, ambulances, fire trucks, etc. for rescuing trapped people, transporting injured individuals to receive medical attention and so on; the transportation of relief supplies (food, water, clothes, etc.) to people in need; and the clearance of debris to reduce its impact on transportation and access to critical facilities (by clearing routes) in disaster areas.

One of the main decisions in debris clearance is to find the best sequence of roads/areas to clear in order to facilitate the flows of relief supplies and search and rescue crews throughout the network and to enable access to critical facilities such as hospitals and fire stations. We showed that even the simplest special cases of the problem, on planar, single stage, uncapacitated networks, are NP-Hard (by reductions from the knapsack problem). Hence, identifying strong lower and upper bounds on the objective function value is a challenging but important task. Generalizations (e.g., with the addition of fairness constraints) or multistage extensions of related combinatorial optimization problems (such as prize collecting steiner tree) are not well studied.

We model this problem as a multi-period network capacity expansion problem, which takes as input an estimation of the current capacity and the condition of the roads (blocked or clear), as well as the available supply and required demand of relief items. Note that, demand and supply are not as well defined as in a regular network problem, where for instance demand is related to the expected sales of a product, and supply capacity is related to a production capacity. In our preliminary model we use the number of vehicles as a common unit for measuring and matching demand, supply, and arc (i.e., road and street) capacities. We assign a penalty for each unit of unmet relief demand during each period of time. This penalty could vary with the type of relief, location of the demand, and the time when the demand is unmet. This has the purpose of reflecting the relative importance of demand, as in the case of route prioritization during the current debris collection planning process. The objective of the model is to minimize a penalty function for unmet relief demand. For every period, the model decides which of the remaining blocked road segments to clear from debris.

New demand and supply could arrive dynamically at any period and we consider two generic types of demand based on their accumulation pattern. The first type is the demand that accumulates through time, i.e., if it is not met completely during the period where it originates, it carries over to the next period; the second type is the demand that does not accumulate. Examples of the first case are medicines or medical attention, while the second case include food and water.

After developing a mixed integer program (MIP) for the above problem, we have explored the results obtained from the linear programming (LP) relaxation of our model using some randomly

generated instances as well as an instance based on real world data. Furthermore, we developed heuristics and compared their performances. Finally we tested the effect of policy alternatives on the efficiency of the operations.

3   Debris Collection - Recovery

Debris related recovery operations begin "after the emergency access routes are cleared and the residents return to their homes and begin to bring debris to the public rights-of-way" [3]. Recovery operations often utilize a combination of own force and equipment, and contractor services. Generally, price unit contracts are used, which establish a fixed cost for each unit (ton, truckload, cubic yard, etc.) of a particular debris collected, and they are executed through a load ticket system, i.e., keeping detailed track of each loading task executed by a registered truck. Hazardous waste and white goods are the two most common types of debris that need special handling, requiring specialized trucks and contractors.

In contrast with the clearance stage, where the activities are driven mostly by the urgency of enabling the emergency operations, during the collection stage the focus is on collecting debris efficiently, i.e., minimizing the costs and time of the overall operations. Much of the work in this stage is done by a variety of contractors, and their main cost relates to the travel distances between pick up sites and debris collection sites. The contractors are reimbursed based on the quantity and the content of the debris they collect. Hence, assigning collection zones to contractors to assure that the profits are fairly balanced across contractors, while maintaining efficiency, is an important consideration. By the beginning of this stage the roads have been cleared and there has been time to accumulate greater information on the debris through first hand accounts from contractors. We develop a decision tool to be used by local authorities for identifying the location of the debris collection sites and allocating the affected area among the contractors in a fair manner while ensuring the timely removal of the debris. Here the fairness to the contractors is equivalent to ensuring that the revenue they generate minus the cost they incur due to moving their assets for collecting the debris is similar for each contractor per asset they provide for the operation.

4   Debris Disposal

Debris disposal refers to the activity of taking the collected debris from collection sites to its final destination. Landfills are common final destinations; however, since the debris quantity may be large and landfill capacities are limited, reduction, recycling, and recently reuse strategies are considered as well. Through mathematical models, we analyze the effect of the decisions made in the first two stages on disposal operations.

References

[1] N. Altay and W.G. Green, OR/MS research in disaster operations management, *European Journal of Operations Research*, 175(1), 475-493, 2006.

[2] Congressional Research Service, Disaster Debris Removal After Hurricane Katrina: Status and Associated Issues, http://www.cnie.org/NLE/CRSreports/07May/RL33477.pdf, 2007.

[3] FEMA, Public Assistance Debris Management Guide, Federal Emergency Management Agency (FEMA), https://www.floridadisaster.org/documents/PODs26Oct05_web.pdf, 2007.

[4] R. Longley, FEMA Announces Two New Disaster Recovery Strategies, About.com: US Info, http://usgovinfo.about.com/od/defenseandsecurity/a/femastrategy.htm, 2006.

[5] R. Myers, FEMA extends registration deadline, sticks to debris removal deadline, Beaumont Enterprise, http://www.beaumontenterprise.com/news/local/fema_extends_registration_deadline_sticks_to_debris_removal_deadline_12-11-2008.html, 2008.

[6] N. Rafee, A.R. Karbassi, J. Nouri and E. Safari, Strategic Management of Municipal Debris Aftermath of an Earthquake, *Int. J. Environ. Res.*, 2(2) 205-214, 2004.

[7] G. Solis, Disaster Debris Management, The Disaster Preparedness Resources Centre at The University of British Columbia, http://dsp-psd.pwgsc.gc.ca/Collection/D82-51-1999E.pdf, 1995.

A Nested Logit Model for the Residential Location Choice: Land Use-Transport interactions.

Stefano Carrese

Department of Civil Engineering

Roma Tre University

Stefano Saracchi

(corresponding author)

Department of Civil Engineering

Roma Tre University

Email: saracchi@uniroma3.it

1 Extended Abstract

In this paper, in the land use and transport interaction framework, the type of link (a correlation or causality) between the built environment and the travel behavior has been investigated: the relationship between neighborhood characteristics and travel behavior is taken into account to understand if attitudes and neighborhood preferences influence the residential location choices and travel behavior. It could be useful to examine, whether neighborhood design influences travel behavior or whether travel preferences influence the choice of neighborhood. This could lead to better understand the effect of the transport policies. A lot of studies (Hansen 1959, Lowry 1964, Nuzzolo 2006, Cao, Mokhtarian and Handy 2008) have verified that a suitable transport planning leads to an urban economics development well-matched with the land use aims (e.g. Road Pricing, Congestion Charge and Eco-pass). The literature in transport research underlines the tools to catch the energetic sustainable town; these tools can be divided into 2 different objectives:

1) minimize the total travel time on the road network (in other terms shorten home-work distance)

2) maximize the modal split on behalf of public transport.

In literature it is still not fully known which are the attitudes for the residential location choice (R.L.C.) and if between these attitudes even the travel preferences are included, and how this influences the R.L.C.. To investigate if there is a casuality or an effect between the two choices (R.L.C. and Travel choice) a survey in order to calibrate a nested logit model in the Rome metropolitan region has been conducted. In the paper, after a literature review of the land use transport models, the Experimental Design Theory will be presented and the main effect of a good survey will de discussed. It will be shown how to investigate the problem and which are the main attitudes in travel behavior and

residential location choice in Rome. The Fig.1 marks the sub-area of the land use and transport system and how they are in connection.
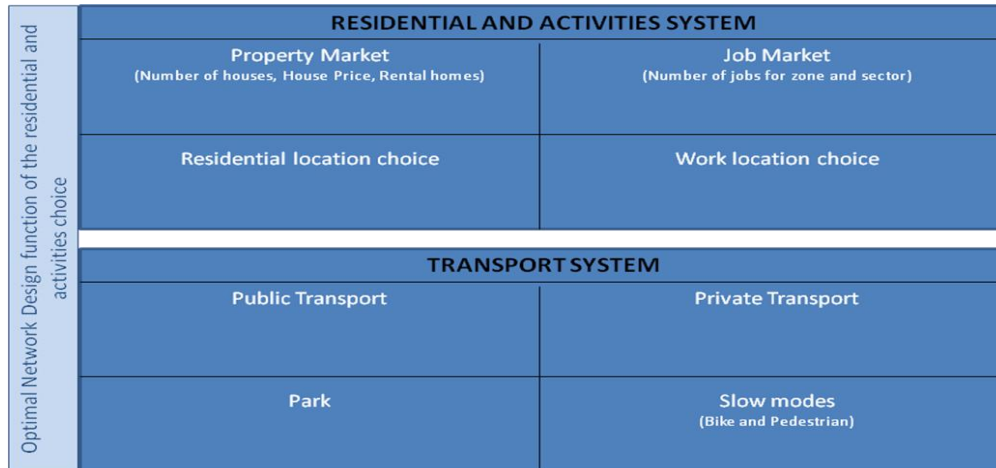
Fig.1 – Land Use Transport interaction.

The cities in Italy are changing according to the decentralization criteria even for the household economic constraint, assuming a feature as different as possible from the old compact urban centre; new roads grow up; the buildings and the blocks are divided by streets of more than two lanes per direction and this gives an increase of the total travel time on the network for the home-work transfer. It's easy to prove that the residential and activities replacing follows the accessibility increase or decrease. The aim of the paper is to individuate the main attributes of the utility function of the R.L.C. to evaluate the different transport policies measure to build the town to serve with an optimal transport solution. In order to obtain a result it will be necessary to make a survey on the population with an optimal experimental design - efficient designs - in order to understand the behavior of the 3 millions Roman residents with the minimum sample and with the minimum error.

Through the survey it is possible to explore the connections among residential accessibility, employment, income, and auto ownership and it will be possible to focus the attention on the potential discrete choice model which will be able to predict the residential choice function of the different transport policies. A first survey has been made and the first results are shown in tab.1 and tab. 2:

Tab.1

Age	Age of House buyer in Rome	Number of interviews made in respect of the latent class
18 – 29	16,33%	(119)
30 – 39	53,06%	(364)
40 – 49	18,37%	(126)
50 – 60	8,16%	(56)
oltre 60	4,08%	(35)

Tab.2

Ranking	Predominant attributes for the R.L.C.	Average μ 1-10	σ s.d.	Respondents
1	**House Price**	**8**	**2,453**	**602** (86%)
3	**House Size**	**7**	**2,145**	**546** (78%)
6	Distance from the city centre	7	2,468	448 (64%)
5	Distance from the work place	7	1,856	462 (66%)
8	Distance from the subway	8	1,414	392 (56%)
10	Shops in the neighborhood	7	2,006	350 (50%)
7	Public Services in the neighborhood	8	1,317	420 (60%)
4	Parking available in the neighborhood	8	1,749	476 (68%)
2	**Neighborhood security**	**9**	**2,063**	**546** (78%)
9	Park in the neighborhood	8	1,600	364 (52%)
11	Distance from the parents house	5	2,375	266 (38%)
12	Privacy	8	1,499	252 (36%)
13	Other	4	2,600	28 (4%)

With the results of this pre-survey a choice tasks survey is calibrated through the Experimental Design Theory. Later through the "Biogeme" software, the nested utility model parameters are calibrated and with the statistical test the validity of the results are estimated. For this study a Nested Logit Model is used (Fig.2).

Fig.2 – the structure model

$$p[j] = p[j/a(j)] \cdot \prod p[r/a(r)]$$
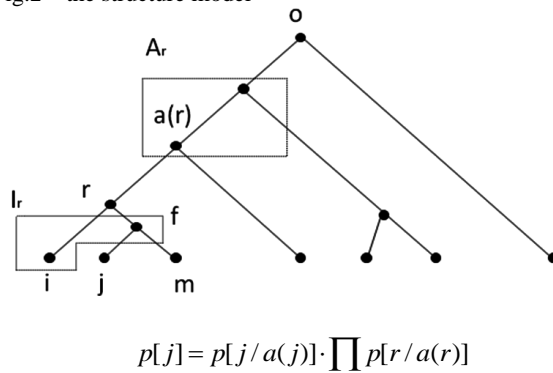
Fig.3 – Johnson distribution

$\beta_{TC} \sim S_B$
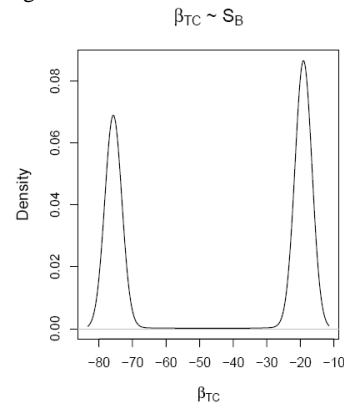
In Fig. 3 the hypothetical distribution of the parameters after the calibration with biogeme is shown. For the parameters calibration even the normal, log-normal and Johnson distribution for the latent class are taken into account in order to view the differences between the outputs and underline which distribution better represents the real R.L.C., even trying to build a Mixed Nested Logit Model.

From these considerations it is possible to understand that the paper is focused on the ability to understand if the first level of the nested logit model is the travel behavior or the Residential Location Choice. In the first case the problem is more complicated to investigate because it is strictly necessary to know the travel behavior attitudes and this is only possible by making a longitudinal survey over the population (a survey from t=t0 to t=t1). In this case the survey should be made on a sample of people that are changing their home in order to catch the differences between the pre-travel

behavior and the post travel one. If this difference in every case is not over a sure amount, it means that the travel behavior attitudes are more strong than the residential location choice. At this point it is easy to see that the paper tries to understand if the predominant model is the first or the second one.

First Nested Logit Model – Travel Behavior influence Residential Location Choice
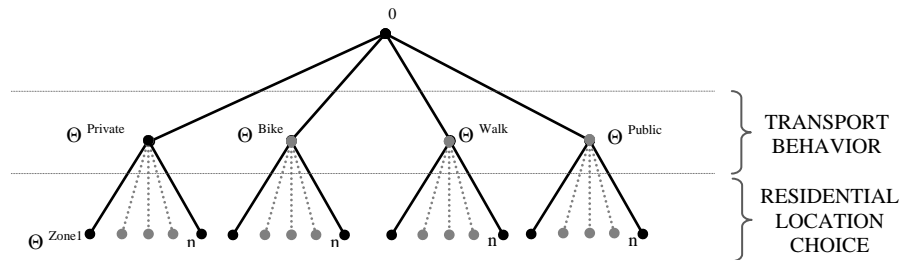
Second Nested Logit Model – Residential Location Choice influence Travel Behavior
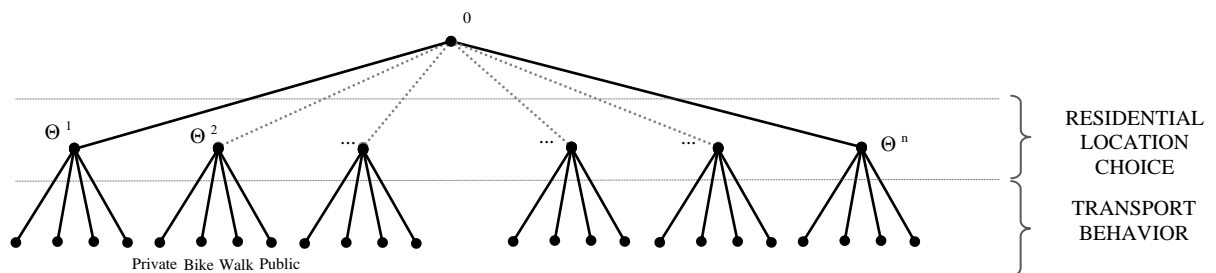
If it is possible to know through a first longitudinal survey, if it is true the first or the second model; and through a second survey to find the values of the Θ, then it will be possible to calibrate the model and to predict the behavior of the house buyer and to build the environment in order to minimize the bad effect in terms of total travel time on the network and pollution in the environment. More attention have to be paid on the Θ study: a) in order to realize a nested logit model, it will be possible to have a single value cross the population (in this case the number of surveys will be small); b) in order to realize a mixed nested logit model it is necessary to carry out a large survey because a continue function across the population will be given on the Θ.

References

[1] MacCallum, R.C., Browne, M.W., Sugawara, H.M.: Power analysis and determination of sample size for covariance structure modeling. Physiol. Methods 1(2), 130–149 (1996)

[2] Newman, P., Kenworthy, J. Sustainability and Cities: Overcoming Automobile Dependence. Island Press, Washington, DC. (1999)

[3] McFadden D., Train K. "Mixed MNL models for discrete response" on Journal of Applied Econometrics. Volume 15, Issue 5, Pages 447-470, (2000)

[4] Bagley, M.N., Mokhtarian, P.L.: The impact of residential neighborhood type on travel behavior: a structural equations modeling approach. Ann. Regional Sci. 36, 279–297 (2002).

[5] Krizek, K.: Residential relocation and changes in urban travel: does neighborhood-scale urban form matter? J. Am. Plann. Assoc. 69(3), 265–281 (2003).

[6] Bierlaire, M. BIOGEME: A free package for the estimation of discrete choice models , Proceedings of the 3rd Swiss Transportation Research Conference, Ascona, Switzerland (2003).

[7] Schwanen, T., Mokhtarian, P.L.: What if you live in the wrong neighborhood? The impact of residential neighborhood type dissonance on distance traveled. Transport. Res. D. 10(2), 127– 151 (2005b)

[8] Chatman, D.G.: How the built environment influences non-work travel: theoretical and empirical essays. PhD Dissertation, Department of Urban Planning, University of California, Los Angeles (2005)

[9] Cao et al.: The causal relationship between the built environment and personal travel choice: evidence from Northern California. PhD Dissertation, Department of Civil and Environmental Engineering, University of California, Davis (2006)

[10] Vance, C. and Hedel, R. The impact of urban form on automobile travel: disentangling causation from correlation, Transportation, 34, pp. 575–588. (2007)

[11] Bierlaire, M., Bolduc, D., and McFadden, D. (2008). The estimation of generalized extreme value models from choice-based samples. Transportation Research Part B: Methodological, 42(4):381{394 (2008).

[12] Carrese S., Saracchi S. "A Land Use-Transport interactions model for the constructions' priority facilities with public expense constraints." Presented to the 4th Kuhmo-Nectar Conference "Transport and Urban Economics" which will be hosted in Technical University of Denmark. June 2009;

[13] Michiel C.J. Bliemer, John M. Rose, David A. Hensher. "Efficient stated choice experiments for estimating nested logit models" Transportation Research Part B: Methodological, Volume 43 Pages 19-35, Issue 1, January 2009.

A column generation approach for a bilevel pricing problem

Aurélie Casier

GOM, Département d'Informatique

Université Libre de Bruxelles, Bld. du Triomphe - CP210/01 - B - 1050 Brussels, Belgium

Email: acasier@ulb.ac.be

Bernard Fortz   **Martine Labbé**

GOM, Département d'Informatique   GOM, Département d'Informatique

Université Libre de Bruxelles   Université Libre de Bruxelles

1 Introduction

Nowadays, in a context of deregulation, overcapacity, increased competition and higher costs, companies need to apply a good pricing to their products or services. However, this is one of the most complex decisions any company is facing. First, customers play an important role in that process, because they react to prices by purchasing - or not - the products. They are looking for good products at lowest prices. But the reaction of competitors is also important. Indeed, as they influence customer choice, they impose practical limitations on pricing alternatives. Hence, companies have to find the best possible prices, low enough so that a large number of customers buy their products, and at the same time high enough to generate large revenues.

We deal with a particular case of a pricing problem, that we call the Product Pricing Problem (PPP), and that consists in determining the prices for a set of new products to be introduced in the market, while taking into account the prices of similar products already present on the market. Each consumer will either buy one of the company products, either go to the competition. In terms of formulation this problem is equivalent to the Network Pricing Problem with Connected Toll Arcs (GCT-NPP), used among other in highway pricing. Indeed, (GCT-NPP) consists of finding the tolls that the company should impose on the highway such as to maximize its revenues. Then, reacting to the tolls, the network users travel on shortest paths from their origins to their respective destinations, i.e. a sub-path of the highway or the toll free path. As one doesn't assume that tolls are additive, each toll sub-path can be priced as independent product (see [3]). This problem has been studied in [1], [2], [3], [4], [5], and it has been shown to be NP-hard ([5], [2]). Unfortunately, until now numerical results show that the bad quality of the linear relaxation and the huge number of variables imply that no commercial solver is able to solve big size instances.

In this paper we propose a column generation method based on an non linear formulation of (PPP). After a judicious Dantzig-Wolfe decomposition, we deal with a linear reformulation of the problem and we propose a polynomial algorithm for solving method for the sub-problem.

2   (Re)formulations and column generation

Pricing problems can easily be formulated thanks to bilevel programming. However, as shown in [4], we can reformulate (PPP) as a one-level mixed integer program. The formulation given below is the one from which we obtain the Dantzig-Wolf reformulation and start the column generation.

As (PPP) is equivalent to (GCT-NPP), we can transpose this problem in a network where products are represented by toll arcs and consumers by commodities that have to choose between using one of the toll arcs (a product of the company) or using a toll free path (representing the competition or a reservation price). Consider a multi-commodity network defined by a set of nodes N, a set of arcs $A \cup B$ and a set of commodities K, each endowed with a demand η^k. Let A be the subset of arcs a on which tolls t_a can be added to the original fixed costs vector c. Let B be the complementary subset of toll free arcs, for which the fixed cost vector c is also given. The cost vector c represents here the "penalty" (or cost in a liberal sense) that each commodity attributes to each product. We assume that for a given taxation policy $t = (t_a)_{a \in A}$, the commodities travel on the shortest paths (it means the path minimizing the penalty of the chosen product) with respect to the tolls and fixed costs on the arcs . Ever since, (PPP) consists in finding a taxation policy t maximizing the revenue of the company. Beyond the variables $(t_a)_{a \in A}$, we introduce the variables $x^k = (x_a^k)_{k \in K, a \in A}$ which specify the flow of each commodity (i.e. $x_a^k = 1$ if the commodity k uses the toll arc a, 0 otherwise), leading to the following non linear formulation, denoted by (HPNL) :

$$\max \quad \sum_{k \in K} \sum_{a \in A} \eta^k x_a^k t_a \tag{1a}$$

$$\sum_{a \in A} (t_a + (c_a^k - c_{od}^k)) x_a^k \leq t_b + (c_b^k - c_{od}^k) \qquad \forall k \in K, \forall b \in A \tag{1b}$$

$$\sum_{a \in A} x_a^k \leq 1 \qquad \forall k \in K \tag{1c}$$

$$(t_a + c_a^k) x_a^k \leq c_{od}^k x_a^k \qquad \forall k \in K, \forall a \in A \tag{1d}$$

$$x_a^k \in \{0, 1\} \qquad \forall k \in K, \forall a \in A \tag{1e}$$

$$t_a \in [0, N_a] \qquad \forall a \in A \tag{1f}$$

where $N_a = \max_k \{M_a^k\}$ with $M_a^k = \max\{0, c_{od}^k - c_a^k\}$. M_a^k represents the maximum toll that the company can add on the arc a such that it stays attractive for the commodity k. N_a is then the maximum toll on arc a to stay attractive for at least one commodity. The objective function (1a) imposes to maximize the company revenue. With the group of constraints (1c), we impose that each commodity chooses exactly one path (one of the toll arcs a or the toll free path $o^k d^k$). The shortest path constraints (1b) come from $\sum_{a \in A}(t_a + c_a^k) x_a^k + c_{od}^k x_{od}^k \leq t_b + c_b^k$ where $x_{od}^k = 1 - \sum_{a \in A} x_a^k$. These constraints impose that the path chosen by the commodity k is the shortest one. Note

that constraints (1d) are redundant but are useful when solving the sub-problem in the column generation process. At last, (1e) and (1f) specify the variables domains.

In this work, we propose a column generation approach for this formulation. The particularity of our developments resides in the fact that we apply such a solving method to an non linear formulation. Our choice of Dantzig-Wolfe decomposition allows us to reformulate the problem as a linear program (the master problem) using convex combination coefficients as new variables.

Let us define $X^a = \{(x_a^k, t_a) : (1d), (1e), (1f)\}, \forall a \in A$. Then, if we denote by $(x_a^{k,j}, t_a^j)_{j=1,...,J}$ the vertices defining X^a, we can see the variables vector as a convex combination of them :

$$\forall a \in A : \begin{pmatrix} x_a^k \\ t_a \end{pmatrix} = \sum_j \lambda_a^j \begin{pmatrix} x_a^{k,j} \\ t_a^j \end{pmatrix} = \lambda_a^1 \begin{pmatrix} x_a^{k,1} \\ t_a^1 \end{pmatrix} + \lambda_a^2 \begin{pmatrix} x_a^{k,2} \\ t_a^2 \end{pmatrix} + ... + \lambda_a^J \begin{pmatrix} x_a^{k,J} \\ t_a^J \end{pmatrix}$$

with $\sum_{j=1,...,J} \lambda_a^j = 1$ and $\lambda_a^j \in [0,1], \forall j \in \{1,...,J\}$. This leads us to the following reformulation :

$$\max \quad \sum_{k \in K} \sum_{a \in A} \sum_j \lambda_a^j \eta^k x_a^{k,j} t_a^j$$

$$\sum_{a \in A} \sum_j \lambda_a^j (t_a^j + (c_a^k - c_{od}^k)) x_a^{k,j} \leq \sum_j \lambda_b^j t_b^j + (c_b^k - c_{od}^k) \qquad \forall k \in K, \forall b \in A \quad \leftrightarrow \delta_b^k \geq 0$$

$$\sum_{a \in A} \sum_j \lambda_a^j x_a^{k,j} \leq 1 \qquad \forall k \in K \qquad \leftrightarrow \gamma^k \geq 0$$

$$\sum_j \lambda_a^j = 1 \qquad \forall a \in A \qquad \leftrightarrow \mu_a \in \mathbb{R}$$

$$\lambda_a^j \in [0,1] \qquad \forall j, \forall a \in A$$

Afterwards, we apply a column generation on the linear relaxation of this reformulation (LRMP), called Master Problem (MP). The corresponding Sub-Problem, aiming at maximizing the reduce cost at each iteration of the Simplex when solving (PM), can be decomposed by arc, is denoted by (SP$_a$) and is the following, where δ_b^k, γ^k et μ_a are the dual variables of (LRMP).

$$\max \quad \sum_{k \in K} \eta^k x_a^k t_a - \left[\sum_{k \in K} \sum_{b \in A} (t_a + (c_a^k - c_{od}^k)) x_a^k \delta_b^k - \sum_{k \in K} \delta_a^k t_a + \sum_{k \in K} x_a^k \gamma^k + \mu_a \right]$$

$$(t_a + c_a^k) x_a^k \leq c_{od}^k x_a^k \qquad \forall k \in K \qquad (3a)$$

$$x_a^k \in \{0,1\} \qquad \forall k \in K \qquad (3b)$$

$$t_a \in [0, N_a] \qquad (3c)$$

In the following, we propose a polynomial method to solve (SP$_a$). It sums up to find the maximum of a sum of piecewise linear functions with a finite number of breakpoints. Indeed, after rearranging of the terms and omission of the index a, (SP$_a$) becomes :

$$s_a = \max \quad \overbrace{\left(\sum_{k \in K} \underbrace{x^k \left[d^k t - e^k \right] + f^k t}_{z^k} \right)}^{z} - \mu$$

$$(t_a + c_a^k) x_a^k \leq c_{od}^k x_a^k \qquad \forall k \in K \qquad (4a)$$

$$x_a^k \in \{0,1\} \qquad \forall k \in K \qquad (4b)$$

$$t_a \in [0, N_a] \qquad (4c)$$

where we define $d^k = \eta^k - \sum_{b \in A} \delta_b^k, e^k = (c_a^k - c_{od}^k) \sum_{b \in A} \delta_b^k + \gamma^k, f^k = \delta_a^k, \mu = \mu_a$. Then $d^k, e^k, \mu \in$

$\mathbb{R}, f^k \geq 0$. When solving (SP_a), a new column will be added in (LRMP) if and only if $z > \mu$ (i.e. iff the reduce cost s_a is positive). We derive a polynomial algorithm to compute the optimal value of z which is described below and which is based on the observation that z^k are piecewise linear functions. Indeed, if $t > M_a^k$, the arc a is more expensive than the toll free path for the commodity k, then $x_a^k = 0$. Hence $z^k = f^k t$, which is an increasing function in t going through $(0,0)$. Reversly, if $t \leq M_a^k$, the arc a is interesting with respect to the toll free path. We have to find $\max_{x^k \in \{0,1\}}[x^k(d^k t - e^k)] + f^k t$. The value of x_a^k will then depend on the sign of $(d^k t - e^k)$. If it is positive, we have an interest to put $x_a^k = 1$, otherwise, $x_a^k = 0$. More precisely, if $t > \frac{e^k}{d^k}, x_a^k = 1$ and $z^k = (d^k t - e^k)] + f^k t$, which is a linear (increasing or decreasing) function in t going through $(0, -e^k)$. Otherwise, $x_a^k = 0$ and $z^k = f^k t$, which leads us to the same case as $t > M_a^k$.

The function $z = \sum_{k \in K} z^k$ is then also a piecewize linear function whose breakpoints are all those of the functions z^k, that is a subset of $\{0, (\frac{e^k}{d^k})_{k \in K}, (M_a^k)_{a \in A, k \in K}, N_a\}$. Moreover, we know that the maximum of such a function is reached at one of it breakpoints. The solving algorithm is then the following one :

step 1 For each $k \in K$, compute the set of z^k-breakpoints B_k, where $|B_k| \leq 4$.

step 2 Compute the set of z-breakpoints $B = \bigcup_k B_k$ and remove the possible duplicates.

step 3 For each $b \in B$ and for each $k \in K$, compute $z^k(b)$ and add this value to $z(b)$.

step 4 The optimal value of $t = argmax_b\{z(b)\}$ and the optimal value of x^k is known thanks to the definition of the function z^k.

Références

[1] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard, "A Bilevel Model and Solution Algorithm for a Freight Tariff Setting Problem.", *Transportation Science* 34 :289-302 (2000).

[2] G. Heilporn, Network Pricing Problems : complexity, polyhedral study and solution approaches, Ph.D. thesis, Université Libre de Bruxelles, Université de Montréal, published to VDM Publishing House Ltd, ISBN 978-3-639-14688-2.

[3] G. Heilporn, M. Labbé, P. Marcotte and G. Savard, "A parallel between two classes of pricing problems in transportation and economics.", *submitted to Journal of Revenue & Pricing Management.*

[4] M. Labbé, P. Marcotte, and G. Savard, "A Bilevel Model of Taxation and its Application to Optimal Highway Pricing.", *Management Science* 44 :1608-1622 (1998).

[5] S. Roch, P. Marcotte, and G. Savard, "Design and Analysis of an Approximation Algorithm for Stackelberg Network Pricing.", *Networks* 46 :57-67 (2005).

Network Pricing Problem: the case of the European Air Traffic Management

Lorenzo Castelli

Dipartimento di Elettrotecnica, Elettronica e Informatica

Università degli Studi di Trieste, Via A. Valerio 10, 34127 Trieste, Italy

Email: castelli@units.it

Martine Labbé and Alessia Violin

Département d'Informatique

Université Libre de Bruxelles

1   Introduction

In Europe, each Air Navigation Service Provider (ANSP) finances its activities by charging all airlines using its airspace. In particular, it imposes to each flight the 'en route charges' which, according to European regulation EC 1794/2006, are calculated as the product of the distance flown by the flight within its national territory, the Maximum Take-Off Weight of the aircraft performing the flight, and a national Unit Rate which is annually fixed by it. As these charges usually account for around 10-20% of the cost of a flight, they can influence the route choice: airlines may decide to fly longer to avoid countries with high Unit Rates [2]. Currently in most states the Unit Rate is set to allow the ANSP to completely recover all the costs it incurs. However in the next years ANSPs, which nowadays are mostly public service agencies, are likely to move toward becoming private service providing companies [1]. In this case, an ANSP would like to fix its Unit Rate to maximize its revenues.

In this paper we show that this optimal Unit Rate value can be identified through a Network Pricing Problem (NPP) formulation in the form of Bilevel Programming (see [3]) where the leader (i.e., the ANSP) owns a set of arcs (the airways in its national airspace) and charges the commodities (i.e., flights) passing through them. Flights have a rational behavior and look for the minimum cost path through the network. As Unit Rate values are decided once per year (in November) and are valid for the following year, here we analyze the strategic airline route choices, i.e.,when short-term unexpected events like weather conditions or airspace congestion are not yet considered. We

prove that the NPP approach to fix the charge on a single toll arc (see, e.g., [3]) can be extended to our case where the charge on each arc is proportional to a constant. In fact, as the Unit Rate is unique for each country and the charge to be paid on an arc linearly depends on it, the leader has to decide on this single value only. Our preliminary findings show that flight travel choices do depend on the Unit Rate value set by the ANSP and identify the revenue-maximizing Unit Rate value.

2 Mathematical Model and Computational Results

We consider the set A of toll arcs, i.e., a flight is charged by the ANSP when passing through any arc of A. Let N be the set of all endpoints of the arcs in A. We denote as $(i, j) \in A$ the generic toll arc where both i and j belong to N. If K is the set of all the commodities, the charge or toll to be paid by the generic flight $k \in K$ is equal to the product of the Unit Rate T fixed by the ANSP, the distance $l_{i,j}$ of the arc (i, j) and the factor w^k depending on the Maximum Take-Off Weight of the aircraft performing the flight. If o^k and d^k are the origin and destination points of flight $k \in K$, respectively, we denote as $d(o^k, i)$ the minimum cost path from origin o^k to node i for all $i, k \in N \times K$ and as $d(j, d^k)$ the minimum cost path from node j to destination d^k for all $j, k \in N \times K$. In this way we represent the portion of flight which is performed outside the airspace controlled by the ANSP. In addition we consider the possibility for each flight to reach its destination without crossing any arc in A. This toll free path should exist for each commodity to guarantee an upper bound of the Unit Rate that the leader can impose on its arcs. We denote as r^k the cost of the minimum cost toll free path. We finally denote as c^k the unit cost of flight k which takes into account all other flight-related costs (e.g., fuel, maintenance and crew costs) besides the en route charges.

The Route Charges Pricing Problem (RCCP) can be written as:

$$\max_{T,x} \quad T * \left[\sum_k \sum_{i,j} x_{i,j}^k \, l_{i,j} \, w^k \right] \tag{1}$$

$$T \geq 0$$

$$\arg\min_{x,y} \quad \sum_k \left\{ \sum_{i,j} \left[d(o^k, i) + l_{i,j}(c^k + Tw^k) + d(j, d^k) \right] x_{i,j}^k + r^k y^k \right\} \tag{2}$$

$$\sum_{i,j} x_{i,j}^k + y^k = 1 \qquad \forall k \in K \tag{3}$$

$$x_{i,j}^k \in \{0, 1\} \qquad \forall i, j, k \in N \times N \times K$$

$$y^k \in \{0, 1\} \qquad \forall k \in K$$

where T is the nonnegative decision variable representing the Unit Rate fixed by the leader and holding on all toll arcs, $x_{i,j}^k$ is a set of binary variables equal to 1 if arc (i, j) is chosen by commodity k and 0 otherwise, and y^k is a set of binary variables equal to 1 if the toll free path with cost r^k is chosen by commodity k, 0 otherwise. The leader chooses the Unit Rate value T which maximizes

its revenue (Equation 1), and knows the reaction of the followers: each commodity considers all possible paths between its origin and destination, and chooses the minimum cost path (Equations 2 and 3). As there is just one decision variable T at the leader level, the bilevel problem can be solved through the following procedure:

1. For each commodity, we calculate the costs for all possible path between its origin and its destination (Equation 2). As the costs of the toll arcs depend on T, we identify the values of T where the commodity has convenience in changing its path choice. We obtain a piecewise linear concave function, bounded at the upper limit by the toll free path r^k, Figure 1(a).

2. The leader's revenue for a single commodity is a non-continuous function, linear in each interval of T previously determined, Figure 1(b).

3. The above steps are repeated for each commodity to find all significant T values. Finally for each T, the leader's total revenue is determined as the sum of the revenues from each commodity, Figure 1(c). It is then straightforward to identify the Unit Rate value which maximizes the leader's revenues.

(a) Cost for one commodity   (b) Leader's revenue for one comm   (c) Total leader's revenue
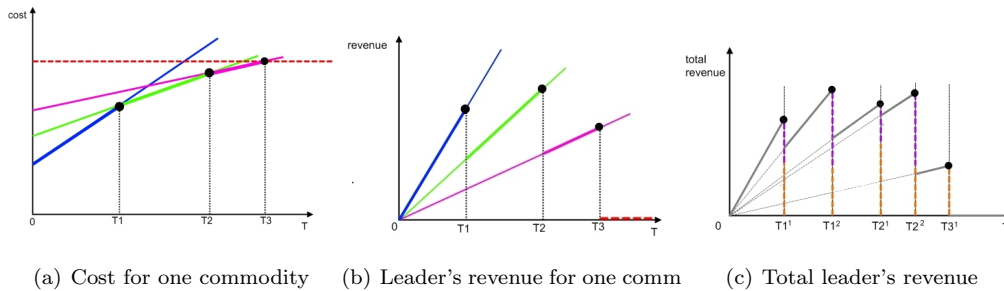
Figure 1: Computational procedure - Functions on T

We delineate a preliminary case study considering a few commodities flying over a central European country. The network topology and arc distances for 10 Origin/Destination pairs have been extracted with the aid of the 'System for traffic Assignment and Analysis at a Macroscopic level' (SAAM) software relying on actual flight data from 29 June 2007. We choose seven different types of aircraft, which are commonly used for European flights. We then derive all other flight cost data from standard figures publicly available. We finally solve the RCCP as previously described. Based on the available commodities, Figure 2 displays the revenue function for this central European ANSP and spots the Unit Rate value which maximizes its revenues.

3   Conclusions

This paper formulates a Network Pricing Problem addressing the case where an authority controlling a set of arcs fixes a unique value such that any commodity traversing these arcs has to
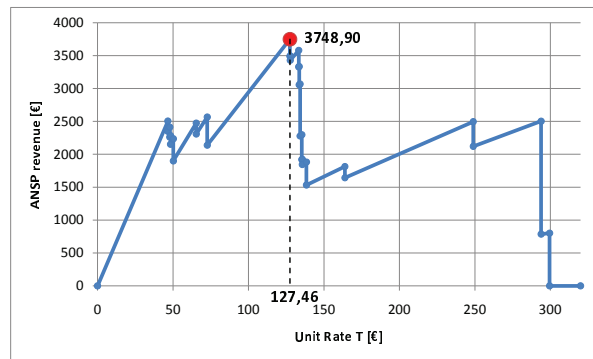
Figure 2: Preliminary example - Revenue of the ANSP

pay a toll proportional to this common value. This framework depicts the way most European ANSPs are likely to behave in the near future when they determine the Unit Rate values which maximize their revenues. In fact, an airline flying through an airway under the responsibility of a given ANSP has to pay to it the so-called en route charges, and according to European regulations these charges are proportional to the Unit Rate set by the ANSP. By exploiting the structure of the problem, we propose an exact algorithm to compute the optimal Unit Rate relying on real air traffic data and realistic flight cost figures. The algorithm is polynomial except for the first step, which enumerates all possible paths in the network for a given origin/destination pair. However, the air network has a fairly simple topology, meaning there are only a few different routes possible for each flight. Our results also suggest that the Unit Rate can indeed be an instrument for an ANSP to modify the path choice of commodities.

We are currently addressing the 'competition' between more ANSPs, as they simultaneously fix their Unit Rates. In this case we face a bilevel problem with multiple leaders. Thus we are investigating whether an equilibrium can be reached, and if cooperation could bring advantages. Finally, the analysis should be carried out on a larger data set and on accurate information on exact airline costs and preferences.

References

[1] K. Button and G. McDougall, "Institutional and structure changes in air navigation service-providing organizations", *Journal of Air Transport Management* 12, 236-252 (2006).

[2] L. Castelli, R. Pesenti and W. Ukovich, "Analysis of Airline Operational Behaviours", Final Report of Work Package 4 of EUROCONTROL Care Innovative Project "Innovative Route Charging Schemes", Brussels, Belgium, 2004.

[3] M. Labbé, P. Marcotte and G. Savard, "A Bilevel Model of Taxation and its Application to Optimal Highway Pricing", *Management Science* 44, 1608-1622 (1998).

Location and Routing Problems for Drug Distribution

Alberto Ceselli

Dipartimento di Tecnologie dell'Informazione

Università degli Studi di Milano

Giovanni Righini

Dipartimento di Tecnologie dell'Informazione

Università degli Studi di Milano

Email: giovanni.righini@unimi.it

Chetan Sharma

Indian Institute of Technology - Kharagpur

Emanuele Tresoldi

Dipartimento di Tecnologie dell'Informazione

Università degli Studi di Milano

1 Introduction

Mathematical programming models and algorithms have been successfully used for decades to optimize operations in distribution logistics: typical examples concern freight carriers, mail services and on-demand pick-up and delivery services.

A more recent field of investigation concerns the application of similar techniques to the optimization of logistics operations in health care systems and emergency management. These sectors are characterized by a larger dependency on "human factors", such as the behavior of the customers (which is often unpredictable), fairness in service provision (which is not an issue in industrial logistics) and lack of reliable historical data (because of the uniqueness of the events considered, especially in case of emergency management).

In this paper we present some preliminary studies on the development of exact optimization algorithms for a variation of the vehicle routing problem (VRP) arising in the context of the distribution of vaccines and anti-viral drugs in case of a pandemic outbreak. The problem requires

a fleet of vehicles to reach the maximum number of citizens within a specified time limit. We present an algorithm based on column generation, where the pricing subproblem is solved through advanced dynamic programming techniques.

We also consider the effect of combining the strategy of delivering the drugs at citizens' homes with the strategy of establishing distribution points where the citizens go by their own means to receive treatments or drugs.

The starting point for our study is a model by Shen et al. [8], who presented a stochastic VRP model which is then reformulated and solved as a deterministic VRP with a tabu search algorithm. Two objectives are considered in a hierarchical way: the most important requires to minimize the fraction of population which is not visited in time; the second objective requires to minimize the arrival time at the distribution points. The number of available vehicles is given and there are no capacity constraints, because the goods to be transported are small, both in weight and in volume. The binding constraint is a deadline within which the distribution points must be visited, as far as possible.

2 A mathematical programming formulation

Our column generation algorithm is based on a set partitioning reformulation of the problem. We do not include any source of non-determinism in this model. The model is as follows.

$$\max \sum_{i \in \mathcal{N}} \sum_{k \in \mathcal{K}} d_i a_{ik} z_k \tag{1}$$

$$\text{s.t.} \sum_{k \in \mathcal{K}} a_{ik} z_k \leq 1 \qquad \forall i \in \mathcal{N} \tag{2}$$

$$\sum_{k \in \mathcal{K}} z_k \leq V \tag{3}$$

$$z_k \quad \text{binary} \qquad \forall k \in \mathcal{K} \tag{4}$$

where $\mathcal{N}$ is the set of sites to be visited, d_i is the number of persons served when site $i \in \mathcal{N}$ is visited, $\mathcal{K}$ is the set of feasible routes for the vehicles, a_{ik} is equal to 1 if and only if site $i \in \mathcal{N}$ is visited in route $k \in \mathcal{K}$ and z_k is a binary variable corresponding to route $k \in \mathcal{K}$. We refer to (1)-(4) as the *master problem*.

The problem turns out to be a special case of the Team Orienteering Problem, because the minimization of the fraction of population that is not visited is equivalent to maximize a "profit" from visiting a subset of the sites. Exact algorithms for the Team Orienteering Problem have been proposed by Boussier et al. [3]. Recent heuristics are those of Tang and Miller-Hooks [9], Archetti et al. [1] and Ke et al. [4].

Since the model above has an exponential number of columns, its linear relaxation is solved by column generation. The pricing problem consists of finding columns with negative reduced cost.

The expression of the reduced costs is as follows:

$$\sum_i d_i a_{ik} - \sum_i \pi_i x_i - \mu \tag{5}$$

where π is the vector of dual variables corresponding to constraints (2) and μ is the scalar dual variable corresponding to constraint (3). The constraints of the pricing problem require that:

- the route is elementary;

- the route must start at the depot (but is not required to go back to it within the deadline);

- all sites along the route must be visited within a specified deadline.

Pricing algorithms. Pricing is the most time-consuming part in the branch-and-price exact algorithm relying upon the previous formulation. To speed up the pricing phase we rely on both heuristic and exact pricing algorithms. Exact pricing is done by bi-directional dynamic programming with decremental state space relaxation, following the approach described in Righini and Salani [5] [6] [7]. Heuristic pricing relies on greedy algorithms and dynamic programming with relaxed dominance conditions.

3   Extensions

The model presented above can be extended in several ways to capture more details of a real logistics system for drug distribution. This can be done by including more features at either pricing or master level.

First, we consider more complex logistic networks, including the management of several depots with different stock capacities or heterogeneous fleets of vehicles [2]. This can be done by partitioning set $\mathcal{K}$ into different sets of feasible routes for each vehicle type and for each depot.

Second, we tackle the problem of providing service for the same persons in different places (at home, at their workplace, at school). In a similar way, since some people may not be found at their homes when the vehicle visits it, multiple visits to the same sites might be imposed; in this case we consider the worst-case scenario in which each person is found only during the last visit. This can be done by modifications to extension rules and resource consumption rules in the dynamic programming algorithm which is used for pricing columns.

Third, we consider a more detailed management of service time and served demand, involving hard and soft time windows and site demand split. These features require further adaptations of the pricing algorithms.

Furthermore additional advanced features, like management of non-determinism in travel time and demand, or success in finding people at each visit, can be taken into account by including special purpose recourse strategies in our framework.

Finally, we discuss alternative and hybrid distribution strategies. In particular, we consider the option of providing goods by means of either distribution centers or door-to-door delivery. This can be modeled by a master problem with two types of columns, generated by two independent pricing algorithms.

References

[1] C. Archetti, A. Hertz and M.G. Speranza, "Metaheuristics for the team orienteering problem", *Journal of Heuristics* 13, 49-76 (2007).

[2] R. Baldacci, M. Battarra and D. Vigo, "Routing a Heterogeneous Fleet of Vehicles", in *The Vehicle Routing Problem: latest advances and new challenges*, B.L. Golden, S. Raghavan and E. Wasil (eds), 3-27, Springer Science + Business Media, 2008.

[3] S. Boussier, D. Feillet and M. Gendreau, "An exact algorithm for team orienteering problems", *4OR* 5, 211-230 (2007).

[4] L. Ke, C. Archetti and Z. Feng, "Ants can solve the team orienteering problem", *Computers and Industrial Engineering* 54, 648-665 (2008).

[5] G. Righini and M. Salani, "Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints", *Discrete Optimization* 3, 255-273 (2006).

[6] G. Righini and M. Salani, "New Dynamic Programming Algorithms for the Resource Constrained Elementary Shortest Path Problem", *Networks* 51, 155-170 (2008).

[7] G. Righini and M. Salani, "Decremental state space relaxation strategies and initialization heuristics for solving the Orienteering Problem with Time Windows with dynamic programming", *Computers and Operations Research* 36, 1191-1203 (2009).

[8] Z. Shen, F. Ordonez and M. Dessouky, "The minimum unmet demand stochastic vehicle routing problem", Technical report, CREATE, 2006.

[9] H. Tang and E. Miller-Hooks, "A tabu search heuristic for the team orienteering problem", *Computers and OR* 32, 1379-1407 (2005).

Game Theoretic Models for Competition in Public Transit Services

Eddie Chan

Department of Systems Engineering and Engineering Management

The Chinese University of Hong Kong

Janny M. Y. Leung

Department of Systems Engineering and Engineering Management

The Chinese University of Hong Kong, Shatin, New Territories, Hong Kong

Email: janny@se.cuhk.edu.hk

1 Introduction

Metropolitan areas have accounted for the majority of increases in population and economic growth in recent decades. China's phenomenal economic development has been fuelled by growth in the major cities, many of which has over 5 million in population. Metropolitan areas account for over half of the population, and a significant majority of the GDP, of the United States. As the geographical size and population of major metropolitan areas have increased, much economic activity remain focussed in the central business districts of the metropolises, thus the average travel distances for work have not decreased as expected. The average commuting distance for London is over 10 kilometres. The need to travel by the populace has placed significant burden on the transport systems of metropolitan areas, leading to increased traffic congestion and attendant safety and environmental concerns.

Development of transport infrastructure and public transit services have not kept apace with the swell and sprawl of metropolitan areas, with serious congestion occurring in central business districts and insufficient coverage in peripheral areas. In metropolises where public transit services are provided by private firms in a relatively free market, operators tend to focus on high-profit routes and outlying smaller communities are under-served. In Hong Kong, the already congested Central business district is often jammed with half-empty double-decker buses from all the bus operators, while bus services to satellite communities in the New Territories are very infrequent and expensive.

In this paper, we discuss some game-theoretic models that can be used to investigate the competitive situation when several service providers offer public transit services, and study the impact on the total set of services offered to the public and the resultant level of ridership of the system. The competition among the operators can be modelled by a class of games called potential games. We discuss mathematical programmes that can be used to find the Nash equilibria for these games. By

examining the equilibria solutions, we can examine the relative merits and tradeoffs for different structures of the transit networks, and the interplay between the services offered and the overall ridership of the system. We hope that our modelling and analysis may provide some insight on the types and bundling of routes being offered by operators, and the locations for transportation interchanges and hubs.

2 Background

Whilst not explicitly acknowledged, concepts of game theory have been pervasively used in traffic studies. As Fisk (1984) pointed out, the famous Wardrop's (1952) user-equilibrium principle is essentially the condition for a Nash (1950) game-theoretic equilibrium among road-users, since no driver can reduce his/her travel time by switching to a different route choice. Wardrop's principle has been a cornerstone in road traffic research for decades. For an overview of traffic equilibrium models, see Patriksson and Labbe (2004).

Other researchers have developed specific game-theoretic models for transport-related problems. Bell (2000) investigates network reliability by studying a zero-sum game between a cost-minimising driver and a 'demon' that sets the link costs. This game is a 'concept' game in the sense that the demon is not a real player, and is used to explore the worst-case scenarios faced by the driver. Other researchers have also explored concept games among road users. James (1998) studies a game among *n* road-users where any player's utility of using the road segment decreases when there are more users. Levinson (2005) also studies congestion by investigating a game where the players (drivers) choose their departure times. Pedersen (2003) investigates road safety by a game where players choose the behavioural level of driving aggression. All these games study the competition among road users. Holland and Prashker (2006) give an excellent review of recent literature on non-cooperative games in transport research.

Surprisingly, studies on the competitive situation amongst public transit operators have received little attention from transport researchers. Castell et al. (2004) modelled a Stackelberg game between two authorities (one determining flow, and the other capacities) in a freight transport network, which is different to a passenger transit network since the route choice is not determined by the transportee (freight). According to Holland and Prashker (2006), the "small number of such games is surprising, considering that NCGT [non-cooperative game theory] seems a natural tool for analysing relations between authorities. ... Trends such as tendering and privatisation, that have a vital role on today's transport agenda, also seem apt to be modelled through games between authorities". Surprisingly, there has been very little research along this line.

Some researchers have studied games between authorities and travellers. Fisk (1984) investigates a Stackelberg game between the authority that sets traffic signals and all travellers who then finds the user-equilibrium solution. Chen and Ben-Akiva (1998) investigates a similar game in a dynamic setting. Reyniers (1992) studies a game between the railway operators who sets the capacities

for different fare classes and the passengers who chooses which class to use. Hollander et al. (2006) studies a game between the parking authority and travellers to explore the incentives for public transport ridership. All these games, however, only involve one operator/authority. Only few researchers have investigated games with several operators and passengers. Van Zuylen and Taale (2004) studies a game with two authorities (one for urban roads and one for ring roads) and the driving public. None of the previous research have studied the strategic competition among public transit operators.

3 Preliminary Results

We have made some preliminary investigation into the strategic gaming situation among competing public transit service providers. In our first-cut model, we assume that all the operators have the same cost and price structure, and that the total ridership between each origin-destination pair is equally divided among all the operators that service that particular route. In this setting, a player of the game is the service provider, and its strategy is the set of routes that it chooses to offer service. Each player tries to maximize its total profit, and a Nash equilibrium occurs when no player can improve its profit by unilaterally changing the set of routes it services.

We can show that this game can be modelled as what is known as a *potential game* (first introduced by Rosenthal, 1973) where the equilibrium can be computed by solving an auxiliary mathematical programme with a 'potential' function as a surrogate objective. The solvability for the Nash equilibrium allows us to make some comparisons between the competitive equilibrium and a centralised monopolistic operation and draw some insights.

Using this initial framework, we have also explored the impact of the network structure on the profit for the service providers. We consider a service area with *n* townships and compared the equilibrium solution for a network structure where direct point-to-point services are offered between every pair of townships, to the equilibrium for a hub-and-spoke network where every route between any two townships involves an interchange via a central hub. In the second network structure, the routes offered are between a township and the central hub, and the total ridership from each origin (to all destinations) is consolidated into the ridership from the origin to the central hub. For each service provider, the profit from a route depends on the operating cost of offering the service, the total revenue due to the ridership and the number of competitors also servicing that route. For the simplistic case where the ridership between every pair of townships are the same and all fixed operating costs are the same, the overall profits depends on the ratio of ridership to route operating cost.

A more realistic model would allow for ridership to depend on origin-destination pairs, on the network infrastructure and also on the set of transet services available. This may lead to a bi-level model where not only do operators compete with each other but the passengers preferences and patronage depends on the set of services offered by the transit operators.

References

[1] Bell, M.G.H. (2000) "A game theory approach to measuring the performance reliability of transport networks", *Transportation Research: Part B, 34,* 533-545.

[2] Castelli, L., G. Longo, R. Pesenti and W. Ukovich (2004) "Two-player noncooperative games over a freight transportation network", *Transportation Science, 38(2),* 149-159.

[3] Chen, O.J. and M.E. Ben-Akiva (1998) "Game-theoretic formulations of interaction between dynamic traffic control and dynamic traffic assignment", In *Tranportation Research Record: Journal of the Transportation Research Board,* No. 1617, National Research Council, Washington, DC., pp. 179-188.

[4] Fisk, C.S. (1984) "Game theory and transportation systems modelling", *Transportation Research B, 18B (4/5),* 301-313.

[5] James, T. (1998) "A game theoretic model of road usage", *Proceedings of the 3rd IMA International Conference on Mathematics in Transport and Control,* 401-409.

[6] Hollander, Y. and J.N. Prashker (2006) "The applicability of non-cooperative game theory in transport analysis", *Transportation, 33,* 481-496.

[7] Levinson, D. (2005) "Micro-foundations of congestion and pricing: a game theory perspective", *Transportation Research: Part A, 39,* 691-704.

[8] Monderer, D. and L.S. Shapley (1996) "Potential Games", *Games and Economic Behaviour, 14,* 124-143.

[9] Nash, J. (1950) "Equilibrium points in n-person games", *Proceedings of the National Academy of Sciences* 36(1):48-49.

[10] Patriksson, M. and M. Labbé (eds.) (2004) *Transportation Planning: State of the Art,* Kluwer Academic Publishers.

[11] Pedersen, P.A. (2003) "Moral Hazard in traffic games", *Journal of Transport Economic Policy 37 (1),* 47-68.

[12] Reyniers, D. (1992) "Crowding levels and fare classes in public transport", In Griffiths, J. (ed.) *Mathematics in Transport planning and Control,* Oxford University Press, pp. 319-327.

[13] Rosenthal, R.W. (1973) "A class of games possessing pure-strategy Nash equilibria", *International Journal of Game Theory, 2,* 65-67.

[14] Van Zuylen, H.J. and H. Taale (2004) "Urban network with ring roads: a two-level three player game", In: *Proceedings of the 83rd Annual Meeting of the Transportation Research Board, Washington DC.*

[15] von Stackelberg, H. F. (1934) *Marktform und Gleichgewicht (Market and Equilibrium)*, Vienna.

[16] Wardrop. J.G. (1952) "Some theoretical aspects of road traffic research", *Proceedings of the Institution of Civil Engineers, Part II, 1,* 325-378.

A clustering approach to estimate route travel time distributions.

Wouter R. J. Charle

Francesco Viti

Chris M. J. Tampère

Centre for Industrial Management

Section Traffic and Infrastructure

Faculty of Mechanical Engineering

Katholieke Universiteit Leuven

Email: wouter.charle@cib.kuleuven.be

1   Introduction

Accurate network travel time estimation is today one of the most challenging problems in traffic theory. The mainstream research on travel time estimation concentrates on the estimation of mean route travel time or some measures of travel time reliability (ie. 10th and 90th percentiles). However, given the growing detail in travel time measurements it is also possible to estimate route travel time distributions. This serves a broader spectrum of applications and provides more usefull information. For this goal, this research presents a method to calculate the travel time histogram of a route, based on link travel time observations.

The aim of this new method is to allow the development of an advanced route planner that is able to optimize route choice reliability. The notion of reliability is defined by the end-user of the route planner and highly depends on the properties of the route travel time distribution. Central in the development of the route planner is the distinction between (cheap) off-line storage and computations and (expensive) on-line computations. For that it is important to minimize the on-line computational effort of calculating a route travel time histogram. The method described in this study can be deployed beyond route planning applications and the field of traffic management. Other applications can be for instance the location optimization of logistic hubs (ie. airports, packaging services, . . .) or public services (ie. hospitals, firestations, . . .) and the evaluation of network performance.

2 Conceptual framework

The focal point of this study is the on-line calculation of a route travel time histogram based on historical link travel time observations. In solving this problem two cases can be distinguished:

- ad hoc: the histogram of route travel times is calculated while searching a route

- ex post: the route is already know when calculating the histogram

The ad hoc calculation of the route travel time histogram has the advantage that a route can be optimized based on full route statistics. More specifically, the statistics of several sections of a route can be compared to select a route over which for example the variance of the travel time distribution is minimized. This advantage is not present in the ex post calculation of a histogram of route travel times. The ex post calculation of the histogram is in this way a simpler problem, but is non-trivial when on-line computational efficiency is an issue. In this paper we focus on the procedure to derive these histograms, not on the routing procedure. For this reason we will refer mainly to the ex-post case.

In this section two different approaches to the ex post calculation are discussed. For both approaches it is assumed that there are n links l_i in the network, and m travel time observations TT_{ij}, made on m different times t_j for each link l_i. The route contains h network links and the resulting histogram is made up of k bins.

A first straightforward approach is to take the sum of the instantaneous travel times TT_{ij} over all links l_i contained in the route for each t_j. The route travel time histogram is then calculated from these sums. Neglecting the calculation of the histogram, the computational complexity of this approach is of order $h \times m$. Considering that in practical applications this has to be done for all relevant routes in a network and for different times of the day, it is easy to understand that it would not be a feasible approach for an on-line application.

It is possible to enhance the computational efficiency by pre-processing the travel time data off-line and using clustering techniques. This is investigated in the second approach. In this approach the travel time histograms of each individual link l_i are calculated off-line. This changes the amount of data per link from m to k in the on-line calculation of the route travel time histogram. However the calculation of the route travel time histogram is no longer a summation of the data of each individual link, but is the convolution of the link histograms. The problem is that the calculation of a density function of the sum of two stochastic variables by convoluting their density functions requires that these variables are statistically independent. This is often not the case for the observed travel times of two different links in a network. This statistical dependence can be due to, for instance, congestion spillback over links, similar road conditions, or other structural similarities. The dependencies between link travel time statistics are in our approach removed by redefining the network. This is done off-line by means of a clustering algorithm which defines

new links as a combination of links that have correlated travel time fluctuations. The travel time histogram of each cluster of links, denoted by clusterlinks in the remainder, is calculated off-line by making use of the first approach. As a result the travel time statistics of the clusterlinks in the resulting network are statistically independent. Because a clusterlink is the combination of 1 or more links, the length of the route is reduced to $g \leq h$ clusterlinks. To cope efficiently with the convolutions, the histograms are stored as fourier transformations (FT) off-line. The advantage of using the FT of the histograms is that the route travel time histogram is obtained as the inverse FT of the multiplication of the FT-ed link histograms instead to their convolution. For that the complexity of the on-line computation is of order $h \times m + g\left(k - m\right) + k \log\left(k\right)$, with the $k \log\left(k\right)$ term due to the inverse FT.

The difference between the two approaches is illustrated by a numerical expample. Suppose the route contains $h = g = 50$ links, the histogram contains $k = 100$ bins and that there are $m = 2500$ travel times observations for each link.

It is found that the second approach (order 6.10^3) outperforms the first approach (order 1.10^5). This is because in this test case $g\left(k - m\right) + k \log\left(k\right) < 0$. The performance of the second approach is also improved as $g < h$. Only if $\frac{k}{g}\left(g + \log\left(k\right)\right) \geq m$ the first approach has equal or better performance than the second approach. This is generally not the case since m must be sufficiently large to obtain reliable travel time statistics.

3   The clustering algorithm

To identify clusters in the network the following heuristic is proposed. The clustering algorithm restructures the network such that the travel time statistics of any 2 subsequent links in the resulting network are linearly independent. This is done by combining successive links based on the correlation of their travel time statistics. The clustering algorithm considers all links l_i in the network and calculates the correlation r with each successive link l_k (temporal propagation of travel time fluctuations are not considered explicitly). The correlation r is tested according to the hypothesis H0: $r = 0$ in a way that minimizes the probability of a Type II error. This means that $\Pr$(H0 is accepted $|r \neq 0$) is reduced to some specified value $0 \leq p \leq 1$.

Link l_i and l_k are combined into a cluster of links if H0 is rejected (ie. it is possible that $r \neq 0$). The statistics of this new clusterlink are obtained as the sum of the instantaneous travel times of both links for each t_j. Each such combination of links is added to the network and expanded until H0 cannot be rejected (ie. no links can be added to the cluster of links). Links that have become redundant due to the clustering are removed from the network. In this way the network is restructured to guarantee that the statistics of each 2 subsequent links in the network are linearly independent. The result of the clustering algorithm is illustrated in figure 1.
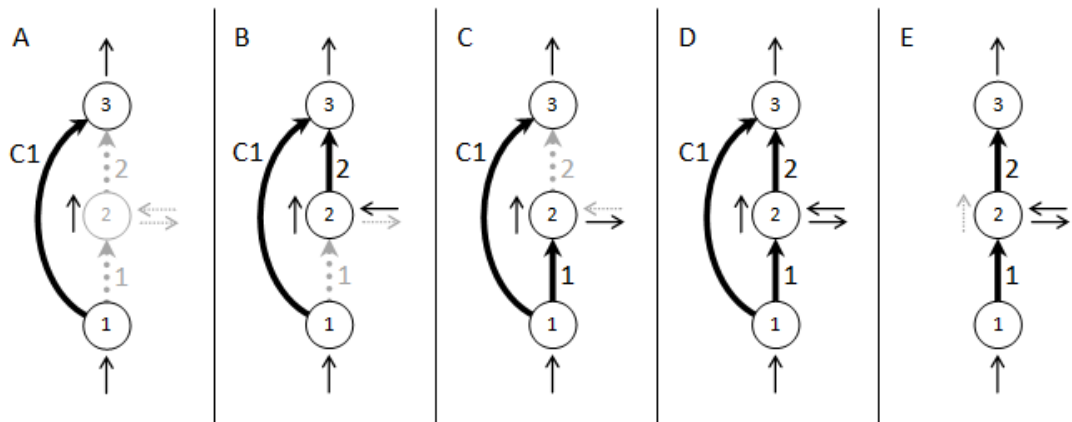
Figure 1: Fundamental clustering diagramme of 3 nodes and 2 correlated links. Depending on the properties of node 2, the clustering of links results in diagramme A, B, C, D or E with cluster C1 of links 1 and 2 . A: Traffic can only pass by node 2. B: Traffic can enter the network at node 2. C: Traffic can exit the network at node 2. D: Traffic can both enter/exit the network at node 2. E: Traffic can enter/exit the network at node 2 but cannot pass over it, links are not clustered.

4 Discussion

Currently the clutering method is tested on a the complete Belgian road network. The main question to be answered is whether the clustering of links significantly improves the quality of the histogram with respect to the histogram obtained without clustering. Also the expansion of the size of the network due to additon of the clusters of links to the network is being investigated. The first results look promising, both for the quality of the histograms as for the computational efficiency of the method. The results will be presented in the full paper.

5 Acknowledgements

Real Option Models for Network Design Under Uncertainty

Joseph Y.J. Chow

Department of Civil & Environmental Engineering

Institute of Transportation Studies

University of California, Irvine

Corresponding Author

Amelia C. Regan

Department of Computer Science

Institute of Transportation Studies

University of California, Irvine, 352 Computer Science, Irvine, CA, USA

Email: aregan@uci.edu

Extended Abstract

The goals of a transportation agency have shifted over the past few decades from systems optimization to dynamic strategic planning. Dynamic strategic planning focuses on two primary aspects: the need to account for uncertainties, and the need to account for decision-makers [1]. Many of the available analytical tools have evolved to reflect this trend, particularly network design models ([2], [3]). There is an abundant literature dealing with uncertainty in the many areas of network design, although most are based on stationary stochastic variables rather than time dependent ones. This is especially the case for bi-level problems in which static decisions are made at a single point in time ([4], [5], [6]). Ignoring the time dependent aspect of the problems limits the extent that flexibility (defined in this context as "the ability of a system to adapt to external changes, while maintaining satisfactory system performance" [7]) can be considered. However, the existing literature suggests that flexibility can be used to address network design models under uncertainty. It also reveals the complexity of such problems, especially the urban problems with a bi-level structure.

In corporate finance the concept of real options has grown significantly in the last few years as a tool for extending flexibility to projects under uncertainty. The value of options as a tool for hedging risk comes when uncertainty is introduced [8]. In real life strategic planning, a decision-maker does not make static decisions and follow through with them regardless of intermediate outcomes. Instead, they incorporate flexibility into their planning by using current information to

1

adjust their plans over time. Besides the different types of options that have been derived to model different management strategies for dealing with uncertainty [9], real option analysis has been applied to a number of transportation-related problems ([6], [10], [11]). However, no solution methodology based on real options has been proposed for multi-period urban network design problems with uncertainty. In a network setting, there are interdependencies in performance due to the interrelated stochastic flows that cannot be ignored.

The simplest approach to considering network design as a real option is to assume that the design solution is an investment and to compute a deferral option value with the link designs treated as exogenous variables. A standard Bellman equation for the option valuation and a network design problem are combined to obtain the hierarchical network investment deferral option (NIDO) model. The model maximizes the decision to defer or invest in a network design that has been solved for expected demand [12].

The formulation has a subtle implication on a network design over time. The NIDO model is shown to be a combination of the static NPV, the basic deferral premium, and the network design premium which includes the flexibility to redesign the network. In other words, the option value can be expressed as $\Phi = NPV + F_D + F_N$ for network designs, where F_D is the premium from deferral only and F_N is the premium from the flexibility of re-designing a network.

Two significant conclusions can be drawn from this network design premium. First, it is prudent to incorporate flexibility into the planning process for transportation agencies by not committing to "preferred alternatives" and to use "conditional alternatives" instead. Second, deferral options based on fixed designs can serve as lower bounds for flexible design options since the network design premium is non-negative. This is crucial for the following two new models, which can be too complex to solve as flexible designs but are feasible as fixed design options.

The existing numerical methods for solving real options can generally be categorized into three classes: finite difference, binomial lattice methods, or Monte Carlo simulation. Due to multi-dimensionality and computational cost requirements, the Least Squares Monte Carlo (LSM) method ([13], [14]) is chosen for solving the NIDO value. The method provides a pathwise approximation to the optimal stopping rule for maximizing the value of an American option and is the most computationally efficient in terms of the number of function evaluations.

A second model involves maximizing the option value with the network design as the set of decision variables. The network option design problem (NODP) maximizes the option value as a function of a committed network design. While the optimal design is for maximizing the fixed design option value, it can be interpreted as a lower bound to the value of the option under a flexible design setting.

As an option maximization model with fixed network design decision variables, the problem can be solved using global heuristics for network design problems. Fast converging global heuristics are necessary because of the computational cost of one evaluation function. A heuristic for solving the

NODP with continuous network design variables is demonstrated with a Metric Stochastic Response Surface (MSRS) method. MSRS is a global stochastic optimization approach that can use radial basis functions (RBF's) to intelligently guess the next point to evaluate using interpolation (MSRBF), and has been shown to work better than the genetic algorithm for network design problems with up to 31 dimensional variables for the Anaheim, CA network [15].

For fixed discrete network designs, real options can be incorporated in a third model, the Link Investment Deferral Option Set (LIDOS). If each link investment is considered a separate investment option, then the network design can be treated as a set of interacting options. By modeling the individual link or projects as separate options, the decision-maker has the flexibility to decouple their design investment with staging strategies. For example, a design composed of two projects would be evaluated as a bundle with NIDO, but with LIDOS there may be greater value from investing in project A and deferring project B. This is essentially a project selection problem under uncertainty.

As formulated, the LIDOS appears to be a backward dynamic program with forward elements, making it seem unsolvable. However, it is possible to solve the model by re-formulating it into one that can be solved by the multi-option LSM solution approach from [14]. The multi-option LSM method can only handle purely compound options where one option depends on whether the prior option is exercised. These dependencies imply that the LIDOS problem needs to be constrained so that decision-maker cannot change the order of investment in the future. The Ordered Link Investment Deferral Option Set (OLIDOS) is solved by enumerating each ordered staging of link investments and solving using mulit-option LSM, then choosing the ordered set that maximizes the option value. By doing so, it is possible to solve a lower bound of LIDOS with the multi-option LSM approach.

The numerical tests for all three models are conducted with the Sioux Falls test network. The first model is tested for sensitivity to demand volatility, time horizon, and standard error for a given number of simulation paths in the LSM algorithm. The second model is compared to the results of the fixed design solution for the first model to note the change in option value and design solution when optimizing the fixed design option directly as a function of the network design variables. The last model takes the results of the second model and obtains an optimal staging for each individual link design solution, and the option value is compared with the other two models. We will discuss the performance of each of these models and examine their potential for solving problems of more realistic size. A more detailed discussion of all of these models is presented in [16].

References

[1] de Neufville, R. (2000). Dynamic strategic planning for technology policy. *International Journal of Technology Management, 19 (3-5)*, 225-245.

[2] Magnanti, T.L. and R.T. Wong (1984). Network design and transportation planning: models and algorithms. *Transportation Science, 18 (1)*, 1-55.

[3] Yang, H. and M.G.H. Bell (1998). Models and algorithms for road network design: a review and some new developments. *Transport Reviews, 18 (3)*, 257-278.

[4] Chen, A. and C. Yang (2004). Stochastic transportation network design problem with spatial equity constraint. *Transportation Research Record: Journal of the Transportation Research Board, 1882*, 97-104.

[5] Snyder, L.V., M.S. Daskin, and C.P. Teo (2007). The stochastic location model with risk pooling. *European Journal of Operational Research, 179 (3),* 1221-1238.

[6] Damnjanovic, I., J. Duthie, and S.T. Waller (2008). Valuation of strategic network flexibility in development of toll road projects. *Construction Management and Economics, 26*, 979-990.

[7] Morlok, E.K. and D.J. Chang (2004). Measuring capacity flexibility of a transportation system. *Transportation Research Part A, 38 (6)*, 405-420.

[8] Trigeorgis, L. (1996). *Real Options: Managerial Flexibility and Strategy in Resource Allocation*. The MIT Press, Cambridge, MA, 427pp.

[9] Huchzermeier, A. and M.A. Cohen (1996). Valuing operational flexibility under exchange rate risk. *Operations Research, 44 (1)*, 100-113.

[10] Garvin, M.J. and C.Y.J. Cheah (2004). Valuation techniques for infrastructure investment decisions. *Construction Management and Economics, 22 (4)*, 373-383.

[11] Zhao, T., S.K. Sundararajan, and C.L. Tseng (2004). Highway Development Decision-Making under Uncertainty: A Real Options Approach. *Journal of Infrastructure Systems, 10 (1)*, 23-32.

[12] Chow, J.Y.J. and A.C. Regan (2010). Real option pricing of network design investments. *Transportation Science*, In Press.

[13] Longstaff, F.A., and E.S. Schwartz (2001). Valuing American options by simulation: a simple least-squares approach. *Review of Financial Studies, 14 (1)*, 113-147.

[14] Gamba, A. (2002). An Extension of Least Squares Monte Carlo Simulation for Multi-options Problems. *Proceedings of the 6th Annual International Real Options Conference*, Paphos, Cyprus, July 2002.

[15] Chow, J.Y.J, A.C. Regan, and D.I. Arkhipov (2010). A faster converging global heuristic for continuous network design using radial basis functions. *Transportation Research Record: Journal of the Transportation Research Board*, in press.

[16] Chow J.Y.J. (2010), Flexible Transportation Network Management Under Uncertainty, PhD Dissertation, University of California, Irvine.

Joint Problem of Traffic Signal Synchronization and Bus Priority

Chiara Colombaroni

Department Hydraulics, Transport and Roads

Sapienza University of Rome

Andrea Gemma

Department of Computer Science and Automation

University Roma Tre

Gaetano Fusco

Department Hydraulics, Transport and Roads Sapienza University of Rome

Via Eudossiana 18, Rome, Italy

Email: gaetano.fusco@uniroma1.it

Introduction

Several methods have been developed to allow bus priority with respect to general traffic in urban areas. Among these, signal priority strategies attempt to reduce delay in two ways: by reducing the probability of a transit vehicle encountering a red signal, and, if this does occur, by reducing the wait time until the green signal. The objective of this study is modeling and simulating a mathematical procedure to provide bus priority along a synchronization arterial, through the combination of passive and active bus priority strategies.

1 State of the art

Passive priority is defined as the use of static signal settings to reduce delay for transit vehicles. Such strategies can be as simple as allocating more green time to the street with the transit route by increasing the split for the phase in which the transit vehicle has right of way. Signal coordination is another strategy that can be used to benefit transit vehicles. Arterial progression, for example, can be designed to favor transit vehicles by timing the green band at the average transit vehicle speed instead of the average automobile speed, which is typically faster [1]. More effective coordination strategies

can combine the maximum green bandwidth and minimum delay problems. Several examples are available in the literature [2][3][4]. Another approach aims at optimizing an objective function that expresses the network performance. Different non convex optimization algorithms are applied to this goal. As an example, the well known Transyt program uses hill climbing and genetic algorithm [5]. However, it has been observed that passive strategies have limited value in order to improve the global transport performances [6].

Active strategies address these limitations of passive strategies by altering signal settings dynamically and only when necessary, making adjustments in real-time to the signal timing in order to minimize delay to an approaching transit vehicle. Several studies have been performed to apply active priority strategies: Liao and Davis [7] take advantage of the already equipped Global Positioning System on buses to develop an adaptive signal priority strategy that could consider bus schedule adherence, number of passengers, location and speed; Stevanovich et alii [8] present a genetic algorithm formulation that optimizes four basic signal timing parameters and transit priority settings using VISSIM microsimulation as the evaluation environment.

A bus priority algorithm could also be integrated into an adaptive network signal control model. For example, the SCOOT [9] system has a number of facilities that can be used to provide priority to buses or other public transport vehicles; the signal timings are optimized to benefit the buses, either by extending a current green signal (an extension) or causing succeeding stages to occur early (a recall). Priority facilities are also available in the UTOPIA [10] system, in which optimal strategies are determined at the higher level on the basis of area traffic prediction, whilst traffic light control is actuated at the local level according to traffic conditions at individual intersections. The aim of the control strategies is to minimize the total time lost by private vehicles, whilst ensuring that public transport vehicles are not stopped at intersections with traffic signals.

2 Methodology

The proposed model is a hybrid model of signal coordination optimization of a urban arterial and bus priority by the application of passive and active strategies. The arterial synchronization is set by optimizing an objective function that considers the road traffic and transit passengers. A simulative approach is followed to optimize pre timed signal settings (passive bus priority, applied as reference timing plan) and to take into account also different active bus priority strategies. It is so possible to assess time varying traffic signal performances during the simulation. Private cars are modeled as platoons that run along the arterial and may be delayed at nodes depending on their arrival time, the signal settings and the number of queued vehicles, if any.

Buses are modeled individually; the model computes the number of passengers, the dwell time and the vehicle position in queue, if any, for each bus at each bus stop. Buses arriving during the same signal cycle are moved as platoons along downstream link. The procedure is used to evaluate

new priority acknowledgement rules that consider several components like bus schedule adherence , number of the passengers on the bus, traffic flow on cross streets, green split, predicted headway between two following priority requests.

3 Optimization Algorithm

In this study an algorithm has been developed to optimize the signal synchronization by taking into account the delay of both the public and private traffic. The algorithm combines the search for a global minimum by a genetic algorithm and a local refinement procedure with predefined steps (similar to the hill-climbing used in Transyt) around the tentative solution point. The objective function (or fitness function) is defined as a linear combination of different components of total delays on the artery, calculated as follows.

$$ f = \left(1 - w_p\right)\left(\left(1 - w_t\right)\left(w_1 \sum_{i=1}^{n} D_i^{(1)} + \left(1 - w_1\right)\sum_{i=1}^{n} D_i^{(2)}\right) + w_t \sum_{i=1}^{m_i} D_{h_i}^{(t)}\right) + w_p\left(\sum_{i=1}^{b} D_b^{(p)}\right) $$

with:

- $D^{(1)}i$ the total delay at node i in direction 1
- $D^{(2)}i$ the total delay at node i in direction 2
- $D^{(t)}i$ the total delay at node i of queue h in lateral approach t
- $D^{(p)}b$ the total delay of passengers in bus b
- w_1 the weight of delay in direction 1
- w_t the weight of the delay at lateral approaches
- w_p the weight of delay for transit passengers

Genetic coding is composed by the signal variables at each intersection, that is: cycle length, green split rates in the direction 1 and 2, offsets.

The Genetic Algorithm (GA) uses the roulette wheel method to apply the well known genetic operators of crossover and mutation. The probability of mutation, in the absence of improvements in the objective function, varies linearly from γmin to γmax in a given number of iterations. This strategy is used to avoid a deadlock into a local minimum. The GA also uses the feature of elitism to keep a quota η of the solutions ordered according to their fitness.

The local adjustment algorithm applies a strategy similar to the Hill-Climbing method used by Transyt. It performs a series of trials sequentially to increase and then decrease the design variables: cycle length, green rate splits and offsets. The algorithm trials for each variable, 3 step lengths $s_1 > s_2 > s_3 > 0$ and apply them if the improvements are obtained. Execution ends when there are no more improvements.

5 Applications

The model is able to simulate:

- private traffic flow;

- flow of public transport on bus lanes with queuing or overtaking at the bus stops;

- active strategies with anticipation, extension and signal recovery of the green;

- passive strategy of synchronization;

- priority acknowledgement rules that depend on the timetable, the number of passengers and the saturation degree.

The procedure described here has been completed and applied to simulate the road traffic in two real cases in Rome. The software program to simulate the bus priority strategies is being finalized and in the final paper will be applied to a urban arterial in Rome, namely Via Tiburtina.

Reference

[1] Davol A.P. (2001), Modeling of Traffic Signal Control and Transit Signal Priority Strategies in a Microscopic Simulation Laboratory. Ph.D. Thesis, Massachusetts Institute of Technology.

[2] Cohen S.L. (1983), Concurrent Use of MAXBAND and TRANSYT Signal Timing Programs for Arterial Signal Optimization, Transpn. Res. Rec., 906, 81-84.

[3] Cohen S.L. e Liu C.C. (1986), The Bandwidth-Constrained TRANSYT Signal-Optimization Program, Transpn. Res. Rec., 1057, 1-7.

[4] Papola N., Fusco G. (1998b), Relationship between Travel Time Minimization and Bandwidth Maximization Problems, TRISTAN III Conference, San Juan, Puerto Rico, June 17-23, 1998.

[5] Hadi M.A., Wallace C.E. (1993), Hybrid Genetic Algorithm To Optimize Signal Phasing and Timing, Transpn. Res. Rec., 1421.

[6] Skabardonis, A. (2000), Control Strategies for Transit Priority, Presented at the 79th Annual Meeting of the Transportation Research Board.

[7] Liao C.F., Davis G.A. (2007), Simulation Study of Bus Signal Priority Strategy, Taking Advantage of Global Positioning System, Automated Vehicle Location System, and Wireless Communications, Journal of Transportation Research Board, N°2034, pp. 82-91

[8] Stevanovic J, Stevanovic A., Martin P.T, Bauer T(2008), Stochastic Optimization of Traffic Control and Transit Priority Settings in VISSIM, Transportation Research Part C 16 (2008) 332–349

[9] McDonald M., Hounsell N.B.(1991), Road Traffic Control: Transyt and SCOOT, Concise Encyclopedia of Traffic & Transportation Systems

[10] Mauro V. (1991), Road Network Control, Concise Encyclopedia of Traffic & Transportation Systems

Benders Decomposition for Large-Scale Uncapacitated Hub Location Problems

Ivan Contreras

Canada Research Chair in Distribution Management and CIRRELT,

HEC Montréal, Canada

Jean-François Cordeau

Canada Research Chair in Logistics and Transportation and CIRRELT,

HEC Montréal, Canada

Gilbert Laporte

Canada Research Chair in Distribution Management and CIRRELT,

HEC Montréal, Canada

Email: ivan.contreras@cirrelt.ca

1 Introduction

Hub Location Problems (HLPs) lie at the heart of network design in transportation systems, especially in the airline and trucking industries. The performance of these systems can be improved by using transshipment points (hubs), where the flows between O/D pairs are consolidated and rerouted to their destinations, sometimes via another hub. Thus, the locations of the hubs as well as the paths for sending the flows between the O/D pairs have to be determined. HLPs consist of locating hubs on a network so as to minimize the total transportation cost.

Over the last decades, several variants of HLPs have been studied (see [1] for a recent survey). Given the inherent difficulty of HLPs, only small to medium-size instances (10-50 nodes) are normally solved to optimality and approximate procedures need to be used to approach larger size instances. In fact, it is only very recently that instances with up to 200 nodes could be solved optimally (see [2, 3]). In this paper, we propose a Benders decomposition algorithm specifically designed to approach large-scale instances of the classical *Uncapacitated Hub Location Problem with Multiple Assignment* (UHLPMA). Moreover, we introduce a new challenging set of benchmark instances ranging from 10 to 400 nodes to test the proposed methodology. Computational experiments assess the efficiency of the algorithm. On the one hand, it is able to speed-up by at

least one order of magnitude the current best algorithm given in [2]. On the other hand, it is able to solve optimally instances with up to 400 nodes within reasonable computational times.

The paper is organized as follows. Section 2 describes the problem and presents a MIP formulation. The basic Benders decomposition is presented in Section 3. Section 4, introduces several algorithmic features that improve the convergence and efficiency of the algorithm.

2 Problem Definition

Let H be a set of potential hub locations and K be the set of commodities. Let W_k denote the amount of commodity k. For each node $i \in H$, f_i denotes the fixed installation cost for locating a hub at node i. Let $E = \{L \subseteq H : 1 \leq |L| \leq 2\}$ be a set of subsets of H containing one or two hubs. The undirected transportation cost for each $e \in E$ and $k \in K$ is denoted as F_{ek}. The UHLPMA consists of selecting a set of hubs to be established and the routing of flow through the network, with the objective of minimizing installation and transportation costs. We define location variables z_i, $i \in H$, that are equal to 1 if a hub is located at node i, and 0 otherwise. We also define the routing variables x_{ek}, $k \in K$ and $e \in E$, that are equal to 1 if commodity k goes via hub edge e, and 0 otherwise. The UHLPMA can be stated as (see [4]),

$$\text{minimize} \quad \sum_{i \in H} f_i z_i + \sum_{e \in E} \sum_{k \in K} F_{ek} x_{ek}$$

$$\text{subject to} \quad \sum_{e \in E} x_{ek} = 1 \qquad \forall\, k \in K \tag{1}$$

$$\sum_{\{e \in E \,:\, i \in e\}} x_{ek} \leq z_i \qquad \forall\, i \in H, \forall\, k \in K \tag{2}$$

$$z_i \in \{0,1\} \qquad \forall\, i \in H \tag{3}$$

$$x_{ek} \geq 0 \qquad \forall\, e \in E, \forall\, k \in K. \tag{4}$$

Constraints (1) guarantee that for each commodity there is a single path connecting its origin and destination nodes. Constraints (2) prohibit commodities from being routed via a node that is not a hub. Finally, constraints (3) and (4) are the classical integrality and non-negativity constraints.

3 Benders Decomposition

At iteration t of the Benders decomposition algorithm, if we fix the integer variables $z = z^t$ we obtain the following primal linear subproblem,

$$(SP_P) \quad \text{minimize} \quad s^t + \sum_{e \in E} \sum_{k \in K} F_{ek} x_{ek}$$

$$\text{subject to} \quad (1), (4),$$

$$\sum_{\{e \in E \,:\, i \in e\}} x_{ek} \leq z_i^t \qquad \forall\, i \in H, \forall\, k \in K, \tag{5}$$

where $s^t = \sum_{i \in H} f_i z_i^t$ is the installation cost associated to solution z^t. Let α_k and u_{ik} be the dual variables associated to constraints (1) and (5), respectively. Then, the dual problem of SP_P at iteration t can be stated as follows:

$$(SP_D) \quad \text{maximize} \quad \sum_{k \in K} \alpha_k - \sum_{i \in H} \sum_{k \in K} z_i^t u_{ik} \tag{6}$$

$$\text{subject to} \quad \alpha_k - u_{e_1 k} - u_{e_2 k} \leq F_{ek} \quad \forall k \in K, \forall e \in E, |e| = 2 \tag{7}$$

$$\alpha_k - u_{e_1 k} \leq F_{ek} \qquad \forall k \in K, \forall e \in E, |e| = 1. \tag{8}$$

For a given iteration t, from the dual objective function (6) we obtain the optimality cut $\eta \geq \sum_{k \in K} \alpha_k^t - \sum_{i \in H}(\sum_{k \in K} u_{ik}^t)z_i$, where α_k^t and u_{ik}^t are the optimal value of the dual variables obtained by solving SP_D at iteration t, and η is a continuous variable for the estimation of the overall transportation cost. We thus can formulate the following master problem:

$$(MP) \quad \text{minimize} \quad \sum_{i \in H} f_i z_i + \eta$$

$$\text{subject to} \quad \eta \geq \sum_{k \in K} \alpha_k^t - \sum_{i \in H}(\sum_{k \in K} u_{ik}^t)z_i \qquad t = 1, \ldots, T \tag{9}$$

$$\sum_{i \in H} z_i \geq 1 \tag{10}$$

$$z_i \in \{0, 1\} \qquad \forall\, i \in H, \tag{11}$$

where T is the current number of iterations. Constraint (10) ensures that at least one hub facility is open in the optimal solution of MP and thus avoids the generation of feasibility cuts.

4 Algorithmic Refinements

4.1 Multiple Benders cuts

One way to improve the convergence of the Benders algorithm is to exploit the decomposable structure of the subproblem to generate several feasibility cuts at each iteration. Instead of adding a single cut per iteration, we can rather separate the information obtained by the SP to generate a set of feasibility cuts associated to subsets of commodities at each iteration. Let $K_i \subset K$ be the subset of commodities whose origin node is i. We thus can generate the following set of cuts:

$$\eta_i \geq \sum_{k \in K_i} \alpha_k^t - \sum_{i \in H}(\sum_{k \in K_i} u_{ik}^t)z_i \qquad \forall i \in H. \tag{12}$$

4.2 Pareto-optimal Cuts

Another way to improve the convergence of the Benders algorithm is by constructing stronger, undominated cuts. These non-dominated cuts are known as *Pareto-optimal* cuts (see [5]). Let Q be the polyhedron defined by (10), impose $0 \leq z_i \leq 1$ for all $i \in H$, and let $ri(Q)$ denote the

relative interior of Q. To identify a dual optimal solution to SP_D that yields a Pareto-optimal cut, we must solve the following problem:

$$(PO) \quad \text{maximize} \quad \sum_{k \in K} \alpha_k - \sum_{i \in H} \sum_{k \in K} z_i^0 u_{ik} \tag{13}$$

$$\text{subject to} \quad (7)-(8),$$

$$\alpha_k - \sum_{i \in H} z_i^t u_{ik} = \hat{\alpha}_k \quad \forall k \in K, \tag{14}$$

where $z^0 \in ri(Q)$ and $\hat{\alpha}_k$ is the optimal solution value of the k subproblem. Constraints (14) ensure that a dual optimal solution from the set of optimal solutions of SP_D is selected.

4.3 Generating Initial Cuts for the Master Problem

Although the Benders decomposition algorithm can be initialized from an empty set of optimality cuts, the choice of this initial set could greatly affect its convergence. Given that optimality cuts are obtained from feasible solutions, we can use some heuristic procedure to obtain a promising initial set of optimality cuts to improve the overall convergence of the Benders decomposition.

4.4 Elimination Tests

Another way to improve the efficiency of the Benders decomposition algorithm is by reducing the size of the model. This can be done by developing some reduction tests capable of eliminating several variables and constraints and thus, making the solution of both the master problem and the subproblem more efficient. These tests use the lower bounds obtained at the inner iterations of the algorithm to close some hub nodes that do not appear in an optimal solution.

References

[1] S. Alumur, B.Y. Kara, "Network hub location problems: The state of the art", *European Journal of Operational Research*, 190, 1–21 (2008).

[2] R.S. Camargo, G. Miranda Jr., H.P. Luna, "Benders decomposition for the uncapacitated multiple allocation hub location problem", *Commputers & Operations Research*, 35, 1047–1064 (2008).

[3] I. Contreras, J.A. Díaz, E. Fernández, "Branch-and-price for large-scale capacitated hub location problems with single assignment", *INFORMS Journal on Computing*, forthcoming (2010).

[4] H.W. Hamacher, M. Labbé, S. Nickel, T. Sonneborn, "Adapting polyhedral properties from facility to hub location problems", *Discrete Applied Mathematics*, 145, 104-116 (2004).

[5] T.L. Magnanti, R.T. Wong, "Accelerating Benders decomposition: Algorithmic enhancement and model selection criteria", *Operations Research*, 29, 464–484 (1981).

The windy clustered prize-collecting problem

Ángel Corberán

Department of Statistics and Operations Research

University of Valencia

Elena Fernández

Department of Statistics and Operations Research

Technical University of Catalonia, Jordi Girona 1-3, Barcelona, Spain

Email: e.fernandez@upc.edu

Carles Franquesa

Department of Statistics and Operations Research

Technical University of Catalonia

José M. Sanchis

Department of Applied Mathematics

Technical University of Valencia

1   Introduction

Routing Problems look for routes that serve demand customers at minimum cost. In Arc Routing Problems (ARPs) customers with demand are represented by a subset of edges or arcs of a given graph and it is usually assumed that all demand customers have to be served. Few problems consider the case where a profit is associated with each demand edge, and a decision must be made to jointly determine a subset of demand edges to be served and a route to serve them. Some of these problems have been studied in [8] and [4]. In Prize-collecting Arc Routing Problems (PARPs) it is assumed that the profit of each serviced demand edge is collected once, independently of the number of times it is traversed. PARPs were introduced in [3] and an algorithm to solve the Prize-collecting Rural Postman Problem was presented in [2]. A further PARP which has been recently studied in [9] and [1] is the Clustered Prize-collecting Arc Routing Problem (CPARP). In the CPARP the connected components defined by demand edges are considered, and it is required that if a demand edge is serviced, then all the demand edges of its component are also serviced. That is, for each component either all or none of its demand edges are serviced.

Many ARPs have been studied on windy graphs. In a windy graph there are two non-negative values associated with each edge, representing the costs of traversing the edge in each direction. Windy ARPs constitute an important class of problems, as the windy version of an ARP is a generalization of its undirected, directed and mixed versions. A global overview of the Windy General Routing Problem which contains as particular cases most of the studied windy ARPs with uncapacitated vehicles is given in [6].

To the best of our knowledge no arc routing problem with profits has been studied on a windy graph. In this work we present the Windy Clustered Prize-collecting Arc Routing Problem (WC-PARP). First, we describe the problem, and give a mathematical programming formulation. Then, we present some polyhedral results, including some families of valid and facet defining inequalities.

2 The problem

Let $G = (V, E)$ be a connected undirected graph with a distinguished vertex $v_d \in V$, the depot, and let $D \subset E$ denote the subset of edges with demand. We assume G has been simplified as in [5, 7], so that V is the set of vertices incident with edges in D plus the depot, and E contains the edges in D plus some other representing shortest paths in the original graph. The connected components of $G_D = (V, D)$ are denoted $C_k = (V_k, D_k)$ $(k \in K)$ and referred to as *clusters*. We assume $v_d \in V_1$. Let b be a non-negative profit function on D. Associated with each edge $(i, j) \in E$ there are two non-negative costs, c_{ij} and c_{ji}, representing the cost of traversing it from i to j and from j to i, respectively.

Feasible solutions to the WCPARP are tours going through v_d such that for each C_k $(k \in K)$ either all its edges are serviced or none of its edges is serviced. The cost of a tour $\mathcal{T}$ is $\sum_{(i,j) \in \mathcal{T}} (t_{ij} c_{ij} + t_{ji} c_{ji})$, where t_{ij} (resp. t_{ji}) is the number of times edge $(i, j) \in E$ is traversed in $\mathcal{T}$ in the direction from i to j (resp. j to i). Each serviced edge $e \in E$ produces a profit b_e, which does not depend on the direction of its traversal and is collected once, independently of the number of times e is traversed. Thus, the gross profit of each serviced cluster is $p_k = \sum_{e \in D_k} b_e$. The WCPARP is to find a set of clusters $\mathcal{K}^* \subseteq K$, and a tour $\mathcal{T}^*$, passing through v_d, that serves all the edges in $\bigcup_{k \in \mathcal{K}^*} D_k$, but none of the edges in $D \setminus \bigcup_{k \in \mathcal{K}^*} D_k$, of maximum net profit

$$\sum_{k \in \mathcal{K}^*} p_k - \sum_{(i,j) \in \mathcal{T}^*} (t_{ij} c_{ij} + t_{ji} c_{ji}).$$

The WCPARP is NP-hard as its undirected version CPARP is already NP-hard [1].

To formulate the WCPARP, for each edge $e = (i, j) \in E$ we define two decision variables x_{ij}, x_{ji} representing the number of times edge e is traversed from i to j and from j to i, respectively. In addition, we have $|K|$ binary variables z_k $(k \in K)$, that take value one iff component k is serviced. Then, the WCPARP can be formulated as follows:

$$\max \quad \sum_{k \in K} p_k z_k - \sum_{e \in E} (c_{ij} x_{ij} + c_{ji} x_{ji}) \tag{1}$$

$$x_{ij} + x_{ji} \geq z_k \qquad (i,j) \in D_k, \ k \in K \tag{2}$$

$$\sum_{(i,j) \in \delta(i)} (x_{ij} - x_{ji}) = 0 \qquad i \in V \tag{3}$$

$$x(\delta(S)) \geq 2z_k \qquad S \subseteq V \setminus \{v_d\}, k \in K \text{ s.t. } V_k \subseteq S \tag{4}$$

$$x_{ij}, x_{ji} \geq 0 \text{ and integer} \qquad (i,j) \in E \tag{5}$$

$$z_k \in \{0,1\} \qquad k \in K \tag{6}$$

Inequalities (2) force the route to traverse all the demand edges of the clusters it serves, equations (3) force the route to be symmetric, whereas inequalities (4) ensure that the route connect the clusters it serves and the depot.

3 Polyhedral results

Next we present some polyhedral results for the polyhedron defined by the feasible solutions to (2)-(6). Due to space limitations we give the results without a formal proof.

Theorem 3.1 *Consider the polyhedron $P = conv\{(x,z) \in \mathbf{Z}^{2|E|+|K|} | (x,z) \text{ satisfies } (2) - (6)\}$.*

- *If G is connected, $dim(P) = 2|E| + |K| - |V| + 1$.*

- *The following inequalities define facets of P:*

  [F1] *Inequalities $x_{ij} \geq 0$, $x_{ji} \geq 0$ and $x_{ij} + x_{ji} \geq z_k$, if $e = (i,j) \in E$ is not a bridge of G, where $k \in K$ is such that $e \in D_k$.*

  [F2] *Inequalities $z_k \geq 0$ and $z_k \leq 1$, for all $k \in K$.*

  [F3] *The connectivity inequalities $x(\delta(S)) \geq 2z_k + \sum_{r \in K_O}(|F_r| - 1)z_r + \sum_{r \in K_e}(|F_r| - 2)z_r$, where $S \subset V \setminus \{v_d\}$, $k \in K$ s.t. $V_k \subseteq S$, $F_r = \delta(S) \cap D_r$ $(r \in K)$, $K_O = \{r \in K : |F_r| \text{ odd}\}$, and $K_e = \{r \in K : |F_r| \geq 2 \text{ and even}\}$.*

  [F4] *The inequalities $x(\delta(S)) \geq 1 - r + \sum_{k \in K} |F_k|z_k + \sum_{k \in K_O} z_k$, where $S \subset V \setminus \{v_d\}$, $F \subseteq \delta(S)$, $|F|$ odd, such that $F = \bigcup_{k \in K} F_k$ with $F_k \subseteq D_k$ $(k \in K)$, $K_O \subseteq K$ the subset of indices such that $|F_k|$ is odd, and $r = |K_O|$.*

- *The following inequalities are valid for P:*

  [V1] *The inequalities $x(\delta(S)) \geq 1 - |F| + 2 \sum_{k \in K} |F_k|z_k$, where $S \subset V$, $F \subseteq \delta(S)$, $|F|$ odd, such that $F = \bigcup_{k \in K} F_k$, with $F_k \subseteq D_k$ $(k \in K)$.*

  [V2] *The K-C inequalities (see [6]), which can be adapted to the WCPARP.*

Some of the above families of inequalities have a size which is exponential on $|V|$, namely inequalities F3, F4, V1 and V2. Facets F3 have as particular case those connectivity inequalities (4) with $S = \bigcup_{k \in Q} V_k$, $Q \subseteq K \setminus \{1\}$ and $k \in Q$. These can be separated exactly in polynomial time although, for a higher computational efficiency, we use heuristic separation first. A heuristic is also used for the general case. Inequalities V1 can be separated exactly in polynomial time, although they are dominated by facets F4 whose exact separation requires the solution of a linear integer problem. We use the exact separation algorithm for inequalities V1 as a heuristic for the separation of inequalities F4. As usual, K-C inequalities are separated by means of an ad-hoc heuristic.

Preliminary computational experiments on a set of 118 benchmark instances of various sizes produce satisfactory numerical results. These results indicate that approximately 75% of the instances can be solved optimally making use of an iterative Linear Programming (LP) based algorithm in which violated inequalities are separated and incorporated to the initial formulation.

References

[1] J. Aráoz, E. Fernández and C. Franquesa, "The Clustered Price-collecting Arc-routing Problem", *Transportation Science* 43, 287–300 (2009).

[2] J. Aráoz, E. Fernández and O. Meza, "An LP-based Algorithm for the Privatized Rural Postman Problem", *European Journal of Operational Research* 196, 886–896 (2008).

[3] J. Aráoz, E. Fernández and C. Zoltan, "The Privatized Rural Postman Problem", *Computers & Opererations Research* 33, 3432–3449 (2006).

[4] C. Archetti, D. Feillet, A. Hertz and M.G. Speranza "The Undirected Capacitated Arc Routing Problem with Profits", *Computers & Operations Research* doi:10.1016/j.cor.2009.05.005.

[5] N. Christofides, V. Campos, A. Corberán, and E. Mota, "An Algorithm for the Rural Postman Problem on a Directed Graph" *Mathematical Programming Study* 26, 155–166 (1986).

[6] A. Corberán, J.M. Sanchis and I. Plana, The Windy General Routing Polyhedron: A Global View of Many Known Arc Routing Polyhedra, *SIAM Journal on Discrete Mathematics* 22, 606–628 (2008).

[7] H.A. Eiselt, M. Gendreau, and G. Laporte, "Arc-routing problems, Part 2: The Rural Postman Problem", *Operations Research* 43, 399–414 (1995).

[8] D. Feillet, P. Dejax and M. Gendreau, "The Profitable Arc Tour Problem: Solution with a Branch-and-Price Algorithm", *Transportation Science* 39, 539-52 (2005).

[9] C. Franquesa, The Clustered Prize-collecting Arc Routing Problem, PhD Thesis, Technical University of Catalonia, Barcelona (2008).

A new approach to the Maximum Benefit Chinese Postman Problem

Isaac Plana
Departamento Matemáticas para la Economía y la Empresa
Universidad de Valencia

Antonio M. Rodríguez-Chía
Departamento de Estadística e Investigación Operativa
Universidad de Cádiz

José M. Sanchis
Departamento de Matemática Aplicada
Universidad Politécnica de Valencia

Ángel Corberán
Departmento de Estadística e Investigación Operativa
Universidad de Valencia, Avenida Doctor Moliner 50, Burjassot (Valencia), Spain
Email: angel.corberan@uv.es

1 Introduction

The Maximum Benefit Chinese Postman Problem (MBCPP) is a generalization of the CPP in which not all the edges have to be traversed, and, associated with each edge of the graph, a cost for its traversal with service, a deadhead cost for its traversal with no service and a set of benefits are considered. A benefit is derived from every traversal with service of an edge. The objective is to find a closed walk (tour) starting and ending at the depot with maximum net benefit. Applications of the MBCPP include the routing of street cleaners and the construction of street snow-plowing and snow-salting tours. An additional benefit is derived when a street is plowed multiple times and the benefit may depend upon whether the link represents an arterial or a low-traffic neighborhood street. Unlike the classical CPP, this problem allows us to obtain solutions that do not traverse some edges while other edges can be traversed multiple times.

More precisely, the Maximum Benefit Chinese Postman Problem can be defined as follows. Let $G = (V, E)$ be an undirected connected graph, where vertex $1 \in V$ represents the depot. Each edge $e \in E$ has two types of costs associated: c_e^s and c_e^d, which we expect $c_e^s \geq c_e^d$. The first one represents the cost of traversing and servicing at the same time edge e, while the second one corresponds to the cost of just traversing without servicing that edge (deadhead cost). Moreover, each edge $e \in E$ has n_e benefits, $b_e^1 \geq b_e^2 \geq \cdots \geq b_e^{n_e} > 0$, giving the gross benefit of servicing the edge for the first, second,..., n_e-th time. Therefore, the net benefit of traversing and servicing edge e for the t-th time is given by $b_e^t - c_e^s$ for $t = 1, \ldots, n_e$, while the net benefit of deadheading an edge is $-c_e^d$. Each traversal of an edge incident with nodes i

and j can be represented by a different parallel edge between nodes i and j, and an additional parallel edge is included representing deadheading. Then, the MBCPP consists of finding a tour, starting from the depot, traversing a subset of edges in E and returning to the same depot with the maximum total net benefit. This problem is NP-Hard because it contains the Rural Postman Problem as a special case.

In the literature of routing problems we can find several attempts at studying this problem. Malandraki and Daskin [8] introduce the MBCPP defined on directed graphs. They show that this problem can be formulated as a minimum cost flow problem together with subtour elimination constraints. Based on this formulation, a branch-and-bound procedure is developed to solve the problem. On undirected graphs, Pearn and Wang [10] present a heuristic algorithm to solve the MBCPP approximately. Other heuristics are proposed in [9]. Up to our knowledge no other result on the MBCPP has been published.

However, some special cases have also been subject of study. For instance, the Prize-Collecting Arc Routing Problem (also called the Privatized Rural Postman Problem) in which, as in other prize-collecting routing problems, it is assumed that the benefit of each serviced edge is collected once, independently of the number of times it is traversed. This problem is introduced in Aráoz et al. [3], where a 0-1 formulation with an exponential number of inequalities is provided. In [2], an iterative algorithm to solve the problem defined on undirected graphs is proposed. A related problem, the Clustered Prize-collecting Arc Routing Problem, has been recently studied by Franquesa [7] and Aráoz et al. [1]. In this last problem, the connected components defined by the edges with net benefit (demand edges) are considered, and for each component either all or none of its demand edges have to be serviced. The same problem defined on a 'windy' graph is studied in [5]. Other arc routing problems with benefits are studied in [6] and [4]. In the first paper, the objective is to find a set of cycles in the graph that maximizes the net benefit subject to constraints limiting the number of times that the benefit is available on arcs and the maximal length of the cycles. To solve this problem a branch-and-price algorithm is proposed. In the second one, authors propose a branch-and-price algorithm and several heuristics for the Capacitated Arc Routing Problem with benefits.

2   Problem formulation

In the MBCPP each edge $e \in E$ has $n_e + 1$ net benefits associated, $\bar{b}_e^t = b_e^t - c_e^s$, corresponding to the successive traversals servicing the edge, $t = 1, \ldots, n_e$, and $\bar{b}_e^{n_e+1} = -c_e^d$ associated with deadheading e. Since we are assuming $c_e^s \geq c_e^d$ and $b_e^1 \geq b_e^2 \geq \cdots \geq b_e^{n_e} > 0$, we have $\bar{b}_e^1 \geq \bar{b}_e^2 \geq \cdots \geq \bar{b}_e^{n_e}$.

Let $M = \max\{n_e + 1, e \in E\}$. For each edge e, we can assume without loss of generality that there are M copies of this edge, each one with associated net benefit $\bar{b}_e^t, t = 1, \ldots, M$. If, for a given edge, $f \in E$, $n_f + 1 < M$, the $M - n_f - 1$ extra copies would have assigned net benefit $\bar{b}_f^{n_f+1}$. In this way, we can consider we are working on a graph with M edges in parallel associated with each edge in the original graph.

We now define

$$
b_e^{odd} \;=\; \max\left\{ \sum_{\ell=1}^{k} \overline{b}_e^{\ell} \; : \; k \text{ odd with } k \le M \right\}
$$

$$
b_e^{even} \;=\; \max\left\{ \sum_{\ell=1}^{k} \overline{b}_e^{\ell} \; : \; k \text{ even with } k \le M \right\} - b_e^{odd}.
$$

It is not difficult to see that solving the MBCPP on the graph with M parallel edges for each original edge is equivalent to work on a smaller graph having only two parallel edges for each edge in the original graph. Or, equivalently, to work on the original graph G, in which the first traversal of edge e has net benefit b_e^{odd}, while b_e^{even} is the benefit associated with its second traversal. Note that to get the net benefit b_e^{even} traversing edge e, we need first to traverse it with net benefit b_e^{odd}. In this way, the MBCPP can be formulated as follows.

For each $e = (i,j) \in E$ we define two binary variables x_e and y_e. Variable x_e takes value 1 if e is traversed and 0 if e is not traversed, while variable y_e takes value 1 if e is traversed twice and 0 otherwise. In other words, variables x_e and y_e represent the first and second traversal of edge e, respectively. We have the following formulation for the MBCPP:

$$
\text{Maximize} \qquad \sum_{e \in E} \left(b_e^{odd} x_e + b_e^{even} y_e \right)
$$

s.t.:

$$
\sum_{e \in \delta(i)} \left(x_e + y_e \right) \equiv 0 \qquad (\mathrm{mod}\ 2), \quad \forall i \in V \tag{1}
$$

$$
\sum_{e \in \delta(S)} \left(x_e + y_e \right) \ge 2 x_f, \qquad \forall S \subset V \setminus \{1\}, \quad \forall f \in E(S) \tag{2}
$$

$$
x_e \ge y_e \qquad \forall e \in E \tag{3}
$$

$$
x_e, y_e \in \{0,1\} \qquad \forall e \in E \tag{4}
$$

Constraints (1) force the vertices to be of even degree in the solution, its connectivity is assured with conditions (2), and constraints (3) guarantee that a second traversal of an edge can occur only if it has previously traversed.

3 MBCPP Polyhedron

Let us call MBCPP tour to each vector $(x,y) \in \{0,1\}^{2|E|}$ satisfying (1) to (4) and let MBCPP(G) be the convex hull of all MBCPP tours. It is obviously a polytope. We have proved that MBCPP(G) is a full-dimensional polyhedron if, and only if, G is 3-edge connected. Hence, in the following, we will assume that graph G is 3-edge connected.

In what refers to the facial description of the MBCPP polyhedron, we have proved that, under several conditions, the following sets of inequalities define facets of MBCPP(G):

- Inequalities $y_{uv} \ge 0$, for each edge $(u,v) \in E$.

- Inequalities $x_{uv} \leq 1$, for each edge $(u,v) \in E$.

- Inequalities $x_{uv} \geq y_{uv}$ for each edge $(u,v) \in E$.

- Connectivity inequalities $\sum_{e \in \delta(S)} \left(x_e + y_e \right) \geq 2x_f$ for each set $S \subset V \setminus \{1\}$ and for each edge $f \in E(S)$.

Conditions (1) are not linear inequalities. In order to force the solution to satisfy these parity conditions, we can use the following linear inequalities:

$$x(\delta(S) \setminus F) - y(\delta(S) \setminus F) \geq x(F) - y(F) - |F| + 1, \quad \forall S \subset V, \quad \forall F \subset \delta(S) \text{ with } |F| \text{ odd}. \quad (5)$$

Parity inequalities above are valid and facet-inducing for MBCPP(G).

At this moment, we are working on new types of facet inducing inequalities for the MBCPP, and in the design and implementation of a branch-and-cut algorithm for its exact resolution.

References

[1] J. Aráoz, E. Fernández and C. Franquesa, "The Clustered Price-collecting Arc-routing Problem", *Transportation Science* 43, 287-300 (2009).

[2] J. Aráoz and E. Fernández and O. Meza, "Solving the Prize-Collecting Rural Postman Problem", *European Journal of Operational Research* 196 (3), 886-896 (2009).

[3] J. Aráoz, E. Fernández and C. Zoltan, "Privatized Rural Postman Problems", *Computers and Operations Research* 33 (12), 3432-3449 (2006).

[4] C. Archetti, D. Feillet, A. Hertz and M.G. Speranza, "The Undirected Capacitated Arc Routing Problem with Profits", *Computers and Operations Research*, DOI:10.1016/j.cor.2009.05.005 (2009).

[5] A. Corberán, E. Fernández, C. Franquesa and José M. Sanchis, "The windy clustered prize-collecting problem". In preparation. (2009).

[6] D. Feillet, P. Dejax and M. Gendreau, "The Profitable Arc Tour Problem: Solution with a Branch-and-Price Algorithm", *Transportation Science* 39, 539-52 (2005).

[7] C. Franquesa, The Clustered Prize-collecting Arc Routing Problem, Ph.D. thesis, Technical University of Catalonia, Barcelona (2008).

[8] C. Malandraki and M.S. Daskin, "The Maximum Benefit Chinese Postman Problem and the Maximum Benefit Traveling Salesman Problem", *European Journal of Operational Research*, 65 (2), 218-234 (1993).

[9] W. L. Pearn and W. C. Chiu, "Approximate Solutions for the Maximum Benefit Chinese Postman Problem", *International Journal of Systems Science* 36 (13), 815-822 (2005).

[10] W.L. Pearn and K.H. Wang, "On the Maximum Benefit Chinese Postman Problem", *OMEGA - The International Journal of Management Science* 31, 269-273 (2003).

Bi-objective conflict detection and resolution in railway traffic management

Francesco Corman

Department of Transport and Planning, Delft University of Technology

Andrea D'Ariano

Dipartimento di Informatica e Automazione, Università degli Studi Roma Tre

Dario Pacciarelli

Dipartimento di Informatica e Automazione, Università degli Studi Roma Tre

via della vasca navale 79, Roma, Italy - Email: pacciarelli@dia.uniroma3.it

Marco Pranzo

Dipartimento di Ingegneria dell'Informazione, Università degli Studi di Siena

Abstract

Railway timetables define routes, orders and timings for all trains running in the network. Usually, timetables provide good connectivity between different train services for a number of origins and destinations. For each pair of connected train services, the waiting train is scheduled to depart sufficiently later with respect to its feeder train in order to allow the movement of passengers from one train to the other.

During operations, train traffic can be seriously disturbed by delays, accidents or technical problems. Major disturbances cause primary delays that propagate as consecutive delays to other trains in the network, thus requiring short-term adjustments to the timetable in order to limit delay propagation. This real-time problem is known as Conflict Detection and Resolution (CDR).

Keeping transfer connections when solving the CDR problem increases delay propagation [5], therefore one of the possible dispatching countermeasures to handle disturbances is the cancellation of some scheduled connections. This action reduces overall train delays but has a negative impact on passenger satisfaction for the passengers affected by the missed connection. Train operating companies are therefore interested in keeping as many connections as possible even in the presence of disturbed traffic conditions, while infrastructure managers are mainly interested in limiting train delays. In fact, infrastructure managers discuss with train operating companies on which

connections must be kept when regulating railway traffic. To support this negotiation process, in this paper we deal with a *Bi-objective Conflict Detection and Resolution* (BCDR) problem, i.e., the problem of finding a set of feasible schedules with a good trade-off between the minimization of train delays and the maximization of respected transfer connections.

The BCDR problem is closely related to the Delay Management (DM) problem introduced by Schöbel [4]. The latter problem adopts a passenger point of view, and aims at the minimization of the sum of all delays over all passengers at their final destination. In this paper we choose a train point of view. A value is associated to each connection (e.g., expressed in terms of the number of passengers who get the connection) and one objective function is the maximization of the total value of respected connections. A further difference is that the DM problem does not take into account the limited capacity of the railway network (i.e., does not deal with the CDR problem), which is the main issue of this paper.

We model the BCDR problem as a special bi-objective job shop scheduling problem, which can be formulated with an alternative graph [2]. A node $i \in N$ of the alternative graph is associated to the starting time t_i of the i-th relevant event. The set F of *fixed arcs* is used to represent precedences between events. The set A of *alternative arcs* is used to represent sequencing decisions. A further set C of *connection arcs* represents connections enforcement.

$$
\begin{aligned}
\min \quad & \left(t_n \; ; \; -\sum_{(i,j)\in C} v_{ij}\delta(t_j - t_i - w_{ij}) \right) \\
s.t. \quad & t_j - t_i \geq w_{ij} & (i,j) \in F \\
& (t_j - t_i \geq w_{ij}) \vee (-) & (i,j) \in C \\
& (t_j - t_{\sigma(i)} \geq w_{\sigma(i)j}) \vee (t_i - t_{\sigma(j)} \geq w_{\sigma(j)i}) & ((\sigma(i),j),(\sigma(j),i)) \in A
\end{aligned}
\tag{1}
$$

$\sigma(i)$ is the operation which follows i on the route of the associated train, and the precedence relation $t_j - t_i \geq w_{ij}$ ensures that j starts after t_i plus a time lag w_{ij}, as described in [1]. There are two objective functions: the minimization of the maximum consecutive delay t_n and the maximization of the total value of all connections enforced. The function $\delta(x)$ is equal to 1 if $x \geq 0$ and is equal to 0 if $x < 0$. $\delta(x)$ is used to take into account the value v_{ij} of each connection $t_j \geq t_i + w_{ij}$ kept.

The solution procedure consists of estimating the *Pareto front* of non-dominated solutions for the BCDR problem. The solution strategy adopted in this paper consists of iteratively solving the CDR problem (with fixed connections) and then searching for a different set of connections to be enforced. For the selection of the connections to be enforced, we develop and test two new algorithms for the BCDR problem, called *Add* and *Remove*, based on the metaheuristic framework proposed by Paquete and Stützle [3]. Both algorithms use the branch and bound described in [1] to solve the CDR problem and maintain an archive Z of non-dominated solutions which is returned at the end of the search. In what follows, C is the set of all transfer connections to keep or drop and $S^C \subseteq C$ is the set of enforced transfer connections. We let $D(S^C)$ be the maximum consecutive

delay associated to an optimal solution to the CDR problem for a given set S^C. We also let $V(S^C)$ be the total value of the connections satisfied in this solution. The pair $[V(S^C), D(S^C)]$ is the associated point in the plane of the two objective functions for the BCDR problem. Figure 1 shows the sketch of Algorithm Add.

Algorithm Add
Set $S^C = \emptyset$ and compute $D(S^C)$
Initialize archive Z with element $S^C = \emptyset$ and attributes $[0, D(S^C)]$ and visited flag $f(S^C) = 0$
    while there is at least an element in the archive with $f(S^C) = 0$ **do**
        Select an element S^C with $f(S^C) = 0$ from the archive Z
        for all connections $j \in C - S^C$ **do**
            Generate a neighbor $\hat{S}^C = S^C \cup \{j\}$
            if the set $\hat{S}^C$ is not in Z **do**
                Compute $V(\hat{S}^C)$ and $D(\hat{S}^C)$
                Append $\hat{S}^C$ to Z with $[V(\hat{S}^C), D(\hat{S}^C)]$ and $f(\hat{S}^C) = 0$
            end if
        end for
        Set $f(S^C) = 1$
        Remove from Z all the dominated elements
    end while

Figure 1: Pseudocode of the Add algorithm.

Each solution in the archive is characterized by the set S^C with attributes $[V(S^C), D(S^C)]$ and a visited flag $f(S^C)$ initially set to 0. This flag is used during the search to keep track of the already visited solutions (with $f(S^C) = 1$). Initially, a starting solution is inserted in the archive depending on the chosen algorithm ($S^C = \emptyset$ for the Add algorithm and $S^C = C$ for the Remove algorithm). A neighbor $\hat{S}^C$ is the set obtained by adding to S^C a single connection in $C - S^C$ (algorithm Add) or removing a single connection from S^C (algorithm Remove).

The computational study is based on the railway network around the main station of Utrecht, in the Netherlands. We use a peak hour of the 2008 timetable with 79 trains, mostly passenger trains and a few freight trains, and 451 resources, either block sections or platforms. The resulting alternative graph has $|N| = 1847$ nodes, $|F| = 2156$ arcs, $|A| = 4773$ pairs of alternative arcs. As for the set C of connections, we analyze two scenarios. The first one includes 12 passenger connections and 7 non-relaxable rolling stock connections. The second, more challenging, scenario includes 24 relaxable connections.

For each scenario, a set of 25 perturbation instances is generated for different values of train delays. Entrance delays range from 0 to a maximum of 1313 seconds, with an average of 181.6 seconds. About 25% of all trains are delayed at their entrance into the area by more than 5 minutes. For each perturbed situation, non-dominated solutions to the BCDR problem are computed. Table 1 reports on the Pareto front generated by the Add and Remove algorithms. For the first scenario we also compare the performance of the two algorithms with the Pareto front computed by enumerating all possible combinations of enforced connections. For each algorithm, we report on the average number of instances of the CDR problem that have to be solved for an instance of

the BCDR problem, on the average time required to compute the Pareto front, on the number of non-dominated solutions, on the Pareto front area, and on the percentage of times the branch and bound code reaches the time limit before proving the optimality of the solution found.

Table 1: Average results of the Pareto front algorithms.

Scenario	Algorithm	Scheduler Calls	Total Time (s)	# of P. F. Solutions	P. F. Area (%)	Time Limits BB (%)
First	Remove	36	283	3.32	26	18
	Add	20	166	3.32	26	23
	Exhaustive	4096	33504	3.32	26	20
Second	Remove	91	705	4.04	28	18
	Add	36	309	4.00	27	25

Both algorithms are very effective in generating the Pareto front. For the first scenario, Add and Remove find the same Pareto front of the exhaustive search within less than 1% of computation time. For the second scenario, the number of non-dominated solutions found by Add algorithm is only 1% less than Remove. Algorithm Add is quite more efficient than algorithm Remove, its computation time being only 58% [respectively 44%] than the computation time of Remove for the first [the second] scenario. Overall, these results demonstrate that finding a compromise solution between delay minimization and connection satisfaction deserves a high potential for advanced performance management.

References

[1] D'Ariano, A., Pacciarelli, D., Pranzo, M. (2007). A branch and bound algorithm for scheduling trains in a railway network. *European Journal of Operational Research* **183** (2), 643–657.

[2] Mascis, A., Pacciarelli, D. (2002). Job shop scheduling with blocking and no-wait constraints. *European Journal of Operational Research* **143** (3), 498–517.

[3] Paquete L., Stützle T. (2006). A study of stochastic local search algorithms for the biobjective QAP with correlated flow matrices, *European Journal of Operational Research* **169** (3), 943–959.

[4] Schöbel, A. (2001). A model for the delay management problem based on mixed-integer programming. Electronic Notes in Theoretical Computer Science **50** (1), 1-10.

[5] Vromans, M.J.C.M. (2005). *Reliability of Railway Systems*. TRAIL Thesis Series T2005/7, The Netherlands.

Optimal location of one-way carsharing depots

Gonçalo Homem de Almeida Correia

Department of Civil Engineering

University of Coimbra, Portugal

Email: gcorreia@dec.uc.pt

António Pais Antunes

Department of Civil Engineering

University of Coimbra, Portugal

1 Introduction

Carsharing systems are an alternative to private vehicle ownership. Instead of owning a vehicle, a person accesses a fleet of shared-use autos, benefiting from choosing the one that best fits his/her needs for a specific objective [1].

Carsharing began in Europe in the 1940s. One of the earliest European experiences is that of a cooperative known as Sefage, in Zurich, Switzerland, in 1948 [2]. In the US it started with the Mobility Enterprise program in 1983, evolving through field experiments into successful carsharing services [2].

Carsharing schemes span from community-level to national organizations with thousands of members. Some schemes are non-profit making, while others are commercial ventures. The concept of carsharing may vary, and has been divided in two different types: "Station-Cars" and "Carsharing". This division derives from the location of carsharing depots: in the first case they are placed at transit stations and in the second they are distributed independently of transit stations [3]. In the second case the service maybe divided in one-way carsharing and round-trip carsharing.

Carsharing systems developed mostly in the two-way modality, also known as "car clubs", and are currently an expanding market in USA. The one-way sharing mode has not been a priority in the meanwhile [4]. In a survey reported in 2006, vendors who sell technology to carsharing programs in the US believed "that carsharing operators are not likely to introduce innovative features such as one-way rentals because of added management complexities" [4]. Moreover, a discouraging experiment happened recently. One of the most innovative carsharing services in the world, offered in Singapore by Honda was terminated after six years [5]. It offered one-way trips between any one of 21 depots with no reservation required and no return time needing to be specified. The service had 2,500 members with access to 100 vehicles. As membership grew the company was not able to maintain the service quality which was set initially.

Recently, carsharing problems started to be addressed from an optimization perspective. Wei et al. [6] addressed the fleet management problem by formulating a stochastic linear integer model for vehicle allocation and best trip selection for maximizing profit, taking vehicle relocation costs into account. Kek et al. [7] went further and addressed the management of a team of people for relocating the vehicles when these operations need to be done by staff members.

Despite the effort, techniques have not considered real OD matrices, nor have addressed the issue of location of carsharing depots and its influence in profit maximization. The article to be presented takes the previous formulations and expands them considering OD matrices and addressing the issue of depot location and size.

2 Mixed Integer Programming Formulation

The one-way carsharing depot location and size problem can be stated as follows: "given a set of geographically scattered sites where depots can be located, with each depot being small, medium or large size (number of parking spaces), and customer pick-up and return matrices, the objective is to determine the location and size of depots as well as allocating trips and relocation operations as to maximize the profit of running the one-way carsharing operation in an average working day".

Consider the following notation:

<u>Sets</u>: $N = \{1, .., W\}$ set of available sites for one-way carsharing depots, where W is the maximum number of depot locations; $S = \{Small, Medium, Large\}$ set of possible sites for depot location; $V = \{1_1, ..., i_{t-1}, i_t, i_{t+1}, ..., N_T\}$ representing all the $N \times T$ nodes with T as the limit for the optimization period.

<u>Decision variables</u>: y_i^k: Binary variable for the existence of one depot located in i of type k $\forall i \in N$ and $k \in S$; $D_{i_t j_{t+\delta_{ij}}}$: Continuous variable for the number of trips between depot i and j from time step t to $t + \delta_{ij}$; $R_{i_t i_{t+\delta_{ij}}}$: Number of relocated vehicles between depot i and j from from time step t to $t + \delta_{ij}$; a_{i_t}: Number of unused vehicles at depot i in instant t.

<u>Parameters</u>: A matrix of travel times between each depot is given by δ_{ij}, *the matrix of distances* is given by γ_{ij}, and the demand matrices for carsharing vehicles at time instant t is given by T_{ij_t}. The known constants are: I: Income per km driven by a client; C_m: Cost of maintaining one parking space per day; C_r: Cost of relocating a vehicle per km driven; C_d: Cost of rejecting the demand of one customer-vehicle trip; C_v: Cost of the depreciation of one vehicle per day; p^k: Number of parking spaces for each depot size k; and M as the maximum number of vehicles to be relocated in each instant for each depot.

Using this notation, the objective function can be written as:

$$Max(P) = I \sum_{i_t \in V} D_{i_t j_{t+\delta_{ij}}} \gamma_{ij} - C_r \sum_{i_t \in V} R_{i_t j_{t+\delta_{ij}}} \gamma_{ij} - C_d \sum_{i_t \in V} \left(T_{ij_t} - D_{i_t j_{t+\delta_{ij}}} \right) - C_m \sum_{i \text{ in } I, k \in S} y_i^k p^k$$

$$- C_v \sum_{i \in N, k \in S} a_{i_1}$$

This function maximizes the total profit (P), taking into consideration as income the number of km driven by the customers, and as costs the relocation expenses, rejecting demand, costs for maintaining each depot according to their size and cost of owning the vehicles.

Subject to,

$$\left(\sum_{j_t \in V} D_{j_{t-\delta_{ij}} i_t} + \sum_{j_t \in V} R_{j_{t-\delta_{ij}} i_t} \right) - \left(\sum_{j_t \in V} D_{i_t j_{t+\delta_{ij}}} + \sum_{j_t \in V} R_{i_t j_{t+\delta_{ij}}} \right) + a_{i_t} = a_{i_{t+1}} \ \forall \ i_t \in V$$

Ensures the conservation of flows at each node at i_t and updates the available number of vehicles at each depot from time step t to $t+1$;

$$\sum_{j \in N} D_{i_1 j_{1+\delta_{ij}}} + \sum_{j \in N} R_{i_1 j_{1+\delta_{ij}}} \leq a_{i_1} \ \forall \ i \in N$$

Guarantees that at the first instant there are only flows of vehicles getting out of each depot and that the total flow has to be lower than the available number of vehicles at instant $t = 1$;

$$D_{i_t j_{t+\delta_{ij}}} \leq T_{ij_t} \times \sum_{k \in S} y_i^k \ \forall \ i_t, j_t \in V$$

Ensures that the number of accepted trips between i and j must be lower than the actual trips, and it is constrained to be zero when there is no depot in the ith site;

$$D_{i_t j_{t+\delta_{ij}}} \leq T_{ij_t} \times \sum_{k \in S} y_j^k \ \forall \ i_t, j_t \in V$$

Forces the accepted trips between i and j to be zero when there is no depot in the jth site;

$$\sum_{j_t \in V} R_{i_t j_{t+\delta_{ij}}} \leq M \sum_{k \in S} y_i^k \ \forall \ i_t \in V$$

Ensures that the relocations will not exceed the M limit for each node i_t and must only apply to existing depots at i^{th} site;

$$\sum_{i_t \in V} R_{i_t j_{t+\delta_{ij}}} \leq M \sum_{k \in S} y_j^k \ \forall \ j_t \in V$$

Guarantees that the relocations will not exceed the M limit for each node j_t and must only apply to existing depots at j^{th} location;

$$\sum_{k \in S} y_i^k \leq 1 \ \forall \ i \in N$$

Ensures that in a given site there will be a small, medium or large depot, or no depot;

$$a_{i_t} \leq \sum_{k \in L} y_i^k p^k \forall \ i \in N$$

Forces the number of idle vehicles in each depot i to be less than the depot's capacity;

$$y_i^k = (0,1) \ \forall \ k \in L$$

$$a_{i_t}, D_{i_t j_{t+\delta_{ij}}}, R_{i_t j_{t+\delta_{ij}}} \geq 0 \ \forall \ i_t \ \in \ V$$

3 Applying the formulation for the Lisbon Metropolitan Area (LMA)

Some initial experiments have been conducted using the Lisbon Metropolitan Area (LMA) as case-study. A geocoded origin-destination survey allowed to obtain OD matrices considering depot coverage areas in a buffer for each possible site.

In these experiments, 20 possible sites were used for sub-areas of the LMA with diversified trip patterns. Tentative results using mean value parameters for the one-way carsharing business point to different potentials from sub-area to sub-area following the natural balance /unbalance of the enclosed trips.

For the cases where commuter trips were not linked to other trips, tentative results indicate that this system is economically disadvantageous, as vehicles have to stay idle for most of their time waiting to be picked up again to return home. Hence depot location appears to be determinant for influencing the viability of one-way carsharing schemes.

References

[1] Shaheen, S. A., Sperling D. and Wagner C., "Carsharing and Partnership Management An International Perspective", *Transportation Research Record*, Vol. 1666, 118-124 (1999).

[2] Shaheen, S. A. and Cohen A. P., "Growth in Worldwide Carsharing - An International Comparison", *Transportation Research Record*, Vol. 1992, 81-89 (2007).

[3] Barth, M. and Shaheen S. A., "Shared-Use Vehicle Systems - Framework for Classifying Carsharing, Depot Cars, and Combined Approaches. *Transportation Research Record*, Vol. 1791, 105-112 (2002).

[4] Shaheen, S. A., Cohen A. P. and Roberts J. D., "Carsharing in North America - Market Growth, Current Developments, and Future Potential", *Transportation Research Record*, Vol. 1986, 116-124, 2006.

[5] The Straits Times, "Innovative Carsharing Program in Singapore Ends", Singapore (2008).

[6] Fan, W., Machemehl R. B. and Lownes N. E., "Carsharing Dynamic Decision-Making Problem for Vehicle Allocation". *Transportation Research Record*, Vol. 2063, 97-104 (2008)

[7] Kek, A. G. H., Cheu R. L., Q. Meng and Fung C. H., "A Decision Support System for Vehicle Relocation Operations in Carsharing Systems", *Transportation Research Part E-Logistics and Transportation Review*, Vol. 45, No.1, 149-158 (2009).

A single-allocation hub location problem with capacity choices

Isabel Correia

Departamento de Matemática - CMA, Faculdade de Ciências e Tecnologia

Universidade Nova de Lisboa, Caparica, Portugal

Stefan Nickel

Institute for Operations Research

University of Karlsruhe, Karlsruhe, Germany

Francisco Saldanha-da-Gama

CIO-DEIO, Faculdade de Ciências

Universidade de Lisboa, Lisboa, Portugal

Email: fsgama@fc.ul.pt

1 Introduction

A major goal in many traffic networks is to provide cost effective means to establish the flow from a set of sources to a set of destinations.

Consider a complete graph and assume that for each pair of nodes there is a bidirectional flow to be sent between them. In practice, for efficiency purposes and in order to reduce costs, some nodes of the network are selected to become hubs and are used as consolidation and redistribution points that together process more efficiently the flow in the network. Accordingly, hubs are nodes that receive traffic (passengers, phone calls, mail, etc) from different origins (nodes) and redirect this traffic directly to the destination nodes (when a link exists) or to other hubs. With the concentration of traffic in the hubs and its shipment to other hubs it is possible also to take advantages of economies of scale, which leads to a natural decrease in the overall cost. The problem of deciding which nodes in a network should become hubs and how the flow should be consolidated and redistributed defines the basic setting of a hub location problem (see Campbell et al. [3]). When a non-hub node must have its traffic routed via exactly one hub we have a single-allocation hub location problem.

Capacitated hub location problems have applications in many areas such as cargo delivery sys-

tems, telecommunication networks and public transportation networks among others (see Alumur and Kara [1] for further details).

In this work we propose an extension to the capacitated single-allocation hub location problem in which the capacities of the hubs are part of the decision making process. We propose and compare two sets of mixed-integer programming formulations for the problem. We also propose several valid inequalities and preprocessing tests. The results of the computational tests performed to evaluate the different models and enhancements proposed are reported.

2   Problem description

In its basic setting, the capacitated single-allocation hub location problem (CSAHLP) has the following features (For further details and literature on this problem see Campbell et al. [3] and Alumur and Kara [1]):

- The inter-hub network is a clique.

  This has been considered in many hub location problems supported by practical requirements although in some works this feature has been relaxed (e.g. Alumur and Kara [2] and Nickel et al. [4]).

- No direct links are created between non-hub nodes.

  Accordingly, the only consolidation points should be the hubs. All traffic originated in each node should be routed via at least one hub.

- The hubs are capacitated.

  We assume that the capacity only constrains the incoming flow (see Campbell et al. [3] for further details on the practical motivations behind this assumption).

- The decisions to be made comprise: i) which nodes should be selected to become hubs; ii) how to allocate the non-hub nodes to the hubs.

- The costs involved in the problem are the set-up costs for the hubs and the costs associated with the flow. In the latter case, three cost components can be distinguished: collection cost (from the non-hub nodes to the hubs), distribution cost (from the hubs to the non-hubs) and discount costs (between hubs).

- The objective is to minimize the overall cost for building and operating the network.

We extend the problem above by considering the capacity of the hubs as part of the decision making process. This is motivated by the fact that often, hubs are large structural facilities requiring several strategic decisions to be made in addition to the location decisions. One such decision is exactly their dimension/capacity for processing flow. For each potential hub we consider

a set of available capacity levels among which at most one can be chosen. Each capacity level determines a capacity for the incoming flow and incurs a specific fixed set-up cost. Economies of scale are assumed for these costs. In addition to the decisions listed above, we consider the capacity level at which each hub should operate.

In order to illustrate the flexibility in the network design process and the advantages in terms of the overall cost that can be obtained with the extension just proposed, consider a problem with 9 nodes depicted in figure 1. Assume that each square in the grid has a unitary length. Additionally assume that i) each node should send 2 units of flow/traffic to every other node (thus, each node originates 16 units of flow); ii) the cost for sending a unit of flow between two hubs is equal to 0.75 times the distance between the hubs; iii) the cost for sending one unit of flow between a non-hub node and a hub as well as between a hub and a non-hub is equal to the distance between the nodes involved; iv) the distance between two nodes in the network is given by the euclidean distance.

Figure 2a represents the optimal solution (with value $546 + 54\sqrt{2}) \approx 622.4$) for the situation in which i) a hub can be installed in every node with a set-up cost equal to 100 for nodes 2, 5 and 7 and equal to 500 for the other nodes; ii) the flow consolidation capacity of each potential hub is equal to 50 (maximum flow/traffic that a hub can receive from the nodes connected to it);

Figure 2b represents the optimal solution to the problem (with value $552 + 32\sqrt{2} \approx 597.3$) assuming that it is possible to make a decision about the capacity of a hub to be installed in nodes 2 or 5. In particular, assume that one additional capacity level equal to 80 is available in each of these nodes, with a set-up cost of 120.
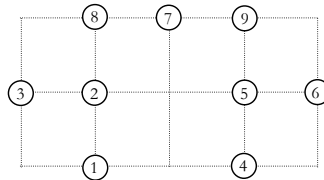
Figure 1: A set of nodes defining a CSAHLP.
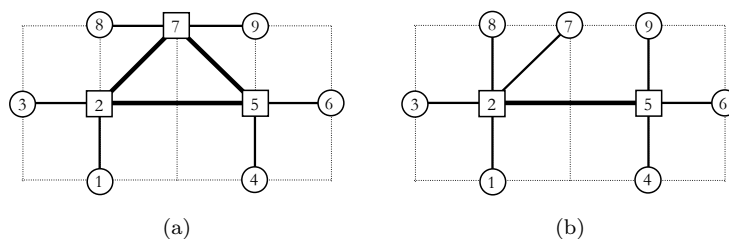
Figure 2: Flexibility in the network design arising from the existence of different capacity levels.

In this small example we can see that with the introduction of capacity choices more flexibility is given to the network design decisions and a lower cost is obtained.

3   Methodology for approaching the problem

The new hub location problem we are considering can be easily formulated by adapting to the new situation several well-known MIP formulations for the CSAHLP. Nevertheless, other possible MIP formulations can be considered. This is the case when we consider allocation variables (of the non-hub nodes to the hubs) indexed in the different capacity choices that exist for the potential hubs. We propose such formulations.

The interesting question arising is how the formulations compare with each other theoretically and how far is it possible to go with them from a computational point of view. Note that we are working with an NP-hard problem (the new problem has the CSAHLP problem as a particular case), so an exact approach is expected to be successful only for small or medium size instances.

We make a theoretical comparison between the formulations proposed namely in terms of the bounds provided by their linear relaxations. We also propose a set of valid inequalities to enhance the models. Several preprocessing tests are also proposed aiming at reducing the size of the models considered.

We run a series of computational tests in order to evaluate the performance of the different models proposed as well as of their enhancements. For doing so and taking into account that no benchmark instances exist for the new problem, we considered benchmark instances for the CSAHLP and generated the necessary data to obtain instances for the new problem. The results emphasize the superiority of one specific model when a general solver was considered for solving the problem to the optimality. The results also show the superiority of a different model in terms of the bound provided by the linear relaxation.

References

[1] S. Alumur and B. Y. Kara, "Network hub location problems: the state of the art", *European Journal of Operational Research* 190, 1-21 (2008).

[2] S. Alumur and B. Y. Kara, "The design of single allocation incomplete hub networks", *Transportation Research Part B* 43, 936-951 (2009).

[3] J. F. Campbell, A. T. Ernst and M. Krishnamoorthy, "Hub location problems", in *Facility Location: Applications and Theory*, Z. Drezner and H. W. Hamacher (eds), 273-407, Springer, 2002.

[4] S. Nickel, A. Schobel and T. Sonneborn, "Hub Location problems in urban traffic networks", in *Mathematics Methods and Optimization in Transportation Systems*, J. Niittymaki and M. Pursula (eds), 1-12, Kluwer Academic Publishers, 2001.

Quantifying variability due to incidents including en-route rerouting

Ruben Corthout

Department of Mechanical Engineering

CIB/Traffic & Infrastructure

Katholieke Universiteit Leuven

Email: ruben.corthout@cib.kuleuven.be

Chris M.J. Tampère

Department of Mechanical Engineering

CIB/Traffic & Infrastructure

Katholieke Universiteit Leuven

Lambertus H. Immers

Department of Mechanical Engineering

CIB/Traffic & Infrastructure

Katholieke Universiteit Leuven

1 Introduction

In dynamic traffic assignment (DTA) problems, reliability is increasingly acknowledged as an important factor influencing the decisions of travellers (such as modal choice, departure time choice and route choice). Several studies have tried to estimate the contribution of reliability to the choice behaviour of travellers, and to determine how and to what extent it is to be considered in choice models in DTA (e.g. [1], [2], [3], [4], [5]). However, before variability itself and the respons of drivers can be adequately modeled by DTA models, several problems have to be overcome (see also [6]):

- Despite various studies (e.g. [1]-[5]), an agreement on the valuation and measure of variability to be used in generalised cost funtions in choice models (determing route choice, departure time choice and – in multimodal models – modal choice) is still to be reached.

- State-of-the-art Dynamic Network Loading (DNL) models used in DTA to propagate traffic over networks are deterministic by nature. With these models, the variability of traffic states and travel times from day to day can only be quantified by performing a large number of Monte-Carlo simulations with varying input (such as demand, route choice and capacity). Due to high computation times, this approach is unfeasible with state-of-the-art dynamic models. Thus, efficient algorithms are to be developped to produce – most likely in an approximate way - probability distributions of traffic states and route travel times, rather than a one-shot deterministic prediction. Recently, some first steps have been taken towards the development of such stochastic DNL models ([7],[8],[9]).

- Introducing variability may necessitate a reformulation of the DTA framework. It is questionnable if feeding DNL models with a route choice that is fully determined prior to departure – even if this route accounts for the influence of variability – and iterating towards a dynamic user equilibrium, sufficiently represents reality. Drivers may respond to variability not by a priori choosing a reliable route, but by opting for the fastest – not necessarily reliable – route and rerouting in case of above average congestion. This strategy is aided by various information systems available to drivers pre- and en-route. How and to what extent different levels of choice need to be considered in DTA to capture the dynamic character of traffic - this may differ depending on the application - is an open issue for future research and debate [10].

2 Including en-route rerouting in the Marginal Incident Computation model

In [11], we introduced the Marginal Incident Computation (MIC) model, a highly efficient algorithm that approximately quantifies congestion spillback and the corresponding travel time increase due to incidents. The MIC model superimposes the effect of every incident onto the outcome of a single base DNL without incidents. This base simulation can be obtained from any existing DNL model. The output needed from the base DNL consist of the curves of the cumulative vehicle numbers, which consequently serves as input to the MIC model. The base cumulative curves of the links where the traffic flows are influenced are altered according to the additional constraint imposed by the incident. This is done according to Newell's simplified first-order kinematic wave theory [12]. Since only the additional congestion due to an incident is calculated, computational effort is limited to a fraction of all links and time intervals. Computation time can be reduced to less than 0.1% compared to a full, explicit simulation of each incident case (depending on the network size). Thus, the MIC model renders extensive Monte-Carlo sampling feasible, allowing fine sampling of incident duration, severity, and starting times. In [9], the usefulness of the MIC model in the context of a stochastic DNL

is demonstrated. By accounting for incident induced variability, an onset of a (partial) answer to the second question formulated in Section 1 is provided.

A rather stringent assumption of the MIC model as presented in [11] was that drivers make the same journey in case of an incident as they do in the base simulation. In this paper, an approximate procedure is added to the MIC model to account for en-route rerouting. This procedure is based on the hybrid route choice modeling introduced in [13]. Herein, a route choice model defines pre-trip route choice for all drivers. However, during the DNL, the pre-trip computed route flow rates are updated at every network node. This allows drivers to re-evaluate the pre-trip route choice and possibly deviate to an alternative route. An additional term (weighted with one single parameter ω) is introduced into the cost functions of the logit route choice model to express drivers' reluctance to move away from their initial route.

En-route rerouting is incorporated into the MIC model in the following way. At every node that the queue spilling back from the incident reaches, a k-shortest path logit route choice model is run. This route choice is performed between each initial route that passes through this node into the link from which congestion spills back and the k alternatives from this node towards the destination of the initial route. The cost function used in the logit route choice model contains the instantaneous route travel times and, for the alternative routes, the additional term to account for drivers' reluctance to reroute (as in [13]). In reality, drivers will have incomplete – and possibly incorrect - information about current and future traffic conditions and travel times. In any case, this route choice is not an equilibrium since this cannot be reached under unexpected traffic conditions. Here, drivers are assumed to base their en-route route choice on instantaneous travel times, which is probably more realistic than using experienced travel times, since these are unknown to drivers. At each node, the en-route route choice model determines how many people reroute and thus how the turning fractions (determining the proportion of traffic in each downstream direction) change. Due to rerouting, a higher proportion of drivers is directed towards non-congested downstream links and thus the outflow of each incoming link of the node will increase. As a result, the incident congestion will spill back slower and reach less far.

References

[1] N. De Palma and N. Picard, "Route choice decision under travel time uncertainty", *Transportation Research Part A* 39, 295-324 (2005).

[2] W. Jackson and J. Jucker, "An empirical study of travel time variability and travel choice behavior", *Transportation Science* 16(4), 460-475 (1981).

[3] T.C. Lam and K.A. Small, "The value of time and reliability: measurement from a value pricing experiment", *Transportation Research Part E* 37, 231-251 (2001).

[4] H.X. Liu, W. Recker and A. Chen, "Uncovering the contribution of travel time reliability to dynamic route choice using real-time loop data", *Transportation Research Part A* 38, 435-453 (2004).

[5] R.B. Noland, K.A. Small, P.M. Koskenoja and X. Chu, "Simulating travel reliability", *Regional Science and Urban Economics* 28(5), 535-564 (1998).

[6] R Batley, J. Bates and S. Hess, "Multimodal travel time variability", *ITS Leeds* (2008).

[7] A. Sumalee, R. Zhong, W.Y. Szeto and T. Pan, "Stochastic Cell Transmission Model: Traffic State Modeling Under Uncertainties", *Presented at the 88th Annual Meeting of the Transportation Research Board*, January 11-15, 2009.

[8] M. Ng and S.T. Waller, "Dynamic Route Choice Modelling in Face of Uncertain Capacities", *Presented at the 88th Annual Meeting of the Transportation Research Board*, January 11-15, 2009.

[9] R. Corthout, C.M.J. Tampère and L.H. Immers, "Stochastic Dynamic Network Loading for Travel Time Variability due to Incidents", To appear in *New Developments in Transport Planning: Advances in Dynamic Traffic Assignment (Proceedings of DTA2008, Leuven),* Edward Elgar, Cheltenham, UK - Northampton, MA, USA (2008).

[10] F. Viti and C.M.J. Tampère, "Dynamic Traffic Assignment: Recent Advances and New Theories towards Real-time Applications and Realistic Travel Behaviour", To appear in *New Developments in Transport Planning: Advances in Dynamic Traffic Assignment (Proceedings of DTA2008, Leuven),* Edward Elgar, Cheltenham, UK - Northampton, MA, USA (2008).

[11] R. Corthout, C.M.J. Tampère and L.H. Immers, "Marginal Incident Computation, An Efficient Algorithm to Determine Congestion Spillback due to Incidents", *Transportation Research Record* 2099, 22-29 (2009).

[12] G.F. Newell, "A simplifief theory of kinematic waves in highway traffic, Part I: General Theory, Part II: Queuing at freeway bottlenecks, Part III: Multi-destination flows", *Transportation Research B* 27, 281-313 (1993).

[13] A.J. Pel, M.C.J. Bliemer and S.P. Hoogendoorn, "Hybrid route choice modeling in dynamic traffic assignment", *Transportation Research Record* 2091, 100-107 (2009).

Integrated Scheduled Service Network Design for Freight Rail Transportation

Teodor Gabriel Crainic

Dept. management et technologie, ESG UQAM, Montréal, Canada

and

Interuniversity Centre for Enterprise Networks, Logistics and Transportation (CIRRELT)

Email: TeodorGabriel.Crainic@cirrelt.ca

Michel Gendreau

Dept. de mathématiques et de génie industriel, École Polytechnique de Montréal
and CIRRELT

Endong Zhu

Dept. d'informatique et de recherche opérationnelle, Université de Montréal
and CIRRELT

Cargo is moved for a large part by *consolidation-based* carriers: railroad, less-than-truckload motor carriers, container ships, regular and express-currier services, etc. The fundamental idea of consolidation-based transportation is to group loads of different shippers, with possibly different origins and destinations, and to load them into the same vehicles for efficient long-haul transportation. The performance and profitability of such a system depend for a large part on efficient and coordinated terminal and long-haul transport operations. A set of regular, often scheduled, transportation services is the result of these operations and constitutes the offer of service the carrier proposes to its potential customers. The tactical planning process producing this transportation plan is generally known as the *service network design problem* [2].

Rail carriers generally implement a double consolidation policy: loaded and empty cars are grouped into so-called blocks, which are then grouped again to make up the trains. Cars with different origins and destinations being present simultaneously in the same yard (a major terminal suitably equipped) are thus sorted and grouped into a *block*, which is moved as **a single unit** by a series of trains until its destination, where it is broken down, the cars being either delivered to their final consignees or sorted for inclusion into new blocks. The associated operations and policies are

denoted car *classification* (sorting), *blocking*, block *transfer* (from one train to another), and train *make up*.

Tactical planning aims to select the train services to operate over the contemplated schedule length (e.g., the week) and their frequencies or schedules (time tables), the blocks that will make up each train, the blocks to be built in each yard, and the routing of the cars loaded with the customers' freight using these services and blocks (empty-car movements are also considered).

A rich literature exists on models and methods addressing these issues (two surveys: [1, 2]). Most either address a single or a limited number of issues, or make significantly simplifying hypotheses. At the best knowledge of the authors, in fact, no model currently available in the literature addresses in an integrated, comprehensive formulation all the tactical planning issues. Our goal is to answer this challenge and present an integrated scheduled service network design methodology for rail freight transportation.

1 Problem Setting

Railroads are complex transportation systems where several major components interact and compete for resources. The infrastructure of the system is made up of a large number of stations where freight originates and terminates, and yards where cars are sorted and blocks and trains are built and taken apart, and rail tracks linking them. To simplify the presentation, we focus on the main-line network and assume all stations operate as yards. Yard operations are constrained by their capacity (with respect to a given time period) in terms of number of cars that may be classified, number of blocks that may be built (number of classification tracks), number of trains that may be made up or stop. Inter-yard movements are also constrained by the capacity of the rail tracks in terms of number of trains that can be accommodated simultaneously.

Service is provided by trains. A train service is made up at an origin yard, follows a given route stopping, eventually, at a series of intermediary terminals, and is broken down at its destination yard. Its schedule indicates arrival and departure times at each one of these yards. Each train service is also characterized by its power and speed, which determine its capacity (for simplicity, we assume it is measured in number of cars) and travel time, respectively. At each station, the train picks up and delivers blocks of cars.

Each customer demand consists of cars of particular types to be moved from their respective origins to their destinations, within their temporal requirements: availability at origin, due-time at destination. At the origin yards, cars are classified and grouped with cars of other customers (and empty cars) into a block, which will be put on a train formed in the same yard or on a train stopping at the yard. They will be then delivered to the final destination by a series of trains and blocks, each block being possibly moved by several trains.

Given forecast customer demand and sets of potential services (train departures) and blocks, the scheduled service network design problem we address simultaneously selects services, the blocks to be built at each yard, and the itineraries - the series of blocks and trains - for each demand, to minimize the total cost of the system.

2 Modelling Framework

Let $G = (V, E)$ denote the physical rail network, where the node set V represents the yards and the set of links E stands for the possible directional movements on track sections between adjacent yards. Let T be the length of the *cyclic* schedule, divided into time periods $\{0, \cdots, T-1\}$, where period 0 follows period $T-1$.

We build a three-layer time-space structure to represent the possible decisions and activities making up the schedule and related to services, blocks, and cars, respectively [3]. The time-space network of each layer is made up of nodes representing the yards at each time period and a number of arcs connecting these nodes and standing for the specific activities and delays, within yards and movements between them, associated to the layer. Let A stand for the links of the car and block layers (on which cars may move or wait in yards). Inter-layer links represent the consolidation (and de-consolidation) operations - cars into blocks, blocks into trains - and complete the time-space network. Each demand p is associated to an origin node and period in the car layer, a maximum delivery time, a destination yard, and a number of cars of particular type.

Three sets of decision variables are defined:

$y_b \quad \in \{0,1\}$ Block selection: $y_b = 1$ if block $b \in B$ is selected, 0 otherwise;

$z_l \quad \in \{0,1\}$ Service selection: $z_l = 1$ if service $l \in L$ is selected, 0 otherwise;

$x_a^p \quad \geq 0$ Number of cars of traffic class p traveling on link $a \in A$.

The Scheduled Service Network Design Problem may then be formulated as

$$\min \sum_{p \in P} \sum_{a \in A} c(p,a) \cdot x_a^p + \sum_{b \in B} c^f(b) \cdot y_b + \sum_{l \in L} c^f(l) \cdot z_l \tag{1}$$

$$\text{s.t.} \quad \sum_{a \in A^+(n)} x_a^p - \sum_{a \in A^-(n)} x_a^p = w_n^p \qquad \forall n \in N, \forall p \in P; \tag{2}$$

$$\sum_{p \in P} x_a^p \leq u_a \qquad \forall a \in A^c; \tag{3}$$

$$\sum_{l \in L(e,t)} z_l \leq u_e \qquad \forall e \in E, \forall t \in \{0, \cdots, \mathbf{T}-1\}; \tag{4}$$

$$\sum_{f \in F(l) | a \in A^v(f)} \sum_{b \in B | f \in F(b)} \sum_{p \in P} x_b^p \leq z_l u_l \qquad \forall a \in A^v, l \in L; \tag{5}$$

$$\sum_{b \in B(a)} y_b \leq u_{v(a)} \qquad \forall a \in A^h; \tag{6}$$

$$\sum_{p \in P} x_b^p \leq y_b u_b \qquad \forall b \in B; \tag{7}$$

$$\sum_{f \in F(l)} \sum_{b \in B | f \in F(b)} y_b \leq z_l u_l |A^v(l)| \forall l \in L; \tag{8}$$

$$x_a^p \geq 0 \qquad \forall a \in A, \forall p \in P; \tag{9}$$

$$y_b \in \{0, 1\} \qquad \forall b \in B; \tag{10}$$

$$z_l \in \{0, 1\} \qquad \forall l \in L. \tag{11}$$

The objective function (1) sums operating costs on all layers, where $c^f(l)$ and $c^f(b)$ are the service and block fixed costs, respectively, and $c(p, a)$ is the unit flow cost for demand p on link a. Equations (2) enforce the car flow conservation constraints at all nodes for all demands. Relations (3) - (6) enforce capacity constraints on the numbers of cars on yard classification links, trains on rail tracks, cars on trains, and blocks formed in yards, respectively. Linking constraints (7) and (8) ensure logical feasibility, i.e., cars are assigned to selected blocks, and blocks are moved by selected trains, respectively, as well as the capacity restrictions of blocks (number of cars) and trains (number of blocks).

The dimension of the resulting mix-integer formulation grows rapidly with the size of the rail network, the density of potential services and blocks, and the length of the schedule. In the presentation, we will discuss the application of this modelling approach to the case of direct services, as well as to the general case of services with intermediary stops. We will also present the meta-heuristics we developed for these cases, based on tabu search and slope scaling ideas, respectively. Numerical results on several classes of test problems, including some derived from actual applications, will also be presented and analyzed.

References

[1] J.-F. Cordeau, P. Toth, and D. Vigo, A Survey of Optimization Models for Train Routing and Scheduling, *Transportation Science* 32(4), 380–404, 1998.

[2] T.G. Crainic and K.H. Kim. Intermodal Transportation. In C. Barnhart and G. Laporte, editors, *Transportation*. Vol. 14 of *Handbooks in Operations Research and Management Science*, 467–538, Elsevier, 2007.

[3] E. Zhu, T.G. Crainic, and M. Gendreau, Integrated Service Network Design for Rail Freight Transportation, Publication CIRRELT-2009-45, Centre interuniversitaire de recherche sur les réseaux d'entreprise, la logistique et le transport, Université de Montréal, 2009.

New Fast Heuristics for the Two-Echelon Vehicle Routing Problem

Teodor Gabriel Crainic

Département management et technologie

Université du Québec à Montréal

and

CIRRELT, Montréal

Simona Mancini

Department of Control and Computer Engineering

Politecnico di Torino

Corso Duca Degli Abruzzi 24, Torino, Italy

Email: simona.mancini@polito.it

and

CIRRELT, Montréal

Guido Perboli

Department of Control and Computer Engineering

Politecnico di Torino

and

CIRRELT, Montréal

Roberto Tadei

Department of Control and Computer Engineering

Politecnico di Torino

Multi-Echelon distribution systems are broadly used in practice and have been widely studied in the literature, but attention was mainly focused on flow assignment issues, while the routing of vehicles supporting the flows was generally not considered in the optimization process. Multi-Echelon Vehicle Routing Problems encompass this issue. They address the management of the

fleets required to provide transportation among the different echelons, and the integrated planning of the associated routes. The goal of the system is to deliver at minimum cost goods from one or more depots to customers by consolidating and routing through intermediate depots, called satellites.

In this paper we study the single depot *Two-Echelon Vehicle Routing Problem*, from now on 2E-VRP, made up of two levels of routing activities. At the first level, goods are delivered by a first-level fleet, from the depot to a set of intermediate depots, named satellites, where they are consolidated into second-level vehicles for delivery to customers. These two routing problems are strongly interdependent and are connected by the customer-satellite assignment problem. Service at each level is provided by an homogeneous fleet. We consider constraints on the maximum number of available vehicles for both levels, and on the satellites capacity, which is expressed as number of vehicles starting from a satellite. Satellites capacity may vary among the satellite. The goal is to serve customers minimizing the total transportation cost of the two-echelon system, without violating the capacity constraints of the vehicles. We consider a single depot and a fixed number of capacitated satellites. All customer demands, fixed and known in advance, must be satisfied.

This problem is faced very frequently in real life applications, both at the strategic level (long term planning) and at the operational one (real-time optimization). Methods which can be applied at both levels must be accurate and, at the same time, very fast. In fact, in long term planning, the *2E-VRP* is part of a simulation framework, that means it must be solved a lot of times during the optimization process and for that reason, computational times should be short. At the operational level, real-time optimization problems, for which a feasible solution is needed in a short time, are often faced. On the other hand, accuracy of the solution, is much important, because, on real applications, even a small gain on the objective function could yield a great saving of money for the transportation company.

A formulation for the 2E-VRP has been presented in [4], where instances up to 32 customers were solved to optimality. In the same paper, the authors derived two math-heuristics able to address instances up to 50 customers. Both are based on the LP model presented in the paper and work on the customer-to-satellite assignment variables. The first math-heuristic, called *Diving*, considers a continuous relaxation of the model and apply a diving procedure on the customer-to-satellite assignment variables which are not integer. To recover possible infeasibilities due to the variables fixing, a restarting procedure is incorporated. The second one, named *Semi-continuous*, considers the arc usage variables as continuous, while the assignment variable are still considered as integer. The method solves this relaxed problem and uses the values of the assignment variables obtained to build a feasible solution for the 2E-VRP. A general time-dependent formulation with fleet synchronization and customer time windows has been introduced in [1] in the context of two-echelon City Logistics systems. The authors indicated promising algorithmic directions, but no

implementation has been reported.

We present two fast and accurate heuristics based on separating the depot-to-satellite transfer and the satellite-to customer delivery by solving iteratively the two resulting routing subproblems, while adjusting the satellite workloads linking them. The first one is a clustering based heuristics, the second one is based on a path-relinking procedure.

Both methods are based on a common idea. We split the problem into two routing subproblems, one at each level. The second level problem can be further decomposed into n Vehicle Routing Problems (VRPs), being n the number of satellites, one for each satellite. In every VRP we consider as depot a satellite and as customers only those which have been assigned to it. The customer-to-satellite assignment problem plays a crucial role in the problem solving. In fact, if we suppose to know the optimal assignment, an optimal solution can be easily obtained by solving to optimality the VRP related to each satellite, and the resultant VRP at the first level, in which we consider as customers the satellite with a demand equal to the sum of customer assigned to it. The 2E-VRP can be treated as an assignment problem in which the objective function is given by the solution of $n+1$ VRPs. Since the computational time is mostly due to the routing solving, we cannot neglect this information while developing a fast heuristic method. In fact, methods involving large neighborhood exploration are not suitable for solving this problem, because of the computational time needed to analyze each solution of the assignment problem. For developing a fast heuristic we need a mechanism which can guide us, inside the solution space, to promising solutions, and allow us to obtain good results without exploring a high number of solutions.

The first heuristic we present, named Clustering Improvement (CI), is a clustering based heuristic which aims to improve the assignment given by an initial solution following a local search approach. The local search is a first improvement in which the order according to which we explore the neighborhood is given by a distance based rule. The neighborhood is defined as the set of assignments in which only one assignment is different from the current solution. Being the size of the neighborhood small, we can explore it in a quite short time. The initial solution is obtained by applying the First Clustering (FC), a rule which assigns each customer to the nearest available satellite (some assignments may not be feasible because of satellites and vehicles capacity constraints). The order in which we explore the neighborhood is given by the following rule: we create a list of customers ordered, in non-decreasing order, by the difference between the distances from the satellite to which it has been assigned and its second nearest satellite. We scan the list to determine which customer will be assigned to its second nearest satellite. We refer the reader to [3] for a complete description of this method, and to [2] for an analysis of instances layout on the global transportation cost, in which FC and CI have been applied.

The second method we develop is a path-relinking based heuristic (PR) in which we start from an Elite Set which is made only by the initial solution obtained by FC, we create perturbed solution,

which are quite far in the solution space (high Hamming distance between the two corresponding assignment vectors), and we relink them to the initial one, following a path composed by solutions in which we change at each step one and only one customer-to-satellite assignment. Two different perturbed solution generation methods, both based on a distance rule, are proposed. In the first one a strong random component is taken into consideration in the assignment perturbation, while, in the second one the random component effect is reduced. Three different relinking methods have been adopted. In the first two methods, the choice of the customer to be reassigned at each step is random, but, while in the first method we start from the initial solution, in the second one we start from the perturbed one. The third method, instead, follows a deterministic distance based rule for determining the customer to be reassigned.

We present computational results on a wide set of instances up to 50 customers and 5 satellites and compare it with results from literature. Our methods outperform the existing methods, both in efficiency and effectiveness.

References

[1] T. G. Crainic and N. Ricciardi and G. Storchi, "Advanced freight transportation systems for congested urban areas", *Transportation Research Part C*, 12, 119-137 (2004)

[2] T.G. Crainic, and G. Perboli, and S. Mancini, and R. Tadei "Two-Echelon Vehicle Routing Problem: A Satellite Location Analysis", in *City Logistics VI: Proceedings of the 6th International Conference on City Logistics*, E. Taniguchi, R.G. Thompson (eds), 65-75, Institute for City Logistic, Japan, 2009

[3] T.G. Crainic and S. Mancini and G. Perboli and R. Tadei, "Heuristics for the Two-Echelon Capacitated Vehicle Routing Problem", Internal report CIRRELT-2008-46, Montreal, Canada, 2008

[4] G. Perboli and R. Tadei and D. Vigo, "The Two-Echelon Capacitated Vehicle Routing Problem", Internal report CIRRELT-2008-55, Montreal, Canada, 2008

An Optimization Model for the Pick-up and Delivery of Trucks & Containers Routing with Multiple Container Loads

Teodor Gabriel Crainic

Department of Management and Technology

Université du Québec à Montréal

Massimo Di Francesco

Department of Land Engineering

University of Cagliari, Piazza d'Armi, Cagliari, Italy

Email: mdifrance@unica.it

Paola Zuddas

Network Optimization Research and Educational Centre (CRIFOR),

Department of Land Engineering

Cagliari University

1 Introduction

We study a problem faced by a maritime container shipping company, which has to meet trucking transportation requests arising in the landside. The shipping company must deliver loaded containers to import customers and provide empties for exporters and is requested to determine the routes of trucks in order to serve customers. Trucks can carry more than one container and are requested to wait containers emptied by some import customers becoming available to satisfy the demand of export customers.

A number of papers have been proposed to address the pick-up and the delivery of containers [1, 2, 3, 4, 5]. These formulations assume, however, that each truck carries one container only, which may lead to highly suboptimal decisions in our setting. Moreover, the combined vehicle and container routing issue is very seldom addressed. We propose an optimization model to address this gap.

2   The Case Study

Consider a shipping company operating a fleet of trucks based at a port location to serve a potentially different set of customers each day. There are two main types of transportation requests: the delivery of loaded containers from the port to importers (deliveries), and the shipment of loaded containers from exporters to the port (pick ups). Notice that the total number of pick up and delivery requests is generally different. In import-dominant regions the number of containers delivered to importers is larger than the number of containers requested for exports and, thus, empty containers must be moved from importers to the port for future transportation opportunities. The opposite activity must be performed in export-dominant regions, where empty containers must be moved from the port to exporters in order to meet all transportation requests.

Due to the constrained customer facilities, containers stay on trucks during loading and unloading operations. As a result, drivers are requested to wait for the conclusion of these operations before moving to other customers. When importers and exporters are located close together, the direct allocation of empty containers from an importer to an exporter may be a valuable opportunity to reduce the total travelled distance. In the current setting, all containers returned by importers can be used to meet the requests of exporters. The fleet of trucks is heterogeneous, each truck can serve any customer, and routing costs depend both on the distance between customers and on the vehicles performing the distribution. For example, some trucks can carry more than one container, but they generate higher operating costs than trucks carrying one container only.

Containers must be used and moved by trucks to serve all customers. The objective is then to determine a set of trucking routes, starting and ending at the port, of minimum total cost, such that all importer and exporter requests are served, a minimum number of containers are used, all deliveries are performed before any pick up, and truck capacity restrictions are not violated.

3   Optimization Model

We consider a port p, a set I of importers, a set E of exporters, and a set K of different trucks, each with capacity u_k. Consider a direct graph $G = (N, A)$, where $N = \{p \cup I \cup E\}$ and the set of arcs A includes all possible ways to move containers between two nodes in N. Since importer nodes must precede exporter ones, we consider a set $\bar{A} \subseteq A$, which does not include arcs from exporters to importers. The set $\bar{A}$ is defined as $\bar{A} = A_1 \cup A_2$, where

$$A_1 = \{(i,j) \in A | i \in I \cup p, j \in N\}$$

$$A_2 = \{(i,j) \in A | i \in E, j \in E \cup p\}$$

The operation cost c_{ij}^k for truck $k \in K$ on arc $(i,j) \in \bar{A}$ is supposed to be nonnegative. Let h_{pj}^k be the nonnegative cost of loading a container on truck k in port p for shipment to node j. An

integer demand of $d_i \geq 0$ containers is associated with each customer $i \in I \cup E$.

The following decision variables are defined:

- x_{ij}^k: Routing selection variable equal to 1 if arc $(i,j) \in \bar{A}$ is traversed by truck $k \in K$, and 0 otherwise;

- y_{ij}^k: Integer variable representing the number of loaded containers moved along arc $(i,j) \in \bar{A}$ by truck $k \in K$;

- z_{ij}^k: Integer variable representing the number of empty containers moved along arc $(i,j) \in \bar{A}$ by truck $k \in K$.

The problem can be formulated as follows:

$$min \sum_{k \in K} \left[\sum_{(i,j) \in \bar{A}} c_{ij}^k x_{ij}^k + \sum_{j \in N} h_{pj}^k (y_{pj}^k + z_{pj}^k) \right] \tag{1}$$

s.t.

$$\sum_{k \in K} \sum_{j \in p \cup I} y_{ji}^k \geq d_i \qquad \forall i \in I \tag{2}$$

$$\sum_{k \in K} \sum_{l \in N} y_{il}^k = \sum_{k \in K} \sum_{j \in p \cup I} y_{ji}^k - d_i \qquad \forall i \in I \tag{3}$$

$$\sum_{k \in K} \sum_{l \in N} z_{il}^k = \sum_{k \in K} \sum_{j \in p \cup I} z_{ji}^k + d_i \qquad \forall i \in I \tag{4}$$

$$\sum_{k \in K} \sum_{j \in N} z_{ji}^k \geq d_i \qquad \forall i \in E \tag{5}$$

$$\sum_{k \in K} \sum_{l \in p \cup E} y_{il}^k = \sum_{k \in K} \sum_{j \in N} y_{ji}^k + d_i \qquad \forall i \in E \tag{6}$$

$$\sum_{k \in K} \sum_{l \in p \cup E} z_{il}^k = \sum_{k \in K} \sum_{j \in N} z_{ji}^k - d_i \qquad \forall i \in E \tag{7}$$

$$y_{ij}^k + z_{ij}^k \leq u_k x_{ij}^k \qquad \forall (i,j) \in \bar{A}, \forall k \in K \tag{8}$$

$$\sum_{j|(j,i) \in \bar{A}} x_{ji}^k - \sum_{l|(i,l) \in \bar{A}} x_{il}^k = 0 \qquad \forall i \in N, \forall k \in K \tag{9}$$

$$\sum_{k \in K} \sum_{i \in I \cup E} z_{ip}^k - \sum_{k \in K} \sum_{i \in I \cup E} z_{pi}^k = \sum_{i \in I} d_i - \sum_{i \in E} d_i \tag{10}$$

Container loading and truck operating costs are minimized in the objective function (1). Constraints (2), (3), and (4) concern the movement of containers to importers. According to constraints (2), each importer node must receive at least a number of loaded containers equal to its demand. When this constraint is not tight, a number of loaded containers is kept on trucks for delivery to other importers. Constraints (3) and (4) are the flow conservation constraints of loaded and empty containers respectively at each importer node. Constraints (5), (6), and (7) concern the allocation of containers to exporters. Constraints (5) ensure that each exporter node receives at

least a number of empty containers equal to its demand. When this constraint is not tight, a number of empty containers is kept on trucks to serve other exporters. Constraints (6) and (7) are the flow conservation constraints of loaded and empty containers, respectively, for each exporter node. Constraints (8) impose that the number of containers moved by each truck does not exceed its capacity. Constraints (9) are the flow conservation constraints for trucks at each node. Finally, constraints (10) represent the flow conservation of empty containers at port p.

4   Conclusion

A number of artificial instances have been effectively solved using CPLEX. The results on a number of real instances will be presented at the conference, together with comparisons to the decisions of the corresponding shipping company in terms of both decision time and total travelled distance.

References

[1] A. Caris and G.K. Janssens, "A Local search heuristic for the pre- and end-haulage of inter-modal container terminals", *Computers and Operations Research* 36, 2763-2772 (2009).

[2] A. Imai, E. Nishimura and J. Current, "Lagrangian relaxation-based Heuristic for the Vehicle Routing with Full Container Load", *European Journal of Operational Research* 176, 87-105 (2007).

[3] H. Jula, M. Dessouky, P. Ioannou and A. Chassiakos, "Container movement by trucks in metropolitan networks: modeling and optimization", *Transportation Research Part E: Logistics and Transportation Review* 41, 235-259 (2005).

[4] H. Jula, A. Chassiakos and P. Ioannous, "Port Dynamic Empty Container Reuse", *Transportation Research Part E: Logistics and Transportation Review* 42, 43-60 (2006).

[5] R. Namboothiri and A.L. Erera, "Planning local container drayage operations given a port access appointment system", *Transportation Research Part E: Logistics and Transportation Review* 44, 185-202 (2008).

Container storage assignment
in an automatic maritime terminal

Pierre DEJAX

Ecole des Mines de Nantes, IRCCyN SLP group

BP 20722, 44307 Nantes Cedex 3, France

corresponding author Email: Pierre.Dejax@emn.fr

Xiaoming SUN

School of Mechanical Engineering

Shanghaï Jiao Tong University, Shanghaï, China

Nathalie BOSTEL
University of Nantes, IRCCyN, SLP group

BP 420, 44606 Saint-Nazaire Cedex, France

Zhiqiang LU, Lifeng XI
School of Mechanical Engineering

Shanghaï Jiao Tong University, Shanghaï, China

1. Problem description and literature review

Planning the container allocation of storage in a maritime container terminal is linked to the handling efficiency of the terminal. This research is concerned with the planning of container allocation of storage in a new automatic handling system using optimization technologies. The objective of the research is to determine an optimized allocation and storage plan for arriving/departing containers in this new configuration of automatic container terminal in order to equilibrate the workload of handling equipments and thus achieve a high performance in the time necessary to execute the loading and unloading import/export operations of vessels over a planning horizon. This problem is recognized as a key factor of the global performance of a maritime terminal and there is a need for developing efficient methods for applications of industrial size. As a matter of fact, repositioning containers in the storage area is necessary and generates unproductive moves.

The system studied here (inspired from the technology proposed by Shanghai Zhenhua Port Machine Co. Ltd, applied to the Port of Shanghaï) is characterized by original features, leading to new constraints in the storage problem. The equipments can handle two 40' containers (or four 20') simultaneously and several types of moves are considered: Inbound (I/B) unloading containers (ULC) from imports or transit are transferred from the Quay Cranes (QC) to the storage area by elevated distribution vehicles transporting them in front of their storage bay; these containers are then taken

down by lifting cranes (LC) in front of each bay; then shuttle vehicles, moving at high speed, transport them in front of their storage place where rail mounted gantry cranes (RMGC) store them at their assigned position. They are then sent out of the terminal by trucks. On the other way round, the receiving (RCC) export outbound containers (O/B) arrive by trucks and are stored at their assigned place. These and the transit containers become loading containers (LDC) and are transported from the storage areas to the quay cranes by the handling equipments in the reverse way.

Decision problems at container terminals are comprehensively described by Iris and de Koster[1] and in Steenken et al. [4] which classify the main logistics processes and operations in container terminals and present a survey of methods for their optimization. For the storage space allocation problem, the main focus is to suggest a method of pre- allocating storage spaces for import and export containers so that the re-handle process could be minimized, thus maximum efficiency in the loading and unloading operations areachieved. The process of determining the storage locations can usually be decomposed into two stages: the space allocation stage and the stage of locating individual containers. Kim and Park [3] deal with the space allocation problem in the first stage for the export containers. A mixed integer model is developed and two heuristic methods are proposed. Kim et al. [2] focus on the stage of locating individual container by determining the storage location of an export container in a pre-assigned yard bay in a way that reduces the expected relocation movement during loading operations. Zhang et al. [5] solve the problem using a rolling-horizon approach, and decompose the problem on two levels: the allocation of the different types of containers to the blocks and then the determination of the exact locations. We adapt the method of Zhang et al. to our particular problem (possibility to handle 1 or 2 containers at a time, containers of 20' or 40', particular configuration of the terminal).

2.   General solution approach

A container terminal operates continuously. To be able to solve the problem, it is necessary to fix a planning period and generate the solution on a rolling horizon. We propose to solve the problem by a hierarchical approach in two stages:

(1) Global allocation plan: At this level, a gross allocation plan will be developed in order to determine a storage area for each incoming container from export customers or arriving vessels at imports. The main input for this stage includes the vessel berth allocation plan and yard storage initial status.

 (2) Detailed allocation plan: Considering the global allocation plan, a detailed storage plan for each container will be elaborated in order to maximize the efficiency of yard cranes operations under the constraints of satisfying the quay cranes operationx schedule, with the goal of minimizing the makespan under the constraints generated by the global plan.

We propose a mathematical optimization model for solving the first problem. For each container type, storage area and period of time for unloading and loading, integer decision variables refer to the number of inbound, transit or outbound containers that are stored in each block, or loaded or unloaded

by the handling equipments, as well as to the level of inventory at the each period of time. Binary variables refer to the location of lifting cranes (LC) for ascent or descent moves for each storage block and time period.

The goal of the problem is to dispatch the loading and discharging containers related to a vessel among the active, or reserved storage blocks of the terminal, in order to balance the workload of the handling equipments in each period of time. The modelled objective function is a nonlinear expression minimizing the weigthed sum for all time periods of the planning horizon and container types of two terms measuring the imbalance of the vessel related number of operations for containers in block i during period t. The weights are adjusted according to the relative importance of the vessel related containers within the total number of containers in the terminal. Several roups of linear constraints are introduced into the model to ensure the practical feasibility of the solution: container flow conservation constraints, constraints on inbound, outbound and transit containers handling, block density constraints, constraints on determination of the location of the lifting crane, and nteger conditions on allocation variables The model described above may be converted to a mixed integer linear programming problem using usual techniques.

The second problem (detailed allocation plan of the containers in each block) can be formulated as an allocation problem on a bipartite graph, to minimize a cost expressing the travel distance of the handling equipments between the quay side and the storage blocks.

3.   Experimentation

We have used the standard optimizer (Xpress-MP) to solve the problem on data sets provided by the Shanghai Zhenhua Port Machine Co. Ltd. We consider a system with 7 blocks in the storage area, 5 dispatching lines, 4 quay cranes, from 8 to 14 lifting cranes (LC), 2 RMGC by block. We define a planning horizon of three days, each day divided into six 4-hour periods. The containers distributions are obtained from data given by the Shanghai Port Statistic Committee.We study the variation of the weights for quayside workload and total workload, total number of handling container number, number of the lifting cranes. The assumption made at the design of the handling system is to use 14 LCs., for a total of 6000 handled containers during the planning period. The result shows that the variation of the weight for quayside workload and total workload, the variation of the total handling number during research period have no significative influence on the optimal objective value; by opposition, the variation of the container type has greatly influence on the objective function. Our results illustrates he balance achieved in the number of the containers between blocks during 180 days planning period. We have also studied he variation of the objective function (container imbalance betwwen blocks) in terms of the number of the lifting cranes.

4.   Conclusions

In this conference we present a new concept of fully automated maritime multi-type container terminal proposed and currently experimented at a prototype terminal od the Port of Shanghaï. In order to optimize the assignement of containers in the terminal and the movements of handling equipments, we have proposed an optimization model to determine the storage areas of the different types of terminals. The model aims at balancing the storage of containers among the storage blocks and limiting the routing of containers between quayside and storage areas, thus ensuring an efficient utilization of the handling equipments. This policy favors an efficient handling of the loading and unloading operations for a given vessel, thus minimizing the makespan. The nonlinear optimization model can be converted to a mixed integer programming model and solved by a commercial solver in combination with an allocation model. Experiments carried on real data within the framework of a rolling horizon procedure provide encouraging results in terms of logistics efficiency and computer running time. The model can be used for operational purposes but also at a strategic level to determine the optimal number of handling equipments to use. Further research will consist in developping a scheduling model for the movements of the different types of handling equipments, as well as s simulation model for the global validation of the procedures.

References

 [1] F.A. Iris, R. de Koster (2003), Transshipment of containers at a container terminal: an overview, *European Journal of Operational Research*, vol 147, 1, May, pp.1-16.

[2] K.H. Kim, Y.M. Park, K.R. Ryu (2000), Deriving decision rules to locate export containers in container yards. *European Journal of Operational Research* 124: 89-101.

[3] K.H. Kim, K.T. Park (2003), A note on a dynamic space-allocation method for outbound containers, *European Journal of Operational Research*, vol. 148, 1, July, pp.92-101.

[4] D. Steenken, S. Voß and R. Stahlbock (2004), Container terminal operation and operations research: a classification and literature review. *OR Spectrum* 26: 3-49.

[5]C. Zhang, J. Liu, Y.W. Wan, K. G. Murty, R. Linn (2003), Storage space allocation in container terminals, Transportation Research part B, Methodological, vol 37, Issue 10, December, pp. 883-903

User equilibrium under reference-dependent route choice in a two-link network

Paolo Delle Site

Dipartimento Idraulica Trasporti Strade

University of Rome "La Sapienza"

Email: paolo.dellesite@uniroma1.it

Francesco Filippi

Dipartimento Idraulica Trasporti Strade

University of Rome "La Sapienza"

1 Introduction

There is a large body of field and experimental evidence that choices are best explained by assuming that carriers of utility are not states but gains and losses relative to a reference point; in addition, losses are valued more heavily than gains, the so-called loss aversion. This has led Tversky and Kahenman to propose new theories of choice: prospect theory [5], which has evolved into cumulative prospect theory [9], which considers risky choices, and reference-dependent theory [8], which considers riskless choices.

Models of route choice are a component of network equilibrium models. Uncertainty in travel time differs from uncertainty in perception captured in conventional stochastic user equilibrium (SUE) since uncertainty on the supply side requires modelling the attitude of users towards risk. Network equilibrium models capturing uncertainty in travel time are found in [6] and in more recent papers which have considered the application of cumulative prospect theory [1, 3].

The riskless case considered in reference-dependent theory is also of interest for equilibrium problems. This theory considers riskless choices where alternatives are characterised by n attributes. In this sense it extends prospect theory which is restricted to consideration of one attribute only (travel time). Reference-dependent route choice models allow to consider loss aversion effects in the trade-off between travel time and money, which is of practical relevance in tolling policies. However, the extension of SUE to consideration of reference-dependent route choice, which hereafter we call RDSUE, has received less attention in the literature.

Given this context the paper addresses RDSUE with two aims. The first aim is theoretical and consists in deriving desirable properties for RDSUE. In addition to existence and uniqueness, the property of reflexivity is considered. The idea of reflexive equilibrium is taken from theoretical economics where it has been proposed for economic equilibria [7]. The second aim is practical and

consists in examining the implications for tolling policies of the consideration of loss aversion in the trade-off between travel time and money. The analysis is carried out for a two-link network.

2 SUE under reference-dependent route choice

We consider a network with two route/link alternatives. We assume that route choice is modelled according to the following hypotheses: (i) utility U depends on gains G and losses L defined relative to a reference point $R0$; (ii) utility is linear and steeper for losses than for gains; (iii) utility depends on two attributes: travel time T and money spent M. Thus we formulate the following random utility model (as in [4]):

$$
\begin{aligned}
U_1^{R0} &= \beta_{GT} \cdot GT_1^{R0} + \beta_{LT} \cdot LT_1^{R0} + \beta_{GM} \cdot GM_1^{R0} + \beta_{LM} \cdot LM_1^{R0} + \varepsilon_1 \\
U_2^{R0} &= \beta_{GT} \cdot GT_2^{R0} + \beta_{LT} \cdot LT_2^{R0} + \beta_{GM} \cdot GM_2^{R0} + \beta_{LM} \cdot LM_2^{R0} + \varepsilon_2 \\
GX &= \max(X_0 - X_i, 0) \\
LX &= \max(X_i - X_0, 0) \\
X &= T, M; \quad i = 1,2
\end{aligned}
\tag{1}
$$

where β are the coefficients and ε the disturbances. Loss aversion is satisfied if $|\beta_L| > |\beta_G|$. Under the usual iid Gumbel assumption on the disturbances we obtain an asymmetric version of multinomial logit. We next consider RDSUE, which is the version of SUE with reference-dependent route choice modelled with (1). It is assumed that users adopt as reference point the status quo. Given a change in supply, choices and network state change to RDSUE. When RDSUE is set, each user chooses the route with the highest utility defined relative to previous choice and network state.

A RDSUE is reflexive if, for each user, the link with maximum utility defined relative to previous choice and network state is also the link with maximum utility defined relative to current choice and network state. The property of reflexivity is of interest because, having assumed that users always adopt as reference point the status quo, i.e. the current choice, it means that nobody has convenience to change choice when the reference point is updated after the RDSUE is set.

Proposition. In a two-link network: (i) the RDSUE exists and is unique if the time-flow functions are continuous and separable strictly-increasing; (ii) the RDSUE is reflexive if the coefficients in (1) satisfy loss aversion for both travel time and money.

Proof. Existence and uniqueness follow from an application of fixed point theorems for conventional multi-class SUE [2]. For reflexivity consider, for a given RDSUE network state, the two sets:

$$
\Omega_i^{R0} = \left\{ (\varepsilon_1, \varepsilon_2) : U_i^{R0} > U_j^{R0} \right\}; \qquad \Omega_i^{Ri} = \left\{ (\varepsilon_1, \varepsilon_2) : U_i^{Ri} > U_j^{Ri} \right\}
$$

Reflexivity follows by proving that $\Omega_i^{Ri} \supseteq \Omega_i^{R0}$. □

3 Application: the town bypass case

A reference-dependent route choice model has been estimated based on data from a 2007 stated preference survey in Rome (Table 1). Statistical significance of loss aversion results from a t-test on the difference in absolute values of the gain and loss coefficients (10% significance level, one-tailed). Model specification has been tested in comparison with a basic logit without asymmetry; the null hypothesis that the basic model is the correct specification is rejected.

This route choice model has been used to calculate the RDSUE in a two-link network representing a town centre route and a bypass route. BPR time-flow functions derived empirically for similar routes are used (in hours): $T = 0.057[1 + (f/800)^{5.2}]$ and $T = 0.045[1 + 0.68(f/1230)^{4.6}]$.

In a do-nothing scenario there is only the town centre route. The intervention scenario consists in the construction of a bypass. The aim of the application is to assess tolling policies for the bypass. Two policies are compared. The first (P1) consists in opening the bypass and, at the same time, charging a toll of 1 € for its use. The second (P2) consists in a two-stage policy. In stage 1 the bypass is opened and nothing is charged. Only in stage 2 a toll of 1 € is charged on the bypass.

Given reference-dependence the equilibrium resulting from P1 is different from the one at the end of stage 2 of P2. Table 2 shows that P2 is, at the end, superior both in terms of toll revenues (which are higher than in P1) and of total travel time spent (which is lower). In P2 the number of users choosing the bypass at the end of stage 2 is higher than the number of users who would choose it if it were opened and simultaneously a toll were charged. The result is consequence of loss aversion embodied in the route choice model. The town-centre route is relatively less preferred in P2 because in stage 2 of P2 it offers a loss in time for users of both routes, while it offers a gain in time in P1. The analysis of the sensitivity to the degree of loss aversion in time, represented by the ratio $|\beta_L / \beta_G|$, shows that the higher this degree the more significant the relative superiority of P2 on P1.

4 Conclusion

The paper has investigated the theoretical properties of existence, uniqueness and reflexivity for RDSUE in a two-link network. Insights are provided on how loss aversion affects tolling policies in the town bypass case. Future research will extend the model here to networks of general topology and non-linear utility functions displaying diminishing sensitivity.

References

[1] E. Avineri, "The effect of reference point on stochastic network equilibrium", *Transportation Science* 40(4), 409-420 (2006).

[2] E. Cascetta, *Transportation Systems Engineering: Theory and Methods*, Springer (2001).

[3] R.D. Connors and A. Sumalee, "A network equilibrium model with travellers' perception of stochastic travel times", *Transportation Research Part B* 43(6), 614-624 (2009).

[4] S. Hess, J.M. Rose and D.A. Hensher, "Asymmetric preference formation in willingness to pay estimates in discrete choice models", *Transportation Research Part E* 44(5), 847-863 (2008).

[5] D. Kahneman and A. Tversky, "Prospect theory: an analysis of decision under risk", *Econometrica* 47(2), 263-291 (1979).

[6] P. Mirchandani and H. Soroush, "Generalized traffic equilibrium with probabilistic travel times and perceptions", *Transportation Science* 21(3), 133-152 (1987).

[7] A. Munro and R. Sugden, "On the theory of reference-dependent preferences", *Journal of Economic Behavior and Organization* 50(4), 407-428 (2003).

[8] A. Tversky and D. Kahneman, "Loss aversion in riskless choice: a reference-dependent model", *The Quarterly Journal of Economics* 106(4), 1039-1061 (1991).

[9] A. Tversky and D. Kahneman, "Advances in prospect theory: cumulative representation of uncertainty", *Journal of Risk and Uncertainty* 5(4), 297-323 (1992).

Table 1. Estimation results for route choice model

	coefficient	t-stat	t-stat for difference in absolute values
time gain (min)	0.10545	9.521	-1.5318
time loss	-0.12270	-9.827	
money gain (€)	1.25287	9.481	-3.302
money loss	-1.67346	-14.075	
summary statistics: 1068 observations; $\rho^2 adj$ =0.439			

Table 2. RDSUE results in the town bypass case

	town centre route		bypass route		time spent (veh-h)	toll revenues (€)
	flow (veh/h)	time (min)	flow (veh/h)	time (min)		
do-nothing scenario						
	1200	31.6	-	-	632	0
policy scenario P1: bypass opened with toll=1€						
	879	9.0	321	2.7	146	321
policy scenario P2: bypass opened without toll (stage 1) + toll=1€ on bypass (stage 2)						
stage 1	563	4.0	637	2.8	67	0
stage 2	867	8.6	333	2.7	139	333

Improving the prediction of the travel time by using the real-time floating car data

Krzysztof Dembczyński   **Przemysław Gaweł**

Andrzej Jaszkiewicz   **Wojciech Kotłowski**

Adam Szarecki

Institute of Computing Science, Poznań Univesity of Technology,

Piotrowo 2, 60-965 Poznań

31 October 2009

1   Introduction

The travel time estimation has always been an important part of intelligent transportation systems (ITS) research domain. An accurate real-time prediction of travel times can be a crucial part of the driver information or the traffic management system.

The travel time estimation based on the real-time floating car data is also especially important for personal car navigation or route guidance systems. The aim of these systems is to guide their users with an optimal route to a chosen destination. Such a system can help, for example, in avoiding the traffic jams. The further consequence can be a reduction of traffic congestion in general, which itself is related to lowering fuel costs and air pollution, etc..

The benefits mentioned above are not the hypothetical ones, since the personal navigation systems are becoming more popular, particularly those that use mobile phones equipped with built-in GPS. These kinds of devices can guide the drivers by using the real-time data from the traffic network.

Travel time estimation in the above context is the main concern of this paper.

2   Research purpose

Given a static black box prediction model based on historical data, the aim of the research is to improve the performance of this model by a dynamic counterpart utilizing real-time GPS floating car data generated by the users of the car navigation system. The system also needs to be simple

and efficient for easy practical implementation and deployment.

The predictions of the model are then to be used in an algorithm for searching the shortest path between any two points in the network. Therefore the prediction must be done separately for a very large number of short segments of the roads.

3 Related Work

The analyzed problem belongs to the field of travel time prediction based on data (as opposed to flow simulation methods or similar). Studies based on similar input exist in the literature [5], but our problem has a few specific features that make it different from most other research in the area.

The scope of the predication covers the entire road network. Some similar global approaches can be found [2], however, the majority of the methods focus on single paths [4, 5], freeways [3, 4], or consider some urban arterial roads only [5]. In these cases, information is usually provided by the loop detectors [4] or other stationary sensors [1].

The introduced model predicts the travel time for each of the segments of the road network. Similar approaches can be found [1], however, the prediction for longer paths is considered more often.

4 Applied methods

Formally, the problem can be defined as learning of a function $f(\mathbf{x})$ to be a good prediction of the unknown value y_{it}. In this case, y_{it} is the travel time on the i-th segment at the time t. Thus, the estimate of y_{it} is:

$$\hat{y}_{it} = f(\mathbf{x}),$$

where $\mathbf{x}$ is a set of features being travel times collected from the entire road network. These can be considered as time series for each of the segments. The goal is to fit $f(\mathbf{x})$ in order to reduce the square-error loss:

$$L(y_{it}, \hat{y}_{it}) = (y_{it} - \hat{y}_{it})^2.$$

We considered the Gaussian process model and the exponential smoothing applied independently to each of the road segments. One can extend this simple model by assuming dependencies between different segments. We applied two such extensions. The first one is based on the data, and relies on computing correlations. The second one is based on domain-knowledge — the travel times are smoothed by averaging over the segment constituting the longer paths of the main roads.

4.1 Gaussian process model and exponential smoothing

Gaussian process model and exponential smoothing belong to the field of stochastic process modeling and time series analysis. They try to predict the travel time on a road segment based on

previous values of the same type. Gaussian process model is based on the assumption that the time series is a Gaussian process with known covariance function, observed at the points of the series. The prediction is then made by calculating the expectation of the conditional distribution given the previous data, at the point for which value needs to be predicted. Exponential smoothing performs a prediction which is a weighted average of previous observations, with weights decaying at exponential rate with respect to the time difference between the prediction and observation times. Of the two methods, the exponential smoothing can be considered the simpler and computationally more efficient one, but potentially yielding inferior (less accurate) results.

4.2   Linear correlations

The linear correlation is used for modeling the relations between different segments. It becomes a measure of similarity of simultaneous (in a defined time window) travel time changes — e.g. showing a tendency for some segments to be congested at the same time. The time series methods are then extended by using observations from the most correlated segments.

4.3   Smoothing by road segments aggregation

Smoothing by road segments aggregation uses the expert rules (based on map structure) to determine longer parts of main roads that are used to smooth the input data — vehicles passing the path are treated as having a constant velocity along the segments. The above method should be equivalent to a prediction based on longer road parts. This can lead to a slightly more complicated routing process, but potentially reduces the prediction noise.

5   Experiments

5.1   Data and methodology

The data was delivered by a Polish company, *NaviExpert*, that provides a commercial on-line navigation system. We obtained the *GPS floating car data* that had already been map-matched, i.e. it had a form of velocity and event time bound to a passage of a specific road segment in a given direction. The data was also quite sparse and unevenly distributed among time and space. It covered two large Polish cities with broad surroundings over a few months.

5.2   Results

In the first experiment, the Gaussian process model have shown a slight improvement over the black box model. The second analysis has shown that the results obtained by the exponential smoothing were virtually indistinguishable from the previous model.

In the next experiment, a simulation (through selection) of more dense input data was conducted and a large improvement was observed.

The further analysis consisted of applying a smoothing filter to the data, which was expected to be an equivalent of a prediction model based on longer parts on main roads (defined through expert rules). Again, a large improvement was discovered.

Lastly, calculating linear correlations between segments and including them in the exponential smoothing also yielded an improvement over the base time series method.

6   Conclusions

The main conclusion arising from the obtained results can be summarized in the following points:

- it is possible to create a simple, yet meaningful system for improving the travel time prediction covering the whole roads network, using possibly sparse floating car data,
- the exponential smoothing is a good and fast substitute of the Gaussian process model for the above purpose,
- the quality of the time prediction depends strongly on the density of the available data — denser input yields a better prediction,
- where possible — prediction on longer paths (as opposed to short road segments) may be desirable, due to less noise and better prediction quality,
- the linear correlations that indicate the dependencies between different segments of the road network, can also be used for improving the prediction.

References

[1] A. Hiramatsu, K. Nose, K. Tenmoku, T. Morita. Prediction of travel time in urban district based on state equation. *Electronics and Communications in Japan*, 92(7):1–11, 2009.

[2] W.H. Lee, S.S. Tseng, S.H. Tsai. A knowledge based real-time travel time prediction system for urban network. *Expert Systems With Applications*, 36(3P1):4239–4247, 2009.

[3] JWC Van Lint. Reliable real-time framework for short-term freeway travel time prediction. *Journal of Transportation Engineering*, 132:921, 2006.

[4] JWC Van Lint, SP Hoogendoorn, HJ Van Zuylen. Accurate freeway travel time prediction with state-space neural networks under missing data. *Transportation Research Part C*, 13(5-6):347–369, 2005.

[5] J.S. Yang. A Nonlinear State Space Approach to Arterial Travel Time Prediction. *Minnesota Department of Transportation, Office of Research Services*, 2006.

Large neighborhood search heuristics for propane delivery

Guy Desaulniers, Eric Prescott-Gagnon

Department of Mathematics and Industrial Engineering and GERAD

Ecole Polytechnique, Montréal, Canada

Email: Guy.Desaulniers@gerad.ca

Louis-Martin Rousseau

Department of Mathematics and Industrial Engineering and CIRRELT

Ecole Polytechnique, Montréal, Canada

1   Problem statement

Propane distributors are faced on a daily basis with the planning problem of determining the routes of their vehicles for the next day T such that their customers always have propane at hand (customer inventories are vendor-managed). These routes must satisfy various operational constraints, including customer time windows and predetermined driver schedules. We assume a single product and a homogeneous set of capacitated vehicles distributed among several depots. En-route replenishment at different locations is possible. Customers are divided into two categories: mandatory customers that must be visited on day T to avoid stock-out on day $T + 1$ and optional customers (no stock-out forecasted before day $T + 2$) that can also be visited on day T to save the marginal cost of visiting them later. The delivery policy is to fill up the customer demand at each visit, which is assumed to be known with high accuracy.

This propane delivery problem can be formally stated as follows. Given a set $\mathcal{K}$ of depots, sets $\mathcal{D}^k$ and $\mathcal{V}^k$ of driver work shifts and vehicles (all identical with a fixed capacity) for each depot k, a set of replenishment stations, a set $\mathcal{M}$ of mandatory customers with known demands and time windows, and a set $\mathcal{O}$ of optional customers with known demands, time windows and marginal cost savings (s_i) for servicing them (equal to estimates of the marginal costs for servicing them later), the problem consists of building feasible vehicle routes such that they can be assigned to the work shifts (at most one per driver), all mandatory customers are serviced, and total net costs (costs minus savings) are minimized. A route is composed of a sequence of customers interspersed

by visits to replenishment stations. It is feasible if it can be assigned to a shift, it starts and ends at the corresponding depot, and it satisfies the visited customers' time windows as well as the vehicle capacity between the replenishments. The costs include travel costs, driver fixed costs that depend on the shift length, and vehicle fixed costs that is charged once for the day, not for each route assigned to the vehicle.

In the literature (see the survey of Dror [1]), this problem has been treated as a stochastic inventory routing problem where the delivery date for each customer as well as the vehicle routes for each day of a planning horizon must be determined. Such a stochastic approach is necessary when demand forecasts are not accurate. Nowadays, forecasting tools have significantly improved and we can assume highly accurate forecasts, yielding deterministic demands. Such demands were considered in early works. In particular, Dror *et al.* [2] proposed two generalized assignment models (customers are assigned to delivery dates and vehicles or only to delivery dates) yielding two different algorithmic decompositions. Our model is similar to their first model but it considers a single delivery date and more complex operational constraints. Furthermore, our solution approach is integrated (that is, not decomposed into two separate steps) and relies on state-of-the-art methodologies.

2 Mathematical model

The propane delivery problem can be modeled as a set partitioning problem with side constraints that relies on the following sets and parameters: $\mathcal{R}^d$ is the set of all feasible routes for driver d; c_r^d is the cost of route r for driver d, including the driver fixed cost; a_{ri} is a binary parameter equal to 1 if route r visits customer i and 0 otherwise; $\mathcal{H}^k$ is the set of times at which the number of vehicles used can vary at depot k (that is, the shift start and end times); ℓ_h^d is a binary parameter equal to 1 if the shift of driver d includes time h; and f is the fixed cost for using a vehicle.

The model uses the following three types of variables: Y_r^d is a binary variable that takes value 1 if route r is assigned to driver d and 0 otherwise; E_i is a binary slack variable that takes value 1 if optional customer i is not serviced and 0 otherwise; V^k is an integer variable indicating the number of vehicles used at depot k.

The proposed set partitioning type model is as follows:

$$\text{Minimize} \qquad \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{D}^k} \sum_{r \in \mathcal{R}^d} c_r^d Y_r^d + f \sum_{k \in \mathcal{K}} V^k + \sum_{i \in \mathcal{O}} s_i E_i \tag{1}$$

$$\text{subject to:} \qquad \sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{D}^k} \sum_{r \in \mathcal{R}^d} a_{ri} Y_r^d = 1, \qquad \forall\, i \in \mathcal{M} \tag{2}$$

$$\sum_{k \in \mathcal{K}} \sum_{d \in \mathcal{D}^k} \sum_{r \in \mathcal{R}^d} a_{ri} Y_r^d + E_i = 1, \qquad \forall\, i \in \mathcal{O} \tag{3}$$

$$\sum_{r \in \mathcal{R}^d} Y_r^d \leq 1, \qquad \forall\, k \in \mathcal{K}, d \in \mathcal{D}^k \tag{4}$$

$$\sum_{d \in \mathcal{D}^k} \sum_{r \in \mathcal{R}^d} \ell_h^d Y_r^d \leq V^k, \qquad \forall\, k \in \mathcal{K}, h \in \mathcal{H}^k \tag{5}$$

$$Y_r^d \text{ binary}, \qquad \forall\, k \in \mathcal{K}, d \in \mathcal{D}^k, r \in \mathcal{R}^d. \tag{6}$$

The objective function (1) aims at minimizing the total net costs, that is, the total route costs (which include travel costs and driver fixed costs), the vehicle fixed costs, and the optional customer savings not realized. Set partitioning constraints (2) and (3) guarantee that each mandatory customer is visited once and each optional customer at most once. Constraints (4) limit to one the number of routes assigned to each driver. Finally, constraints (5) allow to count the number of vehicles used. Note that vehicle availability constraints are not required because the driver shifts were designed to ensure their satisfaction.

3 Large neighborhood search heuristics

In practice, large-sized instances involve up to 30 drivers and 1000 customers, that is, a route visits on average more than 30 customers and two or three replenishment stations. For these instances, model (1)–(6) contains a huge number of variables. To overcome this difficulty, a branch-and-price heuristic can be used. Furthermore, as introduced in Prescott-Gagnon *et al.* [4] for the vehicle routing problem with time windows, we embed such a branch-and-price heuristic into a large neighborhood search (LNS) framework.

LNS is an iterative method that starts from an initial solution. At each iteration, it first defines a neighborhood by applying a destruction operator that destroys parts of the current solution and then uses a construction operator to explore this neighborhood and, hopefully, find an improved solution. In our case, the initial solution is built using a greedy algorithm and the method stops once a predetermined number of iterations is reached. To ensure diversification during the search, four destruction procedures (operators) are available at each iteration. Each procedure releases iteratively customers from their current route and stops when a fixed number of customers is selected. They differ by their selection strategy that is based on customer proximity, the detour yielded by each customer, the visiting time of each customer, or randomness. At each iteration of the LNS method, the destruction operator is selected using a roulette wheel procedure that favors the operators that yielded the best results in the previous iterations. The construction operator is a branch-and-price heuristic. Branch-and-price consists of a column generation method that computes lower bounds at the nodes of a branch-and-bound tree. For the propane delivery problem, the column generation subproblems (one for each driver) are NP-hard elementary shortest path problems with resource constraints (see Irnich and Desaulniers [3]). To solve them rapidly,

we use a tabu search algorithm as in Prescott-Gagnon *et al.* [4]. The search tree is limited to a single branch. One decision is imposed at each node: the Y_r^d variable with the largest fractional value is set to one. This overall heuristic is denoted LNS-BP for LNS with branch-and-price.

For comparison purposes, we also developed another LNS method that is identical to LNS-BP, except that it relies on a completely different construction operator, namely, a tabu search algorithm. This tabu search algorithm relies on various move types: move a customer from one route to another, insert or remove an optional customer, insert or remove a replenishment, change the location of a replenishment, and, for diversification, exchange the routes of two drivers. This algorithm allows intermediate infeasible solutions with respect to customer time windows and vehicle capacity. It stops after a predefined number of iterations, allowing to control the time spent in each LNS iteration. This second LNS heuristic is denoted LNS-Tabu.

4   Computational experiments

Results of computational experiments obtained by both LNS-BP and LNS-Tabu heuristics on medium- to large-sized instances (with up to 600 customers) derived from real-world data sets will be reported at the conference. For the moment, preliminary tests on randomly generated instances show that the LNS-Tabu heuristic can produce good-quality solutions in less than one hour of computational time for instances involving up to 450 customers.

References

[1] M. Dror, "Routing propane deliveries", in *Logistics systems: Design and optimization*, A. Langevin and D. Riopel (eds), 299-322, Springer, New York, 2005.

[2] M. Dror, M. Ball, and B. Golden, "A computational comparison of algorithms for the inventory routing problem", *Annals of Operations Research* 4, 3-23 (1985/86).

[3] S. Irnich and G. Desaulniers, "Shortest path problems with resource constraints", in *Column generation*, G. Desaulniers, J. Desrosiers, and M.M. Solomon (eds), 33-65, Springer, New York, 2005.

[4] E. Prescott-Gagnon, G. Desaulniers, and L.-M. Rousseau, "A branch-and-price-based large neighborhood search algorithm for the vehicle routing problem with time windows", *Networks*, doi: 10.1002/net.20332, (2009).

Advances in Linear Programming
and Column Generation

Jacques Desrosiers
Department of Management Sciences
HEC Montréal and GERAD, Canada
Jacques.Desrosiers@hec.ca

1 Introduction

Dantzig-Wolfe decomposition and column generation embedded into a branch-and-bound scheme are established as leading solution methodologies for many large-scale integer programming problems, especially in the areas of vehicle routing and crew scheduling applications. The proposed talk is an extension of two recent researches in the solution of set partitioning problems by column generation and of linear programs by the primal simplex method.

2 Constraints Aggregation

In the past few years, the concept of *constraints aggregation for set partitioning problems* has been developed [1], [2], [3]. This type of formulation appears in many vehicle routing and crew scheduling applications where each row represents a task to cover while each column provides a feasible vehicle itinerary or crew schedule. Binary parameter a_{ij} of a column takes value 1 if task i is covered by itinerary or schedule j, 0 otherwise. In some applications, it is natural that several tasks be aggregated to form a single task. For example, a pilot usually follows its aircraft so that, if we already know the aircraft itineraries, some of the consecutive flight legs assigned to an aircraft can tentatively be grouped together, hence reducing the number of tasks to be covered by the pilots in the set partitioning formulation of the airline crew scheduling problem.

The reduction of the number of constraints for a set partitioning formulation of size $M \times N$ is done by removing the $M\text{-}m$ degenerate (zero value) variables of a basic solution. The remaining m non-zero basic variables are represented by m columns for which $M\text{-}m$ rows are removed. An $m \times m$ reduced basis is used for the rest of the optimization process where the size m of the basis is dynamically updated. The implementation in GENCOL software system allows reducing by a factor of 50 to 100 the solution times

for problems with 2000 constraints and 50% to 60% degenerate basic variables. The CPU reduction is a combination of many factors: smaller master problem, reduced number of degenerate pivots, smaller sub-problems (the task aggregation is also done at that level), smaller number of column generation iterations, less fractional linear programming relaxation, and smaller branch-and-bound tree.

3 Improved Primal Simplex

The second research area is the generalization of the above mentioned method to linear programs. Given a degenerate solution to a linear program, we first identify a reduced basis and a reduced problem. The IPS method (*Improved Primal Simplex* method) contains two main ideas. (1) A variable is compatible with the current reduced basis and selected to be part of the current reduced problem if the objective value strictly decreases when this variable enters the basis: this is a non-degenerate pivot. A criterion for identifying the compatible variables during the pricing process has been developed for linear programs. (2) When the reduced cost of all the compatible variables is zero or greater than a specified threshold, a complementary pricing problem is solved to select a convex combination of non-compatible variables such that the objective value strictly decreases when they all enter into the current reduced problem. In this case, the size of the reduced problem is modified in terms of the number of rows.

Although the comparisons with CPLEX were done in a very simple manner, that is, an external loop choosing the entering variables, CPU reduction factors of 4 and 12 were obtained for the linear programming solution of bus driver scheduling and aircraft routing problems, respectively [4, 5]. Taking also advantage of degenerate variables not only at zero but at their upper bounds, CPU times were reduced by a factor of 32 compared to CPLEX on fleet assignment problems with bounded variables (multi-commodity flow problems with 5000 constraints and 25 000 variables) for which an upper bound of 1 on arc flow variables was explicitly imposed [6].

Constraint aggregation for set partitioning problems and the improved primal simplex method can be compared in the following way. In the first method, when the degenerate variables are removed from the basis, several rows become identical and only one representative row is kept for each set of identical rows, therefore row-reducing the size of the restricted master problem. In the second method for linear programs, when the degenerate variables are removed from the basis, several rows become linearly dependant such that we only keep a set of independent ones to construct the current reduced problem. Geometrically speaking, for both methods, at a given degenerate extreme point of the linear problem, all but one basis is kept and the next iteration moves to a different extreme point.

4 Improved Column Generation

Since the solution by column generation of a Dantzig-Wolfe reformulation of a compact formulation is essentially an adaptation of the primal simplex method, we propose an adaptation of IPS and its upper bounded version to this decomposition scheme. This has already been partly done in the above mentioned constraints aggregation method for set partitioning problems and it can be generalized to linear and integer linear programs solved by column generation as follows.

The classical restricted master problem of the column generation method does not contain all the variables and these are generated as needed by the solution of a pricing sub-problem. In this presentation, we show how to use a dynamic *row-reduced* restricted master problem to solve the linear relaxation of the master problem. Two types of sub-problems are needed: one to generate columns compatible with the current reduced basis and one to generate columns that are not compatible with that reduced basis.

The first type of sub-problem, i.e., the sub-problem generating columns compatible with the current reduced basis, is the original sub-problem augmented with a set of linear constraints imposing compatibility requirements. This pricing sub-problem selects compatible columns as long as they are useful for non-degenerate pivots in the row-reduced restricted master problem. When the reduced cost of all the compatible columns is zero or greater than a specified threshold, a complementary pricing sub-problem must be solved to select a convex combination of non-compatible columns such that the objective value of the master problem strictly decreases when they all enter into the current reduced problem. In this case, the row-size of the reduced master problem is dynamically modified.

5 Conclusion

The classical column generation method works with a restricted master problem, that is, a subset of the columns. The improved column generation method works with a reduced restricted master problem, that is, it additionally reduces the size of the current basis. As already shown for the above mentioned constraints aggregation on set partitioning problems where CPU times are reduced by a factor of 50 to 100 for some vehicle routing and crew scheduling applications, this additional row reduction of the master problem should have a large impact on the solution time of degenerate linear and integer programs solved by column generation.

References

[1] I. Elhallaoui, D. Villeneuve, F. Soumis and G. Desaulniers, "Dynamic Aggregation of Set Partitioning Constraints in Column Generation", *Operations Research*, 53(4), 632-645 (2005).

[2] I. Elhallaou, A. Metrane, F. Soumis and G. Desaulniers, "Multi-phase Dynamic Constraint Aggregation for Set Partitioning Type Problems", *Mathematical Programming B*, (2008).

[3] I. Elhallaoui, G. Desaulniers, A. Metrane and F. Soumis, "Bi-dynamic Constraint Aggregation and Subproblem Reduction", *Computers & Operations Research*, 35(5), 1713-1724 (2008).

[4] I. Elhallaoui, A. Metrane, F. Soumis and G. Desaulniers, "An Improved Primal Simplex Algorithm for Degenerate Linear Programs". 2008.

[5] V. Raymond, F. Soumis and D. Orban, "A New Version of the Improved Primal Simplex for Degenerate Linear Program". 2008.

[6] V. Raymond and F. Soumis, "A Generalisation of the Improved Primal Simplex for Degenerate Linear Program. 2008.

Delay Management with Passenger Re-Routing: Solving Practical Instances

Twan Dollevoet

Econometric Institute and ECOPT

Erasmus University Rotterdam

P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

Email: dollevoet@ese.eur.nl

Dennis Huisman

Econometric Institute and ECOPT

Erasmus University Rotterdam

&

Department of Logistics, Netherlands Railways

1 Introduction

The punctuality, that measures the percentage of trains that arrive within 3 minutes after their planned arrival time, is currently the main quality indicator for the Dutch railway system. There is one important aspect of railway operations that is not taken into account by this quality measure: the passengers who have to transfer at intermediate stations. When a passenger has to transfer from one train to another, even a small delay of the feeder train can make it impossible for the passenger to catch the connecting one. When a connection is missed, the passenger has to wait for the next train in the same direction. This may increase the travel time of that passenger severely.

The unreliability of the connections is one of the major complaints about the Dutch railway system. Since the privatization of public transport in Europe, customer satisfaction has become an important objective for the railway operators. Increasing the number of maintained connections is one way to improve the service that an operator delivers to its passengers. Netherlands Railways, the largest operator in the Netherlands, has recently introduced the quality indicator *passenger punctuality*, which is defined as the percentage of passengers who reach their destination within 3 minutes after the planned time. If the passenger punctuality can be increased, the customer satisfaction will be higher as well.

Delay management is the field in railway operation control that deals with connections between trains. When one train has a small delay, it might be beneficial for the transferring passengers to delay another train slightly as well, to allow the passengers to transfer to the second train. However, by delaying the connecting train, the travel time for the passengers already in that train will be enlarged. Furthermore, there might be connections from that train to others at subsequent stations that have to be considered. This makes delay management a complex problem, especially in dense railway networks such as in the Netherlands. Delaying trains to maintain connections for passengers is currently done manually by dispatchers of the infrastructure manager and the operators. We feel that decision support systems can be helpful for them to reduce the nuisance in case of delays.

As most European railway operators, Netherlands Railways operates a *cyclic* timetable. In a cyclic timetable, the operations are repeated after every cycle time T. The first approaches to model the delay management problem assume that the delay for a passenger who misses a connection equals the cycle time: if a connection is dropped, passengers have to wait for the next train on the same line (see [2] and [3]). In practice, this assumption will not be valid. There can be many train lines between two stations, and therefore there can be a train on a different line that goes in the same direction. To model the behavior of the passengers more realistically, a model is proposed that takes the routes of the passengers into account explicitly (see [1]). In particular, the model allows the passengers to adjust their route in case of delays. Although some numerical results are presented, it turns out that the size of the integer program is far too large for practical applications.

In the current research, we present an alternative formulation for the delay management problem with passenger re-routing. By limiting the re-routing possibilities in case of delays, we are able to solve practical problems much faster. Furthermore, we will exploit the structure of the problem to develop faster algorithms that solve the delay management problem exactly. Finally, we will compare the performance of the various solution methods.

The remainder of this document is structured as follows. In Section 2 we will briefly discuss the original model for the delay management problem and its extension to deal with passenger re-routing. In Section 3 we discuss an alternative formulation and some techniques to speed up the solution process.

2 Delay Management Models

We will now first present the original delay management model. Then we will discuss its extension that incorporates passenger re-routing.

The original delay management problem can be modeled as an event-activity network $\mathcal{N} =$

$(\mathcal{E}, \mathcal{A})$, in which the nodes represent events that have to be scheduled and the arcs correspond to activities that connect these events (see [3]). The events that are to be scheduled are the departures and arrivals of the trains. There are two types of activities. First, driving from one station to the next and allowing the passengers to get on and off the train are *operational* activities. These operational activities put physical restrictions on the timetable. Second, passengers who want to transfer from one train to another are represented as *transfer* activities. Note that these activities can be dropped in case of delays. If a connection is maintained, the departure of the connecting train should be scheduled after the arrival of the feeder train. The main question in delay management is which connections to maintain. If a connection is maintained, the connecting train should wait for the feeder train. This introduces a delay for all passengers already in the connecting train. On the other hand, if the connection is not maintained, all connecting passengers have to wait for the next train. It is assumed in the original models that the delay for such passengers equals the cycle time T.

To model the behavior of the passengers more realistically, one should incorporate the possibility to adjust the route of a passenger in case of delays. We will briefly describe how to take passenger re-routing into account. A route corresponds to a path in the event-activity network. To allow passenger re-routing, we add for every group of passengers with the same origin and destination a source and a sink to the event-activity network. We refer to the sources as origin events in the event-activity network and to the sinks as destination events. These origin and destination events are then connected to the departures and arrival events of trains, respectively, at the corresponding stations. A route through the railway network now corresponds to a path in the event-activity network. Finding the shortest path from an origin to a destination event for each group of passengers is then part of the optimization. Given the path for a group of passengers, the exact delay for those passengers can be found.

To model the shortest path problems for the passengers in the integer program, a binary variable has to be introduced for every arc in the event-activity network and every passenger group. For practical instances, this leads to a huge number of binary variables. Therefore, solving the problem with standard solvers is impossible. Especially if the approach is to be implemented in practice, a solution should be available on a very short notice, because the dispatchers should be able to react to delays almost instantaneously.

3   Speeding up the solution process

The extended model for the delay management problem that allows for passenger re-routing is very flexible: every path through the event-activity network serves as a possible route for the passengers. This means that passengers can even change their route through the *railway network*.

Indeed, in some cases the optimal routes travel via different stations than the original ones. To use these optimal routes, passengers should have a global knowledge of the timetable and of transfer times at other stations. More importantly, it is crucial to forecast the arrival times of the trains at later stations correctly. In reality, this information is not available. Instead, passengers will leave the feeder train, note that the connecting train has left already and take the first train in the same direction. This observation can be used to reduce the size of the integer program. Instead of including all possible routes for the passengers, we will only allow for the most reasonable ones.

The previous approach restricts the possible routes for the passengers statically . One can also determine the possible routes dynamically using column generation. In this approach, one starts with a small set of possible routes for each passenger group. During the optimization process, alternative routes for passengers with large delays have to be found, and these possible routes should be added. In this way, the program can be kept small on one hand, but on the other hand a large number of routes can be considered for passengers with large delays.

We have implemented the above approaches to solve the delay management problem and applied them to solve practical instances that are obtained from Netherlands Railways. We will compare both the quality and the running times of these approaches, to single out which method finds the best solution within the short amount of time that is available in the dispatching process.

References

[1] T. Dollevoet, D. Huisman, M. Schmidt, and A. Schöbel. Delay Management with Re-Routing of Passengers. In *ATMOS 2009 - 9th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, Dagstuhl, Germany, 2009. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, Germany.

[2] A. Schöbel. A Model for the Delay Management Problem based on Mixed-Integer Programming. *Electronic Notes in Theoretical Computer Science*, 50(1), 2001.

[3] A. Schöbel. Integer Programming approaches for solving the delay management problem. In *Algorithmic Methods for Railway Optimization*, number 4359 in Lecture Notes in Computer Science, pages 145–170. Springer, 2007.

Sensitivity Analysis Method for Trip Mode Choice Behavior of Expo 2010 Shanghai

Du Yuchuan

Key Laboratory of Road and Traffic Engineering of the Ministry of Education

Tongji University

Jiang Shengchuan

Key Laboratory of Road and Traffic Engineering of the Ministry of Education

Tongji University

Sun Lijun

Key Laboratory of Road and Traffic Engineering of the Ministry of Education

Tongji University

Extended Abstract

It is reported that Expo 2010 Shanghai will attract 70 millions visitors during the whole 184 days duration, namely 400,000 visitors daily and 800,000 visitors in peak day[1]. Considering the location of the Expo Park is in the city center of Shanghai, this mega-event is regarded by many experts as one of the great, world-wide transport and logistics challenges. The large number of visitors expected to be carried, combined with the congested urban road network and limited parking spaces, will make it difficult for individual transport to be used during the Expo; as such, high rates of utilization of public transport will be necessary. Hence, exploring the trip mode choice behavior of Expo visitors is the keystone for traffic planning of Expo 2010 Shanghai, especially for efficiency assessment of different traffic policies.

Over the past few decades, research interest on the link between potential travel choice behavior and contribution of the independent variables has blossomed and a substantial amount of

research on this field falls into the category of stated preference (SP) techniques [2,3,4]. More recently, discrete choice models based on SP methods have become popular among academics, governments and consulting companies to explore many aspects of transportation, including mode choice behaviour under different traffic management policies, urban forms, levels of service, prices and so on [5,6,7,8,9,10]. Considering the above features of tourists, the traditional analysis models for a certain homogeneous travellers are not suitable for the trip choice behaviour of tourists. There are many strategies have been proposed for distinguishing among groups of travellers, including ones based on attribute cutoffs, clusters of travel attitudes, motivations or preferences, behavioural repertoires for different activities and hierarchical information integration [11,12,13,14,15,16].

The objective of this paper is to investigate the differences in trip mode choice behaviour among potential Expo visitor groups to support the effect analysis of traffic management policies for Expo 2010 Shanghai. Because of the differences in Expo visitor departure areas and the influence of the various types of departure areas on trip mode behaviour, a two-stage gradual stated preference survey method was used to develop variable multinomial logit models for local and out-of-town Expo visitors. Sensibility analysis method based on point-elasticity and cross-elasticity was applied to acquier the efficient transport management policies and control measures for reducing the proportion of private transport modes.

Because of the range of visitors to Expo 2010 Shanghai and their various attributes, it will obviously be difficult to obtain satisfactory results taking all Expo visitors as one group for analysis and modelling. In addition, as the World Expo has not previously been held in China, there is no reference to aid in the understanding or prediction of visitor trip mode choice behaviour over the duration of this mega-event. Given such a backdrop, this paper developed a two-stage gradual stated preference survey method for the in-depth study of Expo visitor trip mode choice behaviour. Stage 1 Survey is of multi-scenario-comparison choice behavior, mainly aiming at development of trip choice model for Expo visitors, in depth studying influence of parking costs, walking time and travel time on trip mode choice, and supporting forecast of traffic demands of each means of transport to Expo and analysis of various traffic policies for the Expo, so as to promote effective shift from private-transport trip to public-transport trip. Stage 2 SP survey is designed to gain the results of travelers' choice of trip mode on the basis of the Stage 1 Survey's conclusion, by selecting characteristics variable and determining variable level in accordance with characteristics of visitors from different departure areas,

and under the situation of combination of different variable levels, so as to model choice behavior, study sharing ratio of various trip modes and time value of different groups, and determine influence of policy adjustment on trip mode share.

Based on the results of the Stage 1 survey, this paper considers those variables that significantly influence trip mode choice, including time, cost and departure area. The survey data reveal that visitor departure area greatly influences trip mode choice; however, it is difficult to model such influence as a quantifiable factor. This paper takes influence as a basis for model classification and standardises it in accordance with the respective source areas, namely, source areas are represented by fixed-effect dummy variables in the model. Four kinds of models were tested in this paper.

Based on the choice of the abovementioned variables, the multinomial logit model choice utility function for Expo trip mode choice was determined as

$$V_{in} = A_i + \beta_1 x_i + \beta_2 y_i + \beta_3 z_i \, , \tag{1}$$

where:

A_i — Constant for trip mode, $A_1 = 0$;

x_i — Walking time;

y_i — Travel time, including waiting and riding time;

z_i — Travel cost; and

β_i — Coefficients for variables x_i, y_i and z_i.

The trip mode choice set of this model is C = {i = 1 (Taxi); i = 2 (Subway); i = 3 (Expo shuttle bus); i = 4 (Private car)}.

The sensitivity analysis formulas for point-elasticity and cross-elasticity were deduced and shouwn as following:

$$E_{X_{ink}}^{P_{in}} = \frac{X_{ink}}{P_{in}} \cdot \frac{dP_{in}}{dX_{ink}} = \beta_k X_{ink}[1 - P_{in}] \tag{2}$$

$$E_{X_{ink}}^{P_{jn}} = \frac{X_{ink}}{P_{jn}} \cdot \frac{dP_{jn}}{dX_{ink}} = -\beta_k X_{ink} P_{in} \tag{3}$$

where:

X_k —External factors on trip mode choise behaviors, including walking time, travel time and travel cost;

$E_{X_{ink}}^{P_{in}}$ — Point elasticity of variable X_k on trip mode i;

$E_{X_{ink}}^{P_{jn}}$ — Cross-elasticity of variable X_k on trip mode j;

β_k — Coefficients for variables X_k; and

P_i — Share of trip mode i.

These sensitivity analysis formulas were applied to evaluate the effect of various influencing factors on Expo visitor trip choice result. Meanwhile, three traffic management policies: (a) establishing a restricted traffic zone, (b) increasing parking rates around Expo Park and decreasing those at the park and ride Expo shuttle bus transfer points outside the urban area of Shanghai, and (c) providing Expo shuttle bus priority lanes and giving these buses signal priority, are discussed to promote the switch from individual transport modeso public transport mode.

On the basis of the detailed analysis of abundant survey data and the in-depth exploration of external factors on trip mode choice behaviour of Expo 2010 Shanghai, we offer the following main conclusions.

1) For a case such as Expo 2010 Shanghai, which involves diverse visitor groups, a single-level SP survey is insufficient to separate the variables and obtain a standardised model. Considering differences in departure area trip chain characteristics, there versions multinomial logit models for local visitors, out-of-town one-day-trip visitors and out-of-town lodging visitors are developed in this paper.

2) The sensitivity analysis results show that that travel time, walking time and travel cost are all effective influencing factors but differ in utility among the various groups. Local visitors are more sensitive to walking time and total expenses, out-of-town one-day-trip visitors are more concerned about total travel time and out-of-town lodging visitors are highly sensitive to walking time and total travel time.

3) Establishing a restricted traffic zone, adjusting parking rates and giving Expo shuttle buses priority could attract some private transport users to switch to public transport, but the effect of these policies differs among the visitor groups and traffic modes. The implementation of the policies in

combination can effectively control the proportion of private transport use to within 10% but has a negative influence on subway use, which is an issue that needs to be addressed.

References

1. Yin Rui, Li Keping, Yu Jie, 2007. Traffic forecast for visitiors in World expo 2010 Shanghai arena. Journal of Tongji University (Natural Science) 35(8), 1053-1058

2. Davidson, J. D., 1973. Forecasting traffic on STOL. Operation Research Quarterly, 24(4), 561-569.

3. Rowley, G., and Wilson, S., 1975. The analysis of housing and travel preferences: a gaming approach, Environment and Planning, A-7A,171-177

4. Adler J. and Ben-Akiva M, 1976. Joint-choice model for frequency, destination, and travel mode for shopping trips. Transportation research record. 569,136-150.

5. Fowkey Tony and Preston John, 1991. NOVEL APPROACHES TO FORECASTING THE DEMAND FOR NEW LOCAL RAIL SERVICES. Transportation Research Part A, 25(4) 209-218.

6. Bhat CR, 1997. Work travel mode choice and number of non-work commute stops. Transportation Research Part B, 31(1) 41-54.

7. Bhat, CR, 2008. The multiple discrete-continuous extreme value (MDCEV) model: Role of utility function parameters, identification considerations, and model extensions. Transportation Research Part B 42(3), 274-303.

8. Johansson, M.V., Heldt, T., Johansson, P., 2006. The effects of attitudes and personality traits on mode choice. Transportation Research Part A 40 (6), 507‑525.

9. McMillan TE, 2007. The relative influence of urban form on a child's travel mode to school. Transportation Research Part A, 41(1) 69-79.

10. Lu Jin-Long, Peeta Srinivas, 2009. Analysis of the factors that influence the relationship between business air travel and videoconferencing. Transportation Research Part A 43 (8) 709-721

11. Swait, JD, 2001. A non-compensatory choice model incorporating attribute cutoffs. Transportation Research: Part B 35(10), 903-928.

12. Steg, L., 2005. Car use: lust and must. Instrumental, symbolic and affective motives for car use. Transportation Research Part A 39(2-3), 147-162.

13. Van Exel, N. J. A., de Graaf, G., Rietveld, P., 2005. Getting from A to B: operant approaches to travel decision making. Operant Subjectivity 27(4), 194-216.

14. Anable, J., 2005. Complacent car addicts or aspiring environmentalists? Identifying travel behaviour segments using attitude theory. Transport Policy 12, 65-78.

15. Tam, M. L., Lam, W. H. K., Lo, H. P., 2008. Modeling air passenger travel behavior on airport ground access mode choices. Transportmetrica 4(2), 135-153.

16. Molin, E. J. E., Timmermans, H. J. P., 2009. Hierarchical information integration experiments and integrated choice experiments. Transport Reviews 29(5), 635-655.

Robust optimization of bulk gas distribution

Hugues Dubedout

Nicoleta Neagu

Claude Delorme Research Center, Air Liquide

78354 Jouy en Josas - France

Hugues.Dubedout@Airliquide.com

Nicoleta.Neagu@Airliquide.com

Optimization models for transportation/distribution and supply chain problems are generally treated under *certainty assumption* whereas all the data about the problem is know with certitude prior to its solving. As a consequence, when uncertain events happen the 'optimized solutions' may become less optimal or even infeasible, which may induce extra costs. Moreover, large part of real world optimization problems are subject to uncertainties occurring in the problem data and parameters. Gas distribution problems at Air Liquide are particularly concerned by the presence of uncertainty in their data (e.g., unplanned plant outage, resources availability, and demand fluctuations).

The key objective of this research work is to increase the robustness of optimization solutions for bulk gas distribution relatively to uncertain events such as unexpected plant outages. Thus, in this work we investigate new optimization models and methods to build robust routing and scheduling for the distribution of gas in bulk. The optimized methods include in a proactive manner assumptions about unexpected plant outages while searching for solutions. According to some previous studies at Air Liquide the outages at production plants have an important impact on the distribution cost and thus, these unexpected events should not be neglected. The major final goal is to identify *robust* solutions which have a good trade off between reliability to plant outages and the induced extra cost.

Using robust solutions allows Air Liquide to improve its distribution in multiple ways:

- better performance by integrating risk: optimized solutions despite risk and uncertain events.

- reducing the environmental impact by decreasing carbon foot print.

- better working environment for employees by avoiding emergency situations.

- better quality of service for the clients.

Problem Statement

We consider the bulk gas distribution problem in a real life context. The considered bulk distribution problem consists of the following main elements: customers' orders, transport equipment availability, non periodical production and demand forecast, inventory levels (at both plants and customers locations), and driver and power unit availability. The bulk gas is produced at the production plants and the products are distributed from stocks of these plants to customers locations by means of vehicles starting form "bases" and routed to deliver the products to the customers. The customer delivery must be planned over several days to avoid breaking stock at customer and is based on a system of inventory management and customer forecasting models.

This problem is know in the literature as the Inventory Routing Problem with Vendor Management Inventory (IRP-VMI)[1]. The distribution is made either based on the forecast for the VMI customers or based on customers' order, and obeys several constraints, including geographical and temporal. It takes place in a multi period on a rolling time horizon of about two weeks. It is based on the quality of demand forecasts and stock availability of products at plants' stocks, but it appears under uncertainty due to plant outages. The problem is large scale in nature and needs to be treated under uncertainty assumptions related to plant inventories which are often affected by outages.

A study conducted since 2003 identified the plant outage uncertainty as one of the major planning disruption causes, increasing the global distribution cost by several millions of dollars each year in the united states alone. Creating more robust schedule often leads to an increase of the schedule cost. The goal of our research was to generate schedule with only a small increase in cost but with a high gain in robustness.

Robust Discrete Optimization Approach

The IRP problem model represents customers, sources and bases at their geographical positions. For a determined time horizon in the future, a forecasted consumption and/or known orders are given for each customer. A production profile is defined for each source. The model defines the variables for the construction of the scheduling shifts for the transportation of gas from sources to customer so as to avoid customer run-outs. The dependencies between the variables are defined through business and physical constraints. The constraints are relative to the feasibility of the schedules of operations in terms of travel time, customer opening hours, etc. The objective function of the model represents the minimization of the distribution costs while avoiding customers' runouts and missing orders.

The solution method we propose for solving the IRP problem under uncertainty assumptions for plant outages is based on the work of Kouvelis and Yu [4]. It uses a scenario based approach to generate a more robust schedule. We define a generic framework for using discrete robust optimization applied to large scale IRP under uncertainty. In developing the robust discrete optimization

framework we pursue the following main steps:

1. The first step define multiple future realization of the key parameters, so called scenarios. Based on statistic study of historic outage data, we are able to generate a set of scenarios representing the plausible plant outage possibilities.

2. The second step is to generate a set of feasible solutions to the distribution problem. The feasibility of each solution has to be independent of the parameters modified in the scenarios, so that each solution generated is feasible when applied to any of the scenarios. We generate the solutions using a classic local search algorithm with different parameters tuning, which allows us to obtain different solution with good objective function value. We proposed a new search strategy based on inventory criteria which can identify more reliable/robust solutions of the IRP problem.

3. Lastly we compute the cost of each solution applied to each scenario and use a classical min max regret and min max deviation evaluation method to compute the robustness of each solution. Then the solution with the best cost versus robustness trade off is selected.

We apply our approach to several real life test cases and show that the proposed approach manages to generate solutions that are better that the current solution in both and also more robust. We also propose a parallelization method to greatly reduce the computation time necessary to generate the solutions.

Conclusions

We proposed a framework for robust decision making under uncertainty for the logistic optimization of AIR LIQUIDE bulk distribution. We modified the model of the discrete IRP problem to take into account uncertainty aspects generated by plant outages. We then used a scenario based approach which allowed us to optimize the distribution regarding multiple possible future realization of the uncertainty variable. We proposed different methods to generate scenarios, each one of them performing differently. We implemented this framework regarding plant outage and managed to obtain good results on four real-life test cases. In two of the test cases we obtained solution with both a lower cost and a better reliability than the optimization tool currently used in production which ignores possible plant outages when generating routing and scheduling solutions.

References

[1] Bertazzi L., M. Savelsbergh et M.G. Speranza , "Inventory routing", *The vehicle Routing Problem : Latest Advances and New Challenges*, B. Golden and E. Wasil edts., Springer, 49-72 (2008).

[2] M. Savelsbergh, J-H. Song "An optimization algorithm for the inventory routing problem with continuous moves" *Computers & Operations Research* Volume 35, Issue 7, 2266-2282 (2008)

[3] T. Benoist, B. Estellon, F. Gardi, A. Jeanjean (2009). Recherche locale haute performance pour l'optimisation de la distribution de gaz industriels par camions-citernes. In Roadef'09, Nancy.

[4] P. Kouvelis and G. Yu, *Robust Discrete Optimization and its Applications*, Kluwer Academic Publisher (1997).

[5] H. Aissia, , C. Bazgana, and D. Vanderpooten "Minmax and minmax regret versions of combinatorial optimization problems: A survey" *European Journal of Operational Research* 197,Issue 2, 427-438 (2009)

Multi-Period Street Scheduling and Sweeping

Carmine Cerrone

Department of Mathematics and Computer Science

University of Salerno

ccerrone@deis.unical.it

Benjamin Dussault

Department of Applied Mathematics & Statistics, and Scientific Computation

University of Maryland

dussault@math.umd.edu

Bruce Golden

Robert H. Smith School of Business

University of Maryland

bgolden@rhsmith.umd.edu

Edward Wasil

Kogod College of Business Administration

American University

ewasil@american.edu

1   Introduction

Consider a city which must sweep both sides of all of its streets in two days. Both sides of any street cannot be closed for sweeping on the same day because of a necessity for some parking spaces to be available on each street at all times. There are currently parking signs which indicate which side of the street is to be swept on "even" days and which side of the street is to be swept on "odd" days. What closed path should the street sweeper take on each day to minimize total distance traveled while satisfying all the constraints?

This problem can be represented as a Directed Rural Postman Problem which is stated as follows: Given a directed graph $G = (V, E)$, subset $E_R \subseteq E$ of required edges, and non-negative costs associated with each edge of G, determine a closed path with minimum total cost traversing

the links E_R at least once. Thus, each day is treated separately. On even (resp. odd) days, the required subsets are exactly those sides of the streets that must be swept on even (resp. odd) days as indicated by the street signs.

We are motivated by the following variant introduced recently in a major U.S. city: Suppose a city decides to redo all of its parking signs. How should the city schedule the closing of each side of each street so that the length of the optimal sweeping path is minimized while ensuring that both sides of any street are not closed on the same day? It is clear that the relaxing of the schedule can only improve the objective. Each schedule induces an optimal sweeping route for both days, making the determination of an optimal schedule the focus of the problem.

We generalize the problem to "Variant 1" which allows for more parking restrictions. For example, we allow for the possibility that a street does not require parking or does not require sweeping. In particular, this allows for the setting to be changed from sweeping an "entire" city to the more realistic scenario of sweeping a non-connected subset of a city.

We extend further to "Variant 2" which supposes a city has an existing set of street signs. How can the city minimally change these street signs to allow for a schedule with a maximum decrease in distance traveled by the street sweeper? This problem is a natural extension of Variant 1. There is a cost for changing a street sign because it confuses those residents who are familiar with the existing one. Thus, it could be desirable to achieve a significant fraction of the benefit of redoing all the signs at a reduced cost of inconvenience.

2 Literature Review

The generic street sweeping problem can be described at a Directed Rural Postman Problem, for which Christofides et al. [3] give a heuristic and mathematical programming formulation. Using their heuristic, the authors solved twenty-three instances within 1.3% of optimality.

Bodin and Kursh [1], [2], describe a computer-assisted system for scheduling and routing of multiple street sweepers. Their model, like ours, deals with urban settings that involve one-way streets and parking constraints. The required edges are directed, a subset of the entire considered graph, and not necessarily connected. Unlike our work, Bodin and Kursh do not have multi-period parking constraints and instead regard parking constraints on a street as time-window constraints during a single day. Their algorithm seeks to assign streets to sweepers and route the sweepers, while obeying parking regulations, balancing workload, and minimizing deadhead distance. The authors apply the algorithms to pilot studies in New York City and Washington, D.C.

Eglese and Murdock [4] describe their street sweeping application in Lancashire County Council in England. Their work differs from [1] and [2] in that there are no parking considerations to be made and streets can be regarded as bidirectional because, in rural areas, street sweepers are

allowed to traverse a street against traffic.

3   Genetic Algorithm

We employed a genetic algorithm heuristic to generate good solutions in a short amount of time. It acts on a population of feasible schedules and uses a heuristic for the Directed Rural Postman Problem to return a tour whose length serves as the fitness function.

There are several hurdles to overcome in the implementation. First, it is necessary for the required edges to be contained in a strongly connected graph. In our context, the required sides of streets are contained in a city which is assumed to be strongly connected. However, the city may be much larger than the required sweeping area, resulting in an overly large state space of schedules that dictates sweeping constraints on every street. We reduce the state space so that schedules are not concerned with unnecessary streets, greatly reducing running time.

Second, the Directed Rural Postman Problem is an NP-hard problem and its objective value is used as a fitness function which is computationally expensive. Thus, our heuristic for the Directed Rural Postman Problem is carefully chosen for fast running times and good solutions in the context of our problem.

Third, a naive breeding of schedules would be to simply swap components of the schedules in a random fashion. However, one can see that the induced route of a schedule is very sensitive to small changes in the schedule. Requiring sweeping on an edge on an even day rather than an odd day could result in very large detours being required to satisfy the other travel requirements of the schedule. As a result, breeding two good schedules haphazardly will often destroy the good solution structure. We construct a novel breeding process that preserves good solutions while introducing the variety required for a genetic algorithm.

Finally, there are no existing test instances, so we construct algorithms for generating random instances and random instances modeling a city. We also test our methods on actual city data for Washington, D.C.

4   Results

We compare our genetic algorithm against a CPLEX implementation and a local search procedure. Our genetic algorithm performs very well, very nearly matching CPLEX on small instances. However, on only moderately larger instances, CPLEX computation time performance degrades very quickly and CPLEX often fails to find a feasible solution in a reasonable amount of time. Our genetic algorithm performs consistently, obtaining good solutions in far less time with respect to the local search method for realistic and large sized instances.

References

[1] L.D. Bodin and S.J. Kursh, "A Computer-Assisted System for the Routing and Scheduling of Street Sweepers," *Operations Research* 26, 525-537 (1978).

[2] L.D. Bodin and S.J. Kursh, "A Detailed Description of a Computer System for the Routing and Scheduling of Street Sweepers," *Computer and Operations Research* 6, 181-198 (1979).

[3] N. Christofides, V. Campos, A. Corberan, and E. Mota, "An Algorithm for the Rural Postman Problem on a Directed Graph," *Mathematical Programming Study* 26, 155-166 (1986).

[4] R.W. Eglese and H. Murdock, "Routing Road Sweepers in a Rural Area," *The Journal of the Operational Research Society* 42, 281-288 (1991).

Robust and Recoverable Maintenance Routing Schedules

Niklaus Eggenberg   **Matteo Salani**

Michel Bierlaire

Transport and Mobility Laboratory

École Polytechnique Fédérale de Lausanne, Switzerland

Corresponding Author: niklaus.eggeneberg@epfl.ch

November 24, 2009

1   Introduction

In the modern society, the demand for transportation, of goods and people, is constantly increasing in terms of volume and distance. In particular, as the fastest transportation mode for mid and long distances, airline transportation develops at an impressive rate. Due to the competition between the airlines, many of them use operations research techniques to schedule their operations. This allows to keep prices low and thus attract customers while making profit. Airlines have to deal with irregular events, called *disruptions*, making the schedule unfeasible. The process of repairing a disrupted schedule is known as the *recovery* problem. It aims at retrieving the initial schedule as quickly as possible while minimizing the *recovery costs* incurred by recovery decisions (typically delaying or canceling flights).

A major drawback of optimized schedules is that they are sensitive to perturbations. Small disruptions propagate through the whole schedule, and may have a huge impact.

The focus of this study is to implicitly consider the occurrence of future disruptions at the planing phase in order to ameliorate two properties of the schedule, namely:

1. the *robustness*: the ability of the schedule to remain feasible in the presence of small disruptions;

2. the *recoverability*: the average performance of the recovery algorithm when the schedule is disrupted.

At the planing phase, we solve the Maintenance Routing Problem (MRP), which aims at finding a feasible route for each aircraft and a departure time for each flight minimizing the loss of revenue as a metric which depends on the deviation from a desired schedule.

On the day of operation, the problem of recovering the planed schedule from a disrupted state is the Aircraft Recovery Problem (ARP) given the original schedule and the current disrupted *state*. The *recovery costs* are mainly delay and cancelation costs.

The originality of the proposed algorithms is the absence of any explicit predictive model of possible disruptions for the scheduling problem. Uncertainty Features capture implicitly the uncertainty the problem is due to. An additional *budget constraint* ensures that the obtained solution is not too far from the original deterministic optimum, and the computational complexity is similar to the original deterministic problem.

We solve the MRP by applying the Uncertainty Feature Optimization (UFO) framework of Eggenberg et al. (2009) on a real case study and we present computational results for different MRPs using public instances of the ROADEF Challenge 2009. Recovery statistics are obtained with the recovery algorithm presented by Eggenberg et al. (2010, to appear).

1.1 Methodology

Eggenberg et al. (2009) introduce the *Uncertainty Feature Optimization* (UFO) framework. It modifies the original deterministic optimization problem, relaxing the optimality with respect to the original objective in order to maximize structural properties, called *Uncertainty Features* (UF). The underlying assumption is that solutions with higher UF values have higher robustness and/or recoverability.

In the context of airline scheduling, it has been shown that solutions with additional slack, higher number of possible aircraft swaps or increased number of short cycles increase the robustness and the recoverability of a schedule (Ageeva, Y., 2000, Ehrgott and Ryan, 2000, Rosenberger et al., 2004, Lan et al., 2004, Smith and Johnson, 2006, Yen and Birge, 2006, Burke et al., forthcoming, Gao et al., 2009).

Our methodology is therefore to modify existing schedules such as to maximize four different UFs:

1. `IT`: the total slack in the schedule

2. `MIT`: the sum of each aircraft's minimum slack

3. `CROSS`: the number of aircraft crossings (possible aircraft swaps)

4. `PCON`: the total slack for all existing passenger connections

The UFs are maximized using a Column Generation algorithm based on the constraint-specific recovery networks of Eggenberg et al. (2010, to appear), allowing for flight retiming. We use two

restriction levels: on a disaggregate level, we ensure each fight is moved by at most 60 minutes, and on the aggregate level, we ensure that the total deviation (in minutes) is bounded by a constant C.

1.2   Sample of the Simulation Results

In our simulations, we compare the efficiency of different schedules obtained by modifying an original schedule coming from the ROADEF Challenge 2009[1]. We then use a recovery algorithm to solve the aircraft recovery problem for each instance and each modified solution. The solutions are then evaluated using the cost checker provided by the ROADEF Challenge.

Figure 1 shows the *performance profile* Dolan, E.D and Moré, J.J (2002) of some of the most efficient models used to derive new schedules.
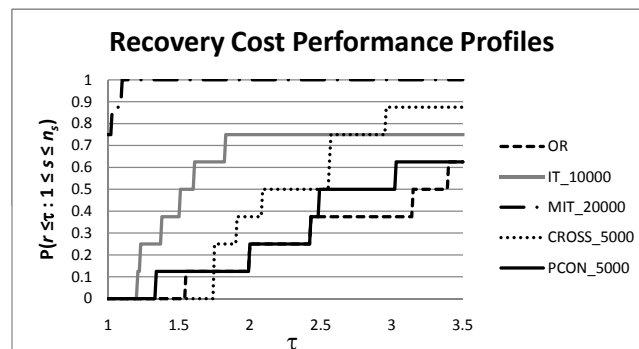
Figure 1: Evolution of the performance curves for different models used to modify the original schedule.

Figure 1 shows that the solutions obtained by model MIT_20000, i.e. maximizing the sum of each aircraft's minimum idle time using an upper bound $C = 20,000$ minutes, leads to the solution with lowest recovery costs in 6 instances out of 8. When it does not have the lowest recovery cost, the solution is however less than 1.1 times higher than the best found solution. Compared to the original schedule, model MIT_20000 always has lower recovery costs. In the best instance, MIT_20000 allows to save up to 3,82 Mio €, i.e. a reduction of the original recovery costs of 68.5%; the highest reduction relative to the original recovery costs is of 93%, corresponding to 1,28 Mio€.

1.3   Conclusion

The drawback of deterministically optimized airline schedules is their sensitivity with respect to disruptions. We show that by modifying the original schedule, we are able to mitigate the sensitivity to disruptions and therefore reduce recovery costs when recovery is required.

---

[1] http://challenge.roadef.org/2009/index.en.htm

References

Ageeva, Y. (2000). Approaches to Incorporating Robustness into Airline Scheduling, *Master Thesis* Massachusetts Institute of Technology.

Burke E.K., DeCausmaecker, P., De Maerea, G., Mulderc, J., Paelinckc, M. and Vanden Berghed, G. (forthcoming). A multi-objective approach for robust airline scheduling, *Computers & Operations Research* **doi:10.1016/j.cor.2009.03.026**.

Dolan, E.D and Moré, J.J (2002). Benchmarking optimization software with performance profiles, *Mathematical Programming Ser. A* **91**: 201–213.

Eggenberg, N., Salani M. and Bierlaire M. (2010, to appear). Constraint-Specific Recovery Networks for Solving Airline Recovery Problems, *Computers & Operations Research* **37**: 1014–1026.

Eggenberg, N., Salani M. and Bierlaire M. (2009). Uncertainty Feature Optimization: an implicit paradigm for problems with noisy data, *Technical Report* **TRANSP-OR 080829**.

Ehrgott, M. and Ryan, D.M. (2000). Bicriteria Robustness versus Cost Optimization in Tour of Duty Planning at Air New Zealand, *25th Annual Conference of Operations Research Society of New Zealand.*

Gao, C., Johnson, E.L. and Smith, B. and Vanden Berghed, G. (2009). Integrated Airline Fleet and Crew Robust Planning *Transportation Science* **43(1)**: 2–16.

Lan, S., Clarke, J.P. and Barnhart, C. (2006). Planning for Robust Airline Operations: Optimizing Aircraft Routings and Flight Departure Times to Minimize Passenger Disruptions, *Transportation Science* **40(1)** : 15–28.

Rosenberger, J.M., Johnson E.L. and Nemhauser, G.L. (2004). A Robust Fleet Assignment Model with Hub Isolation and Short Cycles, *Transportation Science* **38 (3)**: 357–368.

Smith, C.B. and Johnson, E.L. (2006). Robust Airline Fleet Assignment: Imposing Station Purity Using Station Decomposition, *Transportation Science* **40(4)**: 497–516.

Yen, J.W. and Birge, R. (2006). A Stochastic Programming Approach to the Airline Crew Scheduling Problem, *Transportation Science* **40(1)**: 3–14.

Green Logistics: Three Vehicle Routing and Scheduling Case Studies

R.W. Eglese

Department of Management Science

Lancaster University, Lancaster, U.K.

Email: R.Eglese@lancaster.ac.uk

D. Black

Department of Management Science

Lancaster University

1 Introduction

Traffic congestion is a feature of road transport in most countries of the world. Sometimes the congestion is caused by an unexpected event such as an accident which blocks the smooth flow of traffic. But much congestion is the result of the volume of traffic on a restricted road network and the resulting reductions in average speeds on different roads and at different times follow patterns that tend to be followed on a regular basis. Traffic information is now collected in a variety of ways so the speeds of vehicles on roads at different times can be estimated from the speeds observed in the past. Such information can then be used to find the shortest time paths between places. These shortest time paths may change depending on the time when the journey is started. Eglese et al. [1] show how a Road Timetable[TM] may be constructed that records the times and paths between a set of customers and a depot for use in planning deliveries for a vehicle fleet.

Maden et al. [2] describes a vehicle routing and scheduling algorithm called LANTIME that is able to use the information in the Road Timetable[TM] to produce a set of vehicle routes and schedules for a vehicle fleet that will take the predicted congestion into account and produce routes and schedules based on when and where patterns of traffic congestion occur. The paper goes on to describe the effects of using LANTIME in an initial case study.

In this paper two further case studies will be described and the effects of using the LANTIME algorithm will be analysed and contrasted with the first case study. The cases cover different types of distribution for different companies in different parts of the U.K. In each case the traffic information has been supplied by ITIS Holdings who collect data on the position and speed of vehicles in their Floating Vehicle Database. Each vehicle that has been fitted with the appropriate technology is able to transmit its position and speed at regular intervals while it is being used. This data can then be collated to provide a distribution of speeds in any time interval on any road in the network.

2 Related Work

There are a relatively small number of papers that carry out vehicle routing and scheduling using time-dependent travel times. Examples include Fleischmann et al. [3], Ichoua et al. [4] and van Woensel et al. [5].

There have also been related developments in dynamic vehicle routing and scheduling where vehicle routes are modified in real time as they react to observed traffic conditions. Ichoua et al. (2006) [6] and Taniguchi and Shimamoto [7] provide examples. However for applications such as the three case studies described in this paper, the customers to be served by each vehicle must be determined when the vehicles are loaded at the depot based on expected travel times.

These papers evaluate their methods in terms of times and economic costs, but the analysis of the case studies presented in this paper will also consider the effect on Greenhouse Gas emissions.

3 The First Case Study

The first case study is based on the distribution of electrical goods. These are items ordered by electricians for carrying out their work and include cables, switches and tools etc. The case study considers just one part of the supply chain operation where items are taken from a regional distribution centre in Avonmouth, near Bristol, to customers located throughout the South West of the UK, including South Wales. The operation is carried out on a daily basis Monday to Friday. The vehicles used are all 3.5 tonne GVW box vans, so there are no restrictions on the roads on which they may travel. As the items of electrical equipment are relatively small and light there are no effective constraints on the capacity of the vans. However each driver is available only for a maximum 10 hour working day including the statutory breaks for driving time and working time. There are no time window constraints for the deliveries, other than that they must all be delivered on a particular day.

For this part of the operation, about seven vans were normally required, though additional vans and drivers were available if needed. If a schedule did not require a van and driver for the whole day, then there were other parts of the operation where the driver could be used. The number of customers per day varied between 40 and 64 over the nine days when data were collected.

Road Timetables were constructed based on a three-month period at the same time of year in the previous year and used 15-minute time bins covering the day.

Vehicle routes and schedules for each day were first constructed using conventional vehicle routing and scheduling methods where the speeds of the vehicles on each type of road were input as the expected speeds when the roads were uncongested and the free-flowing speeds could be used. These routes and schedules were then examined using the observed vehicle speeds at different times of day taken from the Road Timetable. It showed that many routes took longer than the planned figures due to road congestion and in some cases led to overtime being needed to complete the routes.

However the LANTIME algorithm was able to construct routes and schedules which complied with all the time constraints.

4 The Second Case Study

The second case study concerns a company that distributes fruit and vegetables in London and the South East of England from the main wholesale market in Spitalfields. The customers of this company are typically restaurants and kitchens for catering outlets providing meals for schools and hospitals. The company again use vans for their deliveries and most deliveries are made during the early morning Mondays to Fridays. The mix of customers varies each day. Some customers may need daily deliveries but many of them require deliveries less frequently. The average number of customer drops per day is 130 and the number of vehicles is about 14.

The road network over which these vehicles travel includes roads in the centre of London as well as motorways and main roads outside the main urban area.

5 The Third Case Study

The third case study is based on supermarket deliveries for a company with 26 supermarkets based in the North of England. The deliveries are mainly food and are packed into cages which are then loaded into heavy goods vehicles. The vehicles are driven from a central distribution depot and taken to the stores. After delivering goods, the vehicles may also be used to transport waste or material for recycling from the stores back to the central depot and may also be used to pick up supplies from one of the local producers and backhaul it to the central depot. The company operate with a fleet of about 20 vehicles.

The road network for this case study includes motorway and major trunk roads for travel between cities and towns, but also includes town and city centre road networks that must be used to access the stores.

6 Conclusions

Analysis of the first case study shows that conventional methods that do not take time-varying speeds into account, except for an overall contingency allowance, may still lead to some routes taking longer than the time allowed. The LANTIME approach produces more reliable route times and leads to savings in CO_2 emissions of about 7% for the sample analysed.

The second and third case studies will be analysed in a similar way to examine whether the results from different types of operations over different road networks produce similar benefits.

References

[1] R.W. Eglese, W. Maden and A. Slater, "A Road Timetable [TM] to aid vehicle routing and scheduling", *Computers and Operations Research* 33, 3508-3519 (2006).

[2] W. Maden, R.W. Eglese and D. Black, "Vehicle Routing and Scheduling with Time Varying Data: A Case Study", *Journal of the Operational Research Society* (forthcoming) (2010).

[3] B. Fleischmann, M. Gietz and S. Gnutzmann , "Time-varying travel times in vehicle routing", *Transportation Science* 38, 160-173(2004).

[4] S. Ichoua, M. Gendreau and J.-Y.Potvin, "Vehicle dispatching with time-dependent travel times", *European Journal of Operational Research* 144, 379-396 (2003).

[5] T. van Woensel, L. Kerbache, H. Peremans, N. Vandaele, "Vehicle routing with dynamic travel times: A queueing approach", *European Journal of Operational Research* 186, 990-1007 (2008).

[6] S. Ichoua, M. Gendreau and J.-Y.Potvin, "Exploiting knowledge about future demands for real-time vehicle dispatching", *Transportation Science* 40, 211-225 (2006).

[7] E. Taniguchi and H. Shimamoto, "Intelligent transportation system based dynamic vehicle routing and scheduling with variable travel times", *Transportation Research Part C* 12, 235-250 (2004).

Efficient time-dependent vehicle routing in urban areas

Jan Fabian Ehmke

Dirk Christian Mattfeld

Decision Support Group

University of Braunschweig, Mühlenpfordtstraße 23, Braunschweig, Germany

Email: j-f.ehmke@tu-bs.de

1 Motivation

This paper is about efficient and reliable vehicle routing in city logistics. We consider telematics based data collection, data processing and data utilization in order to provide efficient information models as input for time-dependent vehicle routing in urban areas.

In city logistics, concepts for fast and reliable transportation of goods in terms of efficient and environmentally acceptable pickup and delivery routes are discussed. Nowadays, service providers have to consider dynamics within logistics planning processes, e.g., shorter delivery time, higher schedule reliability and delivery flexibility [1]. Furthermore, city logistics service providers compete against other road users for the scarce traffic space of inner cities. In conurbations, traffic infrastructure is often used to capacity, resulting in traffic jams. This leads to lower service quality and higher costs for service providers [2].

Efficient time-dependent vehicle routing in urban areas is based on empirical traffic data that can be utilized in time-dependent problem formulations. Thus, varying traffic flows and customer time windows have to be considered. Varying traffic flows can be approximated by time-dependent travel time estimates. Recently, such data arises from telematics based vehicular communication networks. Travel time estimates are then integrated into time-dependent routing approaches that meet customer windows. Whereas common vehicle routing is well studied, time-dependent vehicle routing is still a field of potential research due to the substantial efforts in data processing and the resulting complexity in routing algorithms [3]. In particular, the provision and integration of time-dependent information models into appropriate routing formulations is rarely focused. Recent work on time-dependent vehicle routing can be found in [2],[3],[4].

In this contribution, a time-dependent optimization framework for vehicle routing in urban areas is designed. Therefore, key issues of traffic data collection are presented. The appropriate transformation from raw traffic data into time-dependent information models is discussed (Section 2). Then, the information models are utilized in time-dependent routing approaches (Section 3). Several

time-dependent routing heuristics are compared regarding efforts in data provision, running times and quality of travel time estimation. Computational results arise from a case study based on large amounts of real traffic data from Stuttgart, Germany (Section 4). Here, customer time windows are considered.

2 From empirical traffic data to information models

Time-dependent vehicle routing requires empirical traffic data as a key input. Reliable decisions must be derived from this data. Therefore, empirical traffic data has to be transformed into efficient time-dependent information models. The corresponding data processing, ranging from GPS based data collection of raw traffic data to the provision of time-dependent information models, is mainly based on two aggregation steps [5].

Within first level aggregation, empirical traffic data is cleaned, integrated into a central database and precalculated in terms of time-dependent aggregation. The result is a mean speed for every link and every time interval considered. Corresponding to common analysis methods from the area of traffic research (e.g. [6]), we establish 24 time intervals per weekday, resulting in 168 planning intervals in total.

Within second level aggregation, we refer to data mining in order to reduce the data input for vehicle routing algorithms. The time-dependent aggregates from first level aggregation are normalized and links are then clustered, leading to a compact representation of time-dependent travel time estimates in terms of discount factors. The discount factors represent the typical speed variation for a group of links on a specific day of the week. The main idea is to look up a time-dependent discount factor and then weight a link's robust speed figure (e.g. average speed) for time-dependent route calculation instead of using a considerable amount of travel time estimates for handling time dependency.

The resulting information models differ in the volume of input data for routing algorithms and thus in the complexity of data structures to be considered. Whereas a common digital roadmap consists of about 100,000 travel time estimates for a typical large city in Germany (one travel time estimate per link), time-dependent data from first level aggregation leads to 16.8 million travel time estimates to be considered (24 x 7 per link). After reduction by second level aggregation, only 1.01 travel time estimates per link have to be handled, still maintaining planning reliability by consideration of time dependency, but keeping complexity of data structures low. Comprehensive computational experiments with empirical traffic data from Stuttgart, Germany have shown the superiority of information models from second level aggregation regarding efficiency and reliability of route planning in contrast to route planning based on static travel time estimates from a digital roadmap [7].

3 Integration of information models in time-dependent vehicle routing

Efficient vehicle routing depends on the calculation of distances between customers. In contrast to static vehicle routing formulations, varying travel time estimates for each edge must be considered in time-dependent approaches, inducing time-dependent distance matrices. Usually, an edge's travel time is modeled as a function of its departure time.

Travel time functions can be modeled in a discrete or continuous way [8]. Due to the structure of the information models presented, we refer to the discrete case and approximate the travel time estimates by piecewise-linear travel time functions. Therefore, the time horizon is partitioned into a number of time intervals corresponding to the data delivered by the information models.

The information models presented lead to piecewise-linear travel time functions, ignoring the FIFO property. In FIFO networks, vehicles arrive in the order they commence an edge [9]. In non-FIFO networks, the travel time function "jumps" between two time intervals. Thus, passing may occur if the travel time decreases, leading to inconsistencies in common shortest path algorithms. We solve this problem by utilizing a "smoothed" travel time function that transforms non-FIFO edges into FIFO edges [3].

The FIFO adapted information models can be used for time-dependent shortest path calculation in terms of modified label-setting or label-correcting static shortest path algorithms. They provide data for time-dependent traveling salesman and vehicle routing heuristics.

4 Computational experiments

The applicability and the benefit of the time-dependent information models are demonstrated by computational experiments. We investigate a scenario of a city logistics service provider that serves 20 customers with one vehicle from a central depot in the area of Stuttgart, Germany. Therefore, a large amount of telematics based traffic data from this area is analyzed, which serves as input for the determination of the time-dependent information models presented. The corresponding traveling salesman problem is then solved heuristically. Results are improved by a time-dependent 2-opt approach.

In particular, we implement and instantiate the time-dependent optimization framework and compare memory requirements as well as the computational efforts in route calculation, which strongly depend on the information model used. A common digital roadmap serves as a benchmark and is then extended by the time-dependent information models presented so far. The information models are utilized in order to provide travel time estimates for a variety of time-dependent vehicle routing heuristics. Optimization is conducted by Nearest Neighbor, Route Construction and Savings approaches, which are adapted to the time-dependent setting. The start time of a tour is modified to illustrate the influence of varying traffic flows on vehicle routing. Several types of customer time

windows are introduced and results are compared to the case without time windows. Results are compared regarding computational effort for data provision, route calculation and precision in travel time estimation.

Time-dependent information models represent typical travel times. Due to the stochastic traffic process, the consideration of time dependency is only one important step for maintaining the reliability of delivery tours. Thus, we utilize the information models presented for calculation of robust tours. Here, resulting tours are expected to comprise only a minimum of links with high variance in travel times. Several levels of reliability are compared to each other as well as to the time-dependent and to the static case.

References

[1] K. Windt, M. Hülsmann, "Changing Paradigms in Logistics – Understanding the Shift from Conventional Control to Autonomous Cooperation and Control", in *Understanding Autonomous Cooperation and Control in Logistics,* M. Hülsmann, K. Windt (eds.), 1-72, 2007.

[2] R. Eglese, W. Maden, A. Slater, "A road timetable to aid vehicle routing and scheduling", *Computers and Operations Research*, 12, 3508-3519 (2005).

[3] B. Fleischmann, M. Gietz, S. Gnutzmann, "Time-Varying Travel Times in Vehicle Routing", *Transportation Science* 2, 160-173 (2004).

[4] T. Van Woensel, L. Kerbache, H. Peremans, N. Vandaele, "Vehicle routing with dynamic travel times: A queueing approach", *European Journal of Operations Research*, 286, 990-1007 (2008).

[5] J. F. Ehmke, S. Meisel, S. Engelmann, D. C. Mattfeld, "Data chain management for Planning in City Logistics", *Int. Journal of Data Mining, Modelling and Management*, 1, 335-356 (2009).

[6] L. Pinkofsky, Typisierung von Ganglinien der Verkehrsstärke und ihre Eignung zur Modellierung der Verkehrsnachfrage, PhD. Thesis, Technische Universität Braunschweig (2005).

[7] J. F. Ehmke, D. C. Mattfeld, "Data allocation and application for time-dependent vehicle routing in city logistics", *Proceedings of EWGT 2009,* Padua (2009).

[8] B. C. Dean, Shortest Paths in FIFO Time-Dependent Networks: Theory and Algorithms, Massachusetts Institute of Technology (1999).

[9] S. Ichoua, M. Gendreau, J.-Y. Potvin, "Vehicle dispatching with time-dependent travel times", *European journal of operational research*, 2, 379-396 (2003).

Finding optimal toll locations and levels in elastic demand networks
-A MILP approximation approach

Joakim Ekström

Department of Science and Technology

Linköping University, Norrköping, Sweden

Email: joaek@itn.liu.se

Clas Rydergren

Department of Science and Technology

Linköping University

Agachai Sumalee

Department of Civil and Structural Engineering

The Hong Kong Polytechnic University

1 Introduction

The toll design problem (TDP) is that of finding optimal toll locations and levels in a congestion pricing scheme, with the objective to maximize the social surplus given by the Marshallian measure. The TDP is in general non-convex and therefore difficult to solve for a global optimum. A similar problem to the TDP is the continuous network design problem (CNDP), and in [1] the CNDP for fixed demand networks is approximated by a mixed integer linear program (MILP), for which a global optimum can be obtained. The approximation used for the CNDP however relies on the set of used paths to be known in advance.

In this paper we extend the ideas from [1], to the TDP for elastic demand traffic assignment. A link based formulation is adopted which does not rely on the set of used paths being known a priori. Also, when performing the approximation we will ensure that the solution to the MILP will give an upper bound of the objective function value to the TDP.

2 The toll design problem

The traffic network is modeled by a set of links $\mathcal{A}$ and a set of origin destination (OD) pairs $\mathcal{I}$. Let v be the vector of link flows, with v_a denoting the flow on link a. Furthermore, let x^i be the vector of link flows disaggregated by OD pair $i \in \mathcal{I}$. The set of feasible link flows and demands can then be formulated as

$$\Omega = \left\{ v : v = \sum_{i \in I} x^i, Ax^i = b_i q_i, q_i \geq 0, x^i \geq 0 \quad \forall i \in I \right\},$$

where A is the link-node incidence matrix for the network. The vector b_i has length equal to the number of nodes, and defines the origin and destination nodes in OD pair i, with the element at the position of the origin node equal to -1 and that of the destination node equal to 1. The link travel time is assumed to be a monotonically non-decreasing function t_a of v_a. The cost of traveling on link a is made up of both the link travel time and the link toll, τ_a, and is expressed as $c_a(\tau_a, v_a) = \alpha t_a(v_a) + \tau_a$, where α is the value of time. The relationship between travel cost and demand in each OD pair i is given by the inverse travel demand function D_i^{-1}, which is assumed to be a convex function of travel demand q_i, and gives the travel cost in OD pair i.

In each OD pair the road users are assumed to choose routes according to a user equilibrium (UE). A variational inequality formulation (VI) is adopted to describe the UE with elastic demand [2]. The VI is defined for all feasible demand-link flow vectors $(\tilde{q}, \tilde{v})$ in Ω. For the UE, Ω can be assumed to be a bounded polyhedron with a finite number, S, of extreme points, $(\hat{q}^s, \hat{v}^s)$ [3]. We can thus formulate the VI as:

$$\sum_{a \in \mathcal{A}} c_a(\tau_a, v_a)(v_a - \hat{v}_a^s) - \sum_{i \in \mathcal{I}} D_i^{-1}(q_i)(q_i - \hat{q}_i^s) \leq 0, \quad s \in 1..S.$$

Now, the TDP can be formulated as a mathematical program with the VI as constraints, and g_a as the cost of collecting a toll on link a. The TDP is:

$$\max_{\tau,q,v,y} F(\tau, q, v, y) = \sum_{i \in \mathcal{I}} \int_0^{q_i} D_i^{-1}(w)dw - \alpha \sum_{a \in A} t_a(v_a)v_a - \sum_{a \in \mathcal{A}} g_a v_a \tag{1a}$$

$$\text{subject to} \quad \sum_{a \in \mathcal{A}} (\alpha t_a(v_a) + \tau_a)(v_a - \hat{v}_a^s) - \sum_{i \in I} D_i^{-1}(q_i)(q_i - \hat{q}_i^s) \leq 0, \quad s \in 1...S \tag{1b}$$

$$\tau_a \leq y_a \bar{\tau}_a, \ a \in \mathcal{A} \tag{1c}$$

$$y_a \in \{0, 1\}, \ a \in \mathcal{A} \tag{1d}$$

$$v \in \Omega, \tag{1e}$$

where the first sum in the objective function is the user benefits, the second sum is the total travel time, and the last sum is the cost of collecting the tolls. The variable y_a is equal to one if a toll is

located on link a, and zero otherwise, and $\bar{\tau}_a$ is an upper bound (possibly zero) on the toll level of link a. By fixing the y-variables to either one or zero, we will only search for optimal toll levels.

3 The MILP approximation

The TDP is a mixed integer non-linear problem since there are non-linear functions in both the objective and the constraints. If the non-linear functions are approximated by piecewise linear ones, the problem becomes a mixed integer linear program (MILP). The MILP approximation is still non-convex due to the integer variables, but can be solved efficiently by known methods, such as branch and bound, to the global optimal solution.

The piecewise linear approximations will be done by linearizing the non-linear functions in given points. For all the functions but the link travel times and the link toll revenues, the piecewise linear functions can be modeled as linear inequalities. For the approximation of the link travel time we however need to add a set of constraints and integer variables for each link, to describe which linear segment that is active at the current link flow. The link toll revenue, which is a bilinear term, can be underestimated by its convex envelope. To improve the underestimation, the link flow toll level is divided into different segments, where each segment is specified by a lower and upper bound on the link flow and toll level respectively. These lower and upper bounds are combined into four points in the link flow-toll-space, which are used to define each convex envelope. To determine which segments that are active, i.e. which convex envelope that is to be used for the underestimation, a set of linear inequalities and integer variables needs to be added for each link.

The optimal objective function value of the MILP will be an upper bound estimation of the optimal objective function value of the TDP if the link travel cost $t_a(v_a)$, the total OD travel cost $q_i D_i^{-1}(q_i)$, and the user benefits $\int_0^{q_i} D_i^{-1}(w)dw$ are overestimated, and the total link travel cost $v_a t_a(v_a)$, the link toll revenues $v_a \tau_a$, and the inverse travel demand $D_i^{-1}(q_i)$ are underestimated.

The MILP has the following structure:

$$\max_{\tau, d^{inv}, d^{UB}, d^{TC}, r, t, \bar{t}, q, v, y} F_{\text{MILP}}(d^{UB}, \bar{t}, v, y) = \sum_{i \in \mathcal{I}} d_i^{UB} - \sum_{a \in \mathcal{A}} \alpha \bar{t}_a - \sum_{a \in \mathcal{A}} g_a(v_a, y_a) \tag{2a}$$

$$\text{subject to} \quad \sum_{a \in \mathcal{A}} \left(\alpha \bar{t}_a + r_a - (\alpha t_a + \tau_a) \hat{v}_a^s \right) - \sum_{i \in I} \left(d_i^{TC} - d_i^{inv} \hat{q}_i^s \right) \leq 0, s \in 1...S \tag{2b}$$

$$\text{and constraints (1b)-(1e),}$$

where t, $\bar{t}$ and r_a are variables that correspond to the piecewise linearization of the link travel time, the total link travel time, and the link toll revenue functions respectively. The variables d^{inv}, d^{TC} and d^{UB} are given by the piecewise linearization of the inverse travel demand function, the total OD travel cost, and the user benefit functions respectively. Note that additional constraints

and integer variables are needed to give the relationship between these variables and the piecewise linear functions.

The MILP formulation relies on the complete set of extreme points $s \in 1...S$ to be known a priori, and it is burdensome even for a small network to find them all. The MILP can however be formulated with a reduced number, R, of extreme points, referred to as **MILP-EX**, by replacing the complete set of extreme points $1...S$ in (2b) with $1...R$.

Let $(d^{inv^*}, d^{TC^*}, r^*, t^*, \bar{t}^*, \tau^*)$ be the optimal solution to MILP-EX. If all variables are kept fixed, the search for an additional VI-constraint which is violated by the current solution, can be formulated as the linear program

$$\textbf{LP-EX:} \max_{(\bar{q}, \bar{v}) \in \Omega} F_{\text{LP}}(\bar{q}, \bar{v}) = \sum_{a \in \mathcal{A}} (\alpha \bar{t}_a^* + r_a^*) - \sum_{i \in \mathcal{I}} d_i^{TC^*} - \sum_{a \in \mathcal{A}} (\alpha t_a^* + \tau_a^*) \bar{v}_a + \sum_{i \in \mathcal{I}} d_i^{inv^*} \bar{q}_i,$$

with optimal solution $(\bar{q}^*, \bar{v}^*)$.

If $F_{\text{LP}}(\bar{q}^*, \bar{v}^*) \leq 0$ then there exists no additional constraint which would make the current solution to MILP-EX infeasible, and the current solution is thus an optimal solution to the MILP. On the other hand, if $F_{\text{LP}}(\bar{q}^*, \bar{v}^*) > 0$ there exist an extreme point equal to $(\bar{q}^*, \bar{v}^*)$, which violates the VI-constraint. By repeatedly solving MILP-EX and LP-EX an iterative solution algorithm can be constructed, in which LP-EX will either indicate that that optimum has been reached, or identify a new VI-constraint.

The MILP approximation approach has been evaluated on a small network with 12 links, and 8 OD pairs, where the link travel time functions and inverse travel demand functions are all linear. The total link travel time, OD travel cost, and user benefits are approximated by about 20 linear segments each. For the link toll revenues, the link flow and toll levels are divided into ten segments each. The results give that the MILP overestimates the TDP by about 0.1%. When the optimal tolls from the MILP solution are applied to the TDP, the improvement in social surplus are within 0.9% and 2.3%, of the known optimal solution.

References

[1] D. Z. Wang and H. K. Lo. "Global optimum of the linearized network design problem with equilibrium flows". *Transportation Research Part B: Methodological* In Press (2009).

[2] S. Dafermos. "The General Multimodal Network Equilibrium Problem with Elastic Demand". *Networks* 12 57–72 (1980).

[3] S. Lawphongpanich and D. W. Hearn. "An MPEC Approach to Second-best Toll Pricing". *Mathematical Programming* 101 33–55 (2004).

Solving the Weekly Log-Truck Scheduling Problem by Integer Programming

Nizar El Hachemi

CIRRELT and Département de mathématiques et génie industriel

Ecole Polytechnique de Montréal

Issmail El Hallaoui

Département de mathématiques et génie industriel

Ecole Polytechnique de Montréal

Michel Gendreau

CIRRELT and Département de mathématiques et génie industriel

Ecole Polytechnique de Montréal, C.P. 6079, succ. Centre-ville, Montreal, Canada

Email: michel.gendreau@cirrelt.ca

Louis-Martin Rousseau

CIRRELT and Département de mathématiques et génie industriel

Ecole Polytechnique de Montréal

1   The Log-Truck Scheduling Problem

Forest-based industries represent a major economic sector in Canada and in several other countries around the world. In many of these, transportation activities account for a large portion of the costs incurred to exploit forests. For instance, in Quebec, the average distance between forest areas where wood is collected and mills to which this wood is transported is around 150 km, and transportation represents more than 30% of the cost of provisioning for wood transformation mills. Transport activities between forest areas and mills should therefore be organized as effectively as possible. Significant attention has thus been devoted in recent years to transportation-related scheduling problems, mainly for economic and environmental reasons.

In this presentation, we consider the problem of supplying several woodmills (demand points) from a number of forest areas (supply points). Volumes of wood are expressed in units of truckloads at both supply and demand points. In the case at hand, there are no time windows at forest areas

and woodmills, but the trucks that transport wood between these points and the log-loaders that load trucks in the forest and unload them at woodmills must be synchronised as much as possible to avoid waiting time. Demand at woodmills is given on a daily basis, whereas routes and schedules of trucks must be determined on a weekly basis. Another constraint is that each truck should visit a single forest area and a single mill in any given trip. We also assume that there is a single log-loader at each supply or demand point. Finally, there are constraints on the stocks of wood at mills, which require integrating transportation schedules over several days.

This Log-Truck Scheduling Problem (LTSP) is closely related to routing problems encountered in other settings, in particular, pick-up and delivery problems. In general, LTSP is more complex than classical pick-up and delivery problems, the main difference coming from fact that in the LTSP one must synchronise trucks and log-loaders.

Several models and methods have been proposed in the literature to solve the LTSP. Among these, the heuristic-based approach of (ASICAM) [6] has been used successfully since 1990 to produce daily plans for trucks in Chile. Palmgren et al. [5] have proposed more recently a column-based routing model, which is solved using a branch-and-price procedure. Flisberg et al. [3] presented a two-phase solution approach, in which the LTSP is transformed into a standard vehicle routing problem with time windows. Gronalt and Hirsch [4] applied a tabu search algorithm to solve a restricted variant of the LTSP in which the number of trips between each forest area and each mill is given. The same assumption is used by El Hachemi et al. [1] who develop a hybrid method combining Integer Programming (IP) and Constraint Programming (CP): the IP model generates optimal routes in term of deadheading, while CP deals with the scheduling part. Recently, the same authors [2] have presented a novel two-phase approach for solving the weekly variant of the LTSP, in which inventories at woodmills must be taken into consideration. In the first phase, a "tactical" IP model, which is solved approximately by a tabu search heuristic, handles stock and inventory constraints at each demand and supply point. This first phase yields seven daily scheduling problems. For each daily problem, there is a fixed set of transportation requests to perform. The daily problems are solved sequentially, with only one element linking successive days: the location at which a truck finishes a day must be its starting location for the following day. For the second phase, they use a hybrid method that integrates a CP model and a constraint-based local search model (CBLS) for the daily log-truck scheduling problem.

The main contribution of this paper is the integration in the problem definition of important practical considerations, such as the scheduling of lunch breaks, additional supply constraints and the home bases of trucks. This leads to the definition of a new IP model for the daily problem. Both the weekly and the daily IP models are now solved directly with CPLEX 11, which yields excellent results for a reasonable computational effort.

2 Solution Approach

In this paper, we apply the general two-phase approach proposed in [2]. In the first phase, we solve an extended version of the "tactical model" of [2]. This IP model now takes into account the fact that different wood products must be shipped to woodmills. These multi-products demand constraints arise from the fact that many woodmills order logs from given species in specific lengths and diameters to produce given final products. Logs are thus sorted into different assortments that depend on species, usage, quality and dimension. Each supply point consists of a given assortment group (up to 3 products in our case) and each demand point presents a requirement for a given assortment group. In general, the inventory is known at the beginning of the week since it is the stock associated with the last day of the previous week. When demands at woodmills and supplies at forest areas remain constant during a long period covering many weeks, it may be desirable to generate a weekly solution that can be repeated over the whole period. This is achieved by imposing that the inventory at the end of the week equal to the starting inventory. This phase yields seven daily LTSPs, in which there is a fixed set of transportation requests to perform.

To tackle the daily LTSPs, we propose a new flow-based IP model. In this model, each component (activity) of a truck trip (deadhead, loading, loaded travel, unloading) is modeled as an arc in a time-space network. We assume that trucks are assigned to a given number of "regional bases" from which they start and to which they must return, and that the fleet is homogeneous. The objective function is to minimize the sum of the costs associated with truck deadheads and with waiting times of trucks and forest log-loaders. Constraints enforce the availability of trucks for each "regional fleet", the fulfillment of daily transportation requests coming from the tactical model, and structural relationships. Furthermore, since forest companies have expressed the need to ensure for each truck a one-hour break between 11 AM and 3 PM at any woodmill, where drivers can have lunch and trucks can be refueled, we divide the network into two parts at woodmills nodes, one before the break and one after. The arcs linking both parts represent the break activities of trucks and have a duration of one hour as specified by the forest companies. The daily model being based on a time-space network, a key issue in the model definition is the specification of an appropriate time step. An important observation in this regard is that loading and unloading times are approximatively equal and take around 20 minutes (in practice, loading takes a little bit more time than unloading). Choosing the loading time as the basic time step greatly simplifies the model, in particular, the constraints that limit log-loaders to serving only one truck at any time.

While this daily model is a network model, it does not correspond to a simple flow problem and it must therefore be solved as an IP with branch-and-bound. After trying the default branching strategy of CPLEX 11, we developed our own branching strategy aimed at fixing the activities of log-loaders, because there are relatively fewer of them and because their waiting time cost is about twice as expensive as for trucks; they thus have more impact on the objective.

3   Experimental Results

We were provided with two different case studies by an industrial partner. Both of them involve six forest areas and five woodmills. The case studies involve respectively approximately 400 (resp. 700) shipments per week, and the average cycle time to transport a shipment is around of about 4 (resp. 5.5) hours. Each case has three different fleet sizes (resp. 14-16 trucks and 30-32). For each of these scenarios, we performed three tests with different configurations of bases. All daily LTSPs were run for 3 minutes initially and then 10 minutes. For the first phase, we fixed the computational time to 5 minutes, and the first case study was solved optimally within less than 1.5 minute, while for the second case we obtained a solution with a 4.5% gap. These computational experiments showed that one could easily obtain fairly good solutions within the allotted CPU times and that the special branching strategy outperformed CPLEX's default strategy. Comprehensive results for the experiments will be presented at the conference.

References

[1] N. El Hachemi, M. Gendreau, and L-M. Rousseau, "Solving a Log-Truck Scheduling Problem with Constraint Programming", in *Proceedings of the Fifth International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR08)*, 293-297, 2008.

[2] N. El Hachemi, M. Gendreau, and L-M. Rousseau, "A Hybrid LS/CP Approach to Solve the Weekly Log-Truck Scheduling Problem", in *Proceedings of the Sixth International Conference on Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems (CPAIOR09)*, 319-320, 2009.

[3] P. Flisberg, B. Lidén, and M. Rönnqvist, "A hybrid method based on linear programming and and tabu search for routing of logging trucks", *Computers and Operations Research* 36, 1122-1144 (2008).

[4] M. Gronalt and P. Hirsch, "Log-Truck Scheduling with Tabu Search Strategy", *Metaheuristics* 39, 65-88 (2007).

[5] M. Palmgren, M. Rönnqvist, and P. Värbrand, "A solution approach for log truck scheduling based on composite pricing and branch and bound", *International Transactions in Operational Research*, 10, 433-447 (2003).

[6] A. Weintraub, R. Epstein, R. Morales, J. Seron, and P. Traverso, "A truck scheduling system improves efficiency in the forest industries", *Interfaces* 26(4), 1-12 (1996).

A Benders decomposition approach for the design of Demand-Adaptive transit Systems

Fausto Errico

CIRRELT and Université du Quebec à Montréal

Pav. André-Aisenstadt, 2920, Chemin de la Tour, Montréal QC, H3T 1J4 CANADA

Email: fausto.errico@cirrelt.ca

Teodor Gabriel Crainic

CIRRELT and Université du Quebec à Montréal

Federico Malucelli   **Maddalena Nonato**

DEI - Politecnico di Milano   Dip. di Ingegneria - Università di Ferrara

1   Introduction

Traditional transit services are particularly suited to situations where the demand for transportation is *strong*, i.e., when there is a consistently high demand in the considered territory and time period. The high degree of resource sharing makes it possible to provide efficiently and economically high quality service. In contrast, when the demand for transportation is *weak*, e.g., in low-population density zones, operating a good-quality traditional transit system is very costly. In particular, the fixed structure of traditional transit services cannot economically and adequately respond to significant variations in demand. *Demand-responsive systems* are a family of mass transportation services which, evolving toward a personalization of transportation, *respond* to the actual demand in a specific time period: itineraries, schedules, and stop locations are variable and determined according to the particular needs as they change in time. Demand-responsive systems were introduced under the name of *Dial-a-Ride* (*DAR*) as door-to-door services for users with particular needs or reduced mobility and then extended to more general settings.

DAR systems display a number of drawbacks, some of which follow from their inherent flexibility: Users are obliged to book the service in advance, the actual pick up time is often left to the discretion of the operator, the integration of DAR and other traditional transit services is extremely difficult [5]. With the purpose of addressing such issues, a demand-responsive system different from DAR, denoted *Demand-Adaptive System* (*DAS*), was introduced in [3] and further

developed in several works. A similar system, called MAST, has been introduced later by [4].

DAS combines features of both traditional fixed-line bus service and purely on-demand systems. A DAS bus line serves, on the one hand, a given set of *compulsory* stops according to a predefined schedule specified by suitable time windows. This provides the traditional use of the line without in-advance reservations and makes the integration of DAS with traditional services easy. On the other hand, similarly to DAR, passengers may issue requests for transportation involving *optional* stops, inducing detours in vehicle routes.

Similarly to most transportation systems dedicated to serve several demands with the same vehicle, DAS requires a complex planning process involving interrelated decisions. Schematically, the design of the system has to determine the so called *topological design* of the line as the selection of compulsory and optional stops [2]. Moreover, the time windows associated with compulsory stops have to be defined, thus deciding the time available for possible deviations [1].

We address an important problem, called the *General Minimum Latency Problem* (GMLP), arising in the context of the DAS line design. The GMLP is similar to the TSP except for the fact that its objective function takes into account not only the routing cost, but also a *latency* component related to the amount of time spent by users in the vehicles. This makes the GMLP much harder then the TSP. We address the GMLP by a Branch and Cut algorithm based on Benders decomposition and the exploitation of similarities between the GMLP and TSP polyhedra. Preliminary results show the effectiveness of the proposed methodology.

2 The General Minimum Latency Problem

The design of a DAS line is a complex process and several hierarchical approaches have been proposed to address it [2]. A number of core problems play an important role in the most of these approaches and the GMLP is the most challenging one. It can be stated as follows. Consider a complete directed graph $G = (N, A)$, where the node set $N = \{1, \ldots, n\}$ represents a set of stops. To each arc $(i, j) \in A$ is associated a traversing time $c_{ij} \geq 0$. We are also given a set D of node pairs; to each $(h, k) \in D$ is associated $d_{hk} > 0$ representing the amount of transportation demand from the origin h to the destination k. The objective of the GMLP is to find a Hamiltonian cycle (*tour*) that minimizes the sum of the routing costs (c_{ij}) of the selected arcs and of the traveling time of each passenger (*latency*).

The problem can be formulated introducing a binary *design* variable x_{ij} for each arc $(i, j) \in A$ equal to 1 if the arc (i, j) is in the tour. To account for the latency, multicommodity flow variables $0 \leq f_{ij}^{hk} \leq 1$ are introduced, representing the fraction of the demand from h to k traveling on arc

(i, j). One possible formulation is:

$$min \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{(i,j) \in A} \sum_{(h,k) \in D} q_{ij}^{hk} f_{ij}^{hk} \tag{1}$$

$$\sum_{(j,i) \in A} x_{ji} = 1 \qquad\qquad \forall\, i \in N \tag{2}$$

$$\sum_{(i,j) \in A} x_{ij} = 1 \qquad\qquad \forall\, i \in N \tag{3}$$

$$\sum_{(i,j) \in A} f_{ij}^{hk} - \sum_{(j,i) \in A} f_{ji}^{hk} = \begin{cases} 1 & \text{if } i = h, \\ -1 & \text{if } i = k, \\ 0 & \forall i \in N, i \neq h, k \end{cases} \qquad \forall\, (h,k) \in D,\ \forall i \in N \tag{4}$$

$$f_{ij}^{hk} \geq 0 \qquad\qquad \forall\, (i,j) \in A,\ \forall (h,k) \in D \tag{5}$$

$$x_{ij} - f_{ij}^{hk} \geq 0 \qquad\qquad \forall\, (i,j) \in A,\ \forall (h,k) \in D \tag{6}$$

$$x_{ij} \in \{0,1\} \qquad\qquad \forall\, (i,j) \in A \tag{7}$$

where $q_{ij}^{hk} = \sum_{hk} d_{hk} c_{ij}$. Equations (2) and (3) describe a so-called cycle cover. Equations (4) are the demand flow balance constraints. They imply that for each pair $(h, k) \in D$, one unit of flow must be sent from h to k. Equations (6), coupling the flow to the design variables, impose that commodity (h, k) cannot travel on an arc (i, j) if that arc is not in the tour. The objective function (1) represents the sum of routing costs and latency. Note that, in this case, if the demand is sufficiently dense, the so-called *subtour elimination constraints* of the TSP are implied by equations (4) and (6).

3   A Benders decomposition approach

Using a straightforward Branch & Bound algorithm to address the model (1)-(7) is not computationally efficient even for small instances, for two main reasons: 1) the linear relaxation is extremely difficult to compute because of the high numbers of constraints and variables; 2) the lower bound provided by the linear relaxation is very loose. An approach based on a Lagrangean Relaxations of (6), while constraining the x variables to design a tour, was also applied to GMLP [2]. Even if such an approach could tackle larger instances with respect to the linear relaxation, in practice, it never provided better lower bounds.

We propose a Benders decomposition. When we consider a given *master* point $\bar{x}$, the problem (1)-(7) decomposes into $|D|$ minimum cost flow subproblems. Master points $\bar{x}$ are obtained by solving a relaxation of the original problem called *master problem*. The advantage is that the master problem has a much smaller number of variables with respect to the original formulation. Normally, the subproblem is used to infer useful information for the master problem: if the $\bar{x}$ is unfeasible or

not optimal, *feasibility* or *optimality* cuts are added to the master problem, respectively.

We embed the Benders decomposition in a Branch & Cut scheme. The initial master problem contains only cycle covers and a few optimality cuts; integrality constraints are also relaxed. Optimality and feasibility cuts are then generated throughout the optimization process and not only at integer points. In particular, instead of letting the subproblem generating feasibility cuts, we dynamically add to the master problem several families of inequalities which are facet defining for the TSP. In fact, it is possible to show that all the inequalities facet defining for the asymmetric TSP are also facet defining for the GMLP [2].

In Table 1 we compare results obtained solving model (1)-(7) by Cplex 10.1 (columns 2-5) with those obtained by our algorithm based on Benders decomposition (columns 6-9). Instances are generated as follows: stops are randomly chosen on a square of side 100 and the costs of the corresponding complete graph are the Euclidean distances (column 1 reports the number of nodes). The demand matrix is also complete and randomly generated. As it could be appreciated by inspecting these preliminary results, our approach considerably improved the quality of the bounds obtained.

Instance	Cplex 10.1				Benders			
#Nodes	LB	UP	GAP	Time	LB	UB	GAP	Time
10	428.48	428.48	0	19.38	428.48	428.48	0	226.79
15	492.22	559.46	0.12	1800	491.99	539.54	0.09	1800
20	558.00	-	-	1800	569.73	634.92	0.1	1800
25	549.61	-	-	1800	570.81	672.55	0.15	1800
30	566.77	-	-	1800	597.73	712.31	0.16	1800

Table 1: Comparison between Cplex and Benders Dec. Time Limit 2 hours.

References

[1] T. G. Crainic, F. Errico, F. Malucelli, and M. Nonato. Designing the master schedule for demand-adaptive transit systems. *Annals of Operations Research*, (to appear).

[2] F. Errico. *The design of flexible transit systems: models and solution methods.* PhD thesis, Politecnico di Milano, Italy, 2008.

[3] F. Malucelli, M. Nonato, and S. Pallottino. Some Proposals on Flexible Transit. In Ciriani, Johnson, and Tadei, editors, *Operations Research in Industry*, pages 157–182. McMillian, 1999.

[4] L. Quadrifoglio, M. M. Dessouky, and K. Palmer. An insertion heuristic for scheduling mobility allowance shuttle transit (mast) services. *J. of Scheduling*, 10(1):25–40, 2007.

[5] T.R.B. Operational Experiences with Flexible Transit Services. volume 53 of *TCRP Synthesis report*. Transportation Research Board, 2004.

A Decision Tree Approach for a Stochastic Inventory Routing Problem

Eystein Fredrik Esbensen

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Email: eysteinf@iot.ntnu.no

Kjetil Fagerholt

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Lars Magnus Hvattum

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Bjørn Nygreen

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

1 Case Description

We consider a real world case based on a shipping company operating within a restricted geographical region. There are several customers located along the coast. Dry bulk products are imported to the region by *cape size* ships (i.e. 80,000 to 120,000 dead weight metric tonnes), also called *Ocean Going Vessels*, OGVs. However, the waters along the coasts in the region are so shallow that fully loaded OGVs cannot sail to port. Thus, only smaller ships or partially loaded OGVs would be able to serve the demand of the steel plants. The shipping company under consideration has a number of *panamax* (i.e. 60,000 to 80,000 dwt) sized ships with large cranes (called *Transloader Vessels*, TVs) that meet the OGVs at sea (at *lightering points*) and then move just enough bulk from the OGVs to the TVs so that the OGVs can berth without running aground. This operation, *lightering*, is the primary business of this shipping company. The OGVs are owned and operated by other companies.

We present the scheduling problem for this company, which is a special case of the stochastic and dynamic inventory routing problem. There are several constraints, which will be presented in due order. Our main contribution is a study of the solution method, the decision tree, for use in maritime logistics.

There are eight ports, and most of them have one berth that can contain a panamax sized vessel, and one berth that can hold an OGV. In a few cases, there are more panamax berths. Most berths have cranes that are used to discharge the ships. Other berths do not have cranes, but have conveyor belts that are used to load a ship with dry bulk. The TVs have their own cranes and can, if needed, discharge from such a berth.

There are two types of contracts involved. The first one is the regular lightering operation, while the second one is a full load pickup-and-delivery voyage. The OGVs arrive irregularly, and the time between two lightering operations has to be utilised optimally. One of the various pickup-and-delivery voyages can then be performed between arrivals of OGVs.

For each pickup-and-delivery contract, there is a running balance akin to an inventory level at a customer's port. The balance is reduced at a constant rate, regardless of how much the customer actually removes from its inventory, but is incremented whenever a shipment is delivered. The contracts specify minimum and maximum levels for this balance. If the company cannot send a TV when the balance is close to the minimum level, a ship is hired on the spot market to make a delivery. If the balance is close to the maximum level, voyages for this contract cannot be scheduled. Ships from the spot market can be hired and used for pickup-and-delivery operations at any time, but they cannot perform lightering voyages, since they are not equipped with the cranes that are needed to lighter an OGV. The aspect with the running balance makes this problem a special case of the inventory routing problem.

The customers order OGV shipments (and consequently, lightering services) at their own discretion. Sometimes, several OGVs bound for the same customer arrive in short order, making a congestion at the lightering point. By the time that a TV has lightered the first OGV, sailed to port, discharged, sailed back to the lightering point and lightered the second OGV, the first one would still not be fully discharged. Therefore, it is unnecessary to send more than one TV at a time to serve a single lightering customer.

Failing to meet the OGVs at sea on time will result in fines, called *demurrages*. This aspect of the problem can be thought of as soft time windows of zero length. If a TV arrives n days late for the first in a series of OGVs bound for the same customer, the shipping company has to pay n days of demurrage for each OGV. The reason is that if the TV had arrived on time, each OGV would have finished n days earlier.

The vessels described above are not the only ones using the ports, and other ships arrive irregularly. The knowledge of future arrivals is limited. When commencing a pickup-and-delivery

operation, the congestion at the discharging port will be highly stochastic. In addition, the loading and discharging speeds are occasionally reduced for various reasons. For instance, whereas the time spent sailing, loading and discharging might take eight days, the time spent due to port congestions might take an additional week (or no time at all). The company receives demurrages for the time spent in a port congestion. However, the average daily revenue earned from performing voyages with no congestion is more than the demurrage rate.

The main short-term challenge is the trade-off between increasing a contract balance by performing pickup-and-delivery voyages, and the risk of paying demurrages from arriving late for an OGV.

In summary, one needs to decide the next voyage for a TV when it has completed the current one. The choice is between a lightering operation for a customer that is currently not being served by another TV, or a pickup-and-delivery voyage where the running balance is low enough to warrant a new delivery. The company is informed of port congestions by observers at each port. The incoming OGVs report their positions every five days, as well as every 24 hours during the last five days before arrival at the lightering point.

The optimisation problem is to maximise expected profits. The income in terms of shipment revenue is limited by the OGVs ordered by the customers, and the fixed reduction per time unit of the balance for the pickup-and-delivery contracts. Variable revenue consists only of demurrages collected at customers' ports. Variable costs include OGV demurrages and the cost of hiring ships in the spot market. The aim of the scheduling problem is three-fold. First, avoid arriving late for OGVs, and thereby avoiding demurrages. Second, avoid the cost of hiring ships on the spot market when the pickup-and-delivery contract balances get too close to the minimum levels. This is done by keeping the TVs active, that is, avoid waiting at the lightering points and being stuck in port congestions. This way, the balances will stay above the minimum levels. Third, one prefers being stuck in port congestions (and thereby collecting demurrages) over waiting at a lightering point (in which case, no demurrages are collected) for an OGV that is very far away. This last case is only relevant when all pickup-and-delivery balances are close to their maximum levels.

2   Solution Methods

The solution method implemented for this case involves a decision tree and a simulator. A decision is required only when a TV has completed its current assignment. This decision is the root node of the tree. Each decision node has at most c child nodes, one for each contract (serving a lightering customer or performing a pickup-and-delivery voyage) that is available at that time. These nodes are chance nodes, each of them having b (b is called a *branching factor*) child nodes representing a scenario of future stochastic events. The child nodes are again decision nodes, representing the

next decision in the future; i.e. finding the next voyage for another TV. The time for this decision depends on outcomes of events that lead to the decision. This way, there is an alternation between chance and decision nodes until the planning horizon is reached.

The algorithm has two phases: construction and evaluation. The tree is constructed using simulations. For each of the outcomes of the chance nodes, further events are simulated until a new decision needs to be made. This creates the next decision and outcome nodes in the tree.

When the construction is completed, the tree is ready to be evaluated. Whereas construction starts at the root and proceeds towards the leaves, the evaluation process starts at the leaves and ends at the root. The *net present value* (NPV) of profits incurred up until the planning horizon, discounted using a specified *discount rate d*, is used as a measure of the desirability of the series of outcomes ending at the leaf. The average NPV of that leaf and the $b - 1$ other leaves that originated from an option at a decision point is used as an estimate of the expected NPV of that particular option. The average values for each of the options at that decision point are compared, and the one with the superior result is chosen. The average NPV of the chosen option is used as an estimated result for the simulation branch that ended at that decision point. The branch is then treated the same way as the leaves, and is used together with $b - 1$ other branch results to compute an average, being an estimate of the expected NPV of selecting a particular option at a decision point. This process continues until the original decision point is reached. Again, the option yielding the highest average simulated profit is used, and the method has reached its conclusion.

The discount rate d is not a financial interest rate, it is simply a means to adjust the importance of revenue and costs incurred in the near future compared to the revenue and costs incurred at a later point in time. To be useful, this value needs to be far more than discount rates used in finance.

3   Computational Study

A simulation is started at a fixed initial setting. Whenever a decision needs to be made, the chosen decision algorithm is called. The result is then applied, and the simulation can continue. This continues until H days have been simulated, where $H >> h$. The total undiscounted profits after H days are used as an estimate of the suitability of the planning method and its particular parameter settings.

The decision tree method is compared to the company's own planning method, and has shown to perform significantly better. Further results will be presented at the conference, where we show the effects of varying parameters such as planning horizon, discount rate and branching factor.

A two-stage stochastic programming model for optimal design of a biofuel supply chain from wastes to ethanol

Yueyue Fan
Department of Civil and Environmental Engineering
University of California
Davis, CA 95616

Existing trends in energy supply and consumption are neither secure nor sustainable economically, environmentally, and socially. In view of the pressing issues of energy security and climate change, bioenergy has been strongly promoted by US federal policy as a means to reduce oil dependence and greenhouse gas emission [1,2]. However, the challenge of realizing cost-effective energy solutions with minimal impact on food and other natural resource supplies has not been thoroughly investigated. The true potential of bioenergy at a sustainable level needs to be sought through rigorous system analyses for the entire energy supply chain from feedstock resources to end users, using integrated knowledge from spatial economics, operations research, and alternative energy technologies.

The goal of this study is to establish a mathematical model that can be used to support strategic planning of bioenergy supply chain systems and optimal feedstock resource allocation in an uncertain decision environment. From a sustainability standpoint, feedstock resources that make minimal impact on global food supplies and other natural resources should be emphasized. In this paper, we emphasize on biowastes feedstock resources. Compared to traditional corn grain, lignocellulosic biomass feedstock (such as biowastes) has several advantages: higher energy yields, lower agronomic inputs, less impact on food and land resources, and better life-cycle environmental impact [3]. Specific questions to be answered via the decision model include:

- Can ethanol converted from wastes be part of a sustainable energy solution that is economically viable and environmentally acceptable?

- What are the infrastructure requirements to support the production and delivery of such a bioethanol system?

- What are the impacts of such a system on greenhouse gas (GHG) emission and natural resources?

- How would future uncertainties impact the quality of model results and how could the potential risk caused by imperfect information be reduced?

The entire biofuel supply chain includes feedstock resources, biorefineries, terminals, end users (i.e. demand cities), and the flows between adjacent layers of the supply chain [4]. A representative supply chain from biowastes to ethanol is illustrated in Figure 1. The model inputs include the geographic layout and availability of various types of biowastes resources, a set of candidate refinery locations (selected based on their proximity to water, transportation facilities, and labor markets), a set of existing gasoline terminal locations that can be selected for ethanol blending and distribution, a set of fuel demand sites, and all the cost functions associated with procurement, production, storage, and transportation.
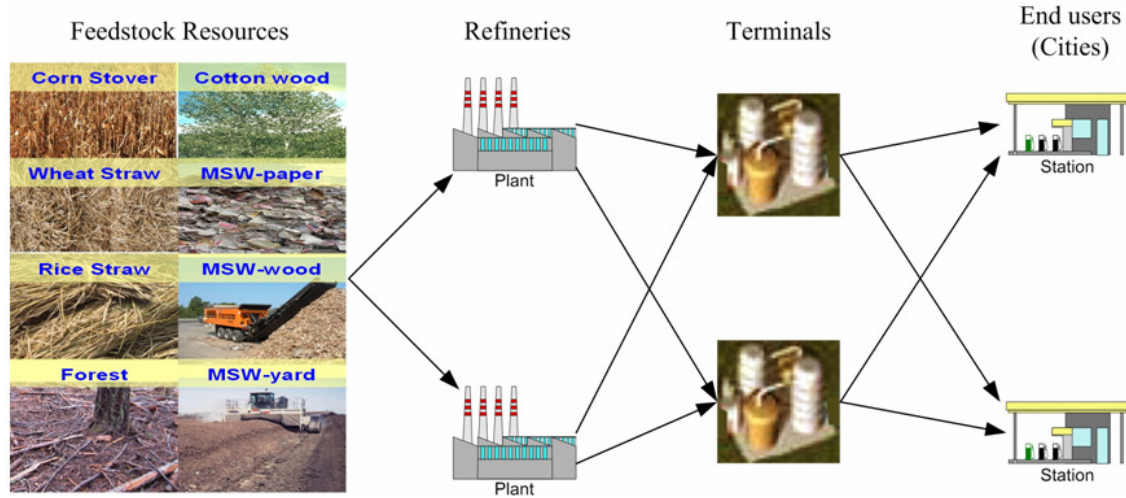
Figure 1. A typical supply chain system for bioethanol converted from biowastes

In this problem, the future demand is assumed to be random, and can be described by a discrete set of possible scenarios and their associated probabilities. A two-stage stochastic mixed-integer programming model [5] is developed to optimize the entire bioethanol supply chain. There are two types of decision variables in the model:

- The planning (first-stage) decisions include the locations and sizes of the refineries and terminals. These decisions are scenario independent, because planning decisions are usually made in advance, and should not be distinguishable between scenarios.
- The operational (second-stage) decisions include the amount of feedstock usage, the quantity of produced ethanol, and the transportation flows between different layers of the supply chain. Operational decisions are easily adjustable according the actual realization of a random event, thus can be scenario dependent.

The objective of the model is to minimize the first-stage cost plus the expected value of the second stage cost. Cost components include the capital and operating cost of the refineries and terminals, feedstock procurement cost, production cost of fuel, transportation cost of feedstock and fuel, and penalty cost of unsatisfied demand.

The stochastic programming model is much larger in size that its deterministic counterpart. A solution algorithm based on progressive hedging (PH) method [6] is design to overcome the computational difficulties. The PH method decomposes a stochastic problem across scenarios and partitions the problem into manageable sub-problems. Let S denote the set of possible scenarios for random demand, $s \in S$ a specific scenario, p_s the probability associated with scenario s. Let x_s, y_s, and $Q_s(x_s, y_s)$ denote the planning decision, operational decision, and the total cost in scenario s respectively. Define G_s as the feasible solution set in scenario s. The problem can be formulated in a compact way as:

$$\min \sum_{s \in S} p_s Q_s(x_s, y_s)$$

subject to $\quad (x_s, y_s) \in G_s,$

$$x_s = z, \forall s \in S.$$

The last constraint is to ensure that the planning decisions across all scenarios are the same, since planning decisions cannot be made with an anticipation of which scenario is actually going to happen. Note that the nonanticipativity constraints are not decomposable. Define the augmented Lagrangian:

$$L_r(x,t,w,z) = \sum_{s \in S} p_s Q_s(x_s, y_s) + (w_s)^T \cdot (x_s - z) + \frac{1}{2} r \|x_s - z\|^2,$$

where w is the vector of dual variables for the nonanticipativity constraints and $r>0$ is a penalty parameter associated with violation of the nonanticipativity constraints. Therefore, the augmented Lagrangian integrates the nonanticipativity constraints with the original objective function. The PH method achieves decomposition by alternatingly fixing the scenario solutions (u,x) and the implementable solution z in the above problem.

The two-stage programming model with the PH based solution algorithm was implemented in a real-world case study based on California settings. Eight types of biowaste resources are considered: cornstover (27 supply clusters), wheat straw (32 clusters), forest residual (47 clusters), rice straw (14 clusters), cotton residual (10 clusters), municipal solid waste (MSW) paper (57 clusters), MSW wood (57 clusters), and MSW yard (57 clusters). There are 28 candidate locations for refineries, 29 candidate locations for terminal, and 143 demand clusters. The model results show that bioethanol can be produced and delivered at an average cost less than $1.5 per gallon (without any subsidy), upon optimization of the entire supply chain, suggesting converting biowastes to ethanol as an economically viable part of future energy solution.

From a modeling viewpoint, we found that the stochastic programming model performs better than a deterministic model (In which only the expected value of random demand was considered). The relative value of stochastic programming solution (VSS) is about 11%. Results also show that stochastic model is less sensitive to imperfect information of model input. From a numerical viewpoint, our experiments demonstrate that PH based solution algorithm performs much better than commercial solver (CPLEX) in case of a nontrivial size of scenarios. In Figure 2, the x axis shows the number of scenarios, and the y axis shows the computing time (in seconds). As the number of scenarios increases, the computational time required by CPLEX solver increases much faster than that by the PH method. On the other hand, we have also noticed that the convergence and stability of the algorithm may be affected by the scale of the penalty parameter r. Recommendations on how to select a suitable range of r and how to speed up the convergence are provided based on our numerical experimental results.
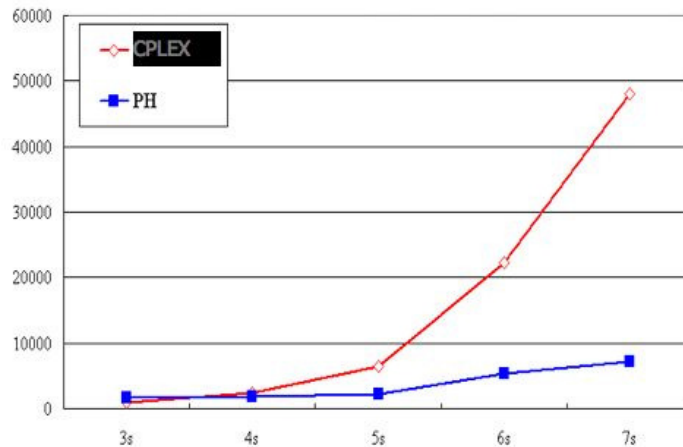
Figure 2. Computing time (in seconds) vs. number of scenarios by different solution methods

References
1. The Energy Independence and Security Act: Renewable Fuels Standard. 2007 Available at http://www.ethanolrfa.org/resource/standard/ (cited 2008 July 5).
2. California Bioenergy Interagency Working Group. Bioenergy Action Plan for California. 2006.
3. Farrell, A.E., R.J. Plevin, B.T. Turner, A.D. Jones, M. O'Hare, and D.M. Kammen. Ethanol Can Contribute to Energy and Environmental Goals. Science, Vol. 311, No. 5760, 2006, pp.506-508.
4. Melo, T., S. Nickel, and F. Saldanha da Gama. Facility location and supply chain management - a comprehensive review. Technical report, Fraunhofer ITWM, 2007.
5. Birge, J. R., and F. V. Louveaux. Introduction to stochastic programming. New York, NY: Springer, 1997.
6. Rockafellar RT, Wets RJ-B. Scenarios and policy aggregation in optimization under uncertainty. Math Oper Res, Vol. 16, 1991, pp.119–147.

A discrete choice approach to simulating airline passenger itinerary flows

Douglas Fearing

Operations Research Center

Massachusetts Institute of Technology

Email: dfearing@mit.edu

Vikrant Vaze

Department of Civil Engineering

Massachusetts Institute of Technology

Cynthia Barnhart

Department of Civil Engineering

Massachusetts Institute of Technology

1 Introduction

Over the past two years, flight and passenger delays have been on the decline due to reduced demand for air travel as a result of the recent economic crisis. As the economy rebounds, demand for air travel in the United States is also expected to recover [1]. Thus, after a brief reprieve, the U.S. will once again face a looming transportation crisis due to air traffic congestion. In calendar year 2007, the last year before the economic downturn, flight delays were estimated to have cost airlines $19 billion (U.S. Congress Joint Economic Committee [2]) compared to profits of just $5 billion (Air Transport Association [3]). In 2007, passengers were also severely impacted, with the economic costs of time lost due to delays estimated at $12 billion according to the Joint Economic Committee report. A similar analysis performed by the Air Transport Association estimated the economic costs of passenger delays at approximately $5 billion for 2007. While there are differences in methodologies, the huge discrepancy between these estimates suggests the need for a more transparent and rigorous approach to measuring passenger delays. Accurately estimating passenger delays is important not only as a means to understand system performance, but also to motivate policy and investment decisions for the National Air Transportation System.

Beyond the need for transparency and rigor, neither of the passenger delay cost estimates listed above include the delays associated with itinerary disruptions, such as missed connections or cancellations. Analysis performed by Bratu and Barnhart suggests that itinerary disruptions and the

associated delays represent a significant component of system performance [4]. Their analysis was performed using one month of proprietary passenger booking data from a legacy carrier. The challenge in extending this analysis systemwide is that publicly available data sources do not contain passenger itinerary flows. For example, on a given day, there is no way to determine how many passengers planned to take the 7:05am American Airlines flight from Boston (BOS) to Chicago (ORD) followed by the 11:15am flight from Chicago (ORD) to Los Angeles (LAX), or even the number of non-stop passengers on each of these flights. The passenger demand data that the Bureau of Transportation Statistics (BTS) provides is aggregated either monthly or quarterly. The methodology we develop in this work is precisely to address this limitation. That is, we use a discrete choice model trained on a small set of proprietary passenger booking data to simulate disaggregate passenger itinerary flows for all airlines. Subsequently, we extend the Passenger Delay Calculator developed in [4] to estimate the magnitude and distribution of U.S. domestic passengers delays for 2007 based on the simulated passenger itinerary flows. Beyond the analysis of historical passenger delays, we expect our approach to be valuable in extending passenger analyses to other contexts where previously only flight information has been available.

2 Description of Data Sources

The U.S. Bureau of Transportation Statistics (BTS) provides a wealth of data related to airline travel. The Airline Service Quality Performance (ASQP) database provides planned and realized flight schedules for all airlines that carry at least 1% of all domestic passengers. For calendar year 2007, this includes 20 airlines from Aloha Airlines with 46,360 flights to Southwest Airlines with 1,168,871 flights. BTS also maintains the Schedule B-43 Aircraft Inventory which provides a historical list of aircraft in inventory for most airlines, matching approximately 75% of the flights in ASQP by tail number.

The Federal Aviation Administration (FAA) maintains the Enhanced Traffic Management System (ETMS) database of all flights tracked by air traffic control. This database is not publicly available, due to the presence of sensitive military flight information, but filtered versions are generally made accessible for research purposes. The benefit of this database over ASQP is that in addition to the planned and realized flight schedules, it contains the International Civil Aviation Organization (ICAO) aircraft equipment code for each flight.

There are two BTS datasets that we depend on for passenger demand information. The first is the T-100 Domestic Segment (T-100) database, which contains passenger and seat counts for each carrier, segment and equipment type aggregated monthly. For example, from this database we can see that in September 2007, American Airlines performed 79 departures from BOS to ORD using Boeing 757-200s with 14,852 seats available and 11,215 passengers. If only one aircraft type is used on a *carrier-segment* (which we define as the combination of the carrier, origin and destination of a flight segment), we can directly estimate the seating capacity of each flight by dividing the number of seats

available by the number of departures performed. By combining Schedule B-43, ETMS, and T-100, we are able to estimate accurate seating capacity for approximately 98.5% of the ASQP flights. The second passenger demand database we depend on is the Airline Origin and Destination Survey (DB1B), which provides a 10% sample of domestic passenger tickets from reporting carriers, including all of the carriers in ASQP, aggregated quarterly by removing information on flight times. For example, in the 3rd quarter of 2007, 128 passengers tickets were sampled that included a one-way trip on American Airlines from BOS to ORD to LAX. We use this data, adjusted according to T-100, to determine the approximate number of monthly passengers travelling on each non-stop or one stop *carrier-route* (which we define as the combination of the carrier, origin and destination for non-stop itineraries or the first carrier, second carrier, origin, connection and destination for one stop itineraries).

3 Discrete Choice Sampling

In this section, we summarize the methodological core of our work. Excluding data processing, the process can be described as an effort to, given the month of travel and the *carrier-route* for a non-stop or one-stop passenger, sample a matching itinerary from a discrete choice probability distribution with regression parameters estimated from proprietary booking data. For example, given a passenger estimated to be traveling on American Airlines from BOS to ORD to LAX in September, there are 526 matching itineraries generated from ASQP using a minimum connection time of 30 minutes and a maximum connection time of 3 hours. Based on features such as local time of departure, day of week, and connection time, we estimate a utility, βx_i, associated with each of these itineraries and then sample one of the choices based on the multinomial logit probability model. Flight seating capacities are an important input into our process, because when a flight becomes full during sampling, we remove the corresponding itinerary and update the probabilities for the remaining choices. To eliminate biases when flights become full, we randomly order passengers before sampling itinerary choices.

In a related context, Coldren, Koppelman and others have applied discrete choice models to estimate airline itinerary shares in [5] and [6] from booking data. In the airline itinerary shares estimation problem, the goal is to predict the share of passenger demand for a market (i.e. all air travel from an origin to destination) that will utilize each of a set of available itinerary choices. Thus, the itinerary shares problem is more general in that all *carrier-routes* for a market are considered simultaneously. In our problem, we are only interested in the estimated choice probabilities for a single *carrier-route*, because the DB1B data effectively splits the market demand among *carrier-routes*. Nonetheless, the success of the Coldren and Koppelman models suggest that a discrete choice model is reasonable in this context.

The linear-in-parameter utility function we use for our discrete choice model includes parameters for the interaction of the local time of departure and day of week as well as parameters for a piecewise linear function of connection time (to model the disutility associated with short and long

connection times). Departure time is split into 4 hour blocks: 1:01 – 5:00am, 5:01 – 9:00am, 9:00am – 1:00pm, ... , and 9:01pm – 1:00am; and each day of week is represented distinctly. We fix the utility associated with departures between 5:01am and 9:00am on Monday to 0 to enable identifiability of the model. The piecewise linear utility for connection time has three parts, corresponding to a mildly increasing utility up to a connection time of 45 minutes, a more dramatically increasing utility up to 60 minutes, and then a mildly decreasing utility beyond 60 minutes. This model is trained with BIOGEME [7] using a quarter of booking data from a single legacy carrier using sampling of alternatives to limit the size of the choice set to 10 alternatives for each observation.

4 Passenger Delay Estimates

Once we have sampled itineraries for each of the passengers represented by T-100 and DB1B, we use the ASQP realized flight schedules to determine which itineraries are disrupted due to missed connections (less than 15 minutes of connection time) or flight cancellations. We then recover these passenger itineraries by rebooking them from the point of disruption to the final destination of the itinerary. The recovery heuristic we use is an extension of the Passenger Delay Calculator described in [4]. In our approach, we set a default of 8 hours of delay for daytime departures (5:01am to 5:00pm) and a default of 16 hours of delay for evening departures (5:00pm to 5:00am). If a passenger is unable to be rebooked to his or her final destination with an expected delay less delay than the default, we assume the delay equals the default. We include this cap to ensure that our passenger delay estimates are conservative, since ASQP does not include all flight options (e.g. non-reporting carriers). In attempting to rebook the passenger, we consider itineraries utilizing the airlines on the passengers original *carrier-route*, along with any sub-contracted or parent airlines, before considering rebooking on a competing airline. Using this approach, we have estimated that in 2007, over 15 billion minutes of passenger delays led to $9.4 billion in economic costs. Of the 15 billion minutes of passenger delay, approximately 50% are due to flight delays affecting non-disrupted passengers, 16% are due to a missed connection, and 34% are due to a flight cancellation.

References

[1] A. Tomer and R. Puentes, "Expect Delays: An Analysis of Air Travel Trends in the United States", Metropolitan Infrastructure Initiative Series, Brookings, October 2009.

[2] U.S. Congress Joint Economic Committee, "Your Flight Has Been Delayed Again: Flight Delays Cost Passengers, Airlines, and the U.S. Economy Billions", May 2008.

[3] Air Transport Association, "2008 Economic Report", 2008.

[4] S. Bratu and C. Barnhart, "An analysis of passenger delays using flight operations and passenger booking data", *Air Traffic Control Quarterly* 13(1), 1-27 (2005).

[5] G.M. Coldren, F.S. Koppelman, K. Kasurirangan, and A. Mukherjee, "Modeling aggregate air-travel itinerary shares: logit model development at a major US airline", *Journal of Air Transport Management* 9, 361-369 (2003).

[6] G.M. Coldren and F.S. Koppelman, "Modeling the competition among air-travel itinerary shares: GEV model development", *Transportation Research Part A* 39, 345-365 (2005).

[7] M. Bierlaire, "BIOGEME: a free package for the estimation of discrete choice models", *Proceedings of the 3rd Swiss Transportation Research Conference*, Ascona, Switzerland, 2003.

Heuristics for special instances of the Double TSP With Multiple Stacks

Angel Felipe

Department of Statistics and Operations Research

University Complutense of Madrid

M. Teresa Ortuño

Department of Statistics and Operations Research

University Complutense of Madrid

Gregorio Tirado

Department of Statistics and Operations Research

University Complutense of Madrid

Email: gregoriotd@mat.ucm.es

The Double Traveling Salesman Problem with Multiple Stacks (DTSPMS) is a pickup and delivery problem that was recently introduced in [7]. It is an interesting vehicle routing problem in which some precedence and loading constraints induced by the use of multiple compartments in the containers of the vehicles must be respected. This problem comes from a real application in the field of logistics and has been receiving increasing attention during the last years.

The DTSPMS consists on a set of orders that must be served at minimum cost. Each order is determined by a pickup location, where some goods must be picked up, and a delivery location, where those goods must be delivered. All pickup locations are placed in one region and all delivery locations are placed in another region that is usually far away from the first one. Then, a pickup route and a delivery route visiting all pickup and delivery locations, respectively, must be determined. The intermediate transportation between the depot of the pickup region and the depot of the delivery region is assumed to be constant.

The container of the vehicle is divided into several compartments, allowing the items to be organized in several independent rows, and the vehicle used to serve the given orders is rearloaded. Furthermore, for safety and insurance reasons repacking is not allowed during pickups or deliveries. As a consequence, each row behaves as a LIFO stack (Last-In-First-Out), and thus

the possible delivery sequences depend on the chosen pickup sequence, inducing the appearance of some precedence constraints on the items to be collected. The available containers have fixed dimensions, so the number of available stacks and their capacity are also fixed.

In real life situations the items to be picked up and delivered are usually Euro Pallets, which can store different kinds of products but have standard dimensions. For this reason, and also for simplicity, it will be assumed that all items are uniform and each order consists of one single item.

Hence, a solution of a DTSPMS instance consists of a pickup route, stating how pickup locations of the first region are visited, a delivery route, stating how delivery locations of the second region are visited, and a loading plan, stating how collected items are stored in the container of the vehicle. One such solution is feasible if all given orders are served and all precedence and capacity constraints in the container of the vehicle are respected. The objective of the problem is to find a feasible solution in which the sum of total traveled distances in both regions is minimized.

The DTSPMS is a pickup and delivery problem (see [6] for a recent survey on this kind of problems) that generalizes the well known Traveling Salesman Problem (TSP) and as a consequence it is NP-hard as well. Two different exact approaches are proposed in [1] and [4], but using these methods instances with only up to 21 orders could be solved to optimality. To solve real-sized instances different heuristic algorithms have been proposed in [2], [3] and [7], with which instances with 33, 66 and 132 orders could be approached. Iterated Local Search, Tabu Search, Simulated Annealing and Large Neighborhood Search heuristics are applied to the DTSPMS in [7] and several neighborhood structures adapted to the problem that are embedded into a Variable Neighborhood Search approach (introduced in [5]) are developed in [2] and [3]. To the best of the authors' knowledge, the best results for real sized instances, that are obtained by a Variable Neighborhood Search approach, are given in [3].

In this paper we give some alternative ideas to design a different algorithm for the problem and propose the generation of instances of different kinds to better evaluate the performance of existing approaches.

When a standard local search procedure is used, the search process moves from one feasible solution to another feasible solution. The new solution is accepted if it is better than the previous one or if it has some different features that may guide the search process to unexplored regions of the solution space. However, if the problem is highly constrained, as it is the case of the DTSPMS, imposing ALL constraints to be verified at each iteration may restrict too much the search process and may make it difficult to reach new regions of the solution space. To avoid this inconvenience we propose the temporal relaxation of some capacity and/or precedence constraints, obtaining a more flexible search process.

On the one hand, if capacity constraints are slightly relaxed and α extra units per compartment are used in the container of the vehicle we obtain what we call α-*feasible* solutions. On

the other hand, if some precedence constraints are removed we obtain what we call *potential* solutions. Combining both ideas we can also consider *α-potential* solutions. The moves to define neighborhood structures for these *intermediate non-feasible* solutions are now much more simple, introducing diversification into the search process and making it more flexible. However, the use of these solutions introduces infeasibilities into the search process, and they should be controlled in order to guarantee the obtaining of final feasible solutions. For this purpose we propose different procedures: reduction algorithms, that rearrange the loading plan of the solutions to eliminate capacity infeasibility, and projection operators, that modify both the loading plan and the routes of the solutions to eliminate both capacity and precedence infeasibilites. Reduction algorithms do not add any extra cost to the initial solutions but do not guarantee the obtaining of feasible solutions in all cases. On the other hand, projection operators always provide final feasible solutions but may add some extra cost to the initial solutions. Both procedures are used in combination to deal with infeasibility.

To evaluate *how infeasible* a solution is, different infeasibility measures are proposed, as for instance the number of orders exceeding maximum capacity, the number of pairs of orders violating precedence constraints, etc. The search process with intermediate non-feasible solution is guided by a new objective function that is a weighted sum of the cost and the proposed infeasibility measures. The weights associated to each term are updated dynamically during the search process to guide the search towards good feasible solutions, but reduction algorithms and projection operators are applied as well to guarantee the obtaining of final feasible solutions. Following this methodology a new algorithm called *Exterior Search* is designed.

To test the proposed heuristics we have generated several sets of instances with different sizes (33, 66 and 132 orders), but we also propose the use of special instances with different characteristics concerning the distribution of orders and the magnitude of distances to better evaluate the performance of heuristics. In contrast with the instances that are usually used in the literature, in which the orders are uniformly distributed, we have generated several sets of instances in which some pickup and/or delivery locations are grouped in several clusters. Clusters may be formed in one or both regions of the problem, the locations may be uniformly distributed among the clusters or clusters with different number of locations may be considered, the size of the clusters and the magnitude of the distances between them may change from one set of instances to another, etc. To take advantage of the special structure of these new instances with clusters some modifications can be introduced into the previously proposed algorithms to try to improve their performance. This study, that is still being developed, provides more information about the heuristics and may allow to determine which algorithms behave better for each kind of instances.

References

[1] C. Archetti, H. Petersen and M.G. Speranza, "Exact solution approaches to the double travelling salesman problem with multiple stacks", *Technical Report, Quaderno di Ricerca n. 313*, Dipartimento di Metodi Quantitativi, Università di Brescia (2008).

[2] A. Felipe, A., M.T. Ortuño and G. Tirado, "New neighborhood structures for the Double Traveling Salesman Problem with Multiple Stacks", *TOP* 17, 190-213 (2009).

[3] A. Felipe, A., M.T. Ortuño and G. Tirado, "The double traveling salesman problem with multiple stacks: A variable neighborhood search approach", *Computers and Operations Research* 36, 2983-2993 (2009).

[4] R. Lusby, J. Larsen, M. Ehrgott and D. Ryan, "An Exact Method for the Double TSP with Multiple Stacks", *Technical Report 2*, DTU, Denmark (2009).

[5] N. Mladenović and P. Hansen, "Variable Neighbourhood Search", *Computers and Operations Research* 24, 1097-1100 (1997).

[6] S. Parragh, K. Doerner and R. Hartl, "A survey on pickup and delivery models, Parts I & II", *Journal für Betriebswirtschaft* 58, 21-51 and 81-117 (2008).

[7] H.L. Petersen and O.B.G. Madsen, "The double travelling salesman problem with multiple stacks - Formulation and heuristic solution approaches", *European Journal of Operational Research* 198(1), 139-147 (2009).

A Combined GPS/Stated Choice Experiment to Estimate Values of Crash-Risk Reduction

Simon Fifer

Institute of Transport & Logistics Studies, Faculty of Business

University of Sydney

Stephen Greaves (corresponding author)

Institute of Transport & Logistics Studies, Faculty of Business

University of Sydney

Email: s.greaves@itls.usyd.edu.au

&

Richard Ellison

Institute of Transport & Logistics Studies, Faculty of Business

University of Sydney

1 Introduction

Recent estimates suggest motor vehicle accidents cost the Australian economy around $17 billion per year [1]. While both the number of crashes and crash rates (crashes/kilometre) has reduced dramatically in the last thirty years, latest statistics show that 1,463 persons were killed on Australian roads in 2008, with 395 killed in the state of New South Wales alone. More worryingly, it appears reductions may have stagnated in recent years, leaving policy-makers searching for other options that might lead to significant drops in crash rates. While engineering-based methods for both roadway infrastructure and vehicles, and regulation and enforcement will continue to play a critical role in future road-safety initiatives, an area of growing interest is the use of charging mechanisms that capture the variable risk effects of the kilometres driven [2]. The notion here is that through incorporating known correlates of increased crash risk (e.g., kilometres driven, night-time driving, speeding) directly into the charges, motorists will be incentivised to change behaviour reducing the overall risk and societal costs of accidents [3].

Within this context, the current paper reports on a study into the hypothetical/stated response of motorists to a kilometre-based charging regime that incorporates elements of risk, specifically night-time driving and speeding. Hypothetical responses are gathered through a Stated Choice (SC) experiment that pivots off actual driving behaviour collected using an in-vehicle Global Positioning System (GPS) device over five weeks [4]. This provision of greater reality using revealed preference (RP) information ensures that the alternatives in the SC experiment are embedded in reality, providing motorists with (in theory) a more realistic context for their choices. In the SC experiment, participants

are asked to trade-off financial rewards against reductions in kilometres driven, night-time driving and speeding for different trip purposes. In turn, this information is used to estimate values of crash-risk reduction and help guide a proposed charging regime that will be used to empirically assess changes in behaviour later this year.

2 Approach

The GPS Phase

In the GPS phase, 148 motorists were recruited and agreed to install a GPS device in their vehicle for several weeks. In addition, motorists were required to complete an online prompted-recall survey, providing trip information such as who was driving, the purpose of the trip, number of passengers and whether any intermediate stops were made (see Figure 1). While full details of the GPS phase are provided by the authors in [4], the data were generally of a very high quality aside and of the original 148 drivers, only 8 have dropped out, 4 of which were due to problems with vehicles.

Establishing the Charging Rates

Per kilometre rates were derived using crash-risk and crash-cost information for New South Wales. While the approach is detailed fully in [5], the main issues were to establish per kilometre rates that were deemed substantial enough to warrant some change in behaviour, while staying within the available project budget. The final rates (shown in Table 1) were therefore derived both through a scientific approach as well as interviews with several participants who completed a pilot study of the processes employed for the study [4].

Table 1: Per Kilometre Charging Rates Used in the Sydney Driving Study (Greaves et al. 2010)

Charging Rates	17-30 Age-Group	31-65 Age-Group
Day - Non Speeding	$0.20	$0.15
Day - Speeding	$0.60	$0.45
Night - Non Speeding	$0.80	$0.60
Night - Speeding	$2.40	$1.20

These rates were combined with the relevant information from the GPS data for each motorist to establish a 'base incentive'. This base incentive represented the starting point from which money would be deducted according to their stated changes in driving. For the 123 motorists who qualified for the charging phase (17 were retained as a control group), the range of base incentives ran from $25 to $870, with an average of $300. For a five-week period, these were generally considered to be significant amounts of money that could potentially be made.

The Stated Choice Experiment

The purpose of the SC experiment was to see explore how respondents might hypothetically change their driving behaviour if they were participating in a kilometre based charging scheme. The SC experiment was implemented for three different trip purposes; work/work-related,

shopping/personal business, and social/recreation. This required some manipulation of the GPS data primarily around the development of GIS-based routines to automatically classify trips into trip tours and reclassify trips returning home to one of these three purposes. In keeping with recent literature on referencing SC experiments to a known experience [8] [9], the SC experiment was designed to pivot off actual trips taken from GPS data collected during the 5 week 'Before' phase. The experiments were based on a choice between maintaining existing trips (the current alternative) and alternatives involving changes to existing trips and receiving a reduced charge (e.g., cancelling trips, reducing speeding, changing time of day). A screenshot of the SC experiment is shown in Figure 2.

A Bayesian efficient design for each trip purpose was generated. This experimental design method was used to produce lower standard errors and provide more reliable parameter estimates for a relatively small sample size [9]. The experimental designs were constructed in Excel, assuming a random uniform distribution of prior parameters, given expected parameters signs. Respondents answered four choice situations for each of the three different trip purposes. Individual trip purpose models will be estimated, although the data may be pooled later for further analysis.

Results

The data collection for the SC experiment has only recently completed, but preliminary insights are that participants found the experiment 'fun' and 'interesting' although many indicate a difficulty in changing behaviour. Basic interim modelling for a sub sample of participants suggests that most parameters are significant and correctly signed. Further analysis is underway to see if these results are consistent with results for the combined sample and will be reported in the full paper.

References

[1] L.B. Connelly and R. Supangan, "The Economic Costs of Road Trauma: Australia, States and Territories", *Accident Analysis and Prevention* 38(6), 1087-93 (2006).
[2] T. Litman, "Distance-Based Vehicle Insurance Feasibility, Costs and Benefits Comprehensive Technical Report." Available from http://www.vtpi.org/dbvi_com.pdf (2008).
[3] J. Zantema, D.H. van Amelsfort, M.C.J. Bliemer, P.H.L. Bovy, "Pay-As-You-Drive (PAYD): A case study into the safety and accessibility effects of PAYD strategies. *CD-ROM proceedings of the 87th meeting of the Transportation Research Board*, Washington DC, January 2008.
[4] S.P. Greaves, S. Fifer, R. Ellison and G. Germanos, "Development of a GPS/Web-based Prompted-Recall Solution for Longitudinal Travel Surveys", *CD-ROM Procs of the 2010 Annual Meeting of the Transportation Research Board*, Washington DC, 2010.
[5] S.P. Greaves and S. Fifer, "Development of a kilometre-based charging regime to encourage safer driving practices", *CD-ROM Procs of the 2010 Annual Meeting of the Transportation Research Board*, Washington DC, 2010.
[6] Rose, J.M., Bliemer, M.C.J., Hensher, D.A. and Collins, A.T. (2008). Designing efficient stated choice experiments in the presence of reference alternatives, Transportation Research Part B: Methodological 42: 395-406.
[7] Transport Data Centre, (2009). 2007 Household Travel Survey: 2009 Release, Sydney: Transport and Infrastructure
[8] Hensher, D. A., (2009), Hypothetical bias, choice experiments and willingness to pay. Fortcoming, Special Issue of Transportation Research B.
[9] Rose, J. and Bliemer, M. (2005). Constructing Efficient Choice Experiments Working Paper, Institute of Transport and Logistics Studies, University of Sydney.
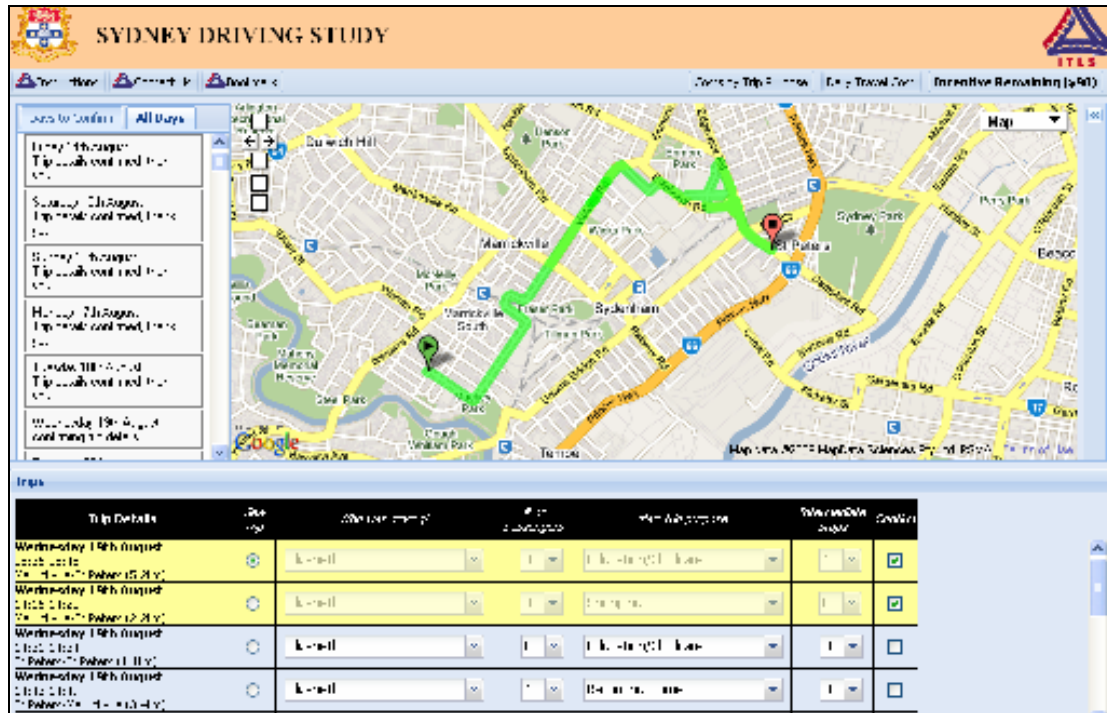
Figure 1: Example of the Prompted-Recall Interface
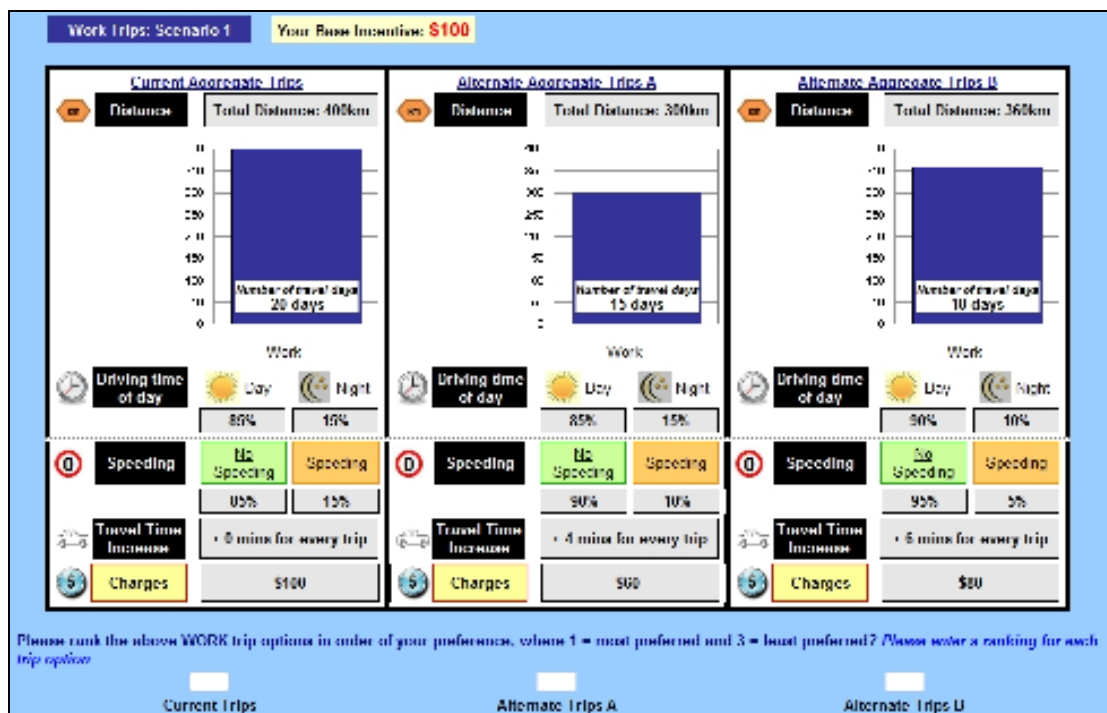
Figure 2: Example Screen from the Stated Choice Survey

GHG Emissions Minimization Vehicle Routing Problem

Miguel Figliozzi

Department of Civil and Environmental Engineering

Portland State University

PSU-CEE, PO Box 751,

Portland, OR 97207-0751, USA

Email: figliozzi@pdx.edu

1 Introduction

The fast rate of commercial vehicle activity growth over recent years and the higher impact of commercial vehicles are increasing preexisting concerns over their cumulative effect in urban areas. In particular, environmental, social and political pressures to limit the impacts associated with greenhouse gas (GHG) emissions and our dependence on fossil fuels is mounting rapidly. A key challenge for public transportation agencies companies is to improve the efficiency of urban freight and commercial vehicle movements while ensuring environmental quality, livable communities, and economic growth.

Private companies are also interested in reducing GHG emission not only for marketing purposes, i.e. the more favorable social perception towards companies that are "greening" their operations, but also for economic reasons. The level of GHG emissions is a proxy for fuel consumption in diesel engines. In addition, in the near future it is likely that GHG emissions will have a clear monetary value, i.e. $/kg for CO_2 emitted, in most countries. For example, CO_2 emissions will have a clear economic value under carbon tax or cap and trade emissions system initiatives, which is already implemented in Europe and currently under study by the United States and other governments.

In terms of transportation and logistics sustainability, it is also important to quantify and minimize GHG emission.

This research aims to formulate, study, and solve a new vehicle routing problem where the minimization of emissions is the primary objective or is part of a generalized cost function. In addition, departure times and travel speeds become decision variables. To the best of the author's knowledge, there is no research or formulation that minimizes GHG vehicle emissions when designing routes in congested environments with time-dependent travel speeds, hard time windows, and capacity constraints. This creates a new type of VRP which is denoted the GHG Emissions Vehicle Routing Problem or simply EVRP.

2 Background and Literature Review

There is an extensive literature related to vehicle emissions and several laboratory and field methods are available to estimate vehicle emissions rates [1]. Urban freight is responsible for a large share, or in some cities the largest share, of unhealthy air pollution in terms of sulphur oxide, particulate matter, and nitrogen oxides in urban areas such as London, Prague, and Tokyo [2, 3]. Research in the area of city logistics have long recognized the need for a balanced approach that aims to reduce shippers and carriers' logistics costs while alleviating a community's traffic congestion and environmental problems. [4-6]

In terms of GHG, Research indicates that carbon dioxide (CO_2) is the predominant transportation GHG and is emitted in direct proportion to fuel consumption, with a variation by type of fuel [7]. For most vehicles, fuel consumption and the rate of CO_2 per mile traveled decreases as vehicle operating speed increases up to approximately 55 or 65 mph and then begins to increase again [7]; hence, the relationship between emission rates and travel speed is not linear.

Congestion has a great impact on vehicle emissions and fuel efficiency. In real driving conditions, there is a rapid non-linear growth in emissions and fuel consumption as travel speeds fall below 30 mph [8]. CO_2 emissions double on a per mile basis when speed drops from 30 mph to 12.5 mph or when speed drops from 12.5 mph to 5 mph. Frequent changes in speed, i.e. stop and go traffic conditions, increases emission rates because fuel consumption is a function of not only speed but also acceleration rates [9]. These results were obtained using an emission model and freeway sensor data in California and weighted on the basis of a typical light-duty fleet mix in 2005. The volume of emissions per mile is a function of the speed profile from the departure time until reaching destination.

In congested urban areas with significant speed changes due to recurrent congestion, e.g. predictable low speeds due to capacity constraints at peak hours, departure time must be considered when designing EVRP routes. The Time Dependent Vehicle Routing Problem (TDVRP) takes into account that links in a network have different costs or speeds during the day. Typically this is used to represent varying traffic conditions. The TDVRP was originally formulated by Malandraki and Daskin

[10]. Time dependent models are significantly more complex and computationally demanding than static VRP models; recent approaches to solve the TDVRP can be found in [11-13]. Palmer [14] studies the minimization of CO_2 emissions utilizing real network data and shortest paths of Surrey county in the U.K.. However, Palmer's methodology does not allow for time-dependent speeds or multi-stop routes.

TDVRP instances are more data intensive than static VRP instances but their solution is likely to achieve environmental benefits in congested areas albeit in an indirect way because emissions are not directly optimized [15]. Other researchers have conducted surveys that indicate that substantial emission reductions can be obtained if companies improve the efficiency of routing operations [16, 17]. Woensel et al. [18] used queuing theory to model the impact of traffic congestion on emissions and recommend that private and public decision makers should take into account the high impact of congestion on emissions. More recently Maden et al. [20] presented the results of a case study in the U.K.; the utilization of time-dependent vehicle routing algorithms can lead to an average reduction of 7% CO_2 emissions when comparing against simpler methods that do not take travel time variations into account. Goodchild and Sandoval [19] discuss the factors that affect emissions in urban areas and potential solution methods, case studies, and public policy applications. However, no formulation, solution methods, or results are provided. To the best of the author's knowledge, there is no published research that deals with the formulation, properties, or solution approaches for the EVRP. The EVRP considered in this research has time windows and capacity constraints as well as time-dependent travel times. The paper deals with a static problem, the dispatcher is assumed to know the impact of recurrent congestion on travel speeds, i.e morning/evening rush hours. For example, in a practical case, the dispatcher/carrier designs the routes the night before the route is serviced; the carrier is committed to visit a specific group customers within a pre-determined and hard time-window. The heuristic is based on an initial construction heuristic followed by improvement phases. New heuristics to minimize emissions are proposed utilizing lemmas and properties of the travel time and emission functions.

3 Contributions

This research is different from previous research in several aspects: (a) a new vehicle routing problem is formulated, (b) an efficient heuristic is presented, (c) in the presented formulation travel speed is a decision variable, (d) using the well known Solomon instances, new problems are proposed to evaluate GHG emissions and other routing costs. In addition to results obtained using modified Solomon instances, results based on real-world instances using real-life travel times from a congested interstate highway network in Portland, Oregon, will also be presented.

Preliminary results indicate that there may be significant emissions savings if commercial vehicles are routed taking emissions into consideration. The results also indicate that in congested areas, it may be possible to reduce GHG emissions with a minimal or null increase in routing costs. However, the results indicate that congestion impacts on emission levels are not uniform. The route characteristics and dominant constraint type (hard time window or capacity constraints) seem to play a significant effect on emissions levels.

References

1. Ropkins, K., et al., *Real-World Vehicle Exhaust Emissions Monitoring: Review and Critical Discussion.* Critical Reviews in Environmental Science and Technology, 2009. **39**(2): p. 79-152.

2. OECD, *Delivering the Goods 21st Century Challenges to Urban Goods Transport.* 2003: OECD Organisation for Economic Co-operation and Development.

3. Crainic, T.G., N. Ricciardi, and G. Storchi, *Models for Evaluating and Planning City Logistics Systems.* Transportation Science, 2009. **43**(4): p. 432-454.

4. Quak, H.J. and M.B.M. de Koster, *Exploring retailers' sensitivity to local sustainability policies.* Journal of Operations Management, 2007. **25**(6): p. 1103.

5. Taniguchi, E., R.G. Thompson, and T. Yamada, *Predicting the effects of city logistics schemes.* Transport Reviews, 2003. **23**(4): p. 489-515.

6. Crainic, T.G., N. Ricciardi, and G. Storchi, *Advanced freight transportation systems for congested urban areas.* Transportation Research Part C: Emerging Technologies, 2004. **12**(2): p. 119.

7. ICF, *Assessment of Greenhouse Gas Analysis Techniques for Transportation Projects, NCHRP Project 25-25 Task 17, ICF Consulting.* 2006: Fairfax, VA.

8. Barth, M. and K. Boriboonsomsin, *Real-World Carbon Dioxide Impacts of Traffic Congestion.* Transportation Research Record 2058, 2008: p. 163-171.

9. Frey, H., N. Rouphail, and H. Zhai, *Link-Based Emission Factors for Heavy-Duty Diesel Trucks Based on Real-World Data.* Transportation Research Record, 2008. **2058**: p. 23-332.

10. Malandraki, C. and M.S. Daskin, *Time-Dependent Vehicle-Routing Problems - Formulations, Properties And Heuristic Algorithms.* Transportation Science, 1992. **26**(3): p. 185-200.

11. Malandraki, C., *Time dependent vehicle routing problems: Formulations, solution algorithms and computational experiments.* 1989, Ph.D. Dissertation, Northwestern University: Evanston, Illinois.

12. Jung, S. and A. Haghani, *Genetic Algorithm for the Time-Dependent Vehicle Routing Problem.* Transportation Research Record, 2001. **1771**(-1): p. 164-171.

13. Figliozzi, M.A. *A Route Improvement Algorithm for the Vehicle Routing Problem with Time Dependent Travel Times.* in *Proceeding of the 88th*

Transportation Research Board Annual Meeting CD rom. 2009. Washington, DC, January 2009, also available at http://web.cecs.pdx.edu/~maf/publications.html.

14. Palmer, A., *The development of an integrated routing and carbon dioxide emissions model for goods vehicles.* PhD Thesis Cranfield University November 2007, 2008.

15. Sbihi, A. and R.W. Eglese, *Combinatorial optimization and Green Logistics.* 4OR: A Quarterly Journal of Operations Research, 2007. **5**(2): p. 99-116.

16. Léonardi, J. and M. Baumgartner, *CO2 efficiency in road freight transportation: Status quo, measures and potential.* Transportation Research Part D, 2004. **9**(6): p. 451-464.

17. Baumgartner, M., J. Léonardi, and O. Krusch, *Improving computerized routing and scheduling and vehicle telematics: A qualitative survey.* Transportation Research Part D, 2008. **13**(6): p. 377-382.

18. Woensel, T.V., R. Creten, and N. Vandaele, *Managing the environmental externalities of traffic logistics: The issue of emissions.* Production and Operations Management, 2001. **10**(2): p. 207-223.

19. Goodchild, A. and F. Sandoval. *Evaluating Emission Reductions and Trade-offs in Urban Pickup and Delivery Systems*. in *Odysseus Workshop 2009, International Workshop on Freight Transportation and Logistics, May 26-29 2009*. 2009. Cesme, Turkey: Conference CD-rom.

20. Maden, W., R.W. Eglese, and D.P. Black, *Vehicle Routing and Scheduling with Time Varying Data: A Case Study.* 2009.

Estimate the value of ITS information in urban freight distribution

Marta Flamini

Data Management S.p.A.

Marialisa Nigro

Dipartimento di Scienze dell'Ingegneria Civile

Università Roma Tre

Dario Pacciarelli

Dipartimento di Informatica e Automazione

Università Roma Tre, 79 Via della Vasca Navale, 00146, Rome, Italy

Email: pacciarelli@dia.uniroma3.it

1 Introduction

Intelligent Transportation Systems (ITS) can play a key role to optimize the organization of intermodal platforms and to reduce the impact of freight traffic on urban congestion. In particular advanced tracking systems, such as radio frequency identification (RFID) and global positioning system (GPS), offer a new opportunity to collect information about network traffic conditions that can be used both in real time, to locate the position of a vehicle, and off line to estimate the travel time of each element of the network with high level of precision and reliability. However, while the cost of implementing such measurement systems can be easily computed, estimating the value generated by advanced tracking systems is more difficult [3].

The objective of this work is to develop a quantitative method to estimate the added value generated by information of advanced tracking systems in urban freight distribution. We report on an application of our method to the retail distribution of perishable goods. The perishable goods market is characterized by the short life time of products and by the high cost associated to late deliveries. As a consequence, there is a need for reliable and accurate data on the road network to plan punctual deliveries at sustainable cost.

2 Research methodology

This section describes the procedure adopted for estimating the value of information for the urban distribution problem considered. The basic idea behind the procedure is that the discrepancy

between planned and implemented solutions is mainly due to the mismatch between the observed data, used to build the planned solution, and the actual travel times occurring in real time. In other words, the actual travel time t_{ij} for a link (i, j) can be expressed as $t_{ij} = d_{ij} + s_{ij}$, where d_{ij} is the deterministic part and s_{ij} is a stochastic variable due to perturbation events on transport demand and supply. The deterministic part d_{ij} is the desired value for solving the freight distribution problem, such as the mean value of t_{ij} or a value achieved with a given probability ψ (i.e., such that the probability of the event $t_{ij} \leq d_{ij}$ is ψ). In practice, the value d_{ij} is estimated by collecting measures of t_{ij} on the network, which can be affected by measurement errors. We let $t_{ij}^{obs} = d_{ij}^{obs} + s_{ij}^{obs}$ be the estimate of t_{ij}, where d_{ij}^{obs} and s_{ij}^{obs} are the observed values of d_{ij} and s_{ij}, respectively. We call *discrepancy* the stochastic variable $\delta_{ij} = t_{ij}^{obs} - d_{ij}$. If t_{ij}^{obs} is a rough estimate of t_{ij}, then the discrepancy can be much larger than the inherent stochasticity of the travel time, i.e., $|\delta_{ij}| >> |s_{ij}|$.

Collecting more reliable information may help to produce a better estimate t_{ij}^{est} of t_{ij}, i.e., an estimate such that $|t_{ij}^{est} - d_{ij}| << |t_{ij}^{obs} - d_{ij}|$. The value of such information is related to the improved performance of the system that would have been achieved if the planned solution was built using the more reliable t_{ij}^{est} instead of t_{ij}^{obs}. Since the discrepancy may vary over the different routes to be traversed, we introduce an aggregated value ε that we call the *unreliability* of the data set.

The procedure computes the value of information with reference to a given solution algorithm $\mathcal{A}$. Given a data set affected by an unreliability ε, we let $\sigma^p(\varepsilon)$ be the planned solution obtained with $\mathcal{A}$ on such data set, and $\sigma^h(\varepsilon)$ be the associated historical solution, obtained by using the same value for the decision variables as in $\sigma^p(\varepsilon)$ but using the real data with unreliability $\varepsilon = 0$. In other words, $\sigma^h(\varepsilon)$ takes into account the implementation of $\sigma^p(\varepsilon)$ in practice. The performance of $\sigma^h(\varepsilon)$ is a stochastic variable associated to the decisions taken by using a data set affected by unreliability ε, and we let $\pi(\varepsilon)$ be the mean value of the performance achieved by $\sigma^h(\varepsilon)$ for a given ε.

Applying the above procedure for different values of ε and different solution algorithms, we get a curve $\pi(\varepsilon)$ for each algorithm being used. If a new advanced tracking systems allows to decrease the information unreliability from a value ε_2 to $\varepsilon_1 < \varepsilon_2$, the associated performance improvement is then $\pi(\varepsilon_1) - \pi(\varepsilon_2)$. This value depends on the chosen solution algorithm, and it is intuitive that different algorithms may have different degrees of sensitivity to data set unreliability. Therefore, when using more reliable data it can also be profitable to develop novel algorithms able to take advantage from more reliable information. In other words, it is important to assess the impact of ITS in combination with different (simple and advanced) vehicle routing algorithms. An advanced tracking system in combination with a robust routing algorithm with poor performance will produce a smaller benefit than in combination with a less robust but more performing vehicle

routing algorithm. It is therefore worth paying the cost of implementing the new tracking system and the new algorithm if the whole system generates sufficient ROI (Return On Investment).

3   Problem description

The retail distribution problem addressed in this work has been formulated as a vehicle routing problem with soft time windows for the deliveries and time dependent travel times. The objective function includes the transportation costs and the cost of late deliveries, i.e., the sum over every route $\rho_i \in \rho$ of: (i) the fixed cost $f_{v(\rho_i)}$ associated to the usage of vehicle $v(\rho_i)$, (ii) the variable cost $c_i(\rho_i, s_i(\rho_i))$ associated to length of route ρ_i and to speed s_i maintained along the route, and (iii) the penalty cost $\sum_{r \in \rho_i} w_r(\rho)$ for late deliveries.

$$\min \sum_{i=1}^{|\rho|} (f_{v(\rho_i)} + c_i(\rho_i, s_i(\rho_i))) + \sum_{r=1}^{R} w_r(\rho). \tag{1}$$

Each retailer r requests a certain quantity of goods d_r to be delivered within a given time window $[t_r, T_r]$. A vehicle arriving at time $a_r > T_r$ at retailer r incurs the penalty cost w_r for late delivery. This penalty depends on the probability p_r that the delivery is refused by the retailer. We assume $p_r = 0$ for on-time deliveries and up to a small delay τ_{min}, i.e., $a_r \leq T_r + \tau_{min}$. The delivery is refused with probability $p_r = 1$ over a delay τ_{max}, and increases linearly from 0 to 1 in the time window $[\tau_{min}, \tau_{max}]$. If the delivery is refused, the merchandise is returned to the logistic platform and delivered the next day, with an associated cost corresponding to the goods depreciation γ_r. Time-dependent travel times have been taken into account to represent different traffic conditions during the planning horizon and to effectively plan the deliveries. The solution approach consists of the development of a simple constructive heuristic and different tabu search algorithms. The first tabu search procedure (hereinafter called ST or *standard tabu search*) implements the main features of the TABUROUTE algorithm introduced by Gendrau, Hertz and Laporte [2]. The second tabu search procedure (hereinafter called AD or *advanced tabu search*) differs from ST for the definition of a larger neighborhood taking into account information about the geographical position of customers and routes.

4   Results

We focus on a real case study, namely an intermodal logistic platform LP located in the suburban area of Rome (Italy), and in particular on the distribution of fresh products in the historical center. The network consists of 250 centroids, 425 nodes and 2346 oriented links.

The distribution of merchandise takes place from 4:00 am to 11:00 am. In order to model the traffic conditions within this time window, about 280.000 vehicles are generated on the net-

work considering a typical variable demand profile. For each hour, link travel times are obtained by simulation using dynamic assignment model where transport demand can change during the simulation interval. For the dynamic simulation the DYNAMEQ model [1] has been used. As a consequence, the travel times between each pair of retailers, as well as between each retailer and the logistic platform LP, are time-dependent and can be represented by a vector where each component is associated to a certain time slice. These travel times values are considered as the actual traffic conditions in the network. Unreliable data have been generated by perturbing these data with different degrees of unreliability, thus obtaining several unreliable data sets.

Preliminary results, obtained by using the constructive heuristic and the two tabu search algorithms AD and ST, show that AD outperforms the other algorithms in both cost function value and resilience to errors in the input data. As far as the value of information is concerned, as expected, there is a clear benefit in using detailed and highly reliable data with respect to aggregated and unreliable data. When the unreliability ε increases, it is still beneficial to use an advanced algorithm, able to achieve good performance for a large range of perturbation. For large ε there is no big convenience in using detailed information for the solution of the vehicle routing problem since the results are almost the same with more aggregated information, e.g., with a small number of time slices.

Future developments of this work will be possible when practical measures on the network links will be available, and will address the design of the most suitable distributions for the link travel time errors and the definition of the right combination of levels of input data aggregation, information reliability and algorithm to be used in practice.

Acknowledgements

References

[1] Florian, M., Mahut, M., Tremblay, N., "A simulation-based dynamic traffic assignment model: Dynameq", *Proceedings of the First International Symposium on Dynamic Traffic Assignment DTA2006*, Institute for Transport Studies - University of Leeds (2006).

[2] Gendreau M., A. Hertz, and G. Laporte, "A Tabu Search Heuristic for the Vehicle Routing Problem", *Management Science* 40, 1276-1290 (1994).

[3] Langer, N., Forman, C., Kekre, S. and A. Scheller-Wolf, "Assessing the impact of RFID on return center logistics", *Interfaces* 37(6), 501-514 (2007).

The Static Repositioning Problem in a Bike-Sharing System

Iris Forma

Department of Industrial Engineering

Tel Aviv University

Tal Raviv

Department of Industrial Engineering

Tel Aviv University

Michal Tzur

Department of Industrial Engineering

Tel Aviv University, Tel Aviv, Israel

Email: tzur@eng.tau.ac.il

1 Introduction

Bicycle is an economical and environmentally friendly mean of transportation for short journeys within cities and it is also a very good complementary for longer regional journeys done by means of public transportation. Many cities are currently implementing bike sharing systems - a service that allows people to rent a bicycle for a short period of time from many automatic renting stations scattered around the city. The largest bike-sharing system as of today is Vélib launched in July 2007 in Paris (www.velib.paris.fr). It now consists of 1,700 renting stations and offers some 23,900 bikes for rent. Many other municipalities, including Tel Aviv, are in a process of examining the viability of deploying a bike-sharing system in their cities.

A crucial factor for the success of a bike sharing system is its ability to meet the fluctuating demand for bicycles at each station. In addition, the system should be able to provide enough vacant lockers to allow the renters to return the bicycle at their destinations. Meeting the demand for bicycles and vacant lockers is a particularly challenging problem due to inherent imbalances in the renting and return rates at the various stations. In some cases, the imbalance is temporary, e.g., high return rate in a suburban train station in the morning and high renting rate in the afternoon. In other cases the imbalance is persistent, e.g., relatively low return rate in stations located on top of hills. Satisfying the users' demand subject to such imbalances requires regularly removing bicycles from stations with high

return rates and transferring them to stations with higher demand rates, using a dedicated fleet of light trucks. We refer to this activity as *repositioning* bicycles.

The bicycle repositioning problem can be classified as a variation of the Pickup and Delivery Problem (PDP). According to the survey of Berbeglia et al. [2] on static PDP, the repositioning problem presented here is a many-to-many single commodity PDP with arbitrary vehicle capacities and stochastic demand, on which no studies are available. Another closely related routing problem is the Swapping Problem (SP), first introduced by Anily and Hassin [1]. In the swapping problem, a vehicle of unit capacity needs to ship objects of different types from node in which they are available to other nodes in which they are required, using the shortest route. Anily and Hassin also showed that the problem is NP-Hard and presented a 2.5 approximation algorithm. Hernández-Pérez and Salazar-González [3] introduced the one-commodity pickup and delivery traveling salesman problem (1-PDTSP), which is a variation of the SP by the following characteristics: only one commodity type is allowed, the capacity of the vehicles and the demand at the nodes are arbitrary, and the vehicle route must be Hamiltonian. Hernández-Pérez and Salazar-González [3] presented a branch and cut strategy that is capable of solving instances with up to 40 nodes. Subsequently, Hernández-Pérez and Salazar-González [4] presented heuristic methods for the problem and demonstrated their applicability for instances with up to 500 nodes. Louveaux and Salazar-González [6] considered the 1-PDTSP with stochastic demand where the objective function to be minimized includes a penalty that is proportional to the unsatisfied demand.

In the repositioning problem the objective function is not typical, i.e., minimizing a measure of dissatisfaction of the users of the system. Measuring it properly is an independent challenging problem. In addition, the repositioning problem includes a fleet of several vehicles, rather than just a single one.

2 Modeling and Formulation

Our research focuses on how to reposition bicycles among stations, as a means of improving satisfaction of the users. We assume, what is commly believed, that the primary factor affecting the users' satisfaction in a bike-sharing system is the availability of bicycles at the desired origin stations and the availability of vacant lockers at the desired destination stations. Thus, we define *user dissatisfaction*, to be minimized, as the weighted number of users who abandon stations without receiving service due to lack of bicycles or due to lack of lockers, with respected weights.

The repositioning operations can be carried out in two different modes: one is during the night when the usage rate of the system is rather low; the other is during the day when the status of the system is rapidly changing. We focus on the former, referred to as the *static repositioning problem*. The static repositioning operation has a practical advantage because it allows the repositioning fleet to travel swiftly in the city without contributing to traffic congestion and parking problems. It needs to be solved once at the beginning of each night, based on the status of the system at that time and the demand forecast for the next day.

Kolka and Raviv [5] define and calculate the expected total user dissatisfaction related to a single station, as a function of its initial inventory (number of bicycles available), under some restrictive assumptions. They were able to show that this function is quasi-convex in the number of bicycles available, and conjectured that the function is in fact convex. In this research we build on their results and formulate the static repositioning problem as a Mixed Integer Linear Programs (MILP) that minimizes user dissatisfaction, by chnaging the number of bicycles available at each station through the repositioning operation. We use the following parameters as input to the problem: I is the set of stations, $\eta_i(\cdot)$ is the user dissatisfaction function of each station i, s_i is the initial inventory level at the station before the repositioning operation takes place, say at 1:00am; C_i is the number of lockers installed at station i (capacity); t_{ij} is the traveling time between stations i and j; and L is the total time allowed for the repositioning problem (e.g., 5 hours between 1am and 6am). The repositioning trucks are assumed to be initially located at a depot station indexed by 0. The capacity of each truck (vehicle) v is denoted by k_v.

The decision variables are as follows: x_{ijv} is a binary variable which equals one if truck v travels directly from station i to station j and zero otherwise; y_{ijv} is the number of bicycles carried on truck v during its journey from station i to station j (it is forced to be zero if no such journey exists); and z_{iv} is the number of bicycles unloaded from the truck at station i by truck v. z_{iv} is negative when the truck removes bicycles from the station. The following is a MILP formulation of the problem:

$$\min \sum_i \eta_i \left(s_i + \sum_v z_{iv} \right)$$

Subject to

$$\sum_j x_{ijv} = \sum_j x_{jiv} \qquad \forall v, i \qquad (1)$$

$$z_{iv} + \sum_j y_{ijv} = \sum_j y_{jiv} \qquad \forall v, i \qquad (2)$$

$$\sum_{j,v} x_{ijv} \leq 1 \qquad \forall i \neq 0 \qquad (3)$$

$$y_{ijv} \leq k_v x_{ijv} \qquad \forall v, i, j \qquad (4)$$

$$0 \leq s_i + \sum_v z_{iv} \leq C_i \qquad \forall i \neq 0 \qquad (5)$$

$$\sum_{i,j \in S} x_{ijv} \leq |S| - 1 \qquad \forall v, S \subset I : 0 \notin S \qquad (6)$$

$$\sum_{i,j} x_{ijv} t_{ij} \leq L \qquad \forall v \qquad (7)$$

$$x_{ijv} \in [0,1], \; z_{iv} \in \mathbb{N} \qquad \forall i, j, v \qquad (8)$$

The objective function is the sum of the expected dissatisfaction for the next day in all stations where the initial inventory is equal to the current inventory, plus the net number of bicycles unloaded at the station during the night. Constraint (1) forces trucks flow conservation; Constraint (2) forces bicycle flow conservation and keeps record of the trucks inventories in z_{iv}. Constraint (3) allows at most one visit at a station; Constraint (4) forces truck capacity and relates the x variables with the y's; Station capacity and non-negativity of bicycle inventory is forced by (5); Constraint (6) is a standard sub-tour elimination constraint that allows only tours that start and end at the depot; Constraint (7) forces the repositioning operation to last no more than the shift length, L. Finally, (8) requires the x variables to be binary and the z variables to be integral. Integrality of the y variables is implied by (2).

There are two technical difficulties in implementing this program using a commercial MILP solver. One is the non-linearity of the objective function, which can be resolved using the conjectured convexity of the $\eta(\cdot)$ functions and replacing the non-linear functions with piecewise linear functions that support it. The other is the exponential number of the sub-tour elimination constraints, which can be resolved by employing branch and cut techniques. In our preliminary computational tests, we replaced the sub-tour elimination constraints with a simpler set of constraints inspired by the sequential formulation of Miller et al. [7] for the traveling salesman problem. Running this formulation on a commercial MILP solver (Ilog CPLEX® 10.2), we are able to solve instances with 50 nodes in a reasonable time. We used in this experiment data representing real locations of Vèlib stations. Using a branch and cut framework, we believe somewhat larger instances can be solved in a reasonable time.

References

[1] Anily, S. and Hassin, R. (1992). The swapping problem. *Networks* 22, 419-433, 1992.

[2] Berbeglia, G., Cordeau, J.-F., Gribkovskaia I., and Laporte, G. (2007). Static pickup and delivery problems: a classification scheme and survey. *TOP 15*, 1-31.

[3] Hernández-Pérez, H. and Salazar-González J.-J. (2004a). A branch-and-cut algorithm for a traveling salesman problem with pickup and delivery. *Discrete Applied Mathematics* 145, 126–139.

[4] Hernández-Pérez H. and Salazar-González J.-J, (2004b). Heuristics for the one-commodity pickup-and-delivery traveling salesman problem. *Transportation Science* 38, 245–255.

[5] Kolka, O. and T. Raviv, Dynamic management of stock levels in a bike sharing system, Working paper, 2009.

[6] Louveaux F. and Salazar-González J.-J. (2009), On the one-commodity pickup-and-delivery traveling salesman problem with stochastic demands, *Mathematical Programming* 119, pages 169-194.

[7] Miller, C.E., Tucker, A.W., Zemlin, R.A. (1960). Integer programming formulation of traveling salesman problems. *Journal of the ACM* 7, 326-329.

Households' multiple vehicle ownership and their car usage – An analysis with the nationwide interview survey in Japan

Daisuke FUKUDA

Associate Professor, Department of Civil Engineering

Tokyo Institute of Technology, Japan

Email: fukuda@plan.cv.titech.ac.jp

Michiko KOBAYASHI

Graduate Student, Department of Civil Engineering

Tokyo Institute of Technology, Japan

Tetsuro HYODO

Professor, Department of Logistics and Information Engineering

Tokyo University of Marine Science and Technology, Japan

1   Introduction

The motor vehicle is an essential transport measure in peoples' daily life and their commercial activities (e.g. logistics). In the past, the total number of vehicles, the total number of licensed drivers and the total mileage were steadily increasing together with the economic growth in Japan. In addition, the government has expected that the growth would continue for the future. Recently, however, with the various structural changes (e.g. falling birthrate, change in peoples' life style, aging population, change in the gasoline price and global warming), the structure of car market has been dramatically changed. Especially, there are two big changes: a decrease in total mileage and the increase in small-sized vehicles (SV) while the number of middle-sized vehicles (MV) is not constantly increasing in Japan (Fig. 1 and Fig. 2). Generally, when households own a single SV or more than 2 any vehicles, regardless of the car type, the average mileage per household becomes lesser. Thus, it is possible that there is a relation between the decrease in the total mileage in Japan and the increase in the number of SVs.

With this motivation, the purpose of this study is to analyze the structural relationship between
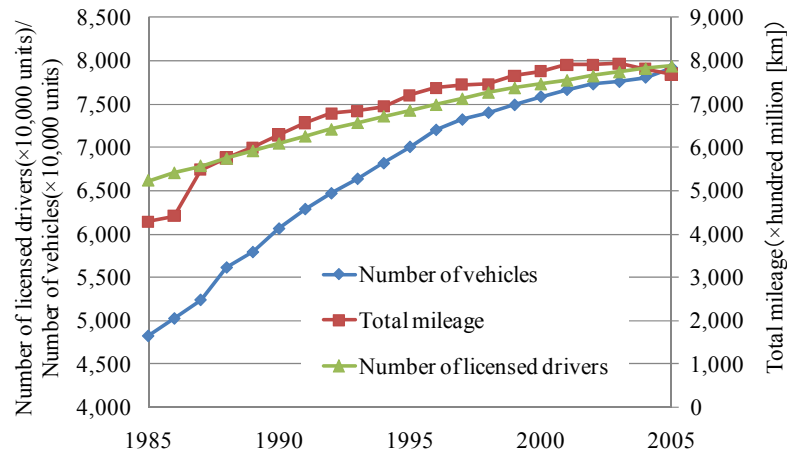
Figure 1: Trend of number of vehicles, number of licensed drivers and total mileage in Japan [3]
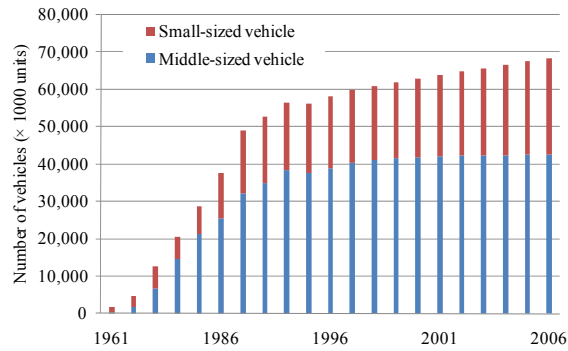
Figure 2: Trend of middle-sized and small-sized vehicles in Japan [3]

the multiple-vehicle ownership and their usage at the household level using a large dataset.

2　Outline of the data

This research uses the data of "Road Traffic Census" conducted by Ministry of Land, Infrastructure, Transport and Tourism in 1999 and 2005. The survey questionnaire includes some attributes of the household, the number of vehicles and their usage, and some specific characteristics of each vehicle (e.g. make of vehicle, daily mileage for a weekday and a weekend). As for the data of socio-economic characteristics, we further complement other data sources (e.g. population density, gasoline price, worker's average income and the number of stations). Please note that gasoline prices are different across regions or cities in Japan. Therefore, it can be incorporated into explanatory variables though only for two year cross-section dataset are applied.

Table 1 represents the summary statistics of the data set. Table 2 outlines the cross-tabulation

Table 1: Summary statistics of the data set

		Year 1999		Year 2005		Total	
	Sample Size	523,016		475,382		998,398	
Vehicles owned	Total	724,853		721,858		1,446,711	
	MV	570,458		504,880		1,075,338	
	SV	154,395		216,978		371,373	
Vehicles owned per household	1 vehicle	370,856	(70.9)	303,617	(63.9)	674,473	(67.6)
	2 vehicles	113,107	(21.6)	116,397	(24.5)	229,504	(23.0)
	More than 3	39,053	(7.5)	55,368	(11.6)	94,421	(9.5)
Households owning MVs	Total	454,982	(87.0)	389,786	(82.0)	844,768	(84.6)
	1 vehicle	362,471	(69.3)	299,535	(63.0)	662,006	(66.3)
	More than 2	92,511	(17.7)	90,251	(19.0)	182,762	(18.3)
Households owing SV	Total	141,357	(27.0)	183,741	(38.7)	325,098	(32.6)
	1 vehicle	129,661	(24.8)	155,625	(32.7)	285,286	(28.6)
	More than 2	11,696	(2.2)	28,116	(5.9)	39,812	(4.0)
Average daily vehicle mileage (km)	Total	17.5		15.9		16.7	
	MV	18.3		16.8		17.6	
	SV	14.9		13.8		14.3	
Average number of vehicles owned per household	Total	1.39		1.52		1.45	
	MV	1.09		1.06		1.08	
	SV	0.30		0.46		0.37	
Average daily trips per household and vehicle	Total	2.36		2.36		2.36	
	MV	1.81		1.62		1.72	
	SV	0.55		0.74		0.64	

Note) The numbers in the parenthesis denotes the share (%).

analysis of households' vehicle holdings. The similar tendency with Fig. 1 and Fig. 2 can be seen from these tables.

3   Joint analysis of households' vehicle ownership & usage

We develop the BSUROPT (Bayesian Seemingly Unrelated Regression, Ordered Probit and Tobit) model, by extending the work of Fang [1]. The BSUROPT model, by taking the form of simultaneous equations, represents households' multiple vehicle ownership and their usage (mileages). So, using this model, it is possible to capture how some factors (e.g. household attributes and socio-economic variables) affects the structural relationship between the number of vehicles and the average mileage.

Table 2: Cross-table analysis of households' vehicle holdings

		SV				Total
		0	1	2	3	
MV	0	-	3151	229	25	3,405
		-	(15.8)	(1.1)	(0.1)	(17.0)
	1	11,383	1,964	233	33	13,613
		(56.9)	(9.8)	(1.2)	(0.2)	(68.1)
	2	1,918	488	91	-	2,497
		(9.6)	(2.4)	(0.5)	-	(12.5)
	3	356	129	-	-	485
		(1.8)	(0.6)	-	-	(2.4)
Total		13,657	5,732	553	58	20,000
		(68.3)	(28.6)	(2.8)	(0.3)	(100.0)

3.1 Model formulation

The latent utilities (y_{1i}^*, y_{2i}^*) of holding each car type and the corresponding latent average mileage (y_{3i}^*, y_{4i}^*) can be written as:

$$y_{ji}^* = \mathbf{x}_{ji}'\beta_j + \epsilon_{ji}, \quad i = 1, ..., N, \quad j = 1, ..., 4,$$

where i denotes household, j means the label of each equation $(y_{1i}^*, y_{2i}^*, y_{3i}^*, y_{4i}^*)$ which can be stacked as a vector $\mathbf{y}_i^*$, $\mathbf{x}_{ji}$ is a vector of independent variables, β_j is a vector of unknown parameters, and ϵ_{ji} is an error term. The vector of error terms follows a multivariate normal distribution with zero means and unrestricted covariance matrix $\boldsymbol{\Sigma}$:

$$\boldsymbol{\epsilon}_j \sim^{\text{i.i.d.}} MVN(\mathbf{0}, \boldsymbol{\Sigma})$$

The structural relationship between the latent utilities and observed variables (car ownership and mileage) may be formulated by Ordered Probit Model and Tobit Model. To do this, we setup the following measurement equations on the holdings of MVs and SVs respectively. The relationship between the observed number of vehicles owned and the latent utility of households

vehicle holding are given by

$$y_{1i} = \begin{cases} 0 & \text{if} \quad y_{1i}^* \leq \alpha_{11} \\ 1 & \text{if} \quad \alpha_{11} < y_{1i}^* \leq \alpha_{12} \\ 2 & \text{if} \quad \alpha_{12} < y_{1i}^* \leq \alpha_{13} \\ 3 & \text{if} \quad \alpha_{13} < y_{1i}^* \end{cases}$$

$$y_{2i} = \begin{cases} 0 & \text{if} \quad y_{2i}^* \leq \alpha_{21} \\ 1 & \text{if} \quad \alpha_{21} < y_{2i}^* \leq \alpha_{22} \\ 2 & \text{if} \quad \alpha_{22} < y_{2i}^* \leq \alpha_{23} \\ 3 & \text{if} \quad \alpha_{23} < y_{2i}^* \end{cases},$$

where α_{11}, α_{12}, α_{13}, α_{21}, α_{22}, α_{23} denotes the thresholds of the latent utilities. These values can be given by $\alpha_{11} = \alpha_{21} = -\Phi^{-1}(1/4)$, $\alpha_{12} = \alpha_{22} = 0$, $\alpha_{13} = \alpha_{23} = \Phi^{-1}(1/4)$ based on Fang [1]. Φ denotes the cumulative distribution function of the normal. The only $\alpha_{12} and \alpha_{22}$ can be estimated in this application for identification.

Finally, the measurement equations about households' vehicle mileages are given by Tobit equations because the vehicle mileages of the households with no vehicles are definitely equal to zero. The Tobit equations are given by

$$y_{3i} = \begin{cases} y_{3i}^* & \text{if} \quad y_{3i}^* \geq 0 \\ 0 & \text{if} \quad y_{3i}^* < 0 \end{cases}$$

$$y_{4i} = \begin{cases} y_{4i}^* & \text{if} \quad y_{4i}^* \geq 0 \\ 0 & \text{if} \quad y_{4i}^* < 0 \end{cases}.$$

3.2 Results of Bayesian Estimation

The BSUROPT model is estimated with Markov Chain Monte Carlo (MCMC) method [2] since the standard maximum likelihood function is analytically intractable. This method iteratively calculates the marginal posterior distribution $\pi(\cdot|\cdot)$ of unknown parameters and it finally converges to the joint posterior distribution. In our application, the following algorithm is applied.

Gibbs Sampling Algorithm

(1) Setup initial values.

(2) Drawing $\boldsymbol{\beta}|\boldsymbol{\Sigma}, \boldsymbol{y}_i^*, \boldsymbol{\alpha}$ from $\pi(\boldsymbol{\beta}|\boldsymbol{\Sigma}^{(t-1)}, \boldsymbol{y}_i^{*(t-1)}, \boldsymbol{\alpha})$.

Drawing $\boldsymbol{\Sigma}|\boldsymbol{\beta}, \boldsymbol{y}_i^*, \boldsymbol{\alpha}$ from $\pi(\boldsymbol{\Sigma}|\boldsymbol{\beta}^{(t)}, \boldsymbol{y}_i^{*(t-1)}, \boldsymbol{\alpha})$

Drawing $\boldsymbol{y}_i^*|\boldsymbol{\beta}, \boldsymbol{\Sigma}, \boldsymbol{y}_i, \boldsymbol{\alpha}$ from $\pi(\boldsymbol{y}_i^*|\boldsymbol{\beta}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \boldsymbol{y}_i, \boldsymbol{\alpha})$

Drawing $\alpha|\boldsymbol{\beta}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \boldsymbol{y}_i^{*(t)}$ from $\pi(\alpha|\boldsymbol{\beta}^{(t)}, \boldsymbol{\Sigma}^{(t)}, \boldsymbol{y}_i^{*(t)})$

(3) Repeat the step (2) until convergence.

We assume that the posterior distributions as follows:

- $\boldsymbol{y}_i^*$: Univariate truncated normal,

- $\boldsymbol{\beta}$: Multivariate normal,

- $\boldsymbol{\Sigma}$: Inverse Wishart, and

- α_{12} α_{22}: uniform.

Using 20,000 samples randomly drawn from the dataset, We take 30,000 MCMC iterations after discarding 10,000 as burn-in in the Gibbs sampling. The diagnosis test indicates that a lot of parameters made convergence.

Table 3 shows the summary result of the posterior distribution of parameters. For the number of MV, the parameter signs of number of household, worker's average income and number of licensed drivers etc. are positive. For the number of SV, gasoline price, number of children and with/without the certification of parking space have positive effects. For the mileage of MV, the parameters of worker's average income and number of stations are negative, so it is different from the result of the number. Thus, it indicates that people have SV, but do not use so much if they have high income or live in the area with a lot of stations. For the mileage of MV, the parameters of worker's average income and with/without the certification of parking space are negative. Because many areas with the certification are located in urban area, people in that area are less likely to access to SV.

3.3   Sensibility analysis

To predict the effects of gasoline prices on car ownership and usage, we calculate the changes in probabilities of vehicle holdings and average mileage when gasoline price increases by 10% , 25% and 50% for each household in the sample. Table 4 summarizes the average change in the probability of holding each vehicle.

As the gasoline price rises, propensity of holding MV reduces whereas the tendency of holding SV increase. This means that with gasoline price rising, people change from holding MV to holding SV. Moreover, for the change in the average mileage, the higher the gasoline price is, the shorter the average mileage of MV is and the longer that of SV is. The overall trend is appropriate, however the accuracy is not enough. Then, I improved the model by including the effects with/without each type of vehicle as a dummy variable. The accuracy to the number of vehicle mileages becomes higher than the previous model. The model for the number of vehicle holdings, however, remains less-accurate.

Table 3: Estimation result of model parameters

Independent variables	Posterior Mean	Standard Dev.
(1) Number of MV		
Number of household	0.0139**	0.00405
log(Population density)	-0.0478**	0.00045
Miles of road	0.0428**	0.00310
Gasoline price(October)	-0.0216**	0.00319
Worker's average income	0.0471**	0.00367
Number of licensed drivers	0.0984**	0.00406
Number of stations	0.0199**	0.00353
Ratio of bus route miles	-0.0066*	0.00290
Threshold	0.0003**	3.0118E-09
(2) Number of SV		
Number of household	-0.0221 *	0.01074
log(Population density)	-0.1485**	0.00119
Certification of parking space	0.0413**	0.00521
Miles of road	-0.0513**	0.00535
Gasoline price(October)	0.0520**	0.00550
Worker's average income	0.0174**	0.00596
Number of children	0.0565**	0.00785
Number of licensed drivers	0.1461**	0.00858
Threshold	0.0005**	2.4968E-09
(3) Average mileage of MV		
Number of household	1.4145**	0.37486
log(Population density)	0.0227	0.04484
Miles of road	2.5412**	0.28064
Gasoline price(October)	-2.0095**	0.29300
Worker's average income	-2.5990**	0.34561
Number of licensed drivers	2.3797**	0.37533
Number of stations	-1.4134**	0.34948
Ratio of bus route miles	-1.7784**	0.27754
(4) Average mileage of SV		
Number of household	-1.7907*	0.87289
log(Population density)	-5.8706**	0.12313
Certification of parking space	-1.4418**	0.43313
Miles of road	-2.8871**	0.43027
Gasoline price(October)	3.3547**	0.44398
Worker's average income	-3.3446**	0.49748
Number of children	5.3776**	0.64995
Number of licensed drivers	9.7975**	0.70288

* (**) mean that 95% (99%) confidence interval does not contain zero.

Table 4: Changes in vehicle choice when gasoline price increases

Change	Probability changes for middle-sized vehicle choice			
in gasoline price	ΔP(M=0)	ΔP(M=1)	ΔP(M=2)	ΔP(M=3)
10%	0.0054	-5.8E-06	-0.0019	-0.0034
25%	0.0135	-0.0001	-0.0049	-0.0085
50%	0.0271	-0.0006	-0.0099	-0.0167
Change	Probability changes for small-sized vehicle choice			
in gasoline price	ΔP(S=0)	ΔP(S=1)	ΔP(S=2)	ΔP(S=3)
10%	-0.0127	0.0043	0.0048	0.0037
25%	-0.0320	0.0105	0.0121	0.0095
50%	-0.0647	0.0198	0.0245	0.0204

4 Conclusion

In this study, we developed an econometric model for analyzing households' multiple-vehicle ownership and their usage simultaneously. The techniques of Bayesian estimation has been applied for parameter estimation. The result of the application to Japanese dataset indicates the effects that the household or the area characteristics have on the vehicle ownership and the car usage. Finally, based on the result of sensibility analysis, it turned out that this model can predict the whole trend, but it is still less-accurate for each household's car ownership behavior.

References

[1] H.A. Fang, "A discrete-continuous model of households' vehicle choice and usage, with an application to the effects of residential density", *Transportation Research Part B* 42, 736-758 (2008).

[2] G. Koop, *Bayesian Econometrics*, Wiley-Interscience, Hoboken (2003).

[3] Cabinet Office, Government of Japan, *A White Paper on Traffic Safety Issues in Japan*, (2007)

An Enhanced Ant Colony System for two Transportation Problems

Luca Maria Gambardella, Roberto Montemanni

Istituto Dalle Molle di Studi sull'Intelligenza Artificiale

Galleria 2, CH-6928 Manno, Switzerland

Email: {luca,roberto}@idsia.ch

1   Introduction

Ant Colony System (ACS, [2]) is a well-known metaheuristic metaphore, and has been succesfully applied to many combinatorial optimization problems. In this work some weaknesses of the method are identified, and some changes to the original paradigm are introduced, in order to enhance the method. Experimental results on two optimization problems arising in transportation are discussed. The results show the effectiveness of the enhancements introduced.

2   An Enhanced Ant Colony System

ACS is a population-based approach initially devoted to solve combinatorial optimization problems. The algorithm takes inspiration from the foraging activity of colonies of ants which are able to compute the shortest path from the food to the nest due to the parallel laying of pheromone on the ground. The basic ACS idea is to use a set of agents (artificial ants) to build feasible solutions for a given optimization problem and to leave these agents to improve iteration after iteration the quality of the best solution. The process is usually driven by a constructive procedure which is based on a graph $G = (V, A)$ with node set V and arc set A given. In ACS each edge of the graph has two associated measures: the heuristic desirability and the pheromone trail (that corresponds to an a-posteriori knowledge of the colony). The heuristic desirability is fixed during the search while the pheromone trail is modified at runtime by ants. Each ant has a random starting node and its goal is to build a feasible solution by adding step by step new edges until a complete solution is generated. The choice of the new edge is made using a probabilistic rule that favors edges with high heuristic desirability and high pheromone trail. Once all ants have built a complete solution, a local search procedure is usually applied. Next, the pheromone trail is updated on the edges of

the best solution. The guiding principle is to increase pheromone trail on edges belonging to high quality solutions, in order to drive the search towards promising regions of the search space.

The constructive phase of the framework described above, can be seen as a diversification process, able to generate solutions that are in the neighborhood of the best solution computed so far. The local search is considered as an intensification phase, able to bring to its local minimum each solution computed by the artificial ants.

One known drawback of the ACS approach is the large total running time required to build new solutions by each artificial ant. Usually the constructive process takes time $O(|V|)$ for each of the $|V|$ steps required. This is acceptable in case of small problems, but it is too expensive in case of larger problems. In fact, ACS algorithms did not show the same performance as in case of small instances when dealing with large routing and scheduling instances. We propose to modify the ACS algorithm in two directions to overcome the slowness drawback: the first enhancement concerns the constructive procedure direclty, while the second impacts on the local search component.

Concerning the constructive procedure, we discuss a new approach which - in contrast with the classic ACS algorithm - directly considers the best solution computed so far already during the constructive phase. In the classic ACS algorithm an ant in node i selects the next edge to visit (outgoing from node i) according to a probabilistic criterion. With probability q_0 the edge selected is that with the best weighted compromise between pheromone trail and heuristic desirability, while with probability $1 - q_0$ the edge is selected according to a Monte Carlo sampling mechanism. In our new proposal, the edge selected with probability q_0 is the edge outgoing from node i in the best solution computed so far (in case this edge is not feasible, the classic mechanism described above is applied). Since probability q_0 is usually grater than 0.9, the new approach drastically reduces the running time required to select the next edge to visit (typically from $O(|V|)$ to $O(1)$).

After the constructive phase is completed, each solution is brought to its local minimum using a local search procedure. The second enhancement we propose comes out here. Our suggestion is to apply the local search procedure only on a (promising) subset of the solutions generated, where the subset usually depends on the problem under investigation, and on the running history of the algorithm. Moreover, the local search is applied only on those solution components which are different from the best solution computed so far (in order to avoid searching the neighbourhood of the same solution over and over again). Notice that the local search enhancements are again in the direction of reducing the total running time.

Despite the two enhancements we propose to heuristically limit the search space considered by the algorithm, the quality of the solution produced by the Enhanced Ant Colony System (EACS) is higher than those of the classic ACS (and often than those of state-of-the-art methods).

3 The Sequential Ordering Problem

The *Sequential Ordering Problem (SOP)*, also referred to as the *Asymmetric Travelling Salesman Problem with Precedence Constraints*, can be modelled in graph theoretical terms as follows. A complete directed graph $D = (V, A)$ is given, where V is the set of nodes and $A = \{(i, j) | i, j \in V\}$ is the set of arcs. A cost $c_{ij} \in \mathcal{N}$ is associated with each arc $(i, j) \in A$. Without loss of generality it can be assumed that a fixed starting node $1 \in V$ is given. It has to precede all the other nodes. The tour is also closed at node 1, after all the other nodes have been visited ($c_{i1} = 0 \ \forall i \in V$ by definition). This artifact creates an analogy with the asymmetric travelling salesman problem. Such an analogy is exploited by many known algorithms. Furthermore an additional precedence digraph $P = (V, R)$ is given, defined on the same node set V as D. An arc $(i, j) \in R$, represents a precedence relationship, i.e. i has to precede j in every feasible tour. Such a relation will be denoted as $i \prec j$ in the remainder of the paper. The precedence digraph P must be acyclic in order for a feasible solution to exist. It is also assumed to be transitively closed, since $i \prec k$ can be inferred from $i \prec j$ and $j \prec k$. Note that for the last arc traversed by a tour (entering node 1), precedence constraints do not apply. A tour that satisfies precedence relationships is called *feasible*. The objective of the *SOP* is to find a feasible tour with the minimal total cost.

Experimenal results. An ACS approach for the SOP is described in [3]. Other approaches are presented in [1, 4, 8]. There are 59 instances for which optimality has not been proven yet. EACS improved 39 of the best solutions obtained by ACS, and matched the remaining 20 best solutions). In general, EACS improved 27 best-known solutions, and matched the remaining 32.

4 The Team Orienteering Problem with Time Windows

The *The Team Orienteering Problem with Time Windows (TOPTW)* can be formally defined as follows. We are given a complete undirected graph $G = \{V, E\}$, with a positive weight t_{ij} associated with each edge, representing the travel time between nodes i and j. For each node $i \in V$ we have the following data: p_i is a positive profit that is collected when the node is visited, $[a_i, b_i]$ is a time window defining the feasible arrival time at the node and s_i is a non-negative service time, that is the amount of time which is spent to visit the node. Two special nodes, numbered 1 and n, where $n = |V|$, are the endpoints of the path to be computed. We have $p_1 = p_n = 0$, $s_1 = s_n = 0$, $a_1 = a_n = 0$ and $b_1 = b_n = T$, where T is equal to the maximum feasible arrival time at node n, that is $T = \max_{i \in V \setminus \{1, n\}} \{b_i + s_i + t_{in}\}$. The TOPTW requires the computation of a set $\mathcal{P}$ of m non-overlapping (apart from origin and destination) elementary paths, such that each path $k \in \mathcal{P}$ is defined as an ordered sequence of nodes starting from node 1 and ending at node n. Given a solution, we indicate with v_i the arrival time at node i.

Experimenal results. An ACS approach for the TOPTW is described in [7]. Other approaches are presented in [5, 6, 9, 10]. There are 224 instances for which optimality has not been proven yet. EACS improved 63 of the best solutions obtained by ACS and matched the best solutions for 152 problems (in 9 cases it was worse). In general, EACS improved 54 best-known solutions, matched 136 best-known solutions, and was worse in 34 cases.

References

[1] D. Anghinolfi, R. Montemanni, M. Paolucci, and L.M. Gambardella. A Particle Swarm Optimization approach for the Sequential Ordering Problem. *Sumbitted for publication*, 2009.

[2] M. Dorigo, G. Di Caro, and L.M. Gambardella. Ant algorithms for discrete optimization. *Artificial Life*, 5:137–172, 1999.

[3] L.M. Gambardella, and M. Dorigo. An ant colony system hybridized with a new local search for the sequential ordering problem. *INFORMS Journal on Computing*, 12(3):237–255, 2000.

[4] F. Guerriero, and M. Mancini. A cooperative parallel rollout algorithm for the sequential ordering problem. *Parallel Computing*, 29(5):663–677, 2003.

[5] M. Kantor and M. Rosenwein. The orienteering problem with time windows. *Journal of the Operational Research Society*, 43(6):629–635, 1992.

[6] R. Mansini, M. Pelizzari, and R. Wolfler Calvo. A granular variable neighborhood search heuristics for the tour orienteering problem with time windows. Technical Report 2008-02-52, Dipartimento di Elettronica per l'Automazione, Universitá di Brescia, 2008.

[7] R. Montemanni, and L.M. Gambardella. An Ant Colony System For Team Orienteering Problems With Time Windows. *Foundations of Computing and Decision Sciences*, to appear.

[8] R. Montemanni, D.H. Smith, and L.M. Gambardella. A Heuristic Manipulation Technique for the Sequential Ordering Problem. *Computers and Operations Research*, 35(12):3931–3944, 2008.

[9] F. Tricoire, M. Romauch, K.F. Doerner, and R.F. Hartl. Heuristics for the multi-period orienteering problem with multiple time windows. *Computers and Operations Research*, to appear.

[10] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe, and D. Van Oudheusden. Iterated local search for the team orienteering problem with time windows. *Computers and Operations Research*, 36(12):3281–3290, 2009.

Lagrangean Decomposition for an Adaptive Location-Distribution Problem

Bernard Gendron

CIRRELT and DIRO, Université de Montréal

Email: Bernard.Gendron@cirrelt.ca

Paul-Virak Khuong

CIRRELT and DIRO, Université de Montréal

Frédéric Semet

LAGIS, Ecole Centrale de Lille

We consider a multi-echelon location problem which arises in the distribution of goods at a national level. The problem is motivated by a case study for a multi-channel retailing company described in Gendron, Semet and Strozyk [4]. The company sells a wide variety of products (clothes, electronic devices, appliances) via Internet, mail order catalogs, and stores. One of its main challenges is to adapt its distribution system according to demand variations while guaranteeing service quality and minimizing the overall logistic cost. Indeed, the demand varies quite significantly over time, while goods have to be delivered within a preset period of time, 24 hours typically. Since most items to deliver are small or medium-size parcels, consolidation is a major concern which is addressed by designing a multi-echelon distribution system. The first echelon is associated with primary facilities such as central warehouses. The second echelon corresponds to facilities such as cross-docking terminals, hereafter simply called terminals. The third echelon is associated to facilities such as small cross-docking terminals called satellites from which tours are issued to serve customers. More precisely, from a small set of warehouses (their locations are assumed known and fixed, following a preliminary strategic analysis), a fleet of large-size trucks delivers parcels to terminals, where they are transferred on medium-size trucks, and then shipped to satellites, where the parcels are sorted and delivered to the customers. The company owns only the central warehouses. Terminals and satellites are neither owned nor rented by the company but by subcontractors such as independent carriers. Satellites can be very basic facilities such as car parks where items are transferred from trucks to vans. This multi-echelon system is *adaptive*, in the sense that terminals and satellites can be opened or closed easily according to demand fluctuations. The problem is to ensure that

customers' requests are satisfied on time at minimum cost, taking into account the transportation costs and the location costs for using the terminals and the satellites.

Gendron and Semet [3] compare two formulations for the problem, arc-based and path-based, showing that the linear programming (LP) relaxation of the path-based model provides a better bound than the LP relaxation of the arc-based model. Although the path-based model is very effective, large-scale instances are still difficult to solve, even for state-of-the-art mixed-integer programming (MIP) solvers. Gendron, Khuong and Semet [2] present a variable neighborhood heuristic algorithm, which provides effective solutions to large-scale instances. In this work, we introduce a Lagrangean decomposition approach embedded in a branch-and-bound scheme, which can deliver provably optimal solutions to large-scale instances of the problem. Before presenting the Lagrangean decomposition, we recall the path-based formulation, which serves as a basis for the approach. Note that, because the locations of the warehouses are assumed to be fixed, we can always assign to each terminal its closest warehouse without losing optimality. The resulting problem can therefore be considered as a *two-echelon* (from terminals to satellites, and from satellites to customers) location-distribution problem.

The following sets define the different types of nodes in the network: D: set of potential sites to locate terminals; S: set of potential sites to locate satellites; L: set of customers; D_j^S: set of potential sites to locate terminals connected to satellite $j \in S$; S_i^D: set of potential sites to locate satellites connected to terminal site $i \in D$; S_l^L: set of potential sites to locate satellites connected to customer $l \in L$; L_j^S: set of customers connected to satellite $j \in S$; L_i^D: set of customers connected to terminal $i \in D$ from some satellite $j \in S_i^D$. The data related to the customer demands and the vehicle capacities are defined as follows (all values are assumed to be positive): n_l: number of product units to deliver to customer $l \in L$; v_l: volume of product units to deliver to customer $l \in L$; Q: capacity (in number of product units) of one batch of products handled at any satellite; P: volumetric capacity of a large-size vehicle transporting product units to any terminal, from its closest warehouse; R: volumetric capacity of a medium-size vehicle transporting product units from any terminal to any satellite. The location and transportation costs are defined as follows (all values are assumed to be nonnegative): f_i: fixed cost for using and operating terminal $i \in D$; g_j: cost per Q product units for using and operating satellite $j \in S$; d_i: transportation cost for using one large-size vehicle to transport product units to terminal $i \in D$ from its closest warehouse; e_{ij}: transportation cost for using one medium-size vehicle from terminal $i \in D$ to satellite $j \in S_i^D$; c_{jl}: transportation cost between satellite $j \in S$ and customer $l \in L_j^S$.

To derive the path-based model, the following sets of binary variables are introduced: $X_{ijl} = 1$, if some product units are transported on path (i, j, l), $i \in D$, $j \in S_j^D$, $l \in L_j^S$; $W_{ij} = 1$, if some product units are transported between terminal $i \in D$ and satellite $j \in S_i^D$; $Y_i = 1$, if some product units are transported to terminal $i \in D$ from its closest warehouse. We also use the

following general integer variables to represent the number of batches handled at any satellite and the number of vehicles used on any terminal-satellite arc or at any terminal: U_j: number of batches of products handled at satellite $j \in S$; T_i: number of large-size vehicles used between terminal $i \in D$ and its closest warehouse; H_{ij}: number of medium-size vehicles used between terminal $i \in D$ and satellite $j \in S_i^D$.

The path-based formulation of the problem can then be written as follows (we omit the constraints that specify the nature of the different types of variables):

$$\min \sum_{i \in D} f_i Y_i + \sum_{j \in S} g_j U_j + \sum_{i \in D} d_i T_i + \sum_{i \in D} \sum_{j \in S_i^D} e_{ij} H_{ij} + \sum_{i \in D} \sum_{j \in S_i^D} \sum_{l \in L_j^S} c_{jl} X_{ijl} \tag{1}$$

$$\sum_{j \in S_l^L} \sum_{i \in D_j^S} X_{ijl} = 1, \quad \forall l \in L, \tag{2}$$

$$\sum_{i \in D_j^S} W_{ij} \leq 1, \quad \forall j \in S, \tag{3}$$

$$\sum_{j \in S_i^D} \sum_{l \in L_j^S} v_l X_{ijl} \leq (\sum_{l \in L_i^D} v_l) Y_i, \quad \forall i \in D, \tag{4}$$

$$X_{ijl} \leq W_{ij}, \quad \forall i \in D, \forall j \in S_i^D, \forall l \in L_j^S, \tag{5}$$

$$W_{ij} \leq Y_i, \quad \forall i \in D, \forall j \in S_i^D, \tag{6}$$

$$\sum_{i \in D_j^S} \sum_{l \in L_j^S} n_l X_{ijl} \leq Q U_j, \quad \forall j \in S, \tag{7}$$

$$\sum_{j \in S_i^D} \sum_{l \in L_j^S} v_l X_{ijl} \leq P T_i, \quad \forall i \in D, \tag{8}$$

$$\sum_{l \in L_j^S} v_l X_{ijl} \leq R H_{ij}, \quad \forall i \in D, \forall j \in S_i^D. \tag{9}$$

The objective function, (1), consists in minimizing all costs incurred by using and operating terminals and satellites, as well as transportation costs between warehouses and terminals, between terminals and satellites, and between satellites and customers. Constraints (2) ensure that each customer is being served by a single satellite. Constraints (3) ensure that any satellite, when it is used, is connected to a single terminal. The forcing constraints (4) ensure that no flow can circulate through a terminal that is not used to transport product units. Constraints (5) and (6) are also forcing constraints that link together the different types of binary variables. Constraints (5) ensure that any customer cannot be routed from a satellite that is not connected to some terminal. Similarly, constraints (6) ensure that any satellite cannot be connected to a terminal that is not used to transport product units. Constraints (7) ensure that the number of product units handled at a satellite cannot exceed the capacity of product batches. Constraints (8) and (9) ensure that the total volume of all product units transported on a network element (terminal or terminal-satellite arc) cannot exceed the capacity of the vehicles used on that network element.

To derive the Lagrangean decomposition approach, we first introduce three sets of variables with corresponding *copy constraints* defined as follows: $N_{jl} = \sum_{i \in D_j^S} X_{ijl}, j \in S, l \in L_j^S$; $V_{il} = \sum_{j \in S_i^D} X_{ijl}, i \in D, l \in L_i^D$; $M_{ijl} = X_{ijl}, i \in D, s \in S_i^D, l \in L_j^S$. Constraints (7), (8) and (9) are then rewritten using the new variables: $\sum_{l \in L_j^S} n_l N_{jl} \leq QU_j, j \in S$; $\sum_{l \in L_i^D} v_l V_{il} \leq PT_i, i \in D$; $\sum_{l \in L_j^S} v_l M_{ijl} \leq RH_{ij}, i \in D, \forall j \in S_i^D$. Finally, the copy constraints are relaxed in a Lagrangean way. The resulting Lagrangean subproblem decomposes into two parts: the first one, called LOC, is a two-echelon uncapacitated location-distribution problem, whose structure is defined by constraints (2) to (6); the second part, called $KNAP$, decomposes itself into $|S| + |D| + \sum_{i \in D} |S_i^D|$ 0-1 knapsack problems with variable capacity, each of these problems corresponding to a structure known as the *unsplittable flow arc set* in the network design literature [1]. Since both LOC and $KNAP$ do not have the integrality property, the lower bound obtained by solving the Lagrangean dual (we use a bundle method for this purpose) dominates the LP relaxation bound, but also the Lagrangean bounds computed by relaxing constraints (2) to (6) (obtaining $KNAP$ as the Lagrangean subproblem), and by relaxing constraints (7) to (9) (obtaining LOC as the Lagrangean subproblem). Upper bounds on the optimal value of the problem are derived directly from the solutions to each LOC subproblem; these solutions are then improved using variable neighborhood heuristic methods [2]. Finally, if there is still a gap between the best lower and upper bounds, a branch-and-bound procedure is called upon, where bounding at each node is performed with the Lagrangean decomposition approach, while branching is based on the violations of the copy constraints.

References

[1] A. Atamtürk and D. Rajan, "On Splittable and Unsplittable Flow Capacitated Network Design Arc-Set Polyhedra", *Mathematical Programming* 92, 315-333 (2002).

[2] B. Gendron, P.-V. Khuong and F. Semet, "Variable Neighborhood Heuristic Methods for a Multi-Echelon Capacitated Location-Distribution Problem", Odysseus 2009, Cesme, May 2009.

[3] B. Gendron and F. Semet, "Formulations and Relaxations for a Multi-Echelon Capacitated Location-Distribution Problem", *Computers and Operations Research* 36, 1335-1355 (2009).

[4] B. Gendron, F. Semet and C. Strozyk, "Adaptive Distribution Systems", in *TRANSTECH: Transport Technology Product & Process Innovation Management*, 36-42, Presses Universitaires de Valenciennes (2002).

Locating Sensors on Traffic Networks: A Survey

M. Gentili

Department of Mathematics and Computer Science

University of Salerno

Email: mgentili@unisa.it

Pitu B. Mirchandani

School of Computing, Informatics and Decision Systems Engineering

Arizona State University

Email: pitu@asu.edu

1.   Introduction

Sensors are used on traffic networks to collect data for the purpose of monitoring and management of the traffic flows. Through the monitoring, travelers may be informed of the network conditions via traveler information systems. Subsequently, better information on the traffic flows, including identification of congestion and bottlenecks, allow users and managers to better control the flows on the network. Important decisions towards this end relate to where the given sensors should be located to in order to better monitor and manage flows.

Many different location models have been proposed in the literature, as well as corresponding solution approaches. The proposed existing models could be classified according to two main criteria: (i) the types of sensors based on what they measure (e.g., counting sensors, image sensors, Automatic Vehicle Identification (AVI) readers, etc.), and, (ii) the objective for the optimal locations, such as, best estimation of origins and destinations, estimation of flows on arcs and routes, maximal interception of flows, and reliably measuring travel times. The purpose of this paper is review such models, including introducing new models, by classifying then with respect to characterizing parameters and objective functions.
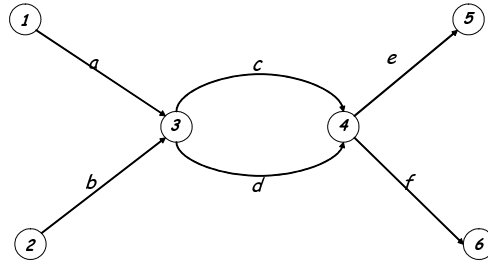
Figure 1: *A network example*

2. Locating Sensors to Measure Flows

Network traffic flows may be characterized in various ways, for example, flow volumes (in number of vehicles per hour) from each origin to each destination, and, for another example, flow volumes on given routes or route segments. Depending on what we are interested in we can deploy sensors to *observe* such flows. The first issue that may be raised is "Are these flows observable?" For example, if one were interested in measuring passenger flows on a route segment, current deployable sensor technologies cannot observe that (but they may be estimated if one assumed a passenger per vehicle distribution). Note, however, we are able to *directly observe* arc flows or, in some cases route flows. Sometimes we are able to *indirectly observe* more aggregated flows such as OD trips. Generally, relationships among various flow volumes can be represented by a system of linear equations where the columns represent the volume of flows and rows amounts come from data from the deployed sensors. In this context, the main interest is either (i) where to deploy sensors so that the related matrix has full rank and thus a unique solution is obtained or (ii) how to choose the best possible solution of the system of equations among all the possible solutions when a deployment for a unique solution is not possible. These two possibilities lead two main classes of problems: the *Sensor Location Flow-Observability Problem* and the *Sensor Location Flow-Estimation Problem*, which are introduced below with an example.

Consider the network in Figure 1 where there are 6 nodes and 6 arcs. There are 4 OD pairs $w_1 = (1, 5), w_2 = (1, 6), w_3 = (2, 5), w_4 = (2, 6)$, and each pair is connected by two different routes, one using arc c and the other using arc d. Table 1 contains all the information regarding OD flows, route flows, arc flows. That is, for example, the flow between the OD pair $(1, 5)$ is 120 units, 48 of them use route $R_1 = \{a, d, e\}$ and the remaining 72 the route $R_2 = \{a, c, e\}$. Arc flow of arc a is 200 units resulting from the sum of the flows of the routes in the network that use arc a, that is R_1, R_2, R_3 and R_4 whose flow respectively is $f_{R_1} = 48, f_{R_2} = 72, f_{R_3} = 40, f_{R_4} = 40$.

Let us assume, for the purpose of this example, we are interested in knowing route flows and consider the two cases where we locate on arcs either (i) counting sensors or (ii) path-ID sensors

OD couple	OD Trips	Routes				Route Flow
(1,5)	120	R_1:	a	d	e	48
		R_2:	a	c	e	72
(1,6)	80	R_3:	a	d	f	40
		R_4:	a	c	f	40
(2,5)	60	R_5:	b	d	e	12
		R_6:	b	c	e	48
(2,6)	100	R_7:	b	d	f	30
		R_8:	b	c	f	70

Arcs	a	b	c	d	e	f
Arc Flow	200	160	206	154	180	180

Table 1: *OD flows, Route flows, Arc flows related to the network in Figure 1*

which measure volumes of each route on the arc. These hypothesis are considered for clarity of the exposition; they can be easily generalized to consider any type of sensors and various flow characterizations.

Case (i): locating counting sensors

Suppose we can locate 3 counting sensors on the arcs of the network. When we locate a counting sensor on an arc of the network we can count the total flow on that arc and express it as the sum of flows of the routes that use the arc. For example, if we locate a counting sensor on arc a we can derive the following linear equation: $f_{R_1} + f_{R_2} + f_{R_3} + f_{R_4} = 200$. If we locate counting sensors on arcs a, c and e we can define the following system of 3 linear equations with 8 unknown variables:

$$
\begin{array}{c}
\begin{array}{cccccccc}
f_{R_1} & f_{R_2} & f_{R_3} & f_{R_4} & f_{R_5} & f_{R_6} & f_{R_7} & f_{R_8}
\end{array} \\
\begin{array}{c}
(arc\ a) \\
(arc\ c) \\
(arc\ e)
\end{array}
\begin{pmatrix}
1 & 1 & 1 & 1 & 0 & 1 & 0 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 0 & 0 & 1 & 1 & 0 & 0
\end{pmatrix}
\end{array}
\begin{pmatrix}
f_{R_1} \\ f_{R_2} \\ f_{R_3} \\ f_{R_4} \\ f_{R_5} \\ f_{R_6} \\ f_{R_7} \\ f_{R_8}
\end{pmatrix}
=
\begin{pmatrix}
200 \\ 160 \\ 206
\end{pmatrix}
\tag{1}
$$

This system does not have a unique solution. The same happens if we choose to locate three sensors on arcs c, d, e. Hence, our interest is focused in choosing the best solution among all possible solutions.

To evaluate when a solution is better that another, we can define an estimation function. Generally such a function is derived from some estimation methods (such as least square estimation and maximum entropy estimation among others). Therefore in this context, the location problem is: *How to locate sensors on the network so that the estimation function is optimized?*

Case (ii): locating path-ID sensors

Suppose we can locate three path-ID sensors on the arcs of the network. When we locate a path-ID sensor on an arc we are able to know the flow volumes on each route that uses the arc. For example, by locating a path-ID sensor on arc a we can observe: $f_{R_1} = 48$, $f_{R_2} = 72$, $f_{R_3} = 40$, $f_{R_4} = 40$; while by locating a path-ID sensor on arc d we can observe: $f_{R_1} = 48$, $f_{R_3} = 40$, $f_{R_5} = 12$, $f_{R_7} = 30$. Therefore, by locating three path-ID sensors on arcs a, c and d we can *observe* the flows of all the routes of the network. Of course, not all the three sensors are needed to observe all route flows. Indeed, by locating only on c and d (or, also, on a and b) a unique solution would be achieved as well. Therefore, in this context, the location problem addresses the observability issue: *what is the best set of location of sensors on the network so that a unique solution of the corresponding system of linear equation is obtained*, where *best* could mean *minimum number*, *minimum cost* or some other optimizing criterion.

Both the problems are now formally defined in the next section.

3.   Observability versus Estimation

We can represent a traffic network by means of a graph $G = (V, A)$ where the set of nodes V represents intersections in the network and the set of arcs A, joining node pairs, represents roads. Flows on the arcs of a network are generated from the users that travel along the network from a given set of origins to a certain set of destinations following different routes. Let us denote by h^w the average number of trips connecting the OD pair w within a given time period. The main relationships among arc flows, route flows and OD flows are the following:

$$\sum_{w \in W} p_a^w h^w = v_a \quad \forall a \in A \tag{2}$$

$$h^w = \sum_{r \in R^w} f_r \quad \forall w \in W \tag{3}$$

$$f_r = h^w p_r^w \quad \forall r \in R \, \forall w \in W \tag{4}$$

$$v_a = \sum_{r \in R} f_r \rho_{ar} \quad \forall a \in A \tag{5}$$

$$v_a = \sum_{w \in W} \sum_{r \in R^w} h^w p_r^w \rho_{ar} \quad \forall a \in A \tag{6}$$

where W is the set of all OD pairs, p_a^w are the arc choice proportions defining the portion of trips between pair w that use arc a, v_a is the flow on arc a, R is the set of all the routes in the network connecting OD pairs, $R^w \subseteq R$ is the set of routes connecting OD pair w, f_r is the flow of route $r \in R$, p_r^w are route choice proportions denoting the proportion of flow of the OD pair w traveling

on route r, and parameter ρ_{ar} is equal to 1 if route r contains arc a and 0 otherwise. Any of the above systems of linear equations would give us information about unknown flows (either OD flows, arc flows, or route flows).

By locating different types of sensors either on the arcs or on the nodes of the network one could obtain additional information on how flows on arcs can be disaggregated in route flows. Therefore, according to (1) the different type of sensors that can be located, (2) assumptions made about the available a-priori information, and, (3) the type of estimates one focuses on, different system of equations should be derived. In this context, the total number of observations and their location play an important role for the determination of the system under consideration. In particular, two questions that arise are:

- If the system is observable, what is the minimum number of sensors and where to locate them to obtain the unique solution of the system?

- If the system is not observable, how to choose sensor locations to improve the quality of the flows estimates?

Existing contributions answer these two questions, and can be then grouped into two main classes of problems:

i) *The Sensor Location Flow-Observability Problem*: identify the optimum location of sensors on the network that allow the unique determination of the solution of the related linear system of equations.

ii) *The Sensor Location Flow-Estimation Problem*: identify the optimum location of sensors on the network to best improve the quality of the related estimates (OD trips estimates, arc flows estimates and, route flow estimates).

Most of the contributions existing in the literature can be then grouped in these two classes of problems. Mainly, the contributions dealing with the estimation of OD flows by locating counting sensors on the arcs of the network deal with an unobservable system of equations.

From the early nineties, most of the studies have concentrated on the Sensor Location Flow-Estimation Problem and therefore the great majority of existing contributions address this problem. More recently, mainly due to new types of sensors using new technologies, the Sensor Location Flow-Observability Problem has gained more attention and many new interesting and relevant problems have been investigated. In this paper, we will review the main contributions on these two classes of problems.

Same-Day Courier Shift Scheduling with Multiple Classes of Requests

Gianpaolo Ghiani

Dipartimento di Ingegneria dell'Innovazione

University of Salento, via per Monteroni, 73100, Lecce, Italy

Emanuele Manni

Dipartimento di Ingegneria dell'Innovazione

University of Salento, via per Monteroni, 73100, Lecce, Italy

Email: emanuele.manni@unisalento.it

Antonella Quaranta

Dipartimento di Ingegneria dell'Innovazione

University of Salento, via per Monteroni, 73100, Lecce, Italy

1 Introduction

Same-day clients usually request couriers with little or no notice, all but eliminating the ability to construct routes or schedules in advance. Once a courier has been assigned a job, he/she proceeds directly to the pickup location, collects the appropriate conveyance, and moves on to the delivery. Automated information-based job allocation systems ([1]) are based on dispatching algorithms able to assign each job to the most appropriate courier on the basis of the current fleet location and status. Advantages of such systems include an improvement of the courier efficiency, a reduction of the requirements of human supervisors as well as the possibility to provide customers a quality of service (QoS) guarantee. In this extended abstract we deal with the same-day Courier Shift Scheduling Problem (CSSP), a tactical problem which amounts to minimize the staffing cost subject to probabilistic service level requirements. In particular, we investigate the value of clustering customer requests into classes, extending the work presented in [3], in which the authors consider a single class of requests. In what follows we assume that couriers are independent contractors paid by the hour, thus creating economic incentives for companies to hire the least amount of labour possible. The remainder of this extended abstract is organized as follows. In Section 2 we model the CSSP as an integer program with nonlinear probabilistic QoS constraints, whereas in Section

3 we describe the main ideas of our solution approach. Finally, in Section 4 we present preliminary computational results.

2 Formulation

At an operational level, same-day couriers must solve a *Dynamic Vehicle Dispatching Problem with Pickups and Deliveries* ([2], [4]) which aims at allocating requests to vehicles as well as scheduling the requests assigned to each vehicle. A pickup and its associated delivery must be serviced by the same vehicle and a pickup must always be made before its associated delivery. Given any dispatching policy P, at a tactical level courier companies face a CSSP, i.e., they must decide how many couriers should be allocated to each shift pattern subject to QoS constraints. The CSSP is usually solved on a weekly or quarterly basis (the demand is usually characterized by significant yearly/weekly/daily seasonal effects) with the aim of minimizing the staffing cost. In what follows, we assume that requests are partitioned into E different classes, which may differ for either the origin-destination pair or the characteristics of the parcel to transport. Moreover, we assume the availability of V different courier types (e.g., bike, motorbike or van couriers). Furthermore, we suppose that a QoS guarantee is provided to those requests arriving into a planning horizon H (e.g., Monday to Friday, 6.00 am to 10.00 pm) and that the QoS is assessed with respect to J time intervals (*QoS intervals*) in which H is partitioned. In particular, we require that the expected service time of a request of class e ($e = 1, \ldots, E$) arriving during time interval j ($j = 1, \ldots, J$) must be less than a given threshold T_j^e. In this extended abstract we assume that the feasible shifts (i.e., shift satisfying rest regulations) can be enumerated. This assumption is realistic since only few patterns are acceptable in real world (e.g., shifts covering 4 consecutive days a week for 10 consecutive hours per day *or* 5 consecutive days a week for 8 consecutive hours per day). Let Q be the number of feasible shift patterns, and let c_{vq} be the wage of a courier of type v covering pattern q ($q = 1, \ldots, Q$; $v = 1, \ldots, V$). The CSSP amounts to determine the optimal number x_{vq} of couriers of type v covering shift pattern q ($q = 1, \ldots, Q$; $v = 1, \ldots, V$):

$$\text{Minimize} \quad z(x) = \sum_{v=1}^{V} \sum_{q=1}^{Q} c_{vq} x_{vq} \tag{1}$$

$$\text{s.t.} \quad g_j^e(x_{11}, \ldots, x_{VQ}) \leq T_j^e \quad j = 1, \ldots, J; \quad e = 1, \ldots, E \tag{2}$$

$$x_{vq} \geq 0, \quad \text{integer} \quad v = 1, \ldots, V; \quad q = 1, \ldots, Q, \tag{3}$$

where $g_j^e(x_{11}, \ldots, x_{VQ}) = E[G_j^e(x_{11}, \ldots, x_{VQ}, \xi)]$. Here, ξ is a vector denoting the random demands across the planning horizon, $G_j^e(x_{11}, \ldots, x_{VQ}, \xi)$ is the *service* (or *system*) *time* of a request of class e ($e = 1, \ldots, E$) arising during QoS interval j ($j = 1, \ldots, J$) under dispatching policy P, and $g_j^e(x_{11}, \ldots, x_{VQ})$ is its expected value. The complexity of the model lies in the $g_j^e(\cdot)$ functions, which are non linear and not known explicitly.

3 Solution strategies

We propose a procedure that collects some statistics when simulating the current solution $x^{(k)}$ at iteration $(k-1)$. Then, we use these statistics into an *Approximated Neighborhood Evaluation* (ANE) procedure that approximates the QoS functions $g_j^e(\cdot)$ with deterministic linear functions of the x_{vq} variables. We divide the planning horizon into m micro-intervals of duration Δt (for instance, $\Delta t = 30$ minutes or 1 hour) and assume that the arrival rate λ_h^e of requests of class e $(e = 1, \ldots, E)$ during micro-interval I_h $(h = 1, \ldots, m)$ is constant. Let a and d be the arrival time and the delivery time of a request, and let $s = d - a$ be its *service time*. When selecting $x^{(k)}$ as the new current solution at iteration $(k-1)$, we compute an estimate $\hat{s}_{hl}^{ev}$ of the conditional expected service time (under operational policy P) of the requests of class e arriving in micro-interval I_h and serviced in I_l $(l \in W_h$, where W_h is the set of micro-intervals after I_h within the same working day) by a vehicle of type $v \in V^e$, where $V^e \subseteq V$ represents the subset of all the vehicle types which can service requests of class e: $s_{hl}^{ev} = E[s | a \in I_h, d \in I_l, e, v, x^{(k)}, P]$. It is worth noting that such an estimate comes at no cost since the QoS provided by $x^{(k)}$ must be evaluated (through simulations runs) before this solution can be declared feasible.

Let H_j be the set of micro-intervals which make up QoS interval j $(j = 1, \ldots, J)$ and let A_l be the set of shifts covering micro-interval I_l $(l = 1, \ldots, m)$. By using the well known total probability theorem, QoS constraints can be reformulated as follows:

$$\sum_{h \in H_j} \left(\frac{1}{\sum_{i \in H_j} \lambda_i^e} \sum_{l \in W_h} \sum_{v \in V^e} \lambda_{hl}^{ev} \hat{s}_{hl}^{ev} \right) \le T_j^e, \quad j = 1, \ldots, J; \quad e = 1, \ldots, E, \tag{4}$$

where λ_{hl}^{ev} is the part of λ_h^e delivered during micro-interval I_l $(l \in W_h)$ by a courier of type v $(v \in V^e)$. Constraints (4) are then used as part of a linear program (not reported here, for the sake of brevity) to determine, at each iteration k, the best neighbor of current solution $x^{(k)}$.

4 Preliminary computational results

The purpose of our computational experiments is to determine whether it is valuable to cluster the requests into classes in place of using a single class. For this purpose, we embed the ANE mechanism into a multi-start heuristic, and compare with the results obtained in [3]. Experiments are performed on randomly generated instances, resembling usual courier operations. In particular, the service territory is a grid with 36 zones, giving rise to 1260 origin-destination pairs, the travel times between adjacent zones are set equal to 15 minutes, and the planning horizon is from Monday to Friday, 8.00 am to 8.00 pm. The requests are clustered, according to the origin-destination pair, into $E = 6$ classes, with 600 overall expected weekly requests. The thresholds T_j^e are set equal to 60 minutes for $e = 1, \ldots, E$, and $j = 1, \ldots, J$. At an operational level we use a cheapest insertion policy, giving an higher priority to requests having their origin-destination pairs in the

Table 1: Preliminary computational results

| Instance | ANE-FP | ANE-MC | |
	OBJ	OBJ	DEV
1	856	821	-4,09%
2	914	889	-2,74%
3	864	812	-6,02%
4	812	801	-1,35%
5	889	850	-4,39%
6	914	861	-5,80%
7	872	850	-2,52%
8	951	901	-5,26%
9	807	776	-3,84%
10	930	898	-3,44%
Average			-3,94%

central zones of the service territory (which, in a real-world setting, would be the downtown part of a city). Preliminary experiments (Table 1) report the performance of the ANE-based heuristic with multiple classes of requests (ANE-MC), and without this demand characterization (ANE-FP), and compare the results obtained by the two approaches under the same conditions: ANE-MC: $\sum_{e=1}^{E} \sum_{h=1}^{m} \lambda_h^e = 600$; ANE-FP: $\lambda_h = \sum_{e=1}^{E} \lambda_h^e$; $\hat{s}_h(x) = \sum_{e=1}^{E} \frac{\lambda_h^e}{\sum_{e'=1}^{E} \lambda_h^{e'}} \hat{s}_h^e(x)$. The results show that ANE-MC provides an average cost reduction of approximately 4%, which could result in consistent monetary savings.

References

[1] A. Attanasio, J. Bregman, G. Ghiani and E. Manni, "Real-time fleet management at eCourier Ltd", in *Dynamic Fleet Management - Concepts, Systems, Algorithms & Case Studies*, V.S. Zeimpekis, G.M. Giaglis, C.D. Tarantilis and I.E. Minis (eds), 219-238, Springer-Verlag, 2007.

[2] M. Gendreau, F. Guerten, J.-Y. Potvin and R. Séguin, "Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries", *Transportation Research Part C* 14, 157-174 (2006).

[3] G. Ghiani, E. Manni and A. Quaranta, "Shift scheduling problem in same-day courier industry", *Transportation Science*, Published online December 9, 2009 DOI: 10.1287/trsc.1090.0293.

[4] G. Ghiani, E. Manni, A. Quaranta and C. Triki, "Anticipatory algorithms for same-day courier dispatching", *Transportation Research Part E* 45, 96-106 (2009).

Hop-indexed Circuit-based formulations for the Travelling Salesman Problem

Maria Teresa Godinho

Centro IO & Escola Superior de Tecnologia e Gestão

Polytechnic Institute of Beja

Email: mtgodinho@ipbeja.pt

Luís Gouveia

Centro IO & Faculdade de Ciências

University of Lisbon

Email: legouveia@fc.ul.pt

Pierre Pesneau

Institut de Mathématique de Bordeaux & Université de Bordeaux 1,
Email: Pierre.Pesneau@math.u-bordeaux1.fr

Consider a graph G=(V;A), where $V = \{1,2,..,n\}$ and $A=\{(i,j): i,j=1,..,n, i \neq j\}$. Assume that we have a cost c_{ij} associated to each arc in A. Consider the following *generic formulation* for the ATSP:

$$minimize \quad \sum_{(i,j)\in A} c_{ij} x_{ij}$$
$$subject\ to \quad x = (x_{ij}) \in Assign \qquad (1)$$
$$\{(i,j): x_{ij} = 1\}\ is\ connected. \qquad (2)$$

with Assign denoting the feasible set of the well-known assignment relaxation arising in formulations for the problem:

$$\sum_{i\in V} x_{ij} = 1 \quad for\ all\ j \in V \qquad\qquad (1a)$$
$$\sum_{j\in V} x_{ij} = 1 \quad for\ all\ i \in V \qquad\qquad (1b)$$
$$x_{ij} \in \{0,1\} \quad for\ all\ (i,j) \in A. \qquad\qquad (1c)$$

Several research works on formulations for the TSP and the ATSP start with this generic formulation and then exploit different ways of expressing (2), see, for instance, [3], [4], [5], [6] and [7] which cover these matters, reviewing and comparing most of the published formulations for

both the TSP and the ATSP. We obtain probably the most well-known formulation for the ATSP by adding the standard cut constraints

$$\sum_{[i \in V \setminus S,\, j \in S]} x_{ij} \geq 1 \qquad \textit{for all } \; S \subseteq V \setminus \{1\} \tag{Cut2}$$

that guarantee that the solution is connected. A formulation using only the x_{ij} variables is called a *natural* formulation in contrast with a so-called extended formulation that uses extra variables to express (2). We start our study, with an extended formulation such that the non explicit part is given as follows

$$\left\{ (i, j) : \; g_{ij}^{k} = 1 \right\} \textit{ contains a non necessarily simple circuit containing node } 1, \textit{ node } k,$$
$$\textit{with exactly } n \textit{ arcs} \qquad\qquad \textit{for all } k \in V \setminus \{1\} \tag{C2a}$$
$$g_{ij}^{k} = x_{ij} \qquad\qquad \textit{for all } (i, j) \in A, \; k \in V \setminus \{1\}. \tag{C2b}$$

In these models, the circuit variables g_{ij}^{k} indicate whether arc (i,j) is in the circuit passing through node k.

One way of writing this "circuit" model is to find an exact model for the underlying constrained circuit subproblem, that is, a model whose linear programming relaxation has integral extreme points. Such a model can be obtained by modelling the problem as an unconstrained shortest path in an appropriate graph, as it has been done by Godinho, Gouveia and Magnanti in [2]. In this paper, we propose a formulation that is more compact than the one presented in [2] and with the same linear programming relaxation bound. The formulation is based on the following algorithm for computing, for each k, the (not necessarily simple) circuit passing through node 1, containing n arcs and including node k: for each integer p (p = 1,...,n) we compute the shortest path from node 1 to node k with exactly p arcs and the shortest path from node k to node 1 with exactly p arcs. Then we combine adequately shortest paths from the first series with the shortest paths from the second (for instance the shortest path from node 1 to node k with exactly p arcs and the shortest path from node k to node 1 with exactly n+1-p arcs gives the shortest circuit such that node k is in position p).

A straightforward shortest path reformulation based on this two-layered graph provides a compact hop-indexed model (the construction just given shows that the corresponding linear programming relaxation is integer) for the underlying circuit subproblem. We associate $z1_{ij}^{hk}$ variables to the arcs of the sub-graph modeling paths in the first part of the circuit and variables $z2_{ij}^{hk}$ to the arcs of the sub-graph modeling paths in the second part of the circuit, that is: we consider variables i) $z1_{ij}^{hk} = 1$ if arc $(i, j) \in A(j \neq 1, i \neq k)$ is in the hth position in the circuit from node 1 to node 1 passing through node k and is before node k (that is in the path from node 1 to node k) and variables ii) $z2_{ij}^{hk} = 1$ if arc $(i, j) \in A(j \neq k, i \neq 1)$ is in the hth position in the circuit, from node 1 to

node 1 passing through node k <u>and is after node k</u> (that is in the path from node k to node 1). Using these variables, we can write the following new model for the Circuit subproblem:

Table I. Modelling a Circuit (from node 1 to node 1 and passing through node k)

$\sum_{j \in V} z1_{1j}^{1k} = 1$		(H-C$_1$)
$\sum_{j \in V} z1_{ij}^{h+1,k} - \sum_{j \in V} z1_{ji}^{hk} = 0$	*for all* $i \in V \setminus \{1\}, h = 1,..,\|V\|-2$	(H-C$_2$)
$\sum_{j \in V} z2_{kj}^{h+1,k} - \sum_{j \in V} z1_{jk}^{hk} = 0$	$h = 1,...,\|V\|-1$	(H-C$_3$)
$\sum_{j \in V} z2_{ij}^{h+1,k} - \sum_{j \in V} z2_{ji}^{hk} = 0$	*for all* $i \in V \setminus \{1\}, h = 2,..,\|V\|-1$	(H-C$_4$)
$g_{ij}^k = \sum_{h=1}^{n} \left(z1_{ij}^{hk} + z2_{ij}^{hk} \right)$	*for all* $(i,j) \in A$	(H-C$_5$)
$z1_{ij}^{hk} \in \{0,1\}$	*for all* $(i,j) \in A, i \neq k, j \neq 1,$ $h = 1,..,\|V\|-1$	(H-C$_6$)
$z2_{ij}^{hk} \in \{0,1\}$	*for all* $(i,j) \in A, i \neq 1, j \neq k,$ $h = 2,..,\|V\|$	(H-C$_7$)

Clearly the subformulation we have used is the tightest we can get for the defined circuit subproblem since the corresponding linear programming feasible set has integer vertices. We can obtain a formulation for the ATSP by replacing (C2a) with this circuit formulation for each k. We let HC-MCF denote this model.

As noted before, this is the starting formulation for our work. We will discuss:

i)   Model enhancements - In the previous model the focus was set on finding good formulations for the sub-problems associated to each node k, each one seen as an independent sub-problem - the sub problems are related to each other only through the linking constraints. However, when we analyse a solution for the ATSP we realize that there is a great deal of information regarding the way the sub-problems are related to each other and that might be used to derive improved formulations. In this section, we analyse formulations for the ATSP problem from this point of view. We will use simple and straightforward proprieties that relate the circuits associated to different nodes and that will permit us to write "interesting" new valid inequalities for the whole problem. Since all of these properties characterize common features of all circuits we will designate the new inequalities by Intersecting-Circuit (IC) inequalities.

ii) We will contextualize the linear programming relaxation of the new enhanced model with the linear programming relaxation of some of the strongest formulations know from the literature. In particular, we will show that among known compact formulations from the literature, the proposed formulation is the one with the tightest linear programming bound.

iii) From a computational point of view, we will show that the proposed formulation is quite interesting for the related and so-called cumulative travelling salesman problem [1], Computational results taken from instances with up to 40 nodes show that the proposed formulation provides linear programming gaps that are within 1% per cent of the optimum. This is a huge improvement to previously know formulations.

References

[1] M.. Fischetti, G. Laporte, and S. Martello, "The Delivery Man Problem and Cumulative Matroids", *Operations Research*, 41, 1055-1064, (1993).

[2] M.T. Godinho, L. Gouveia, and T.L. Magnanti, "Combined Route Capacity and Route Length Models for Unit Demand Vehicle Routing Problems", *Discrete Optimization*, 5, (2), 350-372, (2008).

[3] L. Gouveia, L. and J. M. Pires, "The asymmetric travelling salesman problem and a reformulation of the Miller-Tucker-Zemlin constraints", *European Journal of Operations Research*, 112, (1), 134-146, (1999).

[4] L. Gouveia, and S. Voss, "A Classification of Formulations for the (Time-Dependent) Traveling Salesman Problem", *European Journal of Operations Research*, 83, (1), 69-82, (1995).

[5] A. Langevin, F. Soumis and J. Desrosiers , "Classification of travelling salesman formulations", *Operations Research Letters*, 9, 127-132, (1990).

[6] A.J. Orman and H.P. Williams, ''A survey of different integer programming formulations of the travelling salesman problem'', in: *Optimization, economics and financial analysis. Advances in computational management science,* E.J. Kontoghiorghes and C. Gatau, (eds), 93-106, Sprienger; (2007).

[7] T. Ocan, I. Atinel and G. Laporte, ''A comparative analysis of several asymmetric travelling salesman formulations'', *Computers and Operations Research*, 36, (3), 637-654, (2009).

A quadratic time algorithm for the
U.S. Truck Driver Scheduling Problem

Asvin Goel

MIT-Zaragoza International Logistics Program

Zaragoza Logistics Center, Spain

and

Applied Telematics/e-Business Group,

Department of Computer Science, University of Leipzig, Germany

Email: asvin.goel@uni-leipzig.de

Leendert Kok

Operational Methods for Production and Logistics, University of Twente

P.O. Box 217, 7500AE, Enschede, Netherlands

1   Introduction

In January 2009 present driving and working hour regulations in the United States entered into force which are comprehensively described by Federal Motor Carrier Safety Administration (2009). The most important rules of the U.S. hours of service regulations are that a driver may drive for a maximum of 11 hours after 10 consecutive hours of rest time, and that a driver may not drive after the 14th hour since returning from the last rest period of at least 10 hours. Further, regulations impose constraints on the maximum amount of driving time within a period of seven or eight consecutive days. For simplicity, however, these additional regulations shall not be considered in this contribution.

In this contribution we consider a sequence of locations denoted by $n_1, n_2, \ldots, n_\lambda$ which shall be visited by a single manned vehicle. At each location n_μ some stationary work of duration w_{n_μ} shall be conducted. This work shall begin within a time window denoted by T_{n_μ}. We assume that n_1 corresponds to the driver's location at time zero and that the driver completes her or his work week after finishing work at location n_λ. The driving time required for moving from node n_μ to node $n_{\mu+1}$ shall be denoted by $\delta_{\mu,\mu+1}$. The U.S. Truck Driver Scheduling Problem is the problem of scheduling driving, working, and rest periods in such a way that all customer locations

are visited within the given time windows and that driving and working hours of the truck driver comply with regulations imposed by the U.S. Department of Transportation.

The first work explicitly considering hours of service regulations imposed by the U.S. Department of Transportation within a vehicle routing problem is the work by Xu et al. (2003). They conjecture that determining a minimal cost truck driver schedule for a given sequence of customer locations is NP-hard in the presence of multiple time windows. Recently, Archetti and Savelsbergh (2009) present an algorithm for scheduling driving and working hours of truck drivers in the presence of single time windows and U.S. hours of service regulations. They prove that their algorithm finds a feasible truck driver schedule in $O(\lambda^3)$ time if one exists. This paper shows that by carefully traversing the search space this complexity can be reduced. A tree search algorithm is presented which determines in $O(\lambda^2)$ time whether a feasible schedule exists for a tour of length λ or not. If a feasible schedule exists, the algorithm generates such a schedule. If no feasible schedule exists, the algorithm terminates with a feasible schedule for the largest partial tour $n_1, n_2, \ldots, n_\mu$ with $\mu < \lambda$ for which a feasible schedule exists.

A truck driver schedule can be specified by a sequence of activities to be performed by the driver. Let us denote with DRIVE any period of consecutive driving, with WORK any work period in which the driver is not driving, with REST any period of at least 10 hours in which the driver is neither driving nor working, and with IDLE any period of less than 10 hours in which the driver is neither driving nor working. Let $\mathcal{A} := \left\{ a = (a^{\text{type}}, a^{\text{length}}) \mid a^{\text{type}} \in \{\text{DRIVE}, \text{WORK}, \text{REST}, \text{IDLE}\}, a^{\text{length}} > 0 \right\}$ denote the set of driver activities that may be scheduled.

The algorithm presented in this contribution iteratively appends new activities to a sequence of activities until a truck driver schedule for tour $\theta := (n_1, n_2, \ldots, n_\lambda)$ is found. As the regulation requires that no driving is performed after 14 hours since the end of the last rest period, it may be required to schedule rest periods of more than 10 hours duration. In this contribution we show that we can anyhow restrict our search to truck driver schedules in which each rest period has a duration of exactly 10 hours. For this, however, we have to make sure that the duration of each rest period can be extended in a post-processing step if required. This contribution provides conditions under which such a post-processing step can achieve compliance with the regulation without violating time window constraints.

The state of a truck driver w.r.t. a given sequence of activities can be specified by the time after which all activities are completed, the amount of driving since the last rest period in the sequence, the completion time of the last rest period, and the amount by which the last rest period can be extended. Given such a driver state we can uniquely determine the activities required to travel to the next location in the tour θ. If this location is reached before the opening of the corresponding time window, the driver may either wait idle until the time window opens and start serving the customer at the earliest possible time, or the driver may take an additional rest period of at least

10 hours before serving the customer. For a trip from a location n_μ to $n_{\mu+1}$, we may thus need to consider two alternative truck driver schedules. For the entire tour $\theta := (n_1, n_2, \ldots, n_\lambda)$ we may therefore need to consider $2^{\lambda-1}$ different schedules.

To reduce the number of alternative schedules to be considered in the search we provide dominance criteria that help reducing the computational effort drastically. This contribution presents a tree search algorithm which, in each iteration, cuts off branches of the search tree corresponding to partial schedules dominated by other partial schedules found so far. It is shown that by using these dominance criteria the algorithm terminates after at most $\frac{1}{2}\lambda^2 - \frac{1}{2}\lambda$ iterations.

References

C. Archetti and M. W. P. Savelsbergh. The trip scheduling problem. *Transportation Science*, 43 (4):417–431, 2009. doi: 10.1287/trsc.1090.0278.

Federal Motor Carrier Safety Administration. Interstate truck driver's guide to hours of service. http://www.fmcsa.dot.gov/rules-regulations/truck/driver/hos/fmcsa-guide-to-hos.pdf, 2009.

H. Xu, Z.-L. Chen, S. Rajagopal, and S. Arunapuram. Solving a practical pickup and delivery problem. *Transportation Science*, 37(3):347–364, 2003.

Solving periodic timetabling: Improving the Modulo Network Simplex method

Marc Goerigk

Institut für Numerische und Angewandte Mathematik

Georg-August-Universität Göttingen, Göttingen, Germany

Email: m.goerigk@math.uni-goettingen.de

Anita Schöbel

Institut für Numerische und Angewandte Mathematik

Georg-August-Universität Göttingen, Göttingen, Germany

1   Introduction

The problem of finding a suitable timetable that allows customers to travel without unnecessary delay from one station to another is as economically important as difficult to handle mathematically. Especially the case of periodic timetables, in which events occur repeatedly over a given period, is known to be NP-hard - in fact, even finding a feasible solution is so. Thus even heuristics are rare. On the other hand there are recent achievements like the newly introduced timetable of the Dutch railway system (see [5]) that impressively demonstrate the applicability and practicability of the mathematical model.

Karl Nachtigall and Jens Opitz presented in [1] a new heuristic approach to the periodic timetabling problem based on the classical network simplex method. Besides the drawback that a feasible starting solution still has to be provided, it suffers under relatively high running times and solutions mediocre in quality.

This presentation proposes different approaches we explored to improve the modulo simplex algorithm which result in an algorithm that is able to handle problems of the size of the German intercity rail network.

2   Periodic Timetabling

In 1989 Paolo Serafini and Walter Ukovich introduced in [4] the *Periodic Event Scheduling Problem* (PESP) to treat periodically reoccurring events that have to be scheduled according to given feasible

time spans. Based on the PESP the periodic timetabling problem can be formulated by introducing *Event-Activity Networks* to model the time-dependent behavior of the various vehicles considered. These are directed graphs $G = (\mathcal{E}, \mathcal{A})$ with nodes

$$\mathcal{E} = \mathcal{E}_{arr} \cup \mathcal{E}_{dep}$$

that represent arrival or departure *events* and edges

$$\mathcal{A} = \mathcal{A}_{drive} \cup \mathcal{A}_{wait} \cup \mathcal{A}_{change} \cup \mathcal{A}_{head}$$

representing either driving, waiting and changing *activities* or the necessary security headway between vehicles sharing the same infrastructure.

The goal is to find a timetable assigning a time π_i to each of the events $i \in \mathcal{E}$. Instead of the π_i one can also determine the tension $x_{ij} = \pi_j - \pi_i$ for any activity $a = (i,j) \in \mathcal{A}$. Given a period T and a spanning tree $\mathcal{T} = (\mathcal{E}, \mathcal{A}_{\mathcal{T}})$ with its corresponding network matrix Γ the *periodic timetabling problem* can be formulated as follows.

$$\min \sum_{(i,j) \in \mathcal{A}} \omega_{ij} x_{ij}$$

$$\text{s.t} \quad \Gamma x = T z$$

$$l_{ij} \leq x_{ij} \leq u_{ij}$$

$$x_{ij} \in \mathbb{Z} \ \forall (i,j) \in \mathcal{A}$$

$$z_{ij} \in \mathbb{Z} \ \forall (i,j) \in \mathcal{A} \setminus \mathcal{A}_{\mathcal{T}}.$$

As the variables z_{ij} model the periodic character of the problem, they will be referred to as *modulo parameters.*

Note that the modulo parameters are the reason why this problem is NP-hard. For fixed variables z_{ij} the timetabling problem is called aperiodic and is the dual of a minimum cost flow problem that can be solved efficiently using the well-known network simplex method.

3 The Modulo Network Simplex method

The main idea of the method is to encode solutions as spanning tree structures $(\mathcal{T}_l \cup \mathcal{T}_u)$ by setting the modulo parameters of the tree edges to 0 and the time consumption of these activities either to their respective lower or upper bound. It can be shown that this uniquely determines a periodic timetable. On the other hand, it is shown in [2] that

$$\binom{\pi}{z} \in \mathcal{Q} := \text{conv.hull}\left(\left\{ \binom{\pi}{z} \,|\, l_{ij} \leq \pi_j - \pi_i + T z_{ij} \leq u_{ij}; z \in \mathbb{Z}^m; \pi \in \mathbb{R}^n \right\} \right)$$

is an extreme point of $\mathcal{Q}$ if and only if it is a solution that is given by a spanning tree structure. Thus it is sufficient to investigate only these solutions.

As it is the case in the classical network simplex method, a given feasible spanning tree solution is gradually improved by exchanging tree and non-tree edges that lie in the same fundamental cycle. Due to the modulo parameters, reduced costs as in the classical network simplex do not exist. In consequence, the resulting change of every entry of the simplex tableau has to be calculated, which results in a time-consuming complexity of $\mathcal{O}(n^3)$. Furthermore, as the problem is not convex, local optima will not coincide with global ones, which is the reason why methods of global optimization should be added.

4 Improving the modulo simplex

We proceed in two steps. On one hand, we propose alternative schemes for choosing a base exchange pair, as investigated in [3], in order to improve the running time and the quality of the solution:

Improving running times. As the time needed for investigating a single column of the simplex tableau grows quadratically in its number of non-zero entries, different algorithms are presented that take advantage of sparse columns.

Improving quality. We use and test different versions of the generic Simulated Annealing and Tabu Search algorithms to avoid getting stuck in local optima.

On the other hand, we analyze the single node cuts of [1] and introduce new types of possible cuts:

Single Node Cuts. The time π_i of a single event i is delayed by $\delta \in \mathbb{Z}$. A single node cut is considered to be improving if

$$\sum_{e_{ij} \in \mathcal{A}} \omega_{ij}(x_{ij} - \delta) + \sum_{e_{ji} \in \mathcal{A}} \omega_{ji}(x_{ji} + \delta) < 0. \tag{1}$$

As a given spanning tree structure is optimal with respect to the induced modulo parameters, it can be concluded as a necessary condition that a single node cut changes at least one modulo parameter in order to be improving. We show that this is unlikely to happen, taken into consideration that the tension x_{ij} of at least one of the adjacent edges is set to be l_{ij} or u_{ij}.

Waiting Edge Cuts. To improve the probability of finding a feasible single node cut, another approach is to consider cuts which are induced along an activity (i, j) with a small feasible time span $u_{ij} - l_{ij}$.

Multi Node Cuts. A cut induced by the partition $V_1 \cup V_2$ is defined to be connected if both induced subgraphs $G_1 = (V_1, E(V_1))$ and $G_2 = (V_2, E(V_2))$ are so. As the change of the objective value when applying a cut to a given solution can be calculated by simply adding up the change of every single connected component of the cut, only connected cuts have to be considered. A heuristic algorithm is presented that searches for an improving connected cut.

Random Node Cuts. As the presented formula to calculate the change in the objective value by a single node cut is only a local consideration and thus too pessimistic, cuts may be still improving when (1) does not hold. In our method of random node cuts we apply feasible single node cuts, neglecting whether they improve locally or not.

The different combinations of these methods are compared numerically on data sets based on the German intercity rail network, as part of the *LinTim* project (see [6]), a toolbox of traffic optimization algorithms that allows direct evaluation of the mutual influence of the different planning steps. Our results show that we are able to improve the original modulo simplex significantly.

References

[1] K. Nachtigall and J. Opitz, "Solving Periodic Timetable Optimisation Problems by Modulo Simplex Calculations", ATMOS Workshop 2008.

[2] K. Nachtigall, "Periodic Network Optimization and Fixed Interval Timetables", Habilitationsschrift, 1998

[3] M. Goerigk, "Verallgemeinerte Schnittheuristiken in der periodischen Fahrplangestaltung", Diploma thesis, Georg-August-Universität Göttingen, Göttingen, 2009

[4] P. Serafini and W. Ukovich, "A Mathematical Model for Periodic Scheduling Problems", in *SIAM J. Disc. Math.*, 550-581, 1989

[5] L. Kroon et. al., "The New Dutch Timetable: The OR Revolution", in *Interfaces* 39, 6-17, 2009

[6] M. Schachtebeck and A. Schöbel, "LinTim - A Toolbox for the Experimental Evaluation of the Interaction of Different Planning Stages in Public Transportation", Technical Report 0206, ARRIVAL - Algorithms for Robust and online Railway optimization: Improving the Validity and reliAbility of Large scale systems, 2009

Compact Models for the Mixed Capacitated Arc Routing Problem [1]

Luís Gouveia

Centro IO & Faculdade de Ciências

University of Lisbon

Maria Cândida Mourão

Centro IO & Instituto Superior de Economia e Gestão

Technical University of Lisbon

Email: cmourao@iseg.utl.pt

Leonor Santiago Pinto

CEMAPRE & Instituto Superior de Economia e Gestão

Technical University of Lisbon

Mixed Capacitated Arc Routing

Capacitated Arc Routing (CARP) models are widely used in distribution or collection problems where vehicles with limited capacity, perform certain activities that are continuously distributed along some pre-defined links (routes, streets) of an associated network. The Mixed Capacitated Arc Routing Problem (MCARP) describes a more realistic scenario since it considers directed as well as undirected required links. The MCARP is NP-hard as it generalizes the CARP (Golden and Wong [8]) which is known to be NP-hard. Many real-world applications can be studied in the context of CARP or MCARP models, as reported in Dror's book [5] or, for instance, papers from Assad and Golden [1], Eiselt et al [6]-[7] and Wøhlk [11].

The MCARP study reported in this presentation is motivated by a household refuse collection problem in a quarter of Lisbon. Each quarter can be planed separately as a fleet of identical vehicles is assigned to its refuse collection. Vehicles depart from a special point, the depot, where they should return after completing their collecting period and then empty at the dumpsite. In order to define a problem similar to the ones in the CARP literature, it is assumed that depot and dumpsite coincide. For simplicity, each vehicle performs only one trip compatible with its capacity.

A compact formulation for the MCARP is presented and its validity is shown. In this model flow variables are used to impose the connectivity of the solutions. Variables are indexed by vehicle to guarantee a matching between trips and vehicles. The objective function represents the total cost

(service, deadheading and dump costs) to minimize. Conditions are defined to impose trips continuity at each node; to guarantee that the service in each required link is performed; to ensure that the dump cost is adequately charged in the objective function; to force trips connectivity and also to guarantee that the vehicles capacity is not exceeded.

The model is used within an ILP package to solve medium sized problems and to produce lower bounds on larger instances. Lower bounds are also obtained from the associated linear programming relaxation.

Essentially, our model differs from the one first presented by Golden and Wong [8] for the CARP since it considers the mixed case and it uses flow variables with a different interpretation. Further more, our model includes additional constraints so as to ensure that trips start at the depot. The addition of some valid inequalities strengthens the corresponding linear programming formulation, namely a depot degree constraint, lower bounds on the flow variables, and different types of breaking symmetries restrictions, this latter being used to diminish the number of alternative solutions for the same set of trips and they come up as extensions to a very recent paper [9] published by the authors and that may be seen as a guideline for the present study.

We also present and discuss an aggregated model, where links and flow variables are not disaggregated by vehicle. This option, although not valid, it is attractive for three reasons: (a) the integer optimal solution, providing good lower bounds, is easier to compute than the optimal integer solution of the previous model; (b) for some instances, the optimal solution of the aggregated model is also optimal for the original problem; and (c) proving that the linear programming relaxation values of the two models, aggregated and disaggregated, are equal is an attractive feature, as it then offers a faster alternative to obtain the same lower bounds.

The linear programming relaxation value of the aggregated model is also improved on by adding some valid inequalities. These inequalities are the aggregated version of the inequalities previously considered, and lead to a model with a tighter linear programming bound. Again, it can be shown that the linear programming relaxation value of the two enhanced models, aggregated and disaggregated, is equal.

The first study on lower bounds for the MCARP based on a formulation using only one variable per edge ([2]) is due to Belenguer et al. [3]. Embedded in a cutting plane algorithm and with several valid inequalities added, this model, although not valid for the MCARP, produces good lower bounds. In our aggregated model both capacity and connectivity constraints are enforced by using the additional flow variables and constraints linking the two sets of variables, whilst in Belenguer and Benavent [2] no extra set of variables is used, while in turn, an exponential number of constraints forces connectivity.

Computational experiments were conducted using CPLEX 11.0 to evaluate the performance of the models. For that purpose well known extended CARP instances are used, namely: gdbe [10]; mval and lpr [3]; alba, madri and alda [4]. The aggregated model is competitive as the lower bounds provided do not differ from the better upper bound more than 5%, and the CPU times are quite similar to the ones of Belenguer et al. [3]. The compact model for the MCARP was able to produce optimal solutions for some instances.

The bounds are comparable with the ones presented by Belenguer et al. [3], the best known from the MCARP literature used for medium and large sized instances.

The presentation is organized as follows. Firstly, the MCARP is defined. A valid formulation and some inequalities are explained, as a foreground into the discussion of its aggregated version. Computational results on a set of benchmark problems and final remarks conclude the presentation.

References

[1] A.A. Assad, B.L. Golden, "Arc Routing Methods and Applications", in *Handbooks in Operations Research and Management Science, 8, Network Routing*, M.O. Ball, T.L. Magnanti, C.L. Monma, G.L. Nemhauser (eds), 375-483, Elsevier Science Publishers, Amsterdam, 1995.

[2] J.M. Belenguer, E. Benavent, "A Cutting Plane Algorithm for the Capacitated Arc Routing Problem", *Computers & Operations Research* 30, 705-728 (2003).

[3] J.M. Belenguer, E. Benavent, P. Lacomme, C. Prins, "Lower and Upper Bounds for the Mixed Capacitated Arc Routing Problem", *Computers & Operations Research* 33, 3363-3383 (2006).

[4] A. Corberán, G. Mejía, J.M. Sanchis, "New Results on the Mixed General Routing Problem", *Operations Research* 53, 363-376 (2005).

[5] M. Dror (ed.), *Arc Routing: Theory, Solutions and Applications*, Kluwer Academic Publishers, Boston (2000).

[6] H.A. Eiselt, M. Gendreau, G. Laporte, "Arc Routing Problems, Part I: The Chinese Postman Problem", *Operations Research* 43, 231-242 (1995).

[7] H.A. Eiselt, M. Gendreau, G. Laporte, "Arc Routing Problems, Part II: The Rural Postman Problem", *Operations Research* 43, 399-414 (1995).

[8] B.L. Golden, R.T. Wong, "Capacitated Arc Routing Problems", *Networks* 11, 305-315 (1981).

[9] L. Gouveia, M.C. Mourão, L.S. Pinto, "Lower Bounds for the Mixed Capacitated Arc Routing Problem", *Computers & Operations Research* 37, 692-699 (2010). doi:10.1016/j.cor.2009.06.018.

[10] P. Lacomme, C. Prins, W. Ramdane-Cherif, "An Integer Linear Model for General Arc Routing Problems", Working Paper, Université Technologie de Troyes, 2003.

[11] S. Wøhlk, "A Decade of Capacitated Arc Routing", in *The Vehicle Routing Problem – Latest Advances and New Challenges*, B. Golden, S. Raghavan and E. Wasil (eds), 29-49, Springer, NY, 2008.

Bi-objective Multimodal Time-Dependent Shortest Viable Path Algorithms

Fallou Gueye[1,2,3], **Christian Artigues**[1,2], **Marie-José Huguet**[1,2],
Frédéric Schettini[3] and **Laurent Dezou**[3]

[1]CNRS; LAAS; 7 avenue du Colonel Roche, F-31077 Toulouse, France

[2]Université de Toulouse ; UPS, INSA, INP, ISAE; LAAS; F-31077 Toulouse, France

[3]MobiGIS; ZAC Proxima, rue de Lannoux, 31310 Grenade Cedex France

Corresponding author : Fallou Gueye - Email: fgueye@laas.fr

Problem definition

For the passengers public transport, the development of alternatives to the individual vehicles introduces new challenges for the organization of the travels via the use of various modes of transportation. Taking into account the multimodality of urban transportation networks for individual passenger's itinerary computation introduces a number of additional constraints such as time dependent travel times, restriction and/or preferences in using some modes.

The central problem considered in this work is a bi-objective shortest path on a multimodal and time-dependent network with a single source O, destination D and departure time t. The goal is to find all the non-dominated paths under the two objectives "travel time" and "number of modal transfers". The problem can be defined as the time-dependent extension of the one solved in [2]. We consider that the urban multimodal network is represented by a multi-level graph $G(V, E)$ in which each level is associated to a transportation mode. Considering a set of modes M, each node $i \in V$ is associated with a mode $m_i \in M$. An arc linking two nodes with different modes is called a *transfer* arc. Each arc is associated with a function $a_{ij}(t)$ giving the arrival time to node j given that departure time from i is t. Depending on the mode m, this function can be or not time-dependent. For instance, bicycle and walk are time-independent modes since that they do not vary with the traffic level but car mode could be time-dependent and bus mode depends on timetables. We consider a FIFO network ($a_{ij}(t)$ is non decreasing $\forall (i, j) \in E$) [4]. A viable path is a path that respects given constraints on the sequence of modes (e.g. taking a private car after leaving the bus is not viable). We model the mode viability of paths by a finite state automaton as in [2]. Let S denote the set of states. A transition function δ is defined where $s' = \delta(m, m', s)$ is

the state reached if a transfer from mode m to mode m' is performed. By convention, the state at origine is $s = 0$ and $\delta(m, m', s) = 0$ means that the transition is not viable. This approach allows also to specify a subset of authorized states $S(D)$ to reach the destination (e.g. if no parking is allowed at the destination, private car should be dropped in a public parking area).

Related work

If modes are ignored, the problem resorts to the time-dependent single source, destination and departure time shortest path which can be solved by a standard label-setting algorithm in $O(|E| \log n)$ time [4] using a binary heap [1], where n is the number of nodes and assuming $a_{ij}(t)$ is computed in $O(1)$. For a multimodal network where all paths are viable and time-independence is assumed, a topological algorithm is proposed in [4] to find all the non-dominated path with a number of transfers lower than a given upper limit k_{max} in $O(k_{max}|E| \log n)$ for the label-setting variant. This method has been extended to path viability represented by an automaton by [2]. Although no information is given on complexity issues in [2], the algorithm can be implemented in $O(k_{max}|S||E| \log n|S|)$. In [1, 5], multimodal and time-dependent shortest-path are considered but only the minimum time objective is tackled. In [5], non FIFO networks are considered. In [1], the objective is to compute the $K-$shortest paths under an upper bound of the maximum allowed number of transfers. In these papers, experimental validations are limited to small networks. The largest one, presented in [1], involves 1000 nodes and 2830 arcs.

Proposed Algorithms

As the network is FIFO, the Lozano and Storchi algorithm [2] can be extended trivially to time dependence. We propose a label setting implementation of their method, denoted LS. Using two buckets (priority queues Q_{now} and Q_{next} [4]), the method computes the non-dominated shortest paths in increasing number of transfers k from 0 to a given k_{max} even if there is no shortest path for given number transfert. For each node i, there is at most $|S|$ non-dominated labels for a given number of transfers k and $a_{ij}(t)$ is computed in $O(1)$. So the algorithm LS runs in $O(n|S||E| \log n|S|)$.

We propose a new algorithm which computes only non dominated shortest paths non decreasing order of the number of transfer. A label is now denoted $t_i^s(k)$ and corresponds to the shortest path from the source to i in state s and k transfers. Instead of considering Q_{now} and Q_{next}, we build incrementally a list $\mathcal{Q} = \{Q_0, Q_1, \ldots\}$ of priority queues such that $Q_k \in \mathcal{Q}$ contains labels leading to k modal transfers. Non-dominated labels $t_i^s(k)$ are stored. Let $Dom(s)$ denote the set of states, *dominant* for state s. A state s' dominates a state s if all modes viable from state s are also viable from state s'. Note we have $s \in Dom(s)$. The resulting algorithm (MLMH) is described below and has a worst complexity in $O(n|S||E| \log n|S| + n^3|S|)$ if k_{max} is not part of the input.

---

1. Better amortized time complexity can be obtained with Fibonacci heaps which were not used in our study

Example :

$Q_0 = \{1\}$

$i = 1, k = 0, Q_0 = \{3\}, t_3^0 = 5, Q_1 = \{2\}, t_2^1 = 1$

$i = 2, k = 1, Q_1 = \{4\}, t_4^1 = 6, Q_2 = \{3\}, t_3^2 = 2$

$i = 3, k = 2, Q_2 = \{5\}, t_5^2 = 7, Q_3 = \{4\}, t_4^3 = 3$

$i = 4, k = 3, Q_3 = \emptyset, Q_4 = \{5\}, t_5^4 = 4$

$i = 5, k = 4, Q_4 = \emptyset$ (shortest path with 4 transfers)

$i = 3, k = 0 \ Q_0 = \{5\}, t_5^0 = 10$

$i = 4, k = 1, Q_1 = \emptyset$

$i = 5, k = 2, Q_2 = \emptyset$ (shortest path with 2 tranfers)

...
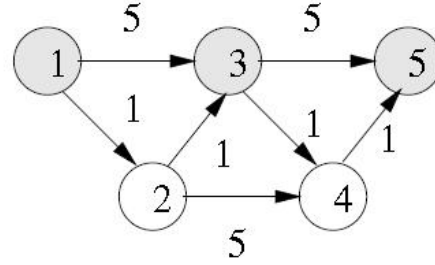
Algorithm 1 Multi-labels multi-heap algorithm (MLMH)

Require: $G(V, E), O, D, d_{ij}, \forall (i, j) \in E, t$

1: Set $\mathcal{Q} = \{Q_0 := \{(O, s_0, 0)\}\}, t_{O,s_0}^0 := t, p_{O,s_0}^0 := (0, s_0, 0), t_{i,s}^0 := \infty, \forall i \in V, \forall s \in S, (i, s) \neq (0, s_0)$

2: set $k_{\max} = \infty$

3: **repeat**

4: Let $(i, s, k) := argmin\{t_{i',s'}^{k'} | (i', s', k') \in \mathcal{Q}\}$ and set $Q_k := Q_k \setminus \{i\}$(minimum time label)

5: **if** $i = D$ and $s \in F$ **then**

6:   store $t_{i,s}^k$ and $p_{i,s}^k$ as the shortest path with k transfers. Discard all $Q_{k'}$ with $k' \geq k$. set $k_{\max} := k - 1$

7: **else if** $t_{i,s}^k < t_{i,s'}^{k'}, \forall k' \leq k, \forall s' \in dom(s)$ **then**

8:   **for** $j \in FS(i)$ **do**

9:    set $s' := \delta(m_i, m_j, s)$

10:    **if** $s' \neq -1$ and $m_i = m_j$ and $a_{ij}(t_{i,s}^k) < t_{j,s'}^k$ **then**

11:     set $t_{j,s'}^k := a_{ij}(t_{i,s}^k), p_{j,s'}^k := (i, s, k)$ and $Q_k := Q_k \cup \{(j, s', k)\}$

12:    **else if** $s' \neq -1$ and $m_i \neq m_j$ and $a_{ij}(t_{i,s}^k) < t_{j,s'}^{k+1}$ and $k + 1 <= k_{\max}$ **then**

13:     set $t_{j,s'}^{k+1} := a_{ij}(t_{i,s}^k), p_{j,s'}^{k+1} := (i, s, k)$ and $Q_{k+1} := Q_{k+1} \cup \{(j, s', k + 1)\}$

14:    **end if**

15:   **end for**

16: **end if**

17: **until** $k_{\max} < 0$ or $\mathcal{Q} = \emptyset$

We propose an adaptation of MLMH in a bidirectional way [3]. The proposed bidirectional algorithm (MLMH-BI) maintains, in a similar way as in MLMH, two priority queue lists $\mathcal{FQ}$ for the forward one and $\mathcal{BQ}$ for the backward one such that FQ_k contains forward labels $ft_i^s(k)$ representing paths reaching i in state s and k transfers and BQ_k contains backward labels $bt_i^s(k)$ representing paths issued from i in k transfers and *possibly* in state s. For backward search, the state at destination is unknown and one has to generate labels in potential states (that may never be reachable). Morever, in backward search,arcs are considered in none time dependant way. When a connection is made between a label $ft_i^s(k)$ and a label $bt_i^s(q)$, if condition $ft_i^s(k) + bt_i^s(q) \leq \min \mathcal{FQ} + \min \mathcal{BQ}$, all priority queues $FQ_{k'}$ and $BQ_{k'}$ with $k' \geq k + q$ can be discarded. Another extension of the bidirectional method to time-dependent networks can be found in [3]. All the above-three algorithms can be further accelerated using the A* principle (the cost of a node in the label setting process is the cost from the origin plus the estimated cost to the destination) yielding variants LS-A*,

MLMH-A* and MLMH-BI-A*.

Experiments

To adapt LS for computing the maximal number of modal transfers, we add to this algorithm a step which computes the shortest path independently to use modes; this shortest path gives then the maximum number of modal transfers. All algorithms have been implemented in C++ and tested on an Intel Pentium2, 2.4 GHz processor on Windows using a real-world multimodal network covering the urban area of Toulouse with 10397 nodes and 22004 arcs. Considered modes are bus, metro, walking and private vehicle, there are time-tables for buses and frequencies for metro from 05pm to 11 :59 am. Experiments concern 100 randomly generated trips in which travel times vary from $28mn$ to $163mn$ (as walking can be the only alternative with no transfer) and lead from 0 to 3 modal transfers. The number of pareto optimal solutions vary from 2 to 4 solutions with 3 non dominated solutions per itinerary on average. The table below presents the average CPU time (ms) obtained by the 6 variants with and without inclusion of dominance rules to prune labels.

	LS	MLMH	MLMH-BI	LS-A*	MLMH-A*	MLMH-BI-A*
av CPU (ms) - without dom. rules	434	470	424	424	326	415
av CPU (ms) - with dom. rules	351	362	433	339	**288**	400

The best CPU time (in bold) is obtained by MLMH-A* which significantly outperforms the LS variants. The dominance rules are useful for all algorithms , the bidirectional approach is not the most efficient on our tests, experiments on larger networks are necessary to precise these results.

Références

[1] M. Bielli, A. Boulmakoul, H. Mouncif. "Object modeling and path computation for multimodal travel systems" *European Journal of Operational Research* 175(3), 1705–1730, (2006).

[2] A. Lozano, G. Storchi. "Shortest viable path algorithm in multimodal networks", *Transportation Research Part A : Policy and Practice* 35(3), 225-241, (2001).

[3] G. Nannicini, D. Delling, D. Schultes, L. Liberti. "Bidirectional A* Search in Time-Dependent Road Networks", *Optimization Online*, paper 2154, (2008).

[4] S. Pallottino, M. G. Scutella. "Shortest path algorithms in transportation models : Classical and Innovative Aspects", *Equilibrium and Advanced Transportation Modelling*, 245-281,(1998).

[5] A. Ziliaskopoulos, W. Wardell. "An intermodal optimum path algorithm for multimodal networks with dynamic arc travel times and switching delays", *European Journal of Operational Research* 125(3), 486-502, (2000).

Ship Traffic Optimization for the Kiel Canal

Elisabeth Günther

Institut für Mathematik

Technische Universität Berlin

eguenth@math.tu-berlin.de

Marco E. Lübbecke

Fachbereich Mathematik

Technische Universität Darmstadt

luebbecke@opt.tu-darmstadt.de

Rolf H. Möhring

Institut für Mathematik

Technische Universität Berlin

rolf.moehring@tu-berlin.de

1 The Kiel Canal

The Kiel Canal connects the North and Baltic seas and is ranked among the world's three major canals. In fact, in terms of traffic, it is the busiest artificial waterway worldwide. In a billion Euro project, the German Federal Waterways and Shipping Administration plans to enlarge the canal during the coming years. This project is about contributing to a well-founded advise on how the enlargement can be optimally done. In order to evaluate the various construction possibilities it is indispensable to first provide an accurate model for the ship traffic and designing an algorithm which (ideally optimally) controls it. This paper is about such optimal traffic control.

The problem very roughly is as follows. There is bi-directional ship traffic on the canal; there are several locks at both ends. Ships are classified in different size categories. Passing and overtaking is allowed only if the sizes of the two ships do not exceed a given threshold which depends on the meeting point. If otherwise a *conflict* occurs, ships have to wait at designated, capacitated places, the *sidings*. The objective is to minimize the total passage time, including lock and siding waiting times. The overall scheduling is currently done by two teams of experienced planners, one for the locks, one for the sidings. In this abstract we concentrate on the latter problem, but both will be treated in an integrated way during the project. Despite significant differences there are certain similarities to train scheduling on a single track line [1].

2 Algorithms for Ship Traffic Control

The canal consists of sidings (set T) which alternate with canal *segments* (set E). More precisely, we represent the canal as $t_1, e_1, t_2, e_2, \ldots, e_{|E|}, t_{|T|}$ with sidings $t_i \in T$ and segments $e_i \in E$. We have a set S of n ships, each pair $(s_1, s_2) \in S \times S$ of which may be forbidden to pass each other on a given segment $e \in E$. In this case, we say that (s_1, s_2) has a *conflict* on e and write $(s_1, s_2) \in C_e$. A ship $s \in S$ needs time $\tau_{s,i} \geq 0$ to pass through a siding or segment $i \in T \cup E$. We formulate a mixed integer program (MIP) which is based on deciding for each ship $s \in S$ when it departs from a siding or segment (this is a natural way of encoding a solution). We have a variable $d_{s,i}$, $i \in T \cup E$ for the respective departure time. For each segment $e \in E$ and each conflicting pair $(s_1, s_2) \in C_e$ of ships we decide which ship will enter e first. This decision is represented by a binary variable $z_{s_1, s_2, e}$ which assumes a value of 1 if and only if s_1 enters e before s_2 does. Finally, a variable $w_{s,t} \geq 0$ represents the waiting time of ship s in siding t. The model reads as follows.

$$\text{minimize} \quad \sum_{s \in S, t \in T} w_{s,t} \tag{1}$$

$$\text{s.t.} \quad d_{s,t_i} + \tau_{s,e_i} = d_{s,e_i} \qquad s \in S, i = 1, \ldots, |E| \tag{2}$$

$$d_{s,e_{i-1}} + \tau_{s,t_i} + w_{s,t_i} = d_{s,t_i} \qquad s \in S, i = 2, \ldots, |T| \tag{3}$$

$$z_{s_1, s_2, e} = 1 \quad \Rightarrow \quad d_{s_1, e} \leq d_{s_2, e} - \tau_{s_2, e} \qquad e \in E, (s_1, s_2) \in C_e \tag{4}$$

$$z_{s_1, s_2, e} = 0 \quad \Rightarrow \quad d_{s_2, e} \leq d_{s_1, e} - \tau_{s_1, e} \qquad e \in E, (s_1, s_2) \in C_e \tag{5}$$

$$\underline{d}_{s,i} \leq d_{s,i} \leq \overline{d}_{s,i} \qquad s \in S, i \in T \cup E \tag{6}$$

$$w_{s,t} \geq 0 \qquad s \in S, t \in T \tag{7}$$

$$z_{s_1, s_2, e} \in \{0, 1\} \qquad e \in E, (s_1, s_2) \in C_e \tag{8}$$

The objective function (1) accounts for minimum total waiting time of all ships, which—when ships travel at their respective full speeds, what is an assumption close enough to reality also for manual planners—is equivalent to minimum total passage time. Constraints (2) and (3) ensure a consistent setting of departure times along each ship's route. Our notation assumes that all ships travel upstream, that is, in the direction of increasing indices of sidings and segments, but—abusing notation—departure times of downstream ships are set analogously. We avoid the distinction here for the sake of an easier presentation.

In a continuous planning, ships may start and end at any point of the canal, not necessarily only at the ends; thus there are release times for each ship $s \in S$. This implies that constraints (2) and (3) ensure that d variables increase in the travel direction of the ship, but also that they decrease, and even below zero, in the other direction. For this reason, d variables are not restricted in sign, but there are lower and upper bounds $\underline{d}_{s,i}$, $\overline{d}_{s,i}$ imposed on departure times. These bounds are

given in (6). Precedence constraints (4) and (5) link the d and z variables. It is easy to incorporate safety distances between ships as well which in particular avoid that ships are at the same place concurrently. Note that we do not respect siding capacities. In fact, the logic of this formulation assumes that *all* ships wait at the same point at the end of a siding, no matter how many ships they are. Thus, this MIP is a relaxation of our original problem formulation.

In fact, to overcome this inaccuracy one needs some sort of time and space discretization. In particular the latter one causes some difficulties due to lots of different ship sizes and arbitrary start and end points. Instead of respecting such a discretization in the model directly we chose to let the algorithm take care of that. We designed a successive shortest path algorithm respecting blocked time windows which constructs conflict-free dynamic routes for the ships one after another. In addition to the basic version described in [2] our algorithm allows more than one ship to be on an edge, handles arbitrary start and end positions, and tries different valid waiting positions in a siding. Thus, an initial solution is constructed which is feasible with respect to all constraints (also those not mentioned here). The running time is a few seconds. A local search based on a loss-benefit calculation improves that first solution in a rolling horizon manner, mildly imitating a manual planner's procedure. Since this local search uses the dynamic routing algorithm as a subroutine also the final schedule respects all the constraints. Our solutions are visualized in exactly the same way the planners are used to see them, in a way-time diagram of the canal, see Figure 1.

3 Lower Bounds

Not only for theoretical reasons we want to assess the quality of the heuristic solutions we obtain. The above MIP is suited for this purpose, however, for practical problem instances it takes much too long to solve it to integer optimality, mainly because of the poor LP relaxation. Instead, we developed a much more involved model which is solved by a full-fledged branch-and-price algorithm. The idea is to base an integer program on schedules for all ships on a given segment. Via a sort of *flow conservation constraints* these schedules for segments are linked together to form a valid schedule for the entire canal. The pricing subproblem is a scheduling problem which is interesting in its own right. In principle, this model is suited to seamlessly integrate the lock scheduling problem as well. To price the corresponding variables, a particular two-dimensional packing problem needs to be solved.

4 Preliminary Results and Perspectives

We are provided with historical ship traffic data from recent years and forecasts for the year 2025. This way, we can compare our schedules to the manual plans. We refrain from giving numerical

results here. However, in an intermediate evaluation the officers in charge of the enlargement were impressed and satisfied by the possibilities operations research and discrete optimization have to offer. We were encouraged to extend our models to take all further details of the planning situation into account, in particular the locks at both ends of the canal.
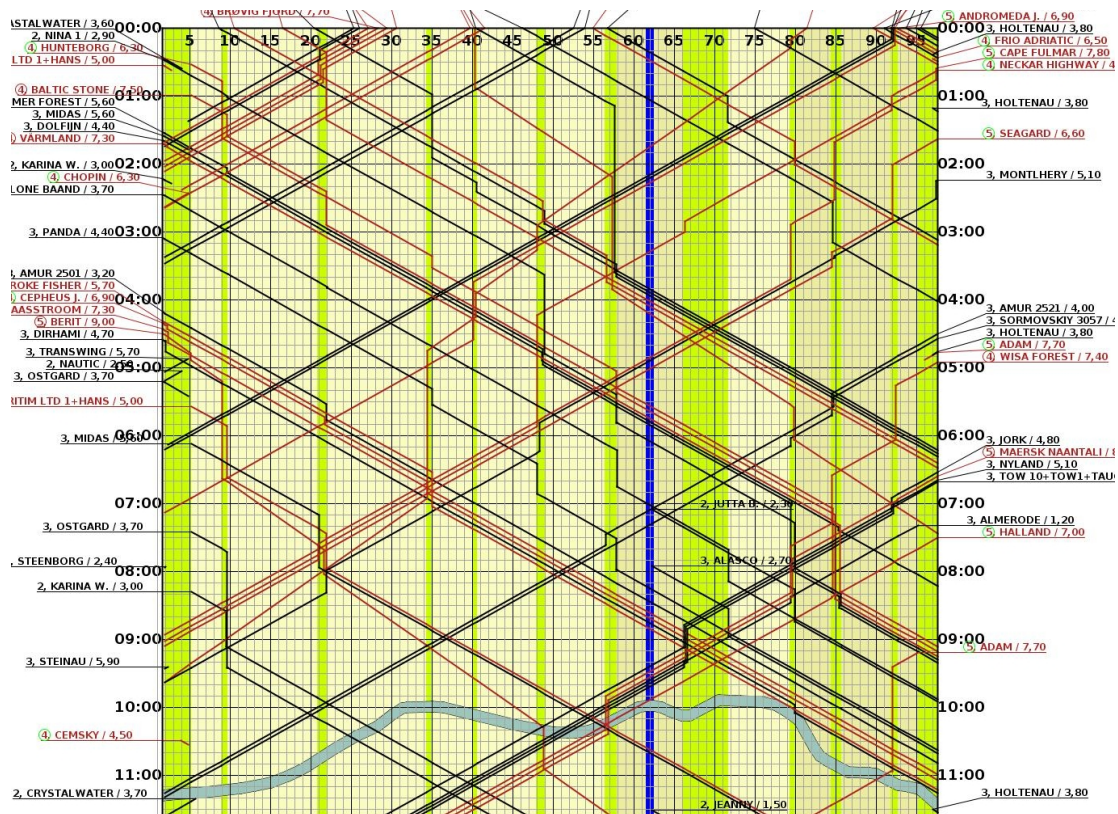
Figure 1: Example of a way-time diagram showing one of our solutions.

References

[1] A. Caprara, M. Fischetti, and P. Toth. Modeling and solving the train timetabling problem. *Oper. Res.*, 50(5):851–861, 2002.

[2] R.H. Möhring, E. Köhler, E. Gawrilow, and B. Stenzel. Conflict-free real-time AGV routing. In H. Fleuren, D. den Hertog, and P. Kort, editors, *Operations Research Proceedings 2004*, pages 18–24, Berlin, 2005. Springer-Verlag.

Fleet size and mix and periodic routing of offshore supply vessels

Kjetil Fagerholt

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Lars Magne Nonås

Norwegian Marine Technology Research Institute (MARINTEK)

Bjørn Egil Asbjørnslett

Department of Marine Technology

Norwegian University of Science and Technology

Elin E. Halvorsen-Weare

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, NO-7491 Trondheim, Norway

and Norwegian Marine Technology Research Institute (MARINTEK)

Email: elin.halvorsen-weare@iot.ntnu.no

1 Introduction

Norway is a major oil and gas producing country with offshore installations in the Norwegian Sea and North Sea. Statoil is the leading operator on the Norwegian continental shelf and controls several onshore supply depots along the Norwegian coastline where supply vessels load cargoes to and discharge backloads from the offshore installations. Statoil hires supply vessels for the supply service on time charter and construct weekly routes and schedules, which are typically valid for a few months ahead depending on when major changes in demand are expected.

The *supply vessel planning problem* consists of deciding weekly voyages and schedules from an onshore supply depot while at the same time determining the optimal fleet of supply vessels. The supply vessel planning problem is a fleet size and mix and periodic routing problem as a schedule is determined that is to be repeated on a weekly basis.

On request from Statoil we started up a project aiming to develop a tool that could be used as decision

support for the supply vessel planning problem. A new model and solution method were developed to be used for this purpose and is presented here. A problem description is given in Section 2, followed by a description of the solution method in Section 3 before the work is concluded in Section 4.

2 Problem description and modelling assumptions

The supply vessel planning problem consists of identifying the optimal fleet of supply vessels that are to service a given number of offshore installations from one common onshore depot while at the same time determining the weekly routes and schedules for the supply vessels. A route is in this setting a combination of one or more voyages that a vessel sails during a week. In each voyage the vessel starts at the supply depot and visits a number of offshore installations before returning to the supply depot.

There are several constraints that need to be considered. The offshore installations may have opening hours for when they can receive visits from supply vessels. Limited capacity at the supply depot sets a bound on the number of vessels that may be prepared for a new voyage on a given day. The fleet of available supply vessels is heterogeneous and each vessel has a given service speed, time charter rate, and capacity. Each installation has a given demand and requires a given number of visits during the week. The total demand for all installations visited on a voyage cannot exceed the capacity of the vessel sailing the voyage. All installations have a given service time that is the lay time for the supply vessels at the installation. The durations of the voyages are measured in an integer number of days and are limited to be two or three days. These limitations are to avoid too short voyages with too few visits that do not exploit the supply vessels' capacities properly, and too long voyages as they involve more uncertainty with regard to sailing time. For the same reasons there are also defined minimum and maximum numbers of visits for a voyage.

Finally, the departures from the supply depot to a given installation should be fairly evenly spread throughout the week. It is more important to spread the departures to an installation than the actual visits, as the demand from an installation is reported continuously. This means, for example, that if an installation requires three services a week and the supply vessels visiting the installation leave the supply depot on three consecutive days, a demand may be called in after the third vessel has left and it will be almost five days until next departure. In such cases it may be necessary to reroute other supply vessels or send out a helicopter to meet the demand. Such solutions will in most cases be very costly and will be less needed if the departures are evenly spread.

3 Solution method

To solve the supply vessel planning problem, a voyage based solution method has been developed. The mathematical formulation for the problem is an extension of the model by (1) who studied a similar problem.

In the mathematical formulation of the problem, let $v \in \mathcal{V}$ represent a vessel, $i \in \mathcal{N}$ an offshore installation, and $t \in \mathcal{T}$ a time period (one day). Set $\mathcal{R}_v \subseteq \mathcal{R}$ contains candidate voyages that vessel v may

sail, indexed by r. These candidate voyages are generated a priori by a full enumeration procedure. Subsets $\mathcal{R}_{v2} \subseteq \mathcal{R}_v$ and $\mathcal{R}_{v3} \subseteq \mathcal{R}_v$ contain voyages for vessel v of duration two and three days, respectively. Further, parameter C_v^{TC} is the time charter cost for a week for vessel v, C_{vr}^S is all sailing and service costs associated with vessel v sailing voyage r, B_t is the number of supply vessels that may be serviced at the onshore supply depot on day t, S_i is the number of services offshore installation i requires during a week, A_{vir} is a constant that is equal to 1 if vessel v services installation i on voyage r and 0 otherwise, D_{vr} is the duration of voyage r sailed by vessel v rounded up to nearest whole day, and F_v is the maximum number of days vessel v may be in service during the planning horizon. Binary variable δ_v equals 1 if vessel v is used in the solution and 0 otherwise, and binary variable x_{vrt} equals 1 if vessel v starts to sail voyage r in time period t and 0 otherwise. The mathematical formulation then becomes:

$$min \sum_{v \in \mathcal{V}} C_v^{TC} \delta_v + \sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} \sum_{t \in \mathcal{T}} C_{vr}^S x_{vrt}, \tag{1}$$

$$\sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}} \sum_{t \in \mathcal{T}} A_{vir} x_{vrt} \geq S_i, \quad i \in \mathcal{N}, \tag{2}$$

$$\sum_{r \in \mathcal{R}_v} \sum_{t \in \mathcal{T}} D_{vr} x_{vrt} - F_v \delta_v \leq 0, \quad v \in \mathcal{V}, \tag{3}$$

$$\sum_{v \in \mathcal{V}} \sum_{r \in R_v} x_{vrt} \leq B_t, \quad t \in \mathcal{T}, \tag{4}$$

$$\sum_{r \in \mathcal{R}_{v2}} x_{rvt} + \sum_{r \in \mathcal{R}_v} x_{rv,((t+1) \bmod |\mathcal{T}|)} \leq 1, \quad v \in \mathcal{V}, t \in \mathcal{T}, \tag{5}$$

$$\sum_{r \in \mathcal{R}_{v3}} x_{rvt} + \sum_{r \in \mathcal{R}_v} \sum_{\nu=1}^{2} x_{rv,((t+\nu) \bmod |\mathcal{T}|)} \leq 1, \quad v \in \mathcal{V}, t \in \mathcal{T}, \tag{6}$$

$$\delta_v \in \{0,1\}, \quad v \in \mathcal{V}, \tag{7}$$

$$x_{vrt} \in \{0,1\}, \quad v \in \mathcal{V}, r \in \mathcal{R}_v, t \in \mathcal{T}. \tag{8}$$

The objective function (1) minimizes the number of supply vessels used and the costs of the voyages sailed by the vessels. Constraints (2) ensure that all installations get their required number of visits during the planning horizon. Then constraints (3) ensure that the total duration of all voyages sailed by a vessel does not exceed the maximum number of days the vessel may be in service during the planning horizon, and that the variable δ_v must take the value 1 if vessel v is used. Constraints (4) ensure that there are no more vessels loaded in the supply depot on day t than there is capacity to service. Constraints (5) and (6) prevent overlapping voyages. Finally, constraints (7) and (8) set the binary requirements for variables δ_v and x_{vrt}, respectively.

There are several ways to formulate constraints that will ensure that departures to installations are fairly evenly spread throughout the planning horizon. As an example, for installations requiring three visits a

week, the following formulation is used:

$$\sum_{v \in \mathcal{V}} \sum_{r \in \mathcal{R}_v} \sum_{\nu=0}^{2} A_{vir} x_{vr,((t+\nu) \bmod |\mathcal{T}|)} \geq 1, \quad i \in \mathcal{N}_3, t \in \mathcal{T}, \tag{9}$$

Here the set $\mathcal{N}_3$ contains all installations that require three visits. Then constraints (9) ensure that there will be at least one departure to these installations during a period of three days (given that the planning horizon is one week).

The mathematical formulation presented so far gives a good description of the real life supply vessel planning problem. Other practical aspects of the problem may appear, and many of these can be handled by the model by adding additional constraints. As an example, this could be case specific considerations, where for instance certain installations require visits by vessels departing the supply depot on given days.

The solution method has been tested on instances based on real data provided by Statoil. The test results show that the method will provide good solutions to the supply vessel planning problems within short CPU time, and for many instances optimal solutions were obtained in less than an hour of CPU time.

4 Concluding remarks

We have presented a real life fleet size and mix and periodic routing problem that appears in the offshore supply vessel service, and a solution method has been proposed. This is a real life problem that originates from a project performed together with the leading operator on the Norwegian continental shelf, Statoil. The solution method has been tested on instances based on real life data, and test results show that the method works well for the purpose of solving supply vessel planning problems of a realistic size. A version of the solution method is used by Statoil in their planning operations to establish routing alternatives and optimal plans for their supply vessels. So far, the model has played an essential role in the process of reducing the number of supply vessels at one onshore supply depot. According to Statoil, the annual cost reductions made possible by the use of the voyage based model at this supply depot has been estimated to 3 million USD.

References

[1] K. Fagerholt and H. Lindstad, "Optimal policies for maintaining a supply service in the Norwegian Sea", *OMEGA-The International Journal of Management Science* 28, 269-275 (2000).

Second Best Congestion Pricing for Transit Networks

Patrice Marcotte

Department of Computer Science and Operations Research

Université de Montréal

Younes Hamdouch

Department of Business Administration

United Arab Emirates University, Al Ain, UAE

Email: younes.hamdouch@uaeu.ac.ae

1   Introduction

In transportation science, congestion pricing is often discussed in the context of vehicular traffic networks (see, e.g., [1] and [4]) where the problem of setting tolls to reduce congestion can be classified as first and second best. In first-best toll pricing, one assumes that optimal travel delays can be induced, for instance through marginal cost pricing. This is the case when all links of the network can be tolled. Whenever "optimal" tolls are not unique, one can optimize a secondary objective (see, e.g. [4]). In contrast, second-best toll pricing problem assumes that some roads are not tollable. In the congestion pricing literature, second-best toll pricing problem is modeled either as a bilevel optimization problem (see, e.g., [2] and [5]) or a mathematical program with equilibrium constraints (see, e.g., [6]).

In this paper, we address second-best congestion pricing of a transit system, with the aim to encourage passengers to select travel strategies that lead to the least travel delay under some constraints on the fares. Similar to [6], we formulate the second-best problem as a mathematical program with equilibrium constraint and propose a cutting constraint algorithm for its solution.

Throughout, we assume that transit users are rational, in the sense that they adopt travel strategies that minimize their individual travel time. In this respect, Section 2 summarizes passenger behaviour upon which equilibrium conditions are based. Movements and travel strategies yielding user equilibrium and system optimal flows are described in Section 3. In Section 4, we formulate the second-best transit problem as a mathematical program with equilibrium constraint

and propose a cutting constraint algorithm for its solution.

2 Passenger movements and travel strategies

Let $G = (N, A)$ be a transit network with node set N, arc set A and group-dependent demand $D^g_{(q,r)}$. We assume that there is a schedule for every transit line that lists the daily scheduled departure times. Let T denote the operating interval of a transit system and c_{ij} the travel time for arc (i, j). We adopt a time-expanded (TE) representation of the network of the form $G(V, E)$, where $V = \{i_t \,|\, i \in N, 0 \leq t \leq T\}$ and E is the union of the following sets:

- in-vehicle arc set $\{(i_t, j_{(t+c_{ij})}) \,|\, (i,j) \in A, t \in \Delta_{ij}\}$, where Δ_{ij} is the set of the departure times from node i of route segment (i, j).

- access arc set $\{(q_t, j_{(t+c_{qj})}) \,|\, (q,j) \in A, 0 \leq t \leq T - c_{qj}\}$.

- egress arc set $\{(i_t, r_{(t+c_{ir})}) \,|\, (i,r) \in A, 0 \leq t \leq T - c_{ir}\}$.

- waiting arc set $\{(i_t, i_{(t+1)}) \,|\, i \in N, 0 \leq t \leq T - 1\}$.

Transit fares and capacities on all in-vehicle arcs given departure time t, are denoted as v^t_{ij} and u^t_{ij}, respectively. We assume that passengers behave strategically. A strategy s assigns, at each node $i_t \in V$, a user-preference set $E^s_{i_t}$ of TE nodes, that may consist of transit, walking, or wait arcs. The user is then assigned to its preferred available choice. Let $S_{(q,r)}$ denote the set of strategies for OD pair (q, r). We denote the set of all feasible strategic assignments (SA):

$$\mathcal{X} = \{X : \sum\nolimits_{s \in S_{(q,r)}} x^s_{(q,r,g)} = D^g_{(q,r)}, \forall (q,r,g)\}.$$

For a given $X \in \mathcal{X}$, $\pi^s_{(i_t, j_{(t+c_{ij})})}(X)$ is the probability that a passenger using strategy s access arc $(i_t, j_{(t+c_{ij})})$. The procedure for computing these access probabilities involves loading the TE network according to a given SA vector X (for more details, see [3]). These access probabilities induce node arrival probabilities, $\alpha^s_{i_t}(X)$, of accessing node i_t using strategy s. The expected cost of a strategy s can be expressed in terms of arc and node probabilities as follows:

$$
\begin{aligned}
C^s_{(q,r,g)}(X) = \quad & \sum\nolimits_{(i_t, j_{(t+c_{ij})}) \in E} (p^g_{(i_t, j_{(t+c_{ij})})}(X) + \gamma c_{ij} + v^t_{ij}) \alpha^s_{i_t}(X) \pi^s_{(i_t, j_{(t+c_{ij})})}(X) \\
& + \sum\nolimits_{(i_t, i_{(t+1)}) \in E} \gamma \alpha^s_{i_t}(X) \pi^s_{(i_t, i_{(t+1)})}(X),
\end{aligned}
$$

where γ is a factor converting time into monetary units and $p^g_{(i_t, j_{(t+c_{ij})})}(\cdot)$ is the penalty function that measures the discomfort level for in-vehicle arcs and that accounts for lost opportunities associated with early departure and arrivals outside the desired arrival interval.

3 User equilibrium and system optimum

This section presents the formulation of the user equilibrium and system optimum problems for schedule-based transit networks with travel strategies and capacity constraints. More details about solution algorithms are given in [3]. A strategic assignment vector X^U is in a user equilibrium if and only if X^U solves the following variational inequality:

$$\text{U-OPT:} \quad C(X^U)^T(X - X^U) \geq 0, \quad \forall X \in \mathcal{X},$$

where $C(X)$ is the vector of expected strategy costs associated with X.

The system optimum problem can be formulated as follows:

$$\text{S-OPT:} \quad X^S = \arg\min_X \{C(X)^T X : X \in \mathcal{X}\}.$$

4 Second best congestion pricing

The second best congestion pricing problem can be expressed as the mathematical program with equilibrium constraint:

$$
\begin{aligned}
\text{SBCP-VI:} \quad & \min_{X,\beta} \quad C(X)^T X \\
& \text{s.t.} \quad X \in \mathcal{X}, \\
& \qquad \beta_{ij}^t \geq 0, \qquad\qquad \forall (i_t, j_{(t+c_{ij})}) \notin F \\
& \qquad \beta_{ij}^t = 0, \qquad\qquad \forall (i_t, j_{(t+c_{ij})}) \in F \\
& \qquad C(X,\beta)^T(Y - X) \geq 0, \quad \forall Y \in \mathcal{X},
\end{aligned}
$$

where β_{ij}^t are time-varying fare adjustments, F is the set of TE arcs for which transit fares v_{ij}^t cannot be adjusted, and $C(X,\beta)$ is the vector of expected strategy costs that accounts for fare adjustments added to each in-vehicle travel cost. If (X^*, β^*) denotes an optimal solution to SBCP-VI, then the following holds:

$$C(X^U)^T X^U \geq C(X^*)^T X^* \geq C(X^S)^T X^S$$

Using the extreme point representation of the bounded convex polyhedron $\mathcal{X}$, SBCP-VI can be reformulated as:

$$
\begin{aligned}
\text{SBCP-EX:} \quad & \min_{X,\beta} \quad C(X)^T X \\
& \text{s.t.} \quad X \in \mathcal{X}, \\
& \qquad \beta_{ij}^t \geq 0, \qquad\qquad \forall (i_t, j_{(t+c_{ij})}) \notin F \\
& \qquad \beta_{ij}^t = 0, \qquad\qquad \forall (i_t, j_{(t+c_{ij})}) \in F \\
& \qquad C(X,\beta)^T(Y^k - X) \geq 0, \quad \forall k = 1, 2, \cdots, n
\end{aligned}
$$

We propose for a solution of the above the following cutting constraint algorithm:

Step 0: Let X^1 be the optimal solution of S-OPT. Set $n = 1$ and go to Step 1.

Step 1: Solve the following master problem

$$
\begin{aligned}
(X^n, \beta^n) = \arg\min_{X,\beta} \quad & C(X)^T X \\
\text{s.t.} \quad & X \in \mathcal{X}, \\
& \beta_{ij}^t \geq 0, && \forall (i_t, j_{(t+c_{ij})}) \notin F \\
& \beta_{ij}^t = 0, && \forall (i_t, j_{(t+c_{ij})}) \in F \\
& C(X, \beta)^T (Y^k - X) \geq 0, \quad \forall k = 1, 2, \cdots, n
\end{aligned}
$$

Step 2: Solve the subproblem $Y^{n+1} = \arg\min\{C(X^n, \beta^n)^T X\}$. If $C(X^n, \beta^n)^T (Y^{n+1} - X^n) \geq 0$ stop and (X^n, β^n) is a solution to SBCP-EX. Otherwise, set $n = n + 1$ and go to Step 1.

While the above scheme provides a sequence that converges to the optimal solution of the problem within a finite number of iterations, it relies on the exact solution of nonconvex subproblems. A key contribution of this work is to show that, based on the structure of the problem, it is possible to solve the subproblems to near-optimality. We will then discuss the extension of the model to the situation where both private and transit modes are considered.

References

[1] R. Arnott and K. Small, "The Economics of Traffic Congestion", *American Scientist* 82, 446-455 (1994).

[2] L. Brotcone, L. Labbé, P. Marcotte and G. Savard, "A Bilevel Model for Toll Optimization on a Multicommodity Transportation Network", *Transportation Science* 35(4), 345-358 (2001).

[3] Y. Hamdouch and S. Lawphongpanich, " Congestion Pricing for Schedule-Based Transit Networks", *Transportation Science* To appear 2009.

[4] D.W. Hearn and M.V. Ramana, "Solving Congestion Toll Pricing Models", in *Equilibrium and Advanced Transportation Modeling*, P. Marcotte and S. Nguyen (eds), 109-124, Kluwer Academic Publishers, Norwell, Massachusetts, 1998.

[5] M. Labbé, P. Marcotte and G. Savard, "A Bilevel Model of Taxation and Its Application to Optimal Highway Pricing", *Management Science* 44(12), 1608-1622 (1998).

[6] S. Lawphongpanich and D.W. Hearn, "An MPEC Approach to Second-best Toll Pricing", *Mathematical Programming Part B* 101, 33-55 (2004).

Evaluation of an Auction Protocol in on-demand mobility services based on a response dynamics and heterogenous

Yusuke Hara

Department of Urban Engineering

The University of Tokyo

Eiji Hato

Department of Urban Engineering

The University of Tokyo, Tokyo, Japan

Email: hato@bin.t.u-tokyo.ac.jp

1   Introduction

Bicycle sharing system has rapidly become a popular tool in developed countries. The basic premise of the bike sharing concept is sustainable transportation. We design the mobility on-demand services such as bicycle sharing, these problems occur: 1) Although total supply exceeds total demand, you can't use or return the mobility due to spatial uneven distribution of demand or supply, 2) You can't use the mobility when you want because total demand exceeds supply. The former problem has been proposed to resolve the spatial imbalance by encouraging a move to a major port from another small-demand port with price incentives. However, if excess demand occurs, the incentive can't solve it.

Another approach is road appointment system or tradable bottleneck permits (TBP) system (Akamatsu, 2007). It is difficult for road authorities to know users' desired arraival and willingness to pay due to the asymmetry of information between road authorities and users. In this system, tradable permits auction makes the optimal price without being presented users' preferences directly. In addition, there is a big advantage not to cause congestion because TBP system does not issue permits more than road capacity. In on-demand mobility, we can solve the capacity problem by incorporating TBP system.
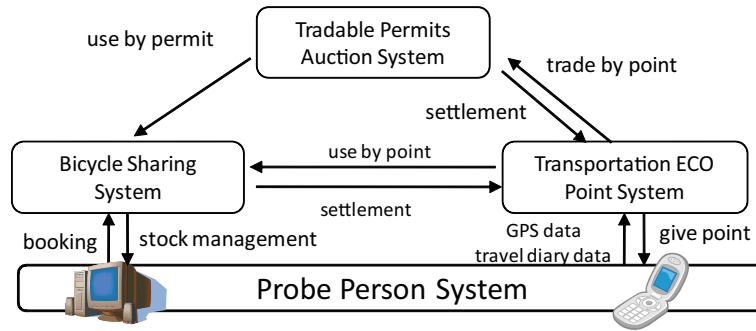
Figure 1: Framework of This Pilot Program System

But, We point out the following two problems of TBP system. One is the problem whether we can implement the complex trading system techinically in daily life. In this regard, we implemented the system in this research and we conducted the pilot program in real urban space. Second problem is that we assume that people in theory do rational decision-making but that real people confront schedule uncertainty. In auction theory, Each user has his or her preference in full recognition.Actually people cannot recognize their preference due to uncertainty of future schedule. On this point, we conducted the above pilot program and observed microscopic trading behavior. And we try to clarify the users' cognitive structure under uncertainty.

2 Pilot program framework and data

In empirical analysis, we implemented Probe Person System, Bicycle sharing system and Tradable permits auction system. They are mutually connected. Figure 1 shows the total framework.

First, Probe Person system (Hato and Kitamura, 2008) is a method to get travel diary data and positioning data in detail by GPS mobile phone. Users operate the mobile phone when they depart and arrive. Application of the mobile phone records the data of trip OD, travel mode, trip purpose, time of departure, time of mode change, time of arrival, and location data during trip. Using Probe Person survey, we can know more travel behavior data than paper-based survey. The example of PP system is Figure 2.

Second, Bicycle Sharing Tradable Permits System is an operation system of a capacity-limited sharing service. The system needs use reservation until the previous day and respondents can reserve bicycle sharing by the application of the mobile phone or the web site. There are two time slot of bicycle sharing and one is morning (9:30 - 13:00) the other is afternoon (13:30 - 19:00). To reserve it, respondents need use permit and can participate the following auction to get use permit. Each permit is set the time when you can use bicycle sharing.

Next, we will describe the tradable permits auction system. There are two systems. One is
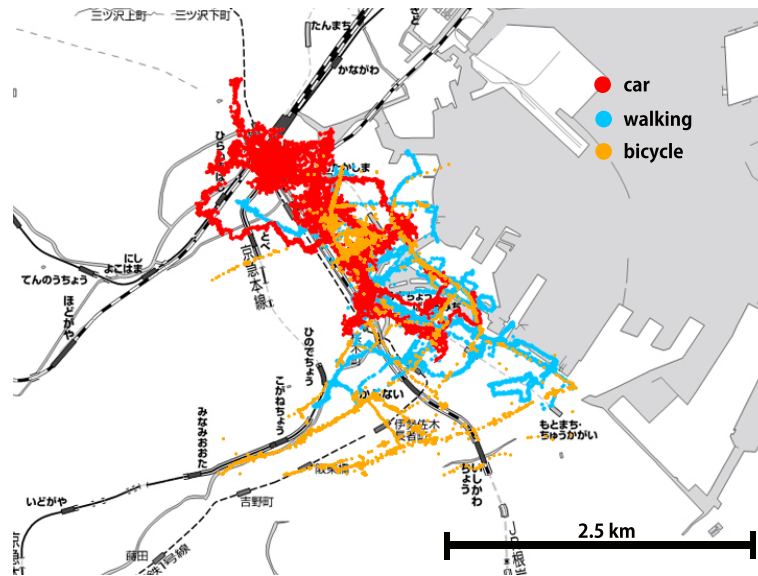
Figure 2: Probe Person Data in Yokohama Downtown Area

single auction. In single, Seller is the administrator of bicycle sharing system only and buyers are users. Users have a bidding choice when they want. As auction rule is set to second price auction, the highest bidder wins the auction and he or she pay the second price. In this way, permits are distributed according to each user's willingness to pay. Single auction is the mechanism achieving an efficient allocation of resources but it doesn't achieve fairness. So, some researchers suggest double auction mechanism as alternative. In double auction, all permits are allocated to all users at random for achieving fairness. Then, if users don't intend to use permits, they can offer them for sale. It is a phase for achieving efficient allocation. This pilot program implemented both single and double auction. And this research scopes the double auction mechanism.

The data used were obtained in 2008 at Yokohama Metropolitan Area, Japan by using the above system. Period is from 10_{th} Nov to 24_{th} Dec and it is 44 days. The activity and trip are recorded by mobile phone with GPS and Web diary. Respondents who did both probe person survey and web diary survey are 118. Only 19 people of the respondents participated in pilot program of bicycle sharing service and we analyzed the small sample data in this paper.

3   Results and Discussion

The number of total tradable permits is 108. But 3 permits are used, 15 are for sale and the remaining 90 permits are done nothing. If she or he doesn't set out to use it or can't use it, they can offer it for sale and should do for efficient allocation. However, many people didn't. In general, expected utility of selling seems to be larger than expected utility of doing nothing. In reality,
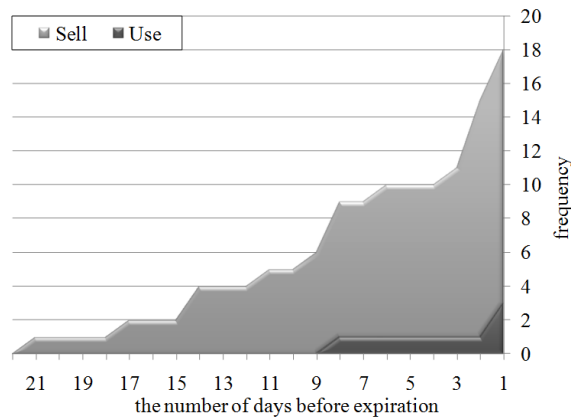
Figure 3: The Lag of Decision-Making

irrational behavior is chosen, that is, they spoiled permits. The reason is thought that respondents didn't decide the choice because of their schedule uncertainty.

Figure 3 shows When respondents decide to use or sell their permits. Use reservation is done until 8 days ago at the soonest and many case are done at the previous day. The choice of selling denotes the same tendency of "use". There is a possibility that schedule uncertainty enables respondents not to decide about permits until the eve of expiration. And this suggests schedule uncertainty as a possible cause of a low fluidity of permits in this auction. Therefore we model trade behavior in bicycle sharing permits auction and we clarify the elements of the irrational behavior.

4 Conclusions

This paper has taken a close look at the evidence on the influence of spatial relevance and uncertainty of schedule in mobility permits auction. We implemented this pilot program and collected the data of trade behavior and travel behavior. What we are modeling from the data is trade behavior considering uncertainty.

References

[1] T. Akamatsu, "A System of Tradable Bottleneck Permits for Transportation Networks", *JSTE Journal of Infrastructure Planning and Management* 63, 287-301 (2007).

[2] E. Hato and R. Kitamura, "Data-oriented travel behavior analysis based on probe person systems", *TRB conference proceedings 42 -Innovations in Travel Demand Modeling: Summary of a Conference-* 2, 187-196 (2008).

The Min-Toll-Booth Problem: Complexity, Algorithms and Experimental Findings

Tobias Harks, Martin Sieg

Institut für Mathematik, Technische Universität Berlin, Germany

Guido Schäfer

Center for Mathematics and Computer Science (CWI), The Netherlands

It is a well known fact that selfish behavior results in outcomes that are inefficient in general. A prime example is the rush-hour phenomenon observed in urban road traffic. Since every traffic participant solely aims at minimizing her individual travel time, the overall outcome is less efficient, e.g., in terms of the total average travel time, as if everybody would have been routed according to a centrally coordinated routing scheme. With the increasing number of traffic participants, the regulation of traffic becomes an increasingly important issue. One of the most promising means to regulate traffic is to impose tolls on roads. The basic idea is to impose tolls such the selfish behavior of the traffic participants leads to an outcome that corresponds to a predetermined routing scheme, e.g., one that minimizes the total average travel time. In this paper, we consider the problem of computing such tolls that additionally optimize a toll-dependent objective function.

A common way to model the selfish behavior of traffic participants is by means of a *(non-atomic) network routing game*: We are given a directed network $G = (V, A)$, k commodities $(s_1, t_1), \ldots, (s_k, t_k) \in V \times V$, and a *demand* $r_i > 0$ for every commodity $i \in [k]$ which specifies the amount of flow that has to be routed from the origin s_i to the destination t_i. Let $\mathcal{P}_i$ be the set of all (simple) directed s_i, t_i-paths in G and define $\mathcal{P} = \cup_{i \in [k]} \mathcal{P}_i$. It is convenient to express a *flow* as a function $f : \mathcal{P} \to \mathbb{R}_{\geq 0}$ that assigns to every path $P \in \mathcal{P}$ a flow-value f_P that is routed along P. A flow f is *feasible* if for every commodity $i \in [k]$ a total of r_i units of flow are routed from s_i to t_i, i.e., for every $i \in [k]$, $\sum_{P \in \mathcal{P}_i} f_P = r_i$. We define the flow on an arc $a \in A$ as $f_a = \sum_{P \ni a} f_P$. Every arc $a \in A$ has a *latency function* $\ell_a : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ associated with it. For each $a \in A$ the latency function ℓ_a is assumed to be *standard* (cf. [13]), i.e., ℓ_a is non-negative, non-decreasing and differentiable and $x\ell_a(x)$ is a convex function of x. The latency $\ell_P(f)$ of a path P with respect to a flow f is defined as the sum of the latencies of the arcs in the path, i.e., $\ell_P(f) = \sum_{a \in A} \ell_a(f_a)$. The total *cost* of a flow f is defined as $C(f) = \sum_{P \in \mathcal{P}} f_P \ell_P(f)$ or, equivalently, $C(f) = \sum_{a \in A} f_a \ell_a(f_a)$.

1

A feasible flow of minimum total cost is called *optimal* and denoted by f^*. A feasible flow f is a *Nash flow* iff

$$\forall i \in [k], \ \forall P \in \mathcal{P}_i, \ f_P > 0, \ \forall P' \in \mathcal{P}_i: \quad \ell_P(f) \leq \ell_{P'}(f).$$

That is, for every commodity the latency of every path that carries some positive amount of flow is minimum; in particular, this implies that all s_i, t_i-paths to which f assigns a positive amount of flow have equal latency. Under the assumption that all latency functions are standard, the cost of a Nash flow is unique (see, e.g., [14]). The *price of anarchy* is defined as the worst-case ratio (over all instances) of the cost of a Nash flow and the cost of an optimal flow, i.e., $C(f)/C(f^*)$. It is well-known (see [14]) that the price of anarchy is unbounded for general standard latency functions.

An efficient means to reduce the price of anarchy in network routing games is by imposing tolls on arcs. Basically, every player that traverses arc $a \in A$ incurs in addition to the experienced latency $\ell_a(f_a)$ a non-negative toll. We represent the tolls by a vector $\tau = (\tau_a)_{a \in A}$, where $\tau_a \in \mathbb{R}_{\geq 0}$ specifies the toll that is imposed on arc $a \in A$. We assume that players are *heterogeneous*. That is, we are given a parameter $\gamma_i \in \mathbb{R}_{>0}$ for every commodity $i \in [k]$ and the total cost of a path $P \in \mathcal{P}_i$ with respect to a flow f is defined as $\ell_P(f) + \gamma_i \tau(P)$, where $\tau(P) := \sum_{a \in P} \tau_a$. The parameter γ_i specifies how the players of commodity i value latency relative to cost. We say that players are *homogeneous* if $\gamma_i = 1$ for all $i \in [k]$.

A question that arises is whether we can efficiently compute tolls $\tau = (\tau_a)_{a \in A}$ such that a predetermined feasible flow f can be realized as Nash flow, i.e.,

$$\forall i \in [k], \ \forall P \in \mathcal{P}_i, \ f_P > 0, \ \forall P' \in \mathcal{P}_i: \quad \ell_P(f) + \gamma_i \tau(P) \leq \ell_{P'}(f) + \gamma_i \tau(P').$$

We call such tolls *f-inducing*. The problem of computing tolls that induce an optimal flow f^* is of particular interest and we call such tolls *opt-inducing*. It is well known (see, e.g., [15]) that opt-inducing tolls are guaranteed to exist for homogeneous players: Define the *marginal cost tolls* as $\tau_a := f_a^* \cdot \ell_a'(f_a^*)$ for every arc $a \in A$. Then f^* is an optimal flow if and only if f^* is a Nash flow with respect to $\ell + \tau$. Although marginal cost tolls assure that opt-inducing tolls always exist, there might be a wide variety of such tolls.

Our Contributions. In this paper, we are interested in computing f-inducing tolls such that an additional (toll-dependent) objective function $z(\tau)$ is minimized (or maximized). There are several natural objective functions that one may want to consider. Here we mainly concentrate on the following fundamental *min-toll-booth problem)*: Given some weights $(w_a)_{a \in A}$ on the arcs, the goal is to compute f-inducing tolls that minimize the sum of the weights of the arcs with positive tolls, i.e., $z(\tau) := \sum_{a \in A: \tau_a > 0} w_a$. The min-toll-booth problem thus models situations where the imposition of a toll on an arc $a \in A$ incurs a certain cost w_a (e.g., operational costs to collect the

tolls). In the unit-weight case, the problem reduces to computing f-inducing tolls such that the number of arcs that are subject to charges is minimized. Our main contributions are threefold:

(1) We prove that a special case of the min-toll-booth problem for single-commodity instances is polynomial time equivalent to the *minimum length bounded cut problem* (see [1]). This result enables us to prove that the min-toll-booth problem is NP-hard and APX-hard, even for very restricted single-commodity instances. While constant approximation algorithms may still be obtainable in the single-commodity case, we rule out their existence for the multi-commodity min-toll-booth problem. Via a reduction from the directed multicut problem, we show that the min-toll-booth problem cannot be approximated within a factor of $2^{\Omega(\log^{1-\varepsilon} n)}$ for every $\epsilon > 0$.

(2) In light of the above hardness results, we derive an approximation algorithm for the min-toll-booth problem. For single-commodity instances we prove that its approximation factor is bounded by the difference between the latencies of the longest flow-carrying path and the shortest path. For general instances the algorithm achieves an (instance-dependent) approximation factor that depends on the largest toll in an optimal solution (which might be difficult to quantify). However, this is the first approximation algorithm for the min-toll-booth problem.

(3) We present experimental findings on real-world instances for the min-toll-booth problem (and some other fundamental network toll problems). The experiments show that our approximation algorithm performs much better in practice than its worst-case approximation guarantee suggests. For most of the test instances our algorithm computes solutions whose cost is at most a factor 4 worse than that of an optimal solution.

Related Work. Pigou [12] already suggested in 1920 that in order to obtain a system optimal traffic pattern vehicles should be charged taxes equal to the difference between marginal social and private cost (marginal cost pricing). The theoretical foundation of marginal cost pricing was further explored by many researchers; see, for example, Knight [10], Beckmann et al. [2], and Smith [15].

A large body of work in the transportation literature is devoted to the characterization of the set of feasible tolls inducing an optimal flow as equilibrium by systems of linear inequalities; see, among others, Bergendorff et al. [3], Hearn and Ramana [8], Larsson and Patriksson [11]. Hearn and Ramana [8] also proposed secondary optimization problems, where the goal is to minimize (maximize) a toll-dependent objective function over the set of feasible tolls. In particular, they were the first to study the min-toll-booth problem. Dial [5, 6] proposed efficient algorithms for finding tolls that minimize the total revenue.

Recent studies addressed the setting of heterogeneous players, where players may have different trade-offs for delay versus toll. In this setting, one can exploit linear-programming duality to obtain tolls that induce an optimal flow; see, e.g., Cole et al. [4], Fleischer et al. [7], Karakostas et al. [9], and Swamy [16].

References

[1] G. Baier, T. Erlebach, A. Hall, E. Köhler, H. Schilling, and M. Skutella. Length-bounded cuts and flows. In *Proc. of the 33rd Int. Colloquium on Automata, Languages and Programming*, pages 679–690, 2006.

[2] M. Beckmann, C. B. McGuire, and C. B. Winsten. *Studies in the Economics and Transportation.* Yale University Press, 1956.

[3] P. Bergendorff, D. Hearn, and M. Ramana. Congestion Toll Pricing of Traffic Networks. In P. M. Pardalos, D. W. Hearn, and W. W. Hager, editors, *Network Optimization*, volume 450 of *Lecture Notes in Economics and Mathematical Systems*, pages 51–71, 1997.

[4] R. Cole, Y. Dodis, and T. Roughgarden. Pricing network edges for heterogeneous selfish users. In *Proc. of the 35th Annual ACM Symp. on Theory of Computing*, pages 521–530, 2003.

[5] R. B. Dial. Minimal-revenue congestion pricing part I: A fast algorithm for the single-origin case. *Transportation Research Part B: Methodological*, 33(3):189–202, 1999.

[6] R. B. Dial. Minimal-revenue congestion pricing part II: An efficient algorithm for the general case. *Transportation Research Part B: Methodological*, 34(8):645–665, 2000.

[7] L. Fleischer, K. Jain, and M. Mahdian. Tolls for heterogeneous selfish users in multicommodity networks and generalized congestion games. In *Proc. of the 45th Annual IEEE Symp. on Foundations of Computer Science*, pages 277–285, 2004.

[8] D. Hearn and M. Ramana. Solving congestion toll pricing models. In P. Marcotte and S. Nguyen, editors, *Equilibrium and Advanced Transportation Modeling*, pages 109–124. Kluwer Academic Publishers, 1998.

[9] G. Karakostas and S. G. Kolliopoulos. Edge pricing of multicommodity networks for heterogeneous selfish users. In *Proc. of the 45th Annual IEEE Symp. on Foundations of Computer Science*, pages 268–276, 2004.

[10] F. H. Knight. Some fallacies in the interpretation of social cost. *The Quarterly Journal of Economics*, 38(4):582–606, 1924.

[11] T. Larsson and M. Patriksson. Side constrained traffic equilibrium models: Analysis, computation and applications. *Transportation Research*, 33(B):233–264, 1999.

[12] A. C. Pigou. *The Economics of Welfare.* Macmillan, 1920.

[13] T. Roughgarden. *Selfish Routing.* PhD thesis, Cornell University, 2002.

[14] T. Roughgarden and É. Tardos. How bad is selfish routing? *Journal of the ACM*, 49(2):236–259, 2002.

[15] M. J. Smith. The marginal cost taxation of a transportation network. *Transportation Research Part B: Methodological*, 13(3):237–242, 1979.

[16] C. Swamy. The effectiveness of Stackelberg strategies and tolls for network congestion games. In *Proc. of the 18th ACM-SIAM Symp. on Discrete Algorithms*, pages 1133–1142, 2007.

Threshold Model of Social Contagion Process on Random Networks: Application to Evacuation Decision Making

Samiul Hasan

Ph.D. Student, School of Civil Engineering

Purdue University

Email: samiul@purdue.edu

Satish Ukkusuri

Associate Professor, School of Civil Engineering

Purdue University, West Lafayette, IN 47907, USA

Email: sukkusur@ecn.purdue.edu

1 Introduction

Over the last decade, the study of complex networks spanning from the Internet to social networks has grown enormously [1, 2]. Understanding the coupled dynamics between the structural properties and the functions of complex networks has been the principal focus of a wide area of research. A fundamental question relates to how the interactions between the nodes or vertices of the network may cause a small movement in the network to propagate throughout the whole network. Such phenomenon called as cascades or contagion can occur, for example, in the transmission of infectious diseases through communities [3, 4], global spread of computer viruses on the web [5, 6], diffusion of activities, beliefs, ideas, and emotions in social networks [7], failures in electrical systems [8] and the collapse of financial systems [10].

Such contagious behavior can also become a vital component to investigate transportation demand analysis under specific situations. Since transportation systems, have a significant coupling between the dynamic demand (manifestation of human behavior) and the supply, small changes in the behavior can have significant impact on the transportation network. For example, the information cascade or social contagion process in the social network can be applicable to the modeling of the complex process of information propagation within the social network in hurricane

evacuations. However, this influence of social interactions among the population is yet to be fully understood for modeling the complex evacuation process during an emergency. Under an emergency situation, individuals in a population usually exhibit herd-like behavior as their decisions are based on the actions of other individuals rather than their own perception of the information about the problem. Whether to evacuate or not; when to evacuate, where and which route to take are some important dimensions of the decisions involved in the process which are influenced primarily by the *information* and by *peers in the social network*.

In this paper, we develop a network science model to investigate the social contagion process within a network where individuals adopt alternate behaviors by following their peers. Specifically, we investigate the threshold model of social contagion on random networks with a particular mixing patterns. The threshold model follows a simple binary decision rule such as an individual agent observes the current states (either 0 or 1 i.e either evacuated or not-evacuated) of k other agents which we call its neighbors, and adopts state 1 if at least a threshold fraction of its k neighbors are in state 1, else it adopts state 0. We test the threshold model of individual decision making process assuming a distribution of threshold value for individuals. We also investigate the effects of mixing patterns of the social network to the propagation or diffusion among the population. As such this paper envisions to bring together concepts from complex networks and transportation to develop an integrated contagion propagation model. Initially, our analytical model will derive strict conditions for the information cascade to diffuse efficiently. Our simulation model will test the findings of the analytical results.

The threshold model of social contagion on random networks was first proposed and studied by Watts [14]. Gleeson [12] introduced a method to determine the mean avalanche size of modular random networks. Dodds and Payne [11] investigated the social contagion process on degree correlated networks. However, the mixing in their networks is based on vertex degrees not based on any socio-demographic characteristics such as income, ethnicity or the race of individuals in the social network. It is found from the literature that the modularity or community structure or assortative mixing in networks are not well represented in studies related to threshold model of contagion process in social networks.

We hypothesize that social mixing patterns play an important role for contagion process in social network and become an important part of the decision making process during an evacuation period. The principal objective of this paper is to find the effects of mixing patterns on the threshold model of social contagion process on random networks.

2 Analytical Model

In this section we try to determine the condition at which a social cascade or contagion will occur within a hypothetical network having mixing pattern based on some discrete characteristics; this type of network is called assortatively mixed network. The analytical derivation of the cascade condition draws from previous work of Newman [13] and Watts [14].

Let us consider a mixing matrix e_{ij}, degree distribution of vertices $p_k^{(i)}$ of type $i = 1, 2, ..., n$ and a distribution of threshold values of $P_{threshold}(\phi)$

Following the definitions provided in Newman [13], let a vertex of type i has degree k. These k edges are divided into n types with some partition $\{r_1, r_2, ..., r_j\}$ where $\sum_{j=1}^{n} r_j = k$

Now, the probability that a particular partition $\{r_j\}$ takes a particular value is given by following equation of multinomial probability

$$P^i(k, \{r_j\}) = k! \prod_j \frac{1}{r_j!} \left[\frac{e_{ij}}{\sum_j e_{ij}} \right]^{r_j} \tag{1}$$

The concept of generating function is used for the distribution of the number of edges for each type,[13, 15]

$G_0^{(i)}(x_1, x_2, ..., x_n) = \sum_{k=0}^{\infty} p_k^{(i)} \sum_{\{r_j\}} \delta(k, \sum_j r_j) P^i(k, \{r_j\}) x_1^{r_1} x_2^{r_2} ... x_n^{r_n}$

$= \sum_{k=0}^{\infty} p_k^{(i)} \left[\frac{\sum_j e_{ij} x_j}{\sum_j e_{ij}} \right]^k$

$= G_0^{(i)} \left(\frac{\sum_j e_{ij} x_j}{\sum_j e_{ij}} \right)$

Where, δ is the Kronecker delta function and $G_0^{(i)}(x) = \sum_k p_k^{(i)} x^k$ is the generating function for the degree distribution $p_k^{(i)}$.

Now, one important feature to find is the distribution of the degree of vertex of type i following a randomly chosen edge,

$$G_1^{(i)}(x) = \frac{\sum_k k p_k^{(i)} x^{k-1}}{\sum_k k p_k^{(i)}} = \frac{1}{z_i} G_0^{(i)'}(x) \tag{2}$$

Where $z_i \equiv G_0^{(i)'}(1)$ is the mean degree of type i vertices.

For the distribution of the edges of different types,

$$G_1^{(i)}(x_1, x_2, ..., x_n) = G_1^{(i)} \left(\frac{\sum_j e_{ij} x_j}{\sum_j e_{ij}} \right) \tag{3}$$

Now if we consider a threshold model where a vertex with degree k is vulnerable with probability, $\rho_k = P[\phi \leq \frac{1}{k}]$ then from Watts [14], probability of a vertex having degree k and being vulnerable is $\rho_k p_k$ and then the generating function for vulnerable vertex degree will be $F_0(x) = \sum_k \rho_k p_k x^k$

Similarly, for our case, the generating function for vulnerable vertex degree of type i will be

$$F_0^{(i)}(x) = \sum_k \rho_k^{(i)} p_k^{(i)} x^k \tag{4}$$

Then the fraction of vulnerable vertices of type i will be $P_v^{(i)} = F_0^{(i)}(1) = \sum_k \rho_k^{(i)} p_k^{(i)}$. And the average degree of a vulnerable vertex of type i is, $z_v^{(i)} = F_0^{(i)'}(1) = \sum_k \rho_k^{(i)} k p_k^{(i)}$.

But the average degree of any vertex of type i will be, $z^{(i)} = G_0^{(i)'}(1) = \sum_k k p_k^{(i)}$

Hence, the generating function for the distribution of the number of vulnerable edges for each type is $F_0^{(i)}(x_1, x_2, ..., x_n) = F_0^{(i)} \left(\frac{\sum_j e_{ij} x_j}{\sum_j e_{ij}} \right)$

Similar to the previous formula for randomly chosen edge, we can also find the generating function for the distributions of the vulnerable vertex degrees of type i following a randomly chosen edge. $F_1^{(i)}(x) = \frac{1}{z^{(i)}} F_0^{(i)'}(x)$. For the distribution of the edges of different types, $F_1^{(i)}(x_1, x_2, ..., x_n) = F_1^{(i)} \left(\frac{\sum_j e_{ij} x_j}{\sum_j e_{ij}} \right)$

An important distribution to observe is the distribution of the number of vulnerable vertices that can be reached by following a random edge to a vertex of type i. The generating function $H_1^{(i)}$ for such distribution satisfies the following self-consistency condition

$$H_1^{(i)}(x) = 1 - F_1^{(i)}(1) + x F_1^{(i)} \left[H_1^{(1)}(x), ..., H_1^{(n)}(x) \right] \tag{5}$$

And similarly the distribution of the number of vulnerable vertices that can be reached from a randomly chosen vertex of type i is generated by

$$H_0^{(i)}(x) = 1 - F_0^{(i)}(1) + x F_0^{(i)} \left[H_1^{(1)}(x), ..., H_1^{(n)}(x) \right] \tag{6}$$

The average number of vulnerable vertices s_i reachable from a vertex of type i is

$s_i = \frac{dH_0^{(i)}}{dx}|_{x=1} = F_0^{(i)} \left[H_1^{(1)}(1), ..., H_1^{(n)}(1) \right] + F_0^{(i)'}(1) \frac{\sum_j e_{ij} H_1^{(j)'}(1)}{\sum_j e_{ij}}$

$= P_v^{(i)} + F_0^{(i)'}(1) \frac{\sum_j e_{ij} H_1^{(j)'}(1)}{\sum_j e_{ij}}$

$= P_v^{(i)} + z_v^{(i)} \frac{\sum_j e_{ij} H_1^{(j)'}(1)}{\sum_j e_{ij}}$

This can be written in matrix format,

$$\mathbf{s} = \mathbf{P_v} + \mathbf{m_0} \mathbf{H_1'}(1) \tag{7}$$

Where $\mathbf{m_0}$ is a matrix $[m_0]_{ij} = \frac{z_v^{(i)} e_{ij}}{\sum_j e_{ij}}$

Now to find s, we will need to know $\mathbf{H_1'}(1)$

$H_1^{(i)'}(1) = F_1^{(i)} \left[H_1^{(1)}(1), ..., H_1^{(n)}(1) \right] + F_1^{(i)'} \left[H_1^{(1)}(1), ..., H_1^{(n)}(1) \right]$

$= F_1^{(i)}(1) + F_1^{(i)'}(1) \frac{\sum_j e_{ij} H_1^{(j)'}(1)}{\sum_j e_{ij}}$

In matrix format,

$$\mathbf{H_1'}(1) = \mathbf{F_1}(1) + \mathbf{m_1} \mathbf{H_1'}(1) = \mathbf{F_1}(1)[\mathbf{I} - \mathbf{m_1}]^{-1} \tag{8}$$

Where $\mathbf{m_1}$ is a matrix $[m_1]_{ij} = \frac{F_1^{(i)'}(1) e_{ij}}{\sum_j e_{ij}}$

$F_1^{(i)'}(1) = \frac{1}{z^{(i)}} F_0^{(i)''}(1) = \frac{1}{z^{(i)}} \sum_k \rho_k^{(i)} p_k^{(i)} k(k-1) = \frac{1}{z^{(i)}}(\langle k^2 \rangle - \langle k \rangle) = \frac{z_{v2}^{(i)}}{z_1^{(i)}}$

Where $z_1^{(i)} \equiv z^{(i)}$ (average degree of vertices) and $z_{v2}^{(i)}$ is the second vulnerable neighbors of a vertex of type i.

So matrix $\mathbf{m_1}$ becomes, $[m_1]_{ij} = \frac{z_{v2}^{(i)} e_{ij}}{z_1^{(i)} \sum_j e_{ij}}$

So finally, equation (7) for average component size becomes,

$$\mathbf{s} = \mathbf{P_v} + \mathbf{m_0} \left[\mathbf{I} - \mathbf{m_1}\right]^{-1} \mathbf{F_1(1)} \qquad (9)$$

A cascade will happen when s diverges and s will diverge when $det[\mathbf{I} - \mathbf{m_1}]$ will reach its first zero. This condition is similar to Newman [13]. However, it is interesting to know when this will happen for assortatively mixed networks.

$1 - \frac{z_{v2}^{(i)} \sum_j e_{ij}}{z_1^{(i)} \sum_j e_{ij}} = 0$

$\Rightarrow z_{v2}^{(i)} = z_1^{(i)}$

This means that a cascade will happen when the average first neighbors are equal to the average second vulnerable neighbors.

3 Simulation and Conclusions

In the full paper, we will develop the full set of analytical models and develop a simulation model to investigate whether our analytical findings correspond to the simulation results. Briefly, the simulation model consists of two stages: first stage builds an assortatively mixed network following [13] and second stage applies the threshold model of social contagion on the network. The steps are as following:

Stage I - Building the assortatively mixed network

1. Choose first number of edges M of the network and then draw M edges from the distribution e_{ij}

2. Count the number of ends of edges of each type i, to obtain the sum of m_i; calculate $n_i = \frac{m_i}{z_i}$ where z_i is the desired mean degree of vertices of type i

3. Draw n_i vertices from $p_k^{(i)}$ of type i making the sum of degrees of the vertices to be m_i

4. Randomly pair up the m_i ends of edges of type i with the generated vertices so that each vertex has the number of attached edges according to its chosen degree

5. Repeat step 3 and 4 for each vertex type

Stage II - Applying the threshold model

1. Each vertex is given a fixed threshold value ϕ sampled from $P_{threshold}(\phi)$.

2. Vertex states update at times $t = 0, 1, 2, ...$ At each time step a vertex can either of the two states σ_0 (Not evacuated) and σ_1 (Decided to evacuate). Each vertex observes the fraction of its neighbors in state σ_1 and switches to σ_1 if the fractions exceeds its threshold ϕ.

We will also investigate, in the full paper, the effects of the strength of each edge. In the above model individual's neighbors have equal weight while calculating the fraction of the people who have decided to evacuate. However, there could be alternative mechanisms such as individuals put higher weights for a particular group (i.e. same ethnicity) of their neighbors while other neighbors have lower weight when calculating the fraction of the people who have decided to evacuate.

This paper will contribute to the transportation literature in terms of building a modeling framework of incorporating the social influence on decision making relevant for transportation modeling. Although the social contagion models have been studied in network analysis more commonly, they have not been used in any previous transportation literature to the best of the authors' knowledge. We believe that such introduction of social influences can improve the modeling capabilities and can become an essential component for future integrated transportation models.

References

[1] M.E.J. Newman, "The Structure and Function of Complex Networks", *SIAM Review* 45(2), 167-256 (2003).

[2] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwanga, "Complex networks: Structure and Dynamics", *Physics Reports* 424, 175-308 (2006).

[3] J.D. Murray, *Mathematical Biology (Third ed.)*. New York: Springer (2002).

[4] R.M. Anderson and R.M. May, *Infectious Diseases of Humans: Dynamics and Control* Oxford: Oxford Univ. (1991)

[5] M.E.J. Newman, S. Forrest, and J. Balthrop, "Email Networks and the Spread of Computer Viruses", *Physical Review E* , 035101 (2002).

[6] J. Balthrop, S. Forrest, M.E.J. Newman, and M.M. Williamson, "Computer Science: Technological Networks and the Spread of Computer Viruses", *Science* 304, 527-529 (2004).

[7] J. Coleman, H. Menzel, and E. Katz, *Medical Innovations: A Diffusion Study*. New York: Bobbs Merrill (1966).

[8] R. Kinney, P. Crucitti, R. Albert, and V. Latora, "Modeling Cascading Failures in the North American Power Grid", *European Physics Journal B* 46, 101 (2005).

[9] M.L. Sachtjen, B.A. Carreras, and V.E. Lynch, "Disturbances in a Power Transmission System", *Physical Review E* , 4877 (2000).

[10] D. Sornette, *Why Stock Markets Crash: Critical Events in Complex Financial Systems*. Princeton, NJ: Princeton University Press (2003).

[11] P.S. Dodds,and J.L. Payne,"Analysis of a Threshold Model of Social Contagion on Degree-correlated Networks", *Physical Review E* 79, 066115 (2009).

[12] J.P. Gleeson, "Cascades on Correlated and Modular Random networks", *Physical Review E* 77, 046117 (2008).

[13] M.E.J. Newman,"Mixing Patterns in Networks", *Physical Review E* 67, 026126 (2003).

[14] D.J. Watts,"A Simple Model of Global Cascades on Random Networks", *Proc. Natl. Acad. Sci.* 99, 5766-5771 (2002).

[15] H.S. Wilf, *Generatingfunctionology*, 2nd ed. Academic Press, London (1994).

Using Heterogeneous Computing for Solving Vehicle Routing Problems

Geir Hasle

Department of Applied Mathematics, SINTEF ICT

P.O. Box 124 Blindern, NO-0314 Oslo, Norway. Email: Geir.Hasle@sintef.no

Oddvar Kloster

Department of Applied Mathematics, SINTEF ICT

Atle Riise

Department of Applied Mathematics, SINTEF ICT

Christian Schulz

Center of Mathematics for Applications, University of Oslo

Morten Smedsrud

Department of Applied Mathematics, SINTEF ICT

1 Introduction

For many applications of optimized transportation management, there is still a large gap between requirements and the performance of decision support systems of today. Vehicle routing is no exception [8]. Although there have been a tremendous increase in our ability to solve ever more complex VRPs (partly due to methodological improvements, partly due to the general increase in computing power), the ability to consistently provide better routing plans in shorter time across a variety of instances will give substantial additional savings. With VRP methods that are more powerful and robust, applications that are too large or too complex for routing tools today will become effectively solvable.

For solving a variety of industrial VRPs, some form of approximate solution method is required in a generic vehicle routing tool [9]. Metaheuristics constitute a popular basis for solving rich and large-size VRPs [3,4,7]. Variants and hybrids of Large Neighborhood Search, Variable Neighborhood Search, and Iterated Local Search methods have proven remarkable performance lately [5,13]. Heuristics based on exact methods such as column generation, and hybrid methods, for instance

combining exact methods and local search, have recently proven to be highly effective in solving complex VRPs.

Parallel computing is one way of improving both performance and robustness of VRP methods. Parallelism comes in many guises, ranging from low level instruction parallelism to coarse-grained cooperative parallel solvers. We distinguish between task parallelism and data parallelism (or stream processing). Parallel methods in discrete optimization are not new [1,15]. According to the recent survey of Crainic [7] however, the literature on parallel methods for the VRP is scarce before year 2000. The survey has 80 references that almost exclusively focus on task parallelization for the VRP.

The optimization methods that are utilized by vehicle routing tools of today are tailored to yesterday's computer architectures and sequential processing. Standard (commodity) PCs today have 2-8 cores that support task parallelism. The number of cores is expected to increase rapidly. Moreover, the performance and programmability of special processors, such as General Purpose Graphics Processing Units (GPGPUs) for data parallel stream processing, is improving very rapidly. Today's GPUs greatly outpace CPUs in arithmetic throughput and memory bandwidth.

Heterogeneous computing aims at combining the task parallelism of traditional multi-core CPUs and accelerator cores to deliver unprecedented levels of performance [2,6]. It has shown impressive results in scientific computing, e.g. in simulation and visualization. Sequential algorithms cannot benefit from the rapid, parallelism based performance increase of modern heterogeneous PC architectures. The new architectures call for a rethinking of optimization algorithms.

2 Solving VRPs with Heterogeneous Computing

The literature on heterogeneous computing in discrete optimization is scarce. We know of no paper that investigates the use of heterogeneous computing for the VRP, but there is ongoing work on GPU processing for local search [12].

We investigate the potential of heterogeneous computing in solving VRPs. The targeted heterogeneous computing platform is a modern multi-core PC with a GPU for stream processing. Our starting point is the well known and much studied Distance Constrained VRP (DVRP), but our aim is to extend the work to a rich VRP model. The solution method is inspired from earlier work on industrial, rich VRPs by a subset of the authors [10], as well as ideas from Irnich et al. [11].

The solution method we investigate is a local search based metaheuristic. In broad terms, it is a hybrid between Iterated Local Search, Variable Neighborhood Search, and Large Neighborhood Search. We use a giant tour representation of solutions, and utilize the general resource concept and resource extension functions to handle constraints.

On each iteration, a Variable Neighborhood Descent with restricted k-opt moves in the giant tour representation, $k=2,...,5$ is executed until a local optimum is reached. Candidate lists are utilized to reduce neighborhood size. A phase of Adaptive Large Neighborhood Search with various destructors

and constructors [13] is then executed until a timeout with no improvement. A new iteration is then started, and the whole process is continued until a given time limit.

Stream processing on the GPU is utilized for data parallel neighborhood evaluation. A general kernel for delta value calculation of the objective and feasibility check of a neighbor executes on the available GPU cores. Small sized neighborhoods (size depending on the specific type of GPU) are executed in true parallel. For larger neighborhoods, the GPU evaluates the neighborhood in several chunks.

Our method will later be extended with task parallelism targeted at multiple heterogeneous computers. Task parallelism will first be targeted at rather embarrassingly parallel tasks of Large Neighborhood Search [14], Variable Neighborhood Search, and Iterated Local Search. Later, we shall investigate heterogeneous computing for cooperative solvers.

In the talk, we briefly explain modern PC architectures and the general principles of heterogeneous computing. We illustrate how multi-core and GPU computing may be utilized for higher performance and more robust VRP solvers, and explain the details of our solution method for the DVRP. We present the results of computational experiments on standard CVRP/DVRP benchmarks from the literature as well as industrial test instances from newspaper distribution. Perspectives and directions for future work are given.

References

[1] Alba E. (editor): *Parallel Metaheuristics – A New Class of Algorithms*. Wiley 2005. ISBN-13 978-0-471-67806-9.

[2] Asanovic K. et al.: "The Landscape of Parallel Computing Research: A View from Berkeley". *Tech. Report UCB/EECS-2006-183*. University of California at Berkeley, December 2006.

[3] Bräysy O., M. Gendreau, G. Hasle, and A. Løkketangen: "A Survey of Heuristics for the Vehicle Routing Problem, Part I: Basic Problems and Supply Side Extensions". *SINTEF Report*, Oslo, Norway, 2008.

[4] Bräysy O., M. Gendreau, G. Hasle, and A. Løkketangen: "A Survey of Heuristics for the Vehicle Routing Problem, Part II: Demand Side Extensions". *SINTEF Report*, Oslo, Norway, 2008.

[5] Bräysy O., W. Dullaert, P. Porkka, G. Hasle: "Heuristic Strategies for Solving Large-Scale Vehicle Routing Problems". *Route2009 - International workshop on vehicle routing, intermodal transport and related areas*, ROLIGHED, Skodsborg, Denmark, Jun 20-24, 2009.

[6] Brodtkorb A. R., C. Dyken, T. R. Hagen, J. M. Hjelmervik and O. O. Storaasli: "State of the Art in Heterogeneous Computing". *Working paper (submitted)*, SINTEF, May 2009.

[7] Crainic T.G.: "Parallel Solution Methods for Solving Vehicle Routing Problems". In Golden B., S. Raghavan, E. Wasil (eds): *The Vehicle Routing Problem – Latest Advances and New Challenges*. Springer 2008. ISBN 978-0-387-77778-8.

[8] Gendreau M., J.-Y. Potvin, O. Bräysy, G. Hasle, A. Løkketangen: "Metaheuristics for the Vehicle Routing Problem and extensions: A Categorized Bibliography". Chapter in *The Vehicle Routing Problem: Latest Advances and New Challenges*, edited by B. Golden, S. Raghavan, and E. Wasil, Springer, 2008.

[9] Hall R.: "On the road to integration". *ORMS Today* 33(3):50-57, 2006.

[10] Hasle G., O. Kloster: "Industrial Vehicle Routing Problems". Chapter (pp 397-432) in Hasle G., K-A Lie, E. Quak (eds): *Geometric Modelling, Numerical Simulation, and Optimization – Applied Mathematics at SINTEF*. ISBN 978-3-540-68782-5, Springer 2007.

[11] Irnich S.: "A Unified Modeling and Solution Framework for Vehicle Routing and Local Search-Based Metaheuristics". *INFORMS Journal on Computing 20*(2): 270-287 (2008)

[12] Luong T.V., N. Melab, E.-G. Talbi: "Parallel Local Search on GPU". *Report 6915*, INRIA 2009.

[13] Pisinger D., S. Ropke: "A general heuristic for vehicle routing problems". *Computers & Operations Research*, Vol. 34, Issue 8 (2007), 2403-2435.

[14] Røpke S.: "Parallel large neighborhood search - a software framework". In *MIC 2009. The VIII Metaheuristics International Conference*, Forthcoming.

[15] Talbi E-G (ed): *Parallel Combinatorial Optimization*. Wiley 2006. ISBN-13 978-0-471-72101-7.

Models and Algorithms for Bin Allocation and Vehicle Routing in Waste Collection Applications

Vera Hemmelmayr, Karl Doerner, Richard Hartl

Department of Business Administration

University of Vienna, Brünnerstr. 72, Wien, Austria

Email: vera.hemmelmayr@univie.ac.at

Daniele Vigo

Dipartimento di Elettronica, Informatica e Sistemistica,

Università di Bologna

1 Introduction

We study a real world problem occurring in the waste collection industry, which we will call the Waste Bin Allocation And Routing Problem (WBARP). In this problem we have to balance the trade-off between the service frequency of a given waste-collection site over a planning period, and the number of bins that can be placed there. More precisely, if a site has a higher service frequency, the routing cost will increase, since we have to visit this site more often, but at the same time the allocation cost is less, because we use a smaller number of bins. Bins used may be of different types, each characterized by different capacity and cost. Moreover, due to possible space limitations at each collection site there is a limit on the total number of bins it can accommodate, as well as a limit on the total number of bins of each type that can be used.

So our problem consists of two aspects. On the one hand there is the routing, on the other hand there is the part of deciding how many bins to place at each site given the restrictions on space and on the total number of bins to be used.

2 Problem Formulation

Concerning the routing part, a similar problem has already been studied in [2] and [3], where an extension of the classical Periodic Vehicle Routing Problem (PVRP), called PVRP with Service Choice (PVRP-SC) is considered. The classical PVRP extends the Vehicle Routing Problem to a

m day planning horizon. A set of customers requires regular visits during the planning horizon. The timing of the visits is not given, but every customer has a certain visit frequency e_i that must be respected. Moreover, for every customer a set C_i of allowable *visit combinations* (i.e., a set of days in which the service must occur) is given. Hence the PVRP consists of choosing for each customer a visit combination and defining a set of routes for each day of the planning horizon with overall minimum cost. In the PVRP-SC the visit frequency is not given, but chosen during the search. Moreover, a cost is defined to measure the benefit of a service frequency. In our problem such a benefit is based on the bin allocation.

As previously mentioned, the second aspect of the problem is the allocation of the bins. This problem may be defined as follows. We are given a set V of n *collection sites*, each characterized by the volume of waste produced per day, q_i, and a maximum capacity for that site, U_i, $i = 1, \ldots, n$.

We are also given a set of m *bin types*, each characterized by a volume capacity Q_j, the amount of space required at a site e_j , a purchase cost C_j^P and by a maximum total number M_j that may be used, $j = 1, \ldots, m$. Note that the purchase cost may also include the maintenance costs, and that all these costs are relative to a single service time horizon. For the the sake of simplicity we only consider a single type of waste.

Service at collection sites may be performed according to different possible *service profiles*. In particular a service profile specifies in which relative days of the service period the bins are served (i.e., emptied). For our purposes, we are given k service profiles, each associated with a maximum number of time intervals between two consecutive visits, a_h, $h = 1, \ldots, k$. Note that to appropriately define the total capacity required at a given collection site the only important information about the service profile is the maximum number of days between two consecutive visits.

We also consider the fact that often sites are grouped into *zones* to which should be assigned the same service profile. To this end, given a set of s zones let us define the site-zone association by means of a matrix of binary coefficients $\pi_{i\ell}$ which take value 1 iff site i ($i = 1, \ldots, n$) belongs to zone ℓ ($\ell = 1, \ldots, s$), and 0 otherwise. Let $f_{\ell h}$ be a set of constants indicating whether service profile h is chosen for zone ℓ.

We may also take into account the current bin allocation and therefore want to optimize the cost of modifying the current configuration. Some additional problem input data are required. First of all, for each site i ($i = 1, \ldots, n$) let p_{ij} be the current number of bins of type j present in the site, $j = 1, \ldots, m$. Note that if we do not want to take the current allocation into account, all p_{ij} are simply set to 0.

For each bin type j ($j = 1, \ldots, m$) in addition to the already defined *purchase* cost, C_j^P, we now introduce specific costs for the *transfer* of a bin from the depot to a site, C_j^T, and for the *removal* from the site to the depot, C_j^R. Note, that the transfer of a bin from one site to another consists

of a removal from one site and a movement to the other one. In practice transfers are performed by collecting all containers, then bringing them to the depot and finally distributing all containers to their destination.

The model makes use of the following decision variables. The integer variable x_{ij} is the number of bins of type $j = 1, \ldots, m$, allocated to site $i = 1, \ldots, n$. Since we need to distinguish between the bins that are actually purchased and those that are simply moved between sites, we introduce new set of continuous variables. In particular, for each pair (i, j) of site and bin type we let z_{ij}^+ and z_{ij}^- denote the number of bins of type j that are added or removed form site i, respectively. Moreover, for each bin type j $(j = 1, \ldots, m)$ let w_j denote the total number of bin of that type to be purchased. The model is then:

$$\min \quad \sum_{j=1}^m C_j^P w_j + \sum_{i=1}^n \sum_{j=1}^m C_j^R z_{ij}^- + \sum_{i=1}^n \sum_{j=1}^m C_j^T z_{ij}^+$$

subject to

$$\sum_{j=1}^m Q_j x_{ij} - q_i \sum_{h=1}^k a_h \sum_{\ell=1}^s \pi_{i\ell} f_{\ell h} \geq 0, \quad \forall\, i = 1, \ldots, n, \tag{1}$$

$$\sum_{j=1}^m x_{ij} e_j \leq U_i \qquad\qquad \forall\, i = 1, \ldots, n, \tag{2}$$

$$w_j \leq M_j \qquad\qquad \forall\, j = 1, \ldots, m, \tag{3}$$

$$z_{ij}^+ \geq x_{ij} - p_{ij} \qquad\qquad \forall\, i = 1, \ldots, n; j = 1, \ldots, m, \tag{4}$$

$$z_{ij}^- \geq p_{ij} - x_{ij} \qquad\qquad \forall\, i = 1, \ldots, n; j = 1, \ldots, m, \tag{5}$$

$$x_{ij} - z_{ij}^+ + z_{ij}^- = p_{ij} \qquad\qquad \forall\, i = 1, \ldots, n; j = 1, \ldots, m, \tag{6}$$

$$w_j \geq \sum_{i=1}^n (z_{ij}^+ - z_{ij}^-) \qquad\qquad \forall\, j = 1, \ldots, m, \tag{7}$$

$$x_{ij} \geq 0 \text{ and integer}, \qquad\qquad \forall\, i = 1, \ldots, n; j = 1, \ldots, m, \tag{8}$$

$$z_{ij}^+, z_{ij}^- \geq 0, \qquad\qquad \forall\, i = 1, \ldots, n; j = 1, \ldots, m, \tag{9}$$

$$w_j \geq 0, \qquad\qquad \forall\, j = 1, \ldots, m, \tag{10}$$

The objective function minimizes purchase, movement and transfer cost. Constraints (1) impose the capacity requirements by requiring that the total capacity of the bins at a given site is not smaller than the maximum amounts of waste produced at the site between two consecutive visits of the selected service profile. Moreover, constraints (2) limit the total number of bins that may be associated with a site, and constraints (3) limit the total number of bins that may be used for each bin type. Constraints (4), (5) and (6) control the number of bins that are added and the number of bins that are removed. Constraints (7) state that the number of bins to be purchased for every

bin type is the difference between the number of bins that are added and those that are removed.

Finally, constraints (8), (9) and (10) define the type of the decision variables.

3   Solution Approaches

We have analyzed and tested the above model for bin allocation and we found that it is solved very fast with CPLEX (within one second), when applied to instance with practically relevant size, i.e. having several hundreds of collection sites. This gives way for incorporating this model within a routing framework. In a first step we develop a method based on a VNS for the PVRP (based on that presented in [1]), that is able to solve the PVRP-SC . Then we incorporate the bin allocation in the routing framework, so as to solve the WBARP. Since the bin allocation model solves very fast, we can incorporate it in the routing part whenever new solutions are found that change the visit frequency of at least one customer. Note that for the bin allocation problem a different service profile means a service profile that has a different maximum number of days in between two consecutive visits, i.e. a different a_h. Changes between visit frequencies with the same a_h do not affect a solution to the bin allocation model. The proposed method is evaluated through computational testing on instances from practical applications.

Acknowledgments

References

[1] V.C. Hemmelmayr, K.F. Doerner and R.F. Hartl," A Variable Neighborhood Search Heuristic for Periodic Routing Problems", *European Journal of Operational Research* 159, 791- 802 (2009).

[2] P. Francis, K. Smilowitz and M. Tzur, "The period vehicle routing problem with service choice", *Transportation Science* 40, 439–454 (2006).

[3] P. Francis, K. Smilowitz and M. Tzur, "Flexibility and complexity in periodic distribution problems", *Naval Research Logistics* 54, 136–150 (2007).

Nested Column Generation for a Ship Routing and Scheduling Problem with Split Pickup and Split Delivery

Frank Hennig and **Bjørn Nygreen**

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology

Email: {frank.hennig, bjorn.nygreen}@iot.ntnu.no

Marco E. Lübbecke

Fachbereich Mathematik, Technische Universität Darmstadt, Germany

Email: luebbecke@mathematik.tu-darmstadt.de

1   Introduction

The splitting of a customer's demand between several vehicles of a given fleet seems natural in the attempt to carry out efficient transportation. Today operations research (OR) accounts for this mainly through study of the split delivery vehicle routing problem (SDVRP). The real problem studied here can be characterized as a split pickup split delivery problem with time windows and capacity constraints and will be referred to as SPSDP(TW) throughout this paper. The pickup and delivery problem (PDP) has been extensively studied in many variants, see for example reviews of [1, 2]. A commented review is provided by [3]. Variants, that treat split of quantities and many-to-many relations for pickup and delivery nodes are scarce. [2] for example names only one study, in which pickup and delivery points are non-paired for the multi-vehicle case. Pickup and delivery problems with split for both pickup and delivery are not mentioned in the review at all. The only PDP with multiple vehicles and allowed split in both pickup and delivery nodes known to the authors is [6].

In this study we transfer ideas used by [4] and [5] for the SDVRP to the SPSDPTW. We develop a branch-and-price algorithm whose master problem is similar to the master problem in [4]. However, our subproblem is considerably more complex and therefore we suggest using column generation for the subproblem itself.

This extended abstract is organized in the following way: Section 2 gives a simplified descrip-

tion of the problem and Section 3 gives an overview of our solution approach. A comment about the results is given at the end in Section 4.

2   Problem description

Our basic model resembles the following situation: Main entities are a set of heterogeneous ships $\mathcal{V}$, a set of pickup and delivery time windows $\mathcal{N}$ and a set of products, $\mathcal{C}$. Ships operate on a directed Graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$ where $\mathcal{N}$ is the set of time windows and $\mathcal{A}$ the set of arcs. Each ship has an individual initial location and may be sent to an individual final location. Each time window $i \in \mathcal{N}$ is associated with a pickup or delivery port, a quantity Q_i of a particular product $c \in \mathcal{C}$, and a time interval $[\underline{T}_i, \overline{T}_i]$ in which service has to start. Pickup and delivery quantities and may exceed ship capacity.

There can be several choices of pickup time windows to supply a given delivery time window and vice versa. Pickup and delivery ports accumulate in certain pickup or delivery regions. Large distances between pickup and delivery regions lead to routes having a *voyage* structure. On a voyage a ship may visit one or several pickup regions followed by one or several delivery regions. It is unrealistic that a ship carries cargo from a delivery region back to any pickup region. If voyage length allows, a ship may undertake several voyages during the planning period. The number of time windows a ship services in a region is limited by practical considerations. Therefore the maximum number of different grades simultaneously onboard is limited, too. Since pickup and delivery times may amount to several days for large tanker ships, the quantities picked up and delivered can influence the time feasibility of a route. For time feasibility it is sufficient that a ship arrives at time window i not later than closing time $\overline{T}_i$. A maximum waiting time between arrival at time window i and time window opening $\underline{T}_i$ may be desired. Sailing times T^S_{ijv} on arcs $(i,j) \in \mathcal{A}$ are assumed to be deterministic. Two types of capacity, weight and volume, have to be taken into account. Capacity changes during operation due to external factors like for example limited water depth and port regulations. Hence, pickup and delivery quantities can be influenced by the order in which time windows are visited. A ship is allowed to serve any fraction of a time window quantity.

3   Solution approach

Our research is targeted at solving the stated problem by an exact column generation method. Preliminary tests with pre-generated columns showed that master problems with continuous quantity variables for pickup and delivery solved much slower than master problems where discrete pickup and delivery quantities had been already imbedded in the columns.

The latter approach has already been studied for the SDVRP by [4] and [5], who successfully exploit the special structure of their subproblems. Facing pickup and delivery, multiple commodities, arc dependent capacities and quantity dependent service times, we had to resort to an extended solution approach, namely *nested column generation* as [7] uses the term. In nested column generation the subproblem is solved by column generation itself.

We define an all-ship column generation master problem and call it *RMP 1*, i.e. restricted master problem on level 1. The RMP 1 has columns representing a subset of possible routes with specific pickup and delivery quantities, called *cargo patterns*, for all ships . A feasible solution of the RMP 1 is a convex combination of cargo patterns for a particular route for each used ship. Some of the columns in this problem are generated on a second level as solutions of single-ship master problems, RMP 2, for each ship. The rest of the columns are found by heuristically changing pickup and delivery quantities for known time window sequences found in the RMP 2. In essence a RMP 2 is a route selection and cargo pattern generation problem. It has variables that represent some of the possible routes for the given ship. But instead of having quantity information in the route columns, we have separate continuous variables to determine pickup and delivery quantities. The level 2 subproblem is a rather standard elementary shortest path problem with time windows (ESPPTW). During solving of a particular RMP 2 a dynamic program provides routes without cargo quantity information that are RMP 2 columns with non-positive reduced cost.

The whole algorithm is implemented in C++ using the non-commercial branch-cut-and-price framework SCIP (scip.zib.de) developed by [8]. For the dynamic program we use the resource constrained shortest path algorithm provided by the boost C++ graph libraries (www.boost.org). To facilitate the full branching capability of SCIP we choose to branch on single variables. Therefore we have defined binary variables for both a ship visiting a time window, and a ship using an arc on both master problem levels. Each of these variables need a matrix row to calculate their values dependent on the use of the route columns.

4 Results and Conclusion

We present results for the described method and compare these with column pre-generation results. The results are based on twelve instances based on realistic data. Column pre-generation for the model with continuous pickup and delivery variables turned out to be a not promising approach, but showed that better solutions can be obtained in comparison to the approach with discrete split. The latter performed well for a reasonable discretization. However, for the largest instances it can be a challenge to execute pre-generation in reasonable time. With the branch-and-price algorithm we were able to improve results since it performs significantly better than the first pre-generation method and provides a larger feasible space than the model with discrete split. Running time

improvements will be discussed.

For the SDVRP, the largest instances optimally solvable today comprise about 100 customers (Desaulniers, 2008). Allowing a split of loads in both, the pickup and delivery time windows implies a complication of this already tremendously difficult optimization problem. The problem gets even more complex with arc dependent capacity constraints and quantity dependent time window service times. As a computational consequence this implies a further combinatorial explosion of solution possibilities, so it is not surprising that the resulting type of SPSDP is even harder to solve. It is precisely this explosion (in finding optimal combinations of split loads) which consumes most of our reported computation times. Thus, being able to provide high quality solutions to instances with almost 50 pickup and delivery locations (as we do) is a respectable achievement already

References

[1] Parragh, S.N. and Doerner, K.F. and Hartl, R.F., "A survey on pickup and delivery problems - Part I: Transportation between customers and depot", *Journal für Betriebswirtschaft* 58(1), 21-51 (2008).

[2] Parragh, S.N. and Doerner, K.F. and Hartl, R.F., "A survey on pickup and delivery problems - Part II: Transportation between pickup and delivery locations", *Journal für Betriebswirtschaft* 58(2), 81-117 (2008).

[3] Berbeglia, G. and Cordeau, J. and Gribkovskaia, I. and Laporte, G., "Static pickup and delivery problems: a classification scheme and survey", *TOP* 15, 1-31 (2007).

[4] Desaulniers, G., "Branch-and-Price-and-Cut for the Split Delivery Vehicle Routing Problem with Time Windows", Technical Report G-2008-32, Les Chaiers du GERAD, Montreal, 2008. Fortcoming in *Operations Research*, doi:10.1287/opre.1090.0713

[5] Ceselli, A. and Righini, G. and Salani, M., "Column Generation for the Split Delivery Vehicle Routing Problem", Technical report, University of Milan - DTI - Note del Polo n. 118, 2009.

[6] McKay, M. D. and Hartley, H. O., "Computerized Scheduling of Seagoing Tankers", *Naval Research Logistics* 21(2), 255-264 (1974).

[7] Vanderbeck, F., "A Nested Decomposition Approach to a Three-Stage, Two-Dimensional Cutting-Stock Problem", *Management Science* 47(6), 864-879 (2001).

[8] Achterberg, T., "SCIP: solving constraint integer programs", *Mathematical Programming Computation* 1, 1-41 (2009).

An exact method to solve the Multi-Trip Vehicle Routing Problem with Time Windows and Limited Duration

Florent Hernandez[1,2,*]   **Dominique Feillet**[3]   **Rodolphe Giroudeau**[2]
Olivier Naud[1]

[1]: Cemagref UMR ITAP, 361 rue JF Breton, Montpellier, France

[2]: LIRMM UMR 5506, 161 rue Ada 34392, Montpellier France

[3]: Ecole des Mines de Saint-Etienne, CMP Georges Charpak, F-13541 Gardanne

[*]: corresponding author: florent.hernandez@montpellier.cemagref.fr, tel.: +334 67 04 63 70

1 Problem description

The Multi-Trip Vehicle Routing Problem with Time Windows (MTVRPTW) is a variant of the classic Vehicle Routing Problem with Time Windows (VRPTW) where vehicles can be scheduled more than one trip within a workday or planning time horizon. In this study, we consider a special case of the MTVRPTW, called MTVRPTW-LD, where duration of trips (routes) are limited.

The MTVRPTW-LD is defined as follows. Let $G = (V, A)$ be a directed graph where $V = \{v_0, \cdots, v_n\}$ with v_0 representing the depot and $v_1, \cdots, v_n$ the customers, a cost c_{ij} for all arcs $(v_i, v_j) \in A$, a fleet of U vehicles with a load capacity Q, a planning time horizon $[0, T]$, a duration limit t_{max} and for each $v_i \in \{v_1, \cdots, v_n\}$ a demand d_i, a time window $[a_i, b_i]$ with $a_i, b_i \in [0, T]$ and a service time st_i. The problem is to find a minimum cost set of trips visiting each customer only once with respect to capacity and time constraints, and such that two routes cannot be travelled at the same time by the same vehicle. *Trip duration* is the elapsed time between the depot departure time, after the vehicle has been loaded, and the arrival time to the last customer of the trip, before the delivery. The schedule of a vehicle must also include, in the *complete trip duration*, the loading time, the service time to last customer and return time to depot.

This problem was addressed in 2007 and 2009 by N. Azi et al. They designed an exact method for the single-vehicle case [1] first, then for the multi-vehicle case [2]. Based on these investigations and on previous works [4], we developed a new exact method, allowing large improvements in terms of computing times.

2 A new exact method for the MTVRPTW-LD

Let us define a "trip structure" s, simply denominated hereafter "structure", as an ordered list of customers than can be visited during a trip while satisfying their time constraints. Let d_s be the minimal complete trip duration needed to visit these customers and come back to depot, in this order. For every structure s, a time window $[a_s, b_s]$ can be calculated such that a_s (b_s, respectively) is the earliest departure time (latest arrival time, respectively) permitting to visit the customers of s with a duration d_s. A trip is now defined as a structure with fixed time position.

The proposed algorithm is composed of two phases: enumeration and column generation.

2.1 Enumeration phase

As long as the duration limit is relatively short, it is possible to generate all the non-dominated structures [2]. In order to do this, we adapted the dynamic programming algorithm described in [3], mainly by modifying resources and dominance rules.

We defined the labels as follows: a path p from the origin v_0 to node v_j is labeled with $L_p = \{c_p, v_j, h_p, q_p, rd_p, a_p, b_p, W_p^{v_1}, \cdots, W_p^{v_n}\}$, where c_p is the reduced cost of this partial path, h_p and q_p are the values of time and load resources, respectively, accumulated along this path; rd_p is the minimal trip duration of the partial path represented by L_p ; a_p and b_p are the start and end of the label time window as defined above; and $W_p^{v_i} = 1$ if node v_i is visited by L_p, 0 otherwise.

As for dominance, we use the following relation: if p and p' are two differents paths from origin v_0 to node v_j with labels L_p and $L_{p'}$, respectively, then p dominates p' if and only if the nodes visited by p and by p' are the same ($W_p^{v_i} = W_{p'}^{v_i}$ for every customer v_i), the time window of L_p includes the time window of $L_{p'}$ ($a_p \leq a_{p'}$ and $b_p \geq b_{p'}$), and $c_p \leq c_{p'}$, $h_p \leq h_{p'}$, $q_p \leq q_{p'}$, $rd_p \leq rd_{p'}$.

2.2 Column generation phase

The second phase of the algorithm is based on column generation and branch and price. We propose a set covering formulation where columns (variables) represent trips. The planned itinerary for a vehicle consists in a set of successive loadings and trips. Two trips of a given vehicle cannot overlap, so we introduce time intervals $d_t = [l_{min} * t, l_{min} * (t + 1)[$ where l_{min} is a small value guaranteeing that the duration of any trip will be at least l_{min}, and $t \in \{0, \cdots, \lfloor \frac{T}{l_{min}} \rfloor\}$. The set of columns is denoted Ω. Column generation is based on the solution of restricted master problems, each corresponding to a subset Ω_1 of columns as follows:

$$z(\Omega_1) = minimize \sum_{r_k \in \Omega_1} c_k \theta_k \tag{1}$$

subject to

$$\sum_{r_k \in \Omega_1} a_{ik}\theta_k \geq 1 \qquad (v_i \in V \setminus \{v_0\}), \qquad (2)$$

$$\sum_{r_k \in \Omega_1} b_{tk}\theta_k \leq U \qquad (\forall d_t), \qquad (3)$$

$$\theta_k \geq 0 \qquad (r_k \in \Omega_1), \qquad (4)$$

where c_k is the cost of trip r_k, a_{ik} indicates whether customer v_i is visited by route r_k or not, $b_{tk} \in [0 , 1]$ is the fraction of the time interval d_t occupied by trip r_k and θ_k are decision variables. Constraints (2) enforce that every customer is visited at least once, constraints (3) enforce that at most U vehicles are used during any time interval.

The subproblem consists in finding trips $r_k \in \Omega \setminus \Omega_1$ with a negative reduced cost $c_k - \sum_{v_i \in V \setminus \{v_0\}} a_{ik}\lambda_i + \sum_{d_t} b_{tk}\mu_t$, where λ_i and μ_t are dual variables respectively corresponding to primal constraints (2) and primal constraints (3).

Every non-dominated structure has been previously enumerated. For every structure s, new trips (columns) are generated by selecting a time position in the time window $[a_s, b_s]$, such that the reduced cost of the trip is negative.

This column generation scheme is integrated in a branch and price tree. A lower bound of solutions is computed at each node of the search tree. We use a classical branching rule on arcs. Note that solutions with fractional variables but integer flows on arcs might exist. It can be shown that such solutions can be transformed into integer solutions.

This algorithm essentially differs from [2] in the definition of the set covering model. In their approach, Azi *et al.* rather define columns as complete itineraries that a vehicle could carry out, which are generated by performing elementary shortest path search in a graph where nodes are what we call here structures.

3   Results

We ran our method, yet without dominance relation, on the Solomon's instances with 25 customers and large time horizon in same conditions as [2]. All tests were performed with a fleet size of two vehicles and data resolution of 0.01. The loading time for customer v_i is equal to 20% of its service time, the travel time is the same as the Euclidean distance between two customer locations and $t_{max} = 220$ for C2 class instances and $t_{max} = 75$ for R2 and RC2 class instances. Some of our preliminary results are presented in Table 1. We close 24 of these 27 instances within an allowed computation time of 30h. Compared to [2], we obtain much lower computation times. As for these, it is worth to note that, as in the classical VRPTW, computation times of instances vary greatly within the same class. Please note that in table 1, solutions with lower costs have been found.

Instance	Solution	Solution [2]	Total Time (sec)	Total Time (sec) [2]	Iter	Column
c201-25	659.02	659.02	0.734	160.16	62	134
c202-25	653.37	653.37	25.25	5095.33	684	558
c203-25	*646.4*	-	146.312	-	1520	791
c204-25	*602.46*	604.51	144.39	45004.70	941	1333
c205-25	636.39	636.39	55.64	349.72	3407	318
c206-25	*636.39*	636.97	683.89	6511.43	33413	498
c207-25	603.22	603.22	62.281	1732.88	1722	588
c208-25	613.20	613.20	68.046	2832.06	2293	489

Table 1: Branch and Price result on C2 Solomon instances with 25 customers and a duration limit of 220

4 Conclusion

In this paper, we adressed the exact solution of the MTVRPTW-LD, previously introduced and investigated in Azi et al. [2]. We proposed an efficient branch and price scheme, achieving a fast improvement compared to [2] with regard to computing times. A main advantage of our approach lies in the efficiency of the column generation subproblem, solved with a fast pseudo-polynomial algorithm.

References

[1] N. Azi, M. Gendreau, and J. Y. Potvin. An exact algorithm for a single-vehicle routing problem with time windows and multiple routes. *European Journal of Operational Research*, 178(3):755–766, 2007.

[2] N Azi, M Gendreau, and J. Y. Potvin. An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles. *European Journal of Operational Research*, 202(3):756–763, May 2010.

[3] D. Feillet, P. Dejax, M. Gendreau, and C. Gueguen. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks*, 44(3):216–229, 2004.

[4] F. Hernandez, D. Feillet, R. Giroudeau, and O. Naud. Multi-trip vehicle routing problem with time windows for agricultural tasks, 2009. Odysseus, Fourth International Workshop on Freight Transportation and Logistics, Izmir, Turkey.

Using Branch-and-Price to Find High-Quality Solutions Quickly *

Mike Hewitt

Department of Industrial and Systems Engineering

Rochester Institute of Technology

Email: mrheie@rit.edu

George Nemhauser, Martin Savelsbergh

Department of Industrial and Systems Engineering

Georgia Institute of Technology

When integer programming (IP) models are used in operational situations there is a need to consider the tradeoff between the conflicting goals of solution quality and solution time, since for many problems solving realistic-size instances to a tight tolerance is still beyond the capability of state-of-the-art solvers. However, by appropriately defining small instances, good primal solutions frequently can be found quickly. This approach is taken, for example, within linear programming-based branch-and-bound algorithms using techniques such as local branching ([5]) and RINS ([3]). These techniques use information from the linear program (LP) solution and incumbent solution to define a small IP, which is then optimized. These techniques can be applied to any integer program and are available in commercial solvers such as CPLEX. Another approach to defining small instances is to use problem structure as in [6] where small IPs are chosen according to the attributes of previous solutions. Combining exact and heuristic search techniques by solving small IPs has received quite a lot of attention recently, see, for example, [4], [9], [2], and [8]. Still another heuristic approach is to use structure to define neighborhoods that can be searched in polynomial time such as the very large scale neighborhood search approach of [1]. A key difference between the methods that are embedded in an LP-based tree search algorithm (local branching and RINS) and the others is that they are connected with a dual bounding procedure so that optimality or weaker tolerance gaps can be proved.

In this research we introduce a new approach to finding good solutions quickly that is capable

of proving optimality as well. It is different from techniques such as local branching and RINS since it uses problem structure to define the small IPs to be solved. It is different from other IP-based local search methods since the IPs to be solved are determined by a column generation scheme. The embedding of this column generation scheme into a branch-and-price algorithm gives the dual bounds that provide the capability of proving optimality.

Our extended formulation, which requires column generation, is very different from typical column generation formulations that employ structurally different objects from the compact formulation, for example, paths rather than arcs. Our extended formulation keeps the original variables from the compact formulation and augments them with an exponential number of additional variables that are used to define problem restrictions to obtain small IPs. By preserving the original compact formulation, we are able to enrich it by preprocessing, cutting planes or any other techniques normally used in a branch-and-cut framework.

We apply the approach to the Multi-commodity Fixed Charge Network Flow (MCFCNF) problem, a challenging problem that lies at the heart of many transportation problems, including those faced by consolidation carriers. The MCFCNF is a classic optimization problem in which a set of commodities has to be routed through a directed network. Each commodity has an origin, a destination, and a quantity. Each network arc has a capacity. There is a fixed cost associated with using an arc and a variable cost that depends on the quantity routed along the arc. For a consolidation carrier, the fixed cost represents the transportation cost of a container and there may be no variable cost. The objective is to minimize the total cost.

We first discuss the approach as applied to a general mixed integer program P given by:

$$
\begin{array}{llll}
\max & cx & + & dy \\
\text{s.t.} & Ax & + & By & = b \\
& & & x \text{ real}, & y \text{ binary}.
\end{array} \tag{1}
$$

Let V_P^* denote the optimal value of P and $S_P = \{(x,y)|\ Ax + By = b, x \text{ real}, y \text{ binary}\}$ be the set of feasible solutions to P. For a given integer matrix N and a given integer vector q, both of appropriate dimension, we define *restriction* $P_N(q)$ of P as:

$$
\begin{array}{llll}
\max & cx & + & dy \\
\text{s.t.} & Ax & + & By & = b \\
& & & Ny & \leq q \\
& & & x \text{ real}, & y \text{ binary}
\end{array} \tag{2}
$$

with optimal value $V_N^*(q)$. We suppose that this restriction can be solved much faster than P. We define $R = \{r|\ r = Ny \text{ for some } (x,y) \in S_P\}$ to be the set of vectors associated with feasible solutions to problem P. Clearly, we have $V_P^* \geq V_N^*(r)\ \forall\ r \in R$ and $V_P^* = V_N^*(r^*)$ with $r^* = Ny_P^*$ for an optimal solution (x_P^*, y_P^*) to P. Thus, a strategy for finding an optimal solution to P is

searching over the set R and solving restrictions $P_N(q)$, which will be feasible if there is an $r \in R$ such that $q \geq r$. A major advantage of such a strategy is that it produces a feasible solution to P each time such a restriction $P_N(q)$ is solved.

Ideally, we would only solve restrictions $P_N(q)$ whose optimal value $V_N^*(q)$ is close to the optimal value V_P^*. Consequently, we would need an oracle that considers all vectors $r \in R$, but returns only those with $V_N^*(r) \approx V_P^*$. The role of this oracle is thus similar to the role of the pricing problem in column generation: consider all columns, but return only columns with positive reduced costs (for a maximization problem). Therefore, we next assume that we know the entire set R and build a model that extends the formulation of P to choosing a vector r from R and solving the resulting restriction $P_N(r)$. Specifically, we define the problem MP:

$$
\begin{array}{rllll}
\max & cx & + & dy & \\
\text{s.t.} & Ax & + & By & = b \\
& & & Ny \quad -Rz & \leq 0 \\
& & & 1z & = 1 \\
& & x \text{ real}, & y, z \text{ binary}, &
\end{array}
\tag{3}
$$

where the binary variables z in MP represent the choice of vector r for which the restriction $P_N(r)$ should be solved. Given the definition of R, MP is a valid reformulation of P. Because knowing the entire set R is neither necessary nor feasible we use a pricing problem to generate its elements dynamically and solve MP using a branch-and-price procedure. In fact, we leverage the prevalence of mutli-core processors by developing a parallelized branch-and-price procedure to solve instances of MP.

Much of the approach is generic. To apply it to a specific class of problems primarily requires defining the matrix N that defines the restriction $P_N(q)$ and the resulting extended formulation, MP. For the MCFCNF, we use a variable-fixing type restriction that removes a subset of arcs.

The computational results demonstrate that the approach achieves its goals. For the MCFCNF instances, the approach often produces a proven near-optimal solution in 15 minutes. More specifically, the primal solution found in 15 minutes is often better than the one CPLEX produces in 6 hours, and the dual bound is usually close to the one CPLEX produces in 6 hours.

Our implementation of the approach is flexible, allowing us to apply the approach to a new class of problem by implementing only the problem and the restriction to be used. We are currently studying its application to some transportation problems with a temporal component such as Inventory Routing and Periodic Vehicle Routing.

References

[1] R. Ahuja, O. Ergun, J. Orlin, and A. Punnen. A survery of very large scale neighborhood search techniques. *Discrete Applied Mathematics*, 123:75–102, 2002.

[2] C. Archetti, M. G. Speranza, and M. W. P. Savelsbergh. An optimization-based heuristic for the split delivery vehicle routing problem. *Transportation Science*, 42:22–31, 2008.

[3] E. Danna, E. Rothberg, and C. Le Pape. Exploring relaxation induced neighborhoods to improve MIP solutions. *Mathematical Programming*, 102:71–90, 2005.

[4] R. De Franceschi, M. Fischetti, and P. Toth. A new ILP-based refinement heuristic for vehicle routing problems. *Mathematical Programming B*, 105:471–499, 2006.

[5] M. Fischetti and A. Lodi. Local branching. *Mathematical Programming*, 98:23–47, 2003.

[6] M. Hewitt, G. L. Nemhauser, and M. W. Savelsbergh. Combining exact and heuristic approaches for the capacitated fixed charge network flow problem. *INFORMS J. Comput.*, To appear, 2009.

[7] N. Mladenovi and P. Hansen. Variable neighborhood search. *Computers and Operations Research*, 24:1097–1100, 1997.

[8] M. Savelsbergh and J.-H. Song. An optimization algorithm for inventory routing with continuous moves. *Computers and Operations Research*, 35:2266–2282, 2008.

[9] V. Schmid, K. F. Doerner, R. F. Hartl, M. W. P. Savelsbergh, and W. Stoecher. A hybrid solution approach for ready-mixed concrete delivery. *Transportation Science*, 43:70–85, 2008.

Modeling the dynamics of all-day activity plans

Willem Himpe

Traffic and Infrastructure Laboratory

Katholieke Universiteit Leuven

Gunnar Flötteröd, Ricardo Hurtubia, Michel Bierlaire

Transport and Mobility Laboratory (TRANSP-OR)

Ecole Polytechnique Fédérale de Lausanne (EPFL)

Email of corresponding author: gunnar.floetteroed@epfl.ch

1   Context: Estimating activities from smart phone data

The modeling effort described in this text is motivated by the problem of identifying a smart-phone user's current activity (we currently consider *work, shopping, leisure, home, education*, and *service*) from sensor data obtained by the phone. This data comprises the current position, nearby wireless devices and networks, and phone inputs of the user. We build on a previous article, where a Bayesian activity estimation framework is presented [3], which consists of two components: The behavioral prior model provides a first distributional estimate about the activity based on socioeconomic attributes and land use data. This information is then updated using a likelihood function that relates the detection of nearby Bluetooth devices of known activity partners to the according activities.

Preliminary experimental results with this system were obtained based on a likelihood function that was modeled from survey data in that the actually conducted activities of the respondent were known. In a real application, such a survey is infeasible and supplementary information from which the activities can be inferred in hindsight is needed. This problem is tackled in this work, which proposes an additional behavioral model that is evaluated at the end of each day, generates an improved estimate of the missing activities, and updates the relation between encountered Bluetooth devices and presumably conducted activities accordingly in the likelihood, which allows for improved real-time activity estimates in the following day. This text describes the specification and estimation of this additional model.

2   Formal specification of activity schedule model

As a preliminary working hypothesis, it is assumed that the switching times between all activities can be inferred unambiguously from the data gathered by the phone. This assumption is likely to be relaxed in future work. It also is assumed that some activities can be identified with high certainty based on phone data only; an example would be the *home* activity, which is very likely to be conducted at the same place every night, or the *work* activity of persons with a fixed working location and regular working hours.

The considered activity schedule consists of N activity slots. The nth activity is denoted by a_n and its duration by τ_n. If it can be inferred from the phone data only, then a_n is assigned one of the values $\{work, shopping, leisure, home, education, service\}$. Otherwise, it needs to be predicted by the model. Denote by $\mathcal{X}$ the index set of all unknown activity slots. The model can then be written in terms of the conditional distribution $P(\{a_x\}_{x \in \mathcal{X}} \mid \{a_n\}_{n \in \{1...N\} \setminus \mathcal{X}}, \{\tau_n\}_{n \in \{1...N\}})$.

In order to reduce the combinatorial complexity of the model, only a specification $P(a_x \mid \{a_n\}_{n \in \{1...N\} \setminus \{x\}}, \{\tau_n\}_{n \in \{1...N\}})$ with a single activity gap x is considered. A simulation-based evaluation of the full model (with an arbitrary number of gaps) is possible through Gibbs sampling [5]. Gibbs sampling is a generic Markov chain Monte Carlo technique that allows to draw from a multivariate distribution if all of its one dimensional conditional distributions are known, which is exactly what the single-gap model provides.

3   Model estimation

The combinatorial complexity of a model that explicitly fills in sequences of activity gaps is avoided by estimating a model only for a single gap and relying on the Gibbs sampling technique for evaluation of the full model. The single-gap model, as from now simply called the model, is built in two steps.

First, a deterministic decision tree is built from the Swiss microcensus 2005 [1], using all information available for the canton of Vaud. For this purpose, an implementation of the C4.5 algorithm [4] in the free Weka software package is deployed [6]. This data-mining method builds a maximum entropy decision tree by repeatedly partitioning the data. Every path from the root of the tree to a terminal node (leaf) constitutes an if-clause for the activity assigned to that leaf. Second, the crisp output of each leaf is replaced by the empirical distribution of activities to which the condition of this leaf applies, hence generating a true distribution of model outputs.

The generation of the tree resembles the estimation procedure of the Albatross activity scheduling simulator [2]. The major difference to that system is that Albatross captures the actual scheduling decisions of households as a sequential decision making process, whereas the proposed model concentrates on the dynamic structure of the schedule, thus being less constrained with respect to

name	possible values	description
start_day	minutes after midnight	time at which the house is left for the first time
activ_start	minutes after midnight	starting time of the activity gap
duration	minutes	duration of the activity gap
tot_*activity*	minutes	total duration of *activity* outside of the gap
employment	full time, part time, student, unemployed	daytime occupation of considered person
weekday	any weekday	considered day of the week
prev_act	an activity type	activity conducted before the gap
next_act	an activity type	activity conducted after the gap

Table 1: Attributes

the meaning of its rules in terms of a real scheduling process.

The attributes given in Table 1 currently exhibit the greatest explanatory power. (The C4.5 algorithm can cope with missing attributes, e.g., if the gap is at the very beginning or end of the day. Also, land use data is yet to be incorporated in the model.) Overall, 2 890 reported schedules are available for the canton of Vaud. Taking out every single activity once out of every schedule results in 8 508 learning data sets. The algorithm generates a tree with 57 leaves (rules). A tenfold cross-validation indicates more than 68 % of correct classifications. Table 2 shows the according confusion matrix. As mentioned before, the crisp predictions of the tree are in a second step

work	shopping	leisure	home	education	service	classified / observed
1308	57	218	127	22	0	**work**
89	568	486	81	5	6	**shopping**
162	293	2251	295	31	2	**leisure**
114	59	259	1277	4	1	**home**
13	3	47	15	391	0	**education**
32	124	133	23	3	9	**service**

Table 2: Confusion matrix

transformed into empirical distributions. The resulting rules are consistent with common sense, which is illustrated in terms of the following example:

IF activ_start_time $\leq 532\,$min (08:52) AND employment $\notin$ {student, no_official_work}

THEN Prob(gap = *work*) = 0.81

This rule states that everyone but students and unemployed people who get up early are likely to work next. Interestingly, the day of the week is not accounted for in this rule because the majority

of applicable data sets indeed indicates a *work* activity. (We speculate that people who do not need to work on weekends are likely to sleep in, which results in an extended *home* activity.)

4   Current integration of the model and further refinements

The specification and estimation of the model is completed in most parts; it only remains to add the land use data, which is available and already in use in the *online* activity tracking system. The next step is to define a consistent update logic for the likelihood function that links the detection of nearby Bluetooth devices to the activities estimated in hindsight by the *offline* model described in this work. This system will be validated with experimental data for which the true activities (and their switching times) are known. Finally, a detection mechanism for activity switches will be added, which is likely to result in a probabilistic model of switching times as well.

An interesting aspect of further work is to investigate to what extent a random utility model can be specified based on information revealed by the decision tree. In this setting, the tree-building algorithm would constitute a data-driven tool that identifies relevant explanatory variables, which then would be transformed into a utility function and re-estimated from the same data. The advantages of this approach would be interpretability of the model coefficients, greater robustness of the model (because structure is imposed a priori), and better extrapolation capabilities.

References

[1] ARE/BfS. Mobilität in der Schweiz, Ergebnisse des Mikrozensus 2005 zum Verkehrsverhalten. Technical report, Federal Office for Spatial Development and Swiss Federal Statistical Office, Bern and Neuenburg, Switzerland, 2007.

[2] Theo A. Arentze and Harry J.P. Timmermans. A learning-based transportation oriented simulation system. *Transportation Research Part B*, 38(7):613–633, 2004.

[3] R. Hurtubia, G. Flötteröd, and M. Bierlaire. Inferring the activities of smartphone users from context measurements using Bayesian inference and random utility models. In *Proceedings of the European Transport Conference*, The Netherlands, October 2009.

[4] Ron Kohavi and Ross Quinlan. Decision tree discovery. In *Handbook of Data Mining and Knowledge Discovery*, pages 267–276. University Press, 1999.

[5] S.M. Ross. *Simulation*. Elsevier, fourth edition, 2006.

[6] Ian H. Witten and Eibe Frank. *Data Mining – Practical Machine Learning Tools and Techniques*. Elsevier, 2005.

MODELLING COMMERCIAL VEHICLE EMPTY TRIPS: THEORY, NOVEL DEVELOPMENTS, AND APPLICATION

José Holguín-Veras, Ph.D., P.E.,
Professor of Civil and Environmental Engineering
Rensselaer Polytechnic Institute

Ellen Thorson,
Senior Research Fellow
University Transportation Research Center, The City College of New York

The 21st century has been characterized by the increasing role of information technology in everyday life. Modern computer systems allow for information to be transferred faster and safer through the Internet, thus making it convenient for consumers to shop online in the comfort of their homes. As this trend continues, the demand for lighter, higher-value goods increases, since they are likely to have a lower cost to the consumer on the Internet. This increase in demand, combined with population growth and other factors, is resulting in an increase in the amount of freight that has to be transported, particularly by truck. The Oregon State Department of Transportation, as well as other ports and institutions all over the United States, expect their freight volumes to more than double by the year 2040, a rate higher than the projected population growth for that state (Metro, 2001). According to the Federal Highway Administration, about 70% of small package freight is transported by trucks; resulting in a contribution by the trucking industry of 48 billion dollars to the national GDP in the year 1998, a 130% increase from 1990 (USDOT, 1999). In light of these trends, it is clear that better, more efficient forms freight transportation systems are essential.

A hindering factor to efficient ground transportation is the larger commercial vehicle traffic that increases congestion on the roads. This results in a significant increase in air and noise pollution as well as the number of dangerous traffic accidents involving trucks. A study by the Organization for Economic Co-Operation and Development revealed that in some European countries noise from truck traffic resulted in a property loss equivalent to nearly .4% of the national GDP. The study also showed that individuals (aggregated) are willing to pay up to .65% of the GDP for a significant reduction in noise (Hecht, 1997).

The increase in truck traffic, along with the accompanying increase in externalities, puts pressure on the trucking industry and on Metropolitan Planning Organizations (MPOs). The trucking industry will need to handle larger delivery volumes, facing lower revenues due to the level of competition and increasingly stringent regulations for externalities. MPOs will have to improve their planning processes in order to accommodate the ever increasing demand for goods transportation. In order to do this, the organizations need more efficient demand models than the ones currently in use. In many cases, MPOs use adaptations of passenger car models to estimate freight demand in the area. Although these simplistic approaches can sometimes provide rough estimates, they are fundamentally flawed since they do not capture the key dynamics of freight phenomena.

One recurrent misconception involves equating commercial vehicle traffic to freight transportation demand, when in fact commercial vehicle traffic is an expression of how the trucking industry organizes itself to satisfy the demand, i.e., the commodity flows. As a result, a significant number of freight demand models focus on modelling vehicle-trips.

Currently there are two major platforms for modelling freight transportation demand: vehicle-trip and commodity based models (Ogden, 1978). Vehicle-trip models focus on modelling the actual number of vehicle trips, which has some practical advantages. Among them are the relative ease and high-quality with which data can be obtained due to an increasing number of Intelligent Transportation Systems. Also, since the model focuses on vehicle trips it has no problem generating the number of empty trips between regions. However, these models have two fundamental limitations. The first one is that these models cannot be applied to multimodal transportation because the vehicle-trip is in itself the result of a mode choice and the selection process is not represented in the data (Holguín-Veras, 2000, 2002). Furthermore, since the models assume that the vehicle is the unit of demand, as opposed to the commodity being transported, the model neglects the economic characteristics of the shipment that have been found to play a significant role in the majority of choice processes in the trucking industry (e.g., Holguín-Veras, 2002).

Commodity based models, as the name points out, focus on modelling the flow of goods from one region to the other (measured in a unit of weight). Since the commodities are the unit of demand, the modeler can capture the underlying factors that determine freight movement, such as value, weight, and volume. In this platform, the loaded trips are estimated by dividing the total flow from one region to the other by an average payload from all loaded trucks. The problem with commodity-based models is that they are unable to model empty trips, which can make up about 30 to 50 percent of the total trips in a region. This occurs since the commodity flow in one direction determines the loaded trips, but does not bear a relationship to the number of the empty trips in the same direction. To resolve this, some complementary models have been developed, such as Hautzinger's (1984), Noortman and van Es' (1978) and Holguín-Veras and Thorson (2003a). The complementary models developed by Noortman and van Es and Holguín-Veras and Thorson will be described and compared in this paper.

This is not a marginal issue, all the contrary. The statistics show that about 20-25% of the truck traffic in urban areas, and 40-50% of intercity trucks are traveling empty (Holguín-Veras and Thorson, 2003). The official statistics in the United States clearly indicate the magnitude of the problem: about 57% of the miles traveled by straight trucks, and 33% of the miles traveled by semi-trailers are empty (U.S. Census Bureau, 2004). Obviously, not properly modeling such important flow—that as said cannot be proportionally added to the loaded traffic—is bound to lead to major estimation errors. The research conducted indicated that not properly modeling empty trips lead to errors on the estimation of directional traffic that are four to seven times larger than when appropriate empty trip models are used (Holguín-Veras and Thorson, 2003).

The paper reviews the modeling approaches that have been suggested to model empty trips, and proposes novel formulations that lead to improved estimation performance. The models range from simple naïve formulations to some more complex ones involving trip chains, destination choice processes, and memory components. The performance of the alternative formulations to model empty trips is assessed by applying these models to sample data sets from different countries. The formulations developed and discussed in the paper have been successfully applied in a number of different countries including Sweden, Norway, Guatemala, and Colombia. The paper starts with some background information on the subject, followed by a brief description of previous developments in the area, a description of the test cases for the model, the methodology, and finally the results and conclusions.

1.2 References

Hautzinger, H. (1984). The Prediction of Interregional Goods Vehicle Flows: Some Modeling Concepts. Ninth International Symposium on Transportation and Traffic Flow Theory, VNU Science Press.

Holguín-Veras, J. (2000). A Framework for an Integrative Freight Market Simulation. IEEE 3rd Annual Intelligent Transportation Systems Conference ITSC-2000, Dearborn Michigan, IEEE

Holguín-Veras, J. (2002). "Revealed Preference Analysis of the Commercial Vehicle Choice Process." Journal of Transportation Engineering, American Society of Civil Engineers **128**(4): 336-346.

Holguín-Veras, J. and E. Thorson (2003). "Modeling Commercial Vehicle Empty Trips with a First Order Trip Chain Model." Transportation Research Part B: Methodological **37**(2): 129-148.

Holguín-Veras, J. and E. Thorson (2003). "Practical implications of Modeling Commercial Vehicle Empty Trips." Transportation Research Record **1833**: 87-94.

Noortman, H. J. and J. van Es (1978). Traffic Model. Manuscript for the Dutch Freight Transport Model.

Ogden, K. W. (1992). Urban Goods Movement. London, England, Ashgate Publishing

U.S. Census Bureau (2004). Vehicle Inventory and Use Survey. http://www.census.gov/prod/ec02/ec02tv-us.pdf.

Duties Scheduling for Freight Trains Drivers: a case study at the French railways

Housni Djellab

SNCF - I&R-AAD, 45 rue de Londres, 75000 Paris Cedex, France
Email: housni.djellab@sncf.fr

Mehdi Lamiri

EURODECISION, 9 Rue de la Porte de Buc, 78000 Versailles Cedex, France

1 Introduction

This paper deals with the problem of duties scheduling for freight train drivers at the French state railways SNCF (Societe Nationale des Chemins de Fer Français). The duties scheduling problem consists in constructing weekly shifts of driving work in order to cover all driving tasks with a minimum cost. This problem can be defined as follows.

We are given a planned timetable for train services (i.e. the actual journeys with freight) to be performed in a certain period. Each train service is made up by a sequence of trips. Each trip (driving task) represents a portion of a train journey which must be performed by the same driver without rest and without the possibility of changing the train. A duty is a sequence of trips which can be performed by one driver in the working day; it starts and ends at a home base (depot) and must satisfy a set of work laws and agreements. The duties scheduling problem focuses on designing duties using the pre-defined set of trips; the objective is to cover all trips while minimizing costs (the number of duties and penalties related to capacity constraints of home bases).

In practice, after solving the duties scheduling problem for each weekday, the resulting duties are assigned to individual drivers and sequenced into rosters, defining a weekly duties assignment for each driver. This process is called crew rostering and it is out of the scope of this work.

Duties scheduling represents a hard problem due to both the dimensions and the work laws and agreements constraints involved. Most of the Operations Research literature on duties scheduling concerns the airline context, the pairing optimisation problem. For a comprehensive review on this topic we refer the reader to [1] and [2]. In the railway context and especially for the passenger trains several approaches have been also proposed, for example in [3], [4], [5], [6] and [9]. However, the specificity of each problem passengers or freight, the company's rules and the level of details to be considered generally require the development of specific solutions methods. Our proposed method is based on coupling heuristic and column generation techniques.

2 Problem definition

In this section, we describe the basic principles of the duties scheduling problem for freight trains at the French railways SNCF; we present the input data, the constraints, the criteria, the degree of freedom and the results.

We are given a set of home bases (drivers' depots) that are located at the main nodes of the French railway network. Train drivers need to be allocated to the different home bases. Each home base has a limited capacity; i.e., a maximum number of drivers starting from that base. We can also start from scratch.

We also have, as an input, the planned schedule of trains for one week. Each train journey is composed of a set of trips; called also "*driving tasks*". A trip represents a part of a line where it is not allowed for a driver to change trains in between. It is characterized by a departure time, a departure station, an arrival time, and an arrival station.

If the departure station of the first trip in the duty is different from the starting home base, then the driver can travel as a passenger (by car, taxis or train) to the corresponding station. Equivalently, by the end of the last trip of the duty, the driver can travel as a passenger to the ending home base if the latter is different from the arrival station. These travels are called "*traveling tasks*"; they can also be performed between two driving tasks within the same duty. A traveling task can be performed between each couple of stations and each pair home base / station. Each traveling task is represented by time duration depending on the departure and the arrival station.

Two categories of constraints must be satisfied by a duties schedule: duties feasibility constraints and global costraints. In the sequel, we give the definition and the carachetristics of a duty, and after that we present these two categories of constraints.

A duty (see figure 1) is a set of sequenced driving tasks that must be performed by one driver; it represents its driving work for that day. It should be noticed that we include night shifts which may start late in one day and finish next day. Each duty starts from a home base and ends at a home base (not necessarily the starting one). In addition of driving tasks included in the duty, it is possible that the driver performs traveling tasks, takes a break meal and/or rests. Two types of rests are considered: short rests and long rests. Short rests do not exceed 1 hour. Long rests have a duration time of at least 1 hour and are not considered as a working time.
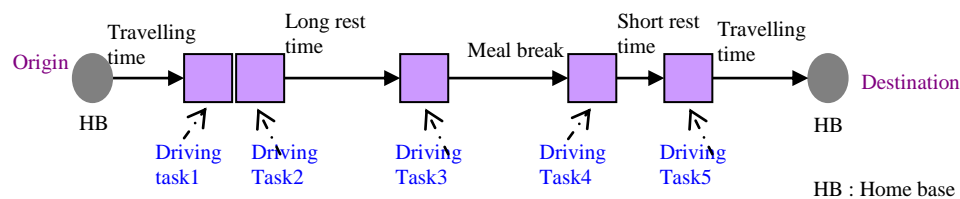
Figure 1 : Duty representation

The duties that are generated in the solution approach have to meet certain work laws and agreements in order to be feasible. Some of these laws and agreements are defined below and each one must be limited by an upper and lower bound:

- Spread time: the time between the departure from the starting home base of the duty and the arrival to the ending home base.
- Driving time: the sum of driving tasks (trips) durations.
- Working time: the driving time plus half of duration of traveling tasks, plus the duration of meal break and all rests, except long rests (longer than 1 hour).
- Overnight label: a duty is considered as overnight if it requires "some work" in the "night period" from 11 pm to 6 am. Indeed, there are several variants of overnight duties with different definitions for "night period" and "some work". For simplicity of presentation, we do not present all of them.
- Sequencing trips. They insure that each pair of consecutive trips *i* and *j* in a duty is compatible, the end location of trip i and start location of the tripe j should be identical and the time between both locations is greater than the possible traveling time allowances, …

In addition of the constraints that must be satisfied by each individual duty, there are also global constraints that are to be satisfied per home base or by the complete final schedule. These global constraints insure the feasibility of a set of duties; some examples are mentioned in the following.

- The average spread time of selected duties (global or per home base) must not exceed a given maximum duration.
- For each home base, the number of assigned duties should be less than its capacity.
- Compatibilities between drivers, these are associated to the home base and type of the engine that are assigned to train service, ...

The defined problem is a multi-criterion optimization problem. It consists to minimise, the number of uncovered tasks, the number of duties, the violated capacity at home base, etc. We have used a lexicographic weight to optimise these criteria.

The defined duties scheduling problem can be modeled as a generalized set partitionning integer program where each column represents a feasible duty.

A GUI (Graphical User Interface) was added to allow users to set a degree of freedom to guide the optimisation and to view the optimization process, to see and analyse the GANTT diagram associated to the list of duties with their composition (travel time, meal break, working time…), performance indicators that allow to assess the quality of the schedule; such as the average driving time, the level of unproductive time, number of travelling tasks, etc.

3 Problem solution

Because of the size of freight problem, we first decompose the week problem into periods problems. Each period is less or equal to two days (parameter). The process (see figure 2) starts by optimising the problem associated to the period P1. Then we freeze the duties that start on day1 and can finish on day2. After that we optimise the problem associated to the period P2 we include all the tasks that are not retained on period P1 and those of the day3. We do the same and so on until all periods are optimised.
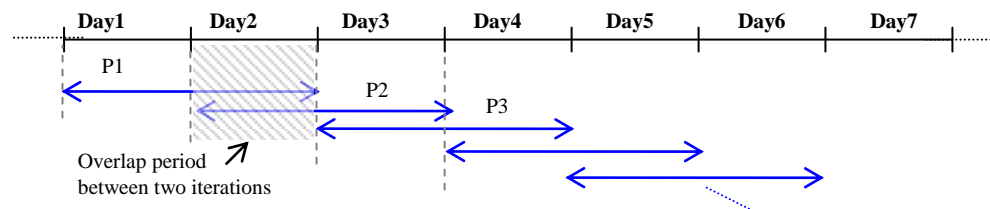

Figure 2 : problem décomposition.

The proposed solution method is divided in sequential and iterative steps:

For each period Pi=(day$_{[1]}$,day$_{[2]}$), i∈{1,...,N}
Step 0" initialize problem" : data preparation for period Pi
Step1 "Iterative random greedy construction method": by taking into account the characteristics of the problem we first construct quickly a good initial feasible solution by relaxing home base capacity constraints.
Step2 "Optimal or near-optimal duties optimization": We start from the initial feasible solution found in phase 1 and try to improve it by an iterative process. This is produced by solving column generation method.
Step3 "Data and output consolidation" : we freeze the duties of day$_{[1]}$ and identify the no retained tasks associated to day$_{[2]}$.
At the end of this process we consolidate the whole results.

Column generation is a popular technique that is widely accepted as an attractive solution method for this type of large scale optimization problems. References [7] and [8] give an interesting state-of-art on this technique.
In this work, column generation approaches have been developed to solve the duties scheduling problem. This approach consists in (i) solving the linear relaxation of the set partitioning problem by the column generation technique, and (ii) using a heuristic method to derive an integer solution, the final duties schedule. The column generation process starts by a restricted master problem (RMP) made-up by a small subset of columns obtained by a heuristic method. After that, the RMP is solved, and then the dual solution is transmitted to the pricing problem witch identifies new columns with negative reduced costs. The RMP is enriched by new columns and solved again. This process is repeated until no improving columns can be identified by the pricing problem. At this stage, the solution of the RMP represents the optimal solution of the linear relaxation of the set partitioning problem. The next stage consists on deriving an integer solution. For this purpose, two heuristic methods have been developed: restricted integer programming and adaptive rounding heuristics. Numerical tests performed on real data show that adaptive rounding heuristics provide solutions of better quality for the duties scheduling problem. In the sequel, we provide numerical results of the column generation approach with an adaptive rounding method (rounding up the largest factional variable at each iteration).

Tests on real world data show that our solution approach provides near-optimal solutions, with a duality gap less than 2%, for large scale problems within a reasonable computation time. Table 1 presents the results related to a real problem instance with about 700 driving tasks (trips). Each row represents a scenario for one day. For each scenario, we provide (i) the duality gap which is relative to the lower bound provided by column generation (ii) the improvement that is achieved comparing to an initial solution provided at the initialization phase of the column genaration approach, and (iii) the computation time. These results show that the solution method finds almost near optimal solutions within 5 minutes for large real world problems. The efficiency of the column generation approach is due to: (i) the structure of the problem that provides a tight relaxation with a good lower bound, (ii) and the use of efficient pricing scheme that generate one or several columns per depot at each iteration and hence accelerating the column generation process.

The proposed tool is intended for use as a strategic decision making tool. The validation results with the experts in charge for planing show that the obtained results are very promising.

Table 1: Results on the for real life data tested.

Scenario #	Duality Gap (%)	Improvement of Initial Solution (%)	Computation Time (sec)*
1	1.28	19.38	336
2	0.43	20.34	306
3	0.86	20.00	301
4	0.00	19.44	263
5	0.43	19.29	289
6	0.43	18.14	263

(*) The numerical experiments have been performed on a 2.33 GHz AMD Athlon PC with the memory of 2 Go, and running Windows XP. The algorithms have been implemented in MS Visual C++.NET 2005 and linked with the CPLEX 11.0 optimization library.

References

[1] Gopalakrishnan B. and Johnson, E. L., Airline Crew Scheduling: State-of-the-Art, Annals of Operations Research, 140, 305-337 (2005).

[2] Ernst, A. T., Jiang, H., Krishnamoorthy, M. and Sier, D., Staff scheduling and rostering: A review of applications, methods and models, European Journal of Operational Research, 153, 3-27 (2004).

[3] Wren, A, Fores, S, Kwan, A., Kwan, R., Parker, M. and Proll, L., A flexible system for scheduling drivers, Journal of Scheduling, 6, 437-455 (2003).

[4] Preling, R. And Lentink, R. M., A Decision Support System for Crew Planning in Passenger Transportation Using a Flexible Branch-and-Price, Annals of Operations Research, 127, 203-222 (2004).

[5] Fores, S., Proll, L. and Wren, A., TRACS II: a hybrid IP/heuristic driver scheduling system for public transport, Journal of the Operational Research Society, 53, 1093-1100 (2002)

[6] Capara, A., Fischetti, M., Guida, P. L., Toth, P. And Vigo D., Solution of large-scale railways crew planning problems: the Italian experience," N.M.H. Wilson (ed.); Computer Aided Transit Scheduling, Springer-Verlag, Berlin, 1-18 (1999).

[7] Barnhart, C., Johnson, E.L., Nemhauser, G. L., Savelsbergh, M.W.P. and Vance P. H., Branch-and-price: column generation for solving huge integer programs, Operations Research, 46, 316-329 (1998).

[8] Lübbecke, M. E. and Desrosiers, J., Selected topics in column generation, Operations Research, 53, 1007-1023 (2005).

[9] Chi-Kang LEE and Chao-Hui CHEN, Scheduling of train driver for Taiwan railway administration, Journal of the Eastern Asia Society for Transportation Studies, Vol.5, October, 2003

Optimization of the railroad blocking problem with temporal constraint

Housni Djellab
SNCF - I&R-AAD, 45 rue de Londres, 75000 Paris Cedex, France
Email: housni.djellab@sncf.fr

1 Introduction

The railroad blocking[1] problem (RBP) is formally defined as follows. We are given a set of shipments that must be routed from their origins to their respective destinations on a railroad physical network (figure 1). This network comprises a set of functional yards (with and without hump) connected by undirected links. Among the yards, some are classification yards where blocking operations (i.e., the grouping of incoming cars for connection with outgoing trains) are performed. In railroad freight transportation, a shipment, which consists in one or more cars with the same origin and destination (OD), may pass through several classification yards on its journey. The classification process often causes delays of several hours for the shipment, making it a major source of delay and unreliable service. Instead of classifying the shipment at every yard along its route, railroads group several incoming and originating shipments together to form a block. A block is defined by an OD pair that may be different from the OD pairs of individual shipments in the block and is defined by a slot. The slot is the mean that allows operator to use the infrastructure. The operator SNCF buys slots from RFF, railway infrastructure manager. The slot is defined by path, arrival/departure times at each node and the used trains characteristics (speed, weight ...). Once a shipment is placed in a block, it will not be classified again until it reaches the destination of that block.

Ideally, each shipment would be assigned to a direct block, whose OD is the same as that of the shipment, to avoid unnecessary classifications and delays. However, infrastructure availability and blocking capacity at each yard, determined by available yard resources (working crews, the number of classification tracks and switching engines), limit the maximum number of blocks and maximum car volume that each yard can handle, preventing railroads from assigning direct blocks for all shipments.
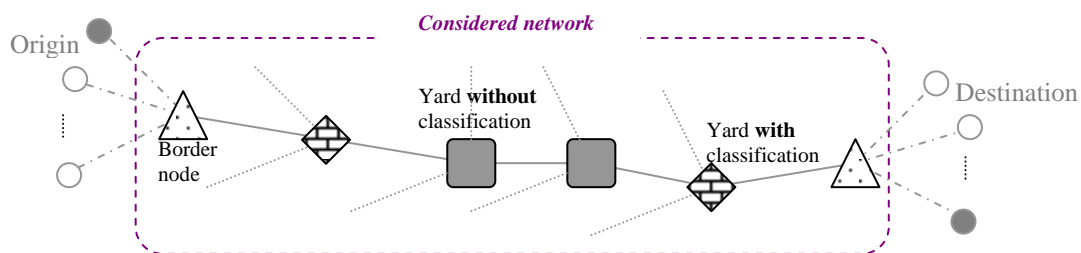
Figure 1. An example of a piece of a blocking network.

The main goal of the designer is to build blocks and to assign each shipment to dedicated blocks by minimizing total cost associated to the used resources (crew, locomotives and slots).

2 Problem definition

This section describes the basic principles of the temporal railroad blocking problem for freight trains at the French railways operator SNCF. It gives the input data of the problem, the constraints, the criteria, the degree of freedom and the output.

---

[1] A "*block*" is a group of cars that move together by one or more trains from a common origin or assembly point to a common destination or disassembly point.

2.1 The input data

This section summarizes the inputs of our blocking problem:

- Planning horizon, in general it is a loaded week
- Infrastructure (network) nodes and links with their characteristics
- Shipments defined by customer, OD, number of cars, ton, day of departure/arrival and characteristics
- Slots that the operator buy from railway infrastructure manager
- Partial solution (current blocking plan if one exists or pre-specified shipment routes)

2.2 The constraints

The constraints can be classified into two categories, hard and soft constraints. The hard constraint must be respected and soft constraints must be respected as far as possible.

- Yards' capacities
- Time windows to pick up and to deliver the shipments (hard constraints)
- Temporal constraints associated to the set of used slots (hard constraints)
- Constraints associated to the partial solution (current blocking plan if one exists or pre-specified shipment routes)

2.3 The criteria

We have defined a lexicographic weight to optimize the following criteria.

- Service quality which is defined by a delay penalty
- Train.km
- Slots cost
- Classification cost

3.4 The output

This section summarizes the outputs of our blocking problem tool:

- Blocks composition
- Routing shipment: how should the shipment, be routed over the blocks made?
- Yard workload
- Performances indicators to help the evaluation and comparison of the studied scenarios

3.5 The degree of freedom

These degrees of freedom help the user to define scenarios and to guide the optimization engine towards a realistic and good solution.

- Possible routing
- Yard capacity
- Slots availabilities
- Order of the criterion
- Discritized time step...

The main goal of this problem is to optimize the resources (locomotives, crew and slots) that are used by the trains to ensure the freight production. This is done by minimizing the number of blocks which is a way to achieve economies of scale and reduce the global cost. In other words we try to ensure that the locomotive always pulls the maximum number of cars. Because, with a locomotive there is driver and in front of there is a slot. The railroad blocking problem without temporal constraint is well known as an NP-Hard problem. Our contribution is to present a combination of two approaches to solve this type of industrial problem with temporal constraint: a heuristic (k-shortest path) and a mixed integer program.

3. Literature review

To the best of our knowledge, this problem has not raised a lot of research in the railway context; all the research that we found refers to the problem without temporal constraints. Chronologically, Bodin et al. [1] set the bases and their paper remains a reference in the domain by formulating the problem as a non linear MIP model. Crainic et al. [2] do not consider the RBP as an explicit problem and propose a model that takes into account the whole freight plan problem. The most recent research on the RBP is that of Newton [3] and Barnhart et al. [4] where the RBP is modeled as a network design model and formulated as a MIP. Ahuja et al. [5] proposes an efficient method relying on an algorithm

called "Very Large-Scale Neighborhood Search". Roughly speaking, this method starts with a feasible solution and iteratively improves the current solution by replacing it with its neighbor solution until the solution can no longer be improved.

The literature exposes very widely the manner to solve the RBP but the underlying models are distant from the reality of our problem, in particular with respect to specific constraints or used resources. For instance, temporal constraints where planning horizon must be discretized at a level that is more precise than the day (e.g. half an hour). For example, [6] presents their impossibility to tackle their problem with these existing models: it is not possible to reduce the number of blocks too much and create too long and heavy trains in certain areas where, as in the North of Italy, the network is constituted of many viaducts and tunnels. On the other hand, France designer must face the lack of slots availability, which makes the time factor a crucial parameter in the plan. The models that we propose in this paper are therefore different from the existing ones.

4 Problem solution and first results

Our proposed algorithm for temporal railroad blocking problem can be sketched as follows:

Step0: *"Initialize problem": read the whole input data (freight network, shipments ...) and initialize the parameters.*
Step1: *"Problem pre-processing": prepare the necessary data that are dedicated to the scenario to be studied. This means that according to the set of, shipments and the slots we reduce the size of the network and the possible and realistic set of blocks. This step has allowed us saving significant amounts of time for step 2 and 3.*
Step2: *"Identify a possible routing": For each shipment we add the set of possible and realistic itinerary (block). It was done by solving the k-shortest paths on the network of step 1.*
Step3: *"Blocks optimization": create and solve the corresponding mathematical model.*
Step4: *"Output " : display the detailed results and the indicators performances.*

In step 2, we generate a set of paths (cf. Figure 1): direct path from border node to border node, path from border node to border node by using yard with classification etc.

In step 3, we create two mathematical models and solve them by using Cplex an optimization solver from ILOG. We have tuned a Cplex parameters which has allowed us to reduce the computational time by 5% and especially to find quickly a good solution in the first iterations.

Our first investigation is to analyze two criteria (slot cost and train.km) and their impact on the reduction of the number of blocks. It is done by favoring one over another. We were not surprised to note that:
- The solution associated to the slot cost criterion, produces more direct routing but incurs a larger number of waiting cars even those with an origin and destination: *it operates to a **temporal** massification*.
- The solution associated to the train.km cost criterion prefers to lengthen the trip by waiting the cars on yard: *it operates to a **spatial** massification.*
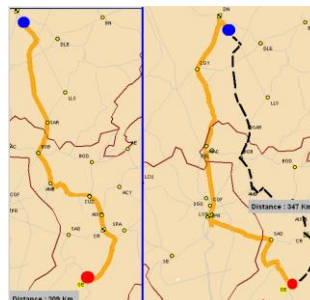

Figure 2. Comparison between slot cost and train.km criterion.

The example on figure 2 illustrates this behavior for an example with one shipment from "*Saulon*" to "*Grenoble*": The length of the trip in the solution with slot cost is 309km instead of 347km for the train.km criterion (+ 11%). The observation for the whole of the shipments is the same.

Based on the real tested data we can give the first conclusion:

1. The train.km criterion reduce of 31% the number of blocks in comparison with the slot cost criterion (by favoring one over another)
2. In a similar way, the size of the blocks (trains) is reduced by a factor of +33%.
3. Our solutions are 6% less costly than manual decision making process
4. The obtained solutions show an economy of 70% in train.km in comparison with a direct transport of each shipment that would consist in producing a train for each of them.
5. The massification has taken place on the yards and between the yards. In fact, we have trains almost three times longer on/and between yards than if the request had to be transported directly
6. The obtained average gap was less than 7% by limiting the computational time to 1 hour[2]. The numerical experiments have been performed on a 2.33 GHz AMD Athlon PC with the memory of 2 Go, and running Windows XP. The algorithms have been implemented in MS Visual C++.NET 2005 and linked with the CPLEX 11.0 optimization library.

By using the degree of freedom to guide the optimization engine the user can choose the order of criteria to find a good solution that: reduces the number of blocks, avoid unnecessary classifications on yards, reduces the length of the trip and guaranteed the best service quality.

The first obtained results are very promising although the solution search is time-consuming and needs to be reduced. Nevertheless, we show on real life data that the proposed method is well oriented to propose a solution that minimizes the number of blocks. The proposed engine is intended for use as a strategic decision making tool. It will help our organization in charge of the blocking plan to reduce the throughput time of the planning process. Meanwhile, it allows the organization to increase the flexibility and react faster to changes in the environment in particular for freight context. Further recognized advantages of the proposed tool are: the fact that the organization becomes less dependent on the experience and the craftsmanship of the planner.

Naturally, although the results are very promising, we need to be careful when drawing conclusion based on few real data only. Therefore, future research is necessary:

- To test the algorithm on other real data as well.
- To add step of constructing an initial feasible good solution to reduce computational time for solving the mathematical model.

References

[1] L.D. Bodin, B.L. Golden, A.D. Schuster, and W.Romig. A model for the blocking of trains. Transportation Research, 14B : 115 - 120, 1980
[2] Teodor Crainic, Jacques-A. Ferland, and Jean-Marc Rousseau. A tactical planning model for rail freight transportation. Transportation Science, 18(2) : 165-184, may 1984.
[3] Harry N. Newton. Network design under budget constraints with application to the railroad blocking problem. PhD thesis, Auburn University, 1997.
[4] Cynthia Barnhart, Hong Jing, and Pamela H. Vance. Railroad blocking : a network design application. Operations Research, 48(4) : 603-614, jul. 2000.
[5] Ravindra K. Ahuja, Krishna C. Jha, and Jian Liu. Solving real-life railroad blocking problems. Interfaces, 37(5) : 404-419, sep. 2007.
[6] Marco Campetella, Guglielmo Lulli, Ugo Pietropaoli, and Nicoletta Ricciardi. Freight service design for an italian railways company. In 6th Workshop on Algorithmic Methods and Models for Optimization of Railways. ATMOS, 2006.

---

[2] No improvement after several hours.

Trajectory-Adaptive Routing in Dynamic Networks with Dependent Random Link Travel Times

He Huang, Song Gao

Department of Civil and Environmental Engineering

University of Massachusetts Amherst

1 Introduction

Traffic networks are inherently uncertain with random disruptions, e.g., crashes, vehicle breakdown, bad weather, special events, construction and maintenance activities. They greatly affect the reliability of transportation systems and create significant congestion, which, as described in [1], is a problem in the United States' 439 urban areas and has gotten worse in regions of all sizes. A stochastic time-dependent (STD) network is required to capture such uncertainties, where link travel times are time-dependent random variables. Furthermore, there usually exist strong dependencies among random link travel times, largely due to traffic flow propagations over time and space. An important aspect of routing in an STD network is that real-time traveler information can be utilized to mitigate the adverse effects of uncertainties. The information can come from a number of sources, including personal observations, radio, variable message signs (VMS), and in-vehicle traffic information systems. A traveler can make his/her routing decision adaptive to information on prevalent traffic conditions, and doing so will generally lead to a better travel time. It is also noted that stochastic dependencies are generally required to capture the benefits of real-time information for network routing, since only through the dependencies over time and space can the knowledge of an incident at the current time result in a better prediction of traffic conditions in the future and elsewhere.

As the routing is adaptive to information, a traveler generally does not follow a fixed path. We loosely define such an adaptive routing process as a routing policy. Among the studies addressing optimal routing policy (ORP) problems, different assumptions have been made in terms of network stochastic dependency and information access (see the taxonomy in [2] and [3]). Network stochastic dependency is characterized by link-wise and time-wise stochastic dependencies of link travel times. When no link-wise or time-wise dependency is considered, travelers cannot make inferences about travel times on other links or in the future, even though they have information on the past. In a network with complete dependency, travel times on all links at all time intervals are dependent. A general representation of complete dependency is a discrete joint distribution of all link travel times. However, ORP problems in such a network could potentially suffer from problems such as the lack of data, large computer memory requirement and complication of algorithm design. Partial dependency lies between no dependency and complete dependency. The limited dependency assumptions are likely to reduce the algorithm complexity, yet how well they reflect real life situations is unresolved. Information access has the following categories: Perfect Online Information, where travelers have knowledge of the realizations of all link travel times up to current time period; No Online Information, where travelers have no knowledge of any link travel time realizations, and the routing is only adaptive to arrival times at intermediate nodes (termed time-adaptive routing); and Partial Online Information, where the amount of information lies between Perfect Online Information and No Online Information, with restrictions over space or time or both. [3] shows that, for optimal adaptive routing in a flow-independent STD network, more error-free information is always better (or at least not worse).

A number of studies works (e.g., [4], [5], [6], [7], [8],) address time-adaptive routing problems in a network with no dependency. Some studies address Partial Online Information cases with partial dependency, e.g., [9], [10], [11]. They either assume that link costs evolve as Markov processes, or adjacent link travel time dependency. [2] studies the case of Partial Online Information with complete dependency and designs exact and approximate algorithms. Later [3] continues the study of Partial Online Information and designs a heuristic algorithm.

The minimum information a traveler can get in an STD network is the travel times they have already experienced, i.e., the travel times on traversed links at corresponding arrival times at the source nodes, which is referred to as trajectory information in this paper. If travelers make decisions at each node based on not only the nodal arrival time but also trajectory information, they are making trajectory-adaptive route choices. [12] designs a label-correcting algorithm addressing multi-criterion trajectory-adaptive routing problems with no dependency. Later [13] investigates the relationship between time-adaptive and trajectory-adaptive routing. This work studies the single-criterion ORP problem with trajectory information in a complete dependency network. Note that the trajectory information is useless in a no dependency network, and in such a case the problem reduces to a time-adaptive routing problem which has been well studied. The complete dependency assumption is more realistic than no dependency for a traffic network, however it poses challenges in the algorithm design.

2 Trajectory-Adaptive Routing

Let $G = (N, A, T, C)$ denote an STD network. N is the set of nodes and A the set of links, with $|A| = m$. There is at most one directional link from node j to k, denoted as (j, k). T is the set of time periods $\{0, 1, \ldots, K-1\}$. Link travel times with entry times between 0 and $K-2$ are time-dependent and random, while those at and beyond $K-1$ static and deterministic. In an STD network with complete dependency, travel times on all links at all time periods are jointly distributed random variables. The travel time on each link (j, k) at each time t is a random variable,

which is assumed positive and integral, with finite number (R) of discrete support points. A support point is defined as a distinct value (vector of values) a discrete random variable (vector) can take. $C = \{C^1, \ldots, C^R\}$ is the set of support points of the joint probability mass function of all link travel times at all times, where C^r is a vector of time-dependent link travel times with a dimension of $K \times m$, $r = 1, 2, \ldots, R$. $C^r_{jk,t}$ is the travel time of link (j, k) at time t in the r-th support point, with probability p_r, and $\sum_{r=1}^{R} p_r = 1$.

Definition 1 (Trajectory) H *is a trajectory of node-time pairs a traveler could experience up to the current node j and time t: $H = \{(j_0, t_0), (j_1, t_1), \ldots, (j, t)\}$, where j_0 is the origin and t_0 the departure time.*

In trajectory-adaptive routing, the information contains the revealed travel time on link (j_x, j_{x+1}) at time t_x, which is $t_{x+1} - t_x$ for all (j_x, t_x) along the trajectory.

We seek the ORP with minimum expected travel time (METT) from all origins and departure times to a given destination. Similar to [2] and [3], the state variables for routing decisions can be specified as (j, t, H), and the decision is what next node k to take. In this case, Bellman's principle of optimality ([14]) will hold, i.e., at any intermediate state, the remaining decisions of an optimal policy must be optimal with regard to (w.r.t.) the state. An exact dynamic programming algorithm similar to that in [2] can be designed to solve the ORP problem. However, the number of states will be huge, due to the potentially exponential number of trajectories to any given node-time pair. In order to circumvent the curse of dimensionality in state space, a definition of trajectory-adaptive routing policy without the trajectory H in the state variable is given as follows:

Definition 2 (Trajectory-Adaptive Routing Policy) *A trajectory-adaptive routing policy $\mu(j, t)$ departing node j at time t is recursively defined as a combination of the next node k and the set of sub-policies $\{\mu_i(k, t_i)\}$, where t_i is the i-th possible arrival time at node k from the marginal distribution of $\widetilde{C}_{jk,t}$.*

Note that this is a recursive definition. The sub-policy μ_i at (k, t_i) is defined similarly as a combination of the

next node k' and sub-policies $\{\mu_i^1(k', t_i^1), \mu_i^2(k', t_i^2), \ldots\}$. The recursion stops at the destination d. Although each policy is defined over a node-time pair only, the recursive nature allows the routing decisions dependent on the trajectory. Consider two different possible trajectories to (j, t) by following a given routing policy. Assume they start to differ at (j', t') due to different arrival times at the next node k, and the next node-time pairs are (k, t_1) and (k, t_2) respectively. The sub-policies at the two node-time pairs can then be defined such that they will both reach (j, t) with a positive probability but contain different sub-policies from (j, t). This way, the decisions at (j, t) differ for the two different trajectories. However, the fact that the trajectory information is not included in the state does make Bellman's principle invalid.

Proposition 1 *A sub-policy of an optimal trajectory-adaptive routing policy as in Definition 2 is not necessarily optimal.*

The optimality is w.r.t. the state (j, t), or equivalently, METT over all support points. The sub-policy of an optimal policy must be optimal w.r.t. any intermediate state (j, t, H), or equivalently, with METT over a subset of support points that are compatible with travel times revealed through H (denoted as $EV(H) \in C$). However, it is not necessarily optimal in the whole set of support points.

With the hope of finding another property that can be maintained in the way optimality is maintained from a routing policy to all its sub-policies in Bellman's principle, non-dominated routing policy is defined. Other studies considering non-dominance include [5], [12], [15], [16]. Unfortunately, the hope evaporates with the fact that a sub-policy of a non-dominated routing policy is not necessarily non-dominated. The reason is similar to that why Bellman's Principle does not hold for trajectory-adaptive routing policy. However, good news is that pure routing policy can be defined based on non-dominated routing policy and it can be proved that for any origin-departure-time pair (j, t), there always exists an optimal routing policy which is pure.

Definition 3 (Non-Dominated Routing Policy w.r.t. Support Point Set B) *A routing policy μ at origin j with departure time t is non-dominated in support point set B iff $\exists$ no routing policy μ' such that $S_{\mu'}(j, t, r) \leq S_\mu(j, t, r), \forall r \in B$ and $\exists r^0 \in B | S_{\mu'}(j, t, r^0) < S_\mu(j, t, r^0)$, where $S_\mu(j, t, r)$ is defined as the travel time of routing policy μ from origin node j and departure time t to the destination node d if support point r is realized.*

Proposition 2 *A sub-policy of a non-dominated trajectory-adaptive routing policy as in Definition 2 is not necessarily non-dominated.*

It is trivial to show that non-dominance can be maintained at any intermediate state (j, t, H) or w.r.t. $EV(H)$. However, for the recursively defined trajectory-adaptive routing policy, non-dominance is checked at intermediate node-time pair (j, t), or w.r.t. the complete set of support points C. A sub-policy μ at (j, t) of a non-dominated policy from the origin could be non-dominated w.r.t. $EV(H)$ in such a way that it has an equal travel time as sub-policy μ' for each support point in $EV(H)$, but is dominated by μ' in $C \setminus EV(H)$, and thus dominated by μ' in C.

Definition 4 (Pure Routing Policy) *A routing policy is pure iff the routing policy itself and all its sub-policies are non-dominated; otherwise, it is a mixed policy.*

Proposition 3 *For any mixed routing policy λ at (j, t), there exists a pure routing policy μ such that $S_\mu(j, t, r) \leq S_\lambda(j, t, r), \forall r$.*

A straightforward conclusion can be drawn that, if mixed routing policy has METT, then there must exist a pure routing policy with the same METT.

Theorem 1 (Optimal Pure Routing Policy) *For any origin-departure-time pair (j, t), there exists an optimal routing policy which is pure.*

Theorem 1 suggests a solution algorithm that finds all the pure routing policies and chooses the one with METT.

3 Algorithm DOT-CD-Traj

Algorithm DOT-CD-Traj is based on the concept of decreasing order of time (DOT). Note that the construction

of routing policies at time t involves only routing policies at times later than t, due to the assumption of positive link travel times. At time $K - 1$ or beyond, the network becomes deterministic and static, and for any node-time pair (j, t) where $t \geq K - 1$ the set of pure routing policies $\chi(j, t)$ contains only one policy (the shortest path). Any deterministic static shortest path algorithm can be employed to compute the policy μ^* and its travel time to destination $e_{\mu^*}(j, t), \forall j \in N, \forall t \geq K - 1$, and assign it to the corresponding support point travel time $S_{\mu^*}(j, t, r), \forall r$. The set of pure routing policies at time $K - 1$ at any node is complete, i.e., no routing policy in the set will become mixed and no new pure routing policies will be constructed from later operations. Therefore the set of pure routing policies at time $K - 2$ constructed from pure sub-policies at time $K - 1$ is also complete. This procedure is continued down to time 0, and pure routing policy sets at all times will be constructed with one pass along the time dimension.

The algorithm will find all pure routing policies upon termination and thus will find the optimal pure routing policies. However it will miss the mixed non-dominated routing policies and thus the optimal mixed routing policies. The algorithm terminates after a finite number of steps, yet the worst-case complexity is exponential, and so heuristics might be needed to work more efficiently. Another way is to introduce hybrid routing, where adaptive routing is allowed at a subset of nodes and at other nodes travelers can only follow a prior path. Computational tests are conducted to evaluate average running time. Moreover, a comparison of Algorithm DOT-CD-Traj results in a dependent STD network and in an independent STD network ([4],[5],[6],[7]) can provide insights into the impact of network stochastic dependency on ORP problems.

References

[1] D. Schrank and T. Lomax, "2009 Annual Urban Mobility Report", Texas Transportation Institute, 2009.

[2] S. Gao and I. Chabini, "Optimal Routing Policy Problems in Stochastic Time-Dependent Networks", *Transportation Research Part B* 40, 93-122 (2006).

[3] S. Gao and H. Huang, "Is More Information Better for Routing in an Uncertain Network?", in The 88th Annual Meeting of Transportation Research Board Compendium of Papers DVD, Washington. DC, January 2009.

[4] R.W. Hall, "The Fastest Path through a Network with Random Time-Dependent Travel Times", *Transportation Science* 20, 182-188 (1986).

[5] E. Miller-Hooks and H.S. Mahmassani, "Least Expected Time Paths in Stochastic, Time-Varying Transportation Networks" *Transportation Science* 34, 198-215 (2000).

[6] D. Pretolani, "A Directed Hyperpath Model for Random Time Dependent Shortest Paths", *European Journal of Operational Research* 123, 315-324 (2000).

[7] E. Miller-Hooks, "Adaptive Least-Expected Time Paths in Stochastic, Time-Varying Transportation and Data Networks", *Networks* 37, 35-52 (2001).

[8] B. Yang and E. Miller-Hooks, "Adaptive Routing Considering Delays due to Signal Operations", *Transportation Research Part B* 38, 385-413 (2004).

[9] H.N. Psaraftis and J.N. Tsitsiklis, "Dynamic Shortest Paths in Acyclic Networks with Markovian Arc Cost", *Operations Research* 41, 91-101 (1993).

[10] S. Kim, M.E. Lewis, and C.C. White, "Optimal Vehicle Routing with Real-Time Traffic Information", *IEEE Transactions on Intelligent Transportation Systems* 6, 178-188 (2005).

[11] S.D. Boyles, Reliable Routing with Recourse in Stochastic, Time-Dependent Transportation Networks, Master's thesis, The University of Texas, Austin, TX (2006).

[12] S. Opasanon and E. Miller-Hooks, "Multicriteria Adaptive Paths in Stochastic, Time-Varying Networks". *European Journal of Operational Research* 173, 72-91 (2006).

[13] D. Pretolani, L.R. Nielsen, K.A. Andersen, and M. Ehrgott, "Time-Adaptive and History-Adaptive Multicriterion Routing in Stochastic, Time-Dependent Networks", *Operations Research Letters* 31, 201-205 (2009).

[14] R. Bellman, "On a Routing Problem", *Quart. Appl. Math* 16, 87-90 (1958).

[15] E. Miller-Hooks, Optimal Routing in Time-Varying, Stochastic Networks: Algorithms and Implementation, PhD thesis, The University of Texas, Austin, TX (1997).

[16] Y. Nie and X. Wu, "Shortest path problem considering on-time arrival probability", *Transportaion Research Part B* 43, 597-613 (2009).

Optimization of multi-modal transportation chains in city logistics

Luce Brotcorne

INRIA Lille-Nord Europe

France

Frederic Semet

LAGIS, Ecole Centrale de Lille

France

Alexandre Huart

LAMIH, University of Valenciennes, France

alexandre.huart@univ-valenciennes.fr

1 Introduction

The growing congestion of road infrastructure, particularly in urban areas, is alarming and requires solutions to increase the logistics productivity [1]. In terms of current practices of the logistics sector, we observe that: vehicle capacity is underutilized, many warehouses have storage capacity unused, rail transportation, metro or waterways are underutilized. Based on this observation and to improve their profitability, service companies managing a fixed capacity aim to increase their occupancy rates. To achieve this goal, we propose an innovative strategy among all actors of the logistic chain which relies on the mutualization of unused logistic capacities.

In this presentation, we first propose a model for routing demands over a network defined by logistic services. Then we propose a strategy of acceptance and allocation of ponctual logistic demands. More precisely, a demand is accepted if the expected utility resulting from its acceptance is greater than the one related to its refusal.

2 Routing demand on a network of logistic services

The mutualization solution described in this presentation is based on a pooling of untapped logistic capacity. The purpose of this sharing policy is to create a new network of logistic services able to increase the overall ability of the system to satisfy efficiently new demands. At the opposite of the current practices, the answer to a logistic demand can result from the consolidation of several capacities under the control of different companies. To incorporate this innovative aspect in logistics, the demand routing problem on the logistic service network is modeled as a multicommodity

flow problem defined on a time-space network. The objective is the maximization of the number of accepted demands. To accept a demand means to route the associated flow on the network.

A transportation service is modeled by an available capacity during a time window and two locations where the means of transportation is available and should be returned. In the context of an intermodal transportation system [2], several modes of transportation are considered (heavy or light vehicles, metro ...). A logistic service can also result into a storage capacity available during a time window. Note that the transfer of goods between vehicles takes place in a warehouse. A logistic demand consists to pick-up goods within a time window in a given location and to deliver them within a time window at destination.

Let $G = (N, A)$ be a directed network modeling logistic services. N is the set of nodes and A is the set of arcs. A node $n \in N$ is defined by a triplet (z, t, k) where z belongs to the set of geographical areas Z, t belongs to the set of time periods T and k belongs to $K = \{V \cup S \cup P\}$. V denotes the set of vehicles, S the set of warehouses and P the set of demands. More precisely, if $k \in V$, (z,t,k) means that the vehicle k is in area z at period t. If $k \in S$, (z,t,k) corresponds to the warehouse k in area z at t, and finally if $k \in P$, (z,t,k) represents the origin or the destination location for demand k. For $i, j \in N$, arc $(i, j) \in A$ means that goods can be moved from i to j. For each arc the capacity and the transportation or storage cost between the two nodes are given.

One key point is how this time-space network can be built. The storage service allow to the cross-docking of goods or their storage. A transportation service is characterized in this network by the origin and destination locations corresponding to the geo-temporal positions. Reducing a service to a direct link between the origin and destination locations does not reflect the potential of transportation that it generates. Therefore we enumerate feasible paths from the origin location to the destination location across the geographical areas visited to model a transportation service. More precisely, for each transportation service, we create a subnetwork G_k that represents the availability of capacity related to service vehicle k. The directed network $G = (N, A)$ including all logistic services consists of $|V|$ layers associated to G_k $(k = 1, ..., |V|)$ as well as $|S|$ layers for storage and $|P|$ layers for the origins and the destinations of demands. With each demand $p \in P$ is associated a pair (o^p, d^p) where $o^p \in N$ and $d^p \in N$ correspond respectively to the origin and the destination nodes of demand p. Moreover, we denote by q^p, the quantity of product p to be transported from o^p to d^p. The binary decision variables are :

$Y_p = 1$ if the demand p is accepted, 0 otherwise.

$X_{ijtk}^p = 1$ if product p is moved by vehicle k from i at period t to j, 0 otherwise.

$B_{ijtk} = 1$ if vehicle k goes from i to j at t, 0 otherwise.

$S_{itkk'}^p = 1$ if product p is transfered from logistic means k to logistic means k', 0 otherwise. .

$S_{itkk}^p = 1$ if product p is stored in k from the period t to $t+1$, 0 otherwise.

We denote by Z_p the subset of zones including the origin and destination locations of demand p. For $k \in V$, Z_k represents the set of zones served by vehicle k while Z_k reduces to the zone where the warehouse is located when $k \in S$. For demand p, T_p represents the set of time periods where goods are available to be moved. T_{1p} and T_{2p} are the sets of periods associated with the origin and the destination. For logistic means k, T_k represents the set of time periods where k is available. T_{kp} is the intersection of T_p and T_k. We denote by ct_{ijtk} the available capacity on vehicle k from i at period t to j and cs_{itk} the available capacity of warehouse k in area i at period t. We formulate the allocation of demands on the network of logistic services as follows:

$$max \sum_{p \in P} Y_p \qquad (1)$$

$$st : \sum_{k' \in V} S^p_{itpk'} = Y_p \qquad i = o^p \in Z_p, \forall t \in T_{1p}, p \in P \qquad (2)$$

$$\sum_{k' \in V} S^p_{itk'p} = Y_p \qquad i = d^p \in Z_p, \forall t \in T_{2p}, p \in P \qquad (3)$$

$$\sum_{j \in Z_k} X^p_{jit'k} - \sum_{j \in Z_k} X^p_{ijtk} + \sum_{k' \in S \cup P} (S^p_{it'k'k} - S^p_{itkk'}) = 0 \quad \forall i \in Z_k, t \in T_{kp}, k \in V, p \in P \qquad (4)$$

$$\sum_{k' \in V} S^p_{i(t-1)k'k} - \sum_{k' \in V} S^p_{itkk'} + S^p_{i(t-1)kk} - S^p_{itkk} = 0 \quad \forall i \in Z_k, t \in T_{kp}, k \in S, p \in P \qquad (5)$$

$$\sum_{p \in P} X^p_{ijtk} q^p <= ct_{ijtk} B_{ijtk} \quad \forall i \in Z_k, j \in Z_k, t \in T_k, k \in V \qquad (6)$$

$$\sum_{p \in P} \sum_{k' \in K} S^p_{itkk'} q^p <= cs_{itk} \quad \forall i \in Z_k, t \in T_k, k \in S \qquad (7)$$

$$\sum_{i \in Z_k} \sum_{j \in Z_k} B_{ijtk} <= 1 \quad \forall t \in T_k, k \in V \qquad (8)$$

$$\sum_{j \in Z_k} B_{jit'k} - \sum_{j \in Z_k} B_{ijtk} = 0 \quad \forall i \in Z_k, t \in T_k, k \in V \qquad (9)$$

$$Y_p \in \{0,1\} \qquad \forall p \in P \qquad (10)$$

$$S^p_{itkk'} \in \{0,1\} \qquad \forall i \in Z_k, t \in T_k, k \in K, k' \in K, p \in P \quad (11)$$

$$X^p_{ijtk} \in \{0,1\} \qquad \forall i \in Z_k, j \in Z_k, t \in T_{kp}, k \in V, p \in P \quad (12)$$

$$B_{ijtk} \in \{0,1\} \qquad \forall i \in Z_k, j \in Z_k, t \in T_k, k \in V \qquad (13)$$

Constraints (2) and (3) ensure that supply and demand requirements are met, relations (4) and (5) correspond to flow conservation constraints while constraints (6) and (7) are the capacity constraints. Constraints (8) and (9) ensure the feasibility of the transportation plan. Note that in constraints (4) and (9), t' is obtained from the transport duration or is equal to $(t-1)$ depending whether we consider the transfer of goods between zones or their storage in a warehouse.

3   Strategies to manage punctual demands

The strategies used here are designed to manage the allocation of punctual demands given a network of logistic services. With each demand is associated a price based on the anticipation, the delivery time requirement and the quality of service. The strategy proposed for demand management aims to answer the following questions:

- Is there a feasible solution?

- Is it profitable to accept this demand or is it better to keep the remaining capacity for high price demand that could arise later?

For each logistic demand request, the expected utility evaluation of a solution is based on three types of demands:

- the already accepted demands which must be moved through the network of services

- the current demand

- the forecasted demands for which it is possible to keep some logistic capacity.

A logistic demand is accepted if it is feasible and the expected utility resulting from its acceptance is greater than the one related to its refusal. The expected utility computation requires the solution of a slightly revised version of the mathematical program described in the previous section. The objective function has to be replaced by the expected utility function including a term associated with each demand type. Capacity constraints need to include forecast demand potentially accepted.

In this presentation, we will describe efficient solution methods to solve the problems defined in Sections 2 and 3. Assessment of the proposed formulation and solution methods will be conducted on randomly generated instances based on real life data of Paris urban area. Preliminary experiments consist in solving the formulation described in Section 2 with the state-of-the-art LP/MIP solver CPLEX (Version 11.2) on a computer operating at 3.2 Ghz and equiped with 16 Gig RAM. Instances with 50 to 100 demands, 10 to 80 vehicles and 10 to 80 warehouses were solved to optimality in less than 35 seconds.

References

[1] T. G. Crainic , "City logistics," *Tutorials in Operations Research, Z.L. Chen and S. Ravaghavan (eds), INFORMS*, 2008.

[2] T. G. Crainic and K. H. Kim, "Intermodal transportation," *Handbooks in Operations Research and Management Science, Transportation*, 14, North-Holland, Amsterdam, pp. 467-537, 2007.

Optimization Method for Evacuation Instructions – Influence of the Parameter Settings

Olga Huibregtse, Serge Hoogendoorn, Andreas Hegyi and Michiel Bliemer

Department of Transport and Planning

Delft University of Technology

o.l.huibregtse@tudelft.nl

1 Introduction

Natural disasters, like bush fires and floods, often cause many casualties. To avoid this as much as possible, authorities have to be prepared for such disasters. This includes creating a plan to evacuate people from a threatened region. During an evacuation, people involved have to leave the region over the accessible. They make decisions about departure times (which includes the decision to evacuate or not), routes and destinations. Without any interventions, the decisions people make are most likely not system-optimal, primarily caused by a lack of information. Giving optimized evacuation instructions to the people can lead to a more effective evacuation (i.e., shorter evacuation times, less casualties).

In earlier research, a method has been presented to optimize evacuation instructions (departure times, destinations and routes) for an arbitrary hazard (kind of hazard and time and spatial pattern) and region (road network and population distribution) [1]. Traffic is simulated by an evacuation simulation model and the performance of instructions is determined as the value of an objective function (often related to the safe arrivals over time), both chosen by the user of the method. Optimization methods for evacuation instructions have been developed before, see e.g. [2] and [3]. The main improvement of our method presented in [1] is the simultaneous optimization of departure time, destination, and route instructions instead of the optimization of only one or two of these variables for a dynamic instead of static evacuation problem. In addition, the method can be used to optimize instructions under uncertain conditions, like uncertainty in the hazard pattern, as discussed in [4].

Application of the method results in instructions which performance approximates the performance of the optimal instructions. This near-optimality (with respect to the arrivals) is shown in [5]: for three different applications of the method, the average intensities over time on the links leading to the destinations are equal to respectively 86, 86, and 88% of the network capacity. The network capacity is an upper bound on the intensities under optimal instructions (the real intensities are unknown), thus the deviations from optimal use of the links are expected to be small (maximum 14%).

The optimization method contains parameters influencing the exploration and concentration of the search process. In this abstract is researched how the parameter settings influence the performance of the optimized instructions and the speed of convergence. The findings provide insight into suitable parameter settings for efficient optimization in new cases.

2 Summary Optimization Method

The optimization method presented in [1] (see [1] for more details) contains an algorithm wherein instructions are created by assigning groups of evacuees to combinations of departure times and routes (indirectly implying destinations). To contribute to the applicability of the method and the resulting instructions, only a selection of all possible combinations are considered in the algorithm. Before applying the algorithm, this selection is made for each origin separately. The selected combinations are called *elements* $u \in U$, where U is the set of all elements and U_r is the set of elements for origin r.

The structure of the algorithm is based on ant colony optimization [6], an algorithm to solve numerical problems that is based on the communication behavior of ants. In each iteration, each ant in a colony M creates instructions for all evacuees by assigning each group of evacuees to an element. For each group, an element u is selected by using the following iteration-dependent selection probability:

$$h_u(i) = \frac{\gamma_u(i)\kappa_u}{\sum_{u \in U_r} \gamma_u(i)\kappa_u} \in (0,1],$$ (1)

where $\gamma_u(i)$ is the value of the so-called *pheromone trail* belonging to element u in iteration i for which holds $\gamma_u(1) = 1, u \in U$, and κ_u is a scalar representing the *problem-dependent information* for element u. The *problem-dependent information* is constant for all iterations and gives elements with relatively low free flow travel times and relatively early departure times a relatively high selection probability, because they are expected to have a positive influence on the evacuation:

$$\kappa_u = \left(\frac{\min_{u \in U_r}\left\{\tau_{p_u}^{\text{free}}\right\}}{\tau_{p_u}^{\text{free}}} \right)^{\xi_2} \left(1 - \xi_3 \frac{k_u}{\max K_{r_u}} \right) \in \langle 0,1],$$ (2)

where $\tau_{p_u}^{\text{free}}$ is the free flow travel time of the route which is part of u, k_u is the departure time which is part of u and K_{r_u} is the set of departure times for which there are elements belonging to r_u, the origin r where u belongs to. For the parameters ξ_2 and ξ_3 holds $0 \leq \xi_2 \leq 1$ and $0 \leq \xi_3 < 1$. *Pheromone trails* are iteration-variant: they are updated at the end of each iteration, whereby the trails of elements included in the global-best instructions are raised:

$$\gamma_u(i+1) = \begin{cases} \rho\gamma_u(i) + \xi_4\Delta\gamma_u(i) & \text{if } u \in U_{gb} \\ \rho\gamma_u(i) & \text{otherwise} \end{cases} \in \langle 0,\infty\rangle,$$ (3)

where $\Delta\gamma_u(i)$ is the amount of pheromone added to U_{gb}, the set of elements where groups are assigned to in the global-best instructions (the instructions with the highest performance over all iterations). The performance of the instructions is determined by using a simulation model and an objective function. The factor $\Delta\gamma_u(i)$ depends on the performance, the population and the number of people assigned to a specific element. For the parameters ρ and ξ_4 holds $0 < \rho < 1$ and $\xi_4 > 0$.

3 Analysis of the Influence of the Parameter Settings

The analysis concerns the parameters ρ and ξ_4, both influencing the exploration and concentration of the search process. Different values for these parameters are tested for different scenarios to study the
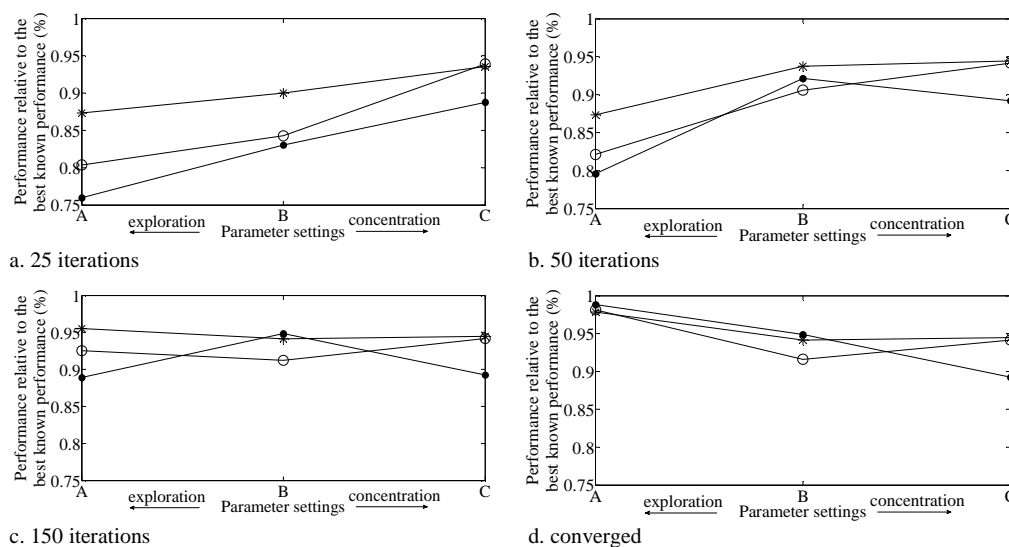
influence on the performance of the optimized instructions and the speed of convergence. The tested parameter settings and scenarios are listed in Table 1. The scenarios are all floods of Walcheren, a peninsula in the southwest of the Netherlands. In Scenario 1 and 2 all residents have to be evacuated, in Scenario 3 only a part of the residents. Scenarios with different sizes of the search field (# possibilities) are included in the test, because of an expected relation between this size and the influence of the exploration and concentration parameter settings. The size follows from the number of elements (depending on the departure time step and the maximum number of routes per departure time) and the number of groups per origin (depending on the maximum group size). For more information about the influence of the elements and the groups and about the Walcheren case, see [1].

Table 1. Case study: parameter settings, and scenarios and their properties

Settings	ρ	ξ_4	Scenario	# Origins	Population	Departure time step	Max # routes per departure time	Maximum group size	# Possibilities
A	0.98	0.1	1	23	121,838	0.5	5	10,000	$9.9.10^{36}$
B	0.9	1	2	23	121,838	1	5	10,000	$8.3.10^{29}$
C	0.8	10	3	8	67,358	0.5	5	10,000	$2.8.10^{15}$

The traffic flows are simulated using the evacuation simulation model EVAQ [7], containing the dynamic travel demand, the en-route travel choice behavior and dynamic network loading with queuing and spill-back. The applied objective function is the maximization of a function of the number of arrivals for each time period, where early arrivals have a higher weight than late arrivals (see [1]).

The optimization method is applied for each combination of parameter settings and scenarios. All evacuees are assumed to comply with the instructions. For each combination, the method is applied four times (because of the stochastic in the method) and the averaged results are shown in Figure 1.

a. 25 iterations   b. 50 iterations

c. 150 iterations   d. converged

Fig. 1 Influence of the exploration and concentration parameters (ρ, ξ_4) on the performance of the best evacuation instructions found over the iterations, in different stadia of the optimization process, for Scenario 1 ($\bullet$, $9.9.10^{36}$ possibilities), Scenario 2 ($\circ$, $8.3.10^{29}$ possibilities), and Scenario 3 (*, $2.8.10^{15}$ possibilities).

Figure 1 shows the following: 1) For a low number of iterations, more concentration leads to higher values of the relative performance, while after convergence, more exploration leads to the higher values. The trend changes in the phases in between. 2) The bigger the size of the search field is, the bigger the improvements in the relative performance are by increasing the number of iterations. 3) For the highest exploration parameters chosen, the relative performance is for all scenarios equal to 98 or 99% of the best known performance.

The results show the differences and equalities in the influences of the exploration and concentration parameters for the different scenarios. Depending on the situation, suitable parameters should be chosen. When instructions are created beforehand, the effectiveness of the instructions is the most important criterion, but when there is a time limit, a balance between the effectiveness and the speed of convergence has to be found because of the computational time of the optimization method.

4 Conclusions

By varying the exploration and concentration parameter values in an optimization method for evacuation instructions, the performance of the optimized instructions and the speed of convergence of the method are influenced. The results provide insight into suitable parameter settings for efficient optimization in new cases. An extension of this research (analysis of more case studies and parameter settings) should lead to rules to choose the exploration and concentration parameter settings for an arbitrary problem, given the size of the search field and the time available to find the best instructions.

References

[1] O.L. Huibregtse, S.P. Hoogendoorn, A.J. Pel, M.C.J. Bliemer, "A Generic Method to Optimize Instructions for the Control of Evacuations", 12th IFAC Symposium on Control in Transportation Systems, California, U.S. (2009).

[2] Y. Liu, X. Lai, G. Chang, "Two-Level Integrated Optimization System for Planning of Emergency Evacuation", *Journal of Transportation Engineering*, 32(10), 800-807 (2006).

[3] E. Miller-Hooks, G. Sorrel, "The maximal dynamic expected flows problem for emergency evacuation planning", *Transportation Research Record* 2089, 26-34 (2008).

[4] O.L. Huibregtse, S.P. Hoogendoorn, M.C.J. Bliemer, "Optimizing of Evacuation Measures under Uncertainty", Proceedings of the 89th Annual Meeting of the Transportation Research Board, Washington DC, U.S. (2010).

[5] O.L. Huibregtse, M.C.J. Bliemer, S.P. Hoogendoorn, "Analysis of Near-Optimal Evacuation Instructions", Proceedings of the 1st International Conference on Evacuation Modeling and Management, Den Haag, the Netherlands (2009).

[6] M. Dorigo, V. Maniezzo, A. Colorni, "The ant system: optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man and Cybernetics*, 26(1), 29-41 (1996).

[7] A.J. Pel, M.C.J. Bliemer, S.P. Hoogendoorn, "EVAQ: A New Analytical Model for Voluntary and Mandatory Evacuation Strategies on Time-varying Networks", Proceedings of the 11th IEEE Intelligent Transportation Systems Conference, Beijing, PR China (2008).

Routing and Scheduling of Roll-on/Roll-off Ships with Simultaneous Cargo Selection and Stowage Decisions

Lars Magnus Hvattum

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, Norway

Email: lars.m.hvattum@iot.ntnu.no

Bernt Olav Øvstebø

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, Norway

Kjetil Fagerholt

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, Norway

1   Introduction

Roll-on/Roll-off (RoRo) ships transport cargo on wheels such as cars, trucks, farming equipment, and military equipment. Intercontinental trade involved more than 17 million vehicles in 2004, growing 5% annually, and regional trade involved more than 26 million vehicles, growing 3–4% per year (MDS Transmodal, 2006). Almost all of this trade is conducted using RoRo ships. The deep sea fleet for vehicle transport, consisting of ships taking more than 3000 CEUs (Car Equivalent Units), had 355 ships in 2006, and the regional fleet of ships taking less than 3000 CEUs consisted of 152 ships. Although they have a special role in international trade, only few scientific publications are concerned with the development or analysis of decision support tools for planning the operations of RoRo ships. Existing publications mainly focus on safety and avoiding roll motion, for example (Vassalos and Konovessis, 2008) and (Kreuzer et al., 2007), where the latter present a simulation tool for calculating how strongly trailers need to be secured to the deck. Examples of work in related areas include Mangan et al. (2002), who present a methodology for port/ferry choice from a shipper's point of view, and Mattfeld (2006), who investigates how transhipment terminals should

handle the transportation of vehicles. We will focus on the shipping company, and the planning required when operating a fleet of RoRo ships.

RoRo ships sail between different regions of the world according to predefined plans. Planning of operations in the maritime transport industry can mainly be divided into three categories: strategic, tactical, and operational planning. Strategic planning is concerned with a time horizon of several years, and typically involves decisions such as determining the fleet size and mix. On a tactical level, the decision maker must determine which ships should operate which routes and when they are required to arrive to each region. These decisions lay the premises for planning on the operational level. On the operational level, one must decide which cargoes to carry, and which routes to follow in order to pick up and deliver these cargoes. In addition for RoRo ships, operational decisions must be made regarding stowage: for a given route with pick-ups and deliveries, it must be determined how the cargoes should be stored on the ship during the voyage.

When creating these plans, the planner must at all times balance the scope of the plan and the tractability of the problem. Increased scope, that is, planning for longer time periods and more ships simultaneously, gives the planner more flexibility. This enables him/her to find solutions that exploit synergy better than if the problem was partitioned into smaller problems and solved sequentially. However, solving a planning problem of large scope is more difficult than solving the subproblems of which it is composed. In addition, problems of larger scope typically require more information, which may not be readily available. In this work we focus on the operational level of planning, trying to incorporate as many decisions as possible while studying whether the resulting problem can be solved efficiently by exact or heuristic solution algorithms.

2 Problem Description

A RoRo ship consists of a number of decks on which vehicles can be stored. Some of these decks have adjustable heights: when carrying farm equipment more height is required between decks than when transporting sports cars. Increasing the height of one deck can only be done by simultaneously decreasing the height of another deck. Each deck also has a given width and length. To utilize the full capacity of the ship it may be necessary to consider which vehicles should be stored next to each other, so as to avoid empty space on the decks. Some cargoes may have flexible quantities, meaning that within certain limitations the shipping company can decide how many vehicles to lift, with a loss of revenue associated with reducing the transported quantity. While the shipping company is obliged through contracts to carry certain mandatory cargoes, additional spot cargoes may be available for which a revenue can be collected if they are transported. Although cars are very light, compared to other materials transported by other types of ships, there are stability requirements to consider: if the centre of gravity is too far to the side or too high, the

ship may become unstable during the voyage. Cargo is loaded/unloaded using a single ramp on a last-in-first-out basis. When many different cargoes are transported together, there is an additional inconvenience connected to cargoes sharing lanes if the cargo loaded first should also be discharged first.

We have studied two related problems. In the first we consider a fixed route, where all port calls have been decided in advance. The task is then to decide which cargoes to carry, how many vehicles to carry from each cargo, and how to stow the vehicles for the duration of the voyage. In the second problem we consider the routing and scheduling decisions in addition to the stowage: each cargo has an origin and a destination, and the task is to decide the sequence of port calls to pick up and deliver all mandatory cargoes as well as optional cargoes if feasible and profitable. There are time windows for each cargo, as well as for an artificial end node. The time windows of the cargoes are very wide, but there is a potential profit from ending the route early: if the ship becomes available earlier it is free to make more port calls in its next voyage, thus potentially gaining more revenue for the shipping company. On top of the routing and scheduling comes the decisions regarding stowage, and it should be noted that the routing decisions may influence whether feasible stowage decisions exist due to the ship stability constraints.

3 Solution Methods

We have modeled the two problems described above as mixed integer programs, and solved them using Xpress MP. A heuristic solution method is also developed, consisting of an initial construction heuristic, a tabu search to improve the routing and scheduling, a squeaky wheel optimization construction heuristic for creating stowage plans, and an additional local search to improve the stowage plans. The following describes how the heuristic works when solving the problem including routing and scheduling decisions.

The initial solution is created by a simple construction heuristic by inserting all mandatory cargoes into a route one by one. The construction heuristic does this twice, first assuming that the minimum quantity is taken for each cargo and then assuming that the maximum quantity is taken. If the solution with maximum quantities is feasible it is taken as the initial solution, otherwise the solution with minimum quantities is used.

Tabu search is used for the main part of the heuristic. The basic move is to reassign a cargo in the current route by removing its pickup and delivery point and reinserting them in the best possible way. A periodic diversification mechanism is used to allow optional cargoes to enter the solution and to adjust the vehicles quantities taken from each cargo. Binary search is used to determine the best quantity to carry of a given cargo, and a reduction in the quantity carried from other cargoes is considered when performing the periodic diversification. A stowage plan is only

created when a new potentially best solution has been identified during the normal tabu search, or when adjusting the cargoes carried during the periodic diversification.

When creating stowage plans we consider a fixed route and fixed cargo quantities. Stowage plans are created using a construction heuristic guided by squeaky wheel optimization (Joslin and Clements, 1999). The construction heuristic ignores ship stability constraints, and to repair solutions when these are violated we propose a simple first improvement local search.

4   Results

Both the heuristic solution method and Xpress MP are analyzed in terms of their ability to efficiently solve the ship routing problem described above. Extensive computational tests have indicated that the two solution methods have significantly different strengths and weaknesses. For tiny problem instances, Xpress performs better than the heuristic, but for slightly larger instances only the heuristic finds any feasible solutions.

References

D.E. Joslin and D.P. Clements. "Squeaky Wheel" Optimization. *Journal of Artificial Intelligence Research*, 10:353–373, 1999.

E. Kreuzer, V. Schlegel, and F. Stache. Multibody simulation tool for the calculation of lashing loads on RoRo ships. *Multibody System Dynamics*, 18(1):73–80, 2007.

J. Mangan, C. Lalwani, and B. Gardner. Modelling port/ferry choice in RoRo freight transportation. *International Journal of Transport Management*, 1:15–28, 2002.

D. Mattfeld. *The Management of Transshipment Terminals: Decision Support for Terminal Operations in Finished Vehicle Supply Chains*. Springer, 2006.

MDS Transmodal. Roro freight market overview, 2006. `http://www.mdst.co.uk/205089presentation.ppt`, accessed 10 September 2008.

D. Vassalos and D. Konovessis. The thematic network SAFER EURORO: An integrated approach to safe European RoRo ferry design. *Marine Technology*, 45(1):1–8, 2008.

An Aggregate Label Setting Policy for the Multi-Objective Shortest Path Problem

Manuel Iori

Dipartimento di Scienze e Metodi dell'Ingegneria

Università di Modena e Reggio Emilia, Italy

Email: manuel.iori@unimore.it

Silvano Martello

Dipartimento di Elettronica, Informatica e Sistemistica

Università di Modena e Reggio Emilia, Italy

Daniele Pretolani

Dipartimento di Scienze e Metodi dell'Ingegneria

Università di Modena e Reggio Emilia, Italy

1   Introduction

Multi-objective shortest path problems consist in finding a set of paths that minimizes a number of objective functions. The objectives commonly include the sum of costs and/or the maximum (*bottleneck*) cost in the path. Such problems have considerable practical relevance, as they appear in a number of real world applications. We refer for example to the transportation of hazardous material (see, e.g., [2]) in which the traveled distance is not the only objective but other costs (probability of accidents, population density, and so on) have a relevant impact. In many applications, the *quality* of the roads (highways, local routes, and so on) or the risk of accident can be seen as bottleneck objectives, see, e.g., [1]. The reader is referred to [4] for an exhaustive review.

Formally, we are given a directed graph $G = (V, A)$, defined by a set of *vertices* $V = \{1, 2, \ldots, n\}$ and a set A of m *arcs* (i, j) $(i, j \in V)$. Each arc $(i, j) \in A$ has k associated non negative *costs* $c_1(i, j), c_2(i, j), \ldots, c_k(i, j)$, and there are k different objective functions, one per cost type. A prefixed vertex $\sigma \in V$ is called the *source*. Given a vertex $i \in V$, and a path $P_q(i)$ $(q = 1, 2, \ldots)$ from σ to i, we define the corresponding *label* of vertex i as the k-tuple $\ell_q(i) = [f_1^q, f_2^q, \ldots, f_k^q]$ that gives the k objective function values of path $P_q(i)$. A path $P_q(i)$ *dominates* another path $P_r(i)$ if $f_h^q \le f_h^r$ for $h \in \{1, 2, \ldots, k\}$ and $f_h^q < f_h^r$ for at least one $h \in \{1, 2, \ldots, k\}$. A path is said to

be *non-dominated* if no other path dominates it. The set of all non-dominated paths is called the *maximal complete set*. The *Multi-Objective Shortest Path Problem* (MOSPP) considered in this paper is to find the maximal complete set of paths from σ to any other vertex $i \in V$, where the objective functions include both sum and bottleneck criteria. The problem is NP-hard.

Most of the approaches to the exact solution of multi-objective shortest path problems fall into two main categories, namely label setting and label correcting algorithms. The label setting algorithms for MOSPP generally adopt a *lexicographic* ordering of the labels. However, other orderings can be adopted, in particular those based on an *aggregate function*, i.e., a weighted sum of the label values. Aggregate orders have been extensively used for the bi-objective case (see [5]), as well as for the resource constrained shortest path problem, in which one looks for a shortest path subject to a number of capacity constraints on additional weights associated with the arcs (see, e.g., [6] for the case of a single capacity constraint). For the general case of MOSPP, aggregate orders were suggested as a possible algorithmic enhancement, but were substantially disregarded. In this work we address the general MOSPP with any number of sum and bottleneck objectives, and devise a label setting algorithm based on a remarkably simple, yet very effective aggregate order. Computational results show that, with a suitable choice of the aggregate function, the aggregate order is up to three times faster than the lexicographic one.

2   Label setting MOSPP algorithms

As in the classical Dijkstra approach, the label setting algorithms extend the paths from the source to the rest of the network by labeling the vertices. For a given s ($s \leq k$) assume that $[f_1^q, f_2^q, \ldots, f_s^q]$ are sum objectives, and $[f_{s+1}^q, f_{s+2}^q, \ldots, f_k^q]$ are bottleneck objectives. In the following we denote by $P_q(i) \oplus (i,j)$ the path obtained by adding the arc (i,j) to path $P_q(i)$. Initially, only one label is defined, namely $\ell_1(\sigma) = [0, 0, \ldots, 0]$, and marked as *temporary*. In the classical implementation each iteration of a label setting algorithm consists of three main steps:

1. **selection:** select a temporary label $\ell_q(i)$ that is *lexicographically* minimal among all temporary labels, and mark it as *permanent*;

2. **propagation:** for each arc $(i,j) \in A$ create a new label, say $\bar{\ell}(j)$, consisting of the k costs corresponding to path $P_q(i) \oplus (i,j)$;

3. **dominance check:** if $\bar{\ell}(j)$ is dominated by one of the (permanent or temporary) labels of vertex j then delete $\bar{\ell}(j)$ and go to the next iteration. Otherwise add a new label, say, $\ell_r(j) = \bar{\ell}(j)$, to the labels of vertex j, and delete temporary labels of j dominated by $\ell_r(j)$.

The algorithm terminates when no further temporary label exists, and it can be modified to handle bottleneck objectives as discussed in [3].

An aggregate approach can be obtained by replacing the order used in the selection step. Basically, this replacement consists in adding to each label $\ell_q(i) = [f_1^q, f_2^q, \ldots, f_k^q]$ an additional aggregate information, say, f_{k+1}^q, defined by a linear combination of the values $f_1^q, f_2^q, \ldots, f_k^q$ of path $P_q(i)$, and selecting, at each iteration, a temporary label for which f_{k+1}^q is a minimum. In the choice of the weights of the linear combination, our goal was to give each objective the same "importance", i.e., the same expected impact on the aggregate information. To this aim we define, for each objective h ($h = 1, 2, \ldots, k$), the average arc cost $\bar{c}_h = \sum_{(i,j) \in A} c_h(i,j)/m$. The aggregate information is then obtained by a linear combination of the normalized objective values, i.e.,

$$f_{k+1}^q = \alpha \sum_{h=1}^{s} \frac{f_h^q}{\bar{c}_h} + \beta \sum_{h=s+1}^{k} \frac{f_h^q}{\bar{c}_h},$$

where α (resp. β) is a positive weight assigned to the sum (resp. bottleneck) objective functions. Choosing a weight $\beta > \alpha$ allows to give bottleneck and sum objectives a similar importance. After the outcome of extensive computational experiments, not reported here, we set $\alpha = 1$ and $\beta = 10$.

3 Computational comparisons

We report here the results of an experimental comparison of the lexicographic and aggregate algorithms. We tested the algorithms on the set of random instances introduced in [3], which is the only benchmark specifically addressing bottleneck objectives proposed so far. For different numbers of vertices n, the graphs are generated according to different percentage densities $d = m/(n(n-1))$. The following data sets are considered: $n \in \{50, 100, 200\}$; $d \in \{5\%, 10\%, 20\%\}$. For each pair (n, d), three cost types C are tested. For each triple (n, d, C) and pair (s, b) ten instances are given. We refer to [3] for further details. In the following we denote the number of bottleneck objective functions as b ($= k - s$). We considered seven pairs (s, b), with $s, b \in \{1, 2, 3\}$ and $k = s + b \in \{3, 4, 5\}$, obtaining a total of 1890 instances. Algorithms were coded in C++. Computational experiments were performed on a Pentium IV PC with 3 GHz and 2 GB RAM.

We summarize the results in Table 1. The entries are here the average values computed over the 270 instances generated for each case. Column $|PS|$ gives the size of the Pareto set, i.e., the maximal complete set. The next two groups of four columns report the results obtained by the two algorithms: *sec* is the average CPU time in seconds, $\%rate$ is the average percentage of temporary labels that become permanent, $\frac{temp}{10^3}$ (resp. $\frac{perm}{10^3}$) are the average numbers of comparisons between the generated labels and the temporary (resp. permanent) labels, expressed in thousands.

Table 1 allow us to draw some remarks on our label setting algorithms. Observe that the aggregate version gives a quite similar reduction (more than 2/3 in the best cases) both in the CPU times and in the number of comparisons to permanent labels, even if the latter is slightly more relevant. Due to the huge number of label comparisons performed by both algorithms, it is

Table 1: Lexicographic and Aggregate algorithms: summary.

			Lexicographic				Aggregate			
s	b	$|PS|$	sec	$\%rate$	$\frac{temp}{10^3}$	$\frac{perm}{10^3}$	sec	$\%rate$	$\frac{temp}{10^3}$	$\frac{perm}{10^3}$
2	1	4,837	0.3	85.0	88	1,569	0.2	89.8	72	733
3	1	11,949	2.0	89.7	445	8,988	1.0	95.6	456	3,021
1	2	10,107	1.2	85.8	223	5,587	0.8	86.9	202	2,711
2	2	30,556	11.1	90.8	1,856	49,122	4.8	95.3	2,073	16,856
3	2	63,385	50.1	93.4	8,629	212,007	17.4	98.1	11,395	58,090
1	3	84,986	87.8	91.6	9,730	381,308	36.4	93.0	11,759	148,538
2	3	213,47	515.8	94.6	64,904	2,194,421	185.0	97.7	91,478	712,864
aver.		59,899	95.5	90.1	12,268	407,572	35.1	93.8	16,776	134,687

conceivable that the CPU time reduction obtained by the aggregate version is explained by the corresponding reduction in the number of label comparisons. Note that in our implementation, and for both algorithms, new labels are compared to permanent labels following a *temporal* order, i.e., in the order in which labels became permanent, which is clearly different for the lexicographic and the aggregate version. The aggregate order is much more effective, as rejected new labels tend to be dominated by "older" permanent labels, i.e., labels with a smaller aggregate weight.

References

[1] L. de Lima Pinto, C.T. Bornstein, and N. Maculan. The tricriterion shortest path problem with at least two bottleneck objective functions. *European Journal of Operational Research*, 198:387–391, 2009.

[2] E. Erkut. Editorial: Introduction to the special issue. *Computers & Operations Research*, 34:1241–1242, 2007.

[3] X. Gandibleux, F. Beugnies, and S. Randriamasy. Martins algorithm revisited for multi-objective shortest path problems with a maxmin cost function. *4OR*, 4:47–59, 2006.

[4] N. Jozefowiez, F. Semet, and Talbi. E.-G. Multi-objective vehicle routing problems. *European Journal of Operational Research*, 189:293–309, 2008.

[5] A. Raith and M. Ehrgott. A comparison of solution strategies for biobjective shortest path problems. *Computers & Operations Research*, 36:1299–1331, 2009.

[6] L. Santos, J. Coutinho-Rodrigues, and J.R. Current. An improved solution algorithm for the constrained shortest path problem. *Transportation Research Part B*, 41:756–771, 2007.

The Profitable Capacitated Rural Postman Problem

Stefan Irnich

Chair of Logistics Management, Mainz School of Management and Economics

Johannes Gutenberg University, D-55099 Mainz, Germany

Email: irnich@uni-mainz.de

1   Introduction

Single-vehicle arc-routing problems a.k.a. postman problems are among the oldest and best studied discrete optimization problems (see Dror, 2000; Eiselt et al., 1995a,b). The talk presents a non-standard extension of classical postman problems, i.e., the *Profitable Capacitated Rural Postman Problem (PCRPP)*. The PCRPP can be characterized by non-connected (postal) delivery regions, where it is possible to select the street segments that are serviced. Moreover, the overall duration of the postman tour is bounded.

Traditional exact methods for the Capacitated Arc Routing Problem (CARP) either rely on branch-and-cut (Belenguer and Benavent, 2003) or on the transformation into the corresponding node-routing problem (see, e.g., Baldacci and Maniezzo, 2006; Longo et al., 2006), the well-known Vehicle-Routing Problem (VRP). An alternative exact approach is the solution of the CARP by column generation (Desaulniers et al., 2005) or Lagrangian-relaxation (Lemaréchal, 2001). Letchford and Oukil (2009) follow this idea (they refer to the conference presentation Gomez-Cabrero et al. (2005) as the first column generation algorithm for the CARP). In essence, Letchford and Oukil (2009) price out elementary routes using a MIP formulation with directed flow variables. In contrast, we propose solving the subproblem *directly* as an undirected postman problem, but still guarantee that routes are elementary in the sense that no required edge is serviced twice. In fact, the subproblem is the PCRPP and we propose its solution by branch-and-cut, instead of transforming it into a node-routing problem and applying corresponding node-routing methods.

2   Definition of the PCRPP

We consider the PCRPP over an undirected graph $G = (V, E)$. The startpoint and endpoint of a postman tour is the depot $d \in V$. Edges $e \in E$ that are serviced generate a profit p_e, but traversing an edge costs c_e. Traversing an edge without servicing it is called deadheading. Both service and

deadheading consume time. Let q_e be the time for service and r_e be the time for deadheading through edge e. The task is to find a postman tour that maximizes the difference of profits and costs, while the overall duration of the tour must not exceed a given bound Q. Note that one can earn the profit p_e only once (when first traversing an edge). Furthermore, providing no service to an edge but deadheading through it can make sense: due to the maximum tour duration Q it might be better to not serve an edge, but to deadhead through that edge in order to reach other (profitable) edges.

A straightforward model for the PCRPP uses decision variables $x = (x_e)_{e \in E} \in \{0,1\}^{|E|}$ to indicate that edges are serviced with (not necessarily positive) contribution margin $\phi = (\phi_e) \in \mathbb{R}^{|E|}$ (profit minus traversal cost; $\phi_e = p_e - c_e \in \mathbb{R}$). Decision variables $y = (y_e)_{e \in E} \in \mathbb{Z}_+^{|E|}$ indicate the deadheading through edges with costs $c = (c_e)_{e \in E} \in \mathbb{R}_+^{|E|}$. It is easy to see that in an optimal solution the variables y_e can only take the values 0, 1, and 2.

$$z_{PCRPP} = \max \phi^\top x - c^\top y \tag{1a}$$

$$\text{s.t.} \quad x(\delta(i)) + y(\delta(i)) = 2w_i \qquad \text{for all } i \in V \tag{1b}$$

$$x(\delta(S)) + y(\delta(S)) \geq 2x_e \qquad \text{for all } e \in E, S \subseteq V \setminus \{d\} : e \subseteq S \tag{1c}$$

$$q^\top x + r^\top y \leq Q \tag{1d}$$

$$x \in \{0,1\}^{|E|} \tag{1e}$$

$$y \in \{0,1,2\}^{|E|} \qquad \text{for all } e \in E \tag{1f}$$

$$w \in \mathbb{Z}_+^{|V|} \tag{1g}$$

The objective (1a) is the maximization of the contribution margin, i.e., profit generated from services minus costs for traversals. Equalities (1b) use additional integer variables w_i for each node $i \in V$ in order to ensure that in the tour every node has an even degree. Connectivity of the tour with the depot results from the connectivity constraints (1c). Constraint (1d) ensures that the tour duration does not exceed the upper bound Q.

The PCRPP is an extension of the so-called Prize-collecting Rural Postman Problem (PRPP) introduced and solved with branch-and-cut in the work of Araoz et al. (2007). However, there are several important differences between the PCRPP and PRPP: First, Araoz et al. (2007) show that variables for deadheading can be restricted to take only values 0 and 1 in the PRPP. This is not true for the PCRPP. Second, Araoz et al. (2007) consider connected components built by the edges $e \in E$ with nonnegative contribution margin when traversed twice, i.e., with $\phi_e - c_e > 0$. They prove that these connected components can be treated similar as the components of required edges in in the undirected rural postman problem (URPP) (cf. Corberán and Sanchis, 1994). In this way Araoz et al. (2007) are able to come up with a pure binary programm with reduced variable set and additional constraints to strengthen the formulation.

We follow another idea, already used by (Ghiani and Laporte, 2000) for the URPP: variables

$y_e \in \{0, 1, 2\}$ can be replaced by two binary variables $y_e, z_e \in \{0, 1\}$. The novelty of the following formulation for the PCRPP is that with these binary variables y_e and z_e the node degree constraints can be fully replaced by cocircuit-inequalities:

$$z_{PCRPP} \quad = \quad \max \phi^\top x - c^\top(y + z) \tag{2a}$$

$$\text{s.t.} \quad x(\delta(S) \setminus F) + y(\delta(S) \setminus G) + z(\delta(S) \setminus G) - x(F) - y(G) - z(H)$$
$$\geq 1 - |F| - |G| - |H|$$
$$\text{for all } S \subseteq V; F, G, H \subseteq \delta(S) \text{ with } |F| + |G| + |H| \text{ odd} \tag{2b}$$

$$x(\delta(S)) + y(\delta(S)) + z(\delta(S)) \geq 2x_e$$
$$\text{for all } e \in E, S \subseteq V \setminus \{d\} : e \subseteq S \tag{2c}$$

$$q^\top x + r^\top(y + z) \leq Q \tag{2d}$$

$$y \geq z \tag{2e}$$

$$x \in \{0, 1\}^{|E|} \tag{2f}$$

$$y, z \in \{0, 1\}^{|E|} \tag{2g}$$

The advantage of this model compared to (1a)-(1g) is that the cocircuit-inequalities (2b) give a tight description of the underlying integer polyhedron. Constraints (2e) are added in order to reduce the inherent symmetry.

3 Solution Methods

We propose solving model (2a)-(2g) with branch-and-cut. The separation of violated cocircuit inequalities can be done with fast exact algorithm by Letchford et al. (2004). Violated connectivity constraints (2c) are easy to find (see Araoz et al., 2007) by solving a sequence of max-flow/min-cut problems. Several other classes of valid inequalities are known, e.g., cover inequalities (related to (2d)), valid inequalities from the TSP etc. Preliminary computational results of a branch-and-cut implementation are very promising. We are convinced that the branch-and-cut algorithm for the PCRPP will be the key component when solving instances of the CARP using column generation or Lagrangian relaxation.

A fine-tuned branch-and-cut implementation is currently under development. The talk at TRISTAN 7 will discuss the most recent version of this implementation and give answers to the following research questions:

1. What is a good (efficient, fast) separation strategy when solving the PCRPP via branch-and-cut? Similar to the study on the elementary shortest-path problem with capacity constraints performed by Jepsen et al. (2008), we will analyze the impact of different classes of valid

inequalities, of the sequence of separation algorithms, number of cuts to add per iteration, and thresholds to use for selecting violated inequalities.

2. What characterizes CARP instances to be well-suited for being solved with a column-generation or Lagrangian-relaxation approach? In which cases works the branch-and-cut solution approach for the subproblem better than traditional solution approaches?

References

Araoz, J., Fernandez, E., and Meza, O. (2007). Solving the prize-collecting rural postman problem. Research Report DR-2007/12, EIO Departament, Technical University of Catalonia, Spain.

Baldacci, R. and Maniezzo, V. (2006). Exact methods based on node-routing formulations for undirected arc-routing problems. *Networks*, 47(1):52–60.

Belenguer, J. and Benavent, E. (2003). A cutting plane algorithm for the capacitated arc routing problem. *Computers & Operations Research*, 30:705–728.

Corberán, A. and Sanchis, J. (1994). A polyhedral approach to the rural postman problem. *European Journal of Operational Research*, 79:95–114.

Desaulniers, G., Desrosiers, J., and Solomon, M., editors (2005). *Column Generation.* Springer, New York, NY.

Dror, M., editor (2000). *Arc Routing: Theory, Solutions and Applications.* Kluwer, Boston.

Eiselt, H., Gendreau, M., and Laporte, G. (1995a). Arc routing problems, Part I: The chinese postman problem. *Operations Research*, 43(2):231–242.

Eiselt, H., Gendreau, M., and Laporte, G. (1995b). Arc routing problems, Part II: The rural postman problem. *Operations Research*, 43(3):399–414.

Ghiani, G. and Laporte, G. (2000). A branch-and-cut algorithm for the undirected rural postman problem. *Mathematical Programming*, 87(3):467–482.

Gomez-Cabrero, D., Belunguer, J., and Benavent, E. (2005). Cutting planes and column generation for the capacitated arc routing problem. presented at ORP3, Valencia, Spain.

Jepsen, M., Petersen, B., and Spoorendonk, S. (2008). A branch-and-cut algorithm for the elementary shortest path problem with a capacity constraint. Technical Report 08/01, Dept. of Computer Science, University of Copenhagen, Denmark.

Lemaréchal, C. (2001). Lagrangian relaxation. In Jünger, M. and Naddef, D., editors, *Computational Combinatorial Optimization,*, volume 2241 of *Lecture Nodes in Computer Science*, pages 112–156. Springer, Berlin.

Letchford, A., Reinelt, G., and Theis, D. (2004). A faster exact separation algorithm for blossom inequalities. In Nemhauser, G. and Bienstock, D., editors, *Integer Programming and Combinatorial Optimization*, volume 3064, chapter 10. Springer, Berlin.

Letchford, A. N. and Oukil, A. (2009). Exploiting sparsity in pricing routines for the capacitated arc routing problem. *Computers & Operations Research*, 36(7):2320–2327.

Longo, H., de Aragao, M., and Uchoa, E. (2006). Solving capacitated arc routing probelms using a transformation to the CVRP. *Computers & Operations Research*, 33(6):1823–1837.

Online TSP and Hamiltonian Path Problems with Acceptance/Rejection Decisions*

Patrick Jaillet

Department of Electrical Engineering and Computer Science

Laboratory for Information and Decision Systems & Operations Research Center

MIT, 77 Massachusetts Ave, Cambridge, MA 02139, USA

Email: jaillet@mit.edu

Xin Lu

Operations Research Center

MIT

March 2010

1 Introduction and Problem Definitions

The context of this paper deals with routing and scheduling problems under incomplete and uncertain data, and in some cases under short time requirements for some of the decisions. Here we want to consider these questions around generalizations of the Traveling Salesman Problem (TSP) and Hamiltonian Path Problem (HPP).

Specifically, we are concerned with online versions of the TSP and HPP on metric spaces where the server doesn't have to accept all requests. Associated with each request (to visit a point in the metric space) is a penalty (incurred if the request is rejected) and a weight (collected if the request is accepted and the point visited). Requests are revealed over time to a server, initially at a given origin, who must decide which requests to serve in order to minimize the time to serve all accepted requests plus the sum of the penalties associated with the rejected requests while collecting enough weights to exceed a given quota. In the first online version of these problems, we assume that the server's decision to accept or reject a request can be made any time after its release date. In the

second version we assume that the server's decision to accept or reject a request must be made exactly at its release date. Formal problem definitions are as follow:

TSP WITH ACCEPTANCE/REJECTION DECISIONS:

Instance: A metric space $\mathcal{M}$ with a given metric d and an origin O. A series of n requests $(l_i, r_i, p_i, w_i)_{i \in \mathcal{N}}$ where $\mathcal{N} = \{1, \ldots, n\}$ is the index set of the requests, $l_i \in \mathcal{M}$ is the location (point in metric space) of request i, $r_i \in \mathbb{R}_+$ its release date (first time after which service can be done), $p_i \in \mathbb{R}_+$ its penalty (for not being served), and $w_i \in \mathbb{R}_+$ its weight (collected if served). A parameter $W_{min} \in \mathbb{R}_+$ (quota of total weight to collect). The problem begins at time 0; the server is initially idle at the origin (initial state), can travel at unit speed (when not idle), and eventually must be back and idle at the origin (final state). The earliest time the server reaches this final state is called the makespan.

Feasible solutions: Any subset $\mathcal{S} \subset \mathcal{N}$ of requests to be served, and a feasible TSP tour with release dates $\tau(\mathcal{S})$ through $\mathcal{S}$ so that $\sum_{i \in \mathcal{S}} w_i \geq W_{min}$.

Offline context: Requests are all revealed to the offline server at time 0.

Online context: Requests are revealed to the online server at their release dates $r_i \geq 0$; assume $r_1 \leq r_2 \cdots \leq r_n$. There are two online versions:

Basic: The online server can accept or reject a request any time after the request's release date.

Real-time: The online server must accept or reject a request immediately at the time of the request's release date. Decisions are then final.

Cost Function: In all cases, minimize {the makespan to serve all accepted requests + the sum of the penalties of all rejected requests} among all feasible solutions.

The HPP WITH ACCEPTANCE/REJECTION DECISIONS is the same problem without requiring the server to return to the origin. In all cases, the **goal** is to find online algorithms that minimize the competitive ratio, which is defined as max{online cost/ offline cost} over all instances.

2 Results

Full descriptions and proofs of results obtained on the TSP are contained in [7]. For the basic version of the problem in a general metric space, we construct an optimal 2-competitive online algorithm, improving the 7/3-competitive online algorithm of [3], and the 2.28-competitive online algorithm of [8]. For the real-time version, we analyze the problem in details for the case with no quotas. We first prove the optimality of a 2.5-competitive polynomial time online algorithm on the non-negative real line. We then provide a 3-competitive online algorithm on the real line and prove a general lower bound of 2.64 and a tighter lower bound of 2.73 among a restricted family of online algorithms. Finally we show that there can't be any finite c-competitive online algorithm on

a general metric space. We show a $\Omega(\sqrt{\ln n})$ lower bound on any competitive ratios, and describe an asymptotically optimal $O(\sqrt{\ln n})$-competitive online algorithm.

With respect to the HPP, we have recently developed a 2-competitive online algorithm on the non-negative line, which, as a by-product, improves upon the 2.06-competitive algorithm of [11] (which was the best result so far for the problem without rejections).

3 Literature review

Research concerning online versions of the TSP have been recent but have been growing steadily. Most related in spirit with this paper is the stream of works which started with the paper by Ausiello et al. [5], in which the authors study an online version of the TSP with release dates (without acceptance/rejection decisions); they analyze the problem on general metric spaces giving an optimal online algorithm with a competitive ratio of 2. They also provide a polynomial-time online algorithm, for general metric spaces, which is 3-competitive. Subsequently, Ascheuer et al. [2] give a 2.65-competitive polynomial-time algorithm, for general metric spaces and a $(2 + \epsilon)$-competitive ($\epsilon > 0$) algorithm for Euclidean spaces. Lipmann [11] develops an optimal online algorithm for the real line, with a competitive ratio of 1.64. Blom et al. [6] give an optimal online algorithm for the non-negative real line, with a competitive ratio of $\frac{3}{2}$. Jaillet and Wagner [9] introduce the notion of a disclosure date, and quantify the improvement in competitive ratios as a function of the advanced notice. A similar approach was taken by Allulli et al. [1] in the form of a *lookahead*. Jaillet and Wagner [10] consider the (1) online TSP with precedence and capacity constraints and the (2) online TSP with m salesmen and give for both problems a 2-competitive online algorithms (optimal in case of the m-salesmen problem). They also study online algorithms from an asymptotic point of view, and show that, under general stochastic structures for the problem data, *unknown and unused by the online player*, the online algorithms are almost surely asymptotically optimal.

Ausiello et al. [4] analyze the online Quota TSP, where each city to be visited has a weight associated with it and the server is given the task to find the shortest sub-tour through cities in such a way to collect a given quota of weights by visiting the chosen cities. They present an optimal 2-competitive algorithm for a general metric space. In Ausiello et al. [3], the authors give a 7/3-competitive algorithm and a lower bound of 2 for the "prize-collecting TSP" in a general metric space, a generalization of the quota problem where penalties for not visiting cities are also included. In Jaillet and Lu [8], we consider the "TSP with flexible service", a special case of the online prize-collecting TSP with no quotas. On the half-line and real line, we provide optimal 2-competitive online algorithms, and a c-competitive online algorithm, where $c = \frac{\sqrt{17}+5}{4} \approx 2.28$ for the general metric space.

References

[1] L. Allulli, G. Ausiello, and L. Laura. On the power of lookahead in on-line vehicle routing problems. In *Proceedings of the Eleventh International Computing and Combinatorics Conference, Lecture Notes in Computer Science*, volume 3595, pages 728–736, 2005.

[2] N. Ascheuer, S. Krumke, and J. Rambau. Online dial-a-ride problems: Minimizing the completion time. In *Proceedings of the 17th International Symposium on Theoretical Aspects of Computer Science, Lecture Notes in Computer Science*, volume 1770, pages 639–650, 2000.

[3] G. Ausiello, V. Bonifaci, and L. Laura. The on-line prize-collecting traveling salesman problem. *Information Processing Letters*, 107(6):199–204, 2008.

[4] G. Ausiello, M. Demange, L. Laura, and V. Paschos. Algorithms for the on-line quota traveling salesman problem. *Information Processing Letters*, 92(2):89–94, 2004.

[5] G. Ausiello, E. Feuerstein, S. Leonardi, L. Stougie, and M. Talamo. Algorithms for the on-line travelling salesman. *Algorithmica*, 29(4):560–581, 2001.

[6] M. Blom, S.O. Krumke, W.E. de Paepe, and L. Stougie. The online TSP against fair adversaries. *INFORMS Journal on Computing*, 13(2):138–148, 2001.

[7] P. Jaillet and X. Lu. Online traveling salesman problems with rejection options. Working paper, Operations Research Center, Massachusetts Institute of Technology, July 2009.

[8] P. Jaillet and X. Lu. Online traveling salesman problems with service flexibility. Working paper, Operations Research Center, Massachusetts Institute of Technology, January 2009.

[9] P. Jaillet and M. Wagner. Online routing problems: value of advanced information as improved competitive ratios. *Transportation Science*, 40(2):200–210, 2006.

[10] P. Jaillet and M. Wagner. Generalized online routing: New competitive ratios, resource augmentation and asymptotic analyses. *Operations Research*, 56:745–757, 2008.

[11] M. Lipmann. *On-line Routing*. PhD thesis, Technische Universiteit Eindhoven, 2003.

A new formulation for the 2-echelon capacitated vehicle routing problem

Mads Jepsen

DTU Management Engineering

Technical University of Denmark

Stefan Røpke

DTU Transport

Technical University of Denmark

Simon Spoorendonk

DTU Management Engineering

Technical University of Denmark

Email: spoo@man.dtu.dk

1   Introduction

The 2-echelon capacitated vehicle routing problem (2E-CVRP) is a transportation and distribution problem and can be described as follows: Given a depot, a set of satellites, and a set of customers each with a demand, our task is to distribute goods from the depot to the customers. Goods can be transported to the satellites before being shipped to the customers vehicles. Two sets of vehicle types are considered, a large capacity vehicle type going between the depot and the satellites and a small capacity vehicle type servicing the customers from the depot and the satellites. Each customer must be visited by exactly one small vehicle. It is optional to use a satellite and vehicles from the depot may deliver split deliveries to the satellites. No vehicle may be loaded so that the vehicle capacity is exceed. The number of vehicles to be used of each type is bounded, and the number of smaller vehicles used at each satellite is also bounded. The objective is to minimize the travel costs of the vehicles and the transshipment costs at the satellites.

The 2E-CVRP is relevant in city-logistic applications. Due to legal restrictions it may not be feasible to use large trucks within the center of large cities. Therefor, to distribute goods efficiently it is convenient to a use two-tier distribution network as in the 2E-CVRP where satellite facilities are located at the outskirts of the city.

The literature on 2E-CVRP is sparse, but in recent years some papers have been presented. Gonzales Feliu et al. [3] and Perboli et al. [4] present a multi-commodity flow inspired mathematical formulation for the 2E-CVRP. They are able to solve instances with up to 32 customers and 2

satellites to optimality. Perboli et al. [4] also present two heuristic methods that can find feasible solution to instances up to 50 customers and 4 satellites. Crainic et al. [1] present a clustering heuristic that can find solutions to instances up to 150 customers and 3 satellites. Crainic et al. [2] analyses the relationship between customers and satellite layouts and present heuristic solutions for instances with up to 200 customers. Perboli et al. [5] present several valid inequalities for the 2E-CVRP.

The contribution of this work is a new mathematical formulation for the 2E-CVRP inspired by the edge flow formulation of the CVRP. The new formulation have much fewer variables than the previously proposed but does have several constraint sets of exponential size. It is shown that some of the cutting planes from [4, 5] is redundant in the new model and that many symmetries regarding the assignment of customer sets to satellites is no longer an issue. This implies the strength of the new formulation. A branch-and-cut algorithm is developed to solve this model to optimality.

2   Mathematical Model

Define the graph $G = (V_0 \cup V_S \cup V_C, E \cup E')$ where $V_0 = \{0\}$ is the depot, V_S is the satellites, V_C is the customers, E is the edges in the first echelon and E' is the edges in the second echelon. Let c_e and c'_e be the travel cost of the edges in the first and second echelon respectively. The size of the two fleets associated with the echelons are given as K and K' with capacity C and C' respectively. K'_i is the maximum of second echelon vehicles serviced at satellite i. Let d_i be the demand of customer i and let h_i be the unit cost for transshipment at satellite i. Variables x_e indicate the use of edges in E and edges connected to the depot can be used twice. Variables y_e indicate the use of edges in E', and if a customer is connected to a satellite on a one-route via edge e then variable z_e indicates the return to the satellite. Variables y_i indicate the use of satellite i with the number of visits from the depot that is given by integer variables x_i. Let l_i be the quantity delivered to satellite i from the depot. Let $r(S)$ be the minimum number of second echelon vehicles needed to service $S \subseteq V_C$. The model is as follows:

$$\min \sum_{e \in E} c_e x_e + \sum_{e \in E'} c'_e y_e + \sum_{e \in \delta(V_S)} c'_e z_e + \sum_{i \in V_S} h_i l_i \tag{1}$$

$$\sum_{e \in \delta(i)} x_e = 2x_i \qquad\qquad i \in V_S \tag{2}$$

$$\sum_{e \in \delta(V_0)} x_e \leq 2K \tag{3}$$

$$l_i \leq \min\left\{ \sum_{j \in V_S} d_j, CK \right\} y_i \qquad\qquad i \in V_S \tag{4}$$

$$\sum_{i \in V_0 \cup V_S} l_i = \sum_{i \in V_C} d_i \tag{5}$$

$$\sum_{e \in \delta(S)} x_e \geq \frac{2}{C} \sum_{i \in S} l_i \qquad\qquad S \subset V_S, |S| \geq 2 \qquad (6)$$

$$y_i \leq x_i \qquad\qquad i \in V_S \qquad (7)$$

$$\sum_{e \in \delta(i)} (y_e + z_e) \leq 2K'_i y_i \qquad\qquad i \in V_S \qquad (8)$$

$$\sum_{e \in \delta(i)} y_e + \sum_{e \in E(i:V_S)} z_e = 2 \qquad\qquad i \in V_C \qquad (9)$$

$$\sum_{e \in \delta(S)} y_e + \sum_{e \in E(S:V_S)} z_e \geq 2r(S) \qquad\qquad S \subset V_C, |S| \geq 2 \qquad (10)$$

$$\sum_{e \in \delta(S \cup \{j\})} y_e + \sum_{e \in E(S:V_S \setminus \{j\})} z_e \geq \frac{2}{C'} \left(\sum_{i \in S} d_i - l_j \right) \qquad S \subset V_C, j \in V_0 \cup V_S \qquad (11)$$

$$\sum_{e \in \delta(S)} y_e \geq 2 \sum_{e \in F} y_e \qquad\qquad S \subseteq V_C, F \subseteq \delta(V_S),$$

$$\qquad\qquad |F \cup \delta(i)| \leq 1, i \in V_S \qquad (12)$$

$$z_e \leq y_e \qquad\qquad e \in \delta(V_0 \cup V_S) \qquad (13)$$

$$\sum_{i \in V_S} \left(\sum_{e \in \delta(i)} (y_e + z_e) \right) \leq K' \qquad\qquad\qquad (14)$$

$$x_e \in \{0, \ldots, K\} \qquad\qquad e \in E \setminus \delta(V_0) \qquad (15)$$

$$x_e \in \{0, \ldots, 2K\} \qquad\qquad e \in \delta(V_0) \qquad (16)$$

$$y_e \in \{0, 1\} \qquad\qquad e \in E' \qquad (17)$$

$$z_e \in \{0, 1\} \qquad\qquad e \in \delta(V_0 \cup V_S) \qquad (18)$$

$$x_i \in \{0, \ldots, K\}, y_i \in \{0, 1\} \qquad\qquad i \in V_S \qquad (19)$$

$$l_i \geq 0 \qquad\qquad i \in V_0 \cup V_S. \qquad (20)$$

The objective (1) minimizes travel cost and satellite operations. Regarding routing in the first layer we have: Constraints (2) ensure that satellites are left again if visited, (3) ensures that at most K vehicles leaves the depot, (4) ensure that a satellite needs to be visited in order to have goods delivered, (5) ensure that the quantity delivered for distribution is equal to customer demands, (6) ensure that the tours covering the satellites are connected, (7) ensure that a satellite cannot be used unless it is visited, and (8) limits the number of second echelon vehicles at the satellites. In the second layer we have: Constraints (9) ensure that all customers are visited, (10) ensure that the tours covering the customers are connected, (11) ensure that if a set of customers are connected to a satellite then the satellite have received enough quantity from the depot to service the customers, (12) ensure that a connected subset of customers are connected to the same satellite, and (13) ensure that an edge can only be used twice if it is a one-customer route. Constraint (14) ensures that at most K' vehicles are used to service the customers. Constraints (15)-(20) are the domain of the variables.

In the talk we will show how some valid inequalities from [4] and [5] are implied by the linear relaxation of (1)-(20). Compared to the previous presented model in [3, 4] symmetry is avoided both due to the undirected graph structure and because the assignment of customers to satellites are implicit, hence sub-routes in the second echelon cannot change assignment between satellites.

3   Solution Approach

We suggest to solve the model (1)-(20) by a branch-and-cut algorithm by relaxing the exponential number of constraints (6), (10), (11), and (12) and separate them when violated.

Constraints (6) are derived from the *fractional capacity inequalities* known from the CVRP, and the separation is polynomial and is done by solving $|V_S| - 1$ maxflow problems. Constraints (10) are the *capacity inequalities* known from the CVRP. Separation is $\mathcal{NP}$-hard if $r(S)$ is not the trivial fractional bound. Constraints (11) are equivalent to (6) but concerns the customer set V_C, hence separation is done by solving $|V_S||V_C|$ maxflow problems. The separation of constraints (12) is also polynomial solvable by solving $|V_S||V_C|$ maxflow problems.

References

[1] T.G. Crainic, S. Mancini, G. Perboli, and R. Tadei, "Clustering-Based Heuristics for the Two-Echelon Vehicle Routing Problem", Technical report, CIRRELT-2008-46, CIRRELT, Montreal, 2008.

[2] T.G. Crainic, S. Mancini, G. Perboli, and R. Tadei, "Two-Echelon Vehicle Routing Problem: A Satellite Location Analysis", Technical report, CIRRELT-2009-15, CIRRELT, Montreal, 2009.

[3] J. Gonzales Feliu, G. Perboli, R. Tadei and D. Vigo, "The Two-Echelon Capacitated Vehicle Routing Problem", Technical report DEIS OR.INGCE 2007/2(R), DEIS, Bologna, 2007

[4] G. Perboli, R. Tadei, and D. Vigo, "The two-echelon capacitated vehicle routing problem: Models and math based heuristics", Technical report CIRRELT-2008-55, CIRRELT, Montreal, 2008.

[5] G. Perboli, R. Tadei, and F. Masoero, "Models and cuts for the two-echelon vehicle routing problem", Proceedings of the International Network Optimization Conference 2009, Pisa, April 2009

A parallel granular Tabu search algorithm for large scale CVRP

Jianyong Jin[a], Arne Løkketangen[a], Teodor Gabriel Crainic[b]

[a]Molde University College, 6402 Molde, Norway

[b]Université du Québec à Montréal, local R-2380 Montréal QC Canada

1. Introduction

The vehicle routing problem (VRP) describes the allocation of transportation tasks to a fleet of vehicles, and the simultaneous routing of each vehicle. The VRP was first described by Dantzig and Ramser (1959), and has been proved NP-hard by Lenstra and Kan (1981). Due to its high industrial relevance and complexity, the VRP has been the object of numerous studies and a great number of papers have proposed solution methods.

The classical or capacitated VRP (CVRP) is defined on a graph G= (N, A) where N= {0,…, n} is a vertex set and A= {(i, j) : i, j ∈ N} is an arc set. Vertex 0 is the depot where the vehicles depart from and return to. The other vertices are the customers which have a certain demand d_i to be delivered (or picked up). The travel cost between customer i and j is defined by c_{ij} >0. The vehicles are identical. Each vehicle has a capacity of Q. The objective is to design a least cost set of routes, all starting and ending at the depot. The customers are visited exactly once. The total demand of all customers on a route must be within the vehicle capacity Q. Some CVRP instances may have an additional route duration limit which restricts the duration of any route does not exceed a preset bound D.

From the literature, one trend in the latest contributions of metaheuristic algorithms for solving vehicle routing problems is to address very large scale classical VRP instances. For example, Li et al. (2005) have applied record-to-record travel with a variable-length neighbor list to a set of instances up to 1200 customers. Kytojoki et al. (2007) present an efficient variable neighborhood search heuristic for CVRP and demonstrate the proposed method is able to solve problem instances with up to 20000 customers. These authors also generated a couple of sets of large CVRP instances.

Another noticeable trait for VRP solution methods is the application of parallelization. Parallel algorithms involve problem solving means that several (sometimes could be many as well)

processes work simultaneously on available processors (computers/workstations) with the common goal of solving a given problem instance. Crainic (2008) describes and discusses the main strategies used in this group of algorithms and also provides an up-to-date survey of contributions to this rapidly evolving field. In a latest instance, Dorronsoro et al (2007) presented a grid-based hybrid genetic algorithm for large scale instances of the CVRP. In their model, the population was divided into islands at the first level, and structured as cellular patterns at the second level. Periodically, each island exchanged individual solutions with its immediate neighbors. The proposed method was tested with the large scale CVRP instances presented by Li et al (2005). The testing platform was a grid composed of up to 125 heterogeneous computers using Proactive to manage all the grid related issues. Computational results showed that the proposed algorithm could find very good solutions for large CVRP instances but long computational time (from 10 hours to 72 hours) was required.

The main focus of this paper is to combine granular Tabu search (Toth and Vigo (2003)) and parallel computing technique for large scale CVRP for the purpose of solving them with high efficiency and flexibility. Computational experiments will be carried out on the large scale benchmark instances of Golden et al. (1998), Li et al. (2005) and Kytojoki et al. (2007).

2. The problem solving methodology

Intuitively, in order to find good solutions very quickly, one had better either reduce search tasks or use more computational power. Following such logic, we apply granular Tabu search in the parallelization setting. Granular Tabu search (Toth and Vigo (2003)) is a mechanism which is able to significantly reduce the computational efforts, especially for large problem instances by getting rid of the unpromising solution components. In this paper, a different granular neighborhood is implemented, which is to select a set of nearest neighbors (plus the depot) for each customer, and at each iteration, only moves involving one of the nearest neighbors will be considered. The size of the set of nearest neighbors is selected by considering the instance size and the requirement of solution quality (or the time available for computation) as suggested in Branchini et al (2009). In addition, to address large instances, several neighborhood structures will be used to increase the effectiveness and the efficiency of the algorithm.

Another aspect of the proposed metaheuristic is the application of parallelization. As suggested in Crainic (2008), cooperative multi-thread strategy and asynchronous information exchange through solution pool (like adaptive memory approach presented by Rochat and Taillard (1995)) are applied in the algorithm for the sake of the employment of more computational power.

3. Implementation issues

The solver will be implemented with C++ and Intel Threading building Blocks (TBB) libraries for parallelization. The computational experiments will be carried out on a computer with Intel® Xeon® E5450 3.00GHZ CPU and 8 GB of RAM.

4. The expected outcomes

In this paper, we are expecting the following outcomes or findings.

(1). Our proposed algorithm is capable of solving large scale CVRP instances very quickly.

(2). Discover the proper size for the set of nearest neighbours.

(3). Investigate the strategies of implementing multiple neighborhoods.

(4). Explore how parallelization can improve the performance of the algorithm.

References

Branchini, R.M., Armentanoa, V.A., and Løkketangen, A. (2009). Adaptive granular local search heuristic for a dynamic vehicle routing problem. Computers&OperationsResearch 36, 2955-2968.

Crainic, T.G. (2008). Parallel solution methods for vehicle routing problems. In B. Golden, S. Raghavan, and E. Wasil, editors, The Vehicle Routing Problem: Latest Advances and New Challenges, pages 171–198. Springer, New York.

Dantzig, G. B., and Ramser, J. H. (1959). The truck dispatching problem. Management Science, 6:80–91.

Dorronsoro, B., Arias, D., Luna, F., Nebro,A.J., and Alba, E. (2007). A grid-based hybrid cellular genetic algorithm for very large scale instances of the CVRP. High Performance Computing & Simulation Conference (HPCS), IEEE Press, Piscataway, NJ, 759-765.

Golden, B.L., Wasil, E.A. , Kelly, J.P. and Chao, I.-M. (1998), The impact of metaheuristics on solving the vehicle routing problem: algorithms, problem sets, and computational results. In: T. Crainic and G. Laporte, Editors, Fleet management and logistics, Kluwer, Boston, pp. 33–56.

Kytojoki, J., Nuortio, T., Bräysy, O., and Gendreau, M. (2007). An efficient variable neighborhood search heuristic for very large scale vehicle routing problems, Computers & Operations Research 34, 2743–2757.

Lenstra J.K., and Kan A.H.G. Rinnooy. (1981). Complexity of vehicle routing and scheduling problems. Networks 11: 221–227.

Li, F., Golden, B.L., and Wasil, E.A. (2005). Very large-scale vehicle routing: New test problems, algorithms, and results. Computers & Operations Research 32, 1165–1179.

Rochat, Y. and Taillard, E.D. (1995). "Probabilistic diversification and intensification in local search for vehicle routing". Journal of Heuristics 1, 147-167.

Toth, P. and Vigo, D. (2003), The granular tabu search and its application to the vehicle routing problem, INFORMS Journal on Computing 15, 333–346.

Intel Threading Building Blocks: http://www.threadingbuildingblocks.org/.

The multi-modal traveling salesman problem

Nicolas Jozefowiez

LAAS-CNRS, INSA, Université de Toulouse, France

Email: nicolas.jozefowiez@laas.fr

Gilbert Laporte

CIRRELT, HEC-Montréal, Canada

Email: gilbert.laporte@cirrelt.ca

Frédéric Semet

LAGIS, Ecole Centrale de Lille, France

Email: frederic.semet@ec-lille.fr

1 Problem definition and formulation

The paper introduces a new routing problem: the Multi-Modal Traveling Salesman Problem (MMTSP). The MMTSP is a bi-objective variant of the Traveling Salesman Problem (TSP) that can be defined as follows. Let $G = (V, E)$ be an undirected graph, where V is the vertex set and E is the edge set, and let C be a set of colours. Edge $e \in E$ has a cost c_e and a colour $\delta(e) \in C$. Colours can be seen as different means of transportation. The goal is to determine a Hamiltonian cycle of least length and least number of colours. A colour $k \in C$ is said to be used if an edge $e \in \zeta(k)$ belongs to the cycle with $\zeta(k) = \{e \in E | \delta(e) = k\}$.

The MMTSP can be formulated as an extention of the Dantzig, Fulkerson, and Johnson model for the TSP [2]. We define binary variables x_e, equal to 1 if and only if e is used and binary variables u_k, equal to 1 if and only if $k \in C$ is used. The problem is then:

$$\min \quad (\sum_{e \in E} c_e x_e, \sum_{k \in C} u_k) \tag{1}$$

$$\sum_{e \in \omega(\{i\})} x_e = 2 \qquad (i \in V) \tag{2}$$

$$\sum_{e \in \omega(S)} x_e \geq 2 \qquad (S \subset V, 3 \leq |S| \leq |V| - 3) \tag{3}$$

$$x_e \leq u_{\delta(e)} \qquad (e \in E) \tag{4}$$

$$x_e \in \{0, 1\} \qquad (e \in E) \tag{5}$$

$$u_k \in \{0, 1\} \qquad (k \in C), \tag{6}$$

where $\omega(S) = \{e = (i, j) \in E | i \in S, j \in V \setminus S\}$.

Additional valid inequalities can be used to strengthen the model. First, note that any valid inequality for the TSP is also valid for the MMTSP. Also, The following constraints are also valid:

$u_k \leq \sum_{e \in \zeta(k)} x_e$ $(k \in C)$. These constraints state that if a colour k is used, then at least one of the edges of colour k must be used. Finally, additional constraints can be derived from valid constraints for the TSP, using one of the following propositions.

Proposition 1.1 *Let T be a subset of E. If $\sum_{e \in T} \alpha_e x_e \leq \beta$ is valid for the TSP then for a colour $k \in C$, the inequality $\sum_{e \in T \cap \zeta(k)} \alpha_e x_e \leq \beta u_k$ is valid for the MMTSP.*

Proposition 1.2 *Let T be a subset of E. If $\sum_{e \in T} \alpha_e x_e \geq \beta$ is valid for the TSP then the inequality $\sum_{k \in C} \min\{\sum_{e \in T \cap \zeta(k)} \alpha_e, \beta\} u_k \geq \beta$ is valid for the MMTSP.*

Three single-objective problems were derived from the MMTSP. The first one, the Label Constraint Traveling Salesman Problem (LCTSP), deals with the goal of finding a minimal length tour that does not exceed a given number of colours. The second problem, the Minimum Labelling Hamiltonian Cycle with Distance Constraint (MLHCDP), is the reverse of the LCTSP, i.e. the goal is to find a Hamiltonian cycle using the minimum number of colour without exceeding a given distance. The last one is the Minimum Labelling Hamiltonian Cycle Problem (MLHCP) or Colourful Traveling Salesman Problem (CTSP) in which the goal is to find a Hamiltonian cycle minimizing the number of colours. The MLHCP has previously been solved heuristically in [1] and [5]. To our knowledge, The other problems have not been solved previously.

2 Branch-and-cut algorithm

Branch-and-cut algorithms were first developped for the single-objective variants of the problem. They use the inequalities defined in the previous section to strengthen the lower bound. Connectivity constraints are used as cuts. Tests were conducted on generated data and modified instances from the TSPLIB. However, the intent of this work was to solve the problem as a bi-objective problem and to find the optimal set of non-dominated solutions. Most exact algorithms for multi-objective combinatorial optimization problems consist one way or the other in a repeated application of a single-objective method. For example, a possibility is to use one of the branch-and-cut algorithms defined earlier in an ϵ-constraint method, as explained in [3]. As far as we know, only one reference [4] proposes an adaptation of a standard and generic branch-and-bound algorithm to multi-objective optimization. The method can find the optimal Pareto set in a single run. However, it cannot efficiently solve the multi-objective problem if the weighted aggregation of the objectives results in an NP-hard single-objective problem, as is the case of the MMTSP.

To avoid this difficulty, we have developed a generic multi-objective branch-and-cut algorithm (MOB&C). This was done with the aim of modifying as little as possible the generic structure of the standard branch-and-cut algorithm. At every node of the search tree, the method works on the full Pareto set. It can therefore be used heuristically as an anytime method that can be

stopped to return an approximation of the optimal Pareto set. In this sense, it can be compared with multi-objective evolutionary algorithms that work on a population of solutions and offer a well-diversified approximation. Such methods constitute the main class of metaheuristics for multi-objective problems.

Differences with the standard branch-and-cut algorithm appear in the definition of the lower and upper bounds, in how branching is performed, and in the presence of a mechanism to avoid useless computations. We define the lower bound as a set of points in the objective space such that any point corresponding to a feasible solution is dominated by at least one of these points. A point in the lower bound set does not necessarily correspond to a feasible solution. The upper bound is a set of points in the objective space corresponding to at least one non-dominated feasible solution found during the search. A node of the search tree can be pruned if all the points in the lower bound are dominated by at least one solution of the upper bound. For the MMTSP, we propose a polynomial algorithm to compute the lower bound. Concerning the branching, since the lower bound is composed of several solutions, it is probable that they will not all possess the same set of fractional variables. Therefore, we have implemented a process to allow branching to be done on multiple variables. The rationale for this choice is that since the lower bound is composed of a set of points, if there is a least one fractional and undominated point, then the node cannot be pruned, but the dominated and integer points will remain or will lead to dominated solutions, irrespective of how the branching is performed or which constraints are added. To avoid these useless computations, a mechanism was used to avoid computing solutions in aeras that are certain to be dominated under the current branching choices.

MOB&C was implemented for the MMTSP and compared on generated data with an ϵ-constraint method (ϵCM) [3] using a branch-and-cut algorithm for the LCTSP. To compare the two methods, aside from the specific aspects of MOB&C and some improvement mechanisms for ϵCM [3], the branch-and-cut characteristics (cuts, inital heuristic ...) are the same. Some results are reported in Table 1 (note that each line corresponds to a mean over 25 instances). The main conclusion is that MOB&C is faster on average than the ϵ-constraint method. The relative difference in performance increases with instance size (especially with the number of colours). Our algorithm also explores far fewer nodes than the ϵ-constraint method. Again, the difference increases with instance size, which suggests that MOB&C will be more efficient on larger instances.

As the optimal Pareto set can be reached well before the end of the search (column "Seconds*" in Table 1), tests were also conducted to evaluate the performance of MOB&C if it is stopped before the proof of optimality and used as a heuristic. It appears that it performs well. For instance, if we explore only 50 percent of the nodes, on average more than half of the optimal set is found. Also, the fact that the size of the approximation is almost the size of the optimal Pareto set and that the gap is less than one percent indicates that the approximation is close to the optimal Pareto set.

Table 1: Summary of computational results for the branch-and-cut algorithm.

			MOB&C				ϵCM		
$|C|$	$|V|$	#Par	#Nodes	#Cuts	Seconds	Seconds*	#Nodes	#Cuts	Seconds
40	30	17.8	1913.0	258.9	58.7	42.7	5806.0	273.4	67.2
40	40	21.7	4406.6	548.2	503.0	349.8	17462.0	578.3	665.8
40	50	26.6	15360.6	926.0	1845.9	1374.5	45306.6	1037.9	3334.5
50	30	18.8	3248.3	355.0	144.0	110.2	12687.6	428.1	224.9
50	40	23.9	8722.7	738.3	1374.4	1097.7	36339.4	797.5	1636.9
50	50	27.7	20680.3	1204.9	4094.0	2902.5	74336.6	1307.5	5938.4

Here, the gap is expressed as the average ratio between the length of the solution using c colours in the approximation and the length of the solution using c colours in the optimal Pareto set. If there is no solution using c colours in the approximation, we use the ratio between the length in the approximation and the closest smaller length in the optimal Pareto set.

References

[1] R. Cerulli, P. Dell'Olmo, M. Gentili, A. Raiconi, "Heuristic approaches for the minimum labelling Hamiltonian cycle problem", *Electronic Notes on Discrete Mathematics* 25, 131-138 (2006).

[2] G. Dantzig, D. Fulkerson, S. Johnson, "Solution of a large-scale traveling-salesman problem", *Operations Research* 2, 393-410 (1954).

[3] J.-F. Bérubé, M. Gendreau, and J.-Y. Potvin, "An Exact ϵ-Constraint Method for Bi-Objective Combinatorial Optimization Problems: Application to the Traveling Salesman Problem with Profits", *European Journal of Operational Research* 194, 39-50 (2009).

[4] F. Sourd and O. Spanjaard, "A Multi-Objective Branch-and-Bound Algorithm Framework. Application to the Bi-Objective Spanning Tree Problem", *INFORMS Journal on Computing* 20, 472-484 (2008).

[5] Y. Xiong, B. L. Golden, E. A. Wasil, "The colorful traveling salesman problem", in *Extending the horizons: Advances in Computing, Optimization, and Decision Techniques* 115-123, Springer, Boston (2007).

A BILEVEL PRICING PROBLEM WITH ELASTIC DEMAND

AIMÉ KAMGAING KUITEING

Department of Mathematics and Industrial Engineering

École Polytechnique de Montréal

PATRICE MARCOTTE

Department of Computer Science and Operations Research

Université de Montréal

GILLES SAVARD

Department of Mathematics and Industrial Engineering

École Polytechnique de Montréal

Email: aime.kamgaing-kuiteing@polymtl.ca

1   Introduction

Several extensions of the network pricing model introduced by Labbé et al. [2] have been proposed but none, to the best of our knowledge, involves elastic demand, which is the topic of this work. More specifically, we consider the problem of maximizing the revenue raised from tolls set on a multicommodity transportation network, taking into account that demand is assigned to cheapest paths, and is actually dependent on the total cost (initial cost of carrying the products + toll) of these paths.

This presentation is concerned with various formulations of the problem, either in arc or path flow space. In the case of a linear demand-price relationship, we propose three mixed integer (MIP) linear formulations. In the case of nonlinear demand functions, we develop one exact and two heuristic solution methods, and provide an upper bound that allows to assess the quality of the heuristic solutions.

2 Mathematical formulation of the problem

The problem is cast into the framework of a leader-follower game which takes place on a multi-commodity network $\mathcal{G} = (\mathcal{K}, \mathcal{N}, \mathcal{A})$ defined by a set of origin-destination couples (commodities) $\mathcal{K}$, a set of nodes $\mathcal{N}$ and a set of arcs $\mathcal{A}$. The latter is partitioned into a subset $\mathcal{A}_1$ of toll arcs which belong to the leader, and the complementary subset $\mathcal{A}_2$ of toll-free arcs. Each arc $a \in \mathcal{A}$ bears a fixed travel cost c_a. A toll arc $a \in \mathcal{A}_1$ involves a toll component t_a, to be determined. The demand for a commodity k is represented by the function $n_k(u_k)$, where u_k is the total cost to travel between the origin $o(k)$ and the destination $d(k)$.

Our formulation follows into the footsteps of that suggested in Didi et al. [3]. Specifically, we introduce the set $\mathcal{P}_k$ of paths from $o(k)$ to $d(k)$, and flow h_p^k on path $p \in \mathcal{P}_k$. The problem can then be formulated as a bilevel program involving bilinear objectives at both decision levels, and bilinear constraints at the lower level, i.e.:

$$\max_{t \geq 0, \ h' \geq 0} \sum_{k \in \mathcal{K}} \sum_{p \in \mathcal{P}_k} \sum_{a \in p} t_a h_p^{'k}$$

$$\forall \, k \in \mathcal{K} \begin{cases} h'^k \in \arg\min_{h^k \geq 0} \sum_{p \in \mathcal{P}_k} \left(\sum_{a \in p} c_a + \sum_{a \in p \cap \mathcal{A}_1} t_a \right) h_p^k \\ \text{s. t.} \quad \sum_{p \in \mathcal{P}_k} h_p^k = n_k(u_k) \\ \qquad u_k = \sum_{p \in \mathcal{P}_k} \left(\sum_{a \in p} c_a + \sum_{a \in p \cap \mathcal{A}_1} t_a \right) h_p^k. \end{cases}$$

Note that the lower level problem is separable by commodity and, since there are no capacities on the arcs of the network, that u_k corresponds to the cost of a cheapest path associated with commodity k. Next, one moves the demand function into the upper level objective, one forces flow variables to be binary, and one drops the constraints involving u_k. Finally, replacing the lower level by its primal-dual optimality condition, and linearizing the complementarity constraints in the classical fashion, we obtain a MIP that involves a nonlinear objective and linear constraints:

$$\text{PATH:} \quad \max_{t \geq 0, \ h \in \{0,1\}} \sum_{k \in \mathcal{K}} n_k(u_k) T_k$$

$$\forall \, k \in \mathcal{K} \begin{cases} T_k \leq \sum_{a \in p} t_a + M^k \left(1 - h_p^k\right) & \forall \, p \in \mathcal{P}_k \\ \sum_{a \in p} c_a + \sum_{a \in p} t_a - M_p^k \left(1 - h_p^k\right) \leq u_k \leq \sum_{a \in p} c_a + \sum_{a \in p \cap \mathcal{A}_1} t_a & \forall \, p \in \mathcal{P}_k \\ u_k = T_k + \sum_{p \in \mathcal{P}_k} h_p^k \sum_{a \in p} c_a \\ \sum_{p \in \mathcal{P}_k} h_p^k = 1. \end{cases}$$

where T_k represents one unit of the leader's revenue raised from commodity k.

3   Linear demand

Let us assume that, for each commodity k, demand assumes the linear form $n_k(u_k) = a_k - b_k u_k$, where a_k, b_k are positive constants. We rewrite the objective function of PATH for each commodity k as the quadratic term:

$$
n_k(u_k)T_k = a_k T_k - b_k T_k \left(T_k + \sum_{p \in \mathcal{P}_k} h_p^k \sum_{a \in p} c_a \right) = a_k T_k - b_k T_k^2 - b_k \sum_{p \in \mathcal{P}_k} T_k h_p^k \sum_{a \in p} c_a.
$$

Upon the introduction of a unit commodity path revenue T_p^k and the replacement of the bilinear terms $T_k h_p^k$ by terms T_p^k that we require to be equal whenever the associated flows h_p^k are positive, we obtain a mixed quadratic program that can be fed into an off-the-shelf software such as CPLEX.

4   Nonlinear demand

In the situation where demand is convex and decreasing, we may express the commodity revenue $n_k(u_k)T_k$ in the objective as the sum

$$
n_k(u_k)T_k = n_k(u_k) \left(u_k - \sum_{p \in \mathcal{P}_k} h_p^k \sum_{a \in p \bigcap \mathcal{A}_1} c_a \right) = n_k(u_k)u_k - n_k(u_k) \sum_{p \in \mathcal{P}_k} h_p^k \sum_{a \in p \bigcap \mathcal{A}_1} c_a.
$$

Next, we introduce new variables and constraints (cuts) that allow to linearize the terms $n_k(u_k)u_k$ and $n_k(u_k)$, yielding a MIP formulation. The latter is refined until the gap between its optimal value and an upper bound falls below a predetermined threshold. Along the lines of the framework proposed by Colson et al. [1], we also designed two heuristic procedures that combine the PATH formulation with a trust-region approach. More specifically, given a current solution (h^l, t^l, T^l) and a trust region radius, the first heuristic linearizes the demand function around the current iterate (h^l, t^l, T^l) and solves the resulting PATH within the trust region. Alternatively, the second heuristic linearizes the objective function (h^l, t^l, T^l). The stopping criterion and the update of the trust region radius are performed according to standard practice.

5   Numerical results

The algorithms have been tested on a range of randomly generated instances. Quite surprisingly, the linear demand model did not prove more challenging than the standard fixed demand model, the ratio of the Cpu times being bounded by a constant factor.

In the nonlinear case, the Cpu time required by the exact method grows rapidly with network size, resulting in the slow decrease of the upper bound. While it can solve to proven optimality on medium size problems within reasonable Cpu time, and that good solutions of large problems

can be found quickly, the poor quality of the upper bound precludes the determination of provably optimal solutions.

To overcome this difficulty, improved upper bounds are computed in the following fashion. We bound the commodity revenue $n_k(u_k)T_k$ in the objective by the expression $n_k \left(T_k + \min_{p \in \mathcal{P}_k} c_p \right) T_k$ and next change the objective function of the PATH formulation to $\sum_{k \in \mathcal{K}} n_k \left(T_k + \min_{p \in \mathcal{P}_k} c_p \right) T_k$. The resulting formulation is then solved through the incorporation of cuts that approximate the term $n_k \left(T_k + \min_{p \in \mathcal{P}_k} c_p \right) T_k$. While more expensive to compute, this bound yields a considerable decrease of the gap defined as the difference between the upper bound and the best feasible solution. It also allows to probe the performance of the heuristics which have been developed. Indeed, numerical tests showed that solutions whose objective lie within 1% of the upper bound could be achieved within reasonable Cpu times, even on large instances.

6   Conclusion and extensions

Although we have already shown that elastic network pricing problems are amenable to efficient solution procedures, there is still room for improvement. In particular, as was done in the fixed demand case, we will analyze the structure and properties of the inverse demand problem that corresponds in optimizing the revenue with respect to given path flows. If there is a single commodity, this reduces to the computation of a cheapest path in some modified graph. In the multicommodity case, its structure is that of a network-based quadratic and concave maximization problem. While its exact solution is costly, an approximate solution could be use to enhance our basic algorithmic scheme. If it proves successful, the approach could then be extended to the nonlinear demand case.

References

[1] B. Colson, P. Marcotte and G. Savard, "A trust-region method for nonlinear bilevel programming: algorithm and computational experience", *Computational Optimization and Applications* **30** (2005) 211–227.

[2] M. Labbé, P. Marcotte and G. Savard, "A Bilevel Model of Taxation and Its Application to Optimal Highway Pricing", *Management Science* **44** (1998) 1595-1807.

[3] M. Didi B., P. Marcotte and G. Savard, "Path-based Formulations of a Bilevel Toll Setting Problem", in Optimization with Multivalued Mappings Theory: Theory, Applications and Algorithms, S. Dempe and V. Kalashnikov eds., Springer (2006) 29-50

Service Oriented Train Timetabling

Mor Kaspi*, Tal Raviv

Department of Industrial Engineering

Tel Aviv University, Israel

Email*: morkaspi@post.tau.ac.il

1 Introduction and Problem Definition

Planning of railway operations has been practiced since the first trains started working early in the nineteenth century. Nowadays, the usage of train transportation is increasing and the expectation of the passengers for better service raises.

The train planning problem can be divided into several interrelated sub-problems, namely Line Planning – deciding which set of lines should be served by the system and in what frequencies subject to total demand for journeys and capacity constraints of trains and of the infrastructure and Train Timetabling – deciding upon the schedule of each train in each line subject to track availability and headway constraints. Other sub-problems that need to be addressed are Track Assignment, Platforming, Rolling Stock Circulation, and Crew Planning. Each of these sub-problems is computationally hard.

Although in practice, some of the planning work is still being done manually, in the last 40 years planners have been using Decision Support Systems and optimization methods in order to improve the quality of their plans and to save labor. Typically, the planning problem is solved hierarchically in a strategic order such that the solution of each sub-problem is used as input for the following problems. This solution strategy enables tackling real-world problems; the disadvantage is that the global optimal solution is lost "on the way." For a comprehensive survey of optimization models see [1] [2] [3].

The development of railway infrastructure takes years, or even decades; therefore, unlike in other transportation systems, the line planning stage is mainly to choose routes between terminals on a given infrastructure. In general, the objective of the Line Planning Process is to balance the tradeoff between operational costs and service quality. Bussieck et al. [4] suggests the minimization of *total travel time* of all passengers as a good way to model quality service. However, since the timetable at this stage of the planning is unknown, it is impossible to calculate the *total travel time*. Therefore, previous studies [4] [5] considered service level indirectly by minimizing the total number of transfers or maximizing the number of direct passengers. These objectives do not fully capture the service level. For example, longer lines with many stops may reduce the number of transfers but prolong the travel

time of some passengers. A partial remedy for this issue is found in [6] where actual time on board plus some arbitrary penalty for transfers is minimized.

For the Train Timetabling stage some studies focus on finding a feasible timetable [7] while others try to schedule as many trains as possible under a cost or profit criterion [8] [9] [10] [11]. In most studies, the demand pattern is ignored at this stage, assuming it was treated at the Line Planning Stage. However, timetabling considerations may have a significant impact on the quality of service. For a detailed survey of scheduling approaches see [12].

Common practice in passenger transportation is to use cyclic timetables. In a cyclic timetable, each trip is operated in a cyclic way. That is, each period of the timetable is the same. From the passengers' point of view, such timetables are more convenient because all they need to remember are the times in a cycle in which trains arrive at their station. From the planners' point of view, since each event occurs again every cycle, it is enough to plan one cycle, thus reducing the search space dramatically. A mathematical model for the *Periodic Event Scheduling Problem* (PESP) is developed in [13]. Our method follows similar ideas.

In our study, we focus on creating integrated Line Plans and Timetables with the objective of minimizing the total time that passengers spend in the system. This includes waiting time at station of origin, time on board the trains, and transfer times. The line planning component of our model is restricted to decisions on stopping stations and frequencies while a set of optional routes is given.

Total travel time had been recognized by previous researches, e.g., [4], as the correct measure for quality of service, but was never tackled in the litrature; probably since its proper optimization requires integration of the line planning and timetabling phases. This study aims to close this gap.

The problem that we formulated is solved using the Cross-Entropy (CE) meta-heuristic technique. See [14] [15] [16]. CE is a randomized evolutionary optimization technique that iteratively applies to following two phases:

1. Generation of a sample random data according to a specified random mechanism.
2. Updating the parameters of the random mechanism, typically parameters of probability mass functions (PMF), on the basis of this data, to produce a "better" sample in the next iteration.

In the rest of this abstract we present we apply the CE meta-heuristic technique to solve the integrated line planning and timetabling problem. The results of a numerical study based on actual data from the Israeli railway system are then briefly reported.

2 Solution Procedure

In this section we formulate the integrated Line Planning and Train Timetabling Problem and we present an effective method to encode a feasible solution. We then apply the CE meta-heuristic.

The input of our model consists of:

- Origin-Destination matrix of passengers for each period of the planning horizon (typically a day).

- Description of the railway infrastructure including minimal traveling time through each block and minimal dwelling time at each station.

- Set of possible routes, maximal frequency and priority for each route. Each instance of a route is taken as a possible train in use.

The infrastructure is encoded as undirected graph where edges represent rail *blocks* and vertices represent either stations, siding or signals. A block is an atomic track section that may normally hold a single train at a time in order to maintain a required safety level while vertices may represent a rail section that can hold several trains at a time. A track segment that connects two vertices may contain several parallel blocks. Routes are directed paths on this graph.

The priorities of the routes are pre-determined by the planner and they define the by-passing order. Maximal frequency can be easily computed by dividing the cycle time by the traveling time on the longest block in the route. Next we present a method to encode a feasible solution in a manner that will be useful for our algorithm. Each possible train is encoded by the following variables.

- IN_USE - Boolean stating whether or not this train is being used.

- FIRST_TIME - The earliest time (in a cycle) that the train can be inserted at the first block.

- STOPPING_STATIONS – Set of passenger stations where the train stops. We represent this set by a characteristic (Boolean) vector.

We refer to the above triplet as a "*gene.*" A feasible solution is represented by a string of genes, one for each possible train. Such a string is referred to as a *chromosome.*

A chromosome is decoded into a timetable of a single cycle by trying to insert all the trains with true value of IN_USE one at a time in non-increasing priority order. This process is similar to the manual process done by planners. The insertion operation is done after checking whether a time slot is available for the train in all of its blocks starting at the first block from time FIRST_TIME and on. For each block we look for an available time slot that is consistent with the previous one. All times are kept in minutes modulo the cycle length (60 minutes). A train may dwell at a station or siding until the next block becomes available and it must dwell for at least some pre-specified time at each station in STOPPING_STATIONS. Next, the cycle is being duplicated over the planning horizon, usually a day, to create a feasible timetable.

A feasible timetable is evaluated with respect to a series of origin-destination matrix that represent the passenger demand over time. We calculate the *total travel time* of all the passengers. To accomplish this calculation an *events graph* is built. Each arrival and departure of trains to or out of a station is represented by a node in this graph. With each node we store the time, the station, and the train of the event. Arcs connect each pair of consecutive events of a train and each pair of consecutive events in a station. For each node in this graph we calculate the earliest reachability time to each station. A specialized reachability algorithm was devised taking advantage of the special structure of the graph (i.e., directed acyclic with maximal out degree of two) and the fact that we are only

interested in reachability to the earliest node of each station. The running time of this algorithm is linear in the number of events times the number of stations, i.e., in its output size.

The *total travel time* of a passenger is the difference between the first reachable time to the destination and his arrival time at the origin station. This calculation is valid under the assumption that the capacity of the trains is not a binding constraint.

We have developed fast methods to encode and generate a feasible cyclic timetable and to evaluate its service quality. This calls for a heuristic approach that will allow us to examine a large amount of solutions and the CE meta-heuristic technique is an attractive alternative.

We randomized a generation of chromosomes using the following multidimensional distribution function. The probability of IN_USE and the STOPPING_STATIONS being true are determined by Bernoulli random variables and FIRST_TIME is drawn from some general discrete (empirical) distribution. Initially the probability of all the Bernoulli random variables is set to 0.5 and the empirical distribution of FIRST_TIME for each train is uniform over some portion of the cycle. After a generation is created and evaluated, this distribution function is updated based on the elite set (e.g., the best 10% solutions) as follows: The probability of the Bernoulli random variables is set to the frequency of the true values and the distribution of FIRST_TIME is calculated based on the empirical distribution. The new distribution function is exponentially smoothed by a weighted average with the distribution function of the previous generation. When the distribution parameters tend to degenerate into a deterministic value the algorithm is stopped.

3 Numerical Experiment

We used the Israeli train network as a test bed. This system is composed of some 47 passenger stations, 130 blocks, and 22 sidings and operational stations. There are 14 inbound routes and 14 outbound routes. The infrastructure consists of both uni- and bi-directional blocks. We constructed cyclic timetables based on these 28 routes. The timetable created by our algorithm can save about 20% of the *total travel time* as compared to the current one. The CE algorithm typically converges in few hours. The same algorithm was used to solve a bi-objective problem to explore the tradeoff between operational costs and service level and some solutions that dominate the currently used schedules with respect to both objectives were found. For example we obtained a solution with operational cost similar to the one currently in use that reduces the total traveling time of passengers by 11%.

4 Conclusion

We devised a method that enables us to solve an integrated Line Planning and Timetabling Problem based on the CE meta-heuristic technique. This is the first application of this meta-heuristic technique to train planning. The solution encoding used by our method can be used in other meta-heuristics such as *Genetic Algorithm, Simulated Annealing* and *Tabu Search*.

References

[1] J.-F. Cordeau, P. Toth, and D.Vigo, "A survey of optimization models for train routing and scheduling", *Transportation Science* 32, no. 4 (1998): 380-404.

[2] A. Caprara, L. Kroon, M. Monaci, M. Peeters, P. Toth (2006) "Passenger railway optimization." In: Barnhart C, Laporte G (eds) Handbooks in operations research and management science, vol 14. *Elsevier*, Amsterdam, 129–187.

[3] M. R. Bussieck, T.Winter and U. T. Zimmermann, "Discrete optimization in public rail transport". *Mathematical Programming*,79 (1997) 415-444.

[4] M. R. Bussieck, P. Kreuzer and U. T. Zimmermann, "Optimal Lines for Railway Systems", *Eur. J. Operational Res*, 96 (1996) 54–63.

[5] U. Pape, Y.-S. Reinecke and E. Reinecke. 1995. Line network planning. J. R. Daduna, I. Branco, J. M. P. Paixäo, eds. 1995. Proc. *6th Internat. Workshop Comput.-Aided Scheduling of Public Transport(CASPT), Lisbon, Portugal,* 1993, Vol. 430. *Lecture Notes in Economics and Mathematical Systems*. Springer-Verlag, Berlin, Heidelberg, Germany, 1–7.

[6] A. Schöbel and S. Scholl, "Line Planning with Minimal Traveling Time", In *Proc. 5th Workshop on Algorithmic Methods and Models for Optimization of Railways* ATMOS 2005.

[7] D. Jovanoví´c and P. T. Harker, "Tactical Scheduling of Rail Operations: The SCAN I System", *Transportation Science*, 25 (1991), 46–64.

[8] U. Brännlund, P.O. Lindberg, A. Nöu, and J.E. Nilsson, "Railway Timetabling Using Lagrangian Relaxation", *Transportation Science* 32 (1998), 358–369.

[9] A. Caprara, M. Fischetti, and P. Toth, "Modeling and Solving the Train Timetabling Problem", *Operations Research* 50 (2002), 851–861.

[10] M. Carey, "A model and strategy for train pathing with choice of lines, platforms and routes", *Transportation Research* Part B, Vol. 28B, No. 5. (1994) 333-353

[11] B. Szpigel, "Optimal Train Scheduling on a Single Track Railway", in M. Ross (ed.), *Operational Research '72*, North-Holland, Amsterdam, 1973, 343–351.

[12] J. Törnquist, "Computer-based decision support for railway traffic scheduling and dispatching: A review of models and algorithms". In *5th Workshop on Algorithmic Methods and Models for Optimization of Railways*, 2006.

[13] P. Serafini and W. Ukovich, "A Mathematical Model for Periodic Event Scheduling Problems", *SIAM Journal on Discrete Mathematics* 2 (1989), 550–581.

[14] R.Y. Rubinstein, "The Cross-Entropy Method for Combinatorial and Continuous Optimization", *Methodology and Computing in Applied Probability* 2, (1999), 127–190.

[15] R.Y. Rubinstein, and D.P. Kroese, The Cross-Entropy Method: A *Unified Approach to Combinatorial Optimization, Monte-Carlo Simulation, and Machine Learning*, Springer-Verlag, New York (2004).

[16] L. Margolin, "On the Convergence of the Cross-Entropy Method", *Annals of Operations Research*, 134, 201–214 (2005).

Invariance principle in regional tax/toll competition

M.M.Khoshyaran

Economics Traffic Clinic (ETC)
34 Avenue des Champs Elysées
F75008, Paris - France

J.P. Lebacque

UPE-INRETS-GRETIA
Le Descartes 2, 2 rue de la Butte Verte
F93166 Noisy-Le-Grand - France

Corresponding Author: M.M.Khoshyaran

Email: etclinic@wanadoo.fr

1 Introduction

The object of this paper is to analyze some possible implications of tax/toll competition between regions. The context is one of increasing regional authority and responsibility in the European Community. As such, the paper is prospective: it deals with possible consequences of current evolutions. Many researchers have studied tax competition; for example, [1], [2], [3], [4], and many others. Some papers have been devoted to the study of commodity tax competition models: [5], [6], [7], [8].

In order to carry out this analysis we build a simplified model for two regions which expands on [9], [10], in which complex transportation costs in regional tax competition are taken into account in the case of two regions. Regions are assumed to compete in order to maximize their revenue (by which we measure how well-off a region is). In order to achieve their objective, regions are assumed to use tax and toll instruments, which affect good consumption and transportation. As a consequence of the structure of the competition, the reaction functions for taxes and tolls are multi-valued and do not yield Nash equilibriums in general.

The iterative process by which the two regions choice choose their optimal toll and or taxes each in turn need not converge. Indeed for each region, there exist also several alternative values of taxes and tolls possible for each value of the taxes and tolls of the other region (or no value at all).

The results extend to general networks. Thus in a process of sequential optimization, in which each region maximizes in turn its revenue, regions are unlikely to reach any Nash equilibrium, and

while some regions profit others are worse off, which expresses the invariance principle (the affluence of regions is being measured by their revenue).

2. A model for two regions competing using tolls and VAT

In the case of two competing regions, we propose a simplified model based on [9], which dealt with the problem of regional tax competition ([1], [8], [4]). [9] has expanded on the model of Kanbur and Keen [5] by introducing the heterogeneity of population and travel times. Further the stability of the regional tax competition equilibrium was analyzed in [11].

The simplified model proposed here makes semi-analytical computations possible; it is adapted so it is compatible with the general model developed in the previous section. The model is based on the following basic ideas:

- The transportation system is simplified: transportation costs are considered only to the border.
- The tax/toll competition problem can be parameterized in terms of two parameters only: the differences in tax level and the sum of tolls.

The elements of the model are the following:

- There are two regions i = 1 and 2,
- Region (i) applies a VAT W_i on consumption and a toll t_i on transportation,
- The population density is $\rho_i(x_i)dx_i$, with x_i *the distance to the closest centre over the border,*
- The density of consumers at distance x_i to the closest centre over border with travel cost τ is $\varphi_i(x_i, \tau)d\tau$,
- The density of the population with respect to the travel cost τ_i is given by:

$$\mathrm{P}_i(\tau_i)d\tau_i \overset{def}{=} d\tau_i \int_{(i)} \rho(x_i)\varphi_i(x_i, \tau_i)dx_i$$

- The total population of region (i) is: $N_i = \int_0^\infty \rho_i(x_i)dx_i = \int_0^\infty P_i(\tau_i)d\tau_i$,
- Cost of buying in region (j): $\lambda(\tau_i + t_i + t_j) + W_j + \eta_i$, with
- η_i a random variable expressing the variability of consumers, the variability of the consumer perception of items such as travel time, the variability of travel costs,
- λ the fraction of the travel cost supported by consumers (as in the previous section 2.4).

- Cost of buying in region (i): $W_i + \zeta_i$, with ζ_i a random variable expressing the variability of consumers and consumer perception.

- The probability for a consumer in region (i) to buy in region (j) is given by:

$$
\begin{aligned}
\mathbf{P}[i \to j] &= \mathbf{P}\left[\eta_i - \zeta_i \le W_i - W_j - \lambda\left(\tau_i + t_i + t_j\right)\right] \\
&= G_i(\Delta W - \lambda t)
\end{aligned}
$$

with notations:

$$
\Delta W \stackrel{def}{=} W_i - W_j, \ t = t_i + t_j,
$$

$$
G_i(\sigma) \stackrel{def}{=} \frac{1}{1 + \exp(-\theta_i\sigma)} \stackrel{def}{=} \int_{-\infty}^{\sigma} g_i(s)ds \quad \text{and} \quad g_i(s) = \frac{\theta_i \exp(-\theta_i s)}{(1 + \exp(-\theta_i s))^2} \quad \text{(a}
$$

simple logistic model)

Now we can calculate the number of consumers in region (i) buying in region (j). Let $N_{i \to j}$ be the number of consumers of region (i) buying in (j).

$$
\begin{aligned}
N_{i \to j} &= \mu\int_0^{\infty} d\tau_i P(\tau_i) G(\Delta W - \lambda t - \lambda\tau_i) \\
&= \mu\int_0^{\Delta W} d\sigma Q_i(\sigma - \lambda t)
\end{aligned}
$$

with μ the factor of impact of demand and with

$$
Q_i(\varsigma) \stackrel{def}{=} \int_0^{\infty} d\tau_i P(\tau_i) g_i(\varsigma - \lambda\tau_i)
$$

Q_i represents the population density of region (i) corrected by the effect of the variability of consumers. If we denote by $N_{k \to \ell}$ the number of consumers in region (k) buying in region (ℓ), we deduce from the above calculation the following results:

$$
\begin{aligned}
N_{i \to i} &= \mu\int_{\Delta W}^{\infty} d\sigma Q_i(\sigma - \lambda t) \\
N_{i \to j} &= \mu\int_0^{\Delta W} d\sigma Q_i(\sigma - \lambda t) \\
N_{j \to j} &= \mu\int_{-\Delta W}^{\infty} d\sigma Q_j(\sigma - \lambda t) \\
N_{i \to j} &= \mu\int_0^{-\Delta W} d\sigma Q_j(\sigma - \lambda t)
\end{aligned}
$$

We deduce the reaction curves for instance for region (i):

$$S_i^W(\Delta W, t) = \begin{cases} W_i = \lambda t_i \dfrac{q_i - q_j}{q_i + q_j} + \mu^{-1} \dfrac{N_{i \to i} + N_{j \to i}}{q_i + q_j} \\[2ex] W_j = -\Delta W + \lambda t_i \dfrac{q_i - q_j}{q_i + q_j} + \mu^{-1} \dfrac{N_{i \to i} + N_{j \to i}}{q_i + q_j} \end{cases}$$

$$T_j^W(\Delta W, t) = \begin{cases} t_i = t - W_j \dfrac{-q_i + q_j}{q_i + q_j} - \lambda^{-1} \mu^{-1} \dfrac{N_{i \to j} + N_{j \to i}}{q_i + q_j} \\[2ex] t_j = W_j \dfrac{-q_i + q_j}{q_i + q_j} + \lambda^{-1} \mu^{-1} \dfrac{N_{i \to j} + N_{j \to i}}{q_i + q_j} \end{cases}$$

with $\quad q_i \stackrel{def}{=} Q_i(\Delta W - \lambda t), q_j = Q_j(-\Delta W - \lambda t)$

3. Extensions

The model can be extended in several ways: inclusion of congestion costs and demand functions. The model can also be extended as a multi-regional model, in which case it does not admit a semi-analytic solution, but must be solved numerically.

The multi-regional model which takes into account consumption of a single generic good, demand functions at population centres, impact on transportation of activity, distribution, assignment on the transportation network with non constant arc costs, and node supply constraints. The model distinguishes local flows from flows induced by economic activity. Regions apply tolls on transportation and taxes (VAT) on the generic good.
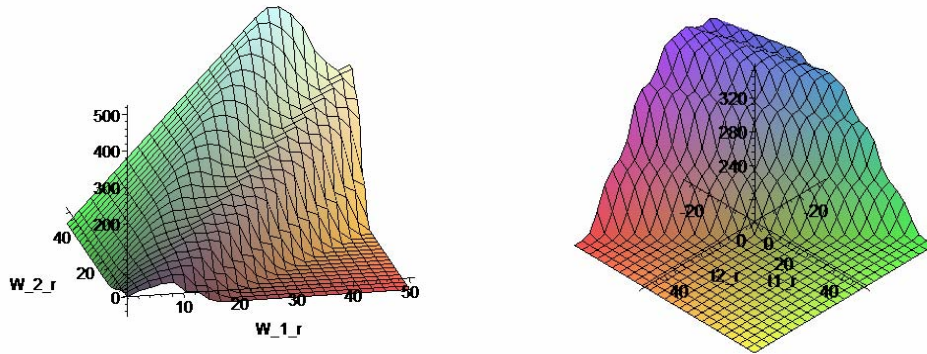
Figure: revenue of region 1, as a function of the VAT W_1, W_2 in the two regions (left), and of the tolls t_1, t_2 in the two regions (right)

In the general model, partial reaction functions can be defined for any region, and as in the case of two regions, the iterative process of regions optimizing their revenues in turn normally does not converge in general. The process results in some regions losing out, and in others gaining, again in conformity with the invariance principle postulate.

In this complex system the reaction functions are necessarily multi-valued and only piecewise continuous.

References

[1] Mintz, J., Tulkens, H., 1986. Commodity tax competition between member states of a federation: equilibrium and efficiency, Journal of Public Economics 29, 133-172.

[2] Wildasin, D.E., 1988. Nash equilibria in models of fiscal competition. Journal of Public Economics 35, 229-240.

[3] Hoyt, W.H., 1993. Tax competition, Nash equilibria and residential mobility. Journal of Urban Economics 34, 358-379.

[4] Bayindir-Upmann T.and A. Ziad. Existence of equilibria in a basic tax-competition model. *Regional Science and Urban Economics*. 35, 1, 1-22, 2005.

[5] Kanbur R. and M. Keen. 1993. Jeux sans frontières: tax competition and tax coordination when countries differ in size. American Economic Review. 877-892.

[6] Ohsawa, Y., 1994. Spatial competition models among local governments. Regional Science and Urban Economics 29, 33-51.

[7] Lockwood, B., 1993. Commodity tax competition under destination and origin principles. Journal of Public Economics 52, 141-162.

[8] Lucas V. Cross-border shopping in a federal economy. *Regional Science and Urban Economics*. , 34, 365-385, 2004.

[9] Khoshyaran, M. , 2002-2004 Jeux sans Frontières revisited: A new study of Tax competition and Tax Coordination when transportation costs are complex. Presented at the *International Conference on Public Economic Theory*, University of Paris 1 – Pantheon-Sorbonne – France, July 4th – 6th, 2002. Published in *Annales d'Economie et de Statistique* , 75 76, 37-67.

[10] Khoshyaran, M. ,Lebacque, J.P. 2006. Existence and Stability of equilibriums in Regional tax competition. EWGT**,** Bari, september 26-29, 2006.

The cost of flexible routing

Philip Kilby

NICTA and Australian National University

Email: Philip.Kilby@nicta.com.au

Andrew Verden

NICTA

Lanbo Zheng

NICTA

1   Introduction

We outline an architecture for solving instances of the Vehicle Routing Problem that have arbitrary constraints that must be observed by solutions. The system uses a Constraint Programming (CP) system to model, propagate and check constraints. The use of the CP system allows the system to be very flexible – producing solutions for essentially arbitrary constraints that model the business practices of the companies that will use the the system. However, this flexibility comes at the price of increased execution time, and may effect solution quality. The primary contribution of the paper is to examine some facets of the trade-off between flexibility, solution quality and execution cost.

2   Vehicle Routing

In the Vehicle Routing Problem (VRP), a fleet of vehicles must deliver goods or services to satisfy a number of customer requests. The primary problem to be solved is the assignment of requests to vehicles, and the ordering of requests within a vehicle so as to minimise overall costs.

The problem has been widely studied in the Operations Research literature (see for instance the reviews [9, 1, 2, 4]). Various constraints have been studied, most commonly *capacity* constraints where the load on each vehicle is limited, and *time windows* constraints where the time service begins is limited to a particular period. In a variant called pickup-and-delivery problems (PDP), requests are paired – one pickup and one delivery. The delivery must follow the pickup, and be assigned to the same vehicle.

However, constraints seen in practice often extend well beyond these basics. The General Vehicle Routing Problem (GVRP) [3] includes a number of constraints that are seen a practice. These include

- multi-dimensional capacity constraints (capacity measured in multiple dimensions, e.g. weight and volume)

- multiple time windows (i.e. service within a one of a set of time windows)

- vehicle compatibility constraints (a request can/can not be served by a particular vehicle)

- request compatibility constraints (two requests must/must not be served by the same vehicle)

- precedence constraints (one request must be served before another)

- Constraints on total route time and length

The GVRP goes a long way towards capturing the constraints found in real-world applications. By combining vehicle compatibility and precedence constraints, pickup and delivery problems are included as a subset.

However, as noted by Kilby and Shaw [5], additional constraints are often found in real-world problems. Examples include

- A PDP where the delivery must be completed within 20 minutes of pickup. (Unlike the usual time window which is known *a-priori*, this time window is not set until the the pickup is finalised)

- Constraints on how requests can be loaded - e.g. 2D packing constraints

- Driver break constraints

- First-in-last-out unloading constraints

The list is as long as there are businesses requiring solutions – each business will have constraints that implement their own business practices. In order to produce a usable solution, a vehicle routing solver must be able to capture these rules, and produce solutions which observe them.

3   A Flexible Solver

This paper suggests an architecture which marries a finite domain constraint programming (CP) solver with a vehicle routing solver to flexibly solve real-world vehicle routing problems. A constraint programming language is used to express the constraints, and the CP solver is used to help construct, and to check, solutions.

The system described, called *Indigo*, has a number of constraints as "native" – that is, the solver "knows" about these constraints, and is able to construct solutions which observe these constraints. The list of native constraints is exactly the list of constraints included in the GVRP specification of [3]. Additional constraints are specified using the Zinc modelling language [6]. This method of specification allows almost arbitrary constraints to be placed on the solution. The constraints are handled during construction and local search using a simple Constraint Programming platform.

The use of CP in solving Vehicle Routing problems has been discussed by Kilby and Shaw [5]. The *Indigo* solver uses some of the ideas discussed there.

The solver proceeds in two phases. First, an initial solution is created *ab initio*. An insertion-based procedure is used by which one of the requests is inserted into the solution at each iteration, until no more requests can be feasibly inserted. No requests are removed during this phase of the method. This style of initial creation allows the CP system maximum scope to propagate the effects of each decision – as mediated by all of the particular constraints of the problem – and allows maximum information flow to the insert procedure.

In the second phase, local search is used to improve the solution. The main, low-level operator explores the Or-Opt neighbourhood [7]. A Large Neighbourhood Search [8] is also applied to improve the solution. Again, Large Neighbourhood Search is an insertion-based procedure, which allows maximum use to be made of the propagations from the arbitrary constraints.

For ease of exposition, the version of *Indigo* used in testing uses standard implementations and parameters for local search and meta-heuristics. *Indigo* also has more advanced techniques available, but the methods used here offer a standard "baseline" by which to compare the effectiveness of side constraint handling.

4   Computational Testing

In order to test the effectiveness of this method of handling arbitrary constraints we look at two of the types of constraint included in the GVRP: Time Window constraints, and PDP constraints.

Were these not already able to be solved by the *Indigo* solver, we would be able to implement them using the Zinc interface. In order to test the effectiveness of the specification and solving using this interface, this is exactly what we do.

We look at solving a set of benchmark VRP with Time Windows problems, and a set of PDP problems with Time Windows (PDPTW), using the *Indigo* solver. In one set of these runs, the *Indigo* solver uses its inbuilt methods to handle the Time Window and PDP constraints. In a second set of runs, these constraints are handled using the constraint programming system.

We are then able to look at the cost of using the CP constraint interface. We look at two main questions

- Are the solutions as good as those generated by the native *Indigo* solver?
- How much longer does it take to handle the constraints using a CP solver, rather than native code?

We will look at the results of runs of *Indigo* solver handling the Time Window and PDP constraints as native, and compare them to solving the same problems using the CP system to handle those constraints.

We know that the interface to the CP solver gives the *Indigo* solver great power to express and solve problems with arbitrary constraints. The answers to these questions will begin to shed light on the cost of this flexibility in terms of objective value and solution time.

References

[1] Olli Bräysy and Michel Gendreau. Vehicle routing problem with time windows, part I: Route construction and local search algorithms. *Transportation Science*, 39(1):104–118, 2005.

[2] Olli Bräysy and Michel Gendreau. Vehicle routing problem with time windows, part II: Metaheuristics. *Transportation Science*, 39(1):119–139, 2005.

[3] Asvin Goel and Volker Gruhn. A general vehicle routing problem. *European Journal Of Operational Research*, 191(3):650–660, 2008.

[4] Bruce L. Golden and Arjang A. Assad, editors. *Vehicle Routing: Methods and Studies*. North-Holland, Amsterdam, 1988.

[5] Philip Kilby and Paul Shaw. Vehicle routing. In F. Rossi, P. Van Beek, and T. Walsh, editors, *Handbook of Constraint Programming*, Foundations of Artificial Intelligence, chapter 23, pages 801–836. Elsevier, 2006.

[6] Kim Marriott, Nicholas Nethercote, Reza Rafeh, Peter J. Stuckey, María García de la Banda, and Mark Wallace. The design of the Zinc modelling language. *Constraints*, 13(3):229–267, September 2008.

[7] I. Or. *Travelling Salesman-Type Combinatorial Problems and Their Relation to the Logistics of Blood-Banking*. PhD thesis, Department of Industrial Engineering and Management Sciences, Northwest University, Evanston, IL., 1976.

[8] Paul. Shaw. Using constraint programming and local search methods to solve vehicle routing problems. In M. Maher and J.-F. Puget, editors, *Fourth International Conference on Principles and Practice of Constraint Programming (CP '98)*. Springer-Verlag, volume 1520, pages 417–431, 1998.

[9] Paolo Toth and Daniele Vigo, editors. *The Vehicle Routing Problem*, volume 9 of *SIAM Monographs on Discrete Mathematics and Applications*. SIAM, Philadelphia, PA, 2001.

Strategic Gang Scheduling in the Railway Industry

Dengfeng Yang

Department of Industrial Engineering and Management Sciences

Northwestern University

Clark Cheng, Edward Lin, Kannan Viswanath, Jian Liu

Norfolk Southern Corporation

Diego Klabjan

Department of Industrial Engineering and Management Sciences

Northwestern University, Evanston, IL US

Email: d-klabjan@northwestern.edu

1 Introduction

The railway industry is an infrastructure intensive industry. Major Class-1 US railways have more than 20,000 miles of tracks. The tracks are heavily used and thus subject to wear and failures. Throughout a year, track maintenance, which encompasses both preventive and real-time maintenance, is an equipment and labor intensive process. Maintenance workers work in groups, called gangs, performing heavy labor duties such as installing track ties, driving track spikes, shoveling ballast, and other similar maintenance track related activities. Gangs work outdoors at almost all weather conditions to install and maintain the tracks properly. A gang consists of several members, typically between ten and one hundred, with different responsibilities such as machine operators, termite welders, assistant extra gang foremen, or extra gang foremen. A gang beat or a beat section is a regular section of a track where some form of a maintenance need to be performed. During a year a gang moves from one beat section to another upon completion of the work at the section. A beat section together with its attributes such as the type of work, the number of days to complete the work, and an underlying time window to perform the task is called a job.

   Gang scheduling is to find a favorable gang schedule for each gang during a given planning horizon that is governed by several regulatory and union rules. A gang schedule consists of a sequence of jobs together with the underlying start time of each job. Typically there are three or four types of gangs (rail, tie and surface, surface, and dual) and more than 1,000 jobs around the U.S. Each job has an underlying required gang type, which implies that a particular job can only be carried out by particular gangs. Certain jobs must obey precedence constraints, e.g., work on ties must precede any

work on the rails. From the operational perspective, it is not desirable to perform two jobs geographically close to each other during the same period since it could substantially disrupt train operations. Few occurrences of such situations create a more robust schedule. At most railways the process of gang scheduling is an intensive manual effort often leading to costly and inefficient schedules.

The main goal of this study is to develop an algorithm to solve the gang scheduling problem considering all business requirements. The objective is to develop an algorithm capable of:

1. minimizing the total cost including the gang related cost of transitioning between jobs and the cost of equipment movement between two locations, as well as the travel allowance covering the transfer from a job location and the home domicile of the underlying gang,

2. obeying all business requirements, e.g., it is suitable for gangs to work at a southern area during the winter period, but jobs in the northern part must be performed during the summer season, and

3. computationally handling the large-scale instances arising in the industry.

The main contributions of this work are:

1. designing a network based model capturing job precedence and robustness, and

2. developing a construction and mathematical programming heuristic (math-heuristic) for solving the underlying model.

2 Model

We formulate a network model $G (N,A)$ for each gang g, where N is the set of all job nodes and A the set of arcs in the model.

Node (j,t) of the network is encoded by job j eligible to be performed by gang g and the associated starting time t. Given time t, job j, and gang g, it is possible to calculate the completion time of the job (from the duration requirement of job j and productivity of gang g). Two nodes are connected by an arc if the completion time of the job associated with the tail plus the transition time from one job to the other one is less than or equal to the start time of the node associated with the head. Other requirements such as gangs not working during weekends and holidays can easily be directly incorporated in the network.

We next focus on the underlying node and arc cost. Cost is driven by travel allowance, and thus all costs are measured in monetary units, i.e., the money paid to a gang for traveling between the locations of two consecutive jobs or for weekend stays at the home domicile. The arc cost is composed of three components:

1. the travel cost between the location of job j, the home domicile of the gang if the transition time includes a weekend, and then return to the location of the adjacent job k,

2. the direct travel cost between the two locations if the weekend is not included, and

3. the equipment movement cost from the location of job j to the location of the adjacent job k.

In addition, each node bears the travel allowance cost for weekend travel to the home domicile over weekends. We note that job duration can span several weeks and thus it might require several weekend home trips.

By construction, each path in the network forms a gang schedule. The underlying mathematical program includes variables that assign a path (gang schedule) to each gang. The underlying constraints impose:

1. each job is assigned to exactly one selected path,
2. each gang is assigned one and only one path,
3. precedence constraints linking paths among the gangs,
4. equal number of work days for all gangs of a certain type,
5. robustness constraints likewise link various path.

3 Methodology

The problem as posed is very hard to solve to optimality since it is an NP-hard problem, and a pure mathematical programming approach does not work due to the sheer size of real-world instances. Either significant ad-hoc preprocessing is required or a branch-and-price algorithm developed. Computational tractability of the latter is questionable due to several unstructured constraints linking paths. For this reason, we resort to a heuristic. Instead of employing a traditional local search strategy, we combine very large-scale neighborhood search ideas with mathematical programming. To this end, we use a two-phase solution methodology. In the first phase a solution covering many jobs but not necessarily all of them is obtained. In the subsequent phase the uncovered jobs are inserted by means of a mathematical program. The solution quality of a feasible solution is then iteratively improved by removing jobs from the incumbent solution and then reinserting them back by means of the same mathematical program.

3.1 Initial Construction Heuristic

First, we designed a heuristic that generates several shortest paths using dynamic programming. Gangs are ranked and then a schedule is found for each gang sequentially by solving a shortest path type problem. Once a path is fixed for a gang, the network is accordingly modified so that each job is assigned at most once, and the precedence and robustness rules are warranted.

To obtain balanced workload within the same gang type we do not choose the exact shortest path with the maximum number of jobs and smallest cumulated cost at the last job node. Instead we choose a path with the total duration falling in some predetermined range. Since each gang has a limited number of working days for the year and each available job has a restricted time window, in this phase, the solution by the shortest path algorithm cannot schedule all possible jobs. An improvement phase is introduced to cover the unassigned jobs.

3.2 Improvement Strategy

The initial solution is then iteratively improved at each subsequent iteration by an interchange algorithm so as to schedule all of the jobs and to guarantee finding a no worse solution than the one we begin with. The basic idea of the improvement phase is to randomly select certain nodes to extract from the starting solution, then extract the nodes, and derive sequences of nodes including extracted nodes and uncovered nodes to reinsert into the short-cut solution. Next we formulate an integer program and solve it using an ILP solver to incorporate the results back into the solution. This process is repeated until an iteration limit is reached or the solution is of an acceptable quality.

3.2.1 Recombination

In this step, we use all the extracted/uncovered nodes to create a pool of subsequences to be potentially reinsterted into the solution. To make the problem simpler and improve the performance, we only consider the short sequences of nodes (1 or 2 jobs). The business rules, such as job precedence and robustness, are also considered in this step.

3.2.2 Reallocation

In this step, we form an insertion ILP. This ILP reinserts back the extracted/uncovered nodes. Note that the model is always feasible since the nodes can always be reinserted back to the original locations. Binary decision variables indicating whether certain sequence s is inserted at some insertion point i, and integer decision variables presenting the shift in time of the job sequences at the short-cut solution after nodes extraction, are introduced into the ILP model. It is critical to introduce the shift time variables since at some insertion point, the time window of the subsequent job sequence must be enlarged or shrinked to accommodate the to-be-inserted job sequence. The objective is to minimize the overall cost. Essentially, the ILP is an assignment problem, but with additional constraints, such as jobs precedence and robustness, and time window restrictions. The complete ILP has a large number of columns. We solve it by randomly choosing a predetermined, large number of columns from the entire set of columns to add to the problem.

3.2.3 Reinsertion

We incorporate the optimal results obtained from the reallocation ILP into the solution using specially designed insertion operations. This reinsertion step marks the end of an iteration, and the entire series of steps is repeated until all jobs are scheduled or an improved solution is obtained.

A Model of Alliances between Competing Carriers

Anton J. Kleywegt [*]

School of Industrial and Systems Engineering
Georgia Institute of Technology
Atlanta, Georgia 30332-0205, USA

November 28, 2009

An important way in which carriers collaborate is through the formation of alliances. Carrier alliances can be structured in many different ways, and the detail rules of an alliance are clearly important for both the stability of the alliance, as well as the well-being of each member of the alliance.

Examples of widely used carrier alliances are the following. Airlines often sell tickets on each others' flights through code sharing agreements. There are various ways in which such an alliance can be structured. The major distinguishing factor between different alliance structures involves the control of the revenue management (in effect, the pricing) of the resources that alliance members have access to. For example, in a so-called "free-sell" or "soft block" alliance, each alliance member (the marketing member) can sell tickets for flights operated by another alliance member (the operating member) and the marketing member can put its own code on the flight. That enables carriers to sell tickets for itineraries that include flights operated by multiple carriers, thereby dramatically increasing the number of itinerary products that each carrier can sell. However, under free-sell, the revenue management for the flights included in a code-share agreement is still controlled by the operating member. In free-sell alliances, an important design parameter of the alliance agreement is the set of flights that each carrier is allowed to market under its own code.

Another type of alliance structure is a so-called "resource-exchange" or "hard block" alliance, in which the sellers exchange resources. For example, carriers exchange seat space on various flights, and ocean carriers exchange capacity on various voyages of container ships. In addition, money may be exchanged. After the exchange, each carrier can control the received resources as though they are the producer of the resources. For example, in a resource-exchange alliance carrier 1 may receive 15 seats on flight A operated by carrier 2, and carrier 2 may receive 10 seats on flight B operated by carrier 1 as well as \$2000. After the exchange, carrier 1 controls the revenue management for the 15 seats on flight A that it received from carrier 2, as well as for the remaining seats on the flights that it operates, and similarly, carrier 2 controls the revenue management for the 10 seats on flight B that it received from carrier 1.

Another example of a widely used carrier alliance is the type of alliance that ocean container carriers enter into when they introduce new joint services. A "service" is a cycle (also called a "loop" or a "rotation") of voyages that repeat according to a regular schedule, typically with weekly departures at each port included in the cycle. Suppose the cycle is ports A,B,C,D,E,A. A set of ships is dedicated to the service, with each ship visiting the ports in the sequence $A,B,C,D,E,A,B,\ldots$. To offer weekly departures at each port included in the cycle, the headway between successive ships

traversing the cycle must be one week. In addition, if it takes a ship n weeks to complete one cycle, then n ships are needed to offer the service. For many services that visit ports in Asia and North America, and services that visit ports in Asia and Europe, it takes a ship approximately 6 weeks to complete one cycle, and thus 6 ships are needed to offer the service. Taking into account that a large container ship can cost several hundred million US dollars (and the trend is towards even larger container ships, because larger container ships tend to have significantly lower per unit operating costs), it becomes clear that for even the large carriers it would require an enormous investment to introduce a new service. A solution is for several carriers to enter into an alliance to offer a new service. The majority of services that visit ports in Asia and North America, and services that visit ports in Asia and Europe, are offered by alliances between two carriers. Each carrier in the alliance provides one or more ships to be used for the service. The capacity on each ship is then allocated to all the alliance members, often in proportion to the capacity that the alliance member contributed to the service. For example, if carrier 1 contributes 2 ships and carrier 2 contributes 4 ships to the service, and all the ships in the service have the same capacity, then carrier 1 can use 1/3 of each ship's capacity, and carrier 2 can use 2/3 of each ship's capacity. That way, each carrier in the alliance can offer weekly departures at each port in the service even though it did not have enough ships by itself to do so.

An important feature of resource exchange alliances is that after the resource exchange the alliance members compete by selling substitutable (and also complementary) products. This subsequent competition should be taken into account when structuring resource exchange alliances. However, in spite of this observation, most academic models ignore the subsequent competition. In our research we take this competition after the exchange into account. We develop models to determine the amount of each resource to be exchanged in an alliance with two members, taking into account the consequences of the exchange on the subsequent competition between the alliance members.

First we briefly discuss the modeling of the competition resulting from a given resource exchange. Consider 2 carriers, indexed by -1 and 1. Carrier i produces a set $\mathcal{R}^i$ of resources indexed by $j = 1, \ldots, R^i$. For example, flight j may denote the Monday 8am flight of carrier i from Atlanta to New York. Carrier i produces b_j^i units of resource j. For example, carrier i may make $b_j^i = 200$ seats available on flight j each Monday. There may be multiple repetitions of the same resource, for example, flight j may be repeated once per week. We assume that there is a time period, for example a week, such that the sets $\mathcal{R}^i$ of resources under consideration, of both carriers $i = \pm 1$, repeat once every time period. Suppose that carrier i allows carrier $-i$ to use up to x_j^i units of resource $j \in \mathcal{R}^i$, with $0 \leq x_j^i \leq b_j^i$. Let $x := (ix_j^i : i = \pm 1, j \in \mathcal{R}^i)$ denote the vector of resource exchange quantities, and let $\mathcal{R} := \mathcal{R}^{-1} \cup \mathcal{R}^1$, that is, $x_j > 0$ indicates that carrier 1 allows carrier -1 to use x_j units of resource $j \in \mathcal{R}$, and $x_j < 0$ indicates that carrier -1 allows carrier 1 to use $-x_j$ units of resource $j \in \mathcal{R}$.

Each carrier i has a set $\mathcal{K}^i$ of product types that carrier i can sell with the resource types in $\mathcal{R}$. Each product k of carrier i requires a_{jk}^i units of resource $j \in \mathcal{R}$. Each carrier i sets a price for and sells units of each product type in $\mathcal{K}^i$ in each time period (over which the sets $\mathcal{R}$ of resources repeat). Each carrier may take into account information specific to the time period when deciding on the prices. Suppose that each carrier observes data ξ, and after observing the data, each carrier i chooses a price $y_k^i(\xi)$ for each product type k. Let $y^i(\xi) := (y_k^i(\xi) : k \in \mathcal{K}^i)$ denote the vector of prices chosen by carrier i. Given observed data ξ and a vector (y^{-1}, y^1) of prices chosen by both carriers, the demand for each product type k of carrier i is denoted by $d_k^i(y^{-1}, y^1, \xi)$. The resulting

total revenue of carrier i is given by

$$g^i(y^{-1}, y^1, \xi) \;\; := \;\; \sum_{k \in \mathcal{K}^i} y^i_k \, d^i_k(y^{-1}, y^1, \xi)$$

Assume that the objective of each carrier is to maximize its total revenue.

Given the observed data ξ and a vector y^{-i} of prices chosen by carrier $-i$, the best response problem of carrier i is given by

$$\max_{y^i \geq 0} \quad \left\{ V^i(x, y^{-1}, y^1, \xi) \;\; := \;\; g^i(y^{-1}, y^1, \xi) \right\} \tag{1}$$

$$\text{s.t.} \quad \sum_{k \in \mathcal{K}^i} a^i_{jk} \, d^i_k(y^{-1}, y^1, \xi) \;\; \leq \;\; b^i_j - i x_j \quad \text{for all } j \in \mathcal{R}$$

Given x and ξ, an equilibrium $(y^{-1*}, y^{1*}) \geq 0$ satisfies the property that for each i, y^{i*} is an optimal solution of (1) given y^{-i*}. If one assumes that after the resources have been exchanged and random data have been observed, each alliance member chooses the prices of its products to maximize its own profit, and that this behavior of the alliance members leads to a unique equilibrium, then the problem can be formulated as a mathematical program with equilibrium constraints.

$$\max_{x} \quad \mathbb{E}\left[V^{-1}(x, y^{-1}, y^1, \xi) + V^1(x, y^{-1}, y^1, \xi) \right] \tag{2}$$

$$\text{s.t.} \quad (y^{-1}, y^1) \text{ solves (1) for } i = \pm 1, \text{ for each } \xi \text{ and for } x$$

The reason the sum of the optimal values for the carriers is maximized is that the carriers can exchange money together with resources, and thus it makes sense to both carriers to maximize their total revenue. The exact division of total revenue can be given by the Nash solution to the corresponding cooperative game.

Important questions are whether, for each resource exchange x and each data realization ξ, there is an equilibrium, and if so, whether the equilibrium is unique. We give sufficient conditions for existence and uniqueness of an equilibrium. We propose an algorithm to compute a solution of this mathematical program with equilibrium constraints.

A heuristic for rich maritime inventory routing problems

Oddvar Kloster

Department of Applied Mathematics

SINTEF ICT

Email: Oddvar.Kloster@sintef.no

Truls Flatberg

Department of Applied Economics

SINTEF Technology and Society

1 Introduction

Martime inventory routing problems involve the coordination of vessel routing and inventory managment. Compared to road-based problems, maritime problems have several characteristics that make this integration relevant. The quantities transported are large, both in terms of vessel capacity and storage capacity at production and consumption facilities. In addition, travel times between facilities are considerable. This leads to a problem where routing decisions has a large impact on the inventory levels and vice versa.

Maritime inventory routing problems have been considered by several authors in the last decade using various approaches. These includes Dantzig-Wolfe decomposition [1], mixed integer programming [2], metaheuristics [3, Lagrangian relaxation and heuristics [4], and a hybrid of genetic algorithms and linear programming [5]. In contrast with the above solutions, the present work is not motivated by a specific business operation, but attempts to tackle maritime inventory routing problems in a more general setting. For more references see the recent review of these problems see [6].

In this talk we will present a heuristic to solve real-world maritime inventory routing problems. To properly capture the aspects of the various industrial settings, the heuristic need to handle a rich set of model elements, constraints and objectives. This works build on and extends the work described in [7]. The most important new model elements are bookings, tank cleaning, contracts and boil-off. Also, the algorithm includes a more proper optimization phase, not just repeated construction.

2 Model description

A fleet of vessels, with given capacities and saliing speeds, is available for transporting products between a set of ports. Some ports have *storages*, which each produces or consumes one product according to a time-varying profile. Each storage has a finite capacity for storing product. The goal is to use the vessels' transportation capacity to avoid overflow or stockout at any storage within the time horizon, while maximizing profit. The production and consumption in some storages can be reduced to some extent to help avoiding overflow/stockout.

In problems where there are more than one product, we model stowage in one or more tanks of given capacities aboard each vessel. Each tank may only contain one product at any time. Cleaning may be required when changing from one product to another in a tank, with a mininmum duration and associated cost. In the case of LNG transportation, we must deal with the phenomenon of boil-off , where the amount of product discharged in each trip is less than the amount loaded, due to evaporation.

The operations at each storage may be regulated by contracts. A contract imposes interval limits for the amount that may be picked up/delivered in given periods. It also determines the associated price, which may be given as a time-dependent curve or by more complex mechanisms. A number of other constraints can be present, such as vessel/port compatibility, draft limits, port closure periods, vessel maintenance periods, inter arrival gaps and minimum/maximum number of visits to a storage in given time periods.

The problem may have aspects of tramp shipping, by containing *bookings*, which are requirements or options to transport product from one port to another, independently of the storages. Bookings have time windows (laycan), quantity limits and other constraints.

3 Solution method

We attack the problem using a neighbourhood search based heuristic. The search works with a plan that contains an explicit representation of the schedule for each vessel. Its main constutients are *port stays*, where a vessel visits a port, and *actions* during port stays, that represent product being loaded or discharged. As the feasible part of solution space is quite constrained, we relax (and penalize) a subset of the constraints, namely those that can be fixed by adding more port stays and actions (e.g. stockout in a consumption storage, or too little product delivered in a contract period).

To construct the initial plan, we start with an empty plan, where the initial vessel positions and stock levels are known, but no sailing takes place. This plan is feasible, since only relaxed constraints are violated. We then iteratively identify violations of relaxed constraints and add port stays and actions to the plan in order to fix them. This continues until there are no more violations, or no fix is found for any remaining violation.

The first step in each iteration is to identify which constraint violation to fix. Each violation is assigned a *critical time*, before which some action must be added. The critical time for e.g. a storage is the time of stockout/overflow, while for a booking, it is the end of the loading time window. The violation with the earliest critical time is selected.

We then generate all different journeys that can fix the violation. A journey consists of a pair of matching load and discharge actions for one vessel. Different journeys arise from choosing different vessels, different load/discharge ports, storages and contracts, and different insertion points in the chosen vessel's schedule. Each journey is tested for feasibility, and the change in the plan's objective value is calculated.

The journey feasibility test is the most complex part of the algorithm. Inserting the journey directly affects later stock levels at the visited storages. Other storages that the vessel visits may also be affected, since the timing of the vessel's actions after the new journey is usually changed. Even though we keep the schedules for all other vessels fixed, large parts of the plan must be considered in order to determine feasible assignments of the load/discharge quantity and timing of the journey.

The feasibility test starts with inserting the journey, with zero quantity, and propagating time for the vessel while assuming that all actions and sailing take place as quickly as possible. Based on these approximate times, we find the largest feasible quantity that satisfies the most important hard constraints. We set this quantity for the journey and propagate correct times for the vessel. If necessary, stowage is determined and tank cleaning actions inserted in the schedule. All hard constraints are then checked to verify that the plan is feasible.

At many points during the feasibility test, we may encounter a violation of a hard constraint and fail. Bu we may also find that the violation can be fixed by delaying one of the vessel's actions (e.g. the vessel has an action that overlaps another vessel's action at the same storage. Wait until the other vessel is finished). In this case, we note the earliest feasible start time for the action and restart the feasibility test. From now on, time propagation will respect the noted time as a lower limit for the action's start time, preventing that the same violation occurs again.

When all the feasible journeys have been determined, we select and insert the one that most improves the objective value, before starting the next iteration of construction. In the case where we wish to generate a number of different initial plans, a random element and other criteria can contribute to diversifying the journey selection.

After obtaining the initial plan as described above, we proceed to the optimization phase. During optimization, we repeatedly tear down and then rebuild a part of the plan, in the hope that this leads to an improvement.

The first step in an optimization iteration is to select a random interval of time, usually around 10% of the planning horizon. All journeys, for all vessels, that have one or both actions within the selected interval are removed from the plan. The plan is then compacted, i.e. some of the remaining actions are moved to earlier times (while respecting the hard constraints). This is possible because removing the journeys frees transportation capacity and may eliminate the need for some delays. Compacting keeps the plan from developing in a particular unfavourable direction, where actions are pushed to ever later times by new delays added in the construction heuristic.

After compacting the plan, we reuse the construction heuristic to add journeys until no more violations of relaxed constraints can be fixed. During this reconstruction phase, violations in the part of

the plan where actions were removed are given priority. The new plan so obtained is evaluated and kept if it is the best of the recently generated plans. Otherwise, we revert to the previous plan. A new optimization iteration then starts.

4 Results

The algorithm has been tested on cases based on real world problems from three different business operations: transport of LNG, cement and oil-based products. The characteristics of three selected cases are summarized in Table 1.

Table 1. *Test case characteristics.*

	Business	Vessels	Storages	Products	Period	Contracts	Bookings	Tanks	Cleaning
Case A	LNG	8	7	1	1 year	X			
Case B	Cement	5	60	11	14 ds			X	
Case C	Oil-based	7	4	14	6 mths	X	X	X	X

For each test case we ran the initial construction algorithm followed by 500 optimization iterations. During the optimization, some degrading moves are allowed to diversify the search. The test results are given in Table 2.

Table 2. *Test case results.*

	Initial objective	Best objective	Time (sec.)
Case A	2.84e+08	6.76e+08	52.3
Case B	-785098	-355715	11.28
Case C	1.38e+07	1.65e+07	4.67

Acknowledgments

References

[1] M. Christiansen, "Decomposition of a combined inventory and time constrained ship routing problem", Transportation Science, 33(1), 1999.

[2] M. Fox, D. Herden, "Ship scheduling of fertilizer products", OR Insight 12(2), 1999.

[3] S. Dauzère-Pérès, A. Nordli, A. Olstad, K. Haugen, U. Koester, P.O. Myrstad, G. Teistklub, A. Reistad, Omya Hustadmarmor Optimizes Its Supply Chain for Delivering Calcium Carbonate Slurry to European Paper Manufacturers. Interfaces 37, 2007.

[4] S-J. Hwang, "Inventory constrained maritime routing and scheduling for multi-commodity liquid bulk", Ph.d. Thesis, Georgia Institute of Technology, Atlanta, 2005.

[5] D. Bredström, D. Carlsson, M. Rönnqvist, "A Hybrid Algorithm for Distribution Problems", *IEEE* July/August 2005.

[6] H. Andersson, A. Hoff, M. Christiansen, G. Hasle and A. Løkketangen, "Industrial Aspects and Literature Survey: Combined Inventory Management and Routing", SINTEF Report A7056, 2008.

[7] M. Christiansen, K. Fagerholt, T. Flatberg, Ø. Haugen, O. Kloster and E.H. Lund, "Maritime inventory routing with multiple products: A case study from the cement industry", Working paper, 2008.

A paradox in dynamic traffic assignment
Dynamic extension of the Braess paradox

Victor L. Knoop

Transport & Planning

Delft University of Technology, Delft, The Netherlands

Email: v.l.knoop@tudelft.nl

Serge P. Hoogendoorn   **Bart van Arem**

Transport & Planning   Transport & Planning

Delft University of Technology   Delft University of Technology

1   Introduction

The Braess paradox [1] is well known by traffic engineers. It states that adding a link to a network can, in special conditions, lead to an *increase* in total travel travel. Braess' work is based on a static network with link travel times. There are several conditions, like the maximum increase of travel time [2], which are derived for this case. However, with dynamic queuing models, the paradox changes. This paper will show that even for a very small network the addition of a link can increase the travel time (section 2). It will be also argued that this is in fact a common situation for real-world networks. The paper also presents a possible solution for the road layout avoiding the extra delay in section 3.

In this extended abstract we will not explain the queuing model in detail. We use a conceptual dynamic queuing model. The only important features are (1) the flow on a link is restricted to capacity and (2) if demand exceeds capacity, a queue will grow upstream of the bottleneck. For the extended abstract we assume that the vehicle speed up to capacity is the free flow speed. This assumption simplifies the calculations in the following section, but is not essential for the concept.

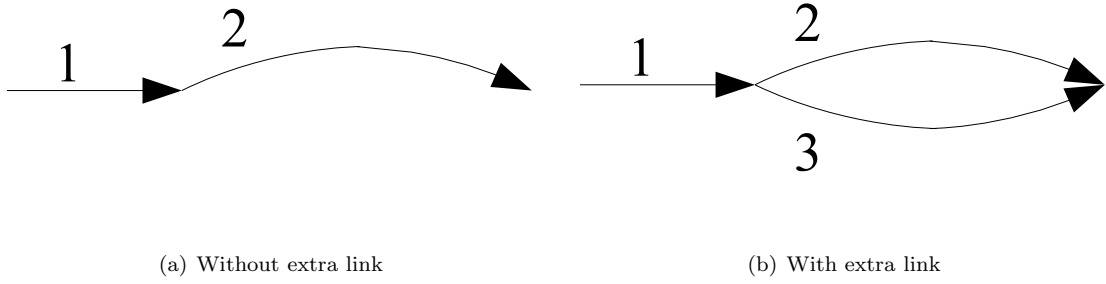(a) Without extra link   (b) With extra link

Figure 1: Network

2   Network and demand

Figure 1a shows the very simple network which we will be considering in this paper. The effect occurs for more networks, but this simple network will be used to show the relevant process. The link capacity is indicated with C and the traffic demand is Q. For this network, $C_1 > Q$ and $C_2 > Q$ (the properties of these links and all links which will be introduced are also shown in table 1). Since the capacity is sufficient, there is no congestion in the network and, because in this extended abstract we assume a free flow speed up to capacity, the average travel time ($T_{\text{without}}^{\text{avg}}$) is the sum free flow travel time on link 1 and link 2:

$$T_{\text{without}}^{\text{avg}} = T_1^{\text{free flow}} + T_2^{\text{free flow}} \tag{1}$$

Now consider adding link 3 (figure 1b), which has a capacity $C_3 < Q$ and a free flow travel time travel time $T_3^{\text{free flow}} = (T_2^{\text{free flow}} + \tau)$ with $\tau > 0$. Since there is no bottleneck on link 2 or link 3, traffic will be in free flow conditions. From the diversion point onwards, it will therefore be faster to take link 3. In a Wardrop equilibrium [3], users will only take the path with the lowest cost (in this case being travel time), meaning the traffic demand to link 3 is the full demand Q. However, since $C_3 < Q$ this will create congestion upstream of the diversion point, i.e. on link 1. All travellers, also travellers which might turn to link 2 will envisage this congestion. Therefore the average travel time is

$$T_{\text{with}}^{\text{avg}} = T_1^{\text{cong}} + T_3^{\text{free flow}} \tag{2}$$

The difference in travel times can be calculated from the equation 1 and equation 2. In the limit that $\tau \to 0$, the extra delay D is:

$$D = T_{\text{with}}^{\text{avg}} - T_{\text{without}}^{\text{avg}} = T_1^{\text{free flow}} + T_2^{\text{free flow}} - T_1^{\text{cong}} - T_3^{\text{free flow}} = T_1^{\text{free flow}} - T_1^{\text{cong}} > 0 \tag{3}$$

This increase in travel time is only due to the addition of a link. As long as $Q > C_3$ the queue will grow and the delay will increase, theoretically to infinity. For the static network with link travel times, the possible travel time is bounded to twice the original travel time [2]; this no longer holds for the dynamic case.

Table 1: Properties of the links

Link	1	2	3	3a	3b
Capacity C	$> Q$	$> Q$	$< Q$	$> C_{3b}$	$= C_3$
Free flow travel time	$T_1^{\text{free flow}}$	$T_2^{\text{free flow}}$	$T_2^{\text{free flow}} + \tau$	$T_{3a}^{\text{free flow}}$	$= T_3^{\text{free flow}} - T_{3a}^{\text{free flow}}$
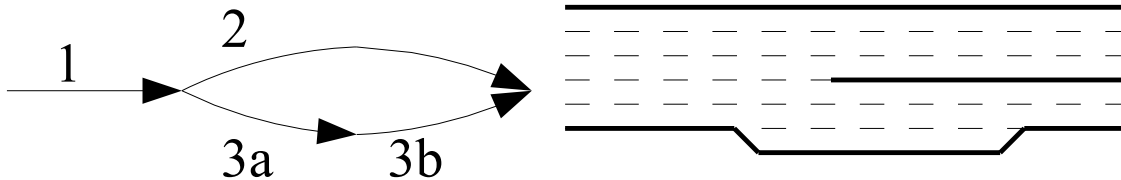
(a) Graph

(b) Implementation in lanes

Figure 2: The solution avoiding extra travel time

Although the network might seem artificial, it is actually a situation which can often occur in practice. Imagine a road approaching a town (link 1). To get to the other side, there is a motorway around the town, or a highway through the town. Often the motorway link will be take more time in case traffic in town is undisturbed.

3 Solution

To avoid this problem, the network designer has to make sure that the travel time on link 2 is larger than on link 3, or that the queue because of the restricted capacity of link 3 will not delay travellers turning onto link 2. This is possible by redesigning the network as shown in figure 2a. If one designs the network such that $C_{3b} < C_{3a}$, a queue will arise if on link 3a the demand exceeds the capacity of link 3b. If link 3a is long enough, traffic to link 2 is not delayed by this queue. Furthermore, once there is a queue on link 3a, the travel time over link 3a increases, which will make more travellers taking link 2 instead. In terms of road layout, this solution is relatively easy to implement. An example is shown in figure 2b.

Another solution would be to artificially increase travel time on link by means of traffic management (for instance, by introducing traffic lights). However, also with the these solutions, the total travel time will not be lower than the original travel time. If the link is not constructed for travel time reduction but for other reasons (e.g., access to a part of the town), then these solutions are useful.

4 Conclusions

The paper presented a paradox based on the Braess paradox. It shows that even in a very simple network layout the addition of an extra link can cause an increase of total travel time. It is furthermore shown that this delay is not bounded. The paper also provides solutions to avoid the extra delay. The network element which causes this delay is very common in real-world networks. Future research should show how large this problem in fact is.

References

[1] D. Braess. Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung*, 12:258–268, 1968.

[2] Tim Roughgarden and Éva Tardos. How bad is selfish routing? *J. ACM*, 49(2):236–259, 2002.

[3] J. Wardrop. Some theoretical aspects of road traffic research. *Proceedings of the Institute of Civil Engineers, Part II*, I(2):325–378, 1952.

Risk-averse Traffic Assignment in a Dynamic Traffic Simulator

Victor L. Knoop

Transport & Planning

Delft University of Technology, Delft, The Netherlands

Email: v.l.knoop@tudelft.nl

Michael G.H. Bell	**Henk J. van Zuylen**
Centre for Transport Studies	Transport & Planning
Imperial College, London	Delft University of Technology

1   Introduction and literature review

Often, travellers needs to be certain to arrive before a certain time, regardless of the road conditions. They therefore tend to avoid routes that can have long delays, even if such delays are seldom. This paper discusses a traffic assignment which takes risk avoidance into account. However, the probability distribution of travel times on links and the probability of incidents on links is not known to the travellers beforehand. They therefore have to make assumptions based on their risk-attitude.

The approach to risk-averse route choice behaviour presented in this paper is introduced by Bell [1]. He assumes that risk-averse drivers anticipate worst cases and minimise their exposure to these. In fact the traffic loads influence which cases are worst. Bell and Cassir [2] extend the concept using an demand-capacity relationship to determine the travel cost rather than two possible fixed costs. In this paper, a blockade means a lower capacity, resulting in a new travel time on the blocked link.

Nagae and Akamatsu [3] continue on this line of research. They point out that it might be too extreme to expect the worst case situation to happen and relax the assumption of people being completely risk-averse. They add two extra terms to spread the breakdown chances over the different scenarios. This changes the perspective on route choice behaviour slightly, but moreover, it makes the mathematical framework much easier to solve. Bell et al. [4] use the same but, compared to the strictly risk-averse simulation, they just relax the perception of link failure probabilities (and

not the route choice). The mathematical advantage then still holds.

Until now, this game theoretical approach to risk-averse assignment has only been combined with static models and not with a dynamic traffic simulation. This paper fills this gap. The risk-averse route choice model proposed here can be used in traffic assignment models. In the future, this risk-averseness might be implemented in journey planners. One can conceive of an on-line journey planner with a slide bar for risk-averseness. In general, risk-minimizing behaviour is relevant for transport of special goods or persons, such as hazardous materials or VIPs.

2 Methodology

The model proposed here extends the work of risk-averse traffic modelling[1, 2, 3, 4]. For the sake of simplicity, we formulate a single-destination model. A multi-destination network is easily fit into this model by changing the destination-specific variables into separate variable for each destination

This method aims at computing the route choice vector ($\mathbf{h}(t)$), giving the route fractions in percentages over the different paths. Therefore, at each time instant t the elements of $\mathbf{h}(t)$ add up to 1. The route choice depends on the travel time in each scenario ($\mathrm{tt}_{ijt}^{\mathrm{link}}$) and the anticipated probability of each scenario, vector $\mathbf{f}$, which size is the number scenarios. In this paper, a scenario (j) is defined as an incident blocking link j. Therefore the number of scenarios equals the number of links. In a general framework, the concept of scenarios (possible states of the network and/or demand level) can be extended, including several disruptions of the network. In the context of this paper it is important to note that the travel time on link i, $\mathrm{tt}_{ijt}^{\mathrm{link}}$ is time dependent and can be calculated using any traffic simulator. Also, influences from downstream links might influence $\mathrm{tt}_{ijt}^{\mathrm{link}}$. The total cost for travelling under scenario j, T_j, can be obtained from the traffic simulator.

The key of a risk-averse approach [1] is that a risk averse user would count on the worst scenario to happen most likely. Therefore, the following equations have to be solved:

$$\mathbf{f}^* \quad = \quad \underset{\mathbf{f}}{\operatorname{argmax}} \left(\langle T(\mathbf{f}, \mathbf{h}^*(t)) \rangle \right) \tag{1}$$

$$\mathbf{h}^*(t) \quad = \quad \underset{\mathbf{h}}{\operatorname{argmin}} \left(\langle T(\mathbf{f}^*, \mathbf{h}(t)) \rangle \right) \tag{2}$$

With Nagae and Akamatsu [3] we argue that maximisation is perhaps too extreme and even risk-averse users have a more balanced expectation of the scenarios. Due to the limited length of this extended abstract, we will here only show the solution of the system; however, the full paper will also show the derivation of this solution. The key is that the scenario which gives the worst performance (anticipated cost of travelling), gets the largest anticipation in the risk-averse users' perception. In this paper, this performance is chosen to be the travel time T. As shown [3], this slightly modified equation 1 leads to a logit-like solution for the "incident anticipation" ($\mathbf{f}$):

$$f_j = \exp\left(\vartheta T_j\right) / \sum_j \exp\left(\vartheta T_j\right). \tag{3}$$

Note the way the parameter ϑ works out in the solution: it indicates how smoothly the incident probability is distributed over the scenarios. $\vartheta = 0$ gives an equal probability to each scenario and $\vartheta = \infty$ gives only weight to the scenario with the highest disruption.

For the traffic assignment it is needed to have the anticipated link travel times, $\langle \mathrm{tt}_{it}^{\mathrm{link}} \rangle$. These can be obtained by taking a weighted average of the link travel times in each scenario:

$$\langle \mathrm{tt}_{it}^{\mathrm{link}} \rangle = \sum_j f_j \mathrm{tt}_{ijt}^{\mathrm{link}} \tag{4}$$

Using the above elements, traffic can be assigned risk-aversely in the following loop. The initial step is to start with an equal probability for each scenario, meaning all elements of $\mathbf{f}$ are equal. Then, we fix the assignment of traffic over the fastest route according to the free-flow travel time. With these initial values we start the optimisation loop. Using the traffic assignment, we calculate the link travel costs for each scenario using a dynamic traffic simulation program which gives the link travel times and the total travel times under each scenario. These, in turn, cause a new anticipation on each scenario (equation 3), thereby causing new anticipated link travel times (equation 4). With these travel times, a new one-shot traffic assignment (fastest route) is calculated which is averaged with the previous one using a Method of Successive Averages [5]. Now, a new iteration in the optimisation loop can be started. In practice, this loop converges to a solution for both the route choice and the anticipated scenario. The full paper will discuss the computation time and convergence speed, as well as methods to reduce these.

3 Results

The method is implemented and tested on a test-network, shown in figure 1, in which traffic has to go from the bottom left to the upper right corner. The network consists of a motorway (links at the left and top), secondary road (roads at the bottom and right) around links in a city centre (the middle). The capacity and the speeds have been adapted to the type of link. Most travellers would take the motorway in non risk-averse equilibrium conditions. Because of this high load, the impact of a blockade on the motorway would be high and therefore risk-averse people anticipate most on a blockade there. The distribution of the weights of the blockade is shown in figure 1a. As expected, the highest anticipation is put on the motorway link at the end. Note that a blockade at the end is more disruptive because that also blocks travellers which already passed the first link, but that have not yet passed the last link at the moment of happening. Figure 1b shows the resulting traffic assignment. This shows that not all travellers will take the motorway, which is the faster route if risk-averseness was not taken into account, indicating that adding risk-averseness changes the actual traffic assignment.
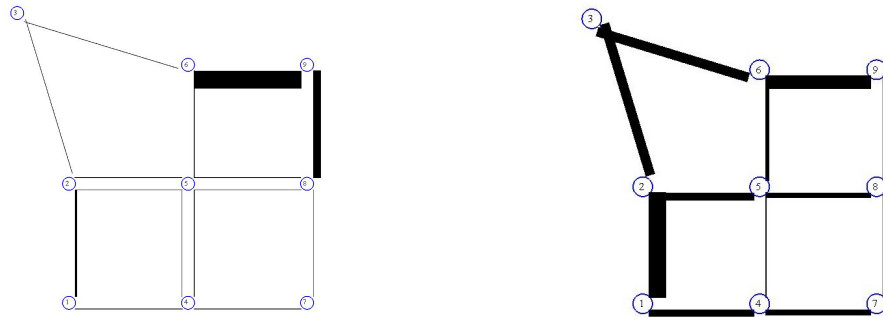
(a) The anticipated blocking probability **f** – the width of
the line indicates the anticipated blocking probability

(b) Routes chosen by the travellers, **h** – the width of the
link indicates the number of drivers taking that route

Figure 1: The results of the test network

4   Conclusions

This paper introduces dynamic queuing into risk-averse traffic simulation. It is shown that with
an innovative approach, it is possible to integrate risk-averse traffic assignment with a traffic
simulation with realistic queuing dynamics. In this simulation the influence of the incident will
expend spatially and temporally, meaning travellers will avoid also other links than the incident
site also in other times than the incident itself.

References

[1] M. G. H. Bell. A game theory approach to measure the performance reliability of transport
networks. *Transportation Research Part B*, 34(6):533–545, 2000.

[2] M. G. H. Bell and C. Cassir. Risk-adverse user equilibrium traffic assignment: an application
of game theory. *Transportation Research Part B*, 36:671–681, 2002.

[3] T. Nagae and T. Akamatsu. An Efficient Algorithm for a Maxmin Routing Model for Haz-
ardous Materials. In *Proceedings of the Third International Symposium on Transport Network
Reliability*, The Hague, the Netherlands, 2007.

[4] M. G. H. Bell, U. Kanturska, J.-D. Smöcker, and A. Fonzona. Attacker-Defender Models and
Road Network Vulnerability. *Philosophical Transactions of the Royal Society A*, 366:18931906,
2008.

[5] Y. Sheffi. *Urban transportation networks: Equilibrium analysis with mathematical programming
methods*. Prentice-Hall, Englewood Cliffs, New Jersey, 1985.

Biobjective Aircraft Route Guidance through Convective Weather

Kenneth Kuhn

Department of Civil and Natural Resources Engineering

The University of Canterbury, New Zealand

Email: kenneth.kuhn@canterbury.ac.nz

Andrea Raith

Department of Engineering Science

The University of Auckland, New Zealand

1 Introduction

Convective weather is a leading source of air travel delay. Pilots flying through areas where convective weather is present select routes aiming to minimize risk and maximize efficiency. Air traffic controllers suggest routes pilots may accept or decline, while also estimating airspace capacities, scheduling aircraft landings, and performing a host of other activities all related to considerations of risk and uncertainty. The goal of this research is to provide aircraft route guidance during periods of convective weather. This work is differentiated from past work in that the problem is explicitly modelled as a biobjective problem and solved to optimality, giving pilots flexibility to choose from a set of non-dominated routes minimizing risk and maximizing efficiency. There are many different efficient algorithms to solve such a biobjective shortest path problem to optimality [1]. There are likewise many ways to define risk including methods based on the evolution of weather patterns over short periods of time or pilot and controller reactions to given weather patterns [2].

2 Weather and Risk

The trajectories of aircraft flying through convective weather are most strongly related to two forms of data currently collected: vertically integrated liquid and radar echo top measurements [3]. Vertically integrated liquid data shows precipitation intensity by latitude and longitude at different times, and is typically transformed to a six point scale known as VIP or NWS level. Radar echo tops show cloud heights and are typically measured in thousands of feet. In much past

research, aircraft are assumed or advised not to fly through discs, squares, or other convex shapes covering all areas reporting VIP levels three and higher [4, 5]. Algorithms then select routes that are optimal in terms of distance travelled. The assumption that pilots avoid VIP level three and higher areas is common in aviation systems engineering and the various covering shapes are used to ease computational burdens.

The past research ignores empirical evidence that VIP level three holds relatively little significance for pilots; see [2, 6] and other studies. One case study found the VIP level three threshold rule of thumb "in some cases, was too conservative" and in other cases "declared as passable regions that pilots consistently avoided" [6]. Radar echo top data is actually a stronger predictor of pilot behavior than VIP level; again see [2, 6] and other studies. Actually, aircraft altitude, echo top height, and VIP data should all be considered [2]. A common, often unstated, finding of empirical studies is that it is impossible to accurately select weather conditions pilots as a group will fly through vs. those they will avoid. Different pilots will have different concerns, available options, knowledge of the weather, etc.. It is also worth noting that it is notoriously difficult to predict how weather patterns will evolve even over limited periods of time. Given all this, it is unrealistic to assume as given a four-dimensional map bifurcating airspace into areas safe and unsafe to fly through some time into the future.

3 Methodology

We model the route flown by an aircraft as a path in a flight network. The airspace is discretized into a grid, where every grid cell is represented by a node. Nodes are connected to adjacent nodes in the eight neighboring grid cells via arcs. A path in this network represents the approximation of a possible route an aircraft may follow through airspace. For a preliminary study, we assume a fixed flight level and a relatively short time period and thus obtain a 2-D flight network. It is easiest to conceptualize the 2-D problem, so we focus on that in this extended abstract. In this study, flight networks considered include hundreds of thousands of nodes.

We measure the efficiency of an aircraft route in terms of distance flown, which forms the first route choice objective. The second objective is minimizing risk along the route. A risk factor is assigned to each arc in the network, where the higher the risk factor, the less attractive an arc is. In this study, higher risks are associated with weather conditions that fewer pilots were observed flying through in the largest empirical study to date [2]. We wish to primarily capture the fact that different pilots have different tolerances for the maximum level of risk they are willing to accept in any part of their flight path. Secondarily, trajectories involving shorter paths through unattractive weather are favored. In order to achieve the desired results, we transform the results of the cited study and assign risk factors of varying orders of magnitude. For example, areas reporting

conditions that anywhere between 30 and 40% of pilots avoided are assigned one risk factor while areas reporting conditions that anywhere between 40 and 50% of pilots avoided are assigned a significantly higher risk factor. [7] also consider exposure to weather as an objective component in the form of "normalized weather intesity" but it remains unclear how the corresponding objective is formulated. Only simulated radar reflectivity showing precipitation intensity is used to determine weather cell severity but echo top measurements are not considered.

Let $n \in \mathcal{N}$ denote a node and $(i,j) \in \mathcal{A}$ with $i,j \in \mathcal{N}$ an arc in the flight network. The set $\mathcal{P}_{s,t}$ is the set of paths (or routes) in the flight network connecting origin node s to target node t. The length of an arc (i,j) is d_{ij} and its risk factor is r_{ij}. The distance traveled along a path is obtained by summing the length of the arcs, $d(p) = \sum_{(i,j) \in p} d_{ij}$, and the risk along a path is obtained correspondingly, $r(p) = \sum_{(i,j) \in p} r_{ij}$. The biobjective aircraft route choice problem is then

$$\min \quad \begin{pmatrix} d(p) \\ r(p) \end{pmatrix} \tag{1}$$
$$\text{s.t.} \quad p \in \mathcal{P}_{s,t}.$$

Our aim is to identify *efficient* routes of (1) with the property that it is not possible to obtain a route with better objective value in one component without worsening the other component. This set of paths represents the best trade off solutions between the most direct and the safest paths and thus constitute a route choice set for pilots – it is likely that pilots will choose one of these routes, but which route is chosen may depend on pilot preference and experience.

While a similar problem has been approximately solved using heuristics [7], there does not seem to be a need to do so as several exact algorithms as discussed in [1] are capable of quickly identifying all efficient solutions. For one example instance involving weather reported around Atlanta, Georgia on 5 May, 2007 at 11:00 GMT, we create a flight network with 122304 nodes and 974236 arcs which is considerably larger (more than two orders of magnitude) than the one considered in [7] and also based on real weather data. To solve (1) for this flight network, we use a biobjective label setting algorithm which extends the single-objective label setting algorithm, also known as Dijkstra's algorithm, to the biobjective case. We are able to identify all efficient solutions with this biobjective label setting algorithm on a standard desktop computer within 1 second without taking advantage of any speedup techniques or network preprocessing. Some of the obtained efficient paths are shown in Figure 1, where origin and destination are circled. The route shown in white is the most direct one, the left-most route shown in black is the safest one and the other routes shown in green have intermediate safety and distance values.

In this study, actual flown trajectories have been compared to model recommendations, revealing both that our approach selects realistic routes and that different pilots have different behaviour when trading-off risk and efficiency. Subsequent work will develop a decision-support tool for air traffic controllers based upon the research presented here.
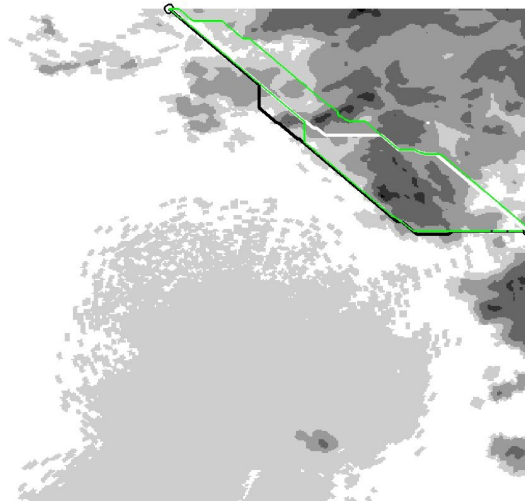
Figure 1: Risky weather and efficient flight paths.

References

[1] A. Raith and M. Ehrgott, "A comparison of solution strategies for biobjective shortest path problems", *Computers and Operations Research* 36, 1299-1331 (2009).

[2] K. Kuhn, "Analysis of Thunderstorm Effects on Aggregate Aircraft Trajectories", *Journal of Aerospace Computing, Information, and Communication* 5, 108-119 (2008).

[3] D. Rhoda and M. Pawlak, "The Thunderstorm Penetration/Deviation Decision in the Terminal Area", AMS Conference on Aviation, Range, and Aerospace Meteorology, Dallas, 1990.

[4] J. Prete and J. Mitchell, "Safe Routing of Multiple Aircraft Flows in the Presence of Time-Varying Weather Data", AIAA Conference on Guidance, Navigation and Control, Providence, 2004.

[5] J. Pannequin, A. Bayen, I. Mitchell, H. Chung, and S. Sastry, "Multiple Aircraft Deconflicted Path Planning with Weather Avoidance Constraints", AIAA Guidance, Navigation and Control Conference, Hilton Head, 2007.

[6] R. DeLaura and S. Allan, "Route Selection Decision Support in Convective Weather: A Case Study of the Effects of Weather and Operational Assumptions on Departure Throughput", Eurocontrol FAA ATM R&D Seminar, Budapest, 2003.

[7] S. Alam, M.H. Nguyen, H.A. Abbas, and M. Barlow, "Ants Guide Future Pilots", in *Progress in Artificial Life*, 36-48, Springer-Verlag, Berlin Heidelberg, 2007.

Development of Stated-Preference Survey System on the Combined WEB and GPS Mobile Phones

Takahiko Kusakabe

JSPS Research Fellow, Department of Civil Engineering

Kobe University

Rokkodai 1-1, Kobe, 657-8501, Japan

Email: t.kusakabe@stu.kobe-u.ac.jp

Kenichiro Sadakane

Department of Civil Engineering

Kobe University

Ippei Yamanaka

Department of Civil Engineering

Kobe University

Yasuo Asakura

Department of Civil Engineering

Kobe University

1 Introduction

In recent years, automakers and other manufacturers are developing new vehicles for individual travellers, called as "Personal Mobility" such as Segway, Toyota's i-REAL and so on. The personal mobility is expected to be adopted as a new urban travel mode. In the following part of this paper, a personal travel mode is referred as the Low Speed Private Travel Mode (LSPTM) (see [1]). Various aspects of operational strategies should be discussed when the LSPTM is implemented in real urban areas. For example, vehicle sharing systems can be the alternative of the conventional ownership systems. The most important issue is to know how the LSPTM is accepted by travellers in their wide-ranging urban activities.

Stated preference (SP) survey is often conducted to obtain travellers' preferences for transport modes under hypothetical situations. The SP survey can be adapted to investigate the transport systems

in the future that have not been realized in the target area. However the SP survey is not always sufficient to reflect actual situations that each examinee will experience. This point may cause biases to the results of the SP survey. In order to reduce those biases, previous studies have combined SP data with revealed-preference (RP) data under actual travel choice situations. Some of these studies created the questions of SP survey depending on the traveller's revealed behaviour in the RP data (for example [2]). It is expected that examinees can answer the questions of SP survey more accurately in the combined survey as they remember their experience in actual situations. Conventional studies used interviews, web questionnaires, or paper-based questionnaires to obtain RP data. Data collection is, however, not easy when travel choice behaviour is observed under wide-ranging urban activities.

A probe person (PP) survey was developed as an effective method to observe travellers' behaviour (See [3]). During the PP survey, every examinee is asked to carry a mobile phone equipped with GPS. GPS mobile phones observe all the trajectories of the trips made by the examinee. These trajectory data can be used to find out the target trips that include the situations of travel choices for the SP survey.

The aim of this study is to develop the data collection system for the travel choice survey using web based SP questionnaire based on the PP system. In this paper, we show the survey system and explain how the survey was conducted to observe the SP data of the mode choice between walk and the new travel mode.

2 Survey System

The survey system consists of three parts; the PP system, the web-diary system, and the web questionnaire system. The PP system is to observe traveller's trajectory of each trip using GPS mobile phones and record it onto database. The web-diary system is an online system in which each examinee can write additional characteristics on their trips. The information consists of actual place of origin, destination, travel mode, and trip purpose. Following the web diary system, the web questionnaire system retrieves the trips that satisfy the conditions for the SP survey. Trip purposes, origins, destinations, travel modes, and areas can be set as those conditions of the retrieval. The system creates web questionnaires corresponding to the retrieved trips.

3 Empirical Surveys

An empirical survey was conducted from 7th through 13th of September 2009. The preference data of the travel mode choice for the LSPTM were collected in the central area of Kobe in Japan. The target area covers 1.9 km from east to west and 3.3 km from north to south. There are 100 examinees,

selected from people who live in Kobe or its neighbourhood areas and who have plans to visit the Kobe downtown during the survey period.

Two types of SP survey were carried out in order to compare the results of the developed method with the conventional methods. The one is the combined method of PP plus WebSP. The other is a stand-alone SP survey for several scenarios of hypothetical trips. The target trips of the PP+WebSP are the trips including more than 5 minutes walk in the target area. In both surveys, the examinees were asked to answer questions: "Would you have switched to the LSPTM from walk if the LSPTM were available in the displayed conditions?" The conditions of the LSPTM consist of the access distance, waiting time, travel time and fare. The access distance is the distance from the observed starting place of a walking trip to the LSPTM pool. The waiting time is the time for waiting for the LSPTM at the pool. The travel time is the time from the pool to the observed destination of the walk. The fare is the charge of one time use of LSPTM. In addition, the stand-alone SP survey also specified the conditions of the steepness and weather. Note that the questions are repeated 5 times for each trip with different conditions of the LSPTM in the PP+WebSP. In the stand-alone SP, they are repeated 12 times.

2173 trips were observed by the PP+Web diary system though the trajectories of 312 trips are not correctly observed mainly because of GPS errors. The remaining 1861 trips were the candidates of the retrieval for the web questionnaires. The trips less than five minutes walk in the target area were excluded. Note that the trips that had the same route as the previous trips of an examinee were also excluded to reduce the examinees' burden. There were 85 trips remained and used to the PP+WebSP. There were 94 examinees who responded to the stand-alone SP survey. As the result, 425 samples of mode choice data were collected by the PP+WebSP and 1128 samples were collected by the stand-alone SP.

The logit model is applied for both the PP+WebSP data and stand-alone SP data. The parameters of the logit models are estimated with each data set separately, as well as with both data sets together. When the parameters are estimated from combined data sets, the variances of random terms of the utility functions in the PP+WebSP model and stand-alone SP model are assumed as:

$$Var(PP + WebSP) = \mu Var(SP) \qquad \text{where } \mu \text{ is the scale parameter} \qquad (1)$$

Table 1 shows the results of the estimation. The coefficients of the utility functions are represented in the leftmost columns (Each coefficient are included in the utility functions that are shown in the parentheses). As the results of the standard logit models, the PP+WebSP model has larger $\bar{\rho}^2$ than that of the stand-alone SP. The results of the combined model show that the scale parameter is significantly smaller than one. These results show that the PP+WebSP data has smaller random noises than that of the standard SP data.

Table 1. Estimation Results (t-statistics in parentheses)

Coefficient	PP+WebSP	Stand-alone SP	Combined Model
Constant (LSPTM) (PP+WebSP)	0.076 (0.131)		-0.060 (-0.205)
Constant(LSPTM) (SP)		-0.438 (-1.522)	-0.313 (-1.003)
Access Distance [m](LSPTM)	-0.021 (-4.862)	-0.010 (-5.396)	-0.013 (-5.395)
Waiting Time[min.](LSPTM)	-0.300 (-3.843)	-0.178 (-5.031)	-0.223 (-5.064)
Fare [yen](LSPTM)	-0.021 (-4.916)	-0.014 (-7.461)	-0.017 (-6.212)
Travel Time [min.](LSPTM)	-0.211 (-2.614)	-0.165 (-5.780)	-0.193 (-5.099)
Travel Time [min.](Walk)	-0.263 (-4.746)	-0.187 (-9.782)	-0.218 (-6.844)
Weather Dummy (LSPTM)		0.728 (5.022)	0.859 (4.122)
(Rain :1 Otherwise: 0)			
Steep Dummy (LSPTM)		0.611 (3.814)	0.711 (3.267)
(Up-hill:1 Otherwise:0)			
Scale Parameter μ			0.827 (6.589)
N	425	1128	1553
$L(0)$	-294.59	-781.87	-1076.46
$L(\bar{\beta})$	-142.32	-599.09	-766.12
ρ^2	0.517	0.234	0.288
$\bar{\rho}^2$	0.510	0.232	0.286

4 Conclusions

This study developed the survey system that aims to obtain SP data for analysing the newly introduced private travel mode: LSPTM. The SP data were obtained from the questionnaires based on the PP survey. The empirical analysis shows that the results of the SP survey can be improved by using developed system.

References

[1] Y. Asakura, T. Iryo, I. Yamanaka, K. Yamabe, "Low Speed Private Travel Mode (LSPTM) for Saving Energy Consumption of Urban Transport Systems", International Workshop on Recent Advances in Transport, Marrakech, Morocco, May 2008.

[2] D. Hensher and J. Rose, "Development of commuter and non-commuter mode choice models for the assessment of new public transport infrastructure projects: A case study", *Transportation Research Part A* 41, 428–443 (2007).

[3] Y. Asakura and E. Hato, "Tracking Individual Travel Behaviour using Mobile Communication Instruments", *Transportation Research Part C* 12, 273–291 (2004).

Traffic breakdowns and freeway capacity as extreme value statistics

Reinhart Kühne

Transportation Studies

German Aerospace Center

Axel Lüdtke

LUAX Software Engineering

Corresponding author:

Reinhart Kühne

Transportation Studies

German Aerospace Center, Rutherfordstr. 2; Berlin; Germany

Email: reinhart.kuehne@dlr.de

1 Extended Abstract

Analyzing probability distributions from traffic volume time series and calculating the first passage time distributions gives the probability of firstly exceeding a given threshold corresponding to a congestion of given scenarios. The method interprets traffic breakdown as extreme events.

Stochastic traffic dynamics can be described as equation of motion for a collective variable, the vehicle cluster size or including fluctuations. The corresponding Langevin equation can be transformed into stochastic differential equation (Fokker Planck Equation) where not only the equilibrium distribution can be deduced but also the time can be calculated when firstly exceeding a given extreme threshold (first passage time). This results is a consequence of statistics of extreme events and opens new insights in probabilistic description and prognosis of the traffic breakdowns.

Three different traffic situations can be distinguished:
(a) Stable traffic flow where any fluctuations decay over time
(b) metastable traffic flow where fluctuations neither decay nor grow and
(c) unstable traffic flow where a breakdown can be expected for sure if the observation time is long enough.

This first passage time gives the time when reaching a critical extreme value. The probability of finding a congestion of a given critical length is calculated by the temporal drop of the probability of finding a congestion anywhere between critical length to zero and describes the probability change as the probability outflow over the boundary "critical length". Treating the congestion length or cluster size n as a continuous variable and expand the balance equation, translates the jam formation into a Brownian motion in a potential which shows the above mentioned different scenarios where the system state tends either to n free traffic flow shows bistability or tends to completely congested traffic.

The traffic dynamics is translated into a first passage time distribution. This describes the distribution of time periods observing for the first time the formation of a traffic jam of a certain length or number of vehicles. The distribution contains a time lag, a maximum corresponding to a time period of a Brownian motion drift reaching the critical jam length, and a tail describing exceptional long waiting times for jam formation. The corresponding stochastic movement of the state variable described by the Fokker-Planck equation is compared with the frequency of breakdowns for different traffic volumes data observed on the autobahn München Holledau with and without speed control. The corresponding stochastic differential equation starts with the discrete balance equation for cluster formation and the associated Fokker-Planck equation which is simplified by a diffusion approximation.

For this aim we consider a straight traffic flow on a freeway section and study the spontaneous formation of a jam regarded as a large car cluster arising on the road. To get rid of some boundary conditions like entries and exits we can idealize the section by a circular road of length L with N cars moving on it. All the cars are assumed to be identical vehicles and can form two phases. One of them is the set of freely moving cars and the other is the congestion called single car cluster. The cluster is specified by its size n, the number of aggregated cars. Its internal parameters, namely, the headway distance and, consequently, the speed of cars in the cluster are treated as fixed values independent of the cluster size n. We note that in the model under consideration there can be only one cluster on the road. The free flow phase is specified also by the corresponding headway distance that, however, depends strictly speaking on the car cluster size n. The larger the cluster is, the less is the number (N - n) of the freely moving cars and therefore the larger is the headway distance. When a vehicular cluster arises on the road its further growth is due to the attachment of the free cars to its upstream boundary, whereas the cars located near its downstream boundary accelerate to leave it, which decreases the cluster size. These processes are treated as random changes of the cluster size n by ± 1 and the cluster evolution is described in terms of time variations of the probability function P(n,t) for the cluster to be of size n at time t. From the probability density the cumulative distribution of observing a breakdown within a given observation time can be calculated and fitted to a Weibull distribution as typical distribution for the probability of surviving.

The cumulative first passage time distribution can be interpreted as breakdown probability distribution for a given traffic volume. It outlines corresponding probability when reaching a

breakdown in an assumed observation time. It leads directly to the probabilistic definition of the capacity as a traffic volume leading to an unstable traffic pattern with a given probability within a given observation time. This definition can substitute the existing definitions for the capacity of a freeway and opens the possibility to quantitatively describing the influence of traffic control systems on the traffic flow.

References

1. BRILON, W.; ZURLINDEN, H. (2003). Überlastungswahrscheinlichkeiten und Verkehrsleistung als Bemessungskriterium für Straßenverkehrsanlagen. Forschung Straßenbau und Straßenverkehrstechnik, Heft 870, Bundesministerium für Verkehr, Bau- und Wohnungswesen, Abt. Straßenbau, Straßenverkehr Bonn (Hrsg.).

2. GARDINER, C.W. (2002). *Handbook of Stochastic Methods,* Springer, Berlin, Heidelberg.

3. JÁNOSI, I.M.; GALLAS, J.A.C. *Growth of companies and water-level fluctuations of the river Danube,* Physica A 271 (1999) 448-457

4. KÜHNE, R. (2001). Application of Probabilistic Traffic Pattern Analysis to Capacity, *Siam Conference, San Diego, July 9-19, 2001*

5. KÜHNE, R., and ANSTETT, N. (1997). New Methods for Determining Critical Sections on Freeways, *Annual Meeting of the Transportation Research Board, Washington D.C..*

6. KÜHNE, R.; MAHNKE, R.; LUBASHEVSKY, I.; and KAUPUŽS, J. (2002). Probabilistic Description of Traffic Breakdowns, *Physical Review E, vol. 65, 066125.*

7. KÜHNE, R.; MAHNKE, R. (2005). Controlling traffic breakdowns *In*: MAHMASSANI, H.S. (ed), *Transp. and Traffic Theory*, Proc. of the 16th Int'l Symp. on Transportation and Traffic Theory, pp. 229-244.

8. KÜHNE, R.; SCHICK, P. (2001). Untersuchungen zum Verkehrsablauf an Streckenbeeinflussungsanlagen (Investigations of traffic flow at corridor control systems), *In: Veröffentlichungen aus dem Institut für Straßen- und Verkehrswesen, Universität Stuttgart, Heft 28, 08/2001,* Stuttgart 2001

9. LORENZ, M. and ELEFTERIADOU, L. (2000). A Probabilistic Approach to Defining Freeway Capacity and Breakdown. *In*: BRILON, W. (ed) *4th Int. Symp. on Highway Capacity, Transp. Research E Circular EDCO/8.*

10. MAHNKE, R.; and PIERET, N. (1997). Stochastic Master-Equation Approach to Aggregation in Freeway Traffic, *Physical Review E, vol. 56, 2666.*

11. MAHNKE, R; and KAUPUŽS, J. (1999). Stochastic Theory of Freeway Traffic, *Physical Review E, vol. 59, 117.*

12. MUTHLY, D.N.P.; XIE, M.; JIANG, R. (2004). *Weibull Models,* Wiley, New Jersey, 2004.

13. REGLER, M. (2004). Verkehrsablauf und Kapazität auf Autobahnen (*Traffic Flow and Capacity on Freeways)*, Dissertation, Universität Bochum.

14. RISKEN, H. (1996). *The Fokker-Planck Equation*, Springer, Berlin, Heidelberg, 1996

15. SCHICK, P. (2003). Einfluss von Streckenbeeinflussungsanlagen auf die Kapazität des Verkehrs von Autobahnabschnitten sowie die Stabilität des Verkehrsflusses *(Influence of corridor control systems on the capacity of freeway sections and on traffic flow stability)*, Dissertation, Universität Stuttgart.

16. ZURLINDEN, H. (2003). Überlastungswahrscheinlichkeiten und Verkehrsleistung als Bemessungskriterium für Straßenverkehrsanlagen, *FE-Nr.: 03.327/1999/ KGB der Bundesanstalt für Straßenwesen.*

17. KÜHNE, R. (2008). Statistics of extrems, natural disasters and traffic jams. 5th European Conference on Complex Systems ECCS 2008; 10.-19. September 2008, Jerusalem, Israel

Bilevel programming and price optimization problems

Martine Labbé

Department of Computer Science

Université Libre de Bruxelles

Email: mlabbe@ulb.ac.be

Consider a general pricing model involving two levels of decision-making. The upper level (leader) imposes prices on a specified set of goods or services while the lower level (follower) optimizes its own objective function, taking into account the pricing scheme of the leader. This model belongs to the class of bilevel optimization problems where both objective functions are bilinear. See e.g. [1] for an overview of bilevel optimization.

Let x and y be real vectors that specify the levels of taxed and untaxed activities and T be a tax vector attached to the activity vector x. Also, let c represent the "before tax" cost vector of the activity vector x and d the cost vector of the activity vector y. For a given tax level vector T, in control of the leader, the follower strives to minimize its operating costs $(c + T)x + dy$, while the leader seeks to maximize its revenue Tx from taxes. The bilevel program can be written as follows:

$$
\begin{aligned}
\max_T \quad & Tx \\
\min_{x,y} \quad & (c + T)x + dy \\
s.t. \quad & (x, y) \in \Pi
\end{aligned}
$$

In this talk, we review this class of hierarchical problems from both theoretical and algorithmic points of view and then focus on some special cases. Among others, we present complexity results, identify some polynomial cases and propose mixed integer linear formulations for those pricing problem.

In the first problem, tolls must be determined on a specified subset of arcs of a multicommodity transportation network. In this context the leader corresponds to the profit-maximizing owner of the network, and the follower to users travelling between nodes of the network. The users are assigned to shortest paths with respect to a generalized cost equal to the sum of the actual cost of travel plus a money equivalent of travel time. This Network Pricing Problem has been considered,

among others, by [2], [3], [4], [5] and [6].

An extension of the Network Pricing Problem is obtained by optimizing the design of the network and the set of tolls on a subset of open arcs, given that users travel on shortest paths, see [7].

The third problem is a special case of the Network Pricing Problem in which the taxable arcs are connected and form a path, as occurred in toll highways. When users travel on at most one taxable subpath, the problem can be reformulated as a Network Pricing Problem on an auxiliary clique graph, [8]. Interestingly this problem is also equivalent to that of determining optimal prices for bundles of products given that each customer will buy the bundle that maximizes her/his own utility function, [9].

References

[1] B.Colson, P. Marcotte and G.Savard , "An overview of bilevel optimization", *Annals of Operations Research* 153, 235-256 (2007).

[2] M. Labbé, P. Marcotte and G. Savard, "A bilevel model of taxation and its application to optimal highway pricing", *Management Science* 44, 1608-1622 (1998).

[3] L. Brotcorne, M. Labbé, P. Marcotte, and G. Savard ,"A Bilevel Model for Toll Optimization on a Multicommodity Transportation Network", *Transportation Science* 35, 345-358 (2001).

[4] S. Roch, P. Marcotte and G. Savard,"Design and analysis of approximation algorithm for Stackelberg network pricing", *Networks* 46, 57-67(2005).

[5] M. Bouhtou, S. van Hoesel and A. van der Kraaij,"Tariff optimization in networks", *Informs Journal on computing* 19, 458-469 (2007).

[6] S. Dewez, M. Labbé, P. Marcotte, G. Savard , "New formulations and valid inequalities for a bilevel pricing problem", *Operations Research Letters* 36,141-149 (2008).

[7] L. Brotcorne, M. Labbé, P. Marcotte and G. Savard, "Joint Design and Pricing on a Network", *Operations Research* 56, 1104-1115 (2008).

[8] G. Heilporn, M. Labbé, P. Marcotte, G. Savard, "A polyhedral study of the Network Pricing Problem with Connected Toll Arcs", *Networks*, to appear.

[9] G. Heilporn, M. Labbé, P. Marcotte and G. Savard, "A parallel between two classes of pricing problems in transportation and marketing", *Journal of Revenue and Pricing Management*, to appear.

Heuristic column generation for railroad track inspection scheduling

Sébastien Lannez[1,2,3]   Christian Artigues[2,3]   Jean Damay[1]

Michel Gendreau[4]

[1] SNCF I&R/A^{2}D, 45 rue de Londres, 75008 Paris, France,

[2] Université de Toulouse ; UPS, INSA, INP, ISAE ; LAAS ; F-31077 Toulouse, France

[3] CNRS ; LAAS ; 7 avenue du colonel Roche, F-31077 Toulouse, France

[4] CIRRELT, Université de Montréal, C.P. 6128, Montréal (Québec), H3C 3J7 Canada

Email: sebastien.lannez@sncf.fr

1 Introduction

The French railway infrastructure manager RFF[1] delegates some maintenance tasks to the French railroad company SNCF[2]. One of them is rail defectoscopy, the detection and survey of defects inside the rails. These actions are performed with ultrasonic propagation analysis. The more loaded the railroad tracks are, the more frequently they must be checked. With the increase in railroad traffic, the need to improve the schedule of these inspection becomes a prevalent task.

Inspection frequencies range from 6 months to 20 years. 2/3 of the total inspection distance is due to tracks that should be visited once or twice a year. These tracks are called *primary tracks*. They constitue a yearly homogeneous subnetwork. The remaining inspections (*secondary tracks*) are planned each year.

The problem we are facing is to visit a given set of tracks tacking into account inspection tasks time windows, track outages and vehicle speeds. Furthermore, vehicle speeds depend on its type and circulation mode (either inspecting or not). These vehicles have limited working capacity defined by the amount of water which can be brought on board. This water is needed to keep coupled sensors and rails. For organisational purposes, tanks can only be refilled at the end of a shift. The objective is to minimise the total deadhead distance. We named this problem the Railroad Track Inspection Scheduling Problem (RTISP).

---

[1]http://www.rff.fr

[2]http://recherche.sncf.fr

2   Literature review

The railroad track inspection scheduling problem (RTISP) can be seen as a general arc routing problem. It consists in visiting, with a heterogeneous fleet, a given set of arcs during valid time windows. Good introductions to arc routing problems are the book (1) and the reviews (2), (3).

The water capacity constraint can be modelled by using a capacitated arc routing problem (CARP)(4), which consists of visiting arcs with vehicles of limited capacity. The heterogeneous capacitated arc routing problem (H-CARP) generalises it by allowing different velocity and capacity characteristics for each vehicle type. A common variant is the one with time windows on arcs (CARP-TW). In the capacitated arc routing problem with intermediate facilities (CARP-IF), vehicles can be refilled at given points at any time. The RTISP is a generalisation of all these problems. It has time windows defined on arcs, vehicles and nodes (refill). Furthermore, two limited capacities are constraining daily circulation: shift duration and water tank size. As the authors of (5) notified, industrial vehicle routing problem are rich, models to solve them are often a generalisation of academic works and input data size can be huge. The RTISP is a good illustration of this fact.

Related industrial problems have been solved in the literature. Some of them are about road weeding, winter road maintenance, waste collection or postal delivery.

3   Model

Train unit circulation are modelled by a graph which contains arcs and edges representing either inspection tasks, deadhead traversals or complex moves in a train station. Arcs are used when the railroad track is unidirectional whereas edges are used when the railroad track has bidirectional equipment. Nodes describe stations and communications between railroad tracks.
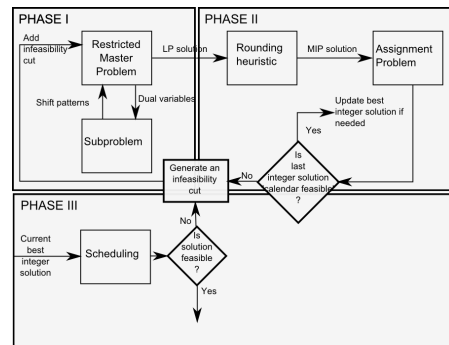
Because refill can only be performed at the end of a shift, every shift is constrained to start and end at a refill station. Hence, every shift is a trip between two refill stations having a total distance to inspect lower than the capacity of the water tank, and a total trip duration lower than the duration of a work shift. Given all the feasible shift paths, the RTISP becomes the problem of selecting and scheduling them in order to satisfy all inspections with the lowest total deadhead distance.

Train unit maintenance rendezvous, vehicle reservation and track outage minimum duration are set to one shift for organisational purposes. Inspection task duration can vary from minutes to hours. Vehicles can be constrained to move on a subpart of the complete railroad network.

4   Column generation based heuristic

The proposed algorithm is based on a mathematical decomposition which is heuristically solved in three phases. During Phase I simple tasks are aggregated into work shifts with the use of a column generation algorithm. The generated continuous solution is used as an incumbent for Phase II. In Phase II a rounding greedy heuristic is used to get an integer solution. This new integer candidate solution is tested against calendar day assignment to check if all selected shifts can be assigned to a calendar day. If it is not, a cut is generated. If it is, the new candidate solution will be used to generate a constraint program for a complete feasibility test during Phase III. This last stage is about testing if scheduling the set of work shifts is feasible. If this test fails, a cut is added to the master problem. Otherwise, a solution with minimum total deadhead traversal distance is selected.

5 Conclusion

In this paper, we presented the railroad track inspection problem and proposed a mixed integer program formulation for it. To tackle its resolution we choose to solve it by an original column and cut generation framework. The computational tests, which has been performed on the collected data of the year 2009 (750 nodes, 1340 arcs, 752 edges, 2 vehicles), highlight the quality of the solution obtained after 4 hours of computing. The algorithm successed in scheduling the 700 tasks of year 2009. The goal of attaining a performance ratio (inspected distance / total distance) greater than 50% has been achieved. A team of experts are actually validating these results.

References

[1] Moshe Dror, editor. *Arc Routing: Theory, Solutions and Applications*. Springer, 2000.

[2] H. A. Eiselt, Michel Gendreau, and Gilbert Laporte. Arc Routing Problems, Part I: The Chinese Postman Problem. *Operations Research*, 43(2):231–242, 1995.

[3] H. A. Eiselt, Michel Gendreau, and Gilbert Laporte. Arc Routing Problems, Part II: The Rural Postman Problem. *Operations Research*, 43(3):399–414, 1995.

[4] Bruce L. Golden and Richard T. Wong. Capacitated arc routing problems. *Networks*, 11(3):305–315, 1981.

[5] Geir Hasle and Oddvar Kloster. *Geometric Modelling, Numerical Simulation, and Optimization*, chapter Industrial Vehicle Routing, pages 397–435. Springer Berlin Heidelberg, 2007.

A Game Theoretic Framework for the Robust Railway Transit Network Design Problem

Gilbert Laporte

University of Montreal (Canada)

Juan A. Mesa

University of Seville (Spain)

Federico Perea*

University of Zaragoza (Spain)

perea@us.es

1 Introduction

This paper proposes a game theoretic framework for the problem of designing an uncapacitated railway transit network in the presence of link failures and a competing mode. It is assumed that when a link fails, another path or another transportation mode is provided to transport passengers between the endpoints of the affected link. The goal is to build a network that optimizes a certain utility function when failures occur. The problem is posed as a non-cooperative two-player zero-sum game with perfect information. The saddle points of the corresponding mixed enlarged game yield robust network designs.

The design or the extension of a *Railway Transit Network* (RTN) is a primary concern in many cities. The reduction of traffic congestion, travel times, energy consumption and pollution justifies investment in these networks, typically metro systems and suburban railways. A number of studies have been devoted to the design of metro and suburban train networks in a deterministic context. Unfortunately, railway networks do not always work as expected because of uncertain input data or unforeseen events. In addition, such networks cannot easily be modified at short notice once they are in place. Therefore, one would like the RTN to optimally work under many possible scenarios, that is, it should be *robust*. Robustness is a difficult term to define because its meaning highly depends on the context to which it applies. According to the Institute of Electrical and Electronic Engineers, robustness can be defined as the degree to which a system or component can function correctly in the presence of invalid inputs or stressful environmental conditions. In railway planning, robustness is generally considered with respect to fluctuations of input parameters (e.g. demand), to disturbances or disruptions, and to integration with other planning phases.

In transportation network design problems three agents are involved: a planner, the users and

an evil agent called *demon*. The planner usually optimizes general goals, such as trip coverage or total travel time. The users seek to optimize individual utilities such as comfort, travel time and cost. The demon is unpredictable and makes the system work suboptimally by means of bad weather conditions, natural disasters, human errors, etc. User behavior is normally described through equilibrium models, which have been widely analyzed in the transportation literature. This paper differs from previous studies on robust transportation network design in a number of ways. Since we restrict our analysis to the design of railway networks, the following two assumptions can be made: the travel times of the railway network to be built are minimally affected by congestion (capacity is assumed to be sufficient), and users take the fastest route, either on the railway network or by using an alternative transportation mode. These assumptions allow us to simplify our analysis since user behavior is known. Therefore, from a game theoretic point of view, only two agents are acting in our problem: the planner and the demon. We wish to design RTNs that will react well to link failures. More specifically, our goal is to ensure that the network will be operative for the largest possible number of passengers in case of failure. Given a known link failure probability distribution, we can optimize the expected utility of the network. This problem is called the *Probabilistic Railway Network Design* (PRND) problem. A generalization of this situation arises when the probability distribution of link failures is unknown. One way of tackling the latter problem is to consider the worst case scenario, that is, to maximize the utility of the network when the link that most reduces the utility fails. The solution to this problem is a *maxmin network*. This problem constitutes a stochastic version of the PRND problem and is called the *Stochastic Railway Network Design* (SRND) problem, which we model in this paper as a non-cooperative two-person zero-sum game with perfect information, where the first player is the operator and the second is a demon who wants to attack some of its links. The payoff matrix of this game is the utility of the network built by player I when the link chosen by player II fails.

2 Models

We now formulate the PRND and the SRND problems. Let $K(r)$ be the utility function to be maximized (e.g. trip coverage) of network $r \in R$, where R is the set of all feasible networks that can be built. We model the physical network by a graph, its stations by nodes and its links by edges. In what follows, the notation i may refer to a station or to a node, similarly e may refer to a link or to an edge. We denote by E the set of all edges of the graph, and we assume that any of these edges can fail. The utility of the network is adversely affected by link failures. Let $K(r, \Delta)$ be the utility function of network $r \in R$ when all links in $\Delta \subset E$ fail. Note that $K(r, e) = K(r)$ if link e does not belong to network r. We assume that any link can fail and we consider that no more than one link can fail at the same time.

If the probability that edge e fails is known and equal to γ_e, the construction of a robust network is a probabilistic problem and can be solved by optimizing the expected utility under all possible scenarios (no failure scenario and single link failure scenarios), that is, by finding a solution to

$$\max_{r \in R}\{(1 - \sum_{e \in E} \gamma_e)K(r) + \sum_{e \in E} \gamma_e K(r,e)\}. \qquad \text{(PRND)} \qquad (1)$$

Unfortunately, the probability distribution of failures is normally unknown, which brings more uncertainty into the problem. Our way of dealing with this uncertainty is to seek an RTN that maximizes the utility function under a worst case scenario, that is, when the link that most affects the utility of the network fails. Therefore, our problem becomes

$$\max_{r \in R} \min_{e \in E} K(r,e). \qquad \text{(SRND)} \qquad (2)$$

If K is to be minimized, for example when it represents a disutility like the total travel time, the problem to be solved is $\min_{r \in R} \max_{e \in E} K(r,e)$.

In general, the problem defined by (2) can be expressed as

$$
\begin{aligned}
\text{maximize} \quad & z \\
\text{subject to} \quad & K(r,e) \geq z, \quad e \in E \\
& r \in R.
\end{aligned}
\qquad (3)
$$

The output of the model is a robust RTN consisting of a number of railway lines and maximizing the minimum trip coverage under the failure of one link. The complexity and size of the problem defined by (2) make it difficult to solve. Our practical approach consists of reducing the number of feasible networks under consideration. This makes sense since the maxmin optimal network may not have a competitive trip coverage when failures do not occur (which is the usual case). Therefore we will impose that the resulting network should have at least some proportion of the trip coverage corresponding to the optimal network. The idea is to repeatedly solve the RND problem described in [1], because it is faster to solve than (3). This can be justified since RND is a particular instance of SRND and has fewer variables and constraints. Once we have built a set of "good" networks, we choose one maximizing the minimum utility when a link fails, that is, a solution to the SRND problem.

3 A Game Theoretic Approach

The main novelty of this paper is to show that the SRND problem defined as a maxmin problem in (2) and (3) can be modelled as a non-cooperative two-player zero-sum game with perfect information. In our problem, the players cannot make agreements and what one player gains is lost by the opponent. Player I is the RTN designer and operator and has as many (pure) strategies $r \in R$ as

the number of networks that can be built. Player II, a demon, has as many (pure) strategies $e \in E$ as the number of links that can fail. General non-cooperative two-player zero-sum games can be simply represented by a matrix $A = (a_{ij})$, where a_{ij} is the payoff obtained by player I if he chooses his i^{th} strategy and player II chooses his j^{th} strategy. Player II then obtains a payoff equal to $-a_{ij}$. In the SRND problem player I wants to maximize his utility by building an RTN, while player II aims at minimizing it, by making a link fail. Therefore, the payoff of player I is the utility function of the network built if the edge chosen by player II is inoperative, whereas the payoff of player II is the opposite. Denoting by $K(r, e)$ the utility of network $r \in R$ when edge $e \in E$ fails, a pair of strategies, (r^*, e^*) is called a *saddle point* if $K(r, e^*) \le K(r^*, e^*) \le K(r^*, e)$, $\forall r \in R$, $\forall e \in E$. Thus, $K(r^*, e^*)$ is the guaranteed trip coverage for player I against any edge player II chooses. It is also the maximum damage that player II can inflict to any network player I constructs. If the game has a saddle point, then it satisfies

$$K(r^*, e^*) = \max_{r \in R} \min_{e \in E} K(r, e) = \min_{e \in E} \max_{r \in R} K(r, e), \tag{4}$$

and (r^*, e^*) is a Nash equilibrium strategy, which means that no player can benefit by changing its strategy unilaterally.

If no saddle point exists it is possible for players to enlarge the available set of strategies by considering probability vectors, and look for a saddle point in the enlarged game, in which players can choose a convex combination of their pure strategies, thus defining a *mixed strategy*. The mixed strategy $r_\beta = r_{(\beta_1,...,\beta_{|R|})}$ for player I means that he builds network r_i with probability β_i, for every $i = 1, \dots, |R|$. Analogously, the mixed strategy $e_\gamma = e_{(\gamma_1,...,\gamma_{|E|})}$ of player II means that he will attack link e_i with probability γ_i. The Von Neumann minmax theorem ensures that there always exists such a saddle point in the enlarged game. Note that the saddle point of the enlarged game leads to a better expected payoff for player I than the strategy obtained by solving the $\max_r \min_e K(r, e)$ problem. This mixed-strategy Nash equilibrium yields an expected utility that does not depend on the actual edge failure distribution and, therefore, the solution is truly "robust".

The reader may note that we have only considered the single-failure case. A simple extension of the models proposed gives us the possibility to deal with multiple failures, although the complexity of the corresponding problems dramatically increases. For more details, see [2].

References

[1] Laporte, G., Marín, A., Mesa, J. A. and Perea, F., "Designing Robust Rapid Transit Networks with Alternative Routes", *Journal of Advanced Transportation* (In Press), (2010).

[2] Laporte, G., Mesa, J. A. and Perea, F., "A Game Theoretic Framework for the Robust Railway Transit Network Design Problem", *Transportation Research B* (In Press), (2010).

1. Midnight speaker:
Gilbert Laporte
Canada Research Chair in Distribution Management HEC Montreal

Title:
Midnight Stories

Abstract:
We will reexamine a number of known optimization problems in a new light.
Some theoretical and empirical findings will be reported.

Nonlinear Road Pricing

Siriphong Lawphongpanich

Department of Industrial and Systems Engineering

University of Florida, Gainesville, Florida, U.S.A.

Lawphong@ise.ufl.edu

Yafeng Yin

Department of Civil and Coastal Engineering

University of Florida

1 Introduction

Nonlinear pricing is prevalent in many industries and generally refers to a case in which the price or tariff is not strictly proportional to the quantity purchased. For example, railroad tariffs are based on the weight, volume, and distance of each shipment. However, discounts are often given to full-car and/or long-distance shipments. In utilities, the price or rate per kilowatt hour is different for different types of users. Heavy users of electricity particularly during peak hours generally pay a higher rate. Airlines routinely offer discount tickets for advance purchase, non-cancellation restriction, round-trip travel, and in competitive markets. In each of these examples, the average price paid per unit varies depend on characteristics of the purchase such as its total size, time of purchase or usage, type of markets, and restrictions.

Economists have been studying nonlinear pricing since Dupuit's discussion of its manifestations in railroad pricing in 1894 and Pigou's later categorization of the phenomenon in 1920. However, the same is not true for road pricing. De Borger [1] and Wang et al. [5] opine that nonlinear pricing has been largely overlooked in the road pricing literature. On the other hand, road pricing in practice is often nonlinear. Area-based or cordon pricing schemes in Singapore, Norway, London, and Stockholm are forms of nonlinear pricing. On Interstate 15 in Utah, solo drivers can pay $50 per month to use its Express Lanes. In addition, a number of road pricing schemes offer quantity discounts. London's Congestion Charge offers monthly and annual passes for frequent users at an approximately 15% discount over its daily access fee.

2. Overview

In this presentation, we consider nonlinear pricing in the context of managing travel demand, reducing congestion, and environmental impact in a given area. In the literature, an area is often assumed to consist of connected roads or roads in a connected geographical area. However, this is unnecessary in

theory and in practice. For example, an "area" can consist of roads, not necessarily connected, but under the jurisdiction of a single entity (a government agency or private company).

Specifically, we consider pricing schemes that may vary with the distance traveled inside a restricted area by the users. Mathematically, we let a function $T(\ell)$ represent the toll amount, where ℓ denotes the distance. When β, μ_1 and μ_2 are appropriately chosen, letting $T(\ell) = \min\{\mu_1 \ell, \beta + \mu_2 \ell\}$ or $\max\{\mu_1 \ell, \beta + \mu_2 \ell\}$ captures common nonlinear pricing functions in the economics and road pricing literature (see, e.g., [5] and [6]). Below, we also refer to the two nonlinear functions as $T^{\min}(\ell)$ and $T^{\max}(\ell)$, respectively, and illustrate them graphically in Figures 1 and 2 for various parameter values.
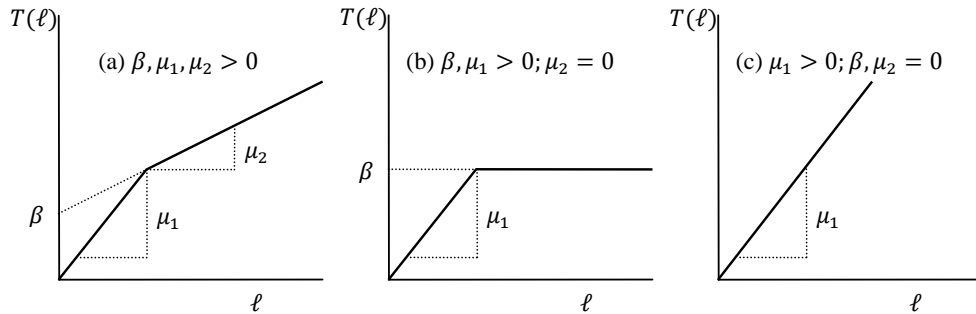
Figure 1: Various forms of $T^{\min}(\ell)$.
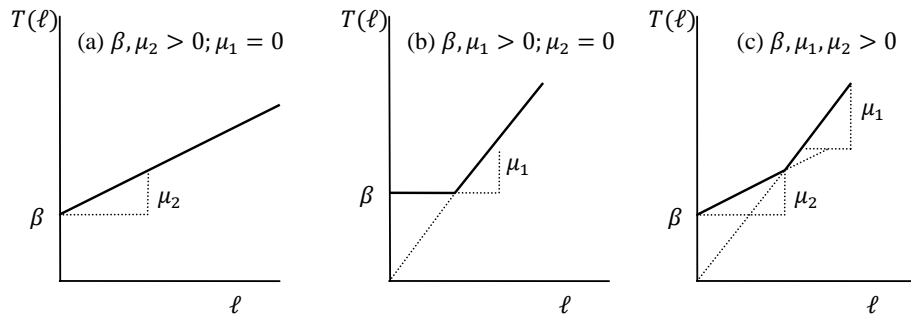
Figure 2: Various forms of $T^{\max}(\ell)$.

In Figure 1, case (a) represents a distance-based pricing structure that offers a discount to heavy road users because, as drawn, $\mu_1 > \mu_2$. (Although it may be a practical form of pricing in many industries, case (a) may not alleviate congestion.) Case (b) allows users to either pay an access, β, or a distance-based fee at rate μ_1. The later is more economical when the distance traveled inside the restricted area is sufficiently short. In case (c), $T^{\min}(\ell)$ becomes linear when β and μ_2 are both zero. For $T^{\max}(\ell)$, the pricing function for case (a) in Figure 2 consists of an access, β, and a distance-based fee with a rate μ_2. Economists refer to this case as a two-part tariff. Similar to (a), the function in case (b) consists of both an access and a distance-based fee. However, the latter only applies when the distance traveled exceeds a threshold. In economics, case (b) is called a three-part tariff. Instead of giving a discount to heavy users, case (c) in Figure 2 discourages heavy road usage by charging a

higher rate when the distance traveled exceeds a threshold. In addition to those shown in the two figures, $T^{\min}(\ell) = T^{\max}(\ell) = \beta$ when $\mu_1 = \mu_2 = 0$. In this case, both $T^{\min}(\ell)$ and $T^{\max}(\ell)$ become area-based pricing (see, e.g., [3]), a pricing scheme that only charges an access fee.

In transportation literature, many (e.g., [3]) state tolled user equilibrium conditions in term of paths. To illustrate, let Ω be the set of links in the road network. We represent an arc or link in Ω as a or a pair (i, j), where i and j denote two distinct nodes in the network. Associated with each arc, there is a travel time or link performance function, $s_a(\cdot)$, and $\boldsymbol{s}(\cdot) \in R_+^L$ is a vector of these functions, where L is the cardinality of Ω and the "+" sign in the subscript indicates that each component of the vector is nonnegative. (Herein, the bold typeface indicates vectors of variables or functions.) In addition, the length of arc (i, j) is denoted as ℓ_{ij}. For travel demands, K denotes the set of origin-destination (OD) pairs and d_k is the demand for OD pair $k \in K$. Associated with each OD pair, there is an inverse demand function $D_k^{-1}(\cdot)$ that determines the value of d_k. Additionally, $\boldsymbol{d} \in R_+^K$ and $\boldsymbol{D}^{-1}(\cdot) \in R_+^K$ are vectors of these demands and their inverse functions, respectively. To satisfy demands, P^k denotes the set of all possible paths for OD pair k. Then, V^f represents the set of all feasible flow-demand distribution, denoted as $(\boldsymbol{f}, \boldsymbol{d})$, where $\boldsymbol{f}$ is a vector in which each of its components, f_r^k, represents the amount of flow on route $r \in P^k$ and the other, $\boldsymbol{d}$, is as defined above. Then, one description of V^f is as follows:

$$V^f = \left\{ (\boldsymbol{f}, \boldsymbol{d}): \sum_{r \in P^k} f_r^k - d_k = 0, \forall k; f_r^k \geq 0, d_k \geq 0, \forall k, r \right\}.$$

In our setting, Ω is partitioned into two subsets, Ω^1 and Ω^2, where the former contains arcs inside the restricted area and the later consists of those outside. By definition, $\Omega^1 \cap \Omega^2 = \emptyset$ and $\Omega = \Omega^1 \cup \Omega^2$. As mentioned previously, the subnetwork induced by Ω^1, i.e. the restricted area, need not be connected. Similarly, P^k is divided into two subsets. One subset, TP^k, consists of paths containing arcs in Ω^1 and using these paths requires paying toll. In general, paths in TP^k may contain arcs in both Ω^1 and Ω^2 in order to connect origins to destinations. On the other hand, paths in the other subset, NP^k, contain no arc in Ω^1 and are thus toll-free. For a given set of β, μ_1, and μ_2, $(\boldsymbol{f}, \boldsymbol{d}) \in V^f$ is in tolled user equilibrium if the following hold:

$$T \left(\sum_{a \in \Omega_1} \delta_{ar} \ell_a \right) + \sum_{a \in \Omega} \delta_{ar} s_a(\boldsymbol{v}) \quad = D_k^{-1}(d_k) \quad \forall r \in TP_{++}^k, k \in K, \qquad (2.1)$$

$$T \left(\sum_{a \in \Omega_1} \delta_{ar} \ell_a \right) + \sum_{a \in \Omega} \delta_{ar} s_a(\boldsymbol{v}) \quad \geq D_k^{-1}(d_k) \quad \forall r \in TP_0^k, k \in K, \qquad (2.2)$$

$$\sum_{a \in \Omega_2} \delta_{ar} s_a(\boldsymbol{v}) \quad = D_k^{-1}(d_k) \quad \forall r \in NP_{++}^k, k \in K, \qquad (2.3)$$

$$\sum_{a \in \Omega_2} \delta_{ar} s_a(\boldsymbol{v}) \quad \geq D_k^{-1}(d_k) \quad \forall r \in NP_0^k, k \in K. \qquad (2.4)$$

In the above, δ_{ar} (equals 0 or 1) indicates whether arc a is on path r and $\boldsymbol{v} \in R_+^L$ is a vector of aggregate link flows, i.e., $\boldsymbol{v}$ has $v_a = \sum_{k \in K} \sum_{a \in \Omega} \delta_{ar} f_r^k$ as its components. In (2.1) and (2.2), $TP_{++}^k =$

$\{r \in TP^k : f_r^k > 0, r \in P^k\}$, i.e., TP_{++}^k is the set of utilized paths using the restricted area and $TP_0^K = \{r \in TP^k : f_r^k = 0, r \in P^k\}$ is the set of non-utilized paths. The sets NP_{++}^k and NP_0^k are similarly defined for paths not using the restricted area. In words, the above conditions essentially state that, at equilibrium, all utilized paths for every OD pair, using the restricted area or otherwise, must have that same generalized cost that equals to the value of the inverse demand function evaluated at the "realized" demand d_k. As pointed out in [3], the generalized cost expression in (2.1) and (2.2) is not linkwise additive. Moreover, $T(\cdot)$, the pricing function, is not necessarily differentiable. These observations lead many (e.g., [3]) to mistakenly conclude that there exists no link-based user equilibrium conditions when a nonlinear pricing function is present and to develop specialized algorithms to find equilibrium flow distributions.

Our goal in this presentation is to show that, for a significant number of parameter values (i.e., β, μ_1, and μ_2), there exist link-based user-equilibrium conditions equivalent to (2.1) – (2.4). In fact, these link-based conditions are the Karush-Kuhn-Tucker conditions of the standard traffic assignment problem with some flow-balance equations either fully or partially replicated. For those parameter values without equivalent link-based equilibrium conditions, we show how to modified standard algorithms such as the Frank-Wolfe algorithm and its variants (see, e.g., [2]) to find equilibrium flow distributions under nonlinear pricing. In most cases, the modified algorithms require solving two shortest path problems per iteration. Some of these problems may have side constraints. Methods for finding optimal parameter values are also discussed.

References

[1] B. De Borger, "Discrete Choice Models and Optimal Two-part Tariffs in the Presence of Externalities: Optimal Taxation of Cars", *Regional Science and Urban Economics* 31, 471–504 (2001).

[2] D.W. Hearn, S. Lawphongpanich, and J. Ventura, "Restricted Simplicial Decomposition: Computation and Extensions", *Matematical Programming Study* 31, 99–118 (1987).

[3] T. Maruyama and N. Harata, "Difference Between Area-based and Cordon-based Congestion Pricing: Investigation by Trip-chain-based Network Equilibrium Model with Non-Additive Path Costs", *Transportation Research Record* 1964, 1–8 (2006).

[4] J. Wang, R. Lindsey, and H. Yang, "Nonlinear Pricing on Private Roads with Congestion and Toll Collection Costs," Technical Report, Departement of Economics, University of Alberta, Edmonton, Alberta, Canada (2008).

[5] R. Wilson, *Nonlinear Pricing,* Oxford University Press, New York (1993).

A stochastic lane assignment scheme for macroscopic multi-lane traffic flow modeling

J.P. Lebacque

INRETS-GRETIA,

2 avenue du Général Malleret Joinville,

F 94114 Arcueil, FRANCE

M.M. Khoshyaran

ETC Economics Traffic Clinic,

35 avenue des Champs Elysées,

F 75008 Paris, FRANCE

Corresponding Author: J.P. Lebacque

Email: lebacque@inrets.fr

November 2009

1   Introduction

Macroscopic traffic flow models constitute a very useful tool for traffic management, simulation and planning at a network level. They are parcimonious in terms of computational needs and model identification, as they require very few parameters, and as boundary and initial conditions are easily estimated. For most applications it is sufficient to consider models which do not distinguish lane specific traffic dynamics.

The LWR model ([13], [16]) is the simplest of such models. It is based on the hypothesis that traffic flow can be described by three variables dependent on location x and time t: the density $\rho(x,t)$, speed $v(x,t)$ and flow $q(x,t)$, related by the following relations:

$$
\left|
\begin{array}{ll}
\frac{\partial \rho}{\partial t} + \frac{\partial q}{\partial x} = 0 & \text{conservation equation} \\
q = \rho v & \text{definition of } v \\
v = V_e(\rho, x) & \text{behavioral equation.}
\end{array}
\right.
\tag{1}
$$

or simpler as:

$$
\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x} Q_e(\rho, x) = 0 \quad,
\tag{2}
$$

with Q_e and V_e the equilibrium flow- resp. speed-density relationships ($Q_e(\rho, x) \stackrel{def}{=} \rho V_e(\rho, x)$).

This model admits a simple and efficient numerical solution, the Godunov scheme ([9], [7], [2], [8]), which can be extended to networks by including boundary conditions and node models ([1], [11], [12]). These node models do not accomodate multi-lane links at this point.

In some important situations, traffic dynamics depend strongly on lanes, for instance in urban networks (intersections with strong and interacting turning movements) or motorways with off-ramp constraints or significant conflicts related to on-ramps. The problem of extending macroscopic traffic flow models in order to accomodate multi-lane traffic has attracted the attention of the research community. Some models have been proposed, which describe only the overall impact of lane change on traffic flow ([5], [6]), without describing explicitly the dynamics of traffic on each lane. Indeed, lane interactions are complex ([18], [10], [15], [17]).

Other models describe the interaction between lanes as flow transfers between lanes ([14], [15]), or represent a multi-lane link as a succession of multi-link nodes ([11], based on a supply/demand approach, [4], [17]). One last approach, on which the present paper expands, considers that the essential mechanism between lane change is the formation of local equilibrium between lanes ([3], [11]): users will chose the most advantageous lane.

2 The proposed model

2.1 Lane assignment

Let us recall the lane assignment model in [11], which generalizes the model [3]. Let I be the set of lanes, $\mathcal{D}$ the set of user classes (for instance, $d \in \mathcal{D}$ will refer to the driver destination), I^d the set of lanes accessible to vehicles of class d, $\gamma_i \rho_{max}$ the maximum density of lanes i, ρ_i^d the density of vehicles d in lanes i, ρ^d the total density of vehicles d . Then the ρ_i^d are the unknowns of the lane-assignment problem and are subjet to the following constraints:

$$
\left|
\begin{array}{ll}
\rho_i^d \geq 0 & \forall d , \ \forall i \in I^d \\
\sum_{i \in I^d} \rho_i^d = \rho^d & \forall d \\
\sum_{d/i \in I^d} \rho_i^d \leq \gamma_i \rho_{max} & \forall i
\end{array}
\right.
\tag{3}
$$

The ρ^d constitute the dynamic data (analogue to OD data in classical deterministic assignment on networks problems) and the γ_i and I^d constitute the geometric data of the lane assignment problem. The unknowns ρ_i^d can be determined by solving

$$
Max \sum_{i \in I} \int_{i \in I}^{\sum_{d/i \in I^d} \rho_i^d} V_e(s) ds
\tag{4}
$$

(Wardrop optimum), with constraints (3). Optimizing (4) under constraints (3) results in the following lane assignment: all vehicles of class d share a common speed ν^d, which is equal to the speed

$$
v_i \overset{def}{=} V_e \left(\sum_{d/i \in I^d} \rho_i^d \right)
$$

of any of the lanes i used by users d (i.e. $\rho_i^d > 0$), and is greater to the speed v_ℓ of lanes ℓ which are not used by users d (i.e. $\rho_\ell^d = 0$).

This is a user equilibrium, and usually the total flow is not maximized.

The partial densities ρ^d satisfy a system of conservation equations:

$$\frac{\partial \rho^d}{\partial t} + \frac{\partial}{\partial x}\left(\rho^d \nu^d\right) = 0 \quad \forall d \in \mathcal{D} \tag{5}$$

Note that ν^d is a function of the partial densities $\rho \overset{def}{=} \left(\rho^\delta\right)_{\delta \in \mathcal{D}}$.

2.2 A stochastic lane assignment model

The analysis of the system of conservation laws (5) is difficult, mainly because the speeds ν^d are implicit. The only known solution concerns the two type of vehicles - two lanes case ([3]). Further the model allows for no variability in the behaviour of drivers or in their perception. In order to improve the model and make it easier to analyze, we propose the following stochastic assignment model. The model is obtained by replacing (4) with

$$Max \sum_{i \in I} \left(\int_{i \in I}^{\sum_{d/i \in I^d} \rho_i^d} V_e(s)ds + \lambda \sum_{\delta/i \in I^\delta} H\left(\rho^\delta\right) \right) \tag{6}$$

with $H(x) \overset{def}{=} x\left(ln(x) - 1\right)$. This model yields a Logit assignment of partial densities in lanes:

$$\left|\begin{array}{l} \rho^d = \sum_{i \in I^d} \rho_i^d \quad \forall d \in \mathcal{D} \\ \rho_i^d = \rho^d e^{-(v_i/\lambda)} / \sum_{\ell \in I} e^{-(v\ell/\lambda)} \quad \forall d \in \mathcal{D}, \quad \forall i \in I^d \\ v_i = V_{e,i}\left(\sum_{d/i \in I^d} \rho_i^d, x\right) \quad \forall i \in I \end{array}\right. \tag{7}$$

Other stochastic models could be proposed along similar principles. Note that the limit of model (7) as the inverses sensitivity parameter $\lambda \to 0$ is the model (4).

The paper will analyze the properties of the model (7), and discuss discretization schemes and examples.

References

[1] Buisson, C., Lebacque, J.P., Lesort, J.B.

1. "Macroscopic modelling of traffic flow and assignment in mixed networks". *Proc. of the Berlin ICCCBE Conf. 1367-1374. (ed. PAHL, P.J., WERNER, H.).* 1995.

2. "The STRADA model for dynamic assignment". *Proceedings of the 1996 ITS Conference.* 1996.

[2] Daganzo, C.F. : "The cell transmission model: a dynamic representation of highway traffic consistent with the hydrodynamic theory". *Transportation Research B, 28B*, 4, pp 269-287, 1994

[3] Daganzo C.F. "A continuum theory of traffic dynamics for motorways with special lanes". *Transportation Research B*, 31, pp 83-102, 1997.

[4] Daganzo C.F. "A behavioral theory of multi-lane traffic flow. merges and the onset of congestion". *Transportation Research Part B* 36, 159-169. 2002.

[5] Greenberg J.M., Klar A., Rascle M. "Congestion on Multilane highways". *SIAM Journal of Applied Mathematics* 63(3), 813-818. 2003

[6] Jin, W.L. "A kinematic wave theory of lane-changing vehicular traffic". *eprint arXiv:math/0503036*. 2006.

[7] Lebacque, J.P. "Les modèles macroscopiques de trafic". *Annales des Ponts.* **67**, 3rd trim., 28-45. 1993.

[8] Lebacque, J.P. "The Godunov scheme and what it means for first order traffic flow models. *Transportation and traffic flow theory", Proceedings of the 13th ISTTT* (J.B. Lesort Editor); Pergamon, 1996.

[9] Lebacque, J.P. "Semimacroscopic simulation of urban traffic". *Int. 84 Minneapolis Summer Conference. AMSE.* 1984.

[10] Laval, G., Daganzo, C.F. "Multi-lane hybrid traffic flow model: a theory on the impacts of lane-changing maneuvres". *TRB Annual Meeting* 2005.

[11] Lebacque, J.P., Khoshyaran, M.M. "Macroscopic flow models". *Presented at the 6th Meeting of the EURO Working Group on Transportation 1998. Published in "Transportation planning: the state of the art"* (Editors: M. Patriksson, M. Labbé), 119-139. Kluwer Academic Press. 2002.

[12] Lebacque J.P., Khoshyaran M.M. First order macroscopic traffic models: intersection modeling, network modeling. In: *Transportation and traffic theory, Flow, dynamics and human interaction.* H. Mahmassani ed. Elsevier 2005.

[13] Lighthill, M.H., Whitham, G.B. " On kinematic waves II: A theory of traffic flow on long crowded roads". *Proc. Royal Soc. (Lond.)* A 229: 317-345. 1955.

[14] Michalopoulos, P.G., Beskos, D.E., Yamauchi, Y. "Multilane traffic flow models: some macroscopic considerations". *Transportation B*, 18, 377-395, 1984.

[15] Ngoduy, D. Macroscopic discontinuity modelling for multiclass multilane traffic flow operations. TRAIL thesis series. Delft University Press 2006.

[16] Richards, P.I. Shock-waves on the highway. *Opns. Res.* 4: 42-51, 1956

[17] Schnetzler B, Lebacque JP, Haj-Salem H, Louis X. "Behavioral factors and lane changes near intersection" accepted for publication in Transportmetrica. 2009.

[18] van Winsum W., de Waard D., Brookhuis K.A. "Lane change manoeuvres and safety margins". *Transportation Research F* 2, 139-149. 1999.

Real-Time Traffic Estimation Using Data Expansion

Roger Lederman

Columbia Business School, Uris Hall, 3022 Broadway, New York, NY 10027, USA

Email: rlederman13@gsb.columbia.edu

Laura Wynter

IBM Research, PO Box 704, Yorktown Heights, NY 10598, USA

Email: lwynter@us.ibm.com

1 Introduction

Real-time traffic data is readily available in many cities around the world. Real-time data comes from numerous sources; some of these sources have been available for decades, such as inductive loops present at traffic signals, and others are more recently prevalent, such as GPS data from equipped vehicles, and digital video. Increasingly, traffic authorities are interested in leveraging these types of data for real-time traffic analytics. Real-time traffic analytics include such capabilities as route guidance and real-time information provision on the road condition for drivers, as well as tools for network operators to use in improving traffic flow. All of these new and emerging capabilities require an accurate estimation of current and near-term predicted traffic on the road network.

In order to address these important challenges, a first step is to assess the availability in real time of traffic data across the road network. In many cases, while the data is available in principle, it includes many gaps, both spatially and temporally. Gaps in the real-time data availability present a serious impediment to the effective use of certain applications. For instance, traffic-dependent route guidance requires estimates of the traffic across the links of the network. Missing data on parts of the network can lead to route suggestions that are highly sub-optimal for the current and future traffic levels. The same predicament arises for network managers who wish to optimize the flow of traffic in real time. If the incoming real-time data has significant gaps, or any gaps on critical links, operational decisions cannot necessarily be made with confidence.

In this paper, we present a method for estimating traffic volumes across a full-sized urban road network in real time. Urban networks are typically characterized by:

1. a large number of links and origin-destination pairs, requiring extreme scalability of any real-time applications.

2. limited data availability in comparison to the more controlled setting of freeway systems.

3. a multiplicity of viable routes between origin-destination pairs, so that driver routing decisions may vary substantially according to the current traffic conditions.

Networks of this type present a sizeable challenge for estimating link volumes in real-time. Our focus will be on these specific computational and data availability issue, rather than a more complete description of traffic conditions that would include other parameters such as speed, occupancy, and travel time. An estimation procedure, when embedded in a full infrastructure with data transmission and database latencies, must complete all calculations in a matter of seconds. Hence, the challenge in this work was to devise a method that can accurately reproduce missing link volumes with very little computational overhead in real time.

The data that can be assumed to be available is a set of historical link volumes that cover some but not all of the network as well as the same type of data as a real-time feed, stored in a database. An example of the cadence of the real-time feed is that a new set of average link volumes is available every five minutes. In the cases of interest, a significant portion of real-time five-minute-average link volumes are missing in the majority of time periods in which data is collected. Typically, in such cases, the missing link volumes cover a portion of the network which is simply not equipped with sensors, but there is also some spatial variation due to faulty network connections or erroneous data that has been filtered and removed from the real-time data feed. In other words, traffic volumes are available some of the time on some of the links, but seldom all of the time or on all of the links.

Consequently, statistical techniques for filling in very limited missing data from a feed of link volumes cannot be used; in the setting in which we are working, the gaps are too large and too persistent. Rather, data can be estimated via the methods more traditionally used in traffic planning, such as traffic assignment or simulation. The difference, however, is that we require those traditional traffic planning approaches to provide estimates in real-time on ultra-short time scales. When used for planning, these approaches do not readily incorporate the real-time traffic characteristics, but rather are based on a typical set of parameters. Typical parameters include origin-destination demands as well as link cost functions. These parameters tend to reflect well average-case conditions, but may not reflect as well the real-time attributes of the traffic at any point in time.

Our approach recognizes that critical parameters may vary from their average-case values, and that real-time data, although limited in our context, provides an important indication of the prevailing conditions. By combining the measured traffic volumes with a model of driver behavior

in equilibrium, we formulate estimates of the missing volumes that reflect the current conditions, rather than the long-term averages of traffic on these links. In particular, we hypothesize that current link volumes conform to a static traffic equilibrium, reflecting an unobserved set of origin-destination demands which may not be consistent with their long-term averages. In this way we model the dependency of current conditions on random events that may effect the usage of the traffic network. The equilibrium paradigm we adopt incorporates the routing adjustments made by urban drivers on a day-to-day basis in the face of such events.

While traffic flow is clearly a dynamic phenomenon, and huge strides have been made in the past decades in the field of dynamic traffic modeling, our work simplifies this aspect of traffic flow to permit its use on full city-wide urban networks in real time. Although advances have been made in both computation capacity and algorithm design, we are not aware of dynamic traffic assignment being used in practice for real-time traffic operations. As regards traffic simulation and its use in real-time operations, we believe that our method can provide more accurate results and provide them faster than simulation in a real-time setting. A significant reason for this is the heavy computational burden that simulation programs and dynamic traffic equilibrium models demand and the inability to leverage their full power in real-time operations.

The Data Expansion Algorithm (DEA) consists of two phases: a real-time estimation phase, and an offline calibration phase. The real-time phase is computationally lightweight and is parameterized by a set of link-to-link splitting probabilities which indicate the proportion of turns taken by drivers at each intersection. The offline phase uses the historical collection of real-time feeds to calibrate these parameters in accordance with the most likely traffic equilibria. The next section provides a review of relevant literature. We give an overview of the algorithm in Section 3. Section 4 concludes with a brief discussion of the merits of our approach as well as some extensions of our model to incorporate additional information that may be available for some road networks.

2   Review of related literature

The problem of expanding link flows is closely related to that of origin-destination (OD) matrix estimation. If OD demands can be estimated from a partial set of links flows, then the full set of link flows can be computed using a traffic assignment model. The static OD matrix estimation problem is often formulated as a bilevel optimization problem, where the lower level problem enforces equilibrium conditions on demand and link flow estimates, and the upper level minimizes some combination of a distance metric between estimated and observed link flows, and the distance form a target set of demands. Nguyen [9] was the first to formulate the problem to include equilibrium conditions. Fisk [5] put the problem into a bilevel optimization framework. The bilevel formulation is non-convex, and heuristic approaches to its solution remains an active area of research [11, 7].

As traditional static estimation approaches work with long-run average volumes, our motivation is closer to that of dynamic traffic simulation applications (e.g. DYNASMART and DynaMIT, see [12] for a survey) which combine historical and real-time information to estimate traffic flow. As in the static OD-estimation problem, these mesoscopic simulators seek to match estimated flows to observed flows, where estimated flows are constrained to satisfy equilibrium conditions. However, equilibrium flows are determined, not analytically, but with a micro-simulator that must be solved iteratively in conjunction with demand estimation [2]. Such a procedure is unsuitable for the large networks and the ultra-short time frames we consider.

In a spirit similar to ours, a few authors have proposed simplified models, typically involving a linear relationship between OD demands and link flows, that can be calibrated using historical data. Cascetta [3] proposed a least-squares OD-estimation problem, where drivers choose routes in fixed proportions. Fixed route proportions have also been assumed in various Kalman filtering approaches, beginning with Okutani [10]. Ashok et al [1] apply Kalman filtering to deviations from initial estimates that are computed offline. In each of these references, the approach is tested on freeway networks, where the number of routes is small, and proportions may be fairly constant. It is unclear how these methods would perform on a large urban network. We also note that while the deviations-based filter uses a historical estimate of demands as starting point for its real-time estimates, reaction to events on the current day is still a result only of the statistical tracking. Mahmassani and Zhou [8] make this criticism, and replace the commonly used autoregressive model with a polynomial trend model that is more responsive to current conditions, but continues to assume fixed route proportions so that deviations are tracked statistically.

Another linearized real-time model is proposed by Lam and Tam [6], and applied in an urban setting in Hong Kong for estimation of travel times. The authors use an offline simulation component to calibrate a variance-covariance matrix for link travel times, all measured at a particular instant. This enables a real-time updating step, based on the deviations of observed travel times from average travel times.

The problem of estimating turning proportions at intersections, as we do in our offline phase has been studied in the context of a single complex intersection by, for example, Cremer and Keller [4]. Here, the interpretation of turning proportions is closer to that of OD demands, since a single turning decision determines a driver's ultimate destination. Link-to link splitting probabilities of our type also figure prominently in some microsimulation models; e.g. [13], where are they in fact determine each driver's simulated route choice.

3   Data Expansion Algorithm Overview

The Data Expansion Algorithm is divided between a very lightweight real-time phase and a more compute-intensive offline calibration phase. The primary purpose of the offline phase is to determine the most likely historical link traffic volumes across the network, according to traffic equilibrium principles. To be useful as a reflection of the real-time traffic, the offline phase needs to be re-run often. This may be once per day or once per week. Time is handled discretely, so that each time period involves a run of the offline calibration algorithm. Then, in real-time, the current link volumes are obtained on those links that have no real-time data available. The connection between the offline and the real-time phases of our algorithm are what we call splitting probabilities. These are analogous to those parameters used in some microsimulation programs: the percent of vehicles at a node which enters each outgoing link from the node.

Figure 1 shows a table of the current link volume data, and a series of historical estimates taken from the same time period, for some portion of a network. Note that for most links there are temporal gaps in the series of observations, and each observation is missing data from some links.

Link ID	03-04	03-05	03-06	03-09	03-10	03-11	03-12
0111	37	-	45	-	-	71	47
0112	98	106	103	95	110	102	111
0113	12	-	-	9	-	-	7
0114	0	-	4	0	0	2	0
0115	-	84	-	56	-	-	-
0116	22	30	29	15	30	31	35
0117	5	20	-	35	7	-	-
0118	-	-	-	-	-	-	-
0119	-	178	200	154	-	205	220
0120	70	-	120	150	140	65	72
0121	-	-	-	-	-	-	-

Link ID	Current Sensor Output
0111	-
0112	102
0113	-
0114	0
0115	40
0116	30
0117	-
0118	-
0119	180
0120	-
0121	-

Figure 1: Sample of historical and current link volume data with spatial and temporal gaps for a given time of day.

Figure 2 diagrams the way in which the offline and real-time phases interact, and historical and real-time data is combined, to produce real-time estimates for the current period. To calibrate spitting parameters for the current time period, s, we use historical observations (l^1, l^2, etc.) taken from the same time period on prior days (in the figure, time is divided within days only, although segmenting can be by time and day of the week as well). By our modeling assumptions, the link volumes in each of these historical time periods, as well as in the current period, conform to a static equilibrium assignment of the demand for that day and period. This demand, which is unobserved, varies randomly from day to day, but daily demands for the same period follow the same probability distribution. Within a single day and period, demands, and consequently link flows are treated as stationary. Obviously, if links flows are entirely stationary across a time period, the current flow
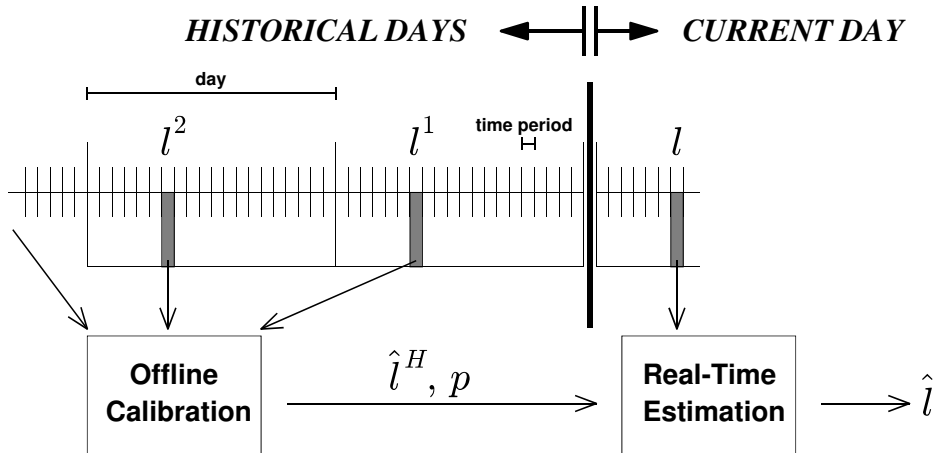
Figure 2: DEA framework for combining real-time and historical data, including the historical expanded (estimated) link flows, $\hat{l}^H$ and resulting splitting probabilities p.

can, in theory, be estimated only once for the entire time period. In practice, of course, volumes are estimated each time the real-time feed is updated to ensure that prior estimates continue to match the current observations. In any case, the duration for which the current volume estimates remains useful does not detract from the need to compute current estimates quickly when they are needed, which is our motivation.

For the real-time, estimation problem we propose a least-squares formulation with linear equality constraints. Its purpose is to find the most likely set of link volumes that satisfy the given splitting probabilities, based both on the offline volume estimates as well as on the real-time link volumes, where they are available. The real-time estimation problem can be solved efficiently even over large networks with very moderate computational complexity. It should be noted that while the offline calibration provides a set of splitting probabilities assumed to be valid over the given time interval, the filled-in link volumes that result from the real-time phase are not constant over the time interval. Indeed, the splitting probabilities are applied in the real-time phase every time a new data set arrives in to the system (for example, every five minutes). The real-time problem is solved using the new real-time data and the splitting probabilities as constraints.

The parameters of that estimation problem are determined through the offline calibration problem. Our approach to solving the offline calibration problem is to solve, repeatedly, static traffic assignments over the network, one for each time period, based on recent estimates of the origin-destination demand. In other words, the origin-destination demand for each time period must be re-estimated, via OD-matrix estimation, and then fed into a static traffic assignment routine. This procedure is run and re-run on a daily or weekly cadence, once for each pre-determined time period. We assume then that the cross-time-period dynamics can be neglected. The choice of this approach

to the offline calibration phase was made based on number of considerations including computational complexity and availability of data. Other approaches, for example simulation, could be used as well for the offline calibration algorithm. The offline calibration problem in our implementation, takes the form of a bi-level program. The formulation which we develop and present here can be calibrated using only historical averages of link volumes stored over the previous weeks. Hence, to summarize, the key considerations are that the offline problem can be solved periodically and that the real-time problem can be solved over a city network in a matter of seconds.

4 Conclusions

We presented an overview of our method for traffic estimation via an approach that we call the *data expansion algorithm*, or DEA. The goal of the method is to fill in missing values in real-time traffic volumes. This is important for enabling real-time traffic data to be used in many new and emerging traffic applications. Indeed, in practice, real-time data on the network flows is often missing, both spatially, with gaps on some links, as well as temporally, with gaps at some points in time. Such gaps in the real-time traffic data render difficult the use of analytic tools such as those for real-time route guidance and network control. We sought a method that would meet those objectives while being computationally lightweight enough to run in real-time. Realizing that much of the computational overhead comes from reading and writing to database, the method itself needs to run in a matter of seconds on a city-wide traffic network.

The method we developed works in two phases. The offline phase involves the resolution of a set of bilevel programs for a number of pre-determined time periods. The online phase of our method is designed to be fast and scalable so that it can be run in real-time and makes use of the parameters computed in the offline phase along with real-time data on traffic flows. Our method was tested on two networks from Germany and shows excellent results, both in terms of accuracy as well as in terms of coverage of the missing values on the network.

Within the two-phase framework we present, there are a number of variations to the specific approach we have implemented. Depending on the details of the problem setting, it may be possible to achieve a closer approximation to path-based estimation. Another interesting extension of this work involves the use of different assignment algorithms, some of which may offer more spreading of flow across paths, a characteristic that would be of use for this particular application. Finally, a desirable extension of the data expansion algorithm is to develop a real-time module to better capture non-recurrent congestion and the effect of incidents.

References

[1] Kalidas Ashok. *Estimation and Prediction of Time-Dependent Origin-Destination Flows*. PhD thesis, Massachusetts Institute of Technology, 1996.

[2] Moshe Ben-Akiva, Michel Bierlaire, Haris Koutsopoulos, and Rabi Mishalani. Dynamit: a simulation-based system for traffic prediction, 1998.

[3] Ennio Cascetta. Estimation of trip matrices from traffic counts and survey data: A generalized least squares estimator. *Transportation Research Part B: Methodological*, 18(4-5):289–299, 1984.

[4] M. Cremer and H. Keller. A new class of dynamic methods for identification of origin-destination flows. *Transportation Research Part B: Methodological*, 21(2):117–132, 1987.

[5] C.S. Fisk. On combining maximum entropy trip estimation with user optimal assignment. *Transportation Research Part B: Methodological*, 22(1):69–73, 1988.

[6] William H.K. Lam and Mei Lam Tam. real-time travel time estimation using automatic vehicle identification data in hong kong. *Lecture Notes in Computer Science*, 4413:352–361, 2007.

[7] Jan T. Lundgren and Anders Peterson. A heuristic for the bilevel origin-destination-matrix estimation problem. *Transportation Research Part B: Methodological*, 42(4):339–354, 2008.

[8] Hani Mahmassani and Xuesong Zhou. A structural state space model for real-time traffic origin-destination demand estimation and prediction in a day-to-day learning framework. *Transportation Research Part B: Methodological*, 41(8):823–840, 2007.

[9] S. Nguyen. Estimating an o/d matrix from network data. University of Montreal, Publication No. 60, 1977.

[10] Iwao Okutani and Yorgos Stephanedes. Dynamic prediction of traffic volume through kalman filtering theory. *Transportation Research Part B: Methodological*, 18(1):1–11, 1984.

[11] Michael Patriksson. On the applicability and solution of bilevel optimization models in transportation science: A study on the existence, stability and computation of optimal solutions to stochastic mathematical programs with equilibrium constraints. *Transportation Research Part B: Methodological*, 42(10):843–860, 2008.

[12] S. Peeta and A. Ziliaskopolous. Foundations of dynamic traffic assignment: The past, the present and the future. *Networks and Spatial Economics*, 1(3):233–266, 2001.

[13] Haifeng Xiao, Ravi Ambadipudi, John Hourdakis, and Panos Michalopoulos. Methodology for selecting microscopic simulators: comparative evaluation of aimsun and vissim. *ITS Institute, Center for Transportation Studies, University of Minnesota*, (CTS-05-05), 2005.

Estimation and Prediction of Traffic Parameters using Spatio-Temporal Data Mining

Axel Leonhardt (corresponding author) and Fritz Busch

Chair of Traffic Engineering and Control

Technische Universität München, Arcisstraße 21, 80333 Munich, Germany

+49-89-289-28587, axel.leonhardt@vt.bv.tum.de

1 Problem Statement and Motivation

High traffic demand and the resulting traffic congestion lead to an increased demand for dynamic traffic management systems (DTMS). Many DTMS rely heavily on a proper knowledge of the actual und near future traffic state, expressed through appropriate traffic parameters, e.g. travel time. Unfortunately, today's typical data collection setup does not provide direct network-wide measurements of these relevant traffic parameters – and naturally no prediction. This leads to the need for estimation and prediction models that operate on the available scattered traffic data. One class of such models is regression or data mining models. Regression models have two basic requirements:

1. A training database with observations of dependent variables y (to be estimated) and patterns of independent variables $X=x_1, ..., x_n$ that can serve as a predictor for y

2. Independent variables X that are continuously available, i.e. they can be accessed at any point in time

In medium or large scale urban agglomerations and corridors there are typically local detector data available. They – though limited in their explanatory power for certain DTMS applications like network wide traffic information – can serve as independent variables x, as they are continuously available and contain information about the actual traffic situation. On the other hand, there may be travel times collected from time to time by other systems (e.g. position data from taxi fleet vehicles, temporarily installed license plate readers or by communicating vehicles in the future) that are well suited to be used as dependent variables y.

The goal of a regression model is to find a suitable function that solves $y' = f(X)$ given a set of observations (y, X). Various methods can be applied, such as analytical regression models, artificial neural networks and adaptive expert systems. In this paper, we propose an instance based learning (IL) model (also known as "non-parametric Regression" or "k-Nearest-Neighbour"); a model class that has been used for estimation and prediction purposes in traffic engineering for almost two decades (e.g. [1], [2] and [3]). The proposed method here is novel in that it uses spatio-temporally weighted traffic data from the network together with calendar variables (day of week and time of day) in a combined approach.

2 Methodology

In order to make a prediction through IL, the database is searched for similar patterns of measures and these patterns are used to fit a local function. What distinguishes IL from other methods is that there is no global function needed to fit all data observations. Instead the data is approximated locally; i.e. only relevant data close to the actual observation are used. Figure 1 demonstrates the working principle of the local approximation (right) in comparison to the global function approximation (left).
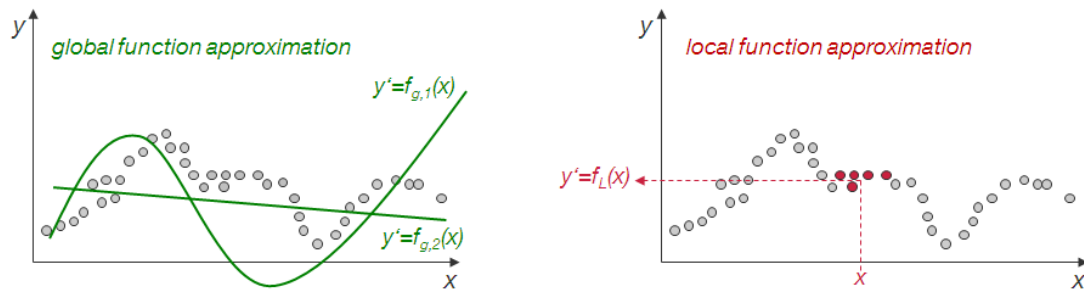
Figure 1: approximation by global (left) and local (right) function

Advantageous properties of IL models are that they are quite intuitively understandable (no "Black-Box"), are adaptive by nature (new observations are simply added to the existing database) and that highly complex functions do not need to be approximated globally. A drawback is that IL models need a lot of computational power as each function approximation is performed during runtime.

In the following, we describe how IL is applied to travel time estimation and prediction in networks using travel time observations as dependent variables y and local occupancy as independent variables X. X is extended by calendar variables in order to consider the time dependency of traffic demand. Figure 2 shows an overview of the prediction process:
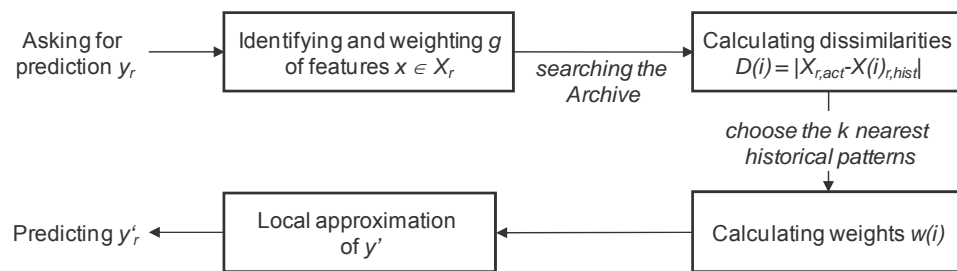
Figure 2: Prediction process

If a prediction $y'_r(t+\Delta t)$ is queried (e.g. travel time on route r for a prediction horizon Δt), for all observations y_r in the database, the relevant features are identified to form the pattern X_r. X_r consists of measurements that are spatially and temporally close to y_r, together with the respective day of week and time of day.

The dissimilarity D of two situations is calculated as

$$D = G_V \times D_V + D_C + D_M,$$

where D_V is the dissimilarity of the traffic patterns, G_V is a weighting factor, D_C is the dissimilarity of the calendar features and D_M is a penalty for missing data, i.e. incomplete traffic patterns.

The dissimilarity of two traffic situations D_V is calculated as

$$D_{V,r,i} = \frac{\left[\sum_j \left(g_j(ds,dt) \times \left| x_{i,j}^{akt}(ds,dt) - x_{i,j}^{hist}(ds,dt) \right|^{k_e} \right) \right]^{\frac{1}{k_e}}}{\sum_j \left(g_j(ds,dt) \right) \times \bar{x}_i^{akt}}$$

with $0 \le g_j \le 1$ being a factor to weight the local traffic data points j in pattern i $x_{i,j}$ according to their relevance for y_r. g_j could for instance be derived by correlation analysis or assignment that quantifies the relevance of x_j for y_r. In this work, g_j is calculated based on the spatial proximity ds (defined as the downstream distance from the detector station to the route) and the age dt of the data point. This assumes that information in urban networks travels mainly in flow direction and that the relevance of a measurement decreases with increasing age. k_e is the weighting penalty for single large differences in the pattern.

The weight g_j is calculated as

$$g_j(ds,dt) = max\left\{ 0; \ 1 - \frac{ds}{ds_{max}} - \frac{dt}{dt_{max}} \right\}$$

with ds_{max} and dt_{max} determining the rate at which the weight decreases with increasing distance and age of the information, respectively.

The calendar dissimilarity $D_{C,i}$ of two patterns is calculated as

$$D_{C,i}\big((D_i, T_i), (D_{act}, T_{act})\big) = \big(E_y(D_i, T_i) - E_y(D_{act}, T_{act})\big)/\bar{y}$$

with E_y being the expected values for the dependent variable y of the actual instant and respective instant belonging to the historical pattern I, respectively, divided by the mean value of all historical observations y_i. Through the inclusion of D_C we reach a preference of a traffic pattern that was observed at a similar time of day and day of week as the actual pattern. The penalty D_M is calculated as the difference between the number of potential data points in the pattern n_{all} and the actual available features n_{valid}, divided by n_{all}. D_M is reduced with increasing prediction horizon t_{pred}, based on the assumption that the traffic patterns' relevance decreases with increasing prediction horizon.

$$D_{M,i} = \frac{n_{all,i} - n_{valid,i}}{n_{all,i}} \times \left[1 - \frac{t_{pred}}{t_{pred,max}} \right] \text{ with } 0 \le t_{pred} \le t_{pred,max}$$

Now, each observation y_i is associated with a weight w_i, which is calculated using a Gaussian kernel function,

$$w_i = e^{\frac{D_i/D_{i,min}}{K_w}}$$

with K_W being the kernel width, which determines the smoothing of the estimate and $D_{i,min}$ being the minimum dissimilarity of all k selected observations. Finally the estimate y' is calculated as

$$y' = \frac{\sum_{i=1}^{k} w_i \times y_i^{hist}}{k}$$

3 Calibration, Application and Results

The proposed IL model has six parameters (smoothing, spatio-temporal weighting, weight of traffic pattern in relation to calendar variables, penalty for single large differences in the pattern). The parameters are optimized using a Genetic Algorithm applied to a training data set of two weeks with the objective to minimize prediction errors.

The method has been tested for several routes in the City of Graz, Austria (using local occupancy data X and taxis as probe vehicles to collect travel times y) and in the city of Munich, Germany (using vehicle re-identification to collect travel times y and local traffic volume data X). Figure 3 shows some sample plots of estimated and reference travel times. Congestion induced delays can be estimated and predicted if the archive contains relevant traffic patterns. Figure 4 shows a comparison of the results obtained with the proposed instance based learning method (IL), estimations obtained using a multilayer feed forward neural network (NN) and a purely calendar-based approach (using typically expected travel times).
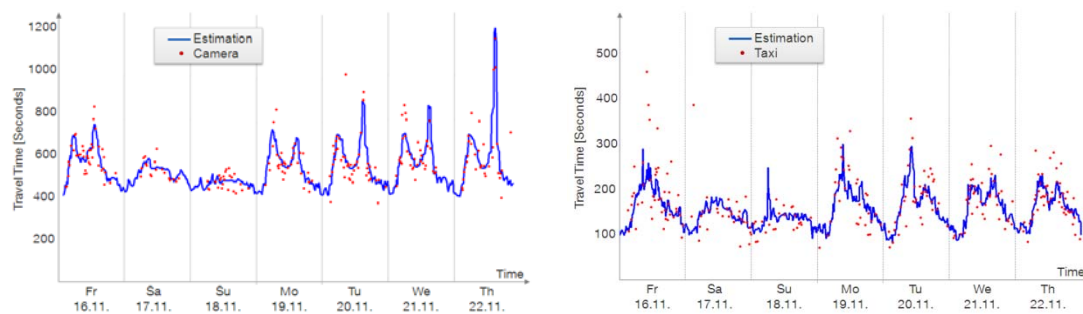
Figure 3: Est. travel time, training data from vehicle re-identification (left) and taxi probes (right)
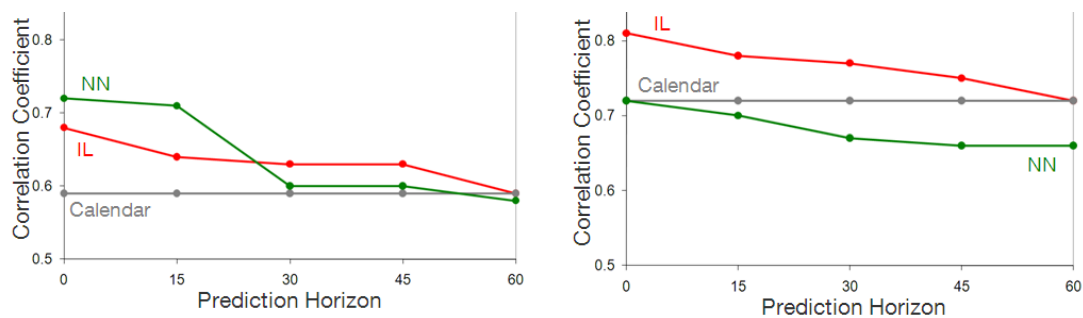
Figure 4: Comparison of different prediction methods (instance based, calendar, neural network)

References

[1] S. Robinson and J.W. Polak, "Modelling Urban Link Travel Time with Inductive Loop Detector data using the k-NN method", 84th TRB Annual Meeting, Washington, D.C. (2005)

[2] G. A. Davis and N. L. Nihan, "Nonparametric regression and short-term freeway traffic forecasting", Journal of Transportation Engineering, 117(2), 178-188 (1991).

[3] R. E. Turochy and B. D. Pierce, "Relating Short-Term Traffic Forecasting to Current System State Using Nonparametric Regression", 2004 IEEE ITS Conference Washington, D.C., USA, Oct. 34, 2004.

A Third Order Highway Multilane Model

Louis X.

GRETIA-INRETS

Paris Est Le Descarte 2 rue de la Butte Verte 93166 Noisy le Grand

Email: louis@inrets.fr

Schnetzler B. & JP Lebacque

GRETIA-INRETS

Paris Est Le Descarte 2 rue de la Butte Verte 93166 Noisy le Grand

1   Introduction

The macroscopic approach in modelling highway traffic is usually based on a monodimensionnal assumption whatever is the number of lanes of the highway link under consideration. These models are all assuming that the value of relevant quantities (such as density, speed, flows) are the same for each lane at a longitudinal point of the network, as well for first order model [1] than for second order models [2] [3] [4] [5]. This is obviously wrong in principle, but as been neglected according to the troubles allready meet when dealing with this weak assumption [6]. In this paper, we will set a model for taking into account parts of the effect caused by lane changing, leading us to a third order model that follows the independant dynamics of density velocity and desired speed $W(t, x) = (\rho, v, v_d)(t, x)$ for each lane of the highway link. The model obtained will turn out to be hyperbolic and the tools to solve it as well analytically than numerically are the wellknown introduced by P Lax[10] and S Godunov[11] in the sixties. To deal with multilane links, one can considere microscopic approach, for instance in [16], another interesting way has been using initial kinetic approach [14] 'macroscopized' through moments method.

The main idea is to use a concept of internal state node ISN (with real and non zero length) adapted to the connexion of different links described by second order macroscopic models [14]. We decompose any multilane link in a sequence of parallel nodes ISN (see Fig 1.), the undelying system of conservative equations of each node is choosed in this very first approach to be based on a ARZ type model even if better second order GSOM model as those newly established in [15] would be better, this adaptation-improvment being quite straitforward to write and code. So the first step is to set the evolution in each node started from the hyperbolic system modeling a link's

case :

$$\begin{cases} \partial_t \rho + \partial_x \rho v = 0, \\ \partial_t \rho I + \partial_x \rho v I = 0. \end{cases} \tag{1}$$

with ρ the density, v the velocity, $I = v - V_e(\rho)$, a lagrangian caracteristic assumed permanetly or temporiraly constant (this second case involves a negative source term with a caracteristic time see [5]) along a trajectory $\frac{dI(t, X(t))}{dt} = 0$, which yields the second equation of the system (1) since $\frac{dX}{dt} = v$ combined with the first. $V_e(\rho)$ is a fundamental diagram, strongly concave in the fluid domain and weakly - almost linear - in the congested one, of which values $(Q_c, \rho_x, \rho_c, V_x)$ are set by calibration considering mesured data of each link.

The solution of the Riemann problem relative to this system is fully described in [8] and [9] for the supply-demand [7] formulation, which is a technic (equivalent to the classical one of wave positionning) of practical interest when dealing with node, or change of fundamental diagram. To summarize what is to be known solving (1), this system is hyperbolic with one genuinely non linear field $\lambda_1 = v + \rho V_e'(\rho)$ and a linearly degenerated one $\lambda_2 = v$, leading to the determination of a single constant intermediate state $W_1 = (V_e^{-1}(v_r - I_l), v_r)$ where r and l stand for right and left initial states, since this system is of Temple's type [5] with common parametrization for the crossing of the first wave (as well for shock than for rarefaction wave) : $[I] = 0$, while the LD second wave gives $[v] = 0$. In the case of a intersection, the node has a matrix of turns γ_{ij} ratio of the amount incoming in i entry and outgoing in j exit. The consequences of this is that for any node with any numbers of entries N_{in} and exits N_{out}, we introduce the density of those exiting by j, and have the second step which leads the evolution of the state W^n in the node:

$$\begin{cases} \partial_t \int \rho_j^n dx = \sum_{i=1}^{N_{in}} \gamma_{ij} q_{in,i} - \sum_{j=1}^{N_{out}} q_{out,j}, \\ \partial_t \int (\rho I)^n dx = \sum_{i=1}^{N_{in}} q_{in,i} I_{in,i} - \sum_{i=1}^{N_{out}} q_{out,j} I^n. \end{cases} \tag{2}$$

with $\rho^n = \sum_{j=1}^{N_{out}} \rho_j^n$, with $dx = $ Segment Lenght.

The values for the flows q are obtained for instance from supply/demand (Ω/Δ) approach which considers partial supplies and demands, according to each in (l for 'left') and out (r for 'right') flows:

$$\begin{cases} \Omega_i^n(\rho, I) = \beta_i \Omega_i(\rho, I_{JZ}), \\ \Delta_j^n(\rho, I) = \frac{\rho_j^n}{\rho^n} \Delta, \\ q_{in,i} = min(\Omega_i^n, \Delta_i^l), \\ q_{out,j} = min(\Delta_j^n, \Omega_j^r), \end{cases} \tag{3}$$

The β_i and I_{JZ} are computed accorded to Jin and Zhang rule [12] (offer proportional to demand

extended here with internal intention I_{JZ} composed with upstream ones in the same proportionality). Now, lanes exchanges are modelled if we considered that a link is a sequence of block of node, each one standing for a lane. To introduce the will for lane change, three things need to be considered. First, the Origin Destination (OD) which (the same approach could be used with as many partial densities as possible destinations, leading to a fully OD model) could be summarized here by a global matrix of turns Γ_{ij} that would impose or force the local γ_{ij} near each real intersection (not the one used to mimic lanes), for instance in the segment upstream the intersection, or in many more upstream segments (say two or three according to reasonnable anticipation). Second, the will to maintain a desired velocity, and then this introduces the need for a third independant variable, the so called "'desired speed"', which is supposed to be constant along trajectories leading us to set the very similar system

$$\begin{cases} \partial_t \rho + \partial_x \rho v = 0, \\ \partial_t \rho I + \partial_x \rho v I = 0, \\ \partial_t \rho v_d + \partial_x \rho v v_d = 0. \end{cases} \tag{4}$$

This purely advected variable which does not pertub the solution presented above (adding $[v_d] = 0$ through the first wave and a second eigenvector for the second wave) may seem to be useless, as long as one does not notice that this allows the integration of the multilane behaviors through the use of dynamical γ_{ij}, for instance setting $c_{kk'} = |vd_k - v_{k'}|$ and $p_{kk'} = e^{\alpha c_{kk'} - \epsilon \delta_{kk'}}$ and at last $\gamma_{kk'}(\epsilon, \alpha) = \frac{p_{kk'}}{\sum_{k'} p_{kk'}}$, the two coefficients beeing to be set through calibration procedure, like the one of the fundamental diagram. And third, the partial demands for lane, when the fluxus are coming from a single lane onramp (from recent considerations based on observations).
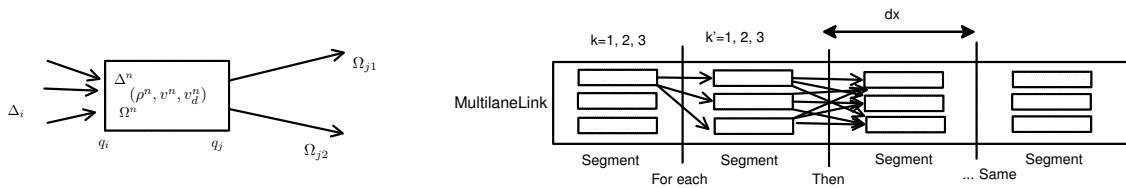
Fig 1.

This model has being programmed in the MFC-C++ based platform Ouranos, and is in phase of experimentation.

References

[1] LIGHTHILL M.H, WHITHAM G.B. On kinematic waves II. "A theory of traffic flows on long crowed roads". Proc. Royal Soc A229: 317-345 (1955).

[2] PAYNE H.J. "Model of freeway traffic and control", Simulation concils, Proc ser Math. Models Public Syst, 28 (1) 51-61 (1973).

[3] ZHANGH H.M. "An non equilibrium traffic model devoid of gas like behavior". Transportation research PartB, 275-290 (2002).

[4] AW A, RASCLE J.M. "Resurection of second order models of traffic flows", SIAM, Journal of mathematics 60 (3) 916-938 (2000).

[5] BAGNERINI P, COLOMBO RM, CORLI A. "On the role of source terms in continuum traffic flow models", Math Comp Modelling, 44, p917-930 (2006).

[6] DAGANZO C.F. "Requiem for second order fluid approximations of traffic flow" Transportation Research B, B29B, p277-286 (1995).

[7] LEBACQUE J.P. "The Godunov scheme and what it means for first order traffic flows models", ISTTT (1996).

[8] LEBACQUE J.P, LOUIS X, MAMMAR S, SCHNETZLER B, HAJ-SALEM H. "Modélisation du trafic autoroutier au second ordre", CRAS Ser1 346, p1203-1206 (2008).

[9] MAMMAR S. "Développement d'un modèle de simulation macro microscopique de simulation du trafic", Thèse de Doctorat, Université Evry (2006).

[10] LAX, P." Hyperbolic systems of conservation laws and the mathematical theory of shock waves" Conf Board Math Sci'11, SIAM (1973).

[11] GODUNOV, S. " An interesting class of quasilinear systems" Dok Akad Nauk, SSSR, 139, p521-523 Eng Tranls in Soviet Math 2, 947-949 (1961).

[12] JIN, W.L. ZHANGH H.M. " On the distribution schemes for determining flows through a merge" Transportation Researsh Part B 40, 251-264 (2006).

[13] LEBACQUE J.P, LOUIS X, MAMMAR S, SCHNETZLER B, HAJ-SALEM H. "A GSOM intersection model" presented at IFAC SEOUL (2008)

[14] NGODUY, D. "Macroscopic discontinuity modeling for multiclass multiflows traffic operations" PhD Thesis, Delft (2006).

[15] SCHNETZLER, B. LOUIS, X. "Anisotropic second order models and associated fundamental diagrams" submited to Tristan, TromsO (2009-10).

[16] YOUSIF, S. HUNT, J. "Modelling lane utilisation on British dual-carriageway roads: effects on lane-changing" Traffic Eng. Control 36 (12), p680-687 (1995).

A greedy construction heuristic for the Liner Service Network Design Problem

Berit Løfstedt

DTU Management Engineering

Technical University of Denmark

Email: blof@man.dtu.dk

3rd of November 2009

The Liner Service Network Design Problem (LSN-DP) is the problem of constructing a set of routes for a heterogeneous vessel fleet of a global liner shipping operator. Routes in the liner shipping context are non-simple, cyclic routes constructed for a specific vessel type. The problem is challenging due to the size of a global liner shipping operation and due to the hub-and-spoke network design, where a high percentage of the total cargo is transshipped. We present the first construction heuristic for large scale instances of the LSN-DP. The heuristic is able to find a solution for a real life case with 234 unique ports and 14000 demands in 33 seconds.

Literature overview: A MIP model of the LSN-DP consist of a highly unconstrained routing problem subject to a large degree of symmetry and a multicommodity flow problem dominating the constraint set and accountable for a large fraction of the cost. Previous work on liner service network design may be found in [1],[2], [3] and [4]. The models are distinct with regards to transshipments and the vessel fleet. In the early paper of [1] transshipments are not supported, whereas they are supported in [2], [3] and [4]. Transshipment costs are accounted for in the objective function of [2],[4] as opposed to [3]. The characteristics of the fleet differ as to whether it is heterogeneous for every vessel [1],[2] or consist of a heterogeneous fleet of homogeneous vessel classes [3],[4]. Non-linear capacity constraints are found in [1],[2] assuming that a vessel may complete its route an integral number of times during the planning horizon. Integrality is not imposed by [4]. The weekly frequency constraint is introduced by [3] assuming a number of homogeneous vessels assigned to each service to offer a weekly visit to each port en route with the capacity of the vessel class in question. Optimal results for smaller instances are presented by [2] and [4]. The approaches of [1],[2] and [3] have focused on solving the liner service network design problem with traditional decomposition and integer programming methods and fail to produce

results for realistic network sizes of a global liner shipping operator anno 2009. This is addressed by [4] benchmarking a tabu search approach presenting results within 3-5% of the optimal solution for up to 7 ports. The multicommodity flow problem is solved in each iteration, which is reported by [4] to become computationally expensive already for the 7 port instance. A case study of 120 ports in [4] show that a heuristic approach may scale to large instances but no execution time is reported and the quality of the solution is hard to evaluate. It is reported to visit important ports infrequently. A global network connects several hundred ports worldwide and the corresponding forecasted cargo demand comprises a commodity set of 4 orders of magnitude. Methods based on relaxation of the proposed models or simply evaluating the objective function during a search is not computationally efficient for large scale problems.

Heuristic approach: A solution to the LSN-DP is a set of routes covering the ports serviced by the shipping operator and transporting the forecasted cargo demand. Viewed as a graph partitioning problem the solution is a set of strongly connected components with a high degree of interconnection. The construction heuristic is based on the Multiple Quadratic Knapsack Problem (MQKP) and relies on a graph of the current schedule, which is divided into a set of dense subgraphs related by demand, expected transshipment flow and geographical proximity. A solution found by the construction heuristic is expected to be feasible and realistic, but the quality of the solution cannot be guaranteed as the heuristic cannot account for the flow problem and the transshipment cost. In MQKP a set of mutually exclusive items $i \in V$ are placed in $\mathcal{R}$ knapsacks with different weight bounds C_r. The objective is to maximise the profit of the knapsacks defined by the profit matrix P.

$$maximize(MQKP) = \sum_{r \in \mathcal{R}} \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} p_{ij} x_i^r x_j^r + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{V}} p_j x_j^r \tag{1}$$

subject to:

$$\sum_{i \in \mathcal{V}} w_i x_i^r \leq C_r \quad \forall r \in \mathcal{R} \tag{2}$$

$$\sum_{r \in \mathcal{R}} x_i^r \leq 1 \quad \forall i \in \mathcal{V} \tag{3}$$

$$x_i^r \in \{0,1\} \quad \forall i \in \mathcal{V} \tag{4}$$

The variables x_i^r indicate whether item i is included in the r'th knapsack. The knapsack constraint (2) makes the total item weight obey the bound C_r and constraints (3) ensure that items are mutually exclusive to the knapsacks. When the MQKP is applied to the LSN-DP the knapsack items $i \in V$ are the accumulated port visits of each port $t \in T$ and the knapsacks $r \in R$ represent services, which are a specific vessel class visiting a sequence of ports. Let A be the set of vessel classes and let N_a denote the number of available vessels of class $a \in A$ in the fleet. Let C_a denote the capacity in TEU of a single vessel of class $a \in A$. The number of services/knapsacks is dependent on the expected rotation time of a service $\sigma(C_a)$. $\sigma(C_a)$ depends on vessel capacity

as large vessels are typically assigned to cross regional services and small vessels are assigned to regional services. The number of knapsacks for the LSN-DP is hence $|R| = \sum_{a \in A} |R_a| = \sum_{a \in A} \lceil \frac{N_a}{\sigma(C_a)} \rceil$. The profit matrix $\mathcal{P}$ defines each entry $p_{ij} = f(l_{ij}, d_{ij}, h_{ij})$ where l_{ij} is the sailing distance in nautical miles and d_{ij} is the demand between ports $i, j \in V$. h_{ij} is the potential hub flow between port $i \in V$ and a hub port $j \in H \subset V$, where $H \subset V$ are ports with a small percentage of demand compared to the terminal capacity. A port $t \in T$ may be visited multiple times m_t by multiple services according to the capacity and schedule requirements of a port. Let M be a vector of size $|T|$ containing the number of weekly visits to each port $t \in T$. In the MQKP port $t \in T$ is duplicated m_t times for the knapsack items $i \in V$, $V = \{T \times M\}$ to represent the current schedule of ports. It is important to observe the weekly frequency constraint of the original problem in order to obtain a feasible solution to the LSN-DP using the construction heuristic. To ensure feasibility, each knapsack $r \in R$ is required to provide a Hamiltonian cycle of the items in knapsack r. The length of the cycle cannot exceed the mileage coverable by the vessels assigned to knapsack r. Edge variables y_{ij}^r and enumeration variables u_i^r are introduced in the MQKP to order the ports in each knapsack into a simple, cyclic route constrained by $\sigma(C_a)$. t_{ij}^a express the sailing time between ports i and j and t_i^a is the expected time spent in port i for vessel type a.

$$maximize(MQKP) = \sum_{r \in \mathcal{R}} \sum_{i \in \mathcal{V}} \sum_{j \in \mathcal{V}} p_{ij} x_i^r x_j^r + \sum_{r \in \mathcal{R}} \sum_{j \in \mathcal{V}} p_j x_j^r \tag{5}$$

$$\text{subject to:} \quad x_i^r x_j^r \geq y_{ij}^r \qquad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R} \tag{6}$$

$$\sum_{j \in V} y_{ij}^r - \sum_{j \in V} y_{ji}^r = 0 \qquad \forall i \in \mathcal{V}, r \in \mathcal{R} \tag{7}$$

$$\sum_{j \in V} y_{ij}^r \leq 1 \qquad \forall i \in \mathcal{V}, r \in \mathcal{R} \tag{8}$$

$$u_i^r - u_j^r + y_{ij}^r \sum_{i \in V} x_i^r \leq \sum_{i \in V} x_i^r - 1 \qquad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R} \tag{9}$$

$$\sum_{i \in V} \sum_{j \in V} y_{ij}^r (t_{ij} + t_i) \leq \sigma(C_a) \qquad \forall r \in \mathcal{R}_a, a \in \mathcal{A} \tag{10}$$

$$\sum_{r \in \mathcal{R}} x_i^r = 1 \qquad \forall i \in \mathcal{V} \tag{11}$$

$$x_i^r \in \{0, 1\} \qquad \forall i \in \mathcal{V}, r \in \mathcal{R} \tag{12}$$

$$y_{ij}^r \in \{0, 1\} \qquad \forall i \in \mathcal{V}, j \in \mathcal{V}, r \in \mathcal{R} \tag{13}$$

$$u_i^r \in \mathcal{Z}^+ \qquad \forall i \in \mathcal{V}, r \in \mathcal{R} \tag{14}$$

Constraints (6) ensure that an edge variable can only be activated if both endpoints of the arc are included in the knapsack. Constraints (7) ensure a cyclic route. Constraints (8) ensure that the cyclic route is simple and constraints (9) that the route is connected. Constraints (10) are the weekly frequency constraint ensuring that the simple, cyclic route has a voyage duration less than the expected rotation time.

The MQKP is solved using a greedy heuristic, where the knapsacks apply the football teaming principle taking turns at picking the best remaining item $max\ \Delta f(l_{ij}, d_{ij}, h_{ij}), i \in r, j \in \bar{V}$ where $\bar{V}$ are the unassigned items.

In a hub-and-spoke network design large vessels are deployed on deep sea services to achieve economies of scale[6], while smaller vessels are deployed between spoke and hub. The algorithm is multilayered to reflect the hub-and-spoke network design of major liner shipping operators. The function $f(l_{ij}, d_{ij}, h_{ij})$ is adapted to each layer of the network and the ports are correspondingly assigned to the layers according to their capacity requirements.

Computational results: The computational results are based on a real life case from Maersk Line with 234 unique ports and 14000 demands. The MQKP is able to find a solution in 33 seconds for the entire network with some ports unplaced. The solutions have been evaluated by optimization managers at Maersk Line regarding them as realistic with some modifications. Current work is on implementing layer specific seeding to improve the number of unplaced ports. We evaluate the actual flow and the network cost of the solution. We believe that meta heuristic approaches are needed to optimize liner service networks of global shipping operators and are working on specializing the Adaptive Large Neighbourhood Search [5] VRP framework towards the context of transshipments and cyclic routes. The first step is the construction heuristic for the LSN-DP presented here. The ALNS is to search for an improved solution according to a more sophisticated objective function and cargo allocation detection.

References

[1] K. Rana and R.G. Vickson, "Routing container Ships Using Lagrangian Relaxation and Decomposition", *Transportation Science* 25, 201-214 (1991).

[2] L. B. Reinhardt and B. Kallehauge, "Network Design for Container Shipping Using Cutting Planes",Conference on Maritime & Intermodal Logistics, Singapore, December 2007.

[3] R. Agarwal and O. Ergun,"Ship Scheduling and Network Design for Cargo routing in Liner Shipping",*Transportation Science* 42, 175-196 (2008).

[4] J.F. Alvarez, " Joint Routing and Deployment of a Fleet of Container Vessels", *Maritime Economics and Logistics* 11, 186-208 (2009).

[5] S. Røpke and D. Pisinger, "A general heuristic for vehicle routing problems", *Computers and Operations Research* 34, 2403-2435 (2007).

[6] I. Akio and K. Shintani and S.Papadimitriou, "Multi-port vs. Hub-and-Spoke port calls by containerships" *Transportation Research Part E* 42 , 740-757 (2009).

1. Midnight speaker:
Patrice Marcotte

Title:
BRANCHING, BOUNDING, CUTTING, PRICING, PRUNING, DIVIDING AND
CONQUERING MATHEMATICAL PROGRAMS WITH EQUILIBRIUM
CONSTRAINTS

Abstract:
Building upon the plenary talk of Martine Labbé, I will try to convince this audience that
every optimization process should lead to a bilevel program. As a corollary, life itself is
bilevel. Since the time allotted to this talk is limited, the elegant proof of this result, which
involves the paradigms stated in the above title, will be deferred to a forthcoming Tristan
conference. So will be the numerical results, based upon a parcimonious implementation
of Wolfe's universal algorithm.

Robust Approximate Dynamic Programming for dynamic routing of a vehicle

Stephan Meisel

Carl-Friedrich Gauss Department

University of Braunschweig, Germany

Dirk Mattfeld

Carl-Friedrich Gauss Department

University of Braunschweig, Germany

Stephan Meisel

Carl-Friedrich Gauss Department

University of Braunschweig,

Muehlenpfordtstr. 23 38106 Braunschweig, Germany

Email: stephan.meisel@tu-bs.de

1 Problem Description

Creation of an Approximate Dynamic Programming (ADP) solution for a specific optimization problem requires both a number of design decisions and a number of decisions on the values of parameters to be set. Taking these decisions the question of the robustness of the resulting algorithm arises, where robustness is defined with respect to variations of the attributes of the problem instance considered.

We provide and analyze a robust ADP algorithm for a dynamic vehicle routing problem with a single vehicle and stochastic customer requests. Details about the relevant problem attributes considered in our analysis are provided in Section 3. The remainder of this section is devoted to the basic description of the vehicle routing problem considered.

In our one vehicle problem, both location and request probability of each customer are known. The goal is maximization of the total number of customers served within a given time horizon. Customer requests appear randomly over time, with each customer requesting at most once. A request has to be either confirmed or rejected after becoming known.

At time $t = 0$ a set of customer requests is already known (early requesting customers). An initial route comprising each of the early request customers and leading from a given start depot to a given end depot is determined. A new route is determined each time new customer requests have been confirmed after completion of a vehicle operation. In particular, completion of a vehicle

operation entails the following events:

- New customers requests (occurring during the previous vehicle operation) become known.

- The new requesting customers are either confirmed or rejected.

- A number of routes including each of the new requesting as well as confirmed customers is determined.

- The next vehicle operation is determined. Possible vehicle operations are either waiting at the current location or proceeding to one of the remaining customers.

This problem emerges within the field of dynamic and stochastic vehicle routing (see, e.g. [5, 4]). It is of practical relevance e.g. in the context of package express or less-than-truckload applications. A similar problem has recently been considered in the literature [2].

2   Solution Methods

A Dynamic Programming formulation of the problem is derived. This model serves as a precondition for our ADP algorithm. In addition a number of heuristic solution methods for the problem are considered. Comparison of the performance of these specific methods to the performance of the ADP algorithm provides deeper insights into the robustness of the ADP approach. In particular three types of solution methods are consulted:

- Approximate Dynamic Programming: We formulate Bellman's Equation for the problem around the post-decision variables as proposed by Powell [3]. Subsequently a value iteration algorithm is derived based on the procedure we proposed in [1].

- Waiting strategies: The performance of the ADP approach is compared to the performance of a number of waiting strategies similar to the waiting heuristics developed by Thomas [2].

- Greedy heuristic: The results of a greedy heuristic serve as a baseline for the assessment of the performance of both the waiting strategies and the ADP approach.

3   Robustness Analysis

The robustness of the ADP approach is analyzed with respect to the following attributes of a problem instance:

- **Geographical distribution of the customer locations.** We analyze the performance of the ADP algorithm on three types of instances. In particular, we consider instances with geographically clustered customer locations, instances with equally distributed locations and semi-clustered instances.

- **Number of early requesting customers.** We consider instances with 20%, 40% and 80% of the requesting customers being early request customers.

- **Distribution of the request probabilities.** We analyze the performance of ADP with respect to request probabilitites generated from different distributions.

The computational results show that the approach leads to good solutions for any of the fairly different problem instances considered. Moreover, analysis of the ADP results in comparison with the quality of the solutions derived by the heuristic methods identifies crucial problem attributes for the performance of ADP.

References

[1] S. Meisel and U. Suppa, and D. Mattfeld "Approximate Dynamic Programming for dynamic routing of a vehicle with request confirmations", ODYSSEUS 2009, Cesme, Turkey, May 2009.

[2] B. Thomas, "Waiting Strategies for Anticipating Service Requests from Known Customer Locations", *Transportation Science* 41, 319-331 (2007).

[3] W.B. Powell, *Approximate Dynamic Programming*, Wiley, Hobokoen, NJ (2007).

[4] A. Larsen and O.B.G. Madsen, and M.M. Solomon, "Recent developments in dynamic vehicle routing systems", in *The Vehicle Routing Problem: Latest Advances and new Challenges*, B.L. Golden and S. Raghavan and E. Wasil (eds.), 199-218, Springer, New York, (2008).

[5] H.N. Psaraftis, "Dynamic vehicle routing: Status and prospects", *Annals of Operations Research* 61, 143-164 (1995).

[6] M.M. Solomon, "Algorithms for the Vehicle Routing and Scheduling Problems with Time Windows", *Operations Research* 35, 254-265 (1987).

A Long-term Liner Ship Fleet Planning Problem With Container Shipment Demand Uncertainty

Qiang Meng

Department of Civil Engineering

National University of Singapore, Singapore

Email: cvemq@nus.edu.sg

Tingsong Wang

Department of Civil Engineering

National University of Singapore

Abstract

Increasing globalization and inter-dependence of various world economies is leading to a tremendous positive growth in the seaborne trade industry. In particular, highly-containerized trade by liner shipping is the fastest growing sector and occupies the most major place in the global seaborne trading transportation. AXS-Alphaliner (2007) reported that the shipping capacity in terms of the TEU containers deployed on the liner shipping service trades has been increased more than double from January 2000 to January 2007. UNCTAD (2008) highlighted that the increase trend of the global container traffic for a long-term horizon. To cope with the period-dependent container shipment demand pattern within a long-term time horizon, a liner shipping company thus has to project its fleet size, mix and deployment on its shipping service routes, which is referred to as the long-term liner ship fleet planning (LTLSFP) problem.

The container shipment demand between any two ports along each liner trade routes operated by the liner container shipping company is input of this problem. The port-to-port container shipment demand at one period, say one year, within a long-term planning time horizon for a liner shipping company is usually estimated or predicted by some time series forecasting methods. The historical container shipment data fully shows uncertainty of the forecasted container shipment demand. A possible explanation is that the liner shipping company encounters many unknowns in the face of container shipment because shippers are allowed to cancel their container shipping contracts signed with the liner shipping company. This paper therefore focuses on model development and solution method design for the LTLSFP problem by taking into account uncertainty of container shipment demand.

To formulate uncertainty of the forecasted container shipment demand, it is assumed that the number of containers transported from one port to another port during a particular period within the planning time horizon is a discrete random variable taking a limited number of possible values with a given probability distribution. In other words, each of these possible values reflects a demand scenario with a known occurrence probability. It is further assumed that there are a number of predetermined fleet size and mix plans comprising ship types and number of ships at each period. A fleet size and mix plan is determined by the planners in the liner shipping company according to their experiences and the budget available. According to a fleet size and mix plan proposed for a particular period, the liner shipping company is able to determine the number of ships purchased or chartered. Given a specific fleet size and mix plan for a particular period with the stochastic container shipment demand, we build a two-stage stochastic programming model to determine the best fleet deployment plan (deploy ships on each service route) by maximize expected value of the profit gained during this period. The induced two-stage programming model can be solved by using a dual decomposition method based integrating sample average approximation method and Lagrangian relaxation method (Schütz et al, 2009). The proposed LTLSFP problem is thus formulated as a multi-period stochastic programming model comprising a sequence of interrelated two-stage stochastic programming models developed for each period in order to maximize the expected value of the total profit within the planning time horizon.

We further show that the multi-period stochastic programming model can be equivalent transformed into a shortest path problem defined on an acyclic network. Hence, we can apply any shortest path algorithm for solving the LTLSFP problem with the container shipment demand uncertainty. Finally, a numerical example is carried out to assess applicability and performance of the proposed model and solution algorithm.

References

[1] AXS-Alphaliner, "Liner Shipping Report". Available from: http://www1.axsliner.com/WWW/research_files/liner_studies/no32/BRS-rep-0701.pdf, (2007).

[2] P. Schütz, A. Tomasgard, and S. Ahmed, "Supply chain design under uncertainty using sample average approximation and dual decomposition", *European Journal of Operational Research*, 199, 409-419 (2009).

[3] UNCTAD, "Review of Maritime Transportation 2008. Paper presented at the United Nations Conference on Trade and Development". New York and Geneva, USA. Available from: http://www.unctad.org/en/docs/rmt2008_en.pdf (2008)

Comparison of control strategies for real-time optimization of public transport systems

Juan Carlos Muñoz, Ricardo Giesen, Felipe Delgado

Department of Transport Engineering and Logistics

Pontificia Universidad Católica de Chile

Email: jcm@ing.puc.cl, giesen@ing.puc.cl, fadelgab@ing.puc.cl

Aldo Cipriano

Department of Electrical Engineering

Pontificia Universidad Católica de Chile

Email: aciprian@ing.puc.cl

Cristián E. Cortés, Doris Sáez, Francisco Valencia

Facultad de Ciencias Físicas y Matemáticas

Universidad de Chile

Email: ccortes@ing.uchile.cl, dsaez@ing.uchile.cl

1 Introduction

The design of a public transport system operated by buses requires the optimization of planning variables, such as routes, fleet composition and frequency. Even though operational variables, such as the frequency, are optimized for different periods and lines, it is difficult to regularize the movement of buses as they are affected by different disruptions as the day progresses, such as traffic congestion, unexpected delays, randomness in passenger demand (both spatial and temporal), irregular vehicle dispatching times, incidents and so on. In the literature, as an attempt to reduce the negative effects of service disturbance, researchers have devoted significant effort to develop flexible control strategies, either in real-time or off-line, depending on the specific features of the problem. The absence of a control system in a bus network usually results in vehicle bunching due to the stochastic nature of traffic flows and passenger demand at bus stops. It also leads to an evident increase in bus headway variance and a consequent worsening of both the magnitude and variability of average waiting times (and also of the number of passengers per vehicle). This in turn impacts heavily on the level of service as perceived by users given that their subjective valuation of this component of total trip time is higher than that of any other (access time, in-vehicle time) [1].

These strategies have been designed to allow the operator reacting dynamically to real-time system disturbances. The most studied strategy of this type in the last years is the holding strategy, in which vehicles are held at certain stations for a determined time, in most cases designed to keep the headway between successive buses deterministic as far as possible. Hickman [2] developed a stochastic holding model at a given control station, obtaining a convex quadratic program in a single variable corresponding

to the time lapse during which buses are held. More recent research has explored holding models relying on online vehicle location. Among them, Eberlein [3] and Eberlein et al. [4][5] developed deterministic quadratic programs under a rolling horizon scheme. In their approaches, the holding decision for a specific vehicle affects the operation of a specific subset of the precedent vehicles. The authors concluded that having two or more holding stations over a corridor is not necessary. On the opposite, Sun and Hickman [6] concluded that holding multiple vehicles in several control stations would be better than having a single station to hold buses.

The vehicle capacity constraint is addressed in Zolfaghari et al [7], who formulate a problem in which the objective function minimizes the waiting times both of users who arrive at a stop and those who have to await more than one bus due to the activation of the capacity constraint. The authors do not, however, consider the extra waiting time endured by passengers held at a stop. Puong and Wilson [8] extend this case by including the latter factor in their objective function in the context of interruptions in train service. They propose a non-linear mixed-integer model in which dwell time is assumed to be constant at any given station. The problem is solved in a reasonable amount of time using a branch-and-cut strategy.

In this study we will focus at comparing the impact of different holding–only control strategies determining which buses are to be held where and for how long. The comparison study is based on two approaches: one deterministic able to optimize over the entire simulation period, but assuming that future stochasticity only depends on the mean values of demand at stops: the second approach is stochastic, based on a hybrid predictive control formulation, which assumes explicitly the stochastic behavior for future demand prediction, but only considering a finite number of steps ahead to perform the online optimization. Different scenarios regarding design frequency and demand levels (capacity being reached and not reached) will be studied to identify under which conditions each strategy outperforms the other.

2 Problem Formulation and Control Strategies

The network is a one-way loop route, with P equidistant stops and b buses running around the loop, under the control of the dispatcher. Passengers arrive at each station at a certain rate by following a negative exponential distribution, with destinations randomly chosen among the stations downstream the station where the passenger is boarding. In addition, batch arriving processes can be considered in case of observing group arrival patterns. From historical data, a representative stop-to-stop demand matrix can be estimated for each modelling period; this is crucial for adding the predictive feature in the real-time model of the system. Online demand data can also be used as a complement to the offline demand matrix to improve this predictive aspect. It is assumed that at any time instant we have real-time information on the position and number of passengers aboard each bus as well as the number of passengers waiting at the various stops. The events are triggered when a bus arrives at a bus stop, which determines a variable time-step. Hereafter, we denote t as the continuous time, k as the event, and t_k as the continuous time at which event k occurs. Note that an event k is always associated with the arrival of a specific bus to a specific bus stop.

2.1 Deterministic control strategy

The first control approach is based on a deterministic non-linear mathematical programming model, which is used on a rolling horizon framework to update operational plans [9]. Each time a bus arrives to a stop the model is solved and determines the holding times of the buses at the various stops along the corridor by each departing bus at each stop, assuming that average travel times between stops and arrival rates of passengers at each stops would occur. The model objective is to minimize the total travel times of passengers from the moment they arrive at a stop to the moment they reach their destination. Since vehicle running times are assumed to be constant, the objective is to minimize both in-vehicle and at-stop waiting times, including explicitly the waiting time experienced by passengers who must await more than one bus due to the capacity constraint. Among the key elements of this model are: i) capacity constraints on buses are considered explicitly without using binary variables which would increase the solution times, thus allowing to consider a long planning horizon at each update; ii) an objective function that distinguish between waiting time at-stop, in-vehicle delay due to holding, and waiting times of passengers forced to

await more than one bus due to capacity constraints, allowing to consider different weights for each one; and, iii) the duration of each holding can take any continuous value.

The system is then completely determined by the following state variables: d_i distance between bus i and the last stop upstream, e_i stop immediately upstream from bus i (if bus i is at stop p, then $e_i = p-1$), B_{ipe_i+1} number of passengers on bus k who boarded at stop i, before arriving at the stop immediately downstream from bus k. ($\forall p < e_i +1$) and Γ_p number of passengers waiting at stop p [9]. The objective function of this model is formulated as follows:

$$Min \quad \theta_1 W_{first} + \theta_2 W_{in-veh} + \theta_3 W_{extra} \tag{1}$$

The first term in (1) refers to the at-stop waiting time experienced by passengers as they wait for the first bus to arrive after the update epoch. This term lends the objective function its non-linear nature given that total waiting time for all users is proportional to the square of bus headway. As for the second term in (1), it states the in-vehicle waiting time for passengers on-board a bus i being held at stop p. The third term represents the extra waiting time of passengers who are prevented from boarding bus i because it is at capacity. Each of the three terms is multiplied by a different weighting factor, θ_1, θ_2, and θ_3. The objective function in (1) is quadratic in h_{ip} but not convex, whereas the model's constraints are linear. To obtain solutions, this mathematical programming model was solved using MINOS.

2.2. Hybrid predictive control strategy

The second approach corresponds to a hybrid predictive control strategy (HPC) proposed in Sáez *et al.*[10] and Cortes *et al.* [11] for a real-time bus system optimization, which is based on a state-space model and an ad-hoc objective function. The model is of stochastic nature, then it considers explicitly the uncertainly of future demand in the prediction but over a shorter horizon than that used in model 2.1. In this work the manipulated variable is the holding $h_i(k)$ action associated with bus i and event k, which is the lapse during bus i is held at the stop associated with event k. The output variables correspond to the estimated passenger load $\hat{L}_i(k+1)$, the estimated headway $\hat{H}_i(k+1)$ of bus i that triggers the event k, with respect to its precedent bus *i-1* when it reaches the same stop The analytical expressions for such a dynamic model be summarized as follows in [10], [11]. The corresponding objective function for HPC is given by:

$$\min_{\{u(k),u(k+1),...,u(k+Np-1)\}} \sum_{\ell=1}^{Np} \Big[\theta_1 \cdot \hat{H}_i(k+\ell)\hat{\Gamma}_p(k+\ell) + \theta_2 \cdot (\hat{H}_i(k+\ell) - \bar{H})^2$$
$$+ \theta_3 \cdot \hat{L}_i(k+\ell)h_i(k+\ell-1) + \theta_4 \cdot \hat{L}_i(k+\ell)\hat{Tr}_i(k+\ell-1) \Big] \Big|_{\substack{i=i(k+\ell-1) \\ p=p(k+\ell-1)}} \tag{2}$$

where $\{u(k),...,u(k+Np-1)\}$ is the control-action sequence with $u(k+\ell-1) = h_i(k+\ell-1)$ when bus i triggers event $k+\ell-1$. Np is the prediction horizon and b is the number of buses in the fleet. Note that $i = i(k+\ell-1) \in \{1,...,b\}$, $p = p(k+\ell-1) \in \{1,...,P\}$, if we consider that the future event $k+\ell-1$ is triggered by one bus $i(k+\ell-1)$ arriving to a specific station downstream $p(k+\ell-1)$. In expressions (2), θ_j, $j = 1,...,4$, are weighting parameters, and have to be tuned depending on the specific problem to be treated and on the physical interpretation of the different components as well. $\hat{Tr}_i(k)$ is the estimated time associated with passenger transference (maximum between the boarding and alighting times). $\bar{H}$ corresponds to the desired headway (set-point) designed for servicing the system demand during a certain time period. The first term in (2) quantifies the total passenger waiting time at stops and depends on the predicted headway along with the bus stop load. The second term captures the regularization of bus headways, to maintain the headway as close as possible to the design headway. The third component

measures the delay associated with passengers on-board a vehicle when they are held at a control station due to the application of the holding strategy. The fourth component corresponds to the extra travel time incurred by the passengers on board due to the transference of passenger process. We use genetic algorithms in order to dynamically solve the formulation in (2) with operational constraints.

3. Simulation Results

The proposed model is now applied, to a prototype of a public transport corridor with 10 km of length, comprising 30 evenly spaced bus stops. Vehicle operating speed between stops for all of the buses is assumed to be 26 Km/h, while boarding and alighting time per passenger is set at 2.5 and 1.5 seconds respectively. In order to evaluate and compare the proposed model under different operational conditions, two different frequency levels are tested in scenarios in which bus loads are concentrated around the center of the corridor: (a) high frequency services are operated and bus capacities are reached; (b) medium frequency services are operated and bus capacities are not reached. For every combination of strategies and scenarios, 30 replications were conducted, each of them representing 2 hours of operation. A warm-up period of 15 minutes is considered for all scenarios, before any control strategy is applied. The next table shows the results with both strategies under the two scenarios defined.

Control strategy (scenario)	Waiting Time (mean) [min]	Waiting Time (std) [min]	Holding Time (mean)[min]	Holding Time (std) [min]
Deterministic (a)	8994.3	429.0	3229.8	271.3
HPC (a)	10117.6	510.7	2746.2	219.3
Deterministic (b)	6625.8	171.7	754.6	101.0
HPC (b)	6917.7	268.0	1751.3	136.6

Results show the trade off between strategies. Among the scenarios tested the deterministic approach assuming a perfect knowledge of the future conditions reports better results. We are now testing different demand conditions under which each approach can result in better holding decisions. The comparison is also being extended to other control rules, such as station skipping, overtaking and so on.

Acknowledgments

References

[1] Ortúzar, J. de D., 1983. Nested Logit Models for Mixed-mode Travel in Urban Corridors. Transportation Research, Vol. 17A, No. 4, pp. 283-299.

[2] Hickman, M., 2001. An analytic stochastic model for the transit vehicle holding problem. Transportation Science 35 (3), pp. 215–237.

[3] Eberlein, X.-J., 1995. Real-Time Control Strategies in Transit Operations: Models and Analysis. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA.

[4] Eberlein, X.-J., Wilson, N.H.M., Bernstein, D., 1999. Modeling real-time control strategies in public transit operations. In: Wilson N., (ed.), Computer-Aided Transit Scheduling, Lecture Notes in Economics and Mathematical Systems 471, Springer-Verlag, Heidelberg, pp. 325-346.

[5] Eberlein, X.-J., Wilson, N., Bernstein, D., 2001. The holding problem with real-time information available. Transportation Science 35 (1), pp. 1-18.

[6] Sun, A. and Hickman M., 2004. The Holding Problem at Multiple Holding Stations [online]. 9th International Conference on Computer-Aided Scheduling of Public Transport (CASPT) San Diego, California, EE.UU. Available from: http://fugazi.engr.arizona.edu/caspt/sun.pdf

[7] Zolfaghari S., N. Azizi, and Jaber, M. 2004. A Model for Holding Strategy in Public Transit Systems with Real Time Information. International Journal of Transport Management, Vol. 2, pp. 99-110.

[8] Puong A. and Wilson, N.H.M. 2004. A Train Holding Model for Urban Rail Transit Systems. 9th International Conference on Computer-Aided Scheduling of Public Transport (CASPT) San Diego, California, 2004. http://fugazi.engr.arizona.edu/caspt/puong.pdf. Accesed March 15, 2008.

[9] Delgado, F., Muñoz, J.C., Giesen, R., Cipriano, A. 2009. Real-Time Control of Buses in a Transit Corridor Based on Vehicle Holding and Boarding Limits. Transportation Research Record 2090, pp. 59-67.

[10] Sáez D., Cortés, C. E., Riquelme, M., Núñez, A., Milla F., Tirachini, A. 2010. Hybrid Predictive Control Strategy for a Public Transport System with Uncertain Demand. Transpormetrica in press.

[11] Cortés, C.E., Sáez, D., Milla, F., Riquelme, M., Núñez, A. 2009. Hybrid Predictive Control for Real-time Optimization of Public Transport System' Operations based on Evolutionary Multiobjective Optimization. Transportation Research C, doi:10.1016/j.trc.2009.05.016.

Distance to the city center and travel behavior: A Case of Ahmedabad City

Dr. Mark Zuidgeest

Assistant Professor, Department of Urban and Regional Planning and Geo-information Management, ITC, University of Twente, The Netherlands

Talat Munshi

Associate Professor, Faculty of Planning and Public Policy

CEPT University, Navrangpura, Ahmedabad – 380009, Gujarat, India

talatmunshi@gmail.com, munshi@itc.nl

1. Introduction

Metropolitan regions in Asia and Africa are growing in terms of their economy, population and spatial extension and as economies grow and focus of development shifts from Industry based to trade and commerce base. Fast economic and urban growth coupled with planning and control methods practices followed in these contexts have caused decentralization of population, jobs and services from inner dense core of the cities to less densely developed suburbs [1]. The recent reforms and developments in a country like India sets up the need to add to the debate on efficiency of diverse urban forms and its impact on travel behavior; which till now is mostly focused on the Anglo-American setting. As the area and the context is not well researched several questions need to be answered as in how and if decentralization of jobs has happened, and polycentric development as in new employment center formation has caused a change in travel characteristics like trips length and choice of mode. Therefore, this paper studies employment trends and change in spatial concentration from 1985 to 2007 to determine the transformation of employment locations to explain the emergence of new centers of employment and their growth pattern. This is then related to the change in average trip length and modal split during the same time period. To develop an understanding on the relation between trip length, mode choice and choice of the employment centre, choice of the employment centre is related to distance individuals travel by different modes for work and shopping purpose trips.

2. Distances Travelled and Choice of Mode and Sub-centers

The concept of polycentric urban areas has been well researched. The characteristics of the non-mono-centric city as defined in Bhandari [1] and Alpkokin P. et al [5] wherein, CBD is still the strongest

center, but loses its share of relative metropolitan employment, and its absolute growth is light as in comparison to growth elsewhere in the region. In literature the mono-centric or non-mono centric structure of a metropolitan region or an urban area has mainly been related to commuting time. Dubin [11] discusses as cities get larger in terms of area and population, they might produce more cross commuting and individual will tend to reduce their travel time by taking up new opportunities. Other authors (e.g. [12-13] [12-13] also suggest that sub-urbanization of employment from the employment centre tends to reduce commuting time and commuting distance. Contrary to this Cevero and Kang Li [14] found that there was a significant modal shift and changed commuting time from 1980 to 1990 in San Francisco area. The effect of de-concentration of traditional city centers has also been analyzed for its effect on modal split. Gordan and Richardson [15] have analyzed the effect of de-concentration on modal split in San Francisco area in terms of the shift in commuting behavior from using public transport to personal modes of transport, similar effects were found to exist for Oslo [16].

Thus it can be observed that mono-centric or poly-centric character of urban form and de-congestion of the employment centre has been studied extensively in relation to change in travel patterns over time. It is the function of availability and spatial distribution of opportunities, the urban form of the location and socio-demographic of individuals.

3. Data and Methods

For the city of Ahmedabad data on jobs in the year 2001 and job projections for the year 2011 is available from [17] at the scale of Traffic Analysis Zones (average size of 550 households). Interpolated values for 2007 from this data were disaggregated to 100 m. x 100 m. grid cell using data derived from combining data on building footprints[18], land use[19] and building heights generated using a digital elevation model created using IRS Cartosat -1 data [20]. For the year 1985 land use map prepared by AMC was updated by Jain [21]. Employment estimates from TCS [22] study on Ahmedabad done in the year 1986 was used to quantify spatial job distribution in 1985. Clusters of employment and city centers were qualified based on methods used in Giuilano and Small [7, 23], where the minimum jobs in a contiguous area with minimum density more than 300 person/acre of 10,000 was considered as a qualifying mark for identifying the city centre.

To determine change in trip length and mode choice Ahmadabad transport study conducted by Central Institute of Road Research, Pune in 1988[22] for Government of India, the feasibility study for IPTS alternatives for Ahmedabad by Louis Berger Group Consortium for Gujarat Infrastructure Development Board in 2001-02[17] and recently (2005-2006) conducted study for Bus Rapid Transport System in Ahmedabad by CEPT University for Ahmedabad Municipal Corporation were used. A survey of 4500 households across the city was conducted and geo-referenced to the location of residence of the respondent to analyze the relation of trip length with the choice of employment centre.

Trip made for shopping and work purposes were analyzed, distance traveled for work trips was computed from base location (place of residence) and distance for shopping trip was computed from origin of the trip, which could be place of residence or workplace or else.

From the analysis done on spatial concentration of jobs six centers of employment could be identified, of these three centers were trade and commerce centre (Cn1) and three industrial centers (Cn2) . Discrete choice model (binary logit with one predictor) with the choice of a particular centre (as against choice of all other centers within the choice set (Cn1; The set Cn2 is not analyzed as very little change in employment concentration levels was observed) as a dependent variable and trip length as an independent variable was developed to understand the relation between trip length and employment centre choice.

4. Main Findings and Conclusion

The economy of Ahmedabad has shifted over the past three decades from manufacturing and industrial based to trade and commerce based [24-25], this is reflected in de-concentration of jobs but to a large extent addition of new job. The result is the formation of two additional city centers, which satisfy the criterion mentioned earlier. It is observed that the two of these main centers are representative of the old centre (the tradition centre was the walled part of the city called walled city and the area that has grown in a continuum on the western part of river running through the centre of the town). The growth of activities has shown negative growth in the walled city area, the outer areas, including CG road, which is representative of a recently developed poly centric node in the city has grown at a fast rate in the past two decades. More than 80 percent work and shopping purpose trips get consumed between Wall city and CG road employment centers, which are located in the central parts of the town.

Looking changes in mode spit in from 1985 to 2007 it is observed that overall, there is small decline in trip made by walking but it still accounts for the maximum share of trips. Share of two wheelers for work trips has almost doubled in two decades; the shift has been from public transport and non-motorized decades; the shift has been from public transport and non-motorized modes.

The total trips increased from 5.55 million in 1988 to 6.54 million in 2006 but the average trip rate/person decreased from 1.57 in 1988 to 1.1 in 2000 and 2006 survey. The decrease in trip rates in Ahmedabad to 1.1 is indicative of fewer non-obligatory trips like shopping and recreation made in the city. The average trip lengths have increased from 4.6 km to 5.6 km in 2001 and to 6.2 in 2006. However, it is still observed that most of the trips are short distance trips very few trips are longer than 10 km. for work purpose and 2.5 km. for shopping purposes. The increase in trips length is negligible and in the scenario that the city would have remained mono-centric the average lengths could have been 7.25 km. which is larger than observed average trip lengths. Thus development of poly centers

has reduced trips lengths in the city to a considerable extent but increase share of private motorized modes.

When the choice of these two city centers are analyzed against distance traveled it is found that walled city centre has a negative beta value whereas CG road centre has a positive beta value and similarly odds ratio are also found on different sides of one (higher than 1 for CG road lower than 1 for Walled city). So it can be said that the traditional centre is losing its gravity over other centre's as the distance travelled increase, both in the case of shopping purpose and work purpose trips. Because the average trip distance in Ahmedabad are relative low at about 5 km., wall city is still the preferred destination, but as the city expands or congestion level or travel distance increases more jobs will shift out from the wall city area to new place in the city.

In terms of mode choice, it is found that wall city is more interesting as the destination for people who use non-motorized transport, and public transport modes. Higher β were observed for all modes for trips made to wall city area, indicating more utility for short trips, but the odd ratio values observed for CG road are very close to 1 for trips made using car and motor cycle as a mode, where as these values are in the range of 0.7 for trips made to walled city area, explaining the choice of mode and trip lengths to these centers. The fact that individuals with private mode are willing to travel more to the new center/s of employment is also indicative of better opportunities available in these areas.

The implication of the result presented here on the city development is quite significant. It basically helps us to understand the driver of poly-centricity in the city and also to establish that poly-centricity does affect travel distance and also contrary is also true. Decentralization of urban land use and development of non-mono-centric urban structure seems to lead to more use of private modes of transport, which may be related to how these centre's are planned and connected to the network.

References

[1] Bhandari, K., et al., Policies, Commuting Patterns and Accessibility in a Non-Monocentric City: A Case Study of Delhi. Proceedings of the Eastern Asia Society for Transportation Studies, 2007. 6.

[2] Bocquier, P., World Urbanization Prospects: an alternative to the UN model of projection compatible with the mobility transition theory. Demographic Research, 2005. 12(9): p. 197-236.

[3] Ravallion, M., On the urbanization of poverty. World, 2001.

[4] Dimitriou, H., Towards a generic sustainable urban transport strategy for middle-sized cities in Asia: Lessons from Ningbo, Kanpur and Solo. Habitat International, 2006. 30(4): p. 1082-1099.

[5] Alpkokin, P., et al., Dynamics of Clustered Employment Growth and Its Impacts on Commuting Patterns in Rapidly Developing Cities. Transport Research Part A, 2008. 42: p. 427-444.

[6] Redfearn, C.L., The Topology of Metropolitan Employment: Identifying Centres of Employment in a Polycentric Urban area. Urban Economics, 2007. (61): p. 519-541.

[7] Giuliano, G. and K.A. Small, Subcenters in the Los Angeles region. Regional Science and Urban Economics, 1991. (21): p. 163-182.

[8] Anderson, N.B. and W.T. Bogart, The Structure of Sprawl-Identifying and Characterizing Employment Centres in Polycentric Metropolitan Areas. American Journal of Economics and Sociology, 2001(60): p. 147-169.

[9] Bogart, W.T. and W.C. Ferry, Employment Centres in Greater Cleaveland: Evidence of Evolution in a Formerly Monocentric City. Urban Studies, 1999(36): p. 2099-2110.

[10] Small, K.A. and S.F. Song, Population and Employment Densities. Journal of Urban Economics, 1994(36): p. 292-313.

[11] Dubin, R., Commuting patterns and firm decentralization. Land Economics, 1991. 67(1): p. 15-29.

[12] Gordon, P. and H.W. Richardson, Are compact cities a desirable planning goal? Journal of American Planning Association, 1997. 63(1): p. 95-106.

[13] Levinson, D. and A. Kumar, Density and Journey to Work. Growth and Change, 1997. 28(2): p. 147-172.

[14] Cervero, R. and W. Kang-Li, Sub-Centering and Commuting: Evidence from the San Francisco Bay Area, 1980-90. Urban Studies, 1998. 35(7): p. 1059-1076.

[15] Cervero, R. and J. Landis, Suburbanization of Jobs and the Journey to Work, in Working Paper. 1991, University of California Transportation Center, Berkeley.

[16] Naess, P. and S.S. Sandberg, Work Place Location, Modal Split and Energy Use for Commuting Trips. Urban Studies, 1996. 33(3): p. 557-580.

[17] GIDB, Feasibility Study on Integrated Public Transit System for Ahmedabad. 2004, Gujarat Infrastructure Development Board: Ahmedabad.

[18] Katuri, A.K., M.A. Sharifi, and C.J. Westen. Urban mapping and land use characterisation for risk assessment. in 2nd international symposium on Geo-Information for disaster management Gi4DM : remote sensing and GIS techniques for monitoring and prediction of disasters. 2006. Goa, India.

[19] AMC, AUDA, and CEPT, City Development Plan Ahmedabad 2006-2012. 2006: Ahmedabad.

[20] Bock, B.B., Creating a DSM from CARTOSAT-1 Imagery: The Case of Ahmedabad, India. 2008, The International Institute for Geo-information Science and Earth Observation (ITC): Enschede, The Netherlands.

[21] Jain, D., Studying and modelling changing urban form, in Department of Urban and Regional Planning and Geo-information Science. 2009, International Institute for Geo-information Science and Earth Observation (ITC): Enschede.

[22] CIRT, Ahmedabad Transport Study, Central Institute of Road Research, Editor. 1988, Government of India: Ahmeadbad.

[23] Giuliano, G. and K. Small, Is Journey to Work Explained by Urban Structure? Urban Studies, 1993. 30(9): p. 1485-1500.

[24] Adhvaryu, B., Enhancing Planning in Developing Countries, Urban Modelling for Ahmedabad, India, in Churchill College and Department of Architecture. 2009, University of Cambridge: Cambridge.

[25] Adhvaryu, B., Enhancing urban planning using simplified models: SIMPLAN for Ahmedabad, India. Progress in Planning, 2010.

[26] Munshi, T., Planning Public Transport of Ahmedabad City in India, in UPLA. 2003, ITC: Enschede.

Sensitivity analysis of velocity and fundamental traffic flow diagrams from modelling of vehicle driver behaviors

Lorenzo Mussone

Politecnico di Milano

Department of Building Environment and Science Technology

Email: mussone@polimi.it

Ida Bonzani

Politecnico di Torino

Department of Mathematics

Email: ida.bonzani@polito.it

1 Introduction

This paper deals with the sensitivity analysis of the velocity and fundamental diagrams derived in steady uniform flow conditions and based on a detailed analysis of the individual behavior of the driver-vehicle subsystem. As known [1,2], these diagrams represent the mean velocity and the flux, respectively, of vehicles according to the hydrodynamic description, namely by locally averaged macroscopic quantities.

An extensive review of the existing literature is available in papers [1,2] and [3], which report that traffic flow phenomena can be modelled at three different scales (macroscopic, microscopic and kinetic) as documented by the literature on the field. The book of Kerner [4] offers a detailed interpretation of the physics of traffic phenomena. Many specific phenomena observed in traffic flow conditions are reported in [4] from the viewpoint of physics.

The knowledge of the velocity and fundamental diagrams is an essential requirement to derive the greatest part of models at the macroscopic scale also in the case of crowd dynamics modelling, as documented in [5]. Indeed, several models (almost all) are based on the assumption that vehicles have a natural trend towards equilibrium. On the other hand, some recent papers have shown that a detailed modelling of the behaviors of the driver-vehicle micro-system leads to models that naturally describe the afore mentioned trend without artificially imposing it. This result is achieved in [6] at the macroscopic scale, and in [7] in the case of kinetic type models. This result is confirmed in [8] in the case of multilane flow. The above contributions [6-8] motivate the contents of the paper focused on the derivation of the velocity and fundamental diagrams by a detailed analysis of the dynamics at the microscopic scale. This modelling approach is useful for several reasons.

Among others:

(i) it provides a unified representation rather than a broad variety of experimental data collected by different devices in different environmental conditions;

(ii) the representation is useful at all scales, namely microscopic, macroscopic and kinetic;

(iii) the velocity (and fundamental) diagrams need only one parameter corresponding to the quality of the road in a broad sense, namely including weather conditions and traffic control actions;

(iv) the representation includes the transition from free to congested flow corresponding to different values of the parameter characterizing the model.

The content of this extended abstract is developed through two more sections. Specifically, Section 2 develops the modelling of the velocity and fundamental diagram, while in Section 3 a sensitivity analysis of the model is proposed.

2 Velocity model

The derivation of the equilibrium velocity diagram, the original focus of the research, can be pursued by a detailed analysis of the behavior of the driver-vehicle micro-system; experimental data are used only subsequently to the definition of the model to compare results.

The general principles followed to achieve such a result can be summarized as follows:

(i) drivers adjust the velocity of vehicles according to the local density, namely they attempt to maintain the velocity below a limit corresponding to the braking distance;

(ii) the velocity corresponding to the braking distance also depends on the quality of the road and environmental conditions and also on traffic control strategy;

(iii) we look for a minimal model characterized by one parameter corresponding only to the road-weather system.

According to the above reasoning, the behavior of the vehicle is modelled deterministically, while a stochastic generalization will be developed in the next section. Bearing all the above in mind, let us consider the one-directional flow of vehicles along a one lane road.

The modelling of the velocity diagram should reproduce qualitative behavior. The phenomenology of the system suggests [9] to approximate the safety distance between vehicles as a power expansion of the velocity truncated at the square power. Using dimensional variables yields:

$$d_s = L + t^* V + c V^2 , \qquad (1)$$

where L is the mean length of vehicles, t^* is a time approximating the psychological-technical delay and c is a constant with dimension of the inverse of an acceleration.

Eq. (1) can be written by dimensionless variables, resulting as follows

$$\delta_s = 1 + \tau v_e + \gamma v_e^2 , \qquad (2)$$

where τ represents the dimensionless delay time, v_e the mean velocity depending on the local density ρ and $\gamma = c\, L/T_r^2$. Reference values can be obtained using L = 6:6m, $V_{M=}$120 km/h and $t^* = 2$ s

which yield $T_r=0:2$ and $\tau=10$. Other relations concern the density range in which velocity is constant: this means that when $\rho \leq \rho_c$ $v_e=1$, otherwise $v_e<1$. Considered that when $v_e=1$, Eq.2 leads to $\gamma =1/\rho_c-(1-\tau)$ and putting $\delta_s = 1/\rho$, Eq.2 can be rewritten as:

$$1/\rho = 1 + \tau v_e + (1/\rho_c-1-\tau)v_e^2 , \qquad (3)$$

ρ_c assumes the meaning of model parameter describing the quality of the road-environment system. The velocity diagram is then obtained:

$$\begin{cases} \rho \leq \rho_c & v_e = 1 \\ \rho > \rho_c & v_e = \dfrac{1}{2\gamma}\left[\sqrt{\tau^2 + 4\gamma\, \dfrac{1-\rho}{\rho}} - \tau \right] . \end{cases} \qquad (4)$$

More details of this model can be found in [10]. Figures 1 show different curves of velocity and fundamental diagram (the latter is derived according the classical relationship $q=\rho v$) by varying the value of ρ_c.
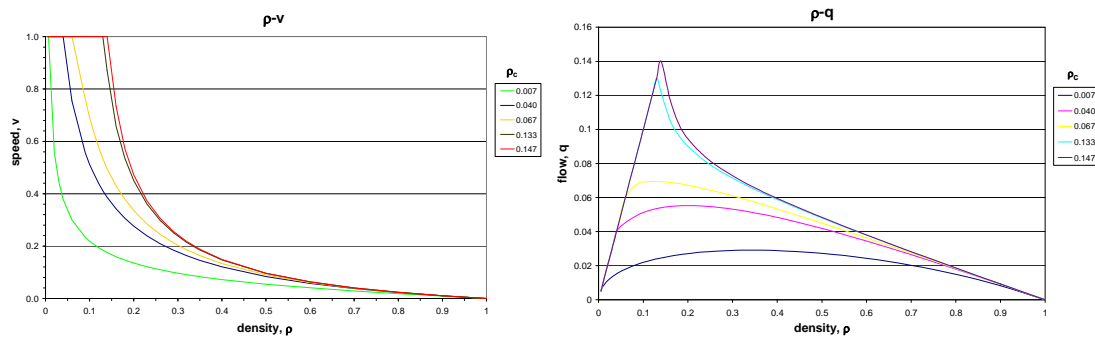
Figure 1: Velocity and Fundamental diagrams varying ρ_c.

3 Sensitivity Analysis

The effect of ρ_c on diagrams shows how environmental conditions (road and meteorological state) affect flow circulation, modifying the phase transition value of velocity and capacity (the maximum allowed flow). These results are obtained by assuming constant some parameters like reaction time t* (and hence τ).

Therefore it is of interest to investigate how other values of t* modify the shape of diagrams. It must be underlined that t* is generally considered a stochastic variable ranging in a wide interval from 0.7s to values greater than 3s.

Figure 2 analyses the effect on velocity and fundamental diagrams by varying τ. To help reader instead of τ the corresponding value of t* is reported in the legend. As expected a lower value of τ means lower capacity and above all a fast decay of performance after the transition point set by ρ_c.

Figure 3 takes into consideration the combined effect of varying both τ and ρ_c and it compares the effect of τ when two different values of ρ_c are considered (0.033 and 0.2). In particular when ρ_c is a high (0.2) the effect of τ is more evident and generally leads to very different performance: a high

value (t*=3s) means a fast decay both in the velocity and fundamental diagrams. A low value of ρ_c (for example 0.033) is not greatly affected by τ. In synthesis roads with large ρ_c (that is high level roads) are more sensitive to τ for high values of density; vice versa roads with low ρ_c (that is low level roads) are less sensitive to τ since their performance are largely low.
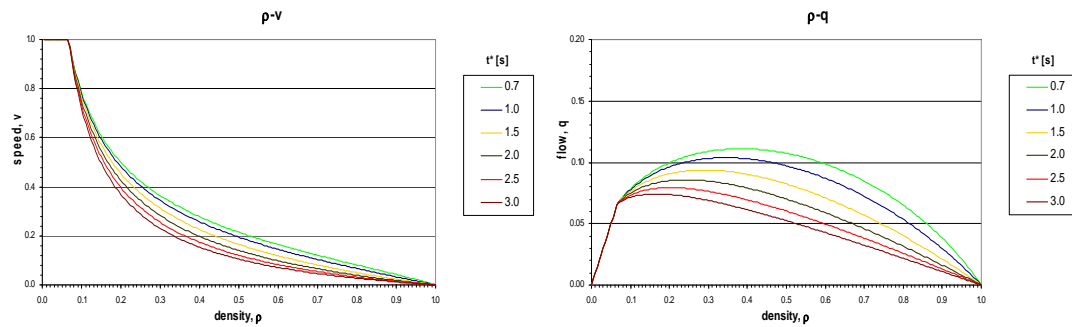

Figure 2: Sensitivity to τ (ρ_c=0.067 and n_M=150 [veh/km]) of Velocity and Fundamental diagrams.
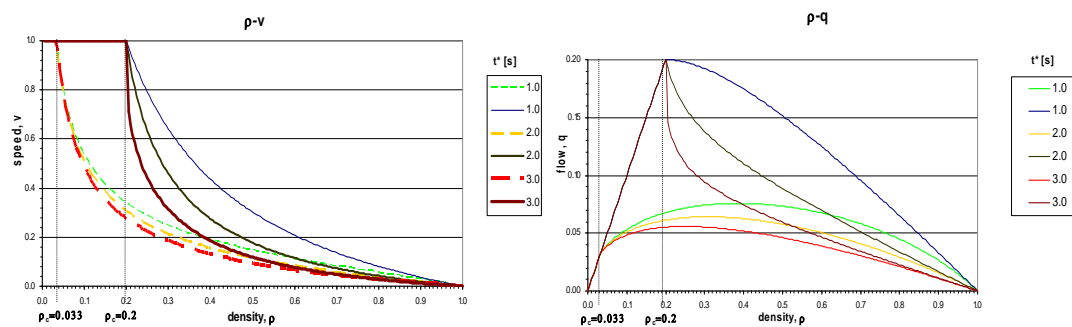
Figure 3: Sensitivity to the combined effect of ρ_c and τ of Velocity and Fundamental diagrams.

References

[1] N. Bellomo, A. Marasco, A. Romano, From the modelling of driver's behavior to hydrodynamic models and problems of traffic flow, Nonlinear Anal. RWA 3 (2002) 339-363.

[2] D. Helbing, Traffic and related self-driven many-particle systems, Rev. Modern Phys. 73 (2001) 1067-1141.

[3] I. Bonzani, L.M. Gramani Cumin, Critical analysis and perspectives on the hydrodynamic approach for the mathematical theory of vehicular traffic, Math. Comput. Modelling (2009), 50 (2009) 526-561.

[4] B. Kerner, The Physics of Traffic, Empirical Freeway Pattern Features, Engineering Applications and Theory, Springer, Berlin, 2004.

[5] N. Bellomo, C. Dogbé, On the modelling crowd dynamics from scaling to hyperbolic macroscopic models, Math. Models Methods Appl. Sci. 18 (Suppl.) (2008) 1317-1345.

[6] P. Degond, M. Delitala, Modelling and simulation of vehicular traffic jam formation, Kinetic and Related Models 1 (2) (2008) 279-293.

[7] M. Delitala, A. Tosin, Mathematical modelling of vehicular traffic: A discrete kinetic theory approach, Math. Models Methods Appl. Sci. 17 (2007) 901-932.

[8] I. Bonzani, L.M. Gramani Cumin, Modelling and simulations of multilane traffic flow by kinetic theory methods, Comput. Math. Appl. 56 (2008) 2418-2428.

[9] I. Bonzani, L. Mussone, From the discrete kinetic theory of vehicular traffic flow to computing the velocity distribution at equilibrium, Math. Comput. Modelling 49 (2009) 610-616.

[10] I. Bonzani, L. Mussone, On the Derivation of the Velocity and Fundamental Traffic Flow Diagram from the Modelling of the Vehicle-Driver Behaviors, Math. Comput. Modelling 50 (2009) 1107-1112.

Routing in Graphs with Applications to Logistics and Public Transport

Rolf H. Möhring

Institut für Mathematik, Technische Universität Berlin

Straße des 17. Juni 136, 10623 Berlin, Germany

Email: rolf.moehring@tu-berlin.de

1 Text

Traffic management and routing in logistic systems are optimization problem by nature. We want to utilize the available street or logistic network in such a way that the total network "load" is minimized or the "throughput" is maximized. This lecture deals with the mathematical aspects of these optimization problems from the viewpoint of network flow theory and scheduling. It leads to flow models in which—in contrast to static flows—the aspects of "time" and "congestion" play a crucial role.

We illustrate these aspects on several applications:

(1) Traffic guidance in rush hour traffic (cooperation with DaimlerChrysler).

(2) Routing automated guided vehicles in container terminals (cooperation with HHLA).

(3) Timetabling in public transport (cooperation with Deutsche Bahn and Berlin Public Transport).

(4) Ship Traffic Optimization for the Kiel Canal (cooperation with the German Federal Waterways and Shipping Administration).

All these applications benefit from new insights into routing in graphs. In (1), it is a routing scheme that achieves traffic patterns that are close to the system optimum but still respect certain fairness conditions, in (2) it is a very fast real-time algorithm that avoids collisions, deadlocks, and other conflicts already at route computation, while for (3), it is the use of integer programs based on special bases of the cycle space of the routing graph. Finally, (4) combines techniques from (2) with special purpose scheduling algorithms.

References

[1] E. Gawrilow, E. Köhler, R. H. Möhring, and B. Stenzel. Dynamic routing of automated guided vehicles in real-time. In W. Jäger and H.-J. Krebs, editors, *Mathematics — Key Technology for the Future. Joint Projects between Universities and Industry 2004 -2007*, pages 165–178. Springer, 2008.

[2] O. Jahn, R. H. Möhring, A. S. Schulz, and N. E. Stier Moses. System-optimal routing of traffic flows with user constraints in networks with congestion. *Oper. Res.*, 53(4):600–616, 2005.

[3] E. Köhler, R. H. Möhring, and M. Skutella. Traffic networks and flows over time. In J. Lerner, D. Wagner, and K. A. Zweig, editors, *Algorithmics of Large and Complex Networks – Design, Analysis, and Simulation*, volume 5515 of *Lecture Notes in Computer Science*, pages 166–196. Springer, 2009.

[4] C. Liebchen and R. H. Möhring. The modeling power of the periodic event scheduling problem: Railway timetables - and beyond. In F. Geraets, L. Kroon, A. Schöbel, D. Wagner, and C. D. Zaroliagis, editors, *Algorithmic Methods for Railway Optimization*, volume 4359 of *Lecture Notes in Computer Science*, pages 3–40. Springer, Berlin/Heidelberg, 2007.

Solving a Rich Vehicle Routing Problem in a cooperative real-world scenario

Andrea Nagel

Department of Information Systems

University of Hagen, Profilstrasse 8, 58084 Hagen, Germany

Email: andrea.nagel@fernuni-hagen.de

Giselher Pankratz

Department of Informations Systems

University of Hagen

Hermann Gehring

Department of Informations Systems

University of Hagen

1 Problem description

In this contribution, we examine a cooperation of four first-class producers in the food and beverages industry. Each of the companies is specialized in a well defined range of high-quality products which are complementary to the products offered by the other companies. A solid and free delivery of goods within a short time period is crucial in this business. In order to improve customer satisfaction and to generate logistics cost savings, the companies have decided to coordinate their distribution activities by inter-organisational transportation planning. In this way the producers jointly plan to establish a virtual full-range supplier while staying focused on their respective core product, thus guaranteeing high quality and full flexibility.

Some of the distributors use own vehicles for delivery. These vehicles are located either in the neighborhood of the company location or in distant regions with high demand. To supply the vehicles in the regions with goods for delivery, an additional transport by a freight company is needed (in the following denoted as long-distance transport). These long-distance transports start at the company location and finish at a transshipment location near the vehicle location. Orders that cannot be delivered using own means of transport are shipped through external food and beverages courier services. A schematic representation of the special situation the cooperation deals with is shown in figure 1.
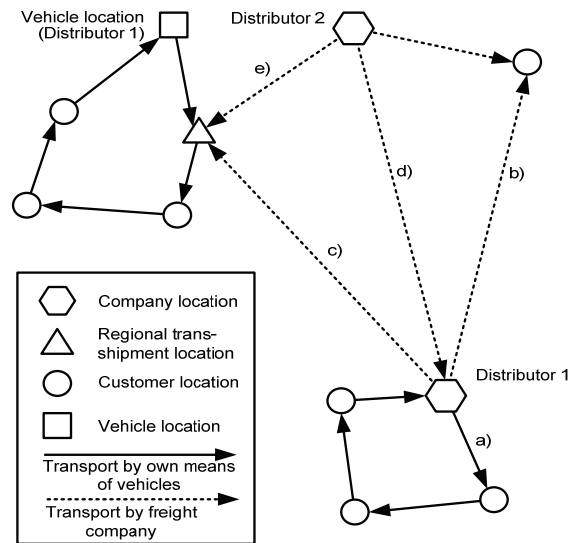
Figure 1: Schematic representation of the distribution structure using the example of 2 distributors.

The different possibilities for delivery are specified according to figure 1 as follows:

a) Distributor 1 delivers with own means of transport from the company location directly (without transshipment) to the customers.

b) Distributor 1 delivers via courier service from the company location directly to the customer.

c) Distributor 1 delivers indirectly, hiring a long-distance transport to a regional transshipment location from where own vehicles undertake last mile delivery.

d) and e) Distributor 2 (who does not own a fleet of vehicles) delivers indirectly, hiring a long-distance transport to the company location of the cooperating distributor 1 or to a regional transshipment location from where vehicles of distributor 1 undertake last mile delivery.

The task of finding a solution for the delivery of goods in the above described network under the given planning scenario can be well interpreted as a Rich Vehicle Routing Problem (RVRP). This RVRP combines widely discussed VRP restrictions like

- limited vehicle capacity (weight and volume),
- a heterogeneous fleet of vehicles,
- time windows for order delivery,
- backhauls of empty boxes and
- a given maximum driver's time

with the following non-standard problem extensions:

- a dynamic inflow of orders and
- the simultaneous planning of own-name transport and subcontracting

under the objective of minimizing total variable transportation costs, which consist of the direct costs of company-owned vehicles and all transportation charges paid to freight companies.

The motivation for the two non-standard requirements can be explained as follows:

Orders are not completely known in advance, but become available during the day, when physical distribution has already started. Moreover, binding advance notices have to be made to the courier companies at different but given times. Due to the concurrency of order inflow and order execution, the planning situation is continuously changing. Thus a dynamic way of planning is considered appropriate.

To decide which orders should be delivered by company-owned vehicles (possibly inducing an additional long-distance transport) and which orders should be delivered by a courier service is far from being trivial since the cost savings caused by a planning option for a given request will strictly depend on which other requests are assigned to the same means of transportation.

2. Basic Procedure based on Large Neighborhood Search

An algorithm that copes with the depicted RVRP has been developed extending the algorithm in [3]. This algorithm allows simultaneous planning of own-name transport and subcontracting, while taking into account heterogeneous time windows, varying vehicle capacities and a maximum drivers time. Additionally, the algorithm implements a comprehensive handling of backhauls restrictions. The algorithm was embedded in a dynamic planning framework that allows solving temporary static problems in a rolling horizon fashion.This approach comprises a Large Neighbourhood Search (LNS) [4] which is combined with a threshold accepting criterion [1] in the following manner:

The construction of a feasible solution uses a simple insertion heuristic, which first tries for each order to assign it to a company-owned vehicle; if applicable, also a long-distance transport has to be planned. If own-name transport is not possible, the order is assigned to a courier service. To improve this starting solution, a metaheuristic approach is used which is a variant of the LNS technique [4]: A certain proportion of orders is removed from the last solution and cost-effectively re-inserted in a randomly chosen sequence. To overcome local minima, a Threshold Accepting [1] procedure is used, which allows a temporary decline of solution quality. After a pre-specified time interval (e.g. 10 minutes), the optimization phase is stopped and the algorithm asks for new orders. At the same time, all parts of the currently best solution which represent irreversible decisions are fixed for all further calculations. Such irreversible decisions are, e.g., orders which have been irreversibly assigned to tours, tours which cannot be further loaded because they have already started, and courier transports which cannot be changed because they have already been bindingly announced. The new orders are inserted according to the rules of the insertion heuristic. Finally, the improvement heuristic is re-started using the updated solution as the best known solution.

Former tests showed that this LNS-procedure yielded cost savings up to 20% when comparing the cooperative scenario to the situation without any cooperation (isolated scenario).

3. GRASP method

The recent research activities primarily focussed on further improving and stabilizing the calculated results. Former results showed that a longer calculation time usually could indeed improve the test results but not to the desired extent – possibly due to the highly randomized (re)insertion procedure. Therefore, a Greedy Randomized Adaptive Search Procedure (GRASP) [2] was implemented.

The GRASP starts with assigning a weight to the orders that currently have to be included in the transportation plan. The weight reflects the importance of choosing a specific order for insertion and depends on the costs that would occur if the order would not be inserted first. The most important orders are accepted as elements of the so called Restricted Candidate List (RCL). One order out of the RCL is now chosen randomly for greedy insertion in the last feasible transportation plan, thus allowing to use the insertion heuristic explained in chapter 2. After having inserted this order it is removed out of the RCL. The weights of all remaining orders are updated and a new RCL is built for the next step. Once all orders are inserted, a neighborhood search starts. A modified version of the above described LNS procedure is used for this purpose. Unlike in the above described LNS, in the modified version the GRASP-insertion method is used for re-inserting and the threshold criterion is omitted.

At the moment, the GRASP is subject to intensive testing using randomly generated problem instances and a simulation engine. Preliminary tests have shown promising results. The behaviour of the two procedures in different situations will be compared to each other. Detailed results will be presented at the conference.

References

[1] G. Dueck and T. Scheuer, "Threshold Accepting: A general purpose optimization algorithm appearing superior to Simulated Annealing, Journal of Comp. Physics 90, 161 -175, 1990

[2] T. Feo and M. Resende, "Greedy Randomized Adaptive Search Procedures", Journal of Global Optimization 6, 109-133 (1995)

[3] A. Nagel, G. Pankratz, H. Gehring, "Tourenplanung in einer Distributionskooperation komplementärer Lieferanten". in Dangelmaier W., Blecken A., Rüngener N. (Hrsg.): *Nachhaltigkeit in flexiblen Produktions- und Liefernetzwerken*. Verlag Monsenstein und Vannerdat, Münster, 2009, S.281-292.

[4] P. Shaw: A new local search algorithm providing high quality solutions to vehicle routing problems, Technical report, APES group, Dept. of Computer Sciences, University of Strathclyde, Glasgow, 1997

Signal Optimisation Using the Cross Entropy Method

Dong Ngoduy

Institute for Transport Studies

University of Leeds, United Kingdom

Mike Maher

Institute for Transport Studies

University of Leeds, United Kingdom

Email: m.j.maher@its.leeds.ac.uk

Ronghui Liu

Institute for Transport Studies

University of Leeds, United Kingdom

1 Introduction

Traffic signals are a vital tool in the efficient and safe use of road space and control of traffic in congested urban networks. Consequently, a great deal of work has been carried out over many decades to develop techniques for determining the optimal timings for signals either at isolated intersections or in a coordinated manner in networks of junctions. In order to find the optimal timings, a traffic model is required that will predict the traffic flow pattern, delays and stops that would result from the implementation of any proposed set of timings, or control policy. Such traffic models come in a variety of forms, macroscopic and microscopic, deterministic or stochastic; some have purely numerical outputs whilst others provide graphical displays. Whilst responsive control has become increasingly prevalent, fixed-time control is still an important and widely-used form of control in many urban networks. Fixed time plans can be set up for the demand pattern expected at different times of day. The problem of finding the optimal timings, according to the predictions from the assumed form of traffic model, is usually far from straightforward except in the case of a single intersection, since there are typically an enormous number of feasible solutions and a very large number of local minima, and especially if re-routeing of drivers in response to signal timings is possible. The problem of finding the global minimum is a complex combinatorial optimisation problem for which many alternative approaches (including, for example, evolutionary algorithms, ant colony optimisation, simulated annealing) have been proposed and tried, but for which there is as yet no fully accepted method.

The research described here is concerned with the application and testing of a relatively new approach to such problems: the cross entropy method, proposed by Rubinstein [1]. The method, which has an appealingly simple structure and a sound theoretical underpinning, will be applied to a number of different forms of signal optimisation problem.

2 Problem formulation

We consider a network of signalised junctions, operating under fixed-time control. The set of signal timings to be implemented consists of the green times for the various stages at each junction, and the offsets between the junctions. A set of timings will be referred to as a *solution* and denoted by $\mathbf{x} = (x_1, x_2, \ldots x_m)$, a vector made up of m elements, in some appropriate form, representing the greens or stage start times at each junction, and the offsets, each expressed as an integer number of seconds. Typically the set X of all possible solutions will be very large.

The choice between alternative solutions will be made according to the traffic model adopted for any particular case. In the examples considered here, drivers are assumed to respond to any set of timings by re-routeing, and this is described through a traffic assignment model. Running the traffic model with a solution $\mathbf{x}$ leads to a set of outputs, from which the value of an appropriate *objective function* or *performance index* (such as the total network travel time) can be calculated. This will be denoted by $z(\mathbf{x})$. The aim is to determine the optimal solution $\mathbf{x}^*$ that minimises this objective function.

3 The cross entropy method

The cross entropy method (CEM) is an iterative process, at each stage of which a set of solutions is drawn randomly from a set of discrete probability distributions p_{ij} ($i = 1, \ldots m; j \in J_i$) in which J_i is the set of possible values for element i in the solution vector. Each of the N solutions generated is evaluated using an appropriate model and its objective function value z obtained. These N solutions are ranked and the best N^* identified (with N^* typically being 5% of N). In this "elite" sample, for each element i, we count up the number of cases r_{ij} in which it takes the jth value. From this information the parameters of the discrete probability distributions are then updated using a weighted average of the old p_{ij} and the observed r_{ij}/N^*. These are then used to generate the solutions in the next iteration.

The process starts with uniform distributions assumed for the p_{ij} : that is, for each element, all possible values are equally likely. Through the process of selection and updating of parameters, the quality of the solutions generated steadily improves, and the distributions steadily become more clustered around a small number of values until no further improvement occurs. The best solution found during the iterative process is the estimate of the global optimum. There is no guarantee, of course, that the global optimum will be found, since the process itself contains random elements.

4 Previous work

Previous work [2] by one of the authors has applied the CEM to the problem of optimisation of fixed-time signal timings on a six-arm signalised roundabout with the cell transmission model being used to describe

the cyclic flow of traffic, and the build-up and decay of queues, under the assumption of constant OD flows. In this case, the traffic model was deterministic and macroscopic and there was no route choice. For that problem, it was calculated that there were of the order of 10^{16} possible (nominally undersaturated) solutions. At each iteration $N = 2000$ solutions were generated, and it was found that the CEM worked very well, with steady improvement through the iterative process and effective convergence occurring in around 15 iterations. It seemed therefore to provide an efficient and appealing approach to such combinatorial optimisation problems.

4 New applications

The current work is to apply the CEM to signal optimisation problems in a network. It has long been recognised (see [3], [4] and [5] for example) that changing the signal timings in a network will generally cause some re-routeing of traffic, and that repeatedly optimising the signal timings assuming that the current flow pattern will remain fixed may lead to a steady deterioration in network performance (the resultant solution is referred to as the "mutually consistent" (MC) solution, because the flow pattern is a function of the timings and the timings are a function of the flows) . On the other hand, if this potential re-routeing effect is recognised and taken account of, it may be that signal control can be used beneficially to persuade drivers into a more preferable routeing pattern. Therefore the impact of a solution $\mathbf{x}$ will be described using a traffic assignment model for the given OD matrix, and the consequent total network performance measure $z(\mathbf{x})$ calculated from the assignment output. Previous work [6] has also tackled this type of problem, using a combination of a genetic algorithm (GA) for the optimisation, and the TRANSYT traffic model and logit-based SUE assignment to model the traffic flows and re-routeing. The work showed there were appreciable potential benefits from the use of this "farsighted" approach in contrast to the "shortsighted" MC solution approach. Our primary aim here is to apply the CEM for the optimisation of the signals in the network, using a (deterministic) UE assignment model to describe the route choices of drivers in response to any signal timings. The results from the CEM will be compared with those from a GA approach to the same problem, considering both the quality of the final solutions obtained and also the computational demands of the two approaches.

In the second application, assignment is carried out using a Monte Carlo, probit-based SUE model (sometimes referred to as the "Burrell method"). In this case, because of the Monte Carlo nature of the assignment, the outputs are subject to random error or noise. Ideally we want to choose the solution $\mathbf{x}$ that minimises the true, long-run average network performance measure $z_0(\mathbf{x})$ but what we observe is not $z_0(\mathbf{x})$ but a noisy version of it: $z(\mathbf{x}) = z_0(\mathbf{x}) + e$ where e is a random error, whose variance s^2 is a decreasing function of the number of MSA iterations used in the assignment. We *can* obtain accurate estimates, but only at the cost of long run times in the SUE assignment. If two trial solutions $\mathbf{x}$ and $\mathbf{y}$ are compared, and $z(\mathbf{x}) < z(\mathbf{y})$, it is not necessarily the case that $z_0(\mathbf{x}) < z_0(\mathbf{y})$. Hence, the ranking of a set of solutions on the basis of their z values is made less efficient or reliable the greater is the amount of noise. The more the noise dominates, the more the ranking becomes effectively random.

Previous work [7] by one of the authors has studied the general nature of this problem and proposed how the CEM should be modified to deal with such noisy combinatorial optimisation problem, including how the assignment run time should be chosen for the evaluation of $z(\mathbf{x})$ for solutions at each

stage of the iterative process, in order that the ranking of solutions generated in any iteration can still be efficient. The application there, though, was with artificial noise: the true value $z_0(\mathbf{x})$ was obtained from a deterministic traffic model and a random error e was generated and added to give $z(\mathbf{x})$ for any solution $\mathbf{x}$. This enabled the performance of the CEM to be evaluated in a controlled environment. In the current work, however, a *real* Monte Carlo assignment model is employed and the aim is to investigate the feasibility of this approach, and establish how best to estimate the value of s for any assignment run and the manner in which this is dependent upon the number of iterations, in order to see whether it is practicable and efficient to use such Monte Carlo assignment models to optimise signal settings in a network with re-routeing. The tests are carried out on a network similar to that used in [5] and [6].

References

[1] Rubinstein R.Y. (1999) The cross-entropy method for combinatorial optimisation. *Methodology and Computing in Applied Probability*, 2, 127-190.

[2] Maher M.J. (2008) The optimization of signal settings on a signalised roundabout using the cross-entropy method. *Computer-Aided Civil and Infrastructure Engineering* 23, 106-115.

[3] Maher M.J and Akçelik R. (1977) Route control: simulation experiments. *Transportation Research* 11, 25-31.

[4] Allsop, R.E., (1974). Some possibilities of using traffic control to influence trip distribution and route choice. In: *Transportation and Traffic Theory* (Proc. 6th International Symposium on Transportation and Traffic Theory). Elsevier, Amsterdam, 345–374.

[5] Allsop, R.E. and Charlesworth, J.A. (1977). Traffic in a signal-controlled road network: an example of different signal timings including different routings. *Traffic Engineering Control* 18 (5), 262–264.

[6] Ceylan H. and Bell M.G.H. (2005) Genetic algorithm solution for the stochastic equilibrium transportation networks under congestion. *Transportation Research B*, 39, 169-185.

[7] Maher M.J. (2007) Monte Carlo traffic simulation models: can we use them for optimisation? *Transportation Systems: Engineering & Management*. Proc. 12th International conference of the Hong Kong Society for Transportation Studies, 8 – 10 December, 2007, Hong Kong, 27 – 37.

New lower bounds and exact method for the m-PVRP

Sandra Ulrich NGUEVEU

LOSI

University of Technology of Troyes, 12 rue Marie Curie, 10000 Troyes, France

Email: ngueveus@utt.fr

Christian PRINS   **Roberto WOLFLER CALVO**

LOSI   LIPN-OCAD

University of Technology of Troyes   University Paris 13

1   Introduction

The m-Peripatetic Vehicle Routing Problem (m-PVRP) is defined on a complete undirected graph $G = (V, E)$ where $V = \{0, ..., n\}$ is the node set (node 0 is the depot and $V' = V \backslash \{0\}$) and E is the edge set. Each client $i \in V'$ has a demand d_i and Q is the capacity of vehicles. A cost c_e is assigned to each edge $e \in E$. The objective of the m-PVRP is to identify a set of edge-disjoint routes of minimal total cost over m periods so that each client is served exactly once per period.

The m-PVRP was introduced in [5]. So far its best known upper and lower bounding procedures are the b-matching and the hybrid tabu search of [6]. Applications include money collection, transfer and dispatch when it is subcontracted to specialized companies. For security reasons, peripatetic and capacity constraints ensure that no sequence of clients is repeated during the m periods and the amount of money allowed per vehicle is limited. The m-PVRP can be considered as a generalization of two well-known NP-hard problems: the VRP ($\approx$ 1-PVRP) and the m-Peripatetic Salesman Problem (m-PSP $\approx$ m-PVRP with a single vehicle).

2   Column Generation approach

The new lower bounding procedure described in this section consists in two dual heuristics H1 and H2 that identify good dual feasible solutions for the linear relaxation of the aggregated set partitioning formulation, following the approach used for example in [2] for the VRP. Let $\Re$ be the

set of feasible routes, c_r with $r \in \Re$ the cost of route r, $\Re_i$ the set of routes crossing node i, $\Re(e)$ the set of routes crossing edge e and y_r the binary variable equal to 1 if and only if route r is used. The aggregated set partitioning formulation (APF) of the m-PVRP is:

$$(\text{APF}) \quad \min \sum_{r \in \Re} c_r y_r \tag{1}$$

s. t.

$$\sum_{r \in \Re_i} y_r = m, \qquad \forall i \in V' \tag{2}$$

$$\sum_{r \in \Re} y_r \geq m \left\lceil \sum_{i \in V'} \frac{d_i}{Q} \right\rceil \tag{3}$$

$$\sum_{r \in \Re(e)} y_r \leq 1, \qquad \forall e \in E \tag{4}$$

$$y_r \in \{0, 1\}, \qquad r \in \Re \tag{5}$$

This formulation is not suitable for solvers because of its exponential number of variables. Its constraints can be dualized by associating respectively penalties $\lambda_i \in \mathbb{R}$ with $i \in V'$, $\lambda_0 \geq 0$ and $\mu_e \leq 0$ with $e \in E$. The linear relaxation of the resulting model (APF(λ, μ)) still requires an exponential number of variables, justifying the use of a column generation approach.

The first dual heuristic (H1) is based upon three key ideas: the approximation of routes with non-elementary routes called $\mathfrak{q}$-routes, the use of a column generation approach to handle the exponential number of variables and the use of dual ascent to estimate the best dual variables values. A route of total load q does not autorise cycles, whereas a $\mathfrak{q}$-route does and can be generated with a pseudo-polynomial algorithm. These cycles are then penalized via a Lagrangian relaxation, to obtain a heuristic fast and efficient. H1 does not require the use of Cplex$^{\circledR}$ or any other solver to solve each master-problem obtained after generating $\mathfrak{q}$-routes of negative reduced cost. The violations of degree constraints are quantified then used in a subgradient procedure to correct the values assigned to the dual variables, until the improvements become negligible or during a predefined number of iterations. This procedure, known as dual ascent, outputs the best dual variables values that will be used to generate new $\mathfrak{q}$-routes of negative reduced cost. If no new $\mathfrak{q}$-routes can be generated, then the dual solution found provides a valid lower bound for the m-PVRP. The dual ascent remedies the dual variables stability problems of classical column generation algorithms.

The second heuristic (H2) is applied after two consecutive iterations of H1 without generating any $\mathfrak{q}$-route of negative reduced cost. It uses procedures similar to H1's, but generates elementary routes instead of $\mathfrak{q}$-routes. Its dual variables must be initialized with H1 to reduce the computing time and memory required, since the generation of elementary routes of negative reduced cost is an NP-complete problem ([3]). H1 is also used to compute lower bounds required by the dominance properties applied within H2 to optimize the generation of elementary routes.

3 Branch-and-cut applied on the edge-based formulation

The branch-and-cut algorithm BC_{EF} described in this section is applied on the edge-based m-PVRP formulation (EF) where $\mathbb{K} = \{1, ..., m\}$ is the set of periods of the m-PVRP, $\delta(S)$ is the set of edges having one node in $S \subseteq V'$ and the other node outside. $r(S) = \left\lceil \sum_{i \in S} \frac{d_i}{Q} \right\rceil$ is a lower bound of the number of vehicles necessary to serve the total demand of S and x_e^k is the binary variable equal to 1 if and only if edge e is used within a route during period $k \in \mathbb{K}$.

$$(EF) \quad \min \sum_{k \in \mathbb{K}} \sum_{e \in E} c_e x_e^k \tag{6}$$

s. t.

$$\sum_{e \in \delta(\{i\})} x_e^k = 2, \qquad \forall k \in \mathbb{K}, \forall i \in V' \tag{7}$$

$$\sum_{e \in \delta(S)} x_e^k \geq 2r(S), \qquad \forall S \subseteq V', S \neq \varnothing, \forall k \in \mathbb{K} \tag{8}$$

$$\sum_{k \in \mathbb{K}} x_e^k \leq 1, \qquad \forall e \in E \tag{9}$$

$$x_e^k \in \{0, 1\}, \qquad e \in E, k \in \mathbb{K} \tag{10}$$

The branch-and-cut algorithm BC_{EF} starts from the linear relaxation of (EF) without the $O(m2^n)$ constraints (8). The valid inequalities sought-after result from the generalization of efficient VRP inequalities applicable on each of the m periods of the m-PVRP: capacity cuts, strengthened combs cuts and multistar cuts. It is important to note that not all VRP cuts are generalizable to the m-PVRP. For example, constraints based on a fixed number of vehicles can not be used for the m-PVRP because its number of routes is not limited and can vary from one period to another (e.g. generalized capacity constraints or hypotour constraints ([1])). The branching is performed on the variable whose value is the closest to 0.5.

4 Branch-and-cut for the aggregated formulation

The aggregated edge-based formulation (AEF) results from the aggregation over k of constraints (7) to (9), the introduction of binary variable $y_e = \sum_{k \in \mathbb{K}} x_e^k$ for each edge $e \in E$ and the deletion of the aggregated constraints (9) because of their redundancy with the aggregated constraints (10).

A solution of (AEF) is a subset of edges which satisfy aggregated degree constraints, but are not specifically assigned to any of the m periods. It may not correspond to a valid m-PVRP solution. In addition, it can be shown that even if this set of edges contained a feasible m-PVRP solution, partitioning the edges between the m periods to extract this solution would be NP-complete. In spite of that, (AEF) is easier and faster to handle than (EF) not only because of the reduced number of variables, but also because of the aggregated constraints which reduce the number of violated valid inequalities to identify.

The branch-and-cut algorithm BC_{AEF} starts from the linear relaxation of (AEF) amputated of the 2^n aggregated capacity constraints. Valid inequalities sought-after are the aggregated capacity cuts, the aggregated strengthened comb cuts and the aggregated multistar cuts. The branching is performed on the variable whose value is the closest to 0.5.

5   Implementation and Results

All algorithms were tested by adding $m \leq 7$ periods to classical instances from the VRP literature (type A, B, P and VRPNC of 19 to 200 nodes). The upper bounds needed were obtained from [6].

H1 is coded in C. The $\mathfrak{q}$-route generator of H1 is then replaced with an adaptation of the procedure GENROUTE (in Fortran) from [2] to obtain H2. Results show that H1+H2 complement the b-matching because, contrary to the latter, it performs faster and better for small values of m.

BC_{EF} and BC_{AEF} required a package of separation routines adapted from CVRPSEP [4]. Cplex$^{\circledR}$ was used to solve the linear problems identified after adding valid cuts. With a time limit of 3 hours, the results show an improvement of 5 to 10% of the ratios between best upper bounds and lower bounds known so far, and one third of the instances have been solved to optimality.

References

[1] P. Augerat. *Approche polyédrale du problème de tournées de véhicules*. PhD thesis, Institut National Polytechnique de Grenoble, France, 1995.

[2] R. Baldacci, N. Christofides, and A. Mingozzi. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming*, 115(2):351–385, 2008.

[3] M. R. Garey and D. S. Johnson. *Computers and Intractability : A Guide to the Theory of NP-Completeness* . W.H. Freeman and Company, New York, 1979.

[4] J. Lysgaard. A package of separation routines for the capacitated vehicle routing problem. http://www.hha.dk/∼lys/, 2003.

[5] S.U. Ngueveu, C. Prins, and R. Wolfler Calvo. Bornes supérieures et inférieures pour le problème de tournées de véhicules m-péripatétiques. In *Actes de la 7ème conférence internationale de Modélisation et Simulation*, vol. 3, pp. 1617–1625, Paris, France, 2008. MOSIM

[6] S.U. Ngueveu, C. Prins, and R. Wolfler Calvo. A hybrid tabu search for the m-peripatetic vehicle routing problem. In T. Stützle and S. Voss, editors, *Matheuristics: Hybridizing Metaheuristics and Mathematical Programming*, Annals of Information Systems Series, 10, pages 253–264. Springer, 2010.

Passenger Oriented Rolling Stock Rescheduling

Lars Kjær Nielsen

Rotterdam School of Management, Erasmus University Rotterdam

Leo Kroon

Rotterdam School of Management, Erasmus University Rotterdam

Gábor Maróti

Rotterdam School of Management, Erasmus University Rotterdam

P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

Email: gmaroti@rsm.nl

1 Introduction

Disruption management in the passenger railway context is the process of dealing with the effects of a disruption and getting back to normal operations afterwards. As part of the process, the rolling stock is rescheduled according to a disrupted timetable. Rolling stock here refers to all the vehicles that utilize the railway network. For early planning, seat demand is given as passenger estimates based on historical data for each timetable service; the stochasticity of the passenger estimates is dealt with by assigning slack capacity. However, passenger flows are dynamic in nature and are highly responsive to changes in the timetable. Such changes may be due to disruptions or delays, but also due to planned infrastructure maintenance. This study considers the rolling stock rescheduling process in the real-time disruption management phase in a railway system. We consider the situation where there is no seat reservation system and passengers occupy the seats on a first-come first-served basis.

The analogous problem in the airline context has been studied by [1]. There are a number of similarities between the airline and railway situations. In both cases vehicles with limited capacities are to be rescheduled, and there are passengers with origins and destinations who wish to use the available capacity. The major difference is that in the airline industry the operator can control the flow of passengers by assigning seats to the passengers; in the railway situation the passengers decide which trains to board as far as capacity is available.

In practice, the overall rescheduling process consists of three steps; (*i*) adapt the timetable to the change in environment, (*ii*) reschedule the rolling stock to serve the trains in the adapted

timetable, (*iii*) reschedule the crew accordingly. Although there is some interdependency between the steps they are still solved with only limited integration.

The problem we consider in this study comes from the major Dutch railway operator Nederlandse Spoorwegen (NS). Some of the most important features of the railway operations at NS are that the railway network is heavily utilized and there are often several possible traveling routes between each pair of stations. This, combined with the fact that there is no seat reservation system, means that passenger flows are hard to control.

The problems arising in the disruption management process are solved manually, often without taking the dynamics of passenger flows into account. Our aim is a model for rescheduling the rolling stock in such a way that sufficient capacity is available to the passengers. In our approach, we also integrate a few timetable related rescheduling options.

2 Problem description and model

We consider the problem of rescheduling the rolling stock to a modified timetable. The passengers are assumed to act as autonomous agents who try to get to their destination on their own using the available train services. As there is no centralized control of the passengers, it is part of the problem to predict the passenger behavior.

We assume that each passenger wants to travel as fast as possible from his origin at a given starting time to his destination. The passenger plans his journey based on the information that is available to him. The available information is the timetable, but not the capacity of each train. For simplicity, we aggregate passengers with identical intended journeys into passenger groups.

A passenger has a *planned arrival time* at the destination according to the trip planned in the regular timetable. The delay of a passenger is measured as the difference between the actual arrival time and the planned arrival time. Some passengers may not even reach their destination due to train cancellations and delays.

A rolling stock schedule is an assignment of the rolling stock to trains that respects limitations on train length, fleet size and possibilities for train length adaptations. The rolling stock schedule implies a pattern of *shunting operations* at the stations throughout the network. Shunting refers to the low level operations inside stations when train lengths are adjusted. Shunting is locally planned, so any changes to the rolling stock schedule must be communicated to the local dispatchers.

The rolling stock assignment decides the capacity of the trains in the network. Together, the timetable and the rolling stock assignment imply a passenger flow and thereby the passenger delays as well. We measure the quality of the rolling stock rescheduling by the following criteria: (1) number of changes to the shunting plans, (2) the sum of passenger delays, (3) number of passengers who do not reach their destination. In short, the intention is to balance the process and

efficiency oriented goals related to the operation of the rolling stock, and the service oriented goals related to passenger delays. Our model for passenger oriented rolling stock rescheduling attempts to reschedule the rolling stock in such a way that it maximizes the quality measure.

3   Solution approach

It is difficult to solve the model directly since the rolling stock decisions and passenger flows can not easily be integrated in one computationally tractable model. We therefore propose an iterative heuristic approach. In each iteration we reschedule the rolling stock according to the modified timetable. Then we simulate the passenger flow in the time expanded network capacitated by the current rolling stock schedule. The passenger flow is evaluated to determine delays and a feedback mechanism penalizes features of the rolling stock assignment that are likely to incur delays for the passengers. This iterative process continues until a stopping criterion is reached.

Rolling stock rescheduling

We use an extension of the Mixed Integer Linear Programming model by [2] to reschedule the rolling stock for the modified timetable. The basic model is well tested, and used by NS for medium-term planning on a daily basis. The extended model allows the decision maker to reschedule the rolling stock and to perform minor modifications to the timetable as well, such as deciding whether certain key connections are maintained. The objective function of the extended model is based on an estimate of how the decisions contribute to the mentioned quality measure of the schedule. The estimates come from a feedback mechanism from the passenger flow simulation. We note that a similar extension of [2] has been described by [3].

Simulation of passenger flows

The passengers choose their routes based on the shortest paths in the time expanded network, and greedily compete for seats in the trains. The passenger groups take up seats proportional to the size of the groups. Once a passenger enters a train, he keeps his seat until he exits the train. Therefore other passengers may not be able to enter a train due to lack of capacity; in that case they immediately re-compute their journeys and thus compete for capacity in the trains on their new route.

Feedback mechanism

The flow of passengers implies the passenger demand on each train. We can therefore evaluate which trains have too little capacity and thereby are likely to cause passenger delays. Assigning more capacity to those trains is encouraged through modifying the objective function in the next round of the rolling stock rescheduling process.

4   Input data

Our approach relies on a large amount of data; in the ideal case we would use real-time data on the traveling patterns of every single passenger to simulate the passenger flow in the time expanded network. Such detailed data is not available, however. We therefore use data from early planning to estimate the traveling patterns. In the near future, better information on passenger flows will be available through the implementation of an electronic traveling card system. This system directly retrieves the starting and ending station, and time of each journey in the network.

5   Computational tests and future work

We are in the process of applying the solution scheme to a number of instances based on real-life cases from NS. The instances concern the situation where a part of the infrastructure is blocked. We assume that the disrupted passenger demand is equal to the normal demand. Preliminary results indicate that it is possible to significantly improve the service aspect of the rolling stock assignment at the cost of more shunting operations.

The approach can also be used to investigate questions on how different levels of passenger control can improve the service perspective. Such measures include improved traveling information and seat reservation systems. They can be implemented in the approach by changing the assumptions on how passengers move in the time expanded network. Furthermore, the approach can be used in short-term planning as well when adapting the rolling stock schedules during planned maintenance projects. In this case more time is available to evaluate the different possibilities due to the planning horizon.

References

[1] S. Bratu and C. Barnhart. "Flight Operations Recovery: New Approaches Considering Passenger Recovery", *Journal of Scheduling* 9(3), 279-298, (2006).

[2] P.J. Fioole, L. Kroon, G. Maróti, and A. Schrijver. "A Rolling Stock Circulation Model for Combining and Splitting of Passenger Trains", *European Journal of Operational Research* 174, 1281-1297, (2006).

[3] L.K. Nielsen and G. Maróti. "Disruption Management of Rolling Stock in Passenger Railway Transportation", Technical Report ERS-2009-046-LIS, Rotterdam School of Management, Erasmus University Rotterdam, The Netherlands (2009).

Risk Approaches for Delivering Disaster Relief Supplies

Frédéric Semet

LAGIS

École Centrale de Lille

Karl F. Doerner

Department of Business Administration

University of Vienna

Pamela C. Nolz

Department of Business Administration

University of Vienna, Austria

Email: pamela.nolz@univie.ac.at

1 Problem Description

In this paper we focus on the aspect of risk on delivery tours for disaster relief supplies.

Especially in a post-disaster situation, which is characterized by a high grade of instability it is important for the affected people to be able to rely on the regular delivery of disaster relief supplies. This concerns not only the time of delivery but also the locations, where the critical items are put at the disposal of the people in need. Therefore it is crucial to plan the routes such that they remain accessible in case of aftershocks or augmenting water levels after inundations.

The contribution of this investigation is threefold. First of all, we develop and apply five approaches in order to evaluate the risk of delivery tours for disaster relief supplies to become impassable, regarding correlated as well as uncorrelated measures. The different risk approaches are included in a multi-objective Covering Tour Problem (CTP) as a third objective function in addition to the two objectives already investigated in Nolz et al. [2] - on the one hand a combination of the minisum facility location criterion and the maximal covering location criterion and on the other hand the minimization of travel time. Secondly, an extension of the Memetic Algorithm (MA) introduced in Nolz et al. [2] is developed. While a straightforward adaptation of the MA is applied to three of the five risk measures, the algorithm is extended by an enrichment phase for the remaining two risk approaches. At last, it is shown that solution quality can be improved by the developed solution method, contributing good compromise solutions for the potentially Pareto-optimal front.

2 Risk Approaches

Five approaches to measure the risk along the water distribution tours are introduced in the following.

Minimal Travel Time. The risk between two nodes of the street network means the probability that the path between the two nodes does not remain accessible. For any connection between two population centers, the total risk is determined as the maximal risk value of all arcs along that path. Taking the maximum is adequate, as the risk values along a path are correlated and should therefore not be summed up.

Considering the risk and travel time values of all possible paths between two population centers, the connection with the smallest travel time and the according risk value is chosen for the *minimal travel time* approach.

Minimal Risk. The second approach considers the smallest risk value and the according travel time among all possible connections between two population centers. The risk of a solution is determined as the maximum threat of all arcs included to become impassable. Between each pair of population centers the connection with the smallest risk value and the according travel time is considered.

Number of Alternative Paths. In contrast to the two approaches introduced so far, which consider the threat of a specific path to become impassable, the third approach takes the number of alternative paths between two nodes into consideration. This means that we do not measure the probability of a path not to be accessible any more, but how many connections could be used between two population centers. Paths between any two nodes are said to be alternative, if they differ at least in two arcs. We do not consider totally arc-disjoint paths, as in a post-disaster situation in a developing country it is more appropriate to avoid critical arcs, such as bridges, that can cause the failure of a whole path even if all the other arcs remain accessible.

Reachability. We include the specifications of the first three risk approaches into a global measure by considering not only the number of alternative paths between two nodes but also their threat of becoming impassable. We call this approach *reachability*, as it determines how vulnerable a population center elected as water distribution point is not to be reachable any more.

Reachability is calculated as the sum of the risk values of all possible connections between two nodes. Figure 1 shows four alternative Pareto-optimal paths between two nodes and their according *reachability*-values.

Risk in Category. The last approach models the risk based on the composition of a path between two nodes. Here we do not consider the maximal risk value, but the number of arcs that have a probability of becoming impassable greater than α. Therefore, the number of arcs belonging to a specific category of risk values $> \alpha$ is determined for each connection between two nodes.
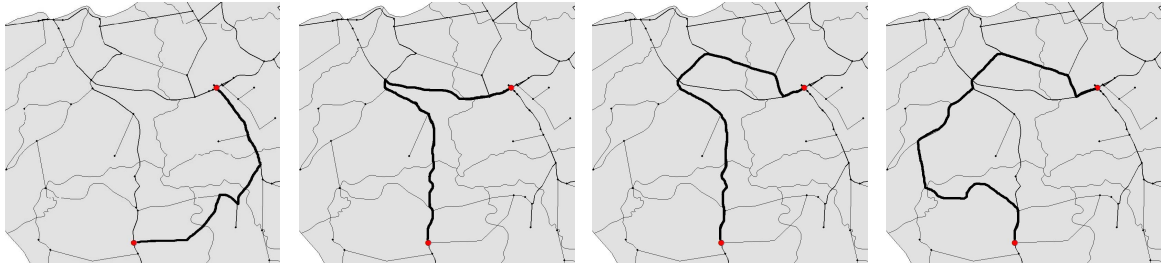
Figure 1: 4 alternative Pareto-optimal connections between two nodes (travel time, risk: 0.424, 0.97/ 0.515, 0.96/ 0.556, 0.87/ 0.7, 0.81)

3   Solution Procedure

An adapted version of the bi-objective MA introduced in Nolz et al. [2] is applied to three of the five risk measures, complemented with an additional objective. For the remaining two risk approaches the MA is extended by an enrichment phase, contributing good compromise solutions for the potentially Pareto-optimal front. The whole process performs as follows. In phase 1, a potentially Pareto-optimal front of the three-objective problem is determined by our MA. Location decisions for water tanks (population nodes visited within the CTP) and as a consequence the objective function value for minisum criterion and MCLC are fixed. In phase 2, by applying Martins' algorithm [1] paths between two nodes in a solution can be modified. Therefore, all possible combinations of paths between the nodes included in the potentially Pareto-optimal solutions are computed. Between each pair of nodes every alternative path is combined with each alternative path between all the other nodes. The solutions found in this way are evaluated considering only travel time and risk as objectives and dominated ones are eliminated. The remaining bi-objective Pareto-optimal solutions are regarded in detail in order to update the travel time and risk matrices used for the MA. The whole process is repeated until the original potentially Pareto-optimal front cannot be improved. With this procedure, it is possible to generate Pareto-optimal solutions with lower risk values that would not have been discovered otherwise. The enrichment phase is performed for the minimal travel time approach and risk in category. For the other approaches it is not useful, as for minimal risk this could only generate higher risk values. Number of paths and reachability are global measures already considering all existing travel time- and risk values between two nodes, where this procedure cannot be applied.

4   Computational Results

The different risk approaches are tested on a real-world instance from Manabí, Ecuador, where an earthquake with a magnitude of 6.5 in the epicenter occurred.

Considering the minimal travel time approach we were able to enrich the potentially Pareto-

optimal front by applying our two-phase approach. The risk values of the solutions contained in the initial potentially Pareto-optimal front range from 0.98 to 0.97. By updating the travel time- and risk matrices the risk values of the solutions proposed by the three-objective algorithm can be enriched, now ranging from 0.98 to 0.81. The potentially Pareto-optimal front for minimal risk contains only paths with a probability of 0.83 to 0.81 to become impassable.

In order to be able to compare the solutions of two Pareto set approximations regarding their multi-objective nature, we applied a performance measure introduced by Zitzler and Thiele [3]. The set of solutions generated with the minimal travel time approach and the according risk values outperforms the Pareto set approximation generated with the number of paths approach. While 33 % of solutions considering minimal travel time are dominated by or equal to the solutions considering the number of paths measure, 46 % of the latter are at least weakly dominated.

The reachability approach gives a set of solutions that is more secure evaluated in terms of risk associated with minimum travel time, than only considering risk values. About 60 % of solutions considering minimal travel times are (weakly) dominated, while only 40 % of the Pareto set approximation generated with the reachability approach are dominated by or equal to these solutions.

The risk in category measure is outperformed by the minimal travel time approach.

Concluding, it can be observed that the reachability approach provides the most global and efficient approach for measuring the risk on the paths of the water distribution tours. It is shown that the approximated Pareto set can be enriched by our two-phase approach.

References

[1] X. Gandibleux, F. Beugnies, S. Randriamasy, Martins' algorithm revisited for multi-objective shortest path problems with a MaxMin cost function, 4OR 4, pp. 47–59, 2006.

[2] P. Nolz, K. Doerner, W. Gutjahr, R. Hartl, A Bi-objective Metaheuristic for Disaster Relief Operation Planning, in C. A. Coello Coello, C. Dhaenens, L. Jourdan, editors, Advances in Multi-Objective Nature Inspired Computing, Springer, pp. 169–191, 2009.

[3] E. Zitzler, L. Thiele, Multiple objective evolutionary algorithms: a comparative case study and the strength of the pareto approach, IEEE Transactions on Evolutionary Computation 3(4), pp. 257–271, 1999.

A Stochastic and Dynamic Policy-Oriented Model of a Large Network of Airports

Nikolas Pyrgiotis

Department of Aeronautics and Astronautics

Massachusetts Institute of Technology

Corresponding Author: Amedeo Odoni

Department of Aeronautics and Astronautics and

Department of Civil and Environmental Engineering

Massachusetts Institute of Technology

Email: arodoni@mit.edu

1 Extended Abstract

As more airports become congested, it is increasingly common to observe network-wide delay propagation. For example, poor weather on any given day at a few critical airports in the United States or in Northwestern Europe often results in disruptions of airline operations in major portions of the entire air transport systems in these parts of the world. The mechanisms through which such nearly-chaotic conditions spread are complicated and the impacts they have on different types of users are hard to predict without the aid of advanced tools. This paper describes the Airport Network Delays (AND) model, which has been developed to study such complex phenomena, and presents the results and insights obtained through this model.

AND is concerned with computing delays at individual airports within a large network of major airports and, more important, with capturing the "ripple effects" that lead to the propagation of local delays to other airports in the network. The model is entirely analytical and is based on queuing theory. It therefore does not require multiple runs for some given set of input parameters, as is the case with stochastic simulation models. To increase computational efficiency, a network decomposition approach has been adopted. The model operates by iterating between its two main components: a stochastic and dynamic queuing engine (QE) that computes local delays at individual airports and a delay propagation algorithm (DPA) that updates flight schedules and demand rates at all the airports in the model in response to the local delays computed by the QE. The model's conceptual approach and a three-airport prototype implementation were originally described in Malone [1] and Malone and Odoni [2]. The ongoing research involves the complete re-design of the model, including a new DPA and data structure, programming in java, numerous improvements to the logic, large-scale network implementation (see below), and the addition of several important new features (see below). Related

references include [3], [4] and [5], all of which use a queuing theory approach, and [6], [7], and [8], that use simulation.

The QE treats each airport as a queuing system with non-stationary Poisson arrivals, k-th order Erlang service times, a single-server and infinite waiting room, denoted as a $M(t)/E_k(t)/1$ system in queuing theory. Because of the flexibility of the Erlang family of probability density functions, one can approximate with this model queuing systems with "general" ("$G(t)$") distributions of service times, such as those encountered at airports. This is done by selecting the value of k that most closely matches the coefficient of variation of the service times, as measured through field data. Values of k in the range of 3 to 12 are typical, depending on the mix of aircraft types and the type of operations (landings, take-offs, or mixed) on each airport's runways.

In order to approximate an infinite capacity system, the number of states, N, of the queuing system must be chosen large enough so that $P_{kN}(t)$, the probability that the system is full at any time t, is very small. For large values of k (i.e., when the service times have a small coefficient of variation or, in practical terms, are nearly constant) the number of states of the system, $kN+1$, can then become very large and the numerical solution of the system's equations time-consuming. For these reasons, the QE uses a very fast and accurate approximation scheme (Kivestu [9] and Malone [1]) that solves a system of $N+1$ difference equations (independent of k), instead of the system of $kN+1$ first-order, differential equations required for the exact solution of $M(t)/E_k(t)/1$ systems.

The Delay Propagation Algorithm (DPA) takes advantage of the fact that airline schedules assign an itinerary to each aircraft in an airline's fleet, i.e., each aircraft must execute on any given day a sequence of flight legs through the network of airports according to a planned set of departure and arrival times. Airline schedules include some "slack", both in the planned gate-to-gate time-lengths of flights and in the "turnaround times" on the ground between consecutive flights of any given aircraft. When these scheduled slacks are insufficient to absorb any long delays that may occur on a particular day, then delay propagates, e.g., from a late departure of an aircraft from airport A to a late arrival at the next airport B and from a late arrival at B to a late subsequent departure of that aircraft from B – the latter leading to the dreaded announcement that "there will be a delay due to a late-arriving aircraft". The DPA algorithm uses highly-efficient data structures and state-updating approaches to capture this delay propagation effect on a network-wide scale. It also keeps track of how much of the total delay incurred by any aircraft arriving at or departing from any airport X is due to (i) "local" delay, because of congestion at X, and (ii) "upstream" delay (or "reactionary" delay, in European terminology) because of delays at airports that this aircraft has visited prior to X.

The model has been fully implemented for a network consisting of the 34 busiest airports in the continental United States. These airports collectively handled 1.12 billion passengers in 2007 or 73% of the total. A large database has been assembled for this purpose with information available through NASA and the US Federal Aviation Administration. The database consists of: detailed demand data for each airport, including demand for general aviation and other non-scheduled flights; airport capacities under different weather conditions for each airport; and, most important, the

scheduled itineraries (i.e., the sequence of flight legs to be flown, along with scheduled arrival and departure times) for the entire fleet of aircraft of each of the twelve major carriers in the United States. As noted already, this last set of data is necessary for estimating through the DPA how delays at any airport will spread to others. As of March 2010, a similar implementation for the network of the 34 busiest airports in Europe is in progress, using data supplied by Eurocontrol.

The AND model is fast computationally, requiring between 2 and 5 minutes, depending on the level of congestion in the network, on typical laptops to run through 24 hours of operations at the 34 airports and compute delay-related statistics for every landing and take-off of each individual aircraft. (By comparison, the state-of-the-art simulation model [8] currently in use by NASA and the FAA requires several hours for a single run, due to its highly-detailed representation of the national air transportation network.) AND thus makes possible the exploration, at a macroscopic level, of the implications of a large number of policy alternatives and future scenarios on system-wide delays and associated costs. Issues that have been investigated to date include: the relative importance of specific airports as potential generators of delays that may disrupt the entire national system; and the system-wide effect on delays of changes in airport capacity resulting from some local interventions, e.g., a new runway at a specific airport. Other expected future applications include analyses of the effects of: national programs designed to increase air traffic predictability (e.g., 4-D trajectories); innovative air traffic flow management approaches; alternative configurations of some airline networks (e.g., configurations that may de-emphasize hub-and-spoke operations);.and airline scheduling strategies that reduce (or increase) the amount of time aircraft spend on the ground at airports between arrival and departure, or the planned "block" times in flight schedules.

The model has already provided new insights into the complex interactions through which delays propagate across airports and into the often-counterintuitive consequences of these interactions. For example, our results demonstrate how the propagation of delay modifies the original schedule of daily demand profiles at individual airports by pushing more landings and takeoffs into the late evening hours. Due to the resulting "smoothing" of the peaks and valleys of demand, delays at the local level are smaller than they would have been with the original demand profiles. An unexpected consequence of this phenomenon is that certain flights and airlines may benefit from the smoothing of demand profiles, by experiencing smaller delays than they would have in the absence of delay propagation. At the same time, other airlines suffer a disproportionately high cost. Airlines that emphasize hub-and-spoke operations at habitually congested hubs are particularly likely to fall into this second category. To immunize themselves from sharing disproportionately in the costs of delays, these airlines must typically position a significant number of "spare" aircraft at the congested hubs – a measure that also carries a heavy cost, as such spare aircraft are typically underutilized.

Some enhancements of the model have been implemented and are described in the paper. These features are optional and can be activated at the users' behest. The three most important are:

(a) *Alternative Queuing Engine*: While the primary stochastic and dynamic queuing engine In AND treats each airport as a $M(t)/E_k(t)/1$ queuing system, a deterministic and dynamic model that

treats each airport as a $D(t)/D(t)/1$ queuing system has also been implemented. By treating the demand and service processes as dynamic deterministic ones, this alternative QE can provide a lower bound on the delays to be expected in a system of airports.

(b) *Airline Recovery Optimization Model:* The AND model in its basic configuration treats the airlines as "passive" participants, i.e., does not capture the tactical actions that airlines take on a daily basis to mitigate in "real time" the impact of delays, as they occur, on their schedules. A user option incorporated into AND, provides a model that attempts to replicate an airline's reaction to long delays at one or more airports in the network. This model optimizes airline schedule "recovery" from delays [10] through the use of a combination of flight cancelations, aircraft re-routing and adjusting the departure times of flights.

(c) *Ground Delay Programs*: An option of initiating ground delay programs (GDPs) when delays at one or more airports exceed a specified threshold has been added to the AND model. This option replicates, within the logic of the delay propagation algorithm (DPA) of AND, the traffic flow management practice of delaying aircraft on the ground before take-off to prevent large airborne delays, when a flight is headed to a congested airport. The option permits observation of the "side effects" of GDPs, such as the added surface congestion and additional departure delays at the airports whence ground-delayed flights originate.

References

[1] K. Malone, Dynamic Queuing Systems: Behavior and Approximations for Individual Queues and Networks, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, MA (1995).

[2] K. Malone and A. Odoni, A Queuing Model of a Network of Airports, Working Paper, Operations Research Center, Massachusetts Institute of Technology, Cambridge, MA (1998).

[3] M. D. Peterson, D.J. Bertsimas and A.R. Odoni, Models and Algorithms for Queueing Congestion at Airports, *Management Science* 41, 1279-1295 (1995).

[4] D. Long, D. Lee, J. Johnson, E. Gaier and P. Kostiuk, Modeling Air Traffic Management Technologies with a Queuing Network Model of the National Airspace System, Technical Report NASA/CR-1999-208988, NASA Langley Research Center, Hampton, VA (1999).

[5] M. D. Tandale, P. Sengupta and P.K. Mennon, Queuing Network Models of the National Airspace System, Working Paper, Optimal Synthesis Inc., Los Altos, CA (2009).

[6] F. Wieland, Limits to Growth: Results from the Detailed Policy Assessment Tool, *AIAA/IEEE Digital Avionics Systems Conference – Proceedings* 2, 9.2.1 – 9.2.8 (1997).

[7] K. D. Bilimaria, B. Sridhar, G.B. Chatterji, G. Sheth and S. Grabbe, FACET: Future ATM Concepts Evaluation Tool, Proceedings of the 3rd USA/Europe Air traffic Management Seminar, Naples, Italy (2000).

[8] Raytheon ATMSDI Team, Airspace Concept Evaluation System: Build 2 Software User Manual, NASA Ames Research Center, Moffett Field, CA (2003).

[9] P. Kivestu, Alternative Methods of Investigating the Time-Dependent M/G/k Queue, S.M. thesis, Massachusetts Institute of Technology, Cambridge, MA (1976).

[10] M. O. Ball, C. Barnhart, G. Nemhauser and A. Odoni, Air Transportation: Irregular Operations and Control, in *Handbook in Operations Research and Management Science* 14, C. Barnhart and G. Laporte (eds.), Elsevier, 1-73 (2007).

A simulation-based optimization approach to perform urban traffic control

Carolina Osorio

Transport and Mobility Laboratory

Ecole Polytechnique Fédérale de Lausanne

Email: carolina.osoriopizano@epfl.ch

Michel Bierlaire

Transport and Mobility Laboratory

Ecole Polytechnique Fédérale de Lausanne

1   Introduction

Since microscopic simulation tools can provide accurate network performance estimates in the context of scenario-based analysis or sensitivity analysis, they are often used to evaluate traffic management schemes. Nevertheless, using them to derive optimal traffic management schemes is a difficult task.

An optimal traffic management scheme can be formulated as:

$$\min_{x,z \in \Omega} E[f(x, z, p, \varepsilon)], \tag{1}$$

where the objective is to minimize the expected value of a suitable network performance measure f. This performance measure is a function of a decision or control vector x, endogenous variables z, exogenous parameters p and a random component ε. The feasible space Ω consists of a set of constraints that link x to z, p and f.

For instance, a traffic signal control problem can take f as the travel time and x as the green splits for the signalized lanes. Elements such as the total demand or the network topology will be captured by p, while the capacities of the signalized lanes will be captured by z. The random component ε describes the noise associated with a given realization of f.

In order to identify optimal schemes, these models need to be integrated within an optimization framework. This is intricate for several reasons: the detailed underlying models lead to noisy non-linear performance measures with no closed form available, their evaluation is also computationally expensive, not to mention the cost of evaluating derivative information.

Given the complexity of performing simulation-based optimization, a common approach is to construct a simplified model of the simulation model. This lower fidelity model is referred to as a surrogate or a metamodel. It is less realistic but is also typically less expensive to evaluate.

Metamodels are classified in the literature as either physical or functional metamodels [6]. Physical metamodels consist of application-specific metamodels, whose functional form and parameters have a physical or structural interpretation.

Functional metamodels are generic (i.e. general-purpose) functions (e.g. polynomials, splines), that are chosen based on their analytical tractability. The optimization algorithms that are based on these models, take advantage of their mathematical properties to ensure global convergence. These general-purpose metamodels can be used to approximate any objective function, but capture little information about the structure of the underlying problem. Furthermore, they require a large initial sample to be fitted, and are thus inappropriate for applications with a tight computational budget.

We believe that in order to perform simulation-optimization given a tight computational budget, the metamodel should combine both a structural and a functional component. In this paper we propose such a metamodel. This metamodel is integrated within a derivative-free trust region algorithm. The framework is used to solve a fixed-time signal control problem for a subnetwork of the Lausanne city center.

2 Metamodel

This section presents the main components of the proposed metamodel.

Simulation model. We use a calibrated microscopic traffic simulation model of the Lausanne city center. A detailed description of this model is given in [3]. For a given decision vector x the simulator provides a realization of a random variable $f(x, z, p, \varepsilon)$ (presented in Equation (1)).

Analytical queueing model. This model resorts to *finite capacity queueing theory* to capture the key traffic dynamics and the underlying network structure, e.g. how upstream and downstream queues interact, how this interaction is linked to network congestion. The model consists of a system of nonlinear equations. It is formulated based on a set of exogenous parameters θ that capture the network topology, the total demand, as well as the turning probabilities. A set of endogenous variables y describe the traffic dynamics, e.g. spillback probabilities, average rates at which a spillback diffuses. For a given decision vector x the network model yields the objective function $T(x, y, \theta)$.

A detailed description of the queueing model and a case study illustrating how the endogenous variables describe the formation and diffusion of congestion is given in [5]. Its formulation

for an urban road network appears in [4], where it has been successfully used to solve a fixed-time traffic signal control problem.

We now describe how f and T are combined to derive the metamodel m. The functional form of m is:

$$m(x, y; \alpha, \beta, \theta) = \alpha T(x, y; \theta) + \phi(x; \beta),$$

where ϕ is a quadratic polynomial in x, and α and β are parameters of the metamodel. The polynomial ϕ is quadratic with diagonal second derivative matrix.

At each major iteration of the trust region algorithm, the parameters β and α of the metamodel are fitted using the available sample by solving a weighted least squares problem, where the weights capture the importance of each point with regards to the current iterate.

We integrate the metamodel within the derivative-free trust region algorithm proposed in [2]. It is formulated for unconstrained problems. We extend its use to constrained problems as suggested in [1].

3 Traffic signal control

We illustrate the use of this framework with a signal control problem for a subnetwork of the city of Lausanne. We consider a fixed-time signal control problem, where the objective is to minimize the expected travel time.

We consider as initial point a uniformly drawn random signal plan. We allow for 1000 simulation runs. The performance of this metamodel is compared to that of a quadratic polynomial with diagonal second derivative matrix, (i.e. the metamodel consists of ϕ). After 1000 simulation runs, each one of these two methods derives an 'optimal' signal plan. The performance of these signal plans is then evaluated by running 50 replications of the simulation model.

Figure 1 presents the cumulative distribution functions of the average travel times across the 50 replications, considering the initial random plan (x_0), and the 'optimal' plans derived by the proposed metamodel (m) and the polynomial (ϕ). It also presents the performance of an existing plan for the city of Lausanne, which is denoted as *base plan* [3].

This figure illustrates how the proposed metamodel yields improved performance in terms of travel times, when compared to the initial plan, the plan proposed by the polynomial and also the base plan. This shows how the structural information provided analytically by the queueing model allows for improvement given a tight computational budget.

With no initial sample, and a tight computational budget, our method is able to identify signal plans that improve the distribution of the average travel time. This framework is therefore an attractive approach for derivative-free applications with a limited computation budget, where short-term performance is of main interest.
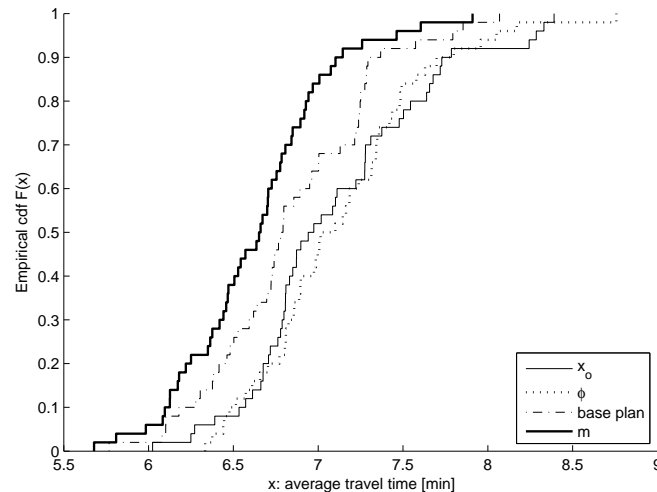
Figure 1: Empirical cumulative distribution functions of the average travel times starting from a random initial plan and allowing for 1000 simulation evaluations.

References

[1] A. R. Conn, K. Scheinberg, and P. L. Toint. A derivative free optimization algorithm in practice. In *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, St. Louis, MO, USA, September 2-4 1998.

[2] A. R. Conn, K. Scheinberg, and L. N. Vicente. Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points. *SIAM Journal on Optimization*, 20(1):387–415, 2009.

[3] A. G. Dumont and E. Bert. Simulation de l'agglomération Lausannoise SIMLO. Technical report, Laboratoire des voies de circulation, ENAC, Ecole Polytechnique Fédérale de Lausanne, Mai 2006.

[4] C. Osorio and M. Bierlaire. Network performance optimization using a queueing model. In *Proceedings of the European Transport Conference (ETC)*, Noordwijkerhout, The Netherlands, October 6-8 2008.

[5] C. Osorio and M. Bierlaire. An analytic finite capacity queueing network model capturing the propagation of congestion and blocking. *European Journal Of Operational Research*, 196(3):996–1007, 2009.

[6] D. B. Serafini. *A Framework for Managing Models in Nonlinear Optimization of Computationally Expensive Functions*. PhD thesis, Rice University, 1998.

Calibration of structural surrogate models for simulation-based optimization

Carolina Osorio, Gunnar Flötteröd, Michel Bierlaire

Transport and Mobility Laboratory (TRANSP-OR)

Ecole Polytechnique Fédérale de Lausanne (EPFL)

Email of corresponding author: gunnar.floetteroed@epfl.ch

Traffic microsimulations have become popular tools for the evaluation of transportation management and control schemes. Their major advantage over analytical models is that they explicitly simulate the individual entities of the transportation system (vehicles, traffic lights, pedestrians, ...). However, this very property also limits their tractability: Microsimulations constitute nonlinear and noisy mappings of their input parameters on their outputs. It is therefore desirable to combine the expressive power of microsimulations and the mathematical convenience of analytical models. The objective of this work is the further development of an existing methodology for the optimization of urban signaling plans within a simulation/optimization (SO) framework [6] that combines a detailed traffic microsimulator with an analytical queueing model of traffic flow [5].

In the SO framework, the purpose of the analytical model is to act as a surrogate for the microsimulation in the optimization, leading to smooth and deterministic objective functions and constraints. The SO framework is based on a derivative-free trust region method [2]. However, since the analytical model is only an aggregate representation of the microsimulation, some means to compensate for the deviation between simulation and analytical model are desirable. In previous work, essentially a quadratic regression model was calibrated from a sample of simulated observations at each major iteration of the optimization procedure to represent the deviation between the analytical model and the simulation. The use of a quadratic polynomial and an appropriate sampling strategy leads to a globally convergent method (i.e., starting from any initial point, it will lead to a local stationary point) [2].

However, it also comes with drawbacks:

- A polynomial regression model captures little to no structural information about the problem under consideration.

- The coefficients of the regression model require a large number of simulations to become

statistically significant.

The new approach presented in this work is that we adjust the analytical queueing model itself to the microsimulation. This resolves the aforementioned issues in the following way:

- The analytical queueing model is based on structurally meaninful equations such that the parameters that guide these equations are readily interpretable and likely to be significant.

- The analytical queueing model can be initialized with plausible parameters before the optimization is started and hence provides relevant information from the very first iteration.

Discarding the supplementary regression model also removes the convergence guarantees it brings along. For now, we concentrate on gaining a deeper understanding of how a direct calibration of the queueing model helps to solve the optimization problem. It still is feasible to again add a possibly further simplified regression model if that is desirable.

Apart from its operational relevance, the new approach has an interesting methodological facet. A calibration of the queueing model against the simulation is itself an optimization problem in that some distance measure between the simulator's output and the queueing model's output is minimized. Since the signal plans are optimized subject to a given parametrization of the queueing model and the queueing model is in turn calibrated from simulation responses that depend on the signal plans, we are dealing with two possibly coupled optimization problems.

The difficulty of the combined optimization/calibration problem depends to a large extent on the calibrated parameters. If they represent structural model properties that are independent of the controls then both problems are decoupled and the signals can be optimized based on an a priori calibrated queueing model. For example, this applies to all parameters that reflect the network geometry. If, however, the parameters depend on the controls, then the problem becomes one of bilevel programming in that the upper-level problem (optimization) needs to account for the effect a change in the signaling has on the parameters of the queueing model through the lower-level problem (calibration).

The SO framework developed so far relies on an analytical queueing model that is calibrated once before the optimization starts. Hence, all parameters that are independent of the controls can be assumed to be set to reasonable values. We therefore focus on those parameters that are fixed in the original analytical queueing model but actually should reflect the microsimulation's response to variations in the signal plans: traffic flow turning fractions at intersections and demand levels at the boundaries of the analysis zone.

The bilevel problem in its entirety is difficult, in particular in combination with a stochastic microsimulator. We therefore begin by experimenting with simpler and operational techniques borrowed from the field of origin-destination (OD) matrix estimation. The OD matrix estimation problem for congested networks constitutes a bilevel problem that is strongly related to our control

problem, and it has received substantial attention in the literature, e.g., [3, 4, 7]. However, the full bilevel problem specification clearly provides a multitude of further research opportunities.

Preliminary Results

We consider a fixed-time traffic signal control problem, where the objective is to minimize the expected travel time. A subnetwork of the Lausanne city center during the evening peak perior (17h-19h) is considered. The calibration of the analytical queueing model is constrained to the turning proportions, the realizations of which can be directly obtained from the microsimulation.

Denote by x^k the signal control plan in iteration k of the iterative optimization procedure, by θ^k the parameters (turning fractions) of the queueing model as calibrated in the same iteration, and by $T(x, \theta)$ the analytical objective function (total travel time), which evaluates a control plan x based on the queueing model only. Furthermore, let Θ^k be the simulated counterpieces of θ^k, i.e., the simulator's response in terms of turning fractions. We implemented a first bilevel heuristic, which we describe in the following together with a preliminary result.

Algorithm 1 Bilevel heuristic

---

1. set k to zero and initialize x^0 and θ^0

2. repeat until stabilization

   (a) feed control plan x^k into the simulation and obtain Θ^k

   (b) increase k by one

   (c) update $\theta^k = \frac{1}{k}\Theta^{k-1} + (1 - \frac{1}{k})\theta^{k-1}$

   (d) calculate new control plan $x^k = \arg\min_x T(x, \theta^k)$

The algorithm is started with a random initial signal plan and uniform turning proportions. In every iteration, it feeds the current signal plan into the simulation, uses the simulated turning fractions to update the turning fractions in the queueing model by the method of successive averages (MSA), [1] and re-optimizes the signal plan based on the updated queueing model.

The performance of the final signal plan is evaluated by running 50 replications of the simulation model. Figure 1 presents the empirical cumulative distribution of the average travel time in the simulation. For comparison, it also shows the distribution of the initial plan, which is evaluated in the same manner as the final plan.

A clear improvement can be observed. This shows that a calibration of the queueing model within the simulation loop is a meaningful approach. The most recent advances of this ongoing

---

[1] The use of MSA is for illustrative purposes; there are more efficient methods at hand [1].
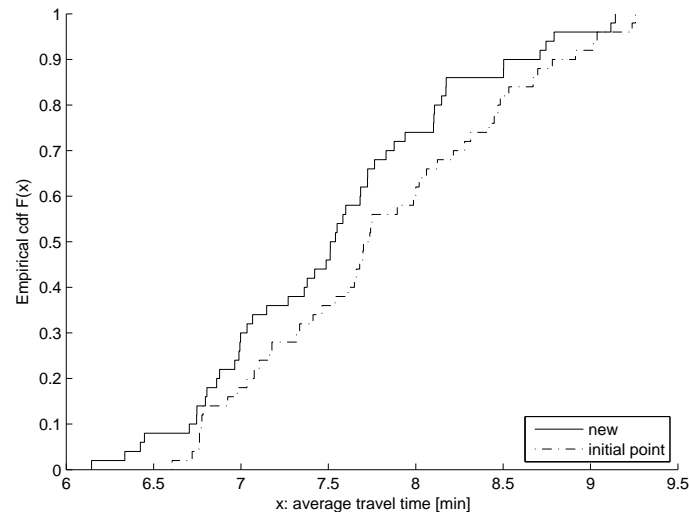
Figure 1: Empirical cumulative distribution functions of the average travel times for a random initial signal plan and the optimized signal plan.

project will be presented at the conference.

References

[1] M. Bierlaire and F. Crittin. Solving noisy large scale fixed point problems and systems of nonlinear equations. *Transportation Science*, 40(1):44–63, 2006.

[2] A. R. Conn, K. Scheinberg, and L. N. Vicente. Global convergence of general derivative-free trust-region algorithms to first- and second-order critical points. *SIAM Journal on Optimization*, 20(1):387–415, 2009.

[3] C. Fisk. Game theory and transportation systems modelling. *Transportation Research Part B*, 18(4/5):301–313, 1984.

[4] J. Lundgren and A. Peterson. A heuristic for the bilevel origin-destination-matrix estimation problem. *Transportation Research Part B*, 42(4):339–354, 2008.

[5] C. Osorio and M. Bierlaire. An analytic finite capacity queuing network model capturing the propagation of congestion and blocking. *European Journal of Operational Research*, 196(3):996–1007, 2009.

[6] C. Osorio and M. Bierlaire. A multi-model algorithm for the optimization of congested networks. In *Proceedings of the European Transport Conference*, Noordwijkerhout, Netherlands, 2009.

[7] H. Yang, T. Sasaki, and Y. Iida. Estimation of origin-destination matrices from link traffic counts on congested networks. *Transportation Research Part B*, 26(6):417–434, 1992.

A Routing Problem for the Science-on-Wheels Project

Emrah ÖZDEMİR

Roketsan Roket Sanayii

Ankara-Samsun Karayolu 40. Km Elmadag, 06780 Ankara

Haldun SÜRAL

Department of Industrial Engineering

Middle East Technical University, 06531 Ankara

Emrah ÖZDEMİR

Roketsan Roket Sanayii

Ankara-Samsun Karayolu 40. Km Elmadag, 06780 Ankara

Email: eozdemir@roketsan.com.tr

1 Introduction

In this study, we introduce a new selective and time windows constrained routing problem, called the science-on-wheels routing problem (SWRP), which is motivated by a project of ILKYAR, a non-governmental organization. The project is based on on-site activities performed in the selected junior boarding schools and is applied once a year. The SWRP plays an important role for the success of the project and involves two main decisions partitioned into two levels. In the first level, a set of schools is selected according to a given criterion subject to the special time windows constraints in a given project period. In the second level, a selected school is assigned to a project day. The problem is modeled using bi-level programming methods. The solution approach is implemented on the test instances that are compiled from the real-life data. Computational results show that our approach generates good solutions in short times.

2 ILKYAR's the Science-on-Wheels Project

ILKYAR, a non-governmental organization focusing on educational programs, develops and organizes several supportive programs for the students in the rural areas enrolled in junior boarding schools

(BSs) [1]. In the last decade, ILKYAR visited more than 250 BSs and these visits were organized as a part of the science-on-wheels projects.

In a two-week science-on-wheels project, a number of BSs are visited in a selected region. In each school, the project team presents an on-site activity based program, spends all day together with the students, stays in the school for a night, and leaves the school early in the next morning for the next BS chosen. The types of activities are vast. The region is chosen in a higher decision making level and not the part of the SWRP. Since the project period is narrow, it is not possible to visit all the schools in the selected region during a project. The SWRP involves two main decisions partitioned into two levels. In the first level, a set of schools is selected among the candidate schools in the pre-specified region in order to maximize the number of students. In the second level, a selected school is assigned to a project day.

Because of widespread geographic distribution of the BS sites in rural areas, it may require traveling over relatively long distances between two successive school sites during a project. Therefore, there is an upper limit on the travel length between two successive visits, which is called the major limit. In returning back to home at the end of the project, the travel length between the last school site and home site should not exceed a particular value, which is called the minor limit. Minimizing the total tour length is not the main concern of the project, but the total tour length should not exceed a given threshold value because of side constraints. These limits are made of the special time windows constraints for the SWRP. In summary, the SWRP is mainly to identify the schools that will be visited during the project and to decide the order of BSs to visit in a given project period. The SWRP is quite different than the classical routing problems with time windows in the literature and the traveling salesman problem (TSP) [2].

3 A Bi-level Model for the SWRP

For ILKYAR, it is very important to reach as many students as possible in a project, but minimizing the route length is very secondary. However, if there is a better way of routing the project without sacrificing the main objective, one should pursue the better route (since minimizing the travel time would maximize the time spent in the school sites). Since the first objective is much more important than the second one for ILKYAR, the SWRP is formulated as a bi-level model.

In a bi-level mathematical programming, there are two optimization problems. The first problem is called the upper-level (or leader) problem, whereas the second problem is called the lower-level (or follower) problem. The lower-level problem is optimized under a feasible region that is defined by the upper-level problem [3].

In the bi-level SWRP model, the first level is to select BSs in a given region that maximize the total number of students while satisfying the feasibility of the time windows constraints, whereas the second level is to find an optimal route of the selected schools. In the first level of the bi-level

SWRP, in order to deal with a feasible route, it is needed to develop a valid sequence of selected schools so that the time windows constraints can be satisfied.

When the SWRP and the bi-level formulation of the VRP proposed by Marinakis, Migdalas and Pardalos [4] are compared, both formulations have similarities since they make the assignments (selections) first and then they find the route for these assignments (selections). However, development of the lower-level problem is done differently than [4] because we apply strict hierarchy between the two objectives in the levels. A verbal bi-level model for the SWRP is given below.

Verbal Bi-level Model for the SWRP:

      (leader)   *maximize the total number of students*

           s.t.

           selection of the BSs in a given region,
           developing a sequence of the selected BSs to construct a feasible route
           satisfying the time windows constraints,

      where

       (follower)   *minimize the total route length for the selected BSs*

           s.t.

           TSP related constraints.

We solve the leader problem of the SWRP first and then solve a TSP for the BSs that are selected in the leader problem of the SWRP. The leader problem is formulated as 0-1 programming model whereas the follower problem is formulated as a TSP model.

4 Experimental Results and Conclusion

In our experimental analysis for the assessment of the solution quality of our approach and the solution effort, all computations are carried out on the test problems derived from the last nine years data of the ILKYAR's science-on-wheels projects. Actually, this is the entire available data for a two-week project because it is implemented only once a year since 2000.

We coded our models using GAMS IDE 23.1 and used CPLEX 11.2. For the lower-level problems, CONCORDE [5] which is a powerful TSP solver developed by Georgia Institute of Technology is called from a C code that converts the original distance matrix to a symmetric one and modifies the symmetric matrix to satisfy the time windows constraints. All the experiments were conducted on Pentium IV 3.20 GHz CPU PCs with 1 GB of RAM.

In the first part of our experiments, we made a brief comparison of the results obtained using the leader problem with two different objectives; namely, the total number of students and the total number of girls enrolled in the BSs to be visited. It is found that our solutions under both criteria are either dominating or non-dominated solutions when compared with the realized figures of the past ILKYAR's projects. The differences between the lengths of the visiting tours (in the follower problem)

under the two different criteria are quite insignificant, which would make the decision process easier. The maximum solution time is less than 100 seconds for the entire set.

Since the selection of the major distance limit would have a direct impact on the objective function value, we tested six different limits, ranging from 30 km to 180 km. Each problem instance is solved six times by incorporating these limits into the instance data. At the end of this part of our experiments, the major distance limit is suggested as 90 km because the results with major limit of 90 km are much more comparable with ILKYAR's results and a longer travel does not yield any significant amount of increment on the objective function value.

The minor distance limit on the return to hometown is another important factor in the project planning since the team members have to start their activities at home right after the project is over. To analyze the effect of the distance limit on the return, in the third part of our experiments, the SWRP model is run under the different settings of the distance limit, changing from 100 km to 1400 km with a step size of 100 km. The problem turned out to be infeasible when the distance limit was less than 800 km. It is found that a minor distance limit about 900 km is very reasonable.

References

[1] ILKYAR. http://www.ilkyar.org.tr/, Last accessed date: August 2009.

[2] Gutin G. and Punnen A.P., *Traveling Salesman problem and Its Variations*, Kluwer Dordrecht, (2002).

[3] Caramia M. and Dell'Olmo P., *Multi-objective Management in Freight Logistics*, 11-36, Springer London, (2008).

[4] Marinakis Y., Migdalas A. and Pardalos P.M., "A new bilevel formulation for the vehicle routing problem and a solution method using a genetic algorithm." *Journal of Global Optimization* 38, 555-580, (2007).

[5] Georgia Institute of Technology, Concorde TSP Solver, 2009. http://www.tsp.gatech.edu/concorde.html, Last Accessed Date: August 25, 2009.

Solving a real-world service technician routing and scheduling problem

Sophie N. Parragh

Department of Business Administration

University of Vienna, Bruenner Str. 72, Vienna, Austria

Email: sophie.parragh@univie.ac.at

1 Introduction

Motivated by the problem situation faced by an Austrian infrastructure service provider, we develop solution methods for what we call the service technician routing and scheduling problem (STRSP): a given number of technicians have to be scheduled in order to fulfill a given number of service tasks. Each task demands a technician that disposes of the appropriate skills of at least the demanded level. Maximum shift lengths have to be respected and lunch breaks have to be scheduled. In addition, two technicians' tours may have to be synchronized at certain points in time in order to complete those tasks that demand two technicians.

Recent publications originating from the field of service technician routing and scheduling include the work of Cordeau et al. [3]. The authors consider a service technician scheduling problem arising in large telecommunications companies. Travel times between the different locations of the tasks to be scheduled are neglected. The focus is put on the configuration of teams and the assignment of tasks to teams according to the required skills and skill levels, respecting task priorities and precedence relationships between the tasks. The objective is to minimize a weighted combination of the makespan of each priority class. In Xu and Chiu [8] staff scheduling is also performed for a telecommunication company. Each technician disposes of a certain skill level for each task. These skills are, however, modelled in a different way than in [3]. While Cordeau et al. use discrete skill levels, Xu and Chiu use percentages. Furthermore, in Cordeau et al. [3] this aspect is modelled in terms of constraints. In Xu and Chiu [8], on the other hand, each technician is associated with a certain proficiency level for each task in the system. The objective function maximizes the number of tasks to technicians weighted by the technicians' proficiency levels. Each task is thus more likely to be assigned to a technician with a high proficiency for this task than to a technician with low or no proficiency for this task.

2   Problem definition

The STRSP is a static problem. As in Tricoire [7] and Bostel et al. [2] the planning should be done for several days in advance, on a rolling horizon basis. Some tasks have a validity period of several days or weeks, while others have to be carried out on a specific day during a pre-defined time window. Every task is associated with a given service time. Additional tasks come in every day. As in Cordeau et al. [3] and Xu and Chiu [8] technicians dispose of different skills and skill levels (we consider discrete skill levels). A technician can only be assigned to a task if he or she disposes of the appropriate skills of at least the demanded level. Moreover, on most of the days a lunch break has to be scheduled within a given time window. The lunch break can be held at any location and may interrupt a task. The length of a technician's working day is limited by labor regulations concerning maximum shift lengths. We distinguish between regular, extra and over time, expressed in terms of differing wage costs. We currently assume that vehicles are not subject to choice, i.e. each technician is associated with a given vehicle. However, distance-based vehicle costs vary from vehicle to vehicle. Technicians are not available every day and they sometimes start and/or end their working days at their home locations. Normally they depart from and return to a central depot at the beginning, respective end, of their shifts. Each technician starts to work at a given point in time. Thus, shift lengths cannot be reduced by delaying the beginning of the technicians' working days. In some rare cases, however, some technicians have to start earlier in order to fulfill tasks that have to be completed before the regular beginning of the respective working day. Finally, a given number of tasks demand two technicians to meet at the respective task sites in order to be completed. This introduces an aspect of tour synchronization into the problem. The objective is to minimize total costs; the different cost components are the following:

- regular, extra- and overtime wages for the technicians,

- distance-based costs for gas and maintenance regarding the vehicles,

- outsourcing costs for tasks that cannot be fulfilled by the available technicians.

3   Solution methods

An exact as well as a heuristic solution method for the problem at hand are developed. In a first step, the STRSP has been formulated in terms of a polynomially sized mixed integer program (MIP). We use several pre-processing steps (time window tightening, variable fixing) in order to reduce the solution space. Additional valid inequalities are added in a cutting plane fashion. The STRSP is related to the vehicle routing problem with time windows (VRPTW). However, capacity restrictions are not considered. Therefore, all known families of inequalities that rely on vehicle capacity cannot be employed. Another complicating aspect refers to the fact that only some tasks

are associated with time windows. Thus, depending on the percentage share of tasks associated with a time window, time related valid inequalities, such as infeasible path inequalities (lifted into tournament constraints [1] and separated by an enumeration procedure) are not always useful. Subtour elimination constraints in their standard form, integrating the outsourcing aspect, and D_k^+ and D_k^- inequalities, initially developed for the asymmetric traveling salesman problem [5] and separated as described in Fischetti et al. [4], are also generated in a cutting plane fashion in order to strengthen the formulation. Furthermore, a greedy solution construction heuristic has been implemented in order to provide an initial upper bound. Note that due to the possibility to outsource those tasks that cannot be scheduled in a feasible way, despite the fixed number of technicians, a feasible solution can always be computed from scratch. Exact solutions for small problem instances can be computed by means of the branch-and-cut algorithm. However, as expected, instances of realistic size cannot be solved. These involve up 600 tasks, up to nine different technicians and a planning horizon of up to ten days.

In order to solve real-world problems, we develop a heuristic method. The two most troublesome issues concern the necessary synchronization of two technicians' routes in such a way that they perform certain tasks together and the insertion of the lunch break. In the greedy solution construction heuristic, the first issue is dealt with in the following way. We first insert those tasks that demand two technicians and we then tighten their time windows to a feasible point in time. This avoids dealing with synchronization in the heuristic scheduling algorithm. As soon as all tasks demanding synchronization have been assigned to technicians, the remaining tasks are inserted using feasible cheapest insertion, ignoring the lunch break. After every successful insertion the location of the lunch break is re-adjusted. All eligible positions are tried and the one causing the least number of time window and route duration violations is identified. Then, a greedy task removal procedure deletes tasks from the respective route until feasibility is re-attained. All removed tasks re-enter the pool of currently unassigned tasks and may be inserted again in a subsequent iteration. In order to avoid cycling, they cannot be assigned a technician's route they were previously deleted from.

This initial solution is then improved as follows. In an iterative fashion, following the variable neighborhood search idea [6], different neighborhood operators are employed, deciding which tasks should be removed (outsourced) and which tasks should be re-inserted. Every task that has been selected for insertion in the current iteration is either assigned to a day and/or to a technician, depending on the chosen assignment procedure. In a similar vein, neighborhood operators are defined that re-assign tasks that are currently scheduled to other days and/or technicians. Like Xu and Chiu [8], the resulting scheduling subproblems are solved to optimality by means of the branch-and-cut algorithm.

4 Outlook

Possible additional real-world aspects involve scarce tools and large spare parts. The tooling aspect concerns, on the one hand, portable equipment which can be loaded into any type of vehicle, and, on the other hand, specific vehicles providing certain equipment, e.g. long ladders that are attached to the roof top or flashing blue light.

References

[1] N. Ascheuer, M. Fischetti, and M. Grötschel. A polyhedral study of the asymmetric traveling salesman problem with time windows. *Networks*, 36:69–79, 2000.

[2] N. Bostel, P. Dejax, P. Guez, and F. Tricoire. Multiperiod planning and routing on a rolling horizon for field force optimization logistics. In B. Golden, S. Raghavan, and E. A. Wasil, editors, *The Vehicle Routing Problem: Latest Advances and New Challenges*, volume 43 of *Operations Research/Computer Science Interfaces*, pages 503–525, New York, US, 2008. Springer.

[3] J.-F. Cordeau, G. Laporte, F. Pasin, and S. Ropke. Scheduling technicians and tasks in a telecommunications company. *J Sched*, 2009. forthcoming.

[4] M. Fischetti, A. Lodi, and P. Toth. Exact mthods for the asymmetric traveling salesman problem. In G. Gutin and A.P. Punnen, editors, *The traveling salesman problem and its variations*, volume 12 of *Combinatorial optimization*, Dordrecht, 2001. Kluwer Academic Publishers.

[5] M. Grötschel and M. W. Padberg. Polyhedral theory. In *The traveling salesman problem*, pages 251–305, New York, 1985. Wiley.

[6] N. Mladenović and P. Hansen. Variable neighborhood search. *Comput Oper Res*, 24:1097–1100, 1997.

[7] F. Tricoire. *Optimisation des tournées de véhicules et de personnels de maintenance : application á la distribution et au traitement des eaux*. PhD thesis, École Nationale Supérieure des Techniques Industrielles et des Mines de Nantes, 2006.

[8] J. Xu and S. Y. Chiu. Effective heuristic procedures for a field technician scheduling problem. *J Heuristics*, 7:495–509, 2001.

On the solution of robust traffic network design and pricing problems

Christoffer Cromvik

Department of Mathematical Sciences

Chalmers University of Technology and University of Gothenburg

Michael Patriksson

Department of Mathematical Sciences

Chalmers University of Technology and University of Gothenburg, Göteborg, Sweden

Email: mipat@chalmers.se

1 Introduction

We investigate the optimization of tolls and other control measures in a traffic network, taking into account several sources of uncertainty. The goal is to produce controls that are robust to changes in the uncertain parameters. Since such hierarchical problems are modelled as mathematical programs with equilibrium constraints (MPEC), we are here considering their stochastic extension, the stochastic mathematical program with equilibrium constraints (SMPEC), as introduced in [6].

Let (Ω, Θ, P) be a complete probability space and consider the problem

$$
\begin{aligned}
(\text{SMPEC}_\Omega) \quad & \min_{(x, y(\cdot))} \quad \mathrm{E}_\omega[f(x, y(\omega), \omega)] := \int_\Omega f(x, y(\omega), \omega)\, P(d\omega), \\
& \text{s.t.} \quad x \in X, \\
& \quad\quad y \in C;\ F(x, y, \omega)^{\mathrm{T}}(z - y) \geq 0, \quad \forall z \in C, \qquad P\text{-a.s.,}
\end{aligned}
\tag{1}
$$

where $y : \Omega \to \mathbb{R}^m$ is a random element of the probability space (Ω, Θ, P). Further, $f : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$, $y \in \mathbb{R}^m$, $C \subseteq \mathbb{R}^m$ is a polyhedron, and $F(x, \cdot) : C \to \mathbb{R}^m$ is smooth.

The vector $x \in \mathbb{R}^n$ represents the design (or primary) variables and $y \in \mathbb{R}^m$ is the response (or secondary) variables. The nonempty, closed and convex set $X \subseteq \mathbb{R}^n$ specifies the set of feasible designs. In view of stochastic programming with recourse, SMPEC is considered as a here-and-now type of problem, where the decision x should be taken before any realizations of uncertain data.

The authors have been much involved in the development of SMPEC and its applications, in particular to the case of traffic control and pricing. The existence of solutions to the general SMPEC

problem was investigated in [6, 3]. In [4, 5] and in [1, 2] we have analyzed the stability of optimal, respectively, stationary solutions to the SMPEC when the underlying probability distribution is itself uncertain. This is motivated by practical applications as discussed previously, since any control scheme implemented should be robust to changes in the underlying data of the problem. In the latter references, it is also shown how to discretize a continuous distribution using sample average approximation (SAA) and that such an approximation will converge. This is important, as we then may utilize any efficient means to solve the deterministic problem. The SMPEC formalism has also been extended to cover both risk and multiple objectives.

2   A motivating example

The main objective in a network design problem is to influence the travel costs and the demands such that some criterion is optimized. The design problem can be formulated as an MPEC, where the traffic equilibrium is described by a system of mixed complementarity constraints. An example of a network design problem is given by setting link tolls through the design parameter $x \in \mathbb{R}^n$, with $n \leq |\mathcal{E}|$ (the number of links), such that the total travel cost $f(x, v) := \sum_{l \in \mathcal{E}} t_l(x, v) v_l$ is minimized, and where, for a given design x, the link flows v is given by the user equilibrium conditions. We present a small numerical example in the application of network design under user equilibrium. The deterministic example is known as Braess' paradox, demonstrating that adding an extra link to a network can cause an increase in the total travel cost. The figures (graphs I and II) below show the network graph with four and five links, respectively.
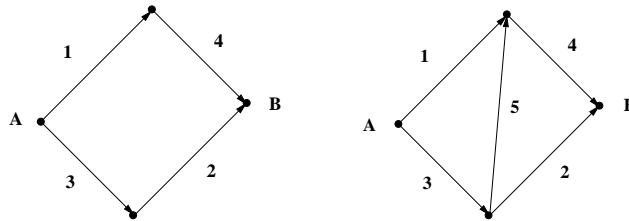
We have one OD-pair (A, B) with a fixed demand of $d = 6$ units. The original network has two paths, using the links $(1, 4)$ and $(3, 2)$, respectively; graph II has three paths, using the links $(1, 4)$, $(3, 2)$ and $(3, 5, 4)$, respectively. The link travel costs are $t_i = 50 + v_i$ for $i = 1, 2$, $t_i = 10 v_i$ for $i = 3, 4$, and $t_5 = 10 + v_5$. Given theses costs, the user equilibrium flows for graph I are $v = (3, 3, 3, 3)^{\mathrm{T}}$, $h = (3, 3)^{\mathrm{T}}$, with equilibrium travel cost $\pi = 83$. For graph II, the user equilibrium flows are $v = (2, 2, 4, 4, 2)^{\mathrm{T}}$, $h = (2, 2, 2)^{\mathrm{T}}$, with equilibrium travel cost $\pi = 92$.

We now consider adding a toll x on the new link, thus altering the travel cost to $t_5 = 10 + v_5 + x$, and consider the problem to minimize $f(x, v) + \tau x^2$ over $x \in X = \{\, x \in \mathbb{R} \mid 0 \leq x \leq 14 \,\}$ and $\tau > 0$ is a penalty parameter against setting a too high toll value. For a sufficiently small value of τ,

the optimal solution is $x^* = 13$ and the optimal total travel cost is $f(x^*, v^*) = 498$. The optimal solution $x^* = 13$ is the threshold value for which there will be no flow on link 5, which in turn will give a lower total travel cost.

Now consider the case when the travel costs are stochastic. In particular, assume that $t_i = 10v_i + \omega_{i-2}$, $i = 3, 4$, and that each component in ω is independent and drawn from a normal distribution with mean 0 and variance 1. The corresponding SMPEC model is solved using the discretization scheme SAA; since travel costs here are strongly monotone, optimal and stationary solutions are stable w.r.t. changes in the probability distribution, and SAA converges.

For a run with a maximum of 500 samples, the solver converged to the stationary solution $x^* = 14$. Below we plot histograms of the resulting objective values and equilibrium travel costs.
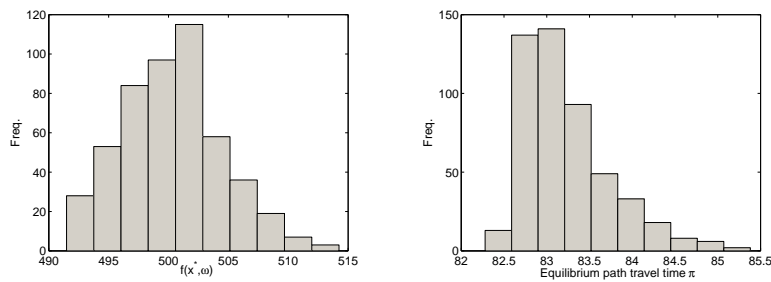
In order to illustrate the influence of the variance of the uncertain parameter on the solution, we also show histograms of the equilibrium travel cost for stationary solutions corresponding to the four values $\sigma_j^2 \in \{0.01, 0.025, 1, 4\}$, $j = 1, 2$ of the variance of the stochastic variable. The results are not surprising: a larger variance implies a larger spread in the response. Having access to histograms for responses, i.e., equilibrium solutions, is a feature of SMPEC which may be valuable for getting specific insights into an application.

3   Numerical examples

The main new contribution of the presentation compared with the existing literature will be numerical examples of solutions of toll optimization problems, both large-scale ones for the deterministic, MPEC, case using global solvers, and for the stochastic, SMPEC, case, using SAA and local solvers. If time permits, we will also investigate the utilization of equity objectives, which is quite naturally included as risk measures.
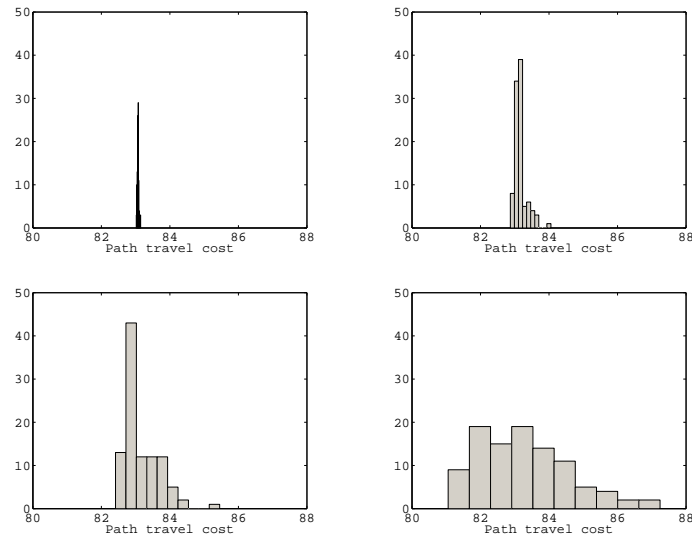
References

[1] C. Cromvik and M. Patriksson, "On the robustness of global optima and stationary solutions to stochastic mathematical programs with equilibrium constraints, part I: Theory", *Journal of Optimization Theory and Applications* 144, 461–478 (2010).

[2] C. Cromvik and M. Patriksson, "On the robustness of global optima and stationary solutions to stochastic mathematical programs with equilibrium constraints, part II: Applications, *Journal of Optimization Theory and Applications* 144, 479–500 (2010).

[3] A. Evgrafov and M. Patriksson, "On the existence of solutions to stochastic mathematical programs with equilibrium constraints", *Journal of Optimization Theory and Applications* 121, 65–76 (2004).

[4] M. Patriksson, "Robust bi-level optimization models in transportation science", *Philosophical Transactions of the Royal Society of London. Series A. Mathematical, Physical and Engineering Sciences* 366, 1989–2004 (2008).

[5] M. Patriksson, "On the applicability and solution of bilevel optimization models in transportation science: A study on the existence, stability and computation of optimal solutions to stochastic mathematical programs with equilibrium constraints", *Transportation Research* 42B, 843–860 (2008).

[6] M. Patriksson and L. Wynter, "Stochastic mathematical programs with equilibrium constraints", Operations Research Letters, 25, 159–167 (1999).

EVAQ: An Evacuation Model for Travel Behavior and Traffic Flow

Adam J. Pel, Michiel C.J. Bliemer, Serge P. Hoogendoorn

Transport and Planning Department

Delft University of Technology, Stevinweg 1, Delft, the Netherlands

Email: a.j.pel@tudelft.nl

1 Context and Problem Description

Traffic simulation models are helpful or even indispensable while planning or managing an evacuation. A multitude of dynamic traffic models have been developed to this end to understand and predict evacuation conditions on a road network, and the effect of traffic regulations and control measures hereon. In many earlier studies, evacuation is recognized as a special case regarding different travel demand patterns, driver behavior, traffic management, etc., resulting in new models dedicated to evacuation (e.g., OREMS [1], CEMPS [2], DYNEV [3], MASSVAC/TEDSS [4]). More recently, a large number of evacuation studies are conducted using traffic models originally developed for regular day-to-day traffic applications, including both microscopic models (e.g., PARAMICS, CORSIM, VISSIM) and macroscopic models (e.g., DYNASMART, DynaMIT, VISTA, CONTRAM). In several studies using microscopic models, model parameters describing driving behavior (headway, acceleration, reaction time) have been adjusted for the case of emergency evacuation.

These models used in past evacuation studies typically focus on traffic flow dynamics to identify bottlenecks where congestion is likely to occur and compute expected evacuation times. They do so by using a dynamic traffic assignment (DTA) model describing travel behavior and traffic flow. In spite of advances in evacuation modeling research, main shortcomings remaining in DTA models used for evacuation studies relate to (i) *route choice behavior* (user-equilibrium assumptions, invalidly disregarding travelers' unfamiliarity with evacuation traffic conditions), (ii) *compliance behavior* (actual travel decisions are modeled equal to instructed departure times, destinations and routes, invalidly disregarding travelers' preferences and compliance decisions), and (iii) *network dynamics* (road infrastructure is static, invalidly disregarding the impact of DTM measures and road network disruption due to the hazard). These essential aspects are typically neglected in current evacuation models, and hence in the scenario analyses when applying these simulation models. The consequences are 1) unreliable and likely wrong model outcomes and 2) being unable to, e.g., evaluate the impact of variations in traveler compliance, or test robustness of an evacuation strategy towards uncertain hazard conditions. In this paper, we propose the new DTA model EVAQ specifically tailored to the case of evacuation thereby solving the abovementioned shortcomings. The EVAQ model is described and applied to a large-scale case study of the evacuation of the Dutch metropolitan area of Rotterdam.

2 Evacuation Model Framework

The evacuation model EVAQ predicts travel behavior and traffic conditions on a road network for a wide range of emergency situations, such as hurricanes, bush fires and floods. Compared to other evacuation traffic models, the advantageous distinguishing features of EVAQ are: (i) modeling of dynamic road infrastructure, (ii) incorporation of adaptive route choice behavior towards network dynamics and travel information, and (iii) incorporation of evacuation instructions and traveler compliance behavior. EVAQ models time-dependent road infrastructure, meaning that speed limits, capacity and flow direction can be time-varying due to the hazard's progress in space and time (e.g., links becoming inaccessible due to flooding) and prevailing traffic regulation and control measures (e.g., contraflow operations to increase outbound capacity). While other models typically relate time-varying road infrastructure only to an impact in traffic flow propagation (e.g., lower speeds results in drivers experiencing higher travel times), in EVAQ this also affects en-route travel choice behavior (e.g., lower speeds leads to drivers adapting their routes at the next intersection). This same rerouting behavior is expressed when travelers receive new information on current traffic conditions. Both pre-trip and en-route route choice behavior are thus modeled by implementing a hybrid route choice model [5]. Traveler compliance towards instructions regarding departure time, destination and route is modeled by internalizing the generalized costs of deviating from these evacuation instructions, where these generalized costs are a function of the difference between the *preferred* travel decision and the
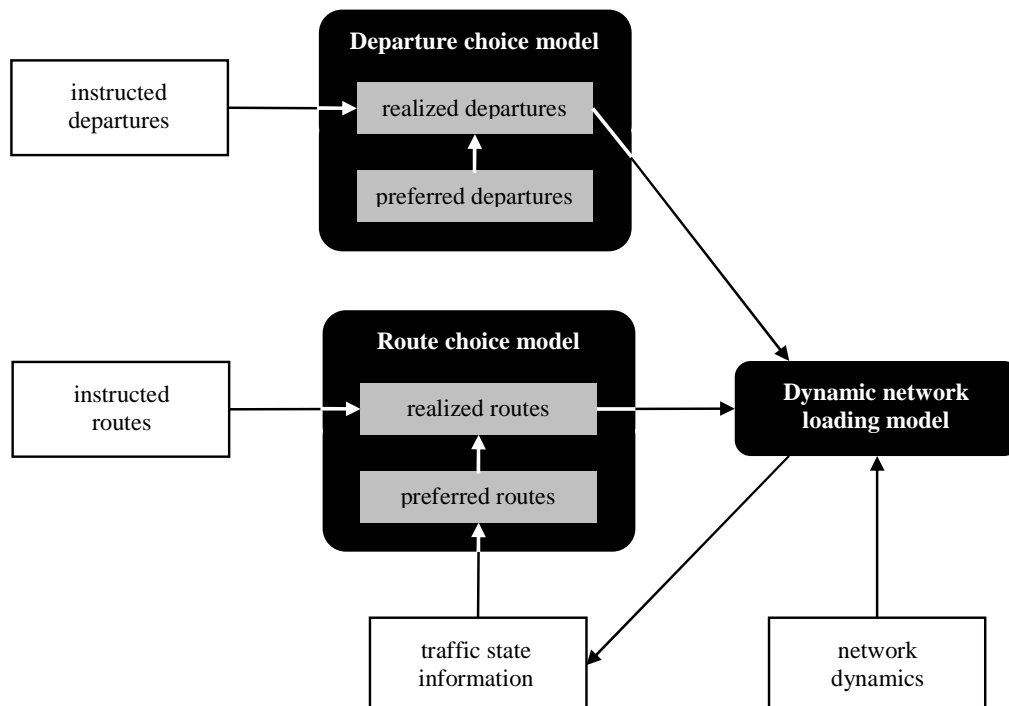
Figure 1: DTA components of EVAQ

instructed travel decision. A parameter is introduced dictating the relative weight of this cost term representing the travelers' willingness to comply and the authorities' enforcement to control. This way, subtle and behaviorally sound thresholds are built into the traveler choice model components allowing the researcher to analyze the impact of variations in traveler information and compliance behavior [6].

The model framework is depicted in Figure 1, showing the three model components for departure time decisions, route decisions, and traffic flow, and the three model inputs relating to instructions, information, and network dynamics (affected by the hazard). In the paper, we give a mathematical formulation of the model components modeling travelers' decisions regarding departure time and route (implying destination), and elaborate on how we incorporate the impact of travel information and instructions on these decisions. Subsequently, we present the traffic flow component simulating the traffic conditions, and show how network dynamics due to traffic control and network disruptions are modeled.

3 Model Application

To illustrate the scalability and potential of EVAQ, the model is applied to a case study describing the evacuation of the Dutch metropolitan area of Rotterdam (see Figure 2). With a population exceeding 600,000 inhabitants, the municipality forms the second largest in the country. The road network used in this study consists of the Rotterdam ring road, motorways connected to this ring road, main (provincial and urban) arterials, and collector roads, leading to approximately 500 links and 220 nodes, including 80 origins. EVAQ is implemented in Matlab. In case of applying a time step of 20 seconds in the traffic simulation model and simulating a time horizon of 48 hours, the CPU running time on a Windows XP computer with 2.2 GHz processor ranges from 10 to 20 minutes.
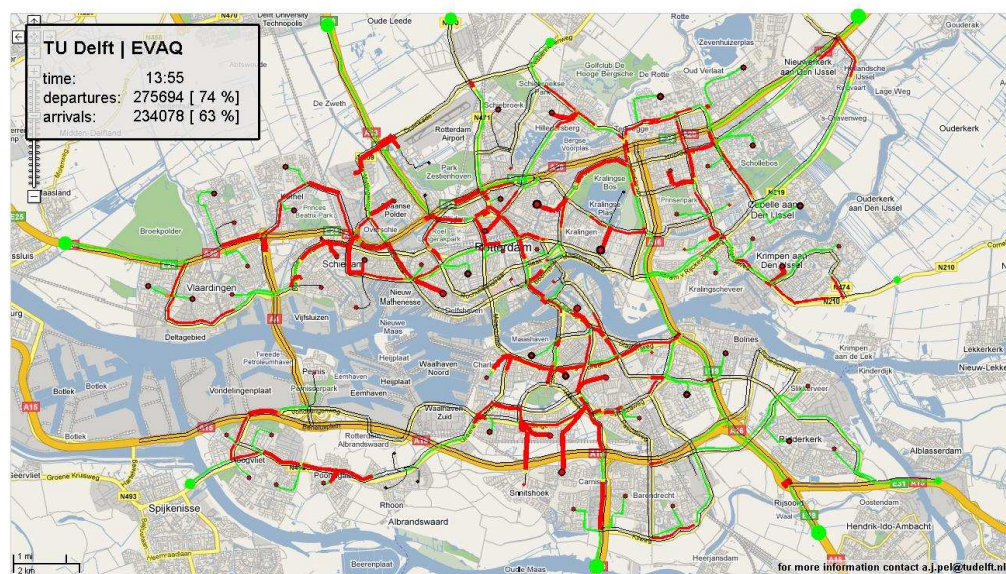
Figure 2: Rotterdam evacuation network

Rotterdam being a large harbor city, evacuation due to flooding, industrial accidents, or terrorist threats can be considered conceivable. Multiple simulations have been run varying in possible network exit points, traffic information levels, evacuation instructions, traveler compliance behavior, and network dynamics. These behavioral and control settings determine dynamic travel demands and route flow rates, and thus traffic states and network outflow utilization, where these relationships are shown to be (in some cases highly) non-linear and non-monotonic. Network clearance times for the different settings vary from 24 to 48 hours. More on the evacuation study can be found in [6].

EVAQ outputs link inflow and outflow rates, departure and arrival patterns, travel times, average speeds, queue lengths, etc. This dynamic information can be used to make founded decisions on, e.g., the latest possible time to start evacuation, the best evacuation routes, the impact of traffic information and evacuation instruction provision including compliance behavior, the most suitable dynamic traffic control measures, etc.

4 Discussion

This contribution presents the evacuation DTA model EVAQ, dealing with several shortcomings of current evacuation models, as mentioned above. The model framework is discussed and each of the components is described, as well as the manner in which these interact. The scalability and potential of EVAQ is illustrated by a real-life application describing the evacuation of the Dutch municipality of Rotterdam. In conclusion, the model framework and formulation, case results, discussion and conclusions presented in the paper can be used 1) to give direction to further research along this line on incorporating traveler choice behavior, compliance behavior, and network dynamics in evacuation simulation models, and 2) to understand the role of these aspects in the evacuation process and their assessment in evacuation planning studies.

References

[1] A.K. Rathi and R.S. Solanki, "Simulation of Traffic Flow during Emergency Evacuation: A Microcomputer based Modeling System", 25th Winter Simulation Conference, Los Angeles, CA, December 1993.

[2] M. Pidd, F.N. de Silva and R. Eglese, "CEMPS: A Configurable Evacuation Management and Planning System – A Progress Report", 25th Winter Simulation Conference, Los Angeles, CA, December 1993.

[3] KLD: "Formulations of the DYNEV and I-DYNEV Traffic Simulation Models used in ESF", Technical report prepared for the Federal Emergency Management Agency, KLD Associates, Inc., 1984.

[4] H.D. Sherali, T.B. Carter and A.G. Hobeika, "A Location-Allocation Model and Algorithm for Evacuation Planning under Hurricane/Flood Conditions", *Transportation Research Part B*, 25(6), pp 439-452, 1991.

[5] A.J. Pel, M.C.J. Bliemer and S.P. Hoogendoorn, "Hybrid Route Choice Modeling in Dynamic Traffic Assignment", *Transportation Research Record*, 2091, pp 100-107, 2009.

[6] A.J. Pel, S.P. Hoogendoorn and M.C.J. Bliemer, "Evacuation Modeling including Traveler Information and Compliance Behavior", 1st International Conference on Evacuation Modeling and Management (ICEM), The Hague, September 2009.

Congestion Pricing for Airport Efficiency

Georgia Perakis

Sloan School of Management, MIT, Cambridge, MA 02139, USA, georgiap@mit.edu

Wei Sun

Operations Research Center, MIT, Cambridge, MA 02139, USA, sunwei@mit.edu

1   Introduction

Airport congestion is a serious problem in many cities around the world. In 2007, flight delays cost passengers, airlines and the U.S. economy more than \$40 billion. Unfortunately, increasing capacity is not an option in many cases. Despite a vast amount of literature on tolls in ground transportation, the classical results on road congestion are not applicable to airport. The fundamental difference is that road users are non-atomic while an airport is served by a relatively small number of airlines. In particular, an airline experiences additional delays imposed on its own flights if it schedules an additional flight at a congested airport.

In air transportation, Daniel [2, 3] was the first to illustrate the potential benefits of congestion pricing with a simulation model. Brueckner [1] and Pels and Verhoef [4] show that an airline should only pay for the congestion damage that it imposes on other airlines.

The main goals of our work are to determine when congestion pricing is beneficial, to quantify its potential efficiency gains, and to explore novel ways to implement it such that both airlines and passengers can benefit. In particular, the main contributions of this paper are:

- Contrary to most existing literature which focuses on identical airlines, we allow asymmetry across many airlines with multiple types of aircrafts.

- We capture the reality that only part of the toll revenue collected may be used to benefit airlines and passengers.

- We analyze the problem from the societal perspective *and* quantify the individual impact on airlines and passengers.

- We propose an alternative implementation approach which is based on welfare sharing and achieves both efficiency and fairness.

2   Model

We consider an airport in a single period when congestion is always present. To reduce notation and enhance the transparency of the model, we only describe the setting for multiple airlines with one type of aircraft in this abstract. The results extend to the case with multiple aircrafts per airline.

In this setting, there are n airlines competing by deciding how many tickets to sell, $\mathbf{q} = (q_i, ..., q_i)$, where q_i is the number of tickets sold by airline i. We adopt the assumptions in [4], where each airline's price demand function $p_i(\mathbf{q})$ and average congestion cost $l_i(\mathbf{q})$ are linear functions of the total volume of traffic in the airport. However, contrary to [4] which studies an identical duopoly setting, we allow $p_i(\mathbf{q})$ and $l_i(\mathbf{q})$ to vary across airlines.

We analyze a three-level model: at the base level, passengers choose airlines according to the Wardrop Equilibrium Principle, that is, selecting the cheapest and fastest way to travel; the airlines form the middle level and participate in an oligopolistic quantity competition to maximize their own profit; at the highest level, the airport regulator maximizes the total social welfare by imposing a toll on each aircraft that uses the airport.

Airline's profit, π_i, is obtained by deducting its operating cost and congestion cost from the revenue generated from ticket sales. Under quantity competition, taking the competitors' optimal output $\mathbf{q}^*_{-\mathbf{i}}$ as given, each airline solves the following profit-maximizing problem, i.e., $\max_{q_i \geq 0} \pi_i(q_i, \mathbf{q}^*_{-\mathbf{i}})$.

The regulator acts as the Stackelberg leader and determines tolls anticipating that airlines will select their ticket quantities according to a Nash equilibrium. To better reflect reality, we introduce a parameter $\rho \in [0, 1]$ to denote how efficiently the regulator utilizes the toll revenue. $\rho = 1$ arises in the ideal situation that all the toll proceeds have been used to improve social welfare.

The total social welfare incorporates the consumer surplus (CS), all airlines' profits (PS), and the actual utilization of the toll revenue (TR), ρTR. For a fixed toll utilization rate ρ, the regulator's welfare-maximization problem is defined as follows,

$$\max_{\mathbf{t}} \quad w(\rho) = CS(\mathbf{q}(\mathbf{t})) + PS(\mathbf{q}(\mathbf{t})) + \rho TR(\mathbf{q}(\mathbf{t}))$$

$$s.t. \quad \mathbf{q}(\mathbf{t}) = \arg \max_{q_i(t) \geq 0} \pi_i(q_i, \mathbf{q}^*_{-\mathbf{i}}), \quad \forall i = 1, 2, ..., n. \tag{1}$$

3   The "Price" of Decentralization

We denote the optimal total social welfare obtained by solving (1) and the total welfare under the no-toll setting as $w^*(\rho)$ and w^{uo} respectively.

Lemma 3.1 $w^*(\rho)$ *increases in* ρ *and there exists* $\bar{\rho}$ *such that* $w^*(\bar{\rho}) = w^{uo}$.

The results give a criterion for when congestion pricing should be implemented: if $\rho \leq \bar{\rho}$, the regulator should not intervene in the market. It is interesting to note that $\bar{\rho}$ is often larger than 0.8, which implies a rather stringent requirement on the regulator's ability of utilizing the toll revenue in order to justify imposing congestion pricing.

When $\rho = 1$, i.e., all toll revenues directly benefit society, the optimal total social welfare, w^*, is achieved. We use $\frac{w^{uo}}{w^*}$ as a measure of efficiency loss in the no toll setting.

Definition 3.1 *The congestion sensitivity ratio is defined as* $\gamma = \min_i \frac{l_i'(\mathbf{q})}{p_i'(\mathbf{q})}$.

While $l_i'(\mathbf{q})$ measures the airline's marginal profit loss due to congestion (e.g., increase in fuel expenses and crew costs), $p_i'(\mathbf{q})$ measures the marginal revenue from selling an extra ticket, excluding the impact of congestion. A high γ indicates that airlines are more sensitive to congestion, implying a serious congestion problem in that airport. On the other hand, when γ is small, airlines are not too concerned about congestion.

Theorem 3.1 *When* $\gamma \leq 1$, $\frac{w^{uo}}{w^*} \geq \frac{3}{4}$; *when* $\gamma > 1$, $\frac{w^{uo}}{w^*} \geq \frac{1}{1+\gamma}$.

When γ is small, the no-toll setting loses at most 25% of the total welfare compared to the setting with optimal tolls. However, when γ is large, i.e., the congestion problem at the airport is severe, the loss of welfare can be arbitrarily large, which implies large potential efficiency gains can be achieved from implementing congestion pricing.

4 Welfare Redistribution

As pointed out in the previous section, when $\rho > \bar{\rho}$, the total social welfare exceeds w^{uo} (social welfare without tolls). However, both airlines and passengers are worse off. The producer surplus $PS(\rho)$ decreases because airlines are forced to produce under the profit-maximizing quantity. Meanwhile, a smaller number of traveling passengers results in a lower consumer surplus $CS(\rho)$. We are interested in finding out how to redistribute the toll revenue such that airlines and passengers will have higher surpluses than their counterparts PS^{uo} and CS^{uo} under the no-toll setting.

Suppose the regulator plans to utilize ρTR by giving ϕ portion of the toll revenue to airlines and the rest to passengers. We denote the new producer surplus and consumer surplus after the welfare redistribution as $\overline{PS}$ and $\overline{CS}$ respectively, whereby $\overline{PS} = PS(\rho) + \phi \rho TR$, and $\overline{CS} = CS(\rho) + (1 - \phi)\rho TR$.

Theorem 4.1 *If the toll revenue is redistributed according to* ϕ^*, *where* $\phi^* = \frac{PS(\rho)}{PS(\rho)+CS(\rho)}$, *then* $\overline{PS} \geq PS^{uo}$ *and* $\overline{CS} \geq CS^{uo}$.

The result incorporates the notion of "fairness" into welfare sharing, i.e., the fraction of the toll revenue given to airlines and passengers is proportional to their original surplus. In addition, when

the toll revenue is divided according to ϕ^*, both airlines and passengers are better-off than the no-toll setting. One natural question is how to implement the welfare sharing in practice. While toll revenue could be given to airlines as some form of rebate, it is not immediately clear how to pass it on to passengers.

In view of this difficulty, we propose an alternative way to implement congestion pricing which is efficient and both airlines and passengers enjoy higher surpluses.

Theorem 4.2 *There exists a price function $\tilde{p}_i(\mathbf{q}) = p_i(\mathbf{q}) + f(\phi^*)$, such that the total social welfare in the no-toll setting achieves w^*. In addition, airlines and passengers achieve $\overline{PS}$ and $\overline{CS}$ respectively.*

Instead of charging a toll on each flight and then giving back part of the toll revenue to achieve higher surplus, the regulator could impose a surcharge on fares, $f(\phi^*)$. Under the new price demand function, the society operating at the Nash equilibrium achieves the optimal total welfare, w^*. This is because the increase in fare prices reduces the demand, consequently reducing congestion in the airport. At the same time, higher fare prices allow airlines to achieve higher profits and traveling passengers enjoy higher utility. Simulation results show that the surcharge is usually small, e.g., less than 5% increase in price.

This alternative approach of implementing congestion pricing (i.e., modifying the price function with a surcharge) bypasses the difficulty of physically redistributing the toll revenue to airlines and passengers. It is also efficient as the society achieves optimal welfare. Furthermore, since this approach promises airlines higher earnings, it is more likely to gain support and acceptance in practice.

Acknowledgement

References

[1] Brueckner, Jan K., "Airport Congestion When Carriers Have Market Power", *The American Economic Review* 92, 1357-1375 (2002).

[2] Daniel, Joseph I, "Congestion Pricing and Capacity of Large Hub Airports: A Bottleneck Model with Stochastic Queues", *Econometrica*, 63, 327-70 (1995).

[3] Daniel, Joseph I, "Distributional Consequences of Airport Congestion Pricing", *Journal of Urban Economics*, 50, 230 - 258 (2001).

[4] Eric Pels and Erik T. Verhoef, "The Economics of Airport Congestion Pricing", *Journal of Urban Economics*, 55, 257-277 (2004).

VRP Solved by Bounding and Enumeration of Partial Paths

Mads Kehlet Jepsen

DTU Management Engineering

Technical University of Denmark, Denmark

Bjørn Petersen

DTU Management Engineering

Technical University of Denmark, Denmark

Email: bjorn@diku.dk

1 Introduction

The CVRP can be described as follows: A set of customers, each with a demand, needs to be serviced by a number of vehicles all starting and ending at a central depot. Each customer must be visited exactly once and the capacity of the vehicles may not be exceeded. The objective is to service all customers traveling the least possible distance. In this abstract we consider a homogeneous fleet, i.e., all vehicles are identical. The VRPTW extends the CVRP by imposing that each customer must be visited within a given time window. The overlap of the CVRP and the VRPTW will in the following be referred to as the VRP.

The standard Dantzig-Wolfe decomposition of the arc flow formulation of the VRP is to split the problem into a master problem (a Set Partitioning Problem) and a pricing problem (an Elementary Shortest Path Problem with Resource Constraints (ESPPRC), where capacity (and time) are the constrained resources). A restricted master problem can be solved with delayed column generation and embedded in a branch-and-bound algorithm to ensure integrality. Applying cutting planes either in the master or the pricing problem leads to a Branch-and-Cut-and-Price algorithm (BCP). [1] implemented a successful BCP algorithm for the VRPTW by applying *sub-tour elimination* constraints and *two-path* cuts, [2] generalized the *two-path* cuts to the *k-path* cuts, and [3] applied a range of valid inequalities for the CVRP based on the branch and cut algorithm of [4]. Common for these BCP algorithms is that all applied cuts are valid inequalities for the VRPTW respectively the CVRP with regard to the *original* arc flow formulation, and have a structure which makes it possible to handle values of the dual variables in the pricing problem without increasing the

complexity of the problem. The BCP algorithm was extended to include valid inequalities for the master problem by applying the subset row (SR) inequalities to the Set Partitioning master problem in [5] and later by applying Chvátal-Gomory Rank-1 (CG1) inequalities in [6]. [7] use an approach where columns with potentially negative reduced cost is enumerated after good upper and lower bounds are found, this sometimes leads to memory issues with difficult instances. After enumeration a general MIP solver is called. Recently, [8] presented an new decomposition model based on bounded partial paths, where the solution space of the pricing problem is limited by bounding some resource.

We propose to combine the latter two strategies, i.e., enumeration of columns with potentially negative reduced with the columns being bounded partial paths. The main ideas of [7] would be utilised until the enumeration step where the partial path columns would be used instead of the much larger set of elementary routes, thus hopefully solving the memory issues notied in [7]. The gap between the lower (LB) and the upper bound (UB) of the master problem obtained with the elementary routes can be maintained by bounding LB.

2 Mathematical Model

The VRP can formally be stated as: Given a graph $G(V, A)$ with nodes V and arcs A, a set R of resources $R = \{\text{load (and time)}\}$ where each resource $r \in R$ has a lower bound a_i^r and an upper bound b_i^r for all $i \in V$ and a positive consumption τ_{ij}^r when using arc $(i, j) \in A : i \in C$, find a set of routes starting and ending at the depot node $0 \in V$ satisfying all resource limits, such that the cost is minimized and all customers $C = V \setminus \{0\}$ are visited.

In the following let c_p be the cost of partial path $p \in P$, λ_p be the binary variable indicating the use of p, and T_{ij}^r (the resource stamp) be the consumption of resource $r \in R$ at the beginning of arc $(i, j) \in A$. Let $\delta^+(i)$ and $\delta^-(i)$ be the set of outgoing respectively ingoing arcs of node $i \in V$. Finally, let LB be a given lower bound. The master problem:

$$\min \ \sum_{p \in P} c_p \lambda_p \tag{1}$$

$$\text{s.t.} \ \sum_{p \in P} c_p \lambda_p \geq LB \tag{2}$$

$$\sum_{p \in P} \sum_{(i,j) \in \delta^+(i)} \alpha_{ij}^p \lambda_p = 1 \qquad \forall i \in C \tag{3}$$

$$\sum_{p \in P : e^p = i} \lambda_p = \sum_{p \in P : s^p = i} \lambda_p \qquad \forall i \in V \tag{4}$$

$$\sum_{p \in P} \lambda_p = K \tag{5}$$

$$\sum_{(j,i) \in \delta^-(i)} \left(T_{ji}^r + \sum_{p \in P} \tau_{ji}^r \alpha_{ji}^p \lambda_p \right) \leq \sum_{(i,j) \in \delta^+(i)} T_{ij}^r \qquad \forall r \in R, \ \forall i \in C \tag{6}$$

$$a_i \sum_{p \in P} \alpha_{ij}^p \lambda_p \le T_{ij}^r \le b_i \sum_{p \in P} \alpha_{ij}^p \lambda_p \qquad \forall r \in R, \ \forall (i,j) \in A \qquad (7)$$

$$T_{ij}^r \ge 0 \qquad \forall r \in R, \ \forall (i,j) \in A \qquad (8)$$

$$\lambda_p \in \{0,1\} \qquad \forall p \in P \qquad (9)$$

Where α_{ij}^p is the number of times arc $(i,j) \in A$ is used on path $p \in P$ and s^p and e^p indicate the start respectively the end node of partial path $p \in P$. Constraints (3) ensure that each customer is visited exactly once. Constraints (4) link the partial paths together by flow conservation. Constraint (5) is the convexity constraint ensuring that K partial paths are selected. Constraints (6) and (7) enforce the resource windows.

3 Algorithmic Overview

The algorithm is inspired by the one of [7]. Heuristics can be applied where ever being beneficial.

  i By the use of column generation with columns being elementary routes and all advantageous cuts being utilized (e.g., (SR)), find a big LB and a small UB.

  ii Solve the LP-relaxed master problem (1)–(9) with the LB from i.

  iii Due to the bounds (9) each column in P cannot be in a solution more than once, hence, any partial path $p \in P$ in an optimal solution must satisfy: $\bar{c}_p \le UB - LB$, where $\bar{c}_p$ is the reduced cost of column p in the last iteration of ii. Enumerate all these.

  iv Apply all the columns from iii to the master problem, add the cuts of Section 4, and give the problem to a general MIP-solver of your own choice.

4 Tightening Bounds

Constraints (6) and (7) can be tightened by:

$$\sum_{p \in P: e^p = i} (a_p + \tau_p)\lambda_p \le \sum_{(i,j) \in \delta^+(i)} T_{ij} \qquad \forall i \in C \qquad (10)$$

$$\sum_{p \in P, s^p = i} a_p \lambda_p \le \sum_{(i,j) \in \delta^+(i)} T_{ij} \le \sum_{p \in P, s^p = i} b_p \lambda_p \qquad \forall i \in V \qquad (11)$$

where a_p, b_p, and τ_p are bounds on the partial path p and due to integrality on p can yield tighter bounds. Lower bound a_p for p is defined as the latest possible departure time from the start-node s without changing the earliest possible arrival time at the end-node e. Upper bound b_p for p is defined as the latest possible departure time from s while p still being feasible. Travel time τ_p is defined as the time spend on p, i.e., traversing edges and waiting for windows to open. It is noted that τ_p is always constant given a_p and b_p as defined above no matter which departure time t :

$a_p \le t \le b_p$, since the traversal times of edges are constant and a difference in waiting time would yield a conflict with the definition of a_p. As a consequence of this, $a_p = b_p$ if there is waiting time on p, and if $a_p \ne b_p$ then no waiting time occurs on p.

Even though the influence on the reduced cost with these cuts can be handled in the pricing problem, experience points to it not being easy and not without negatively influencing the running time. In the context of enumeration the dual cost of these cuts do not have to be handled since they are added after the enumeration procedure, thus obtaining the smaller solution space for "free".

References

[1] N. Kohl, J. Desrosiers, O. B. G. Madsen, M. M. Solomon, and F. Soumis, "2-path cuts for the vehicle routing problem with time windows," *Transportation Science*, vol. 33, no. 1, pp. 101–116, 1999.

[2] W. Cook and J. L. Rich, "A parallel cutting plane algorithm for the vehicle routing problem with time windows," Tech. Rep. TR99-04, Computational and Applied Mathematics, Rice University, Houston, Texas, USA, 1999.

[3] R. Fukasawa, H. Longo, J. Lysgaard, M. P. de Arago, M. Reis, E. Uchoa, and R. Werneck, "Robust branch-and-cut-and-price for the capacitated vehicle routing problem," *Mathematical Programming*, vol. 106, no. 3, pp. 491–511, 2006.

[4] J. Lysgaard, A. N. Letchford, and R. W. Eglese, "A new branch-and-cut algorithm for the capacitated vehicle problem," *Mathematical Programming*, vol. 100, no. A, pp. 423–445, 2004.

[5] M. Jepsen, B. Petersen, S. Spoorendonk, and D. Pisinger, "Subset-row inequalities applied to the vehicle routing problem with time windows," *Operations Research*, vol. 56, no. 2, pp. 497–511, 2008.

[6] B. Petersen, D. Pisinger, and S. Spoorendonk, "Chvátal-Gomory rank-1 cuts used in a Dantzig-Wolfe decomposition of the vehicle routing problem with time windows," in *The Vehicle Routing Problem: Latest Advances and New Challenges* (B. Golden, R. Raghavan, and E. Wasil, eds.), pp. 397–420, Springer, 2008.

[7] R. Baldacci, N. Christofides, and A. Mingozzi, "An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts," *Mathematical Programming*, vol. 115, no. 2, pp. 351–385, 2008.

[8] M. K. Jepsen and B. Petersen, "Partial path column generation for the vehicle routing problem," tech. rep., DTU Management Engineering, Technical University of Denmark, Produktionstorvet 424, 2800 Kgs. Lyngby, Denmark, 2009.

The Simultaneous Vehicle Scheduling and Passenger Service Problem

Hanne L. Petersen[1], Allan Larsen, Oli B.G. Madsen, Stefan Røpke

Department of Transport

Technical University of Denmark, Kgs. Lyngby, Denmark

[1] Email: hlp@transport.dtu.dk

1   Problem description

Passengers using public transport systems often experience waiting times when transferring between two scheduled services. We propose a planning approach which seeks to obtain a favourable trade-off between the conflicting objectives of operating cost and quality of passenger service as part of the vehicle scheduling process for such systems.

The well-known Vehicle Scheduling Problem (VSP) is concerned with determining a set of vehicle schedules to operate a given timetable at the lowest possible cost. This problem is often encountered within bus operation in public transport.

The Simultaneous Vehicle Scheduling and Passenger Service Problem is based on the VSP, with two significant modifications: First, the trips of the timetable are allowed to be shifted by a few minutes to an earlier or later departure time, in the hope that this increased flexibility can lead to a lower operating cost, without introducing significant changes to the timetable. Secondly, a measure of passenger service is introduced for the evaluation of solutions, in order to control the effects of this timeshifting.

The measure of passenger service used in the calculations is based on the passenger waiting times at transfers between different lines. These transfers can take place between two lines that are both under the control of the model (and thereby have varying departure times), or to/from a line that is external to the model (and thereby has a fixed timetable). It should also be noted that passenger waiting times considered in this work, are only waiting times at transfers and not those experienced by passengers entering the system.

An application from the Greater Copenhagen Area is studied, dealing with the network of local express buses, which is build up around the local train network. A schematic overview of this network can be seen in Figure 1, where the dashed lines represent train lines (roughly with the

form of a fan), and the solid lines (all shades) show the available bus lines, of which most are perpendicular to the train lines.

The SVSPSP takes as input existing timetables and data regarding passenger flow at transfer points. For the SVSPSP these passenger flows are assumed to be fixed and independent of the operated timetables. The set of input timetables dictates the required level of service, and the number of departures is unchanged by the solution. Additionally, each departure can only be timeshifted by a limited number of minutes, meaning that the overall distribution of departures over the span of the day is not changed significantly.

To the authors' knowledge, no existing literature deals with a problem identical to the SVSPSP, and few similar problems have been treated, integrating vehicle scheduling and timetabling problems. The integration of timetabling and multi-depot vehicle scheduling is studied in [2] with the aim of reducing costs (reducing the number of vehicles) while ignoring passenger waiting times. The approach allows the trip starting times for each line to be timeshifted to allow greater flexibility in the vehicle scheduling part. The paper presents integer programming models as well as a local search algorithm that solves a network flow problem in each local search iteration. [1] also integrates vehicle scheduling and timetable synchronisation in an optimisation problem. The authors consider several terms in the objective function: number of vehicles required, number and quality of transfer possibilities and the so-called headway evenness. The second term aims at minimising passenger inconvenience, while the last term attempts to make regular arrivals on each line. These three terms are weighted together.
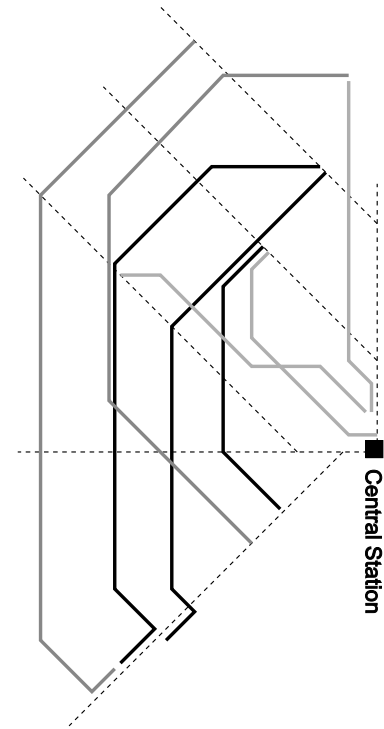
Figure 1: The structure of the overall public transportation network in the Greater Copenhagen Area

2 Data issues

The problem has been solved on a dataset derived from real-life data from the Greater Copenhagen Area, providing the geographic information and route networks. The currently operated train timetables have been used as fixed input, and the current bus timetables have been used as the starting point for the solution.

Real-life data regarding numbers of (dis)embarking passengers at each stop are unfortunately

not available at present, and have instead been estimated depending on line, location, and time of day. Furthermore, a percentage distribution of (dis)embarking passengers among lines available for transfer has been estimated. Finally, the combination of these numbers has led to the number of transferring passengers at each existing connection.

The units of the different objectives (operating cost and passenger waiting time) have been combined by conversion to a common monetary unit, by using the value of travel time recommended by the Danish Ministry of Transport.

3   Solution and results

The SVSPSP has been solved using a Large Neighbourhood Search (LNS) approach [3], where the initial solution is constructed using a greedy VSP heuristic on the existing timetable. The *destroy* operator removes trips either at random, or based on similarity with previously removed trips, where the measure of similarity is based on shared trip end points or proximity in time. At *insertion* each trip may be timeshifted with a certain percentage, and the procedure is then based on a cheapest insertion principle. The acceptance criterion for a new solution is based on simulated annealing.

The solution algorithm has been tested on instances of 3 different sizes based on subsets of increasing size of the real-life data. First a "small" instance, containing 3 bus lines (black in Figure 1), which all have quite many (5–6) intersections with the train network, but few connections between buses. These lines have a rather high passenger intensity, and a total of 538 trips per day. The "medium" instance contains 5 bus lines and is a superset of the small instance (black and dark grey in Figure 1), with on average slightly fewer intersections with the train network, and not quite as many passengers (792 trips per day). Finally, the "large" set contains all 8 bus lines that are in current operation in the real-life dataset, with a mixture of transfers to trains and buses and a total of 1400 trips per day.

	cost red.	empty	time	shifts
3 lines	3.3%	−8.9%	18.1%	73.8%
5 lines	3.2%	−7.8%	22.5%	78.2%
8 lines	2.0%	−7.1%	16.4%	76.4%

Table 1: Solution improvements for different problem sizes

Table 1 summarises the results obtained by running the LNS heuristic for 24 hours, with reductions compared to a reference solution obtained using the same heuristic without allowing timeshifting. The table shows the total cost reduction, the reduction in empty mileage (a negative reduction indicating that empty mileage has increased), the reduction in passenger waiting time,

and the number of trips that have been timeshifted.

The total cost that is considered in the table is a combination of operating cost and cost of passenger waiting time, and the results show that a certain reduction of this cost (2–3%) can be obtained by the suggested solution procedure. This happens at the expense of empty mileage (increasing by 7–9%), but leads to a considerable reduction of passenger waiting time at transfers. Finally, we can see that around 75% of all trips have been timeshifted in the final timetable, indicating that timeshifting certainly has an impact.

Furthermore, we can consider the degree of "memorability" of the obtained solutions, which we use to express that the timetable is easier for passengers to remember if buses depart at regular intervals. This is appreciated by operators, but has not been included in the optimisation. For the solutions reported above, the memorability is in the range 35–50%, which is a considerable reduction in comparison to the reference solution, where these values are in the range 72–84%.

4   Further work

Some suggestions for further work on the problem presented here would be twofold; first, it could be interesting to obtain better data for the passenger movements in the case study, and second, some improvements concerning the solution procedure are possible.

In particular, the SVSPSP could be treated as a multiobjective problem, regarding the operating cost and passenger waiting time as separate objectives, for example by considering the improvement of passenger waiting time, that can be obtained by limiting the allowed increase of total operating cost (possibly to zero).

Furthermore, the memorability of a solution could be considered an additional objective and added to the solution procedure. The results so far indicate that this value does indeed deteriorate under the current solution approach.

References

[1] V. Guihaire, J.-K. Hao, "Transit Network Re-Timetabling and Vehicle Scheduling", *Communications in Computer and Information Science* 14, 135-144 (2008).

[2] A. van den Heuvel, J. van den Akker, M. van Koten Niekerk, "Integrating timetabling and vehicle scheduling in public bus transportation", Technical Report, Department of Information and Computing Sciences, Utrecht University, 2008.

[3] P. Shaw, "Using constraint programming and local search methods to solve vehicle routing problems", CP-98 (Fourth International Conference on Principles and Practice of Constraint Programming) 1520, 417-431 (1998).

Airline Network Design under Congestion

João P. Pita

Department of Civil Engineering

University of Coimbra, Portugal

Email: jpita@mit.edu

António P. Antunes

Department of Civil Engineering

University of Coimbra, Portugal

Email: antunes@dec.uc.pt

1 Introduction

Airport congestion and flight delays are among the main problems faced today by the air transportation industry, being at the origin of important losses for the airlines and for the economy as a whole (see [1] and [2] for information about the US). In this article, we present a mixed-integer optimization model for airline network design where the implications of airport congestion and flight delays are taken into account. An application of the model made for a simplified version of the TAP Portugal network indicates that the model can provide very useful results within reasonable computation effort.

2 Problem

We focus on the problem of an airline wanting to determine the "next season" flight schedule and fleet assignment that maximizes its profit. The airline has a given fleet, serves a given number of markets with a hub-and-spoke network, and operates mainly in slot-constrained airports. Vehicle costs per aircraft and leg and airport costs per aircraft are assumed to be known, as well as the average revenue per leg. The unconstrained forecast demand for non-stop and connecting itineraries is also known. This demand can be satisfied in non-stop and one-stop itineraries, the latter made either by the airline or by partner airlines in one of the flight legs. Market share is a function of the flight frequency in the market, assuming that the airline infers what the competitors will do. In each airport the number of slots that the airline can use are known – they can be fixed in time or vary within a given time period. Finally, the arrival distribution pattern per airport and hour, which is related with the "level of congestion", is known for the last season. It is important to underline that this is a short-term model, i.e. these are tactical decisions for the next season based on recent data from airports and airlines and from short-term forecasts.

3 Optimization model

Consider the following notation:

Sets: $A = \{1, …, A\}$ - set of airports; $P = \{1, …, P\}$ - set of O/D travel demand periods; $T = \{1, …, T\}$ - set of slot time windows; $F = \{1, …, F\}$ - set of aircraft types. Two sub-sets are needed: $A_0 = \{1, …, A_0\}$ - set of hubs airports; $T_P = \{1, …, T_P\}$ - set of slot time windows belonging to each demand period p.

Parameters: r^N_{jk} - average revenue for non-stop itineraries on market jk (\$ per passenger); r^C_{jk} - average revenue for connected itineraries on market jk (\$ per passenger); c_{Vjkf} - vehicle cost for an aircraft of type f

on flight leg jk ($ per flight); c_{Ajf} - airport cost for an aircraft of type f in airport j ($ per flight); α^N_{jkp} – recapture rate on market jk for demand period p on non-stop itineraries; α^C_{jkp} – recapture rate on market jk for demand period p on connected itineraries*;* c_{Df} – delay cost for an aircraft of type f ($/slot time window); $p_{jt,t'-t}$ – probability of a flight set to arrive on airport j on time period t being delayed more than $(t'-t)$ slot time windows; $c_T{}^{MIN}{}_j$ – minimum connection time for passengers on airport j (measured in slot time windows); c_P - average cost for each disrupted passenger ($/pax); t_{jk} - travel time without delay between airports j and k; s_{Ajt} - available slots on airport j in slot time window t; n_f - number of aircraft of type f ; n_f - number of aircraft of type f; d^N_{jkp} - demand for non-stop itineraries on market jk in period p; M_{jk} – airline market share on market jk; d^C_{jkp} - demand for connected itineraries on market jk in period p; s_{Vf} - capacity of an aircraft of type f;

<u>Decision Variables</u>: q^N_{jkt} - passengers that fly non-stop between airports j and k in time window t; q^{C1}_{jhkt} - passengers on itinerary j-h-k that fly from airport j to hub h taking off from j in time window t; q^{C2}_{jhkt} - passengers on itinerary j-h-k that fly from hub h to airport k taking off from h in time window t; x_{jkft} - number of flights by aircraft type f on leg jk that take off in slot time window t; z^N_{jkp} - spill passengers on non-stop itineraries on market jk in period p; z^C_{jkp} - spill passengers on connected itineraries on market jk in period p; $w_{jhktt'}$ - waiting passengers at airport h on itinerary j-h-k that are set to arrive on airport h in time period t and depart from airport h on time period t'; y_{jft} - number of aircraft of type f that are ready to take off from airport j in slot time window t.

Using the notation above, an airline network design problem can be represented with the following optimization model:

$$
\begin{aligned}
\max \Pi = &\sum_{j,k \in A} \sum_{t \in T} r^N_{jk} \times q^N_{jkt} + \sum_{j,h,k \in A} \sum_{t \in T} r^C_{jk} \times q^{C1}_{jhkt} \\
&- \sum_{j,k \in A} \sum_{f \in F} \sum_{t \in T} \left(c_{Vjkf} \times x_{jkft} + c_{Ajf} \times x_{kjft} \right) \\
&- \sum_{j,k \in A} \sum_{p \in P} r^N_{jk} \times \left(1 - \alpha^N_{jkp}\right) z^N_{jkp} - \sum_{j,h,k \in A} \sum_{p \in P} r^C_{jhk} \times \left(1 - \alpha^C_{jhkp}\right) z^C_{jhkp} \\
&- \sum_{j,k \in A} \sum_{f \in F} \sum_{t \in T} \sum_{t' \in T : t' > t} c_{Df} \times p_{jt,t'-t}\left(t'-t\right) \times x_{jkft} \\
&- \sum_{j,h,k \in A} \sum_{t \in T} \sum_{t' \in T : t' > t + c_T{}^{MIN}{}_j} c_P \times w_{jhktt'} \times p_{jt,t'-t-c_T{}^{MIN}{}_j}\left(t'-t-c_T{}^{MIN}{}_j\right)
\end{aligned}
\tag{1}
$$

$$
\sum_{k \in A} \sum_{f \in F} x_{jkft} + x_{kjf,t-t_{kj}} \leq s_{Ajt} , \forall j \in A, t \in T
\tag{2}
$$

$$
\sum_{k \in A} \sum_{t \in T} x_{jkft} - x_{kjft} = 0, \forall j \in A, f \in F
\tag{3}
$$

$$
\sum_{j \in A} y_{jft} + \sum_{j,k \in A} x_{jkft} + \sum_{j,k \in A} \sum_{r \in T : r > t, r < t + t_{jk}} x_{jkfr} = n_f, \forall f \in F, t \in T
\tag{4}
$$

$$
\sum_{k \in A} x_{jkft} \leq \sum_{k \in A} x_{kjf,t-t_{kj}} + y_{jf,t-1}, \forall j \in A, f \in F, t \in T
\tag{5}
$$

$$
\sum_{k \in A} x_{jkf1} \leq y_{jfT} + \sum_{k \in A} x_{kjf,T-t_{kj}}, \forall j \in A, f \in F, t \in T
\tag{6}
$$

$$d_{jkp}^N \times M_{jk} + \alpha_{jkp}^N \times z_{jk,p-1}^N - z_{jkp}^N = \sum_{t \in T_P} q_{jkt}^N, \forall j,k \in A, p \in P \tag{7}$$

$$d_{jkp}^C \times M_{jk} + \alpha_{jkp}^C \times z_{jk,p-1}^C - z_{jkp}^C = \sum_{h \in A_o, t \in T_P} \sum q_{jhkt}^{C1} + \sum_{h \in A', t \in T_P} \sum q_{jhkt}^{C2}, \forall j,k \in A, p \in P \tag{8}$$

$$q_{jhkt}^{C2} \le \sum_{t' \in T : t' < t + t_{jh}} q_{jhkt'}^{C1} - \sum_{t'' \in T : t'' < t} q_{jhkt''}^{C2}, \forall j,k \in A, h \in A_0, t \in T \tag{9}$$

$$\sum_{f \in F} x_{jkft} \times s_{V_f} \ge \sum_{h \in A} (q_{jkht}^{C1} + q_{hjkt}^{C2}) + q_{jkt}^N, \forall j,k \in A, t \in T \tag{10}$$

$$\sum_{t' \in T : t' < t - c_{T_j}^{MIN}} w_{jhkt't} = q_{jhkt}^{C2}, \forall j,h,k \in A, t \in T \tag{11}$$

$$\sum_{t' \in T : t' > t + c_{T_j}^{MIN}} w_{jhktt'} = q_{jhkt}^{C1}, \forall j,h,k \in A, t \in T \tag{12}$$

The objective function (1) of this model expresses the maximization of the operational profit (Π) of the airline. The revenues are average values for non-stop and connecting itineraries. The costs considered are: vehicle costs (e.g., fuel, crew, maintenance) airport costs (e.g., landing fees), spill costs, and delay costs. Spill costs are the loss in revenue due to not been able to satisfy the potential demand. Aircraft delay costs are the direct costs to the airline of having an aircraft delayed – fuel and crew costs and possible costs at the airport. Mathematically, the aircraft delay cost is equal to the average unit aircraft delay cost multiplied by the probability of a flight being delayed more than (t'-t) time windows. Passenger disruption costs are the costs to the airline of having a passenger that misses a connection due to delays in a previous flight-leg. It is defined by the multiplication of the average unit cost of a disrupted passenger by the probability of having a delay higher than the connection time and by the number of passengers in that itinerary. Both congestion costs (aircraft and passengers) assume that the delay patterns at an airport follows a probability function which depends on the airport and period of day. The probability function is derived from historical data.

Twelve sets of constraints are included. Capacity constraints (2) restrict the use of an airport to the existing slots. Balance constraints (3) ensure that at the end of the time plan the number of take-offs and landings is equal per aircraft type and airport. Availability constraints (4) limit the use of aircraft to the existing fleet. Continuity constraints (5, 6) guarantee aircraft continuity in each airport, time period, and aircraft type. Demand equation (7) specifies for non-stop passengers whether the airline demand for each market is either satisfied or spilled. The non-stop demand is the number of passengers willing to travel in the period multiplied by the airline market share plus the passengers that were transferred from the previous period. Constraints (8) play a similar role for connecting passengers, while taking into account whether both legs are flown by the airline or one of them is flown by a partner airline. In both demand equations (7 and 8), the market share (*M*) is a piecewise linear approximation of the frequency share of the airline, assuming that rival airlines offer a known frequency. In connecting itineraries with both legs flown by the same airline the passengers in the first and second flight-legs must be the same – constraint (9). Constraints (10) guarantee that in each flight the number of seats must be higher than the

passengers assigned to the flight. Constraints (11) and (12) define the number of waiting passengers in the hub airport of each connecting and their waiting time.

4 Model application

The model was applied to TAP Portugal, an airline that flies non-stop to more than 60 airports, half of them with a weekly frequency of at least 7 flights (the ones used to test the model). Preliminary results show that the profit can be increased up to 5% by including the delay considerations and the market share approximation in the model. The results show that the main TAP Portugal hub (LIS) gains flight frequency from the secondary hub (OPO) and that domestic frequency is decreased from Lisbon to Azores and Faro. The reduction of flights from OPO is accompanied with an increased in frequency to that markets from Lisbon, an airport in which competition is higher (Figure 1). It also noteworthy that
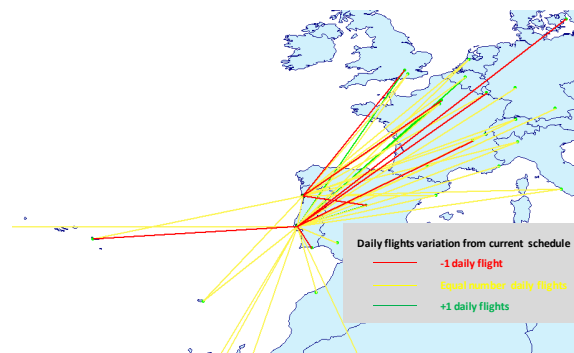
Figure 1 - Changes in daily non-stop frequency per flight-leg from current schedule

daily flight frequencies varies at most 1 flight/day. The percentage of delay costs in the total costs is between 10% and 13% which is a little higher than usual because it includes possible costs with passenger disruption. The passenger waiting time is increased around 10% to around 100 minutes but expected costs due to disruption can be reduced by 30%. The average number of feasible connections per flight is 4 and the number of flights without any feasible connection diminishes. In transcontinental flights this number can go up to 29, which reveals the importance of considering passenger disruption. For hub-and-spoke networks with few hubs (TAP Portugal case), instances with 30 airports, 100 legs, and 300 markets can be solved with a top-quality optimizer in around 3 minutes when slots are known, and in 5 minutes when they can be chosen (within a given maximum number of slots per day in each airport).

References

[1] Air Transport Association, *Economic Report* (2009).

[2] Donohue, D. and Sherry, L., U.S. Airline Passenger Trip Delay Report 2007, Center for Air Transportation Systems Research (2007).

[3] Lohatepanont, M. and C. Barnhart,"Airline Schedule Planning: Integrated Models and Algorithms for Schedule Design and Fleet Assignment", Transportation Science 38, 19-32 (2004).

A Strategy for Adaptive Traffic Signal Control with Emphasis on Offset Optimization

Tobias Pohlmann (corresponding author)

Institute of Transport and Urban Planning

Technische Universität Braunschweig, Germany

Email: t.pohlmann@tu-braunschweig.de

Bernhard Friedrich

Institute of Transport and Urban Planning

Technische Universität Braunschweig, Germany

1 Introduction

Different Adaptive Traffic Control Strategies (ATCS) for traffic signal control in urban networks exist. Well-known strategies are SCOOT and SCATS. An example of a more recent strategy is TUC. In Germany, BALANCE and MOTION are used in several cities. The improvement of traffic modeling techniques and increasing computing power promote enhancement and further development of such sophisticated ATCS systems.

This abstract sketches the conceptual design of a newly developed ATCS with emphasis on three alternative algorithms for offset optimization. The performance of each algorithm has been assessed in a comprehensive microsimulation study. Some condensed results are presented as well.

2 Conceptual Design

The strategy consists of different modules. It is designed for use in interconnected urban sub-networks containing several signalized intersections. The strategy optimizes signal plans and coordination patterns for consecutive time intervals of 15 minutes. These signal plans are used as continuously updated fixed time plans. It is also conceivable to use them as framework plans in traffic responsive controllers. The following modules are executed every 15 minutes:

Forecasting of detector counts and subsequent **OD demand, route and link volume estimation:** A forecasting technique using current and reference space-time-patterns of detector counts is used, followed by an iterative procedure of traffic assignment and OD matrix estimation in order to estimate traffic flows on all links and routes in the network for the next optimization interval. A detailed description can be found in [1], [2].

Cycle length and green split optimization: While other research projects tried to use heuristics to optimize the whole range of signal plan settings (e.g. [3]), analytical methods have been

used in this work where possible. Therefore cycle length and green splits are adapted to the forecasted traffic demand by using either the Webster formula or a saturation based approach. A common cycle length is chosen that is dictated by the most heavily loaded intersection.

Strictly speaking, this approach only applies to undersaturated or nearly saturated conditions. A strategy for oversaturation has not yet been implemented into the prototypical ATCS.

Offset optimization: Emphasis of the ATCS is on the offset optimization. While choosing offsets for arterials might be less difficult, it is more complex in interconnected networks. In a network with n intersections and a common cycle length t_c the solution space comprising all possible offset combinations has a size of t_c^{n-1}. To solve this problem, an offline method has been presented in [4]. It has now been transformed into an online application. Offsets are optimized using Genetic Algorithms in combination with the Cell Transmission Model (CTM) [5], [6]. This allows assessing hundreds of different offset combinations within only a few minutes. The CTM has been extended in such a way that even complex intersections and permitted left turning movements can be modeled. It is used to calculate the fitness of a solution, i.e. the total delay induced by a certain offset combination given the forecasted demand. The effects of necessary transitions from the previous to the new coordination pattern are considered as well. They may have a detrimental effect on otherwise possibly good solutions (cp. [7]).

Two variations of Genetic Algorithms have been tested. The Parallel Genetic Algorithm (PGA) optimizes offsets of all intersections simultaneously, whereas the Serial Genetic Algorithm (SGA) starts with optimizing only the offsets of the intersections along the heaviest loaded route, followed by those along the second heaviest loaded route and so on until all intersections have been considered. A third algorithm called Sequential Enumeration (SE) has also been tested. All offsets are set to zero in the first place. Then, a complete enumeration and evaluation of all possible offsets between zero and t_c-1 is done successively for every single intersection in the order of decreasing traffic load on the routes. The best offset found for the current intersection is kept before proceeding to the next intersection. After all intersections have been considered, the whole process is repeated again and again until no further improvement is achieved.

3 Evaluation

Several microsimulation studies employing Aimsun NG have been conducted to assess the performance of the ATCS. In the first case (whose condensed results are presented below) traffic demands of all optimization intervals have been assumed to be known exactly (i.e. the real demands fed to the system during simulation have been used for optimization). This approach allows assessing the potential of the plain optimization algorithms while neglecting the imprecision of traffic demand estimation.

A real sub-network in the city of Hanover, Germany, with 10 signalized intersections has been modeled. Figure 1 shows the network in Aimsun and its CTM representation. Varying traffic demand patterns for 56 consecutive time intervals of 15 minutes have been used with two peak hours starting at 8am and 5pm respectively.
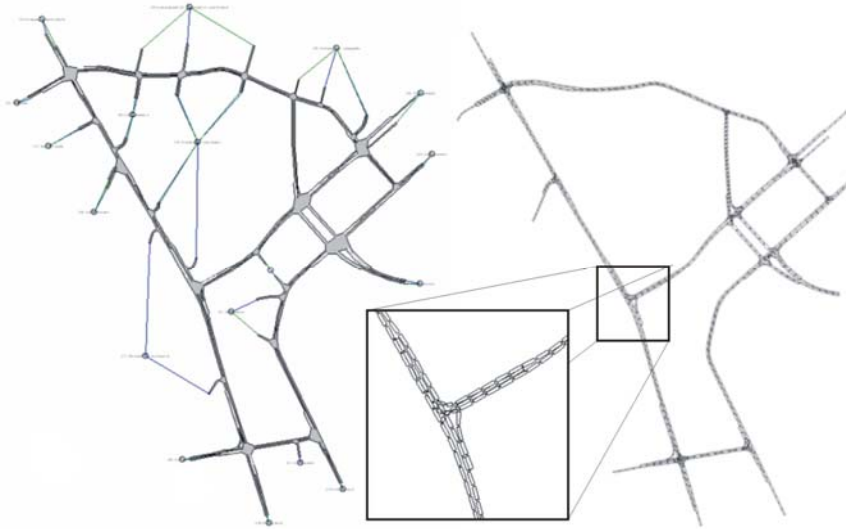
Figure 1: Test network modeled in Aimsun NG (left) and as Cell Transmission Model (right)

Figure 2 (left) shows the evolution of fitness for all three offset optimization algorithms for one sample optimization interval during peak hour. Since the two GA variants are stochastic algorithms, their respective graphs show the averages of six different optimization runs. Considering all six runs, the final fitness varied in a range of 725 veh·s (PGA) and of 547 veh·s (SGA) respectively. SE as a deterministic algorithm always produces exactly the same result given the same demand. It can be seen that both GA find an equally good and stable solution after 150 seconds which makes the ATCS real-time capable. SE finds a slightly better solution in even shorter time while a clear benefit of the GA had been expected in the first place because of its capability to use the whole solution space.
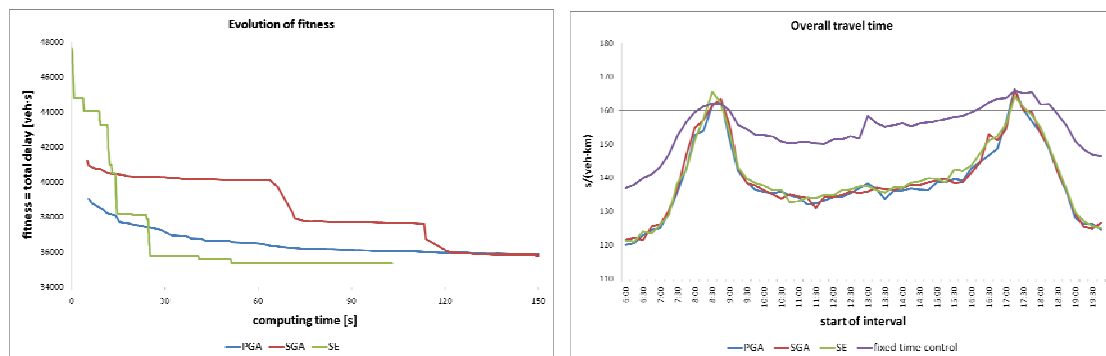
Figure 2: Comparison of offset optimization using GA and sequential enumeration

Figure 2 (right) shows how the signal plans calculated by the different algorithms affect the results of the simulation. The figure comprises the whole duration of the simulation, i.e. the

aforementioned optimization has been executed 56 times for each algorithm. All three algorithms achieve a comparable performance with only little statistically significant differences. Note that even though figure 2 (left) suggests that SE performs best, it reflects only one sample interval. When more intervals are considered (figure 2, right), the performance of all three algorithms is more or less the same on average. Furthermore, the fitness of a solution is expressed in total delay [veh·s] whereas the results during simulation are given in average travel times [s/(veh·km)] which makes differences even smaller.

As a reference, a TRANSYT 7F optimized fixed time control with two signal plans designed for the morning and afternoon peak hour has been used. The results are shown in figure 2 (right) as well. Even though it would be preferable to compare the strategy to other ATCS such as SCOOT, SCATS or TUC, this is not possible because their code is not at the authors' disposal.

More detailed findings do not only include overall travel times for the whole system but also for individual routes. Furthermore, a second test network has been used. However, its presentation would exceed the limited scope of this extended abstract.

References

[1] T. Pohlmann, B. Friedrich. "Combined Short-Term Forecasting and Traffic Demand Estimation for Online Control Strategies". Proc. XIII Euro Working Group on Transportation Meeting, Padua, Italy, 23-25 September 2009

[2] T. Pohlmann, B. Friedrich, "Short Term Estimation of OD Matrices and Traffic Streams in Urban Networks", Proc. mobil.TUM 2009 International Scientific Conference on Mobility and Transport, Munich, Germany, May 12-13, 2009.

[3] R. Braun, C. Kemper, F. Weichenmeier, "TRAVOLUTION – Adaptive Urban Traffic Signal Control with an Evolutionary Algorithm", Proc. 4th International Symposium Networks for Mobility, Stuttgart, Germany, September 25-26, 2008.

[4] E. Almasri, "A New Offset Optimization Method for Signalized Road Networks", Dissertation at the Institute of Transport, Road Engineering and Planning, Leibniz Universität Hannover, Hannover, Germany, 2006

[5] C. Daganzo, "The Cell Transmission Model: A dynamic Representation of Highway Traffic consistent with the Hydrodynamic Theory", *Transportation Research B* 28 (4), 269-287 (1994)

[6] C. Daganzo, "The Cell Transmission Model, Part II: Network Traffic", *Transportation Research B* 29 (2), 79-93 (1995)

[7] T. Pohlmann, B. Friedrich, "Traffic Signal Transition in Coordinated Meshed Networks", Proc. TRB 89th Annual Meeting, Washington D.C., USA, January 10-14, 2010

Railway Crew Rescheduling under Uncertainty

Daniel Potthoff

Econometric Institute

Erasmus University Rotterdam, Burg. Oudlaan 50, Rotterdam, The Netherlands

Email: potthoff@ese.eur.nl

Dennis Huisman   **Albert P.M. Wagelmans**

Econometric Institute   Econometric Institute

Erasmus University Rotterdam   Erasmus University Rotterdam

and

Department of Logistics

Netherlands Railways

1 Introduction

Effective disruption management is a key to a good operational performance for passenger railway companies. Within the disruption management process (see [2] for a detailed discussion), the ability to reschedule the main resources rolling stock and crew is crucial. In recent years Operations Research based approaches for these problems have been proposed by the scientific community (see [3], [5], [4], and [6]). The models for crew rescheduling do assume that an accurate estimate about the duration of the disruption is available at the time the rescheduling is done. The same holds for models developed for crew rescheduling in the airline industry (see [1] for a recent literature review). However, this assumption is not realistic.

Let us consider a small illustrative example taking place in the north of the Netherlands. Due to a broken power supply, no train traffic is possible between Hoogeveen (Hgv) and Beilen (Bl) from 7:10 on. It is estimated that the repair works will last between 3 and 4 hours. The timetable will be updated according to a pattern described by an emergency scenario. In this case, the trains of the train lines 500, 700 and 9100, operated between Zwolle (Zl) and Groningen (Gn), will be turned at intermediate stations. In Figure 1 we show how the timetable between Zwolle (Zl) and Groningen (Gn) would be updated. Since the repair works will take at least 3 hours the turning pattern will be applied for sure for three southbound and three northbound trains of each the three involved train lines. For the trains in the fourth hour after the start of the disruption, it is
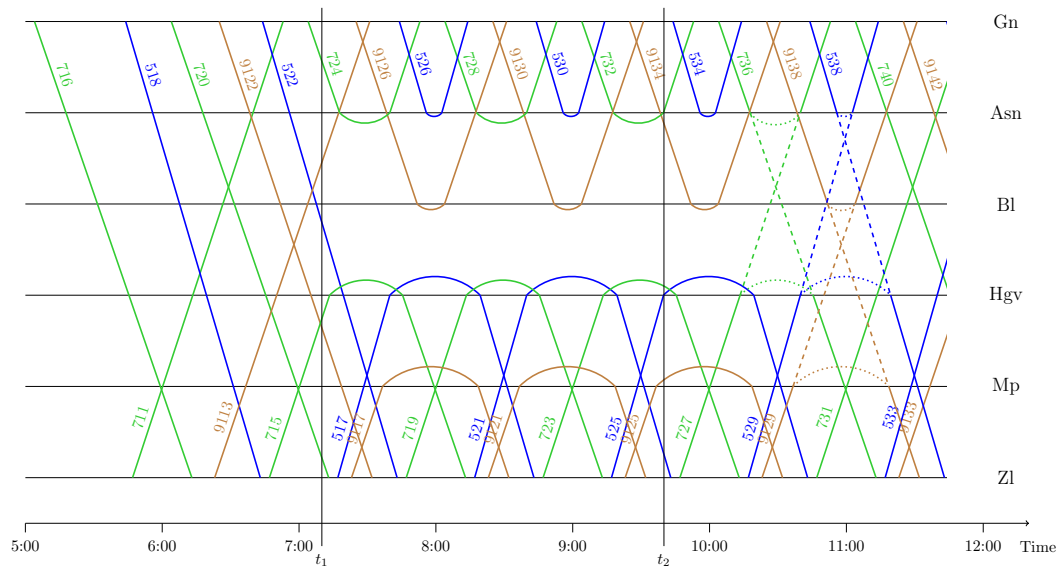
Figure 1: Time space diagram showing how the timetable between Groningen (Gn) and Zwolle (Zl) would be updated, if the route between Beilen (Bl) and Hoogeveen (Hgv) is blocked.

uncertain if the trains will take their normal routes (dashed lines in Figure 1) or if they will be turned as well (dotted lines in Figure 1).

Current crew rescheduling approaches would deal with this situation as follows. At time point t_1 it is estimated that the blockage will be over by 10:10. Therefore, the modified timetable that is given as input to the crew rescheduling assumes that the trains 727, 736, 529, 538, 9129 and 9138 can run between Beilen and Hoogeveen as planned. However, it might happen that at time point t_2, 9:40 in the example, new information saying the route will be blocked until 11:10, becomes available. This means that the timetable has to be updated again and that the trains 727, 736, 529, 538, 9129 and 9138 must also be turned at intermediate stations. At t_2 the crew schedule would be rescheduled again given the new information.

If at t_1 the uncertainty about the duration of the disruption, and therefore the uncertainty about the timetable that will be operated, is not taken into account, above procedure will correspond to a wait-and-see approach. Two research questions arise from that.

1. How does wait-and-see perform compared to the case when the perfect information would be available at time t_1?

2. Could the uncertainty be taken into account at t_1 such that the resulting rescheduling problem stays computationally tractable?

In this paper we try to answer both questions. First, we are going to present a case study presenting the results of applying the wait-and-see approach to some rescheduling instances of Netherlands Railways. Second, we present a quasi robust approach to crew rescheduling under uncertainty.

2 Problem description

We consider crew rescheduling under uncertainty as a two stage process. In stage one at t_1 an estimate of the duration h_1 of the disruption is know. In the case of a malfunctioning switch for example, this estimate could be be based on the initial judgment of a repair crew. Based on the estimated duration, the original timetable T_0 will be adjusted according to the unavailable infrastructure. The result is an adjusted timetable T_1. Then the crews are rescheduled according to this adjusted timetable. Later, at time t_2 it becomes clear when the infrastructure can definitely be used again. Often this is later than the expected time $t_1 + h_1$. Usually this means that timetable T_1 can not be operated and instead another adjusted timetable T_2 will be operated. This could mean that the crews need to be rescheduled again according to T_2.

Assume that the timetable that will be operated in the end must be one of small number of possibilities. We refer to these possibilities as scenarios $s_0 \ldots s_n$, where s_0 corresponds to the most likely scenario which would be used for the first rescheduling at t_1 in a wait-and-see approach. The crew rescheduling problem under uncertainty can be stated as follows. Given a most likely scenario s_0 and a set of alternative scenarios $s_1 \ldots s_n$, find a new crew schedule valid for s_0 such that the cost of this schedule and the cost for the additional rescheduling at t_2 are minimized.

The crew rescheduling problem can be written as a set covering problem with side-constraints. The decision variables in this model are feasible completions of the planned original duties. The set covering constraints correspond to the train driving activities of the modified timetable. Moreover, the side-constraints ensure that exactly one feasible completion is selected for each original duty.

3 A quasi robust approach

The idea behind this approach is to use feasible completions that are quasi robust against all scenarios and in this way minimize the effort that is needed at time point t_2 if a scenario other then s_0 occurs. We will give an informal definition of quasi robust feasible completions. A feasible completion is called quasi robust if all tasks corresponding to a train activity that are covered in scenario s_0 can still be covered in every other scenario. In the rescheduling at time point t_1 we only allow quasi robust feasible completions. A resulting crew schedule has the property that, if at time point t_2 it becomes certain that the timetable will be operated according to scenario s_1, rescheduling can be done using already know alternatives for the feasible completions that are

affected by the timetable changes. Moreover, in the assumed case when the differences between the timetables for s_0 and s_1 are well defined, all tasks can be covered by the new crew schedule.

We will solve the quasi robust crew rescheduling problem with an adapted version of the column generation based heuristic proposed by [4]. We modify the pricing problems which are modeled as shortest path problems with resource constraints on directed acyclic graphs. We use additional resources in order to only generate quasi robust feasible completions. The bounds of the resource windows for these additional resources are determined in a preprocessing step. We will present computational results for instances provided by Netherlands Railways. Moreover, we will compare the results to the wait-and-see approach.

References

[1] J. Clausen, A. Larsen, J. Larsen, N.J. Rezanova, "Disruption management in the airline industry - Concepts, models and methods", *Computers & Operations Research* 37, 809-821 (2010).

[2] J. Jespersen-Groth, D. Potthoff, J. Clausen, D. Huisman, L.G. Kroon, G. Maróti, and M. Ny-have Nielsen, "Disruption Management in Passenger Railway Transportation", *Robust and Online Large-Scale Optimization*, R.K. Ahuja, R.H. Möhring and C.D. Zaroliagis (eds), 399-421, Springer, Berlin, 2009.

[3] L.K. Nielsen and G. Maróti, "Disruption Management of Rolling Stock in Passenger Railway Transportation", Working Paper ERS-2009-046-LIS, Erasmus Research Institute of Management (ERIM), Erasmus University Rotterdam, Rotterdam, The Netherlands, 2009.

[4] D. Potthoff, D. Huisman and G. Desaulniers, "Column Generation with Dynamic Duty Selection for Railway Crew Rescheduling", Working Paper EI 2008-28, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 2008.

[5] N.J. Rezanova and D.M. Ryan, "Train driver recovery problem - A set partitioning based and solution method", *Computers & Operations Research* 37, 845-856 (2010).

[6] L.P. Veelenturf, D. Potthoff, D. Huisman and L.G. Kroon, "Railway Crew Rescheduling with Retiming", Working Paper EI 2009-24, Econometric Institute, Erasmus University Rotterdam, Rotterdam, The Netherlands, 2009.

A Column Generation based Approach for the Dynamic Vehicle Routing and Scheduling Problem with Soft Time Windows

Ali Gul Qureshi

Department of Urban Management

Kyoto University

Eiichi Taniguchi

Department of Urban Management

Kyoto University

Tadashi Yamada

Department of Urban Management

Kyoto University

Ali Gul Qureshi

Department of Urban Management

Kyoto University, Katsrura Campus, Kyoto, Japan

Email: aligul@kiban.kuciv.kyoto-u.ac.jp

1 Introduction

Transportation accounts for considerable cost in the supply chain. The Vehicle Routing and scheduling Problem with Time Windows (VRPTW) is a useful tool employed by logistics firms to optimize their operations. The classical VRPTW is defined for static input values such as fixed customer locations and static travel time. However, the traffic conditions in urban areas change with time due to varying congestion levels and incidents resulting in varrying travel time on the infrastructure links. With the introduction of the Intelligent Transportation Systems (ITS) such as the Vehicle Information and Communication System (VICS) in Japan, it is possible to collect and store such dynamic travel times on a link. As far as the logistics is concerned, changes in the travel time may affect the distribution or pick-up routes of the delivery vehicles resulting in unexpected long delays if the routing is fixed and based on a static value of travel time (such as the average travel time).

In the VRPTW related literature, a routing system is defined as Dynamic Vehicle Routing and scheduling Problem with Time Windows (D-VRPTW), in which, complete or a part of input information (such as number and location of customers or travel time on arcs) is not available to the decision maker at the start but it is revealed during the scheduling horizon (day of operation), and if the decision maker *reacts* to this new information by evoking some sort of *re-optimization* mechanism [1].

There exists a large body of literature on the dynamic customers case of the D-VRPTW but the dynamic travel times-related literature is rather scant. In fact, only two references could be found dealing with the dynamic travel times, viz. Taniguchi and Shimamoto [2], and Flieschmann et al [3]; both considering the Dynamic-Vehicle Routing and scheduling Problem with Soft Time Windows (D-VRPSTW). Taniguchi and Shimamoto [2] have used a macro-simulation scheme to generate the dynamic travel time data for a theoretical test network, whereas, Flieschmann et al [3] have used the data from an ITS implemented in Berlin, Germany, named as LISB, which provides the travel time data on links for every 5 minutes slot. However, both of these research works have adopted heuristics approaches to solve the D-VRPSTW. The heuristic techniques are sometimes faster and easily implemented than exact solutions, yet they do not guarantee to identify the exact solution or state how close to the exact solution a particular feasible solution is [4]. Therefore, this research proposes a column generation based exact solution approach for the D-VRPSTW with dynamic travel times to fill the existing research gap. The exact approach can be used for small to medium instances (25 to 50 customers) as well as for the evaluation and calibration of the heuristics approaches.

2 General Framework

The proposed Dynamic Vehicle Routing and scheduling Problem with Soft Time Windows (D-VRPSTW), only considers the travel time uncertainty, therefore, all remaining information such as customers' locations and demands are assumed to be known and fixed. The soft time windows enable delivery after the close of customer specified time windows with an associated late arrival penalty, however the waiting is allowed without any penalty. The D-VRPSTW would be modeled using the *rolling horizon* scheme, in which the complete scheduling horizon is divided into various time slots, each representing a time-based scenario. Thus initially, it can be defined on a complete Graph G_{T_1} for the first time slot (T_1), which consists of all customers with vehicles (k_i) stationed at the central depot (as shown in Figure1(a)). The routes for the time slot T_1, would be planned as per the average travel times. No *divergence* is allowed once a vehicle leaves to visit a customer, i.e. the first customer on the route of an en-route vehicle is fixed. The time slots are marked with *vehicle-based events*, which means a new time slot is initiated as soon as any of the vehicle reaches the first customer on its route.

With no diversion allowed, the locations of all vehicles are forecasted and the Graph is updated (G_{T_2} (Figure 1(b))) showing vehicle locations along with all customers except those which are either serviced or are the first customers of an en-route vehicle. The arcs in G_{T_2} contain the updated

travel times, therefore, routes planned in T_2 would be based on these updated travel time values. It may be noted that, nodes other than depot, which contain a vehicle resource, only have outbound links. A vehicle can abandon its remaining planned route and head back to the depot as well.

This procedures goes on till the last customer is serviced (as shown by G_{T_3} in Figure 1(c)). As the customers assigned to a particular vehicle may get changed at any of the route revision epoh (at the change of time slot), the proposed D-VRPSTW is suitable to either model a delivery system of single commodity such as heating oil, gasoline etc., or it can model any pick-up service.

| (a) | (b) | (c) |
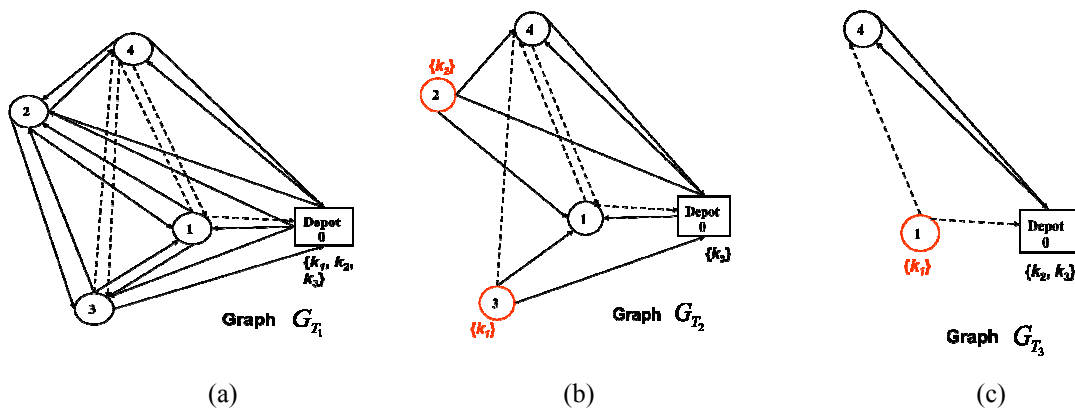
Figure 1. Representation of the dynamism of D-VRPSTW in rolling horizon scheme

3 Methodology

The D-VRPSTW would be solved by extending a similar column generation approach as developed for the static VRPSTW [5]. Precisely, using the Dantzig-Wolfe decomposition, the D-VRPSTW would be decomposed into a set partitioning linear master problem, and in an Elementary Shortest Path Problem with Resource Constraint and Late Arrival Penalties (ESPPRCLAP) [6]. However, as the G_{T_2} (Figure 1(b)) shows, the proposed framework leads to a Multi depot-Dynamic Vehicle Routing and scheduling Problem with Soft Time Windows (MD-VRPSTW) as each node with a vehicle resource would be considered as a virtual depot. Therefore, the proposed column generation based exact approach must incorporate the multi depot aspect as well. It can be modeled by formulating a separate ESPPRCLAP for each of the virtual depot. At each column generation iteration, the set partitioning master problem would receive columns (routes/paths of negative reduced costs) from all subproblems and then it would optimize the complete problem by selecting the best set of routes covering the demands of all the customers at hand. The dual variables' values (*prices*) would be obtained as a byproduct, which will be used to define new reduced cost matrices for each of the subproblems to generated new promising paths/columns, and the whole process is repeated till the subproblems fail to provide a negative reduced cost column. At this stage if the solution of the master problem LP is not integer a branch and bound tree would be explored. Therefore, at every route revision epoh, the column generation

algorithm would be embedded in a branch and price algorithm. In order to track and keep the total number of vehicles in the system, a new branching scheme would be needed.

4 Results and Discusions

Changing traffic conditions and incidents cause the travel time variation in urban areas. If the operations of a logistics company is planned, by ignoring this reality, it may affect the distribution or pick-up routes resulting in unexpected long delays. On the contrary, dynamic vehicle routing and scheduling can result in better route selection, avoiding congested roads due to incidents or any other reason. This would help logistics firms in reducing their distribution/prick-up costs as well as in improving their reliability because the delays and related late arrival penalties would also be minimized. Furthermore, the amount and distribution of the environmental emissions would also be changed if the delivery/pick-up vehicles would follow updated routes, diverting from pre-defined routes containing congested roads.

Results obtained on benchmark and reallife logistics instances would be included in the final submission and would be presented at the conference.

References

[1] H.N. Psaraftis, "Dynamic Vehicle Routing: Status and Prospects", *Annals of Operational Research* 61, 143-164 (1995).

[2] E. Taniguchi and H. Shimamoto, "Intelligent Transportation System based Dynamic Vehicle Routing and Scheduling with Variable Travel Time", *Transportation Research Part C* 12, 235-250 (2004).

[3] B. Fleischmann, S. Gnutzmann and E. Sandvoss, "Dynamic Vehicle Routing based on Online Traffic Information", *Transportation Science* 38, 420-433 (2004).

[4] R.G. Thompson and J.H.R.Van Duin, "Vehicle Routing and Scheduling", in *Innovations in Freight Transportation*, E. Taniguchi and R.G. Thompson (eds), 47-63, WIT Press, Southampton, 2003.

[5] A.G. Qureshi, E. Taniguchi and T. Yamada, "An Exact Solution Approach for Vehicle Routing and Scheduling Problems with Soft Time Windows", *Transportation Research Part E* 45, 960-977 (2009).

[6] A.G. Qureshi, E. Taniguchi and T. Yamada, " Elementary Shortest Path Problem with Resource Constraints and Time Dependent Late Arrival Penalties", *JSCE Journal Division D* (*Doboku Gakkai Ronbunshuu D*) 63, 579-590 (2007).

Biobjective Traffic Assignment to Model Network User Behaviour in Networks with Road Tolls

Andrea Raith, Judith Y.T. Wang, Matthias Ehrgott

The University of Auckland, New Zealand

Email: {a.raith, j.wang, m.ehrgott}@auckland.ac.nz

1   Introduction

Traffic assignment is a key component in the conventional four-stage transport planning model. It models travel behaviour in terms of route choice. Being able to model route choice decisions correctly is essential to accurately forecast travel demand and most importantly to enable the correct assessment of the benefits/disbenefits of changes in transport policies and infrastructure developments. The presence of road tolls influences the route choice of travellers. Instead of considering a linear combination of travel time and (toll) cost, we explicitly distinguish time and toll cost as separate route choice objectives. This leads to the concept of biobjective traffic assignment (BTA), where we assume that "[...] traffic arranges itself in such a way that no individual trip maker can improve either his/her toll or travel time or both without worsening the other component by unilaterally switching routes", see [1].

2   Literature

One frequent assumption made in the literature on traffic assignment is that travellers choose their route with the aim of minimising a linear combination of time and (toll) cost, where time is multiplied by a so-called value of time (VOT) e.g. [2]. Others assume that there exist different classes of network users each with their individual VOT value, e.g. [3]. Others assume that VOT follows a distribution, e.g. [4, 5, 6]. All these approaches have in common that a linear combination is considered rather than treating the two objectives separately. However, there do exist other routes that are efficient in the sense of the definition of BTA without being optimal for such a linear function, see Fig. 1 where z^2 is such a combination of time and cost. Assuming a distribution of VOT may assign flow to the paths corresponding to time/cost vectors z^1, z^3, z^5, z^6 but not to z^2, z^4 (the latter is optimal only for a single VOT value). We believe that a solution of BTA should
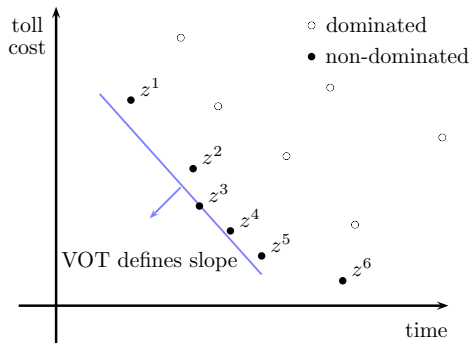
Figure 1: Time and toll cost for all paths connecting one OD pair.
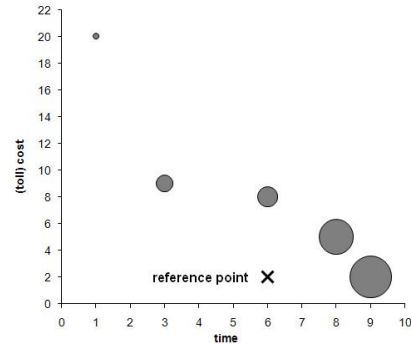
Figure 2: RPT flow shares of efficient paths given by size of corresp. non-dominated point.

permit flow on any of the *efficient* paths, which have a non-dominated time/cost vector in Fig. 1.

3 Solving Biobjective Traffic Assignment

For the standard traffic assignment problem, equivalence of the traffic assignment equilibrium problem to other mathematical problems such as optimisation and variational inequality problems, is exploited to develop solution algorithms. It is discussed in [7] that the corresponding equivalences do not hold in the bi- and multiobjective case.

Here, we present heuristic solution algorithms instead. The popular method of successive averages (MSA) can be adapted to solve BTA: The modified MSA initially assigns all travel demand along the efficient paths assuming fixed travel time and cost derived from zero flow. Efficient paths can be identified via any bi-objective shortest path algorithm, e.g. [8]. In subsequent iterations, the efficient paths are calculated and assigned flow according to fixed time and cost with respect to the current flow. In every iteration of MSA, decreasing portions of flow are re-assigned. There is great liberty in how to assign the travel demand between the different efficient paths, we will discuss a few assignment schemes in the following. The final solution of BTA of course depends on how flow is assigned in each iteration.

[1] explore two simple ways to carry out BTA: (1) to split the demand equally between the efficient paths (EQS); and (2) to split the demand based on cost per unit time savings assignment (CTS), where flow is assigned to each of the efficient paths according to the cost per time unit saved when compared with the cheapest path (z^6 in the Fig. 1) for which a distribution is given. CTS is obviously more realistic as compared with EQS.

We propose new traffic assignment schemes here. We briefly describe the different approaches, but omit the details of the algorithms in this abstract. Firstly, in reference point (RPT) assignment, a reference point is given that represents a time/cost vector to which network users are attracted.
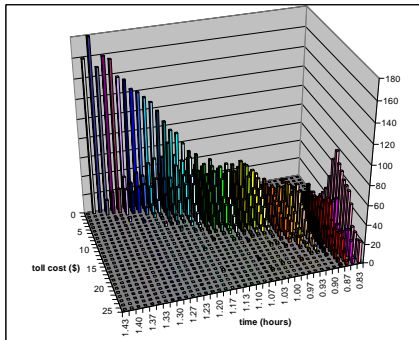
Figure 3: Possible distribution of users for ADO assignment.
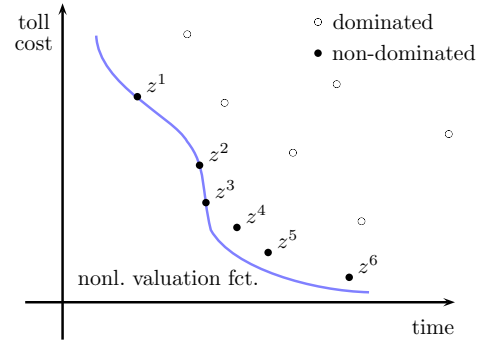
Figure 4: Nonlinear valuation function, optimal points are z^1, z^2, z^3.

The closer one of the cost vectors in Fig. 2 is situated to the reference point, the more travellers choose the corresponding path. The share of each path can be computed in various ways, Fig. 2 contains an example where the shares of efficient paths are indicated by the size of the circle marking the corresponding non-dominated time/cost vector.

Alternatively, we propose area of domination (ADO) assignment. A survey of road users can be conducted asking every user the question "How much travel time do you spend at the moment, how much would you (realistically) want to save and how much money are you prepared to pay for it?". Then every road user can be represented by their own ideal point in objective space. A road user would only select a path with cost vector dominating his/her ideal point (i.e. better or equal in both criteria). If there is no such point, users must settle for a compromise (or not travel at all) and thus we assume they choose the fastest path with lower cost than their ideal point. Fig. 3 shows the aggregated results as they might be obtained by a survey of 10000 network users.

Instead of the VOT value or VOT distribution, a non-linear valuation function may be given. According to this valuation function, one or more of the efficient paths may be optimal, see Fig. 4. In each iteration of MSA, all demand is assigned to path(s) optimal with respect to the valuation function. For a non-linear valuation function, any of the non-dominated points may be optimal, which removes the disadvantage of using a linear valuation function. In order to apply a non-linear valuation function, it is necessary to first identify the efficient paths with respect to time and toll as the problem can no longer be solved as a single-objective shortest path problem.

We finally propose a different approach to MSA, called path equilibration. For traditional traffic assignment, one solution algorithm is known as path equilibration which was first proposed in [9]. For a single objective, an initial assignment of all flow to the paths with minimal cost at flow zero, is performed. In every iteration, and for every OD pair, the current shortest path and the longest with postive flow are identified. Those two paths are the least in equilibrium. Flow is re-assigned from the longest to the shortest path until their travel times are equal or the flow

on the longest path is zero. The equilibration algorithm can be adapted to BTA: Instead of the shortest path, the efficient paths are identified, together with one or more non-efficient paths (e.g. one or more paths furthest away from the efficient paths). Again, flow is shifted from non-efficient to efficient paths.

It can be observed that a BTA equilibrium has been reached by the path equilibration algorithm when only efficient paths have positive flow. We are currently unable to confirm an equilibrium has been obtained with the MSA approach as only link flow, but not path flow, is recorded in MSA. We do believe that the solutions obtained by MSA are a good approximation of a true BTA equilibrium.

Next steps include the application of the presented methods to a realistic transport network and further development of assignment algorithms.

References

[1] J.Y.T. Wang, A. Raith and M. Ehrgott, "Tolling analysis with biobjective traffic assignment", in *Multiple Criteria Decision Making for Sustainable Energy and Transportation Systems*, M. Ehrgott, B. Naujoks, T. Stewart and J. Wallenius (eds), 117-129, Springer, 2010.

[2] Y. Sheffi, *Urban Transportation Networks: Equilibrium Analysis with Mathematical Programming Methods*, Prentice-Hall (1985).

[3] A. Nagurney, "A multiclass, multicriteria traffic network equilibrium model", *Mathematical and Computer Modelling* 32, 393-411 (2000).

[4] R.B. Dial, "A model and algorithm for multicriteria route-mode choice", *Transportation Research* 13B, 311-316 (1979).

[5] R.B. Dial, "Bicriterion traffic assignment: Basic theory and elementary algorithms", *Transportation Science* 30, 93-111 (1996).

[6] F. Leurent, "Cost versus time equilibrium over a network", *European Journal of Operational Research* 71, 205-221 (1993).

[7] A. Raith and M. Ehrgott, "On vector equilibria, vector optimisation and vector variational inequalities", Technical report, The University of Auckland, 2009.

[8] A. Raith and M. Ehrgott, "A comparison of solution strategies for biobjective shortest path problems", *Computers & Operations Research* 36, 1299-1331 (2009).

[9] S. Dafermos and F.T. Sparrow, "The traffic assignment problem for a general network", *Journal of Research of the National Bureau of Standards–B. Mathematical Sciences* 73B, 91-118 (1969).

Branch-and-Price for creating an Annual Delivery Program of Multi-Product Liquefied Natural Gas

Jørgen Glomvik Rakke, Henrik Andersson, and Marielle Christiansen

Department of Industrial Economics and Technology Management

Norwegian University of Science and Technology, Trondheim, Norway

Email: jorgen.rakke@iot.ntnu.no

Guy Desaulniers

GERAD and Department of Mathematics and Industrial Engineering

École Polytechnique, Montréal, Canada

1 Introduction

During the past decade natural gas has become an increasingly important mean to fulfill the worlds soaring demand for energy. Traditionally, natural gas was transported by pipelines, and the option of transporting the gas as liquefied natural gas (LNG) was considered economically undesirable. However, the combination of higher prices, lower production costs, and rising import demand has set the stage for increased LNG trade and more use of LNG ships as a mean to transport natural gas. In addition to the rapidly growing demand for natural gas and more flexible long-term contracts, a new spot market for LNG has surfaced in the recent years. The combination of an increased number of ships and deliveries, and the new complexity introduced by the LNG spot market, makes the distribution planning problems much harder to solve. Hence, there is a growing need for decision support systems in maritime LNG shipping.

We consider a combined ship routing and inventory management problem for one of the world's largest producers of LNG. The problem is how to manage the producer's inventory and fleet of ships to create an Annual Delivery Program (ADP) that respects the long-term contracts at lowest possible cost, while maximizing the expected revenue from spot contracts. It has some similarities to the problem solved in [2], but has a different network structure, more ships and terminals, and a longer planning horizon. For more literature in the area of maritime inventory routing see the recent survey [1]. The purpose of this paper is to describe the mathematical model for the ADP planning problem and to introduce a branch-and-price method in order to be able to solve real size

problems.

2 Problem Description

The problem addressed in this paper is to design an ADP, which is a list of scheduled voyages, where each voyage consists of the contract served, the ship used, the days of loading and unloading, and the day the ship returns to the loading port. The ADP must respect the inventory constraints on the supply side, but the producer has no inventory obligations at the delivery terminals where the LNG is re-gasified (regasification terminals). The objective of the ADP is to minimize the total cost of fulfilling long-term contracts while maximizing the expected contribution from spot sales , and abiding all physical constraints in the LNG value chain.

At the producer's storage and liquefaction facility there is a variable production of two types of LNG, namely rich LNG (RLNG) and lean LNG (LLNG). There is a limited number of berths available for loading each type of LNG onto specially constructed LNG ships. The liquefied gas it is stored in high pressure tanks to ensure that it is kept in a liquefied state throughout the voyage.

The producer's fleet of ships is heterogeneous, with capacities ranging from 126 000 to 265 000 m^3. If the producer does not have enough ships available at a given time, additional ships may be chartered in. Several factors influence the availability and use of the fleet, one being that a central depot does not exist. Hence, some ships may become available after the start of the planning horizon. The ships may also be unavailable due to certain pre-allocated activities, e.g. dry-docking for maintenance. Finally, not all ships can visit all regasification terminals due to vessel acceptance policies at the ports, and the fact that some ships are owned by one, or a group of customers, limiting them to only visit their owner's regasification terminals.

A shipload must contain either LLNG or RLNG, but a ship can carry different types of LNG on consecutive voyages without any intermediate preparations. The ship tanks are always filled to their capacities at the loading port and visits only one regasification terminal on a scheduled voyage. The cost of sailing a scheduled voyage is assumed to be dependent only on the capacity class of the ship, the duration of the voyage and the regasification terminal visited.

The producer is obliged to deliver a certain quantity of a given type of LNG to a specified regasification terminal each year according to a long-term contract. The contract may outline the monthly demand that is to be delivered, or simply state that the LNG is to be delivered "fairly evenly spread" throughout the year. In addition to serving the long-term contracts, the producer has the opportunity to sell LNG on the spot market.

3 Solution Approach

First, a MIP formulation of the ADP planning problem utilizing the underlying transportation network, the voyage characteristics, and the spread requirements of the long-term contracts will be presented. This model, called the Basic Voyage Model (BVM), is based on pre-generation of all possible scheduled voyages for all contracts and ships within the planning horizon. The planning horizon $\mathcal{T} = \{\underline{T}, \ldots, \overline{T}\}$ is discretized on days. The main variables of the BVM, denoted x_{cvt}, are binary and indicate if ship v starts a voyage on day t to the terminal associated with contract c. The demand constraints are implemented as soft constraints, meaning that you can deviate from the contracted quantity, but you have to pay a penalty. Hence, variables representing over- and under-delivery (y_{cp}^+ and y_{cp}^-) are also introduced in order to be able to penalize deviation from contracted demand in period p. In the BVM, the demand constraints are expressed as follows:

$$\sum_{v \in \mathcal{V}_c} \sum_{t \in \mathcal{T}_p} L_v^{CAP} \times x_{cvt} + y_{cp}^- - y_{cp}^+ = D_{cp} \qquad \forall c \in \mathcal{C}^{LT}, p \in \mathcal{P} \tag{1}$$

where $\mathcal{P}$ is the set of all time periods and $\mathcal{T}_p = \{\underline{T}_p, \ldots, \overline{T}_p\} \subseteq \mathcal{T}$ is the set of days in time period p. A time period can i.e. be a week, a month, or a year. $\mathcal{C}^{LT}$ is the set of long-term contracts, L_v^{CAP} is the quantity delivered by ship v, and D_{cp} is the contractual demand of contract c in time period p. Over- and under-deliveries are penalized in the objective function:

$$minimize \sum_{c \in \mathcal{C}^{LT}} \sum_{p \in \mathcal{P}} C_{cp}^{D+} \times y_{cp}^+ + \sum_{c \in \mathcal{C}^{LT}} \sum_{p \in \mathcal{P}} C_{cp}^{D-} \times y_{cp}^- + \textbf{sailing cost} - \textbf{revenue} \tag{2}$$

where C_{cp}^{D+} and C_{cp}^{D-} are the cost of over and under-delivery, respectively. One problem with the BVM is that the LP-relaxation is weak, mainly because it is always possible to deliver fractional combination of ships in order to avoid penalties. We know that this is not possible in the MIP solution, and hence it introduces a large gap between the LP-solution and the MIP-solution. On a small test instance with an optimal MIP value of -63 270 we observe that the LP-solution has 0 contractual delivery penalty, while the optimal MIP-solution has a penalty of 35 173.

In order to reduce this gap, the BVM is reformulated using delivery pattern for each contract c and each period p, and an additional set of variables (z_{cpi}) is introduced. A pattern i stipulates the number of deliveries by each specific ship to a given contract c during a period p, and the total penalty in p. The pattern does not include the starting time of the various deliveries. In the reformulation we replace (1) with

$$\sum_{i \in \mathcal{Y}_{cp}} z_{cpi} = 1 \qquad \forall c \in \mathcal{C}^{LT}, p \in \mathcal{P} \tag{3}$$

and

$$\sum_{t \in \mathcal{T}_p} x_{cvt} - \sum_{i \in \mathcal{Y}_{cp}} N_{cpi}^v \times z_{cpi} = 0 \qquad \forall c \in \mathcal{C}^{LT}, p \in \mathcal{P}, v \in \mathcal{V} \tag{4}$$

where $\mathcal{Y}_{cp}$ is the set of delivery patterns for contract c in time period p, and N_{cpi}^v is the number of deliveries made by ship v to contract c in time period p in pattern i. We also change the objective to include the pattern penalty instead of the penalties for over- and under-deliveries:

$$minimize \sum_{c \in \mathcal{C}^{LT}} \sum_{p \in \mathcal{P}} \sum_{i \in \mathcal{Y}_{cp}} C_{cpi}^Y \times z_{cpi} + \textbf{sailing cost} - \textbf{revenue} \tag{5}$$

where C_{cpi}^Y is the penalty for pattern i. In this way we force the LP-solution to also incorporate some of the penalty costs, reducing the gap to the MIP-solution. However, the number of possible patterns is far too large to be generated a priori. This makes it necessary to generate the patterns dynamically using column generation. There is one column generation subproblem per contract c and period p. Such a subproblem is an IP containing the original contractual constraints, the non-negativity constraints on the over- and under-delivery variables, and the integrality constraints on the schedule voyage variables. It does not possess the integrality property, opening up the possibility to raise the bound and, thus, reduce the gap.

4   Computational Results

Computational results for the branch-and-price-and-cut formulation will be presented and compared to the results from running the BVM using commercial optimization software and different heuristics.

References

[1] M. Christiansen and K. Fagerholt. *Encyclopedia of Optimization, second edition*, chapter Maritime inventory routing problems, pages 1947–1955. Springer US, 2009.

[2] R. Gronhaug, M. Christiansen, G. Desaulniers, and J. Desrosiers. A branch-and-price-and-cut method for a liquefied natural gas inventory routing problem. Technical Report Les Cahiers du Gerad G-2008-49, HEC Montreal, July 2008.

Risk Taking and Strategic Thinking in Route Choice: A Stated-Preference Experiment

Michael D. Razo

Department of Civil and Environmental Engineering

University of Massachusetts Amherst

Song Gao

Department of Civil and Environmental Engineering

University of Massachusetts Amherst

October 2009

1 Introduction

With Advanced Traveler Information Systems (ATIS) rapidly growing in both prevalence and capabilities, effective analysis and prediction of traffic demand requires proper accounting for the impact that real-time information has on travelers. While route choice modeling has historically relied only on a priori information about traffic conditions, the next generation of modeling must account for information that travelers receive en route to their destinations. Adaptive choice behavior, such as that explored in [1], allows for path alteration in response to information received en route. Strategic choice behavior extends this by anticipating en route information and including possible detours in the assessment of alternatives. Empirical verification of such behavior remains largely unexplored [3].

This research investigates several models for route choice in risky networks with real-time information, using stated preference data from interactive maps. The results will provide some insight into the importance of acccounting for strategic behavior.

2 Objectives

The primary objective of this research is to determine the extent to which drivers think strategically when choosing routes. "Strategic" thinking is defined as choice behavior which accounts in advance for information that will be received en route, and for any detours that might be taken based on such information. Strategic drivers choose routes according to a *routing policy*, a set of decision rules based on information available at the time of each decision.

The specific questions being addressed are:

1. Do drivers think strategically when plan-

ning routes in uncertain networks with real-time, en route information?

2. Can observations of route choice be used to estimate a predictive model that accounts for both risk attitude and strategic thinking?

3 The Experiment

The experiment was conducted on the UMass Amherst campus, with 74 participants recruited from both UMass and the surrounding communities. Subjects answered an interactive survey, using graphical maps with a point-and-click interface. After completing several warm-up scenarios, each with an introduction from a study coordinator, subjects completed six groups of scenarios, with breaks between each. The study coordinator introduced each scenario group to provide context and highlight any changes. In total, over 3500 choice observations were collected.

Figure 1: Example map interface, with information and detour

Two different map types are used in the sur-

vey: a simple two-path map, designed to measure risk attitude, and a more complex map with a detour, designed to measure strategic thinking. Each risk scenario is presented in both map types, which enables direct comparison of behavior in simple and strategy maps.

Simple Risk Maps

The simple risk maps are aimed at determining a subject's risk attitude without the influence of real-time information or detours. The subject decides between two routes, one with a deterministic travel time, and the other with a stochatic travel time. This network type is represented in Figure 2.
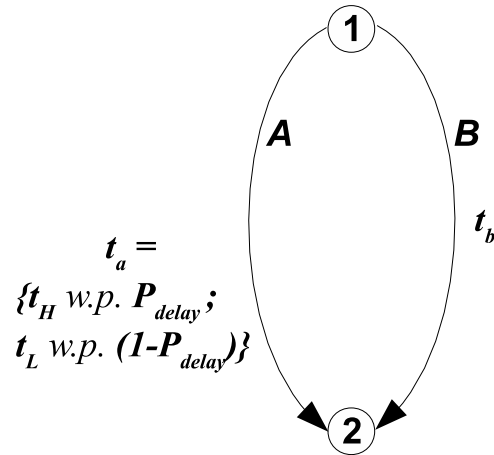
Figure 2: Abstract network for simple risk map type

The user sees the exact travel time of the deterministic route ($Link\ B$) before the trip begins. The stochatic travel time on $Link\ A$ has two possible outcomes, t_H w.p. P_{delay}, and t_L, w. p. $(P_{delay} - 1)$. Before the trip, the user is shown the possible outcomes and probability of delay, but the actual travel time of route A is revealed only if and when the subject traverses

the stochastic link.

Strategy Maps

The strategy maps (Figure 3) measure the extent to which a subject recognizes strategically advantageous real-time information. Subjects choose between a deterministic path and an alternative which branches into a stochastic link (C) and a deterministic detour (link D). Subjects who traverse link A will learn the actual travel time of link C before they must decide whether to use it. This allows the subject to choose the faster of link C or D.
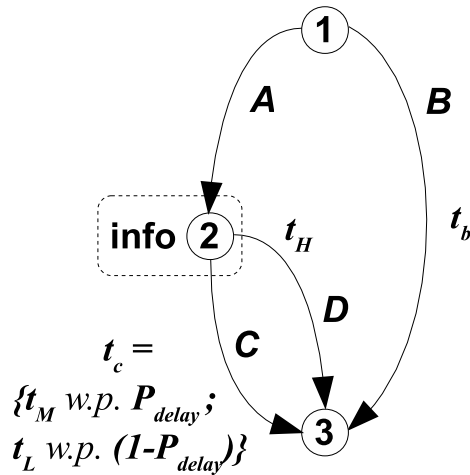
Figure 3: Abstract network for routing strategy tests

Subjects who plan in advance for this informed choice will recognize that they can always avoid the large potential delay of the stochastic link. Such "strategic" subjects would tend to assess the stochastic branch based on the expected minimum travel time of links C and D.

"Non-strategic" subjects would instead tend to view the stochastic branch as two individual paths, with expected travel times $E[t_c]$ and t_d.

To more easily identify strategic behavior, the experiment is designed such that both expected times are higher than the strategic assessment.

4   Analysis of Results

Results from the survey are analyzed in two stages. In the first stage, metrics are developed to validate the data and illuminate any major trends. The findings are used to guide the second stage, in which quantitative statistical models are developed and estimated against the data.

Since strategic behavior is only distinguishable in the presence of risky alternatives, risk attitude must be accounted for before attempting any analysis of strategy. This is done by analyzing data from the simple risk maps, which had no strategic component.

Risk Attitude

In the first stage of risk attitude analysis, a benefit/risk ratio is calculated as a rating for the stochastic alternative in each scenario. Benefit is defined as expected savings in travel time over the deterministic alternative, and risk is defined as the standard deviation of outcomes for the stochastic alternative.

An approximate benefit/risk threshold is calculated for each subject, including a small "ambivalent range" of ratios for which the user's choice behavior was not consistent. The threshold levels appear to depend heavily on delay probability, with very risk-averse behavior observed at low probabilities and very risk-prone behavior observed at high probabilities. This is

consistent with the findings in [2].

This dependency is further confirmed by model estimation, which shows significant differences in the weighting of risk at different levels of P_{delay}. The model form used in this analysis is

$$
\begin{aligned}
V_x &= \beta_{ett} ETT_x + \theta_{p02} * P02 * \sigma_x \\
&\quad + \theta_{p05} * P05 * \sigma_x + \theta_{p08} * P08 * \sigma_x \\
&\quad + ASC_x
\end{aligned}
$$

where ETT_x is the expected travel time of alternative x, σ_x is the standard deviation of possible outcomes for alternative x, and ASC_x is an constant applied only to stochastic alternatives. The parameters θ_{p02}, θ_{p05}, and θ_{p08} are dummy variables to account for the three levels of delay probability presented in the experiment.

While this model form is ideal for analyzing the particular effects of the experimental variables, applicability to other research or real-world data is limited. A more predictive modeling framework, based on the cumulative prospect theory (CPT) approach developed in [4], is also explored in detail. The CPT model defines a generally applicable functional form for the behaviors observed in this experiment.

Strategic Thinking

Each strategy map has a corresponding simple risk map with identical travel times and distributions. A "strategic" subject who chose the stochastic alternative in a simple map would be expected to choose the stochastic branch in the strategy map. A "non-strategic" subject, on the other hand, would perceive a much larger potential delay in the strategy map and most likely avoid the stochastic branch.

The first stage of strategy analysis compares each subject's choices in each simple map and the corresponding strategy map. Overall percentages of distinguishably strategic and non-strategic observations are calculated, as well as an estimate of how many users can be considered "strategic". These calculations yield upper and lower bounds on the expected prevalence of strategic behavior, and provide guidance and validation for the second stage of analysis. This analysis estimates an upper bound of approximately 84.2% strategic observations, and a lower bound of approximately 30.6%.

The second stage of analysis uses latent-class choice models to estimate the probability of an observation resulting from strategic behavior. The models incorporate the findings of the simple risk analysis, in order to control for the influence of risk alone. Estimation results are consistent with the first-stage strategy analysis, finding that significant levels of both strategic and non-strategic behavior are present. The overall estimated percentage of strategic observations is approximately 81%.

The sample size was sufficient to obtain significant results for all parameters of the models based on expected travel time and standard deviation of travel time. Limitations of estimating the CPT-based models given the sample size and variability of the experiment are explored in detail.

References

[1] M. A. Abdel-Aty and M. F. Abdalla, *Examination of multiple mode/route-choice paradigms under atis*, IEEE Transactions on Intelligent Transportation Systems **7** (2006), no. 3, 332–348.

[2] E. Avineri and J. N. Prashker, *Violations of expected utility theory in route-choice stated preferences*, Transportation Research Record **1894** (2004), 222–229.

[3] S. Gao, E. Frejinger, and M. Ben-Akiva, *Adaptive route choice models in stochastic time-dependent networks*, Transportation Research Record **2085** (2008), 136–143.

[4] S. Gao, E. Frejinger, and M. Ben-Akiva, *Adaptive route choice in risky traffic networks: A prospect theory approach*, Transportation Research Part C (2009), DOI: 10.1016/j.trc.2009.08.001.

An Enhanced Exact Algorithm for the Multi-Vehicle Routing Problem with Stochastic Demands

Michel Gendreau

Département de mathématiques et génie industrielle

Polytechnique Montréal

CIRRELT

Walter Rei

Département de management et de technologie

École des sciences de la gestion

Université du Québec à Montréal

CIRRELT

Email: walter.rei@cirrelt.ca

1   Introduction

The vehicle routing problem (VRP) and its many variants have been extensively studied within the operations research community (see [8]). Both innovative models and efficient algorithms have been developed to formulate and solve these hard combinatorial optimization problems. However, it should be noted that the majority of the research that has been done within this field has been aimed at addressing deterministic versions of the VRP. When using a deterministic optimization model to formulate a given problem, one makes the implicit hypothesis that all information concerning the parameters of the model are readily available at the moment when the problem is to be solved. This is rarely the case in practice, where one more likely has an idea of the distribution of the different parameters without knowing the exact values that will be observed. Therefore, in more recent years, stochastic versions of the VRP have been developed, where given parameters are modeled using either random variables or possible scenarios. Our aim here is to address the classical VRP with capacity constraints when client demands are stochastic.

In the VRP with stochastic demands (VRPSD), using a fleet of vehicles of limited capacity

(we consider here the case where all vehicles have the same capacity), one must deliver, or pick up, some amounts of an homogeneous good to a set of customers. The particularity of the present problem is that, for each customer, the exact amount of demand is only revealed once a vehicle arrives at the customer's location. Given this characteristic of the problem, a vehicle may reach a client and then face the situation where the residual capacity is insufficient to serve the observed demand. In such a case, a *route failure* occurs and a recourse decision must be made to correct it.

Formulating the proper recourse decisions to consider has been an important focus of the modeling effort conducted on the VRPSD. In classical models, these recourse decisions are defined as return trips to the depot to reload (or unload) the vehicle whenever a *route failure* is observed. However, more complex recourse policies may be considered. Examples of such policies are the restocking rule [9], the reoptimization approach [7] or the pairing strategy [1]. In all cases, these recourse decisions entail extra costs, either in terms of additional distances traveled by the vehicles or in terms of additional resources used. What differentiates the stochastic VRP models from deterministic ones is the fact that these extra costs are explicitly considered when establishing the routes to be performed by the vehicles. Our study will focus on the extra costs incurred when considering the classical recourse definition, which can serve as a benchmark for all possible policies.

The VRPSD is formulated as a two-stage stochastic programming model. The first stage corresponds to a planning phase where one constructs a series of routes (one for each vehicle) that visit all clients once. It should be noted that capacity constraints are imposed to restrict the total expected demand for the goods that are distributed (or collected) on each route to be less than the capacity of a vehicle. By doing so, one may obtain more balanced routes. The second stage represents an operation phase where the routes are executed and recourse actions are taken according to the observed demands. The objective of the model then becomes to minimize the sum of both the routing cost and the average recourse cost.

To obtain an optimal solution to this model, the 0-1 integer L-Shaped approach proposed in [4] has been applied in several cases (see [2, 3, 5]). Given the complexity associated with stochastic optimization problems, solving to optimality larger instances of the VRPSD remains a challenge. Our main contribution is to propose a series of new strategies, derived from the inclusion of local branching valid inequalities [6] and a generalization of the lower bounding functional (LBF) originally proposed in [3], which will be shown to significantly enhance this solution approach on the considered problem. As a consequence, the proposed strategies, which help improve the bounds generated by the algorithm, allow us to successfully tackle larger instances of the problem. In the next two sections of the present document, we will briefly present the 0-1 integer L-Shaped algorithm as it applies to the VRPSD and give an outline of the various strategies that we propose in order to produce a more efficient exact algorithm for the problem.

2   The L-Shaped method applied to the VRPSD

The 0-1 integer L-Shaped algorithm is a branch and cut method that is based on the general principles of Benders decomposition. When applying this solution approach to the case of the VRPSD, in addition to replacing the recourse function by a valid lower bound within the master problem, the integrality requirements on the variables as well as the subtour and capacity constraints are also relaxed. The algorithm then proceeds by gradually imposing integrality requirements through a branching process and adds both violated subtour and capacity constraints as they are obtained by a separation strategy. Since the problem consists of finding in the first stage a series of feasible routes, both subtour and capacity constraints serve as the feasibility cuts (i.e., valid inequalities that are added to find first stage solutions that induce feasibility in the second stage) for the VRPSD. It should be noted that the extensive amount of research that has been done in the case of deterministic VRPs, to provide both valid inequalities and separation procedures aimed at finding violated subtour and capacity constraints (see [8]), can again be used in the stochastic case. As for optimality cuts (i.e., valid inequalities that are used to bound the recourse function), one is added whenever a set of feasible routes is obtained. In this lies the main challenge to efficiently apply the considered solution method to the VRPSD. Since a set of feasible routes must be obtained before improving the lower bound on the recourse function and given that the information provided by the added optimality cut is very local (see [4]), there is a risk of enumeration for the algorithm.

3   Enhancing the L-Shaped method for the VRPSD

In this section we briefly outline the different strategies proposed to improve the 0-1 integer L-Shaped algorithm when applied to the VRPSD. These strategies include the use of both local branching cuts and a set of extended cuts that generalize the LBF valid inequalities originally proposed in [3].

3.1   Applying Local Branching cuts

It was shown in [6] how one can derive a set of valid inequalities for the relaxed master problem used by the 0-1 integer L-Shaped algorithm following a local branching descent. These inequalities bound the recourse function in different local branching neighbourhoods defined around a given point. We propose to use this strategy to improve the optimality cuts generated by the algorithm, by performing local branching descents around feasible routes. In addition to improving the bounds on the recourse function, as was shown in [6], improvements to the upper bound for the solution process can also be expected. Furthermore, we will propose a general separation strategy, based on local branching, to produce a more efficient exploration of the subregions defined by the subproblems processed by the algorithm.

3.2 Extended LBF cuts

LBF cuts were developed to bound the recourse value associated with partial routes for the one vehicle VRPSD in [3] and extended to the multi-vehicle case in [5]. It was shown in [5] that a great number of these cuts have to be added to the master problem in oder to solve the multi-vehicle VRPSD. We propose to generalize the definition of partial routes to produce a series of different LBF cuts that can be used to bound the recourse value associated with a broader number of partial solutions. An exact separation procedure is proposed to both identify general partial routes and produce a series of violated LBF cuts. It will be shown that this strategy can greatly improve the lower bound obtained by the algorithm.

References

[1] A. Ak and A. Erera, "A paired-vehicle recourse strategy for the vehicle-routing problem with stochastic demands", *Transportation Science* 41, 222-237 (2007).

[2] M. Gendreau and G. Laporte and R. Séguin, "An exact algorithm for the vehicle routing problem with stochastic demands and customers", *Transportation Science* 29, 143-155 (1995).

[3] C. Hjorring and J. Holt, "New optimality cuts for a single-vehicle stochastic routing problem", *Annals of Operations Research* 86, 569-584 (1999).

[4] G. Laporte and F. Louveaux, "The integer L-Shaped method for stochastic integer programs with complete recourse", *Operations Research Letters* 13, 133-142 (1993).

[5] G. Laporte, F.V. Louveaux and L. Van Hamme, "An integer L-Shaped algorithm for the capacitated vehicle routing problem with stochastic demands", *Operations Research* 50, 415-423 (2002).

[6] W. Rei, M. Gendreau and P. Soriano, "Local branching cuts for the 0-1 integer L-Shaped algorithm", Working Paper CIRRELT-2007-23, Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT), Montréal, Québec, 2007.

[7] N. Secomandi and F. Margot, "Reoptimization approaches for the vehicle-routing problem with stochastic demands", *Operations Research* 57, 214-230 (2009).

[8] P. Toth and D. Vigo, *The Vehicle Routing Problem*, SIAM, Monographs on Discrete Mathematics and Applications (2001).

[9] W. Yang, K. Mathur and R.H. Ballou, "Stochastic vehicle routing problem with restocking", *Transportation Science* 34, 99-112 (2000).

Uncertainty in Planning City Logistics Operations

Nicoletta Ricciardi

Dept. di Statistica, Probabilità e Statistiche Applicate

Sapienza Università di Roma, Italy

Email: nicoletta.ricciardi@uniroma1.it

Teodor Gabriel Crainic, **Fausto Errico**, **Walter Rei**

Dept. management et technologie

École des sciences de la gestion, UQAM, Montréal, Canada

and

Interuniversity Centre for Enterprise Networks, Logistics and Transportation

(CIRRELT)

1   City Logitics

For urban areas, the transportation of goods constitutes both a major enabling factor for most economic and social activities and a major source of disturbances, e.g., [1, 6] and references within. Several concepts have been introduced and several projects have been undertaken in recent years to reduce the impact of freight-vehicle movements on city-living conditions and, in particular, improve the congestion/mobility and pollution conditions, while not penalizing the city social and economic activities. The fundamental idea which underlines most initiatives is to stop considering each shipment, firm, and vehicle individually, but rather as components of an *integrated logistic system*. The term *City Logistic* has been coined to describe such systems and the optimization of their activities [7]. Several organizational models have been proposed, but it is acknowledged by now that significant gains can only be achieved through a streamlining of distribution activities resulting in less freight vehicles traveling within the city and a better utilization of these vehicles. The *consolidation* of loads of different shippers and carriers within the same vehicles associated to some form of *coordination* of the resulting freight transportation activities are among the most important means to achieve these goals.

We focus on City Logistics systems appropriate for large urban areas, where consolidation and coordination activities are performed at facilities organized into a hierarchical, two-tiered structure with major terminals sited at the city limits and satellite facilities strategically located close to or

within the city-center area, and particular vehicle fleets dedicated to each system tier [3]. Assume, for simplicity of presentation, that demand is present at the first level of facilities as loads destined for particular customers. Each customer demand for a given product is characterized by a quantity, time of availability at the external zone, and delivery time window. Loads are transported on *urban vehicles* from first-level facilities to satellites, where they are transferred to and consolidated into *city freighters*, vehicles adapted for utilization in dense city zones, which perform the actual delivery routes in the second tier of the system. Satellites operate according to a vehicle-synchronization and cross-dock transshipment operational model, i.e., urban vehicles and city freighters meet at satellites at appointed times, with short waiting times permitted, loads being transferred without intermediate storage.

Similarly to any complex transportation system, City Logistics systems require planning at strategic, tactic, and operational levels [1, 7]. Tactical planning aims to provide the means to consolidation-based carriers and their customers for cost and service-quality efficient resource allocation and operations, through a transportation plan and schedule to be operated repetitively over a given planning horizon. For City Logistics, this efficiency must also be achieved for the city traffic and environmental conditions. A modeling framework for City Logistics tactical planning was introduced in [4], where the main focus was on building a detailed plan for the "next-day" activities specifying the urban vehicle and city freighter routes and schedules, as well as the delivery routes from the major terminals, through satellites, to the final customers. Similar to most of the City Logistics literature, the authors did not address uncertainty issues, nor did they study in any detail the broader issue of defining a tactical plan for regular operations. Our goal is to start filling this gap and present processes and models to build tactical tactical plans for two-tiered City Logistics systems that account for the uncertainty in transportation demand.

2 Uncertainty and planning

Tactical planning is often performed using deterministic service network design models and forecasts for the values of internal (e.g., operation times) and external (e.g., demand) problem parameters [2]. Several recent contributions (e.g., [5]) have shown, however, that plans built considering the variability of some of these parameters are qualitatively different from those obtained through traditional approaches and offer increased flexibility and robustness. It is therefore not without interest to explore these issues for City Logistics.

We focus in this presentation on the variability associated to *demand* and its treatment in building tactical plans for the regular operations of the City Logistics system. The first issue that needs to be discussed is the definition of the information process, the scope of the tactical plan, and the actions to be left for the actual adjustment of the plan to actual demands and operational

conditions.

Similarly to most processes for consolidation-based carriers, a tactical plan concerns a span of time, the *season*, where demand is relatively stable (a slightly different plan may be built for each *day*, but we will not address this aspect in this presentation). The plan is built some time before the start of the season and it is then applied repetitively each day of that season, once it has been adjusted to the particular conditions of the day. With respect to demand variability (or one's confidence in the forecasting processes) and the scope of the planning process, the problem we address stands at some distance from the two classical extreme cases: total confidence, deterministic model to plan and schedule all activities, and pay for extra resources when executing if needed, on the one hand, and high variability, no or little *a priori* planning, and real-time decisions, on the other hand. Demand is defined for products and demand zones, defined in [4] as homogeneous geographical aggregations of customers with respect to the product and the delivery time window. Given the cooperation and information system connecting customers, their suppliers and the City Logistics system, a rather precise forecast may be assumed regarding the regular demand for each customer-zone demand, as well as the expected variation. The plan is then built to allow the allocation of the major resource groups for the season, in particular, the number of departures from external zones, the satellite work loads, the allocation of customer zones to satellites, the size of the vehicle fleets required. Then, for every day of the season, the actual demand is observed at the individual customer level, and the plan is instantiated by, in particular, defining the routes and schedules of the city freighters to perform the actual deliveries.

We therefore examine two-stage formulations with recourse: $\text{Minimize}_{x \in X, \omega \in \Omega} f(x) + \text{E}_{x_i}(Q(x, x_i(\omega)))$, where x stands for the first stage decisions, i.e., the *a priori* plan based on available information, $x_i(\omega) = d(\omega)$ for the realization of demand for $\omega \in \Omega$, and $Q(x, x_i(\omega))$ for the cost of operating the system using the *a priori* plan x, the realized demand $x_i(\omega)$, and a recourse policy which instantiates the plan once the uncertain data is resolved.

The first stage decisions concern the design of the first tier service network and the itineraries delivering the demands to satellites, given an approximation of the delivery cost from satellites to the customer zones. This corresponds to the restricted planning problem of [4]. We examine two recourse strategies, increasing the amplitude of the recourse decisions.

The *Routing* recourse inherits from the first stage decisions the scheduled service network of the first tier, and the number of city freighters, of each type, that leave satellites at each time period. The second stage problem than aims to determine the work assignments - routes and schedules for the day - of all city freighters given the stochastic demand. This part corresponds to the synchronized, scheduled, multi-depot, multiple-tour, heterogeneous vehicle routing problem with time windows introduced in [4]. The recourse is completed by including in the formulation the possibility to use extra capacity (the urban-vehicle services and numbers of city freighters

might not be sufficient for the realized demand) offered by city-freighters operating pick-up and delivery-type of routes (with LIFO loading) linking external zones and customers.

The *Service Time-Shift* recourse starts from the same the first stage as previously inheriting the selected urban-vehicle services and the bundles of customer zones to be served by each city-freighter type at each satellite and time period. It then proceeds to examine each customer bundle (by city-freighter type) and identifies satellite and service *opportunity windows* such that customers could still be served on time. The recourse then optimizes a restricted planning problem (the full model of [4]), where additional departures are permitted for the previously selected services and the city-freighter work assignments are determined to deliver demand on time while respecting satellite capacity constraints. The previous extra capacity utilization is also included in this formulation.

We will present and discuss the models for the different recourse strategies together with algorithmic perspective and preliminary results.

References

[1] A. Benjelloun and T.G. Crainic, "Trends, Challenges, and Perspectives in City Logistics", In *Transportation and Land Use Interaction, Proceedings TRANSLU'08*, 269–284. Editura Politecnica Press, Bucharest, Romania, 2008.

[2] T.G. Crainic and K.H. Kim. "Intermodal Transportation", In *Transportation*. C. Barnhart and G. Laporte (eds), Vol. 14 of *Handbooks in Operations Research and Management Science*, 467–538, Elsevier, 2007.

[3] T.G. Crainic, N. Ricciardi, and G. Storchi, "Advanced Freight Transportation Systems for Congested Urban Areas", *Transportation Research Part C: Emerging Technologies*, 12(2): 119–137, 2004.

[4] T.G. Crainic, N. Ricciardi, and G. Storchi, "Models for Evaluating and Planning City Logistics Transportation Systems", *Transportation Science* 43(4): 432–454, 2009.

[5] A.-G. Lium, T.G. Crainic, and S.W. Wallace, "A Study of Demand Stochasticity in Service Network Design", **Transportation Science** 43(2): 144–157, 2009.

[6] "Delivering the Goods: 21st Century Challenges to Urban Goods Transport", Technical report, Organisation for Economic Co-operation and Development, OECD Publishing, http://www.oecdbookshop.org, 2003.

[7] E. Taniguchi, R.G. Thompson, T. Yamada, and J.H.R. van Duin, *City Logistics: Network Modelling and Intelligent Transport Systems*. Pergamon, Amsterdam, 2001.

The Multiple Vehicle Travelling Purchaser Problem

Jorge Riera-Ledesma   **Juan José Salazar-González**

DEIOC - University of La Laguna, Tenerife, Spain

jriera@ull.es , jjsalaza@ull.es

1   Introduction

The motivation of this work comes from the problem of designing routes to school buses in urban areas. In this problem we have to approach two issues: bus stop selection for each student, and bus route design for each vehicle. Therefore our problem fits in *Location-Routing* (see, e.g., [1]). In most of previous approaches these two issues has been addressed independently (see, e.g., [2]). To approach the two issues simultaneously, we present the *Multiple Vehicle Traveling Purchaser Problem* (MV-TPP), defined as follows.

Let us given a set of pupils. All pupils are assumed to go to the same school. Each pupil is assigned with a set of potential bus stops. A bus stop may be reachable by more than one pupil. A fleet of homogeneous vehicles is available in a central depot with the purpose of carrying all pupils to the school. In addition to the routing cost associated with each connection, there is a cost associated with the assignment of a pupil to a potential stop. This cost is related to the distance from the pupil home to the bus stop. We also consider a fixed cost for using each vehicle. The aim of this problem is assigning a stop to each pupil and designing the routes such that the vehicle capacity holds and the total cost (including routing and assignment) is minimized.

The combination of these two issues has already been approached from a heuristic point of view following both *location-allocation-routing* (LAR) [3, 4, 5, 6] and *allocation-routing-location* (ARL) [7, 8] strategies (see, [1, 8]). The LAR strategy determines in a first step a set of bus stops for a school with the purpose of assigning the students to them. Thus, a second step generates routes for the involved bus stops. On the other hand, ARL strategy allocate the students into clusters while satisfying vehicle capacity constraints. Then, the bus stops are selected, and a route is generated for each cluster. Exact approaches based on Mathematical Programming techniques have been also proposed [9, 10]. Boding and Berman [3] proposed a specific heuristic approach for a school bus routing problem with a constraint on the maximum allowable travel time for each pupil. In

Dulac et al. [4] pupils are located on the street segment of which they live but, as in [3] these users are assigned to the nearest stop. Their problem is constrained on the number of visited stops, the transportation time for each vehicle and on the distance walked by each pupil. In Desrosiers et al. [5, 6] consider that the upper bound of the length of each route in urban area is negligible because of the high population density. The objective is then to generate a minimum number of routes filling them as close as possible to their capacity. Chapleau at al. [7] and Bowerman et al. [8] make use of the ARL strategy.

The MV-TPP generalizes the classical Capacitated Vehicle Routing Problem (VRP) [11] considering variable demand for each client. The MV-TPP is a clear generalization of the Traveling Purchaser Problem (TPP) [12] in which a single vehicle without limit on its capacity has to collect all users. As pointed in [13] the TPP becomes *intractable* [14] since it generalizes both the Traveling Salesman Problem and the Uncapacitated Facility Location Problem. Figure 1 shows a feasible solution of a MV-TPP with capacity $Q = 5$.
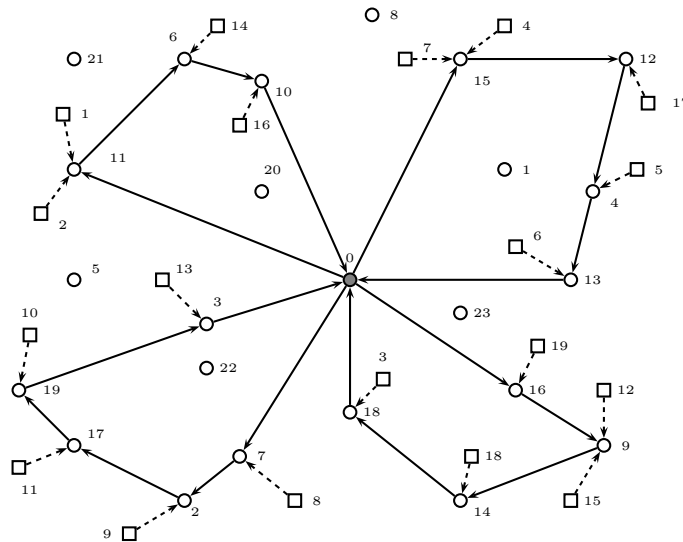
Figure 1: Feasible solution of MV-TPP

2 Mathematical formulation

This section formulates the MV-TPP. Although the initial integer program for the MV-TPP is based on a non-linear model, we derive later different families of linear inequalities replacing those families involved in the non-linearity.

2.1 Notation

Let U denote the set of pupils to be carried, N the set of potential stops, and 0 represents the school and central depot, where the fleet of homogeneous vehicles is available.

Because of the constraints on the distance walked by each pupil to the bus stop, we also denote by $N(u)$, for all pupil $u \in U$, the set of potential stops reachable by u. Conversely, $U(i)$ denotes the set of pupils able to reach the bus stop i, for all $i \in N$.

We model MV-TPP on a directed graph $G = (V, A))$, where $V = N \cup \{0\}$. The set of arcs of graph G is defined as $A = \{(i, j) : i \in N \cup \{0\}, j \in (N \cup \{0\}) \setminus \{i\}\}$. Each vehicle of the fleet has a maximum and identical capacity of Q pupils. For each arc $a \in A$, c_a is the routing cost of arc a. For each pair (u, i) denoting a potential assignment, for all pupil $u \in U$ and all stops reachable by u, $i \in N(u)$, c_{ui} is the cost of assigning u to i. For any subset of locations $S \subseteq V$, we define $\delta^+(S) = \{(i, j) \in A : i \in S, j \notin S\}$ and $\delta^-(S) = \{(i, j) \in A : i \notin S, j \in S\}$. If $S = \{i\}$, we simply write $\delta(i)$ instead of $\delta(\{i\})$. We denote by $A(S, T)$, for all $S, T \subseteq V$ and $S \cap T = \emptyset$, the set of arcs going from each vertex of S to each vertex of T.

2.2 A non-linear formulation for the MV-TPP

$$\text{Minimize} \quad \sum_{(i,j) \in A} c_{ij} x_{ij} + \sum_{u \in U} \sum_{i \in N(u)} c_{ui} z_{ui}$$

subject to

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} = y_i \qquad \forall i \in N, \tag{1}$$

$$\sum_{(i,j) \in \delta^+(i)} x_{ij} = \sum_{(j,i) \in \delta^-(i)} x_{ji} \qquad \forall i \in V, \tag{2}$$

$$\sum_{(j,i) \in \delta^-(i)} f_{ji} - \sum_{(i,j) \in \delta^+(i)} f_{ij} = \sum_{u \in U(i)} z_{ui} \qquad \forall i \in N, \tag{3}$$

$$f_{ij} \leq \left(Q - \sum_{u \in U(j)} z_{uj} \right) x_{ij} \qquad \forall (i,j) \in A, \tag{4}$$

$$f_{ij} \geq \sum_{u \in U(i)} z_{ui} x_{ij} \qquad \forall (i,j) \in A, \tag{5}$$

$$\sum_{i \in N(u)} z_{ui} = 1 \qquad \forall u \in U, \tag{6}$$

$$f_{ij} \geq 0 \qquad \forall (i,j) \in A, \tag{7}$$

$$x_{ij} \in \{0, 1\} \qquad \forall (i,j) \in A, \tag{8}$$

$$y_i \in \{0, 1\} \qquad \forall i \in N, \tag{9}$$

$$z_{ui} \in \{0, 1\} \qquad \forall u \in U, \forall i \in N(u). \tag{10}$$

2.3 Further details

During the talk we will show how we have linearized the previous model and how we strengthen the LP relaxation. We will also present an alternative column-generation model. Merging the two models, we have implemented a branch-and-cut-and-price approach to MV-TPP. We will present and discuss computational experiments on randomly generated instances.

References

[1] G. Laporte, Location-routing problems, in: B. Golden, A. Assad (Eds.), Vehicle Routing: Methods and Studies, North-Holland, Amsterdam, 1988, pp. 163–198.

[2] J. Park, B.-I. Kim, The school bus routing problem: A review, Eur. J. Oper. Res. 202 (2010) 311–319.

[3] L. Bodin, L. Berman, Routing and scheduling of school buses by computer, Transportation Sci. 13 (1979) 113–129.

[4] G. Dulac, J. A. Ferland, P. A. Forgues, School bus routes generator in urban surroundings, Comput. Oper. Res. 7 (1980) 199–213.

[5] J. Desrosiers, J. Ferland, J.-M. Rousseau, G. Lapalme, L. Chapleau, An overview of a school busing system, in: N. Jaiswal (Ed.), Scientific Management of Transport Systems, North-Holland, Amsterdam, 1981, pp. 235–243.

[6] J. Desrosiers, J. Ferland, J.-M. Rousseau, G. Lapalme, L. Chapleau, Transcol: a multi-period school bus routing and scheduling system, TIMS Studies in the Management Sciences 22 (1986) 47–71.

[7] L. Chapleau, J.-A. Ferland, J.-M. Rousseau, Clustering for routing in densely populated areas, Eur. J. Oper. Res. 20 (1985) 48–57.

[8] R. Bowerman, B. Hall, P. Calamai, A multi-objective optimization approach to urban school bus routing: Formulation and solution method, Transportation Res. A 29 (1995) 107–123.

[9] J. Riera-Ledesma, J. J. Salazar-González, Solving a school bus routing problem, in: Odysseus 2006, Third International Workshop on Freight Transportation and Logistics, Altea, Spain, 2006, pp. 285–288.

[10] P. Schittekat, M. Sevaux, K. Sörensen, A mathematical formulation for a school bus routing problem, in: IEEE 2006 International Conference on Service Systems and Service Management, Troyes, France, 2006.

[11] P. Toth, D. Vigo (Eds.), The Vehicle Routing Problem, Vol. 9, SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, PA., 2002.

[12] T. Ramesh, Traveling purchaser problem, Opsearch 18 (1981) 78–91.

[13] G. Laporte, J. Riera-Ledesma, J. J. Salazar-González, A branch-and-cut algorithm for the undirected traveling purchaser problem, Oper. Res. 51 (2003) 940–951.

[14] M. R. Garey, D. S. Johnson, Computers and Intractability, A Guide to the Theory of NP-Completeness, W. H. Freeman and Company, New York, 1979.

New Benchmark Results for the
Capacitated Vehicle Routing Problem

Roberto Baldacci   **Aristide Mingozzi**

DEIS   Department of Mathematics

University of Bologna   University of Bologna

Roberto Roberti

DEIS

University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

Email: roberto.roberti6@unibo.it

1   Introduction

The Capacitated Vehicle Routing Problem (CVRP) considered in this paper involves a fleet of identical vehicles located at a central depot and a set of customers, each with a given demand of goods to be supplied from the depot. Every route performed by a vehicle must start and end at the depot, and the load carried must be less than or equal to the vehicle capacity. It is assumed that the "cost matrix" of the least cost paths between each pair of customers is known. The cost of a route is computed as the sum of the costs of the arcs forming the route. The objective is to design vehicle routes (one route for each vehicle) so that all customers are visited exactly once and the sum of the route costs is minimized. In this paper, the cost matrix is assumed to be symmetric.

The most effective exact algorithms for the CVRP are due to Baldacci et al. [1], Lysgaard et al. [7], Fukasawa et al. [5], and Baldacci et el. [2].

Fukasawa et al. [5] described a Branch-and-Cut-and-Price (BCP) for solving the Set Partitioning (SP) model of the CVRP strengthened by the valid inequalities introduced by Lysgaard et al. [7]. The lower bound is computed by a column-and-cut generation method that uses q-routes (see Christofides et al. [4]) instead of feasible CVRP routes. This method is combined with the Branch-and-Cut (BC) of Lysgaard et al. [7]. At the root node, Fukasawa et al. [5] decide which algorithm to use: the BC or the new BCP.

Baldacci et al. [2] proposed an exact method for the CVRP based on the SP model that on average outperforms all other exact methods on the main CVRP instances from the literature.

Table 1: Summary results for the CVRP

Class	np	Lysgaard et al. [7]		Fukasawa et al. [5]		Baldacci et al. [2]		Our method	
		nopt	t_{AVG}	nopt	t_{AVG}	nopt	t_{AVG}	nopt	t_{AVG}
A	22	15	6,638	22	1,961	22	118	22	31
B	20	19	8,178	20	4,763	20	417	20	67
E-M	9	3	39,592	9	126,987	8	1,025	9	305
P	24	16	11,219	24	2,892	22	187	24	81
Avg			**10,438**		**18,009**		**323**		**89**
Tot	**75**	**53**		**75**		**72**		**75**	

They proposed an additive bounding procedure that combines two dual ascent heuristics, called H^1 and H^2, to derive a near-optimal dual solution used by a cut-and-column generation algorithm, called H^3, to initialize the master problem. H^3 attempts to close the integrality gap by adding in a cutting plane fashion both generalized capacity and clique constraints. The final dual solution achieved is then used to generate a reduced SP problem containing only the routes whose reduced cost is smaller than the gap between an upper bound and the lower bound obtained. The resulting reduced problem is then solved by an integer programming solver.

In this paper, we further improve the method of Baldacci et al. [2] using new ideas introduced by Baldacci et al. [3] for solving the VRP with Time Windows. In particular, we use a new route relaxation, called ng-route, that strongly improves other relaxations of feasible routes proposed in the literature for the CVRP and increases the efficiency of the pricing algorithms. We improve procedure H^3 using Subset-Row inequalities (see Jepsen et al. [6]) and a novel strategy for solving the pricing subproblem. Finally, a new fathoming criterion based on the dual solution achieved by H^2 is used to speed up the solution of the pricing subproblems and reduce the size of the final SP model.

2   Computational Results

In this section, we report a computational comparison of the new results obtained with those of Lysgaard et al. [7], Fukasawa et al. [5] and Baldacci et al. [2] on five classes of CVRP instances from the literature, called A, B, E, M and P (available at http://branchandcut.org/VRP/data).

The algorithm of Lysgaard et al. [7] was run on an Intel Celeron at 700 MHz while the algorithms of Baldacci et al. [2] and Fukasawa et al. [5] used Pentium 4 processors running at 2.6 GHz and 2.4 Ghz, respectively. Our algorithm was run on an IBM Intel Xeon X7350 Server at 2.93 GHz. According to SPEC benchmarks, our machine is three times faster than the Pentium 4 processors of Baldacci et al. [2] and Fukasawa et al. [5] and ten times faster than the machine of Lysgaard et al. [7].

Table 1 reports a summary of the computational results obtained by the four exact methods

Table 2: Results on difficult CVRP instances

Name	z^*	Lysgaard et al. [7]			Fukasawa et al. [5]			Baldacci et al. [2]			Our method		
		%LB	t_{LB}	t_{TOT}	%LB	t_{LB}	t_{TOT}	%LB	t_{LB}	t_{TOT}	%LB	t_{LB}	t_{TOT}
A-n54-k7	1,167	97.3	30	7,246	98.9	125	1,409	99.5	43	86	99.8	14	14
A-n64-k9	1,401	96.5	132	tl	98.9	265	11,254	99.5	32	120	99.6	21	24
A-n80-k10	1,763	97.0	201	tl	99.5	1,120	6,464	99.6	117	194	99.7	247	250
B-n50-k8	1,312	97.6	26	tl	98.7	97	2,845	99.7	640	662	99.8	118	147
B-n66-k9	1,316	98.7	80	tl	99.4	145	1,778	99.5	216	227	99.9	283	292
B-n68-k9	1,272	98.9	65	tl	99.3	260	87,436	99.5	254	6,168	99.7	336	526
E-n51-k5	521	99.6	24	59	99.5	51	65	100.0	13	13	100.0	2	2
E-n76-k7	682	97.7	72	118,683	98.2	264	46,520	99.0	146	3,371	99.8	290	291
E-n76-k8	735	97.7	136	tl	98.8	277	22,891	99.3	104	873	99.9	63	64
E-n76-k10	830	96.4	158	tl	98.5	354	80,722	99.5	60	174	99.5	19	21
E-n76-k14	1,021	95.0	181	tl	98.6	224	48,637	99.6	17	45	99.4	7	9
E-n101-k8	815	98.5	222	tl	98.8	1,068	801,963	99.0	250	-	100.0	638	638
E-n101-k14	1,067	96.2	555	tl	98.8	658	116,284	99.7	154	1,230	99.6	319	453
M-n101-k10	820	100.0	33	33	100.0	119	119	100.0	47	47	100.0	19	19
M-n121-k7	1,034	98.4	979	tl	99.7	5,594	25,678	99.8	944	2,448	99.9	1,247	1,249
P-n50-k8	631	95.4	28	tl	97.7	102	9,272	99.0	12	596	99.1	12	14
P-n55-k10	694	95.4	53	tl	98.2	107	9,076	99.3	16	66	99.3	6	8
P-n70-k10	827	96.2	90	tl	98.5	292	24,039	99.4	35	774	99.5	116	118
P-n76-k5	627	98.5	92	10,970	98.4	273	14,546	98.8	122	-	100.0	437	438
P-n101-k4	681	99.6	127	281	99.6	1,055	1,253	99.4	371	-	100.0	1,154	1,155

considered. Column np of this table reports the number of instances in the corresponding class. For each method and for each class, the table shows the number of instances solved to optimality ($nopt$) and the average computing time in seconds for solving the instances (t_{AVG}).

The last two lines of Table 1 report the average computing times over all classes and the total number of instances solved by each method. The method of Baldacci et al. [2] was not able to solve to optimality 3 instances solved by Fukasawa et al. [5] and by our exact method. Taking the different computers used into account, Table 1 indicates that our method is on average faster than the method of Fukasawa et al. [5].

Table 2 reports a detailed comparison of the computational results of the four methods considered on a selected set of difficult CVRP instances. Column $Name$ indcates the name of the instance. z^* is the cost of the optimal solution. Columns %LB, t_{LB} and t_{TOT} report the average percentage ratio of the lower bound with respect to the optimal solution value, the average time in seconds for computing the final lower bound, and the average total time in seconds for solving the instance. For the exact method of Lysgaard et al. [7], "tl" indicates that the time limit has been reached, and for the exact method of Baldacci et al. [2], "-" denotes that the memory limit has been reached.

References

[1] R. Baldacci, E. A. Hadjiconstantinou, and A. Mingozzi. An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations Research*, 52:723–738, 2004.

[2] R. Baldacci, N. Christofides, and A. Mingozzi. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming Ser. A*, 115(2):351–385, 2008.

[3] R. Baldacci, A. Mingozzi, and R. Roberti. Solving the vehicle routing problem with time windows using new state space relaxation and pricing strategies. *Submitted for publication*, 2009.

[4] N. Christofides, A. Mingozzi, and P. Toth. Exact algorithms for the vehicle routing problem based on spanning tree and shortest path relaxation. *Mathematical Programming*, 10:255–280, 1981.

[5] R. Fukasawa, H. Longo, J. Lysgaard, M. Poggi de Aragão, M. Reis, E. Uchoa, and R.F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming Ser. A*, 106:491–511, 2006.

[6] M. Jepsen, B. Petersen, S. Spoorendonk, and D. Pisinger. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research*, 56(2):497–511, 2008.

[7] J. Lysgaard, A. N. Letchford, and R. W. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming Ser. A*, 100:423–445, 2004.

Comparative evaluation of Logit and Fuzzy Logic models of gap-acceptance behavior

Riccardo Rossi
Department of Structural and Transportation Engineering
University of Padova, Italy
Email: riccardo.rossi@unipd.it

Claudio Meneguzzer
Department of Structural and Transportation Engineering
University of Padova, Italy

Massimiliano Gastaldi
Department of Structural and Transportation Engineering
University of Padova, Italy

Gregorio Gecchele
Department of Structural and Transportation Engineering
University of Padova, Italy

1 Introduction

In studies of vehicular gap-acceptance behavior, the choice to accept or reject a gap of a certain size is generally considered the result of a driver decision process which includes, as inputs, subjective estimates of a set of explanatory variables, given specific objective factors. These subjective evaluations are usually affected by a high degree of uncertainty, which can be properly treated both by classical probabilistic models (e.g. Logit [1]) and by fuzzy system theory [2]. In this paper we propose a comparative analysis among these two approaches to the representation of gap-acceptance behavior based on experimental data collected at a priority junction. The method used to carry out this comparison is the so-called ROC (*Receiver Operating Characteristic*) curve analysis, which, to our knowledge, has never been applied in the area of transport modeling.

2 Experimental data

The field data used in the analysis are gap-acceptance observations (driver decisions) collected at a rural three-leg priority intersection. All observations relate to the right turn movement from a minor street controlled by a "yield" sign. The identification/estimation of both Fuzzy and Logit models has been carried out using the same 70% random subsample (calibration dataset) of the full sample (2.340 decisions, acceptance and rejections); the random subsample consisting of the remaining 30% of data has been used for model validation (validation dataset).

3 Logit model calibration

Several Logit models of gap-acceptance behavior were specified and estimated in this study using the HieLoW® and Gauss® programs; here we report the results obtained from one of them, indicated as GA_L (Tab. 1).

Model	Rho-square	Corrected Rho-square	Percent Right
GA_L	0,736	0,732	92,26%

Alternative specific constant (acceptance)	-6,602 (-17,72)
Time interval size	+0,99 (19,20)
Total delay	+0,02 (2,47)
Interval type	+2,25 (8,96)

Tab. 1 – Calibration results: goodness-of-fit indicators , parameter estimates and corresponding Student's t-statistics (within brackets).

The GA_L model includes, as explanatory variables, the size s of the time interval (gap or lag), the driver's total delay d on the minor approach (queuing delay plus stop-line delay), and the type of interval (represented by a dummy gl, which takes the value of one in the case of a lag and zero in the case of a gap). The estimated GA_L model has the following expression:

$$P_{acceptance} = \frac{1}{1 + \exp\left[-\left(-6,602 + 0,99 \cdot s + 0,02 \cdot d + 2,25 \cdot gl\right)\right]}.$$

In which, as expected, the acceptance probability increases with both the interval size s and the total delay d ; also, for given values of interval size and total delay, the lag-acceptance probability is higher than the gap-acceptance probability (Fig. 1).
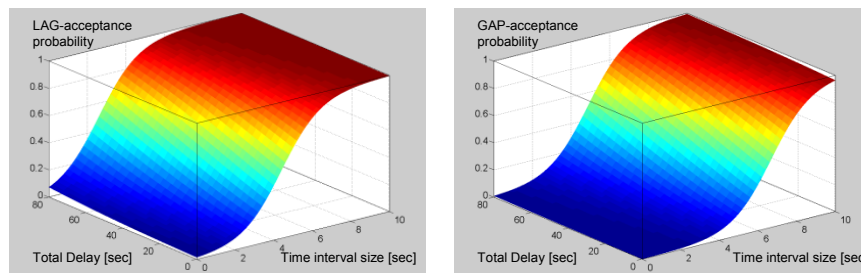
Fig. 1 – GA_L model. Lag and Gap acceptance surfaces as a function of time interval size and total delay.

4 Fuzzy model identification

For the identification of the membership functions of the premise and consequence fuzzy sets, and rules of inferences the so-called FPA (Fast Prototype Algorithm [3]) has been used. A satisfactory value of goodness-of-fit has been obtained. The fuzzy system knowledge base was characterized by six trapezoidal fuzzy sets in the domain of the time interval size, five trapezoidal fuzzy sets in the domain of the driver's total delay and by two "singletons" in the domain of the crisp variable representing the type of interval. A simple Mamdani-type method of inference has been adopted. The calibration process produced forty compensatory and two non-compensatory rules.

Acceptance surfaces obtained from the fuzzy model (GA_F) are shown in fig. 2.
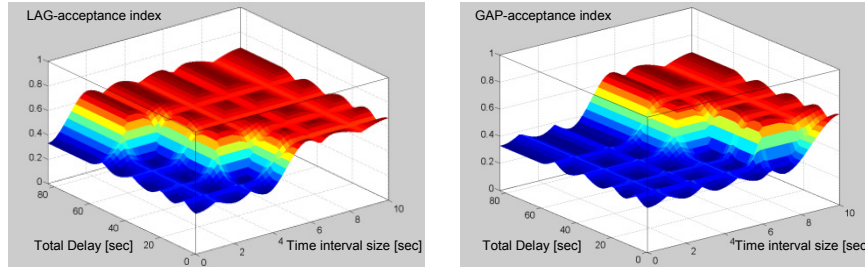
Fig. 2 – GA_F model. Lag and Gap acceptance surfaces as a function of time interval size and total delay.

In agreement with the GA_L (Logit) model, the GA_F model shows that the value of the acceptance index increases with the time interval size and with the total delay on the minor approach, and suggests that lag acceptance behavior is more aggressive than gap acceptance behaviour for given interval size and total delay.

5 Comparing the predictive ability of the models: preliminary results

The predictive ability of the two models has been tested by means of the *ROC curve analysis*, a method used in various research fields for evaluating and comparing the discriminatory power of models having binary outputs [4]. Apparently, this method has never been applied before in the area of transport modeling. The basic idea of ROC curve analysis may be explained by considering an experiment with only two possible outcomes, A and B (for example, gap acceptance and rejection in our application). Suppose we are using a model that predicts the outcome of the experiment based on a classification threshold ("cutoff"), and denote by:

True A: the situation in which the model predicts A and the actual outcome is A;

False A: the situation in which the model predicts A but the actual outcome is B;

True B: the situation in which the model predicts B and the actual outcome is B;

False B: the situation in which the model predicts B but the actual outcome is A;

Then we may define the probability of correctly identifying A as:

PCA = n. of *True A*/(n. of *True A* + n. of *False B*) = TPR

and the probability of correctly identifying B as:

PCB = n. of *True B*/(n. of *True B* + n. of *False A*) = TNR

Clearly, the discriminatory power of the model increases with both PCA and PCB. The ROC curve describes the relationship between PCA, called "sensitivity", and (1-PCB), called "1-specificity", for all possible classification thresholds. Since:

1-PCB = n. of *False A*/(n. of *True B* + n. of *False A*) = FNR

the ROC curve may be interpreted as describing the relationship between the "percentage of hits" and the "percentage of false alarms" obtained with the model. It is known that the area under the ROC

curve (AUC) is related to the accuracy of the model predictions, and increases with it; in particular, when this area is equal to one the model produces perfect forecasts, and when it is equal to 0.5 the model produces random forecasts (no discriminatory power). The AUC is equivalent to the Gini coefficient = 2*AUC-1, and also to the Mann–Whitney–Wilcoxon two-independent sample non-parametric test statistic [5]. Additional performance metrics adopted are also: FPR=1-PCA, Precision metric, that represents the percentage of correct acceptance prediction, the F-measure, that is the harmonic average of Precision and PCA, and the percent right (or accuracy), that is the percentage of correct predictions globally made.

Some preliminary results of the analysis are illustrated in Fig. 3. While the ROC curves (left-hand part of the figure) seem to suggest a slight superiority of the Logit model, a more complete analysis using additional statistics (shown in the right-hand part) indicates that neither model definitely dominates the other. A detailed examination of all these indicators will be presented in the full paper.
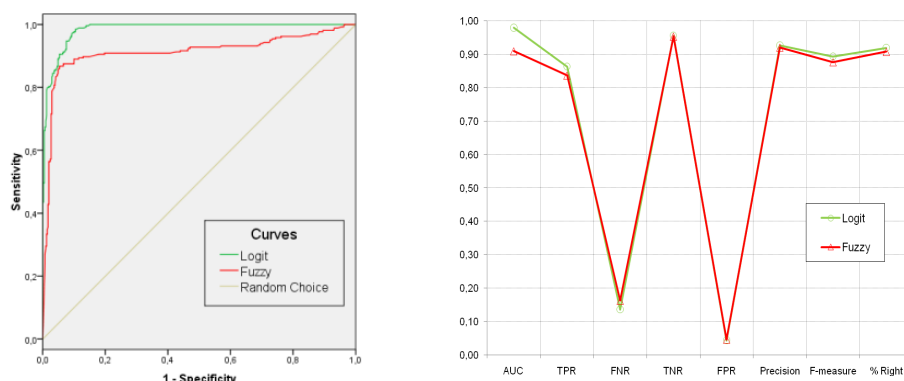
Fig. 3 – GA_L model vs. GA_F model. ROC curves and model performance comparison.

References

[1]   S. Teply, M. I. Abou-Henaidy and J. D. Hunt. "Gap acceptance behaviour – aggregate and Logit perspectives: Part 1", Traffic Engineering and Control 9, 474-482 (1997).

[2]   R. Rossi and C. Meneguzzer, "The effect of crisp variables on fuzzy models of gap-acceptance behaviour", Proceedings of the 13-th Mini-EURO Conference "Handling Uncertainty in the Analysis of Traffic and Transportation Systems", Bari (Italy), 240-252 (2002).

[3]   P.-Y. Glorennec. *Algorithmes d'apprentissage pour systems d'inférence floue*. Editions Hermès, Paris (1999).

[4]   T. Fawcett. "An introduction to ROC analysis". *Pattern Recognition Letters*, 27(8), 861–874 (2006).

[5]   J.A. Hanley and B.J. McNeil "The meaning and use of the area under a receiver operating characteristics curve", Radiology 143, 29-36 (1982).

Integrating Medium- and Short-Term Decisions in Airline Crew Planning

Michael Römer

Institute of Business Information Systems and Operations Research

Martin Luther University Halle-Wittenberg, Halle (Saale), Germany

Email: michael.roemer@wiwi.uni-halle.de

Taïeb Mellouli

Institute of Business Information Systems and Operations Research

Martin Luther University Halle-Wittenberg

1 Introduction

Airline crew planning can be viewed as a hierarchical planning process consisting of multiple stages ranging from long-term decisions such as planning the crew size to short-term crew scheduling. While Operations Research methods have successfully been applied at all stages, the short-term planning step of crew scheduling is the most active area of research (for an overview cf. [1]).

In this paper, we focus on the interdependencies between medium-term crew planning tasks and crew scheduling. Under the umbrella of medium- term planning, we subsume all planning tasks that affect the availability of a given set of crew members for crew scheduling. In particular, the assignment of absence periods (part-time and vacation leave) and the scheduling of non flying activities such as training, simulator and office fall into this category. In medium-term planning, the common approach to anticipate subsequent planning steps is to constrain the number of absent crew members (globally or by domicile).

While the number of available crews is certainly the most important consequence for crew scheduling, it is only an aggregated figure. In a more detailed view, medium-term planning affects the time-dependent crew availability by determining the gaps in the roster of each crew member that can be used for flying. In our ongoing research, we study the question whether it is possible and beneficial to anticipate crew scheduling in a greater detail in the medium-term planning stage. We present a modeling framework that integrates crew scheduling and medium-term planning tasks and provide some experimental results based on real world problem instances.

2 Modeling approach

Medium-term planning tasks can be modeled as network flow models with side constraints. For every crew member, a separate network is constructed (see Figure 1). It contains a timeline which includes nodes that represent the start and end events of the activities to be planned and arcs that represent "idle time" of the respective crew member. Activities with a fixed start and end time (e.g. already planned part-time absences, exact vacation requests) are represented by a single arc in the network. If the duration of the activity is fixed but the start time is variable (e.g. 14 vacation days in a given time period), several arcs are generated for alternative start times in conjunction with a bundle constraint. For activities with a variable duration, additional arcs are added as shown on the right in Figure 1. Note that the network structure of the model implicitly ensures that all planned activities are non-overlapping as the flow in a crew member network is limited to 1 and a flow unit is required for every activity.
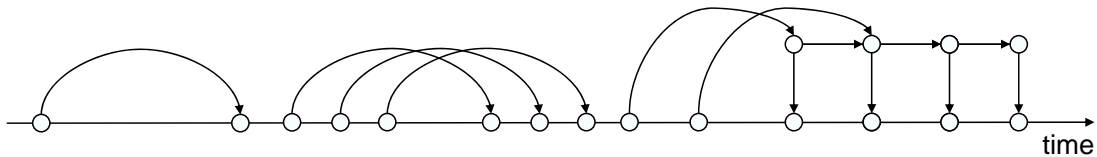
Figure 1: Example structure of crew member network used in medium-term planning

In order to apply this model structure to a specific medium-term planning task, problem-dependent side constraints can be added. We successfully applied this modeling approach in a decision support system for vacation planning that is in productive use at a medium-sized German airline, see [2]. *Autovacation* supports the planners in different planning stages by automatically processing and awarding the vacation requests of the crew members and by generating alternative proposals for not awarded requests while accounting for all contractual and company rules.

Our approach to an integration with the crew scheduling problem is based on an aggregated network flow model for the *crew pairing chain problem (CPCP)* proposed by Mellouli [3, 4, 5]. It forms the key part of a decision support system that is used for crew scheduling at our partner airline with 11 crew domiciles for several years. This model and the respective solution approach to crew scheduling can be contrasted to traditional set partitioning approaches to the crew pairing problem (CPP) (cf. [1]) in at least two ways: Firstly, instead of optimizing pairings separately, it simultaneously optimizes pairings and their sequencing into *pairing chains*, explicitly integrating aggregated capacities for pre-planned crew member activities and absences. Secondly, as the underlying network enforces all pairing-related contractual and governmental rules (e.g. EU-OPS) *by construction*, the model can directly be solved by a state-of-the-art mixed integer programming solver instead of resorting to a decomposition method.

The *pairing chain model* for the CPCP is a multicommodity flow model on a state-time network where a commodity represents an anonymous crew member from a certain crew domicile, an arc represents an activity or a connection between activities and a node corresponds to a crew state at a point in time. The computed optimal flow in the network can be decomposed into a set of *pairing chains* containing spaces for days off and for pre-planned activities. For every crew domicile there is a timeline consisting of arcs representing idle crew members and nodes corresponding to activity start and end events. Due to the state-expansion of the network, every deadhead and flight can occur several times in the network. Pre-planned activities are represented as arcs with a fixed flow. Additional side constraints ensure the regular distribution of days off and the balancing of flight hours among the crew domiciles.

In both networks described above, a flow unit represents a crew member. The main difference is that in the crew member network, a flow unit can be mapped to a concrete crew member whereas in the pairing chain network the flow is aggregated and therefore anonymous. To combine the pairing chain and the medium-term model, we first construct both networks separately. For every starting and ending activity of an individual crew member, we then add one node to the respective domicile timeline in the pairing chain network. The outgoing and incoming activity arcs of these nodes can be seen as copies of the corresponding arcs in the crew member network, carrying the *same* flow variables that represent the individual activities. Thus, in the mathematical model, these variables occur in the flow balance constraints of the corresponding timeline nodes in both networks. As a consequence, every individual activity requires one flow unit from *both* the aggregated and the crew member network to begin and releases these units when it is finished.

Note that by keeping the crew member timeline in the model instead of simply adding the crew member individual activities to the respective domicile timelines, an overlapping of of crew member activities is still implicitly avoided in the model. In addition to the integration of the network structure of the two models, some components of the pairing chain model (e.g. the constraints for the distributing of days off) that depend on the medium-term activities are adapted.

3   Computational results and outlook

The aim of our first experiments is to study the potential benefits of integrating medium-term planning with crew scheduling. For each problem instance, we compare the results of two different models: First, we use the standard pairing chain model (CPCP) where all medium-term activities are fixed. In the second model (CPCP+MT), we use our integrated approach in the following way: Every pre-planned vacation or off period that completely falls into the planning period can be shifted to a maximal extent of 3 days inside the planning period. In both models we restrict the maximum length of a pairing to 3 days to reduce model size and computation time.

We used several real world instances from our partner airline with a complete half month planning period. All models are generated and solved on a personal computer (Intel Core 2, 2.4 GHz, 2GB RAM) using single-threaded CPLEX 10.2. One typical instance for first officers has the following properties: 15 days, 41 aircrafts, 260 crew members distributed over 11 crew domiciles and 3546 flights. The following table summarizes some key features of the model and its solution:

model	columns	rows	gen. (min)	solve (min)	costs (%)	shiftable periods
CPCP	155842	40232	35:21	8:10	100	-
CPCP+MT	156545	40475	35:39	6:13	98.3	123

The generation of the pairing chain model takes much more time than the solution of the model itself. This is due to the complex steps to construct and aggregate the network for the pairing chain model and the need to check all relevant working rules. We solved additional comparable instances resulting in cost savings between 0.5 and 2.2 percent with similar running times.

As can be seen from the experimental results, our approach is computationally tractable for real-life instances of a moderate size. Moreover, the provided flexibility concerning the medium-term activities has a positive impact on the solution of the crew pairing chain problem. We plan to conduct further research by applying the presented modeling framework to different medium-term crew planning tasks. Furthermore, we think that our approach could be used in more general settings as it allows to integrate aggregated and disaggregated network flow models in a simple and efficient way. This could be valuable especially in cases where some aspects of the problem exhibit a strong combinatorial nature and where other aspects need to be modeled in a more detailed way.

References

[1] B. Gopalakrishnan and E. Johnson, "Airline crew scheduling: State-of-the-Art," *Annals of Operations Research*, vol. 140, pp. 305–337, Nov. 2005.

[2] M. Römer, "Medium-term crew planning: Process integration and models," in *AGIFORS Crew Management Study Group Conference*, (Paris), 2008. Electronic Proceedings.

[3] T. Mellouli, "A network flow approach to crew scheduling based on an analogy to a Train/Aircraft maintenance routing problem," in *Computer-Aided Scheduling of Public Transport* (S. Voss and J. Daduna, eds.), vol. 505 of *LNEMS*, pp. 91–120, 2001.

[4] T. Mellouli, "Scheduling and routing processes in public transport systems," *Professorial dissertation, University of Paderborn*, 2003.

[5] Y. Guo, T. Mellouli, L. Suhl, and M. P. Thiel, "A partially integrated airline crew scheduling approach with time-dependent crew capacities and multiple home bases," *European Journal of Operational Research*, vol. 171, no. 3, pp. 1169–1181, 2006.

A customer-centric solution approach to cyclic inventory routing

Birger Raa

Department of Management Information and Operations Management

Ghent University, Tweekerkenstraat 2, 9000 Gent, Belgium

Email: birger.raa@ugent.be

Wout Dullaert

Institute of Transport and Maritime Management Antwerp

Antwerp University, Keizerstraat 64, 2000 Antwerpen, Belgium

Email: wout.dullaert@ua.ac.be

Antwerp Maritime Academy, Noordkasteel Oost 6, 2030 Antwerp, Belgium

1   Problem description

In this paper, we consider a distribution system consisting of a single warehouse from which a set of geographically scattered customers S with constant demand rates d_i ($i \in S$) has to be replenished. Because of the constant consumption rates, a cyclic distribution pattern is most appropriate, in which the time between consecutive deliveries to a customer is always the same. Even if the consumption rates are somewhat variable, the cyclic approach is still valid as the variability can be buffered with a limited amount of safety stock at the customers [5].

In setting up the cyclic distribution for this system, the trade-off between distribution and inventory costs has to be made. Distribution costs consist of fixed vehicle dispatching costs (φ_0 per route), drop-off costs (φ_i per delivery, $i \in S$) and variable transportation costs (δ per km). Further, because the cyclic plan is to be repeated over an infinite time horizon, the vehicle fleet is also variable and a fixed cost ψ is incurred per vehicle used. Inventory costs on the other hand, consist of holding costs (η_i per unit per day, $i \in S$) for the cycle stock at the customers. A routing and an inventory management problem therefore have to be solved simultaneously, hence the problem is called the cyclic inventory routing problem.

Replenishing a customer with a large quantity results in a larger inventory cost, compared to when the replenishment quantity would be smaller, but the customer needs to be visited less

often so distribution costs could decrease. However, the large quantity also means that there is less space left in the vehicle to replenish other customers in the same route. Therefore, a larger delivery quantity can also lead to an increase in distribution costs if an extra vehicle needs to be dispatched from the warehouse.

Furthermore, a single vehicle can also be used for several different routes. E.g. the same vehicle can be used to make two different full-day routes that both have to be repeated every two days, but not to make two different full-day routes where one has to be repeated every two days and the other every three days. This means that the cycle times of the different routes also need to be aligned to limit the number of vehicles required and the corresponding fixed vehicle costs [3].

From this discussion, it is obvious that the cyclic inventory routing problem is very complex, because it involves making the following interrelated decisions:

- What is the cycle time (and delivery quantity) for each of the customers?
- On which days are the customers being replenished?
- How are the different deliveries on any given day allocated to different routes?
- Which vehicle makes which routes?

The cyclic inventory routing problem can be viewed as a generalization of the periodic vehicle routing problem (PVRP) that (i) includes an inventory cost component and (ii) considers the cycle times of the customers as a decision variable instead of a given parameter.

2 Literature review

In the literature, several solution approaches for the cyclic inventory routing problem are presented. Many of these approaches use the so-called 'fixed partition policies', as introduced by Anily and Federgruen (1990) [2]. In these approaches, customers are clustered geographically, and all customers in a cluster are always replenished together in a single route.

Viswanathan and Mathur (1997) [6] take a more general approach by adopting a stationary nested joint replenishment policy. 'Nested' means that if the replenishment interval of a given customer is larger than that of another customer served by the same vehicle, the former is a multiple of the latter. In other words, a customer that is assigned to a route, is not necessarily visited during every iteration of that route.

Aghezzaf et al. (2006) [1] are the first to take fleet sizing into account. In their approach, a single vehicle may make multiple routes, albeit all with the same frequency. This is further generalized in the approach of Raa and Aghezzaf (2009) [4], who allow a vehicle to make multiple routes with different frequencies. For each vehicle, a base cycle time is determined, and the cycle time of each of the routes of the vehicle is an integer multiple of this base cycle time.

3 Solution approach

In this paper, two new solution methods for the cyclic inventory routing problem are developed.

The first solution method generalizes the approach of Raa and Aghezzaf (2009) by relaxing the following two restrictions that are inherent to that approach: (i) all customers in a route have the same cycle time, and (ii) each route is always made by the same vehicle. The first restriction can be relaxed by introducing the 'nestedness' concept of Viswanathan and Mathur (1997). The second restriction can be relaxed by, instead of assigning a route to a vehicle, assigning each iteration of a route separately to a vehicle.

In all solution methods described so far, every customer is always visited in the same route. These solution methods all construct solutions by assigning customers to routes and then determining the best cycle time for each route to minimize total cost rates. The second solution method presented in this paper steps away from this 'route-centric' approach and takes a more 'customer-centric' approach. This new method determines the cycle time for each customer, then assigns customers to days, and builds routes for each day afterwards.

4 Illustrative example

Consider the 4-customer example in Figure 1.
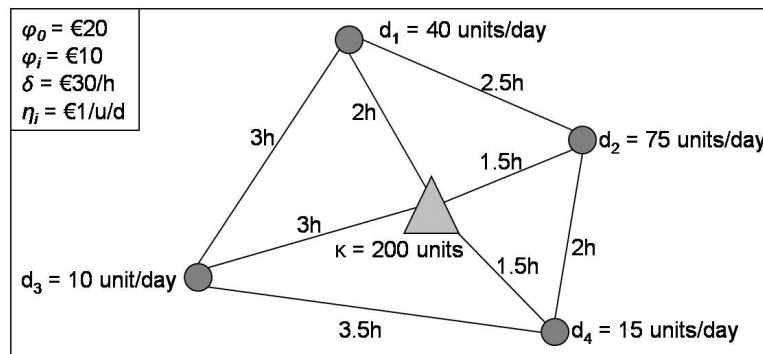
Figure 1: Illustrative example

With a route-centric method, customers 2 and 4 are replenished in a 5-hour route that is repeated every two days. Customers 1 and 3 are replenished in an 8-hour route that is repeated every four days. The cost rate of this solution is 355.00 euro per day.

2,4		1,3	2,4	
Day 1		Day 2	Day 3	Day 4

With a customer-centric method, a solution with a cost rate of only 350.83 euro per day can be obtained.

Customer 2 is replenished every two days, customers 1 and 4 are replenished every three days, and customer 3 is replenished every six days. On day 1 of the resulting six-day cycle, customers 1 and 2 are visited in separate routes because of the limited vehicle capacity. However, a single vehicle can make both short routes. On day 4, customer 1 is replenished together with customer 3, whereas customer 1 has a separate replenishment on day 1. This would also possible be in the route-centric method if the nestedness is incorporated (customer 3 is visited only every second iteration of the route). Customers 2 and 4 are in separate routes on days 1, 2 and 3, while they are in a joint route on day 4. This is only possible in the customer-centric approach.

2	1		4		2		1,3	2,4	
Day 1			Day 2		Day 3		Day 4	Day 5	Day 6

5   Computational experiments

A wide range of computational experiments are performed on the datasets of Raa and Aghezzaf (2009) [4] as well as on periodic vehicle routing benchmarks from the literature, to assess (i) the performance of the different solution approaches, and (ii) the value of the additional flexibility that the two proposed solution methods offer, i.e. their potential for further cost savings.

References

[1] E. H. Aghezzaf, B. Raa, and H. Van Landeghem. Modeling inventory routing problems in supply chains of high consumption products. *European Journal of Operational Research*, 169(3):1048–1063, 2006.

[2] S. Anily and A. Federgruen. One warehouse multiple retailer systems with vehicle-routing costs. *Management Science*, 36(1):92–114, 1990.

[3] B. Raa and E.-H. Aghezzaf. Aligning frequencies in cyclic delivery scheduling. In *Proceedings of the 3rd Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA)*, 2007.

[4] B. Raa and E.-H. Aghezzaf. A practical solution approach for the cyclic inventory routing problem. *European Journal of Operational Research*, 192(2):429 – 441, 2009.

[5] B. Raa and W. Dullaert. Cyclic inventory routing under demand uncertainty. In *Proceedings of the Fourth International Workshop on Freight Transportation and Logistics (Odysseus)*, Çesme, Izmir, Turkey, 26-29 May 2009.

[6] S. Viswanathan and K. Mathur. Integrating routing and inventory decisions in one-warehouse multiretailer multiproduct distribution systems. *Management Science*, 43(3):294–312, 1997.

An integrated solution method for order batching and picking

Birger Raa

Department of Management Information and Operations Management

Ghent University, Tweekerkenstraat 2, 9000 Gent, Belgium

Email: birger.raa@ugent.be

Wout Dullaert

Institute of Transport and Maritime Management Antwerp

University of Antwerp, Keizerstraat 64, 2000 Antwerpen, Belgium

Email: wout.dullaert@ua.ac.be

Antwerp Maritime Academy

Noordkasteel Oost 6, 2030 Antwerp, Belgium

1 Introduction

This paper is based on the work of Louis (2009) [6] and investigates the optimization of order picking efficiency in picker-to-part warehouses with multiple blocks by integrating operational decisions related on batch sizing, the composition of batches and the routing of order pickers to collect the batches. Although there is an extensive literature on each of these three individual aspects of order picking efficiency, integrated approaches are scarce to non-existing. When comparing routing algorithms, most authors assume that the batches of orders to be picked on the same route are already composed. We therefore start by briefly explaining the order batching and sorting process before highlighting existing routing research and identifying the need for an integrated solution approach.

1.1 Order batching

If orders are small relative to the capacity of the picker, it can be more efficient from a travel time/distance perspective to group orders instead of picking them individually (single order picking, discrete picking, pick-to-order). However, order batching will increase the processing time of orders, which can call for separate or prioritized batching of urgent orders.

The main order batching strategies are proximity or time-window based. Proximity batching will assign orders to a batch based on its storage position compared to the position of the other orders in the batch. Exact approaches have been developed for problems with a limited number of orders. For more realistic settings, heuristic approaches have been proposed, of which the vast majority consists of savings or seed-based construction heuristics. For the latter, addition rules, based on comparing the aisle numbers in the batch and the aisle number in the order under consideration, are quite efficient and therefore quite popular. More recently, data mining and metaheuristic approaches are being explored.

Time window batching models the picking process as a queuing system to determine the required number of pickers for a given service level. Time window batching is recommended if (i) a large number of orders is to be processed, (ii) information on the orders is not known beforehand, (iii) a limited number of pickers is available. Of the different approaches, only Won and Olafsson (2005) [11] present a joint order batching and picking algorithm.

Although Roodbergen and De Koster (2001a) [8] and Vaughan and Peetersen (1999) [10] have already pointed out that dividing the warehouse into several blocks (by introducing aisles and cross aisles) can have a positive effect on the travel time of a picker, most seed and savings heuristics are specifically designed for single block warehouses and only Le-Duc and De Koster (2007) [5] address batching in a 2-block warehouse. As such, there is no evidence on the performance of existing batching algorithms in real-life multi-block warehouses.

1.2 Composing orders

As a picker visits the storage locations of the various items on his list, he can sort these items into separate orders as he travels from one location to the next (sort-while pick) or the orders can be composed at a sorting station at the end of the picking route (pick-and-sort). Order batching reduces travel time, but increases administrative and sorting costs to compose the indivial orders. Order splitting amongst different order pickers can further reduce distance travelled, but it comes at the expense of higher order sorting costs (see e.g. De Koster et al., 2007 [2]).

1.3 Routing order pickers

As order pickers spend about 50% of their time travelling from one storage location to the next, the savings potential of improving order picker routing is significant. As a result, routing of order pickers receives more scientific attention than any other operational warehouse decision problem.

The problem of sequencing and routing order pickers in conventional multi-parallel-aisle systems classifies as a Steiner Travelling Salesman Problem [1]. The Steiner nodes indicate all "crossing nodes" between two or more aisles and the objective is to visit all order locations (non-Steiner nodes) once, at minimum overall cost. For this NP-hard Steiner TSP, exact algorithms only exist

for warehouses with at most three cross aisles, while for other warehouse types literature provides a selection of dedicated construction heuristics such as S-shape (a.k.a. traversal) heuristic, largest gap and combined heuristics.

In Theys et al. (2010) [9] we examined how reformulating and solving the order picking problem as a classical TSP leads to performance improvements compared to existing dedicated heuristics. Average savings in route distance amounted to 47% when using the LKH (Lin-Kernighan-Helsgaun) TSP heuristic, with solutions deviating only 0.1% from optimum on average. Computational testing further revealed that by complementing existing construction heuristics with a first-accept 2-opt local search operator, average distance savings of 10 to 42% can be obtained.

This paper wants to integrate (i) batching, (ii) sequencing, (iii) route construction and (iv) improvement methods to provide a comprehensive approach to operational decision making in multiple block warehouses. Moreover, we want to make a critical assessment of the impact of existing approaches on the overall cost of operating a warehouse.

2 Computational experiments

To compose the batches of orders, four seed heuristics are developed: first-come-first-serve, two extensions of the single-block selection rules of Ho and Tseng (2006) [4] and a new selection rule based on the number of blocks in which the items of an order are located. The possibility of splitting orders over multiple pickers is also considered. Sequencing procedures are used to determine picking sequences. For the shortest path routing, the order sequence is obtained by a sequential insertion heuristic. For the S-shape routing heuristic, the quality of the initial sequence is irrelevant as routes are built from scratch. Previous research on routing order pickers has illustrated the value of local search to improve solution quality [7, 9]. We therefore consider a swapping operator for items reminiscent of [7] and a swapping operator to relocate orders between batches. In the computational experiments all relevant combinations of these procedures are combined into 11 integrated approaches which are evaluated on an extensive set of benchmark instances based on order features suggested in [3, 4].

3 Conclusions

To the best of our knowledge, this paper is the first to provide integrated approaches for order batching, order sequencing and routing of order pickers. Computational experiments show that (i) both the capacity of the picker and the number of blocks in the warehouse have a significant impact on the performance of the algorithms, (ii) batching algorithms should consider both travel distance and capacity utilization, and (iii) improvement heuristics have a significant impact on solution quality. Order splitting is shown to reduce travel distances by up to 50%, depending on

the integrated method used, illustrating that these cost savings will have to be traded off with the resulting increase in sorting costs. At the conference, the different procedures, computational experiments and results will be discussed in detail and directions for further research will be presented.

References

[1] G. Cornuéjols, J. Fonlupt, and D. Naddef. The traveling salesman problem on a graph and some related integer polyhedra. *Mathematical Programming*, 33:1–27, 1985.

[2] R. de Koster, T. Le-Duc, and K. J. Roodbergen. Design and control of warehouse order picking: A literature review. *European Journal of Operational Research*, 182(2):481–501, 2007.

[3] R. de Koster, K. J. Roodbergen, and R. van Voorden. Reduction of walking time in the distribution center of de bijenkorf. *New Trends in Distribution Logistics*, 480:215–234, 1999.

[4] Y. C. Ho and Y. Y. Tseng. A study on order-batching methods of order-picking in a distribution centre with two cross-aisles. *International Journal of Production Research*, 44(17):3391–3417, 2006.

[5] T. Le-Duc and R. M. B. M. de Koster. Travel time estimation and order batching in a 2-block warehouse. *European Journal of Operational Research*, 176(1):374–388, 2007.

[6] P. Louis. *Een geïntegreerde oplossingsaanpak voor order-picking in magazijnbeheer*. Master's thesis. Ghent University, 2009.

[7] P. A. Makris and I. G. Giakoumakis. k-interchange heuristic as an optimization procedure for material handling applications. *Applied Mathematical Modelling*, 27(5):345–358, 2003.

[8] K. J. Roodbergen and R. de Koster. Routing methods for warehouses with multiple cross aisles. *International Journal of Production Research*, 39(9):1865–1883, 2001.

[9] C. Theys, O. Brysy, W. Dullaert, and B. Raa. A tsp-based solution framework for routing order pickers in warehouses. *European Journal of Operational Research*, 200(3):755–763, 2010.

[10] T. S. Vaughan and C. G. Petersen. The effect of warehouse cross aisles on order picking efficiency. *International Journal of Production Research*, 37(4):881–897, 1999.

[11] J. Won and S. Olafsson. Joint order batching and order picking in warehouse operations. *International Journal of Production Research*, 43(7):1427–1442, 2005.

On the Optimum Expansion of Airport Networks

Miguel G. Santos

Department of Civil Engineering

University of Coimbra, Portugal

Email: msantos@dec.uc.pt

António P. Antunes

Department of Civil Engineering

University of Coimbra, Portugal

1 Introduction

Air traffic has grown at an average annual rate of more than 4.0 percent over the last three decades, giving an important contribution to the development of the world economy [1]. The growth in air traffic has not been accompanied with a comparable increase in runway and terminal capacity. As a consequence, the number of delayed flights has been increasing every year, particularly in the largest airports.

Airport congestion problems can be and are being dealt with at various levels (air transportation authorities, airports, airlines) and in many different forms. In the short-term, demand management measures such as slot allocation systems and congestion pricing can play an important role [2]. However, in the long term, air traffic can only keep growing at significant rates if the capacity of existing airports is expanded and/or new airports are built.

The literature dealing with airport expansion and/or location problems at the network level is quite scarce. This is especially true for the optimization-based literature. Indeed, to our best knowledge, [3] is the only paper where an optimization model is applied to this kind of problems.

In this paper, we present an optimization model for assisting air transportation authorities in their strategic decisions regarding the expansion of an airport network. The model determines in a comprehensive manner the best expansion actions to implement for each airport (or multi-airport system) while complying with a given budget. The objective is to maximize the revenue passenger kilometers (RPK) travelled in the network, taking into account the capacity of airports and the impact of travel costs upon demand.

2 Model formulation

The airport network capacity expansion (ANCE) problem can be formulated as a bi-level discrete network design problem: the upper level determines the optimal set of expansion actions to apply in

order to maximize RPK, and the lower level determines the link and airport flows subject to user equilibrium conditions.

The following notation has been used to formulate the ANCE model: N = set of airports in the network; L = set of links in the network; Mj = set of expansion actions applicable to airport j; R_{jk} = set of routes available for travelling between airports j and k; L_{jkr} = set of links in route r connecting airports j and k; N_{jkr} = set of airports in route r connecting airports j and k; d_{jk} = distance between airports j and k; q_{jk} = number of trips between airports j and k; z_j = final capacity of airport j; s_j = initial capacity of airport j; g_{jm} = capacity increase in airport j if expansion action m is applied; y_{jm} =1 if expansion action m is applied to airport j, and 0 otherwise; e_{jm} = expenditure for applying expansion action m to airport j; b = budget available for expansion actions; v_{jkr} = number of trips on route r connecting airports j and k; u_{jk} = number of trips on the link connecting airports j and k; w_j = flow in airport j; c_{jkr} = travel cost on route r connecting airports j and k; c_{jk} = average cost between airports j and k.

The upper level problem can be formulated as follows:

$$\max \sum_{j \in N} \sum_{k \in N} d_{jk} q_{jk} \tag{1}$$

$$z_j = s_j + \sum_{m \in \mathbf{M}_j} g_{jm} y_{jm}, \forall j \in \mathbf{N} \tag{2}$$

$$\sum_{m \in \mathbf{M}_j} y_{jm} \le 1, \forall j \in \mathbf{N} \tag{3}$$

$$\sum_{j \in \mathbf{N}} \sum_{m \in \mathbf{M}_j} e_{jm} y_{jm} \le b \tag{4}$$

$$z_j \ge w_j, \forall j \in \mathbf{N} \tag{5}$$

The objective function (1) expresses the maximization of the RPK. Constraints (2) define the capacity of airport j as the sum of its initial capacity and the capacity increase due to the expansion action applied. Constraints (3) ensure that at most one expansion action will be applied to each airport. Constraints (4) ensure that the total expenditure must comply with the budget available for expansion actions. Constraints (5) establish that the capacity of the airports must be able to accommodate the traffic flow.

The lower level problem can be formulated as follows:

$$q_{jk} = \alpha \left(p_j p_k \right)^{\mu} \phi_{jk}{}^{\varphi} c_{jk}{}^{-\beta}, \forall j, k \in \mathbf{N} \tag{6}$$

$$v_{jkr} = \frac{e^{-\gamma c_{jkr}}}{\sum_{p \in \mathbf{R}_{jk}} e^{-\gamma c_{jkp}}} q_{jk}, \forall j, k \in \mathbf{N}, r \in \mathbf{R}_{jk} \tag{7}$$

$$c_{jkr} = \sum_{l \in \mathbf{L}_{jkr}} C_1(d_l, u_l) + \sum_{n \in \mathbf{N}_{jkr}} \left[C_2 \left(\frac{w_n}{z_n} \right) + f_n + x_n \right], \forall j, k \in \mathbf{N}, r \in \mathbf{R}_{jk} \tag{8}$$

$$w_j = \sum_{l \in \mathbf{L}_j} u_l, \forall j \in \mathbf{N} \tag{9}$$

$$u_l = \sum_{j \in \mathbf{N}} \sum_{k \in \mathbf{N}} \sum_{r \in \mathbf{R}_l} v_{jkr}, \forall l \in \mathbf{L} \tag{10}$$

Constraints (6) define the number of passengers travelling between j and k as a function of the size of the cities, p_j and p_k, the competition between travel modes, represented by the split function $_{jk}$, and the average travel cost. Constraints (7) define the number of passengers travelling in route r between j and k as a function of the travel cost for that route. Constraints (8) define the cost of traveling in route r connecting j and k, which is given by the sum of the link costs for the set of links on that route and the airport costs for the set of airports on that route. The link costs are assumed to increase with distance and to decrease with the total amount of traffic (due to economies of scale). Airport costs are given by the sum of a fixed cost, f, a variable cost dependent on the airport occupancy rate, and a congestion tax, x. Constraints (9) define the traffic in the airports, and constraints (10) define the traffic in the links. α, μ, φ, β, and γ are statistical calibration parameters.

3 Model solving

The model presented above is extremely difficult (if not impossible) to solve to exact optimality. Thus, a heuristic algorithm was developed, based on similar algorithms applied to road network design models [4]. The upper level model, solved with a greedy-type algorithm, gives tentative expansion actions. With the new airport capacity values, the lower level model is solved in order to compute the link and airport flows, which are, in turn, used to evaluate new tentative expansion actions. This iterative process is repeated until there are no expansion actions within the given budget that can improve the best solution found.

The set of experiments made up to now to assess the effort required to solve the model is limited to instances with up to 30 airports. The average time required to find the solution on a top-market PC was about 20 seconds for 10-airport instances, 75 seconds for 20-airport instances, and more than one hour for 30-airport instances. As one could expect the computation effort increases sharply with instance size.

4 Application example

The type of results that can be obtained through the application of the optimization model are now illustrated for a small test instance. Consider a region with six cities, each one served by an airport. Both the population and the location of the cities were randomly determined. The set of expansion actions are: $g_j = \{40, 60, 70, 80, 100, 120\} \times 10^3$ passengers/day. If no budget is assigned to expansion actions, the total RPK is equal to 332.3×10^3 pax$\times$km/day, and three airports are very congested – see Figure 1 (left). The top left airport, in particular, is very congested, and the cost for passengers using this airport includes a congestion tax equal to 50\$. If 23×10^8\$ are assigned to expansion actions, the three congested airports are expanded and the total RPK increases to 399.8 pax$\times$km/day (+20.0%) – see Figure 1 (middle). With half of this budget, only two airports are expanded and the total RPK is equal to 384.8 pax$\times$km/day (+15.8%) – see Figure 1 (right).
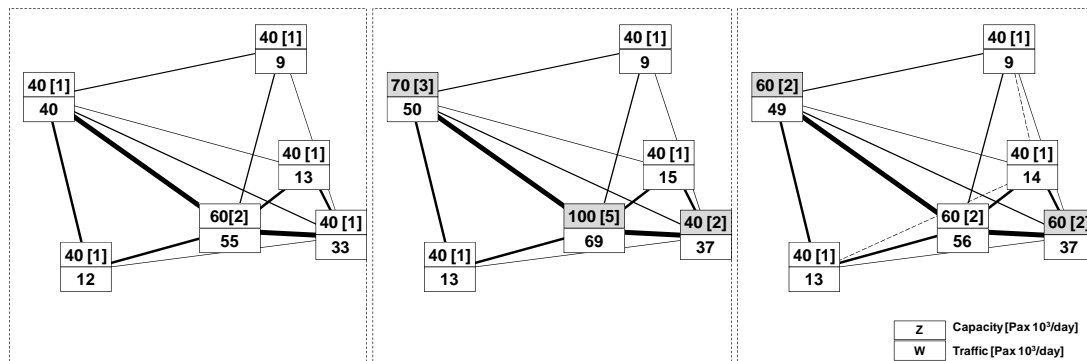
Figure 1 – Solution obtained for: b=0\$ (left), b=23×10^8\$ (middle), b=12×10^8\$ (right)

5 Conclusion

The objective of the model presented in this paper is to find the set of expansions actions to apply to an airport network that maximizes the number of revenue passenger kilometers travelled in the network, taking into account the capacity of the airports and the impact of travel costs upon demand.

As illustrated for a small-size network, the model can be of great practical utility. An application to the US is currently under way, and another application to the EU is sought. In the future, some new features will be incorporated in the model. In particular, we plan to deal with the construction of new airports in addition to the expansion of the existing ones, consider an objective more relevant from the economic standpoint, and include equity, robustness and flexibility issues in the model.

References

[1] M.A. Ishutkina and R.J. Hansman, "Analysis of the Interaction between Air Transportation and Economic Activity: A Worldwide Perspective", Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA (2009).

[2] T. P. Fan and A. R. Odoni, "A Practical Perspective on Airport Demand Management", *Air Traffic Control Quarterly* 10(3), 285-306 (2002).

[3] O. Saatcioglu, "Mathematical Programming Models for Airport Site Selection", *Transportation Research B* 16(6), 435-447 (1982).

[4] J. d D. Ortúzar and L. G. Willumsen, *Modelling Transport*, New York: John Wiley (1991).

Solution methods for the dynamic dial-a-ride problem with stochastic requests and expected return trips

Michael Schilde, Karl F. Doerner, Richard F. Hartl

University of Vienna, Department of Business Administration

Bruenner Strasse 72, 1210 Vienna, Austria

{michael.schilde, karl.doerner, richard.hartl}@univie.ac.at

1 Introduction

This work was motivated by the problem the Austrian Red Cross faces with patient transportation. Patients can call in to place transportation requests between their home location and a hospital or practitioner's. Each such request is assigned time windows for the pickup and the delivery location. Some of these requests are known in advance (i.e. a patient placed them the day before), whereas others arise as the day progresses. The requests have to be served by a fixed fleet of vehicles based at a common depot location. The aim is to design vehicle routes that accommodate all requests and minimize tardiness, number of vehicles used, and route duration. Additionally, these routes must not violate a set of additional constraints such as vehicle capacity and ride time limitations.

The problem of transporting persons from a given pickup location to a delivery location is commonly modeled as a dial-a-ride problem (DARP). Much effort has been spent in finding adequate solution methods for this problem class (see e.g. [3] for a recent survey). However, the problem at hand imposes an additional interesting aspect, as some information about the occurrence of future events might be available. Let us think about a person who is regularly transported from home to the hospital for dialysis and back home afterwards. Such a treatment normally consumes a predictable amount of time, no matter whom the person is or when it takes place. Thus, it might be possible to forecast the return transport and use this information when designing the vehicle routes. The same procedure would of course make sense for every type of return transport if the information is available. Note that for the dynamic transportation requests from the patients' home location to the hospital no stochastic information will be exploited. But for the possible return trips from the hospital to the patients' home available stochastic information will

be used for the route planning in the applied sophisticated solution procedures. This stochastic information is generated from the expected treatment duration for standard treatments. There exist different approaches that considered the dynamic aspect of the problem (see e.g. [1]). To our best knowledge the integration of stochastic information in the dynamic DARP has not been studied so far.

The contribution of this work is to show the adaptation and applicability of existing solution techniques for the stochastic dynamic DARP. Therefore, we compared three different solution approaches to tackle the dynamic and stochastic aspects of the problem at hand: Variable Neighborhood Search (VNS, [4]), Multiple Plan Approach (MPA, [2]), and Multi Scenario Approach (MSA, [2]). Section 2 gives a brief description of the implemented solution methods. The work concludes with a short summary of the results and an outlook on future research in Section 3.

2 Solution methods

We implemented three different solution strategies for this problem: VNS, MPA, and MSA. All three methods were adapted to the requirements of the problem at hand. Initial solutions are generated using a greedy approach. All approaches simulate a 10 hour workday within 1 hour of run time.

2.1 Variable neighborhood search

The concept of the well known VNS metaheuristic was first introduced by Mladenović [4]. The implementation we used in this work is based on four shaking operators similar to the ones used by Parragh et al. [5]: move, swap, chain, and zero split. Extensive tests have shown that the local search step included in the traditional VNS structure does not improve the resulting solutions for the specific problem at hand and we therefore decided not to use this step in our implementation. Our VNS implementation accepts only improving solutions as a new current incumbent solution. Deteriorating solutions and infeasible solutions are never accepted. New requests are inserted into the current incumbent solution whenever they arise.

Move: the move operator randomly selects one route from which one randomly selected request is removed. This request is re-inserted into the route in which it causes the lowest increase in the objective function. For the κth neighborhood, this is repeated κ times.

Swap: the swap operator randomly selects two routes. For the κth neighborhood, a sequence of up to κ consecutive requests is removed from each route, starting at a randomly selected position. The removed requests are then re-inserted into the other route at the best position, respectively.

Chain: the chain operator randomly selects an origin and a destination route. For the κth neighborhood, a sequence of up to κ consecutive requests is removed from the origin route, starting

at a randomly selected position. The removed requests are then re-inserted into the destination route at the best position, respectively. Next, the destination route is used as origin route, a new destination route is randomly selected, and the procedure is repeated κ times.

Zero split: the zero split operator randomly selects one route and determines all its zero split positions (positions between two services, at which the vehicle is empty). For the κth neighborhood, up to κ consecutive sequences bounded by zero split positions are selected. The corresponding requests are removed from the route and re-inserted into the route in which they cause the lowest increase in the objective function, respectively.

2.2 Multiple plan approach

MPA was proposed by Bent and Van Hentenryck [2] for the vehicle routing problem with time windows. It maintains a pool of solutions which is updated whenever new solutions are found, new requests arise, or a solution becomes incompatible with the current decisions made. To continuously generate new solutions from the solutions present in this pool, the VNS method as described before was used.

At every point in time, one of the solutions in the pool is selected as current incumbent solution. Extensive testing showed, that selecting the best available solution leads to much better results for the specific problem at hand than the originally proposed consensus function (see [2]). New requests are inserted into all solutions in the pool whenever they arise. At runtime, MPA continuously checks if vehicle departures in all solutions are consistent with decisions made in the current incumbent solution.

2.3 Multi scenario approach

MSA was also proposed by Bent and Van Hentenryck [2] and is a logical extension of MPA for situations in which additional information about the underlying data can be exploited. Based on MPA, the algorithm tries to create so called "scenarios" during execution, meaning, that the algorithm creates a set of yet unknown future requests by sampling the given distributions. This information is used when searching for additional solutions. This is aimed at creating gaps in the schedule of known events that, at a later point in time, can be used to accommodate "real" requests when they arise.

By sampling the known distributions, MSA generates a set of possible future return transports for already known transports. These additional sampled requests are inserted into the current VNS starting solution. If VNS finds a new solution that does not yet exist in the pool of found solutions, it removes all sampled requests and stores the resulting solution in the pool. This way, all solutions in the pool contain only known requests. The remainder of the MSA structure is the same as for the MPA described before.

3 Summary

Based on real world data provided by the Austrian Red Cross, we created 40 sets of test instances consisting of 25 samples of the respective distribution parameters. Our results show, that all three methods are effective solution approaches to the problem at hand. The results obtained by our simple modified VNS procedure prove to be as good as the ones obtained by the more sophisticated MPA and MSA methods.

Future research will shed light on the question, if incorporating information about daytime dependent travel times can improve the solution quality of MSA when compared to the solutions of VNS and MPA (not using this information).

Acknowledgements

References

[1] A. Attanasio, J.-F. Cordeau, G. Ghiani, and G. Laporte. Parallel tabu search heuristics for the dynamic multi-vehicle dial-a-ride problem. *Parallel Computing*, 30(3):377–387, 2004.

[2] R. W. Bent and P. Van Hentenryck. Scenario-based planning for partially dynamic vehicle routing with stochastic customers. *Operations Research*, 52(6):977–987, 2004.

[3] J.-F. Cordeau and G. Laporte. The dial-a-ride problem (DARP): Variants, modeling issues and algorithms. *4OR*, 1(2):89–101, 2003.

[4] N. Mladenović. A variable neighborhood algorithm: A new metaheuristic for combinatorial optimization. In *Abstract of papers presented at Optimization Days*, page 112, Montréal, 1995.

[5] S. N. Parragh, K. F. Doerner, and R. F. Hartl. Variable Neighborhood Search for the Dial-a-Ride Problem. *Computers & Operations Research*, In Press, Accepted Manuscript:–, 2009.

Integrating routing decisions in public transportation models

Marie Schmidt

Institute of Numerical and Applied Mathematics

University of Göttingen

Email: m.schmidt@math.uni-goettingen.de

Anita Schöbel

Institute of Numerical and Applied Mathematics

University of Göttingen

Email: schoebel@math.uni-goettingen.de

1   Passenger-oriented planning using OD-data

According to [2] the planning process in public transportation includes several phases such as strategic planning (e.g. network design), tactical planning (as line planning or timetabling), operational planning (e.g. vehicle scheduling) and real time control. Many models for these planning steps exist, some of them focussing on the costs, others on the benefit for the passengers. To this end, data about the passengers is necessary and has to be included in the models, in particular in the objective functions.

Many approaches assume a two-step procedure: in a first step, the data about the passengers is distributed over the public transportation network using traffic-assignment procedures. In line planning, e.g., one ends up with so called *traffic loads* w_e giving an (approximate) number of passengers who want to use edge e. Also in timetabling and delay management it is usually assumed that the number of passengers who want to take a certain vehicle at a certain station is known beforehand. This reduces the complexity of the models but is not realistic from a practical point of view since the routing decisions of the passengers depend on the lines or timetables which are not known before the optimization problem is solved.

Only a few approaches integrate the routing decisions. In line planning this has been done recently in [5, 9, 1, 7]. In delay management, a first integrated model allowing a re-routing of passengers has been presented in [3]. The timetabling models we are aware of assume that the

passengers' weights are fixed beforehand.

In this paper we reformulate some of the common models for line planning, timetabling and delay management taking into account origin-destination data and including the routing of the passengers in the optimization process. It turns out that the integration of line planning and routing is NP-hard even in linear graphs and even with only one OD-pair. One ends up with a type of (NP-hard) resource-constrained shortest path problem. Nevertheless, some special cases will be identified in which the integration of routing and line planning can be solved in polynomial time.

However, including the routing decisions of the passengers in aperiodic timetabling turns out to be as efficiently solvable as aperiodic timetabling itself, if the start and destination event of every OD-pair is known. If only the stations (with or without time window) are given for the OD-pairs, the integration of aperiodic timetabling and routing is again NP-hard.

Delay management is by itself an NP-hard problem, so the NP-hardness of its integration with passengers' routing is not surprising. If only one OD-pair is considered, the integrated problem can be solved in polynomial time.

In the following we will speak of vehicles which are meant to be busses or trains depending on the transportation mode under consideration.

2 Line planning with OD-pairs

The goal of line planning is to determine a set of *lines* $\mathcal{L}$ together with their frequencies f_l for all $l \in \mathcal{L}$. There exist cost-oriented and passenger-oriented objective functions, where the latter may consider the number of direct passengers or the traveling time of the passengers. A few recent approaches allow that passengers are freely routed (see [9, 1, 7]). Given a set of OD-pairs OD, we investigate the following model

$$\min \sum_{(i,j) \in OD} W(i,j) \quad \text{s.t.} \quad \sum_{l \in \mathcal{L}} f_l \, cost_l \leq Budget$$

in which $cost_l$ is an approximation of the (variable) costs of line l and $W(i,j)$ contains the traveling time for OD-pair (i,j). This traveling time includes the riding time and a penalty for every transfer. It is hence a shortest path in a suitable defined network which depends on the choice of the lines and is in this way integrated in our model. In the constraint we require that the costs of the line system do not exceed a given budget.

Integrating the routing decisions leads to a difficult problem: Even in linear networks with lines all having the same speed and where the penalties for transfers do not depend on the line or on the station where the transfer takes place, the problem is NP-hard [9]. We show that for the case of only one OD-pair, the problem is still NP-hard if we abandon any of the assumptions made above,

that means if the public transportation network is not linear, the lines do not have the same speed, or the penalties for transfers depend on the lines and stations. However, we present two different solution approaches for special cases of the situation described above: one linear algorithm which can be used if the costs b_l for the lines are all equal and there is either only one OD-pair or the penalties of the transfers are all 0, and a polynomial algorithm for the case of one OD-pair.

3 (Aperiodic) Timetabling with OD-pairs

Given the line plan, the timetabling problem searches for the arrival and departure times for all lines at all stations. To this end, one models every arrival and every departure of a vehicle as an event. The resulting events $\mathcal{E}$ are connected by driving, waiting, or changing activities. If π_i denotes the time of event i, and $a = (i, j)$ an activity linking event i and event j, a timetable $(\pi_i), i \in \mathcal{E}$ is feasible if every activity $a = (i, j)$ satisfies that

$$l_a \leq \pi_j - \pi_i \leq u_a$$

for some given lower and upper bounds l_a and u_a on the duration of activity a. Given a fixed number of passengers w_a for every activity a, the goal is to minimize the sum of traveling times. The problem can be solved efficiently by linear programming [8].

In our model we do not start with such fixed weights w_a but with a set of OD-pairs which can be freely routed through the network. We are able to show that the complexity of the timetabling does not change if we now minimize the sum of traveling times for the given OD-pairs as long as the start event and the destination event of every OD-pair is known. The idea of our approach is to add a virtual activity for every OD-pair and minimize the traveling times of these new activities. Surprisingly, the problem gets NP hard if not the departure event but only the departure station is known. Also here, the case of one single OD-pair can be solved in polynomial time.

We remark that for periodic timetabling even the feasibility problem is NP-hard (for an overview about aperiodic timetabling see [6] and references therein), hence also the integration of periodic timetabling and routing is NP-hard even in the case of one single OD-pair.

4 Delay Management with OD-pairs

Delay management is a part of real-time control and concerns the decision if a connecting service should wait for a delayed feeder vehicle or if it should depart on time. As in timetabling the problem has been treated with fixed passengers' weights w_a for every activity a. Most of its variants are NP-hard (see [4]). In a recent study [3], the problem has been investigated allowing a re-routing of the passengers in order to reach their destinations as soon as possible. In its general form it is

NP-hard if re-routing is included in the delay management problem, even if all OD-pairs have the same origin or the same destination. However, we are able to show that integrating the routing decision of only one OD-pair with the wait-depart decisions of delay management can be treated by a modification of Dijkstra's algorithm making sure that delays of vehicles are correctly accounted in the future in $O(n^2)$ time in the number n of given events.

References

[1] R. Borndörfer, M. Grötschel, M. E. Pfetsch, "A column generation approach to line Planning in Public Transport", *Transportation Science 41*, pp. 123-132, 2007.

[2] G. Desaulniers and M. Hickman "Public Transit", in *Transportation*, G. Laporte and C. Barnhart (eds.), Handbooks in Operations Research and Management Science 14, pp. 69-127, 2007

[3] T. Dollevoet, D. Huisman, M. Schmidt, A. Schöbel "Delay Management with Re-Routing of Passengers" in *9th Workshop on Algorithmic Methods and Models for Optimization of Railways*, Dagstuhl Seminar Proceedings, 2009.

[4] M. Gatto, R. Jacob, L. Peeters, A. Schöbel, "The Computational Complexity of Delay Management", in *Graph-Theoretic Concepts in Computer Science: 31st International Workshop (WG 2005)*, D. Kratsch (ed.) LNCS 3787, 2005.

[5] G. Laporte, A. Marín, J.A. Mesa and F.A. Ortega, "An integrated methodology for rapid transit network design", in *Algorithmic Methods for Railway Optimization*, LNCS 4359, Springer, pp. 187-199, 2007.

[6] C. Liebchen, "Periodic Timetable Optimization in Public Transport", Dissertation, TU Berlin, dissertation.de – Verlag im Internet, 2006.

[7] K. Nachtigall and K. Jerosch, "Simultaneous Network Line Planning and Traffic Assignment", in *8th Workshop on Algorithmic Approaches for Transportation Modeling, Optimization, and Systems*, M. Fischetti and P. Widmayer (eds), 2008.

[8] R. Tyrrell Rockafellar, "Network flows and monotropic optimization", John Wiley & Sons, Inc., 1984.

[9] A. Schöbel and S. Scholl, "Line Planning with Minimal Travel Time", in *5th Workshop on Algorithmic Methods and Models for Optimization of Railways*, Dagstuhl Seminar Proceedings 06901, 2006.

A Vehicle Routing Problem with Time Windows and Driver Familiarity

Michael Schneider, Christian Doppstadt, Bastian Sand

Chair of Business Information Systems and Operations Research

University of Kaiserslautern, Email: schneider@bisor.de

Andreas Stenger, Michael Schwind

Chair of IT-based Logistics, Goethe University Frankfurt

1 Introduction

This work is motivated by an industry project with a French small package shipping company. Their routing operations have several distinguishing characteristics (cp. [1]), of which explicit consideration of driver familiarity with routes and customers is deemed crucial by our industry partner. One way to achieve driver familiarity benefits is to have the same driver visit the same service territory on each delivery trip. Thus, the driver becomes acquainted with the territory and the customer locations therein and is able to serve the customers more efficiently. The disadvantage of such an approach is the flexibility forfeited by having fixed delivery areas and driver assignments. Faced with varying demand, this yields route configurations that are suboptimal concerning the total traveled distance. This tradeoff between driver familiarity benefits and routing flexibility is investigated in comprehensive simulation studies by [2]. To balance the tradeoff, [3] introduce the concepts of "cell", "core area" and "flex zone" and explicitly consider driver learning in their vehicle routing model. Moreover, they provide an extensive review of related literature.

However, both works neglect the existence of time windows, which strongly conflicts with recent practical developments. Our industry partner states that up to 60% of their orders are time-definite, which is consistent with the industry statistics given in [4]. If time windows are considered, routing flexibility is not only needed to achieve distance-efficient route configurations but also to fulfill customer delivery time requirements. Thus, the value of routing flexibility increases, which is likely to have significant negative effect on the solution quality of any approach based on (partially) fixed delivery areas. By fixing delivery areas, methods implicitly put too much emphasis on driver familiarity while neglecting the value of routing flexibility.

To be able to find the optimal tradeoff, our routing method forgoes any fixing of delivery areas.

Our vehicle routing model (see Section 2) explicitly considers driver knowledge by means of driver specific travel and service times. Thus, drivers have an incentive to stay in familiar areas due to shorter driving and service times while maintaining their flexibility. We develop an ant colony optimization method specifically tailored to the described routing problem (see Section 3). The numerical studies to investigate the performance of our approach are described in Section 4.

2   Model

The objective of our vehicle routing problem with time windows and driver familiarity model (VRPTWDF) is to minimize the total driving and service time given a fixed number of vehicles (drivers). This number can be computed by heuristic methods like e.g. [5] or is predetermined by real-world requirements. The two main differences from [5]'s VRPTW are driver specific service times s_{ik} for node i and driver k and driver specific travel times c_{ijk} for driver k when traveling from i to j. These differences necessitate slight modifications to capacity and time window constraints. Motivated by our industry partner, we consider hard time windows since cost penalties seem unsuitable to cover the long-term effects of missing a time window, e.g. the loss in reputation caused by dissatisfied customers. We add working hour constraints to comply with legal requirements.

3   Solution Method

Good solutions to our problem trade off driver familiarity and flexibility, but will probably employ each driver roughly in the area he is most familiar with. This knowledge is incorporated into our solution method, an ant colony system (ACS) based on [6]. ACS provides a natural way of solving our problem; each ant is associated with a single driver and creates a route based on driver-specific heuristic information and pheromone values that are traded off against each other. Our ACS comprises the following steps: solution generation, local probability update, a local search step and global probability update.

Solution generation   For route creation, we use a parallel procedure, where each ant starts at the depot and works in a separate process. This means that ants are competing to visit nodes, which is resolved by means of a first-come, first-served rule. Let $J_k(i)$ denote the set of nodes that ant k can visit after node i without violating time window, capacity or cycle constraints. If $j \notin J_k(i)$, the probability of ant k taking edge (i, j) is zero, otherwise it is given by:

$$p_k(i, j) = \alpha \cdot \tau(i, j) + \beta \cdot \eta_k(i, j) + \gamma \cdot \vartheta(j).$$

$\tau(i, j)$ is the pheromone trail on edge (i, j). The heuristic information $\eta_k(i, j)$ is driver-dependent. It is defined as the reciprocal of the sum of the driver's travel time c_{ijk} from i to j plus his service time value s_{jk} at node j. To prioritize nodes that are critical in terms of fulfilling a given deadline

or a capacity constraint, we use attractiveness values $\vartheta(j)$. α, β, γ are parameters weighing the described measures. We apply the pseudo-random-proportional rule of [6] to determine the ratio between search space exploration and exploitation. We allow infeasible solutions, where nodes have not been visited after all ants return to the depot. Infeasible solutions are penalized.

Local probability update   After tours are created, we perform the local pheromone update described in [6] plus the following update of node attractiveness for critical nodes:

$$\vartheta(j)_{new} = \begin{cases} \vartheta(j)_{old} + s_0 & \text{if } j \notin V_{visited} \\ \vartheta(j)_{old} - \frac{(|V| - |V_{visited}|) \cdot s_0}{|V_{visited}|} & \text{else.} \end{cases}$$

$V_{visited}$ is the set of visited nodes. Parameter $s_0 \in [0, 1]$ weights the strength of the attractiveness update. Solution generation and local probability update are run a given number of times. The best feasible solution is used as input for the local search step described next. If no feasible solution is found, the local search step is omitted.

Local search   Our local search is based on [7], who present an efficient tabu search method for solving VRPTW. To speed up the search and guide it towards promising regions, we move or exchange nodes only between spatially close routes and/or try to shift nodes to the route of a driver that is familiar with this node. To this end, we restrict possible exchange candidates and the routes that exchange them as follows. For each driver, we determine a polygon connecting the outermost nodes of the route returned by ACS and a polygon connecting the outermost of all nodes that the considered driver is familiar with. Now, all nodes that belong to other routes but lie within the union of nodes enclosed by the two polygons and a radial area around the depot (cp. [3]) qualify as exchange candidates. Since this strongly reduces candidates, it does not seem reasonable to use operators that simultaneously consider several nodes. We only use operators that exchange few nodes, like e.g. relocation (one node is shifted from one route into another).

Global probability update   Based on the solution found by local search, we perform the global pheromone update described in [6]. In early iteration steps, the results of ACS and tabu search may differ significantly In order not to put too much emphasis on the tabu search solution, we use a low pheromone decay parameter that increases after each iteration step.

4 Numerical Studies & Outlook

The goal of our numerical studies are twofold. First, we create a set of VRPTWDF benchmark instances to study the performance of our solution method for single instances. Parameter choice is inspired by small package shipping characteristics (see [1], [3], [4]) and discussions with our industry partner. To judge solution quality, we use the commercial solver CPLEX and put special effort on determining lower bounds.

Second, we consider a series of δ consecutive days on which we solve VRPTWDF instances. To

link consecutive days, we use a learning model similar to the one presented in [3]. It describes how drive and service times have to be updated at the end of each day. We compare the performance of our approach to the fixed delivery area method described in [8]. Moreover, we investigate the following questions for the series of solutions produced by our ACS method: What are the shapes of the areas visited by each driver in the δ days? How strong is the overlap between neighboring areas? If we declare the resulting areas as (possibly overlapping) fixed areas, how do they perform compared to our ACS and [8]'s method?

References

[1] R. Wong, "Vehicle Routing for Small Package Delivery and Pickup Services", in *The Vehicle Routing Problem: Latest Advances and New Challenges*, B.L. Golden (ed), 475-485, Springer, New York, 2008.

[2] M.A. Haughton, "The Efficacy of Exclusive Territory Assignments to Delivery Vehicle Drivers", *European Journal of Operational Research* 184(1), 24-38 (2008).

[3] H. Zhong, R.W. Hall and M. Dessouky "Territory Planning and Vehicle Dispatching with Driver Learning", *Transportation Science* 41(1), 74-89 (2007).

[4] A.M. Campbell and B.W. Thomas, "Runtime Reduction Techniques for the Probabilistic Traveling Salesman Problem with Deadlines", *Computers & Operations Research* 36(4), 1231-1248 (2009).

[5] Y. Nagata and O. Bräysy, "A Powerful Route Minimization Heuristic for the Vehicle Routing Problem with Time Windows", *Operations Research Letters* 37(5), 333-338, 2009.

[6] M. Dorigo and L.M. Gambardella, "Ant Colony System: A Cooperative Learning Approach to the Traveling Salesman Problem", *IEEE Transactions on Evolutionary Computation* 1(1), 53-66 (1997).

[7] J.F. Cordeau, G. Laporte and A. Mercier, "A Unified Tabu Search Heuristic for Vehicle Routing Problems with Time Windows", *Journal of the Operational Research Society* 52(8), 928-936, 2001.

[8] K.F. Wong and J.E. Beasley, "Vehicle routing using fixed delivery areas", *Omega* 12(6), 591-600, 1984.

Anisotropic second-order models and associated fundamental diagrams

Schnetzler B.

Génie des REseaux de Transport et Informatique Avancée

Institut National de Recherche sur les Transports et leur Sécurité – Paris Est

Le Descartes 2, 2 rue de la Butte Verte, 93166 Noisy-le-Grand Cedex, France

Email: bernard.schnetzler@inrets.fr

Louis X.

Génie des REseaux de Transport et Informatique Avancée

Institut National de Recherche sur les Transports et leur Sécurité – Paris Est

1 Introduction

The first-order traffic model is a rough traffic model assuming an homogenenous traffic, but some second-order hyperbolic macroscopic traffic flow models assume an heterogeneous traffic. For instance the traffic is a mixture of personnal cars and trucks. Traffic heterogenenity explains positive transferences within the congested part [1]. Moreover when taking into account this heterogeneity a better traffic forecasting is expected. The first-order model [2] is based on the vehicle conservation equation $\partial_t \rho + \partial_x(\rho v) = 0$ and a fundamental diagram expressing the dependence between speed and density. Second-order hyperbolic models do not need an explicit reference to a fundamental diagram, which is replaced by a second equation. The Riemann invariant associated to this equation can take the place of the fundamental diagram. Anisotropic second-order models were revived ten years ago [3][4][5]. These anisotropic second-order models enter within the frame of a general formulation expressed by the "Generalized Second-Order Model (GSOM)" [6]: assuming that I is the second Riemann invariant, the second equation comes from $d_t I = 0$. Then the second variable $y = \rho I$ is carried by the traffic, where I depends on either a vehicle characteristics or a driver's attribute. Therefore a traffic which carries different values of invariant is a mixture. Within such a framework, the couple of variable is $(\rho, y = \rho I)$ and the speed is deduced from these two variables; this is the main difference between these models and previous models, such as the Payne's model or more generally gas-kinetic based traffic models, which depend on variables (ρ, v).

• The reinterpreted ARZ (Aw-Rascle-Zhang) model is explained by the intention of the driver $I = v - v_e(\rho)$, who wants to travel at a higher or lower speed than the equilibrium speed. A priori, any fundamental diagram $v_e(\rho)$ is admissible.

• The delayed acceleration model [7] assumes that some vehicle within a queue accelerates after a constant delay with respect to its predecessor. This hypothesis is admissible within the congested phase only. It yields the invariant definition $I = \rho^{-1} - v\tau$, where τ is the delay. Such a modeling produces linear congested fundamental diagrams such as $I = \rho_{\max}^{-1}$.

• For comparison with the delayed acceleration model, the invariant of the Colombo's model is rewritten depending on variables (ρ, v). Let $I = \rho^{-1} - \rho_M v\tau/(\rho_M - \rho)$ be the second Riemann invariant where ρ_M and τ are two parameters. As the delayed acceleration model, this model applies to the congested phase only. Such a modeling produces convex or concave congested fundamental diagrams, depending on the invariant value.

These three formalizations mean that each driver or vehicle carries the value of one parameter (the value of the invariant) of a generic fundamental diagram which is described by the analytical expression of the invariant. Now a practical use of such models must ensure that the diagrams match real data; generally speaking, some combination of parameters and invariant values are not admissible.

2 Fundamental diagrams

A fundamental diagram $Q(\rho)$ is a mathematical function which fits a set of points. Because the real traffic is a mixture and data are mean measurements, this function describes a mean behavior. Considering real data, there are many fundamental diagram patterns. But though there is a large variability some general properties are expected. Some parameters, which are mean values, are universal: the free-flow speed, the critical density, the capacity and the maximum density. The function is concave and the outflow tends to zero when the density tends towards zero or the maximum density; these rules apply wether the traffic is homogeneous or a mixture.
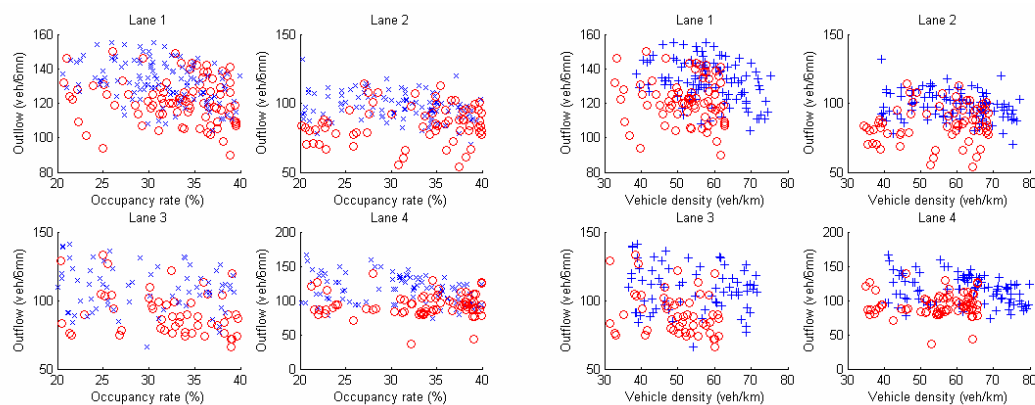
Fig. 1. Outflow dependence on the vehicle length (four lanes highway).

Dots are for high truck percentages (length greater than the mean length plus half of the standard deviation); crosses are for low truck percentages (length less than the mean length minus half of the standard deviation).

Considering current second-order models, the traffic is a mixture but it is assumed that each part depending on the same value of the invariant is homogeneous. In what follows, we will assume that the traffic is a mixture depending on a vehicle characteristics and not on the driver's behavior. Because a physical characteristics is observable and more constant than a behavioral factor, the first interpretation is better. Our first traffic assumption is that the main physical characteristics is the vehicle class of which the two possible values are *personnal car* and *truck*; this class is deduced from the vehicle length. Thus, in what follows, the invariant value should depend on the vehicle length.

Double magnetic loop provides three measurements: the outflow, the occupancy rate and a mean vehicle length. The vehicle length distinguishes between personnal car and truck and the observed mean vehicle length depends on the ratio of personnal cars to trucks. Measurements taken on the A1 highway, between Roissy Airport and Paris, show that the outflow depends on the mean vehicle length (see fig. 1).

3 Admissible invariants

Admissible invariants are proposed for the three models cited above. Of course the pattern of the fundamental diagram should not be the only one constraint to be taken into account.

When using the Aw-Rascle-Zhang model (see fig. 2), the outflow at the interface is $Q(\rho) = Q_e(\rho) + \rho I$ where $Q_e(\rho)$ is the mean fundamental diagram and $I = v - v_e(\rho)$ is a constant translation. However such a modeling shows two drawbacks. First, there is an undesirable outflow discontinuity when $\rho = \rho_{max}$. Second, because the needed property $\forall \rho \in [\rho_c, \rho_{max}]: Q'(\rho) < 0$, such modeling does not allow the use of any fundamental diagram $Q_e(\rho)$ and any value I. To solve these problems, the ARZ model is generalized: the invariant definition $I = v - v_e(\rho)$ is replaced by $I = v - v_e(\rho)/\rho(\rho_{max} - \rho)^n$ with $n \geq 1$. This new definition means that the intention vanishes when the density tends towards either zero or the maximum density.
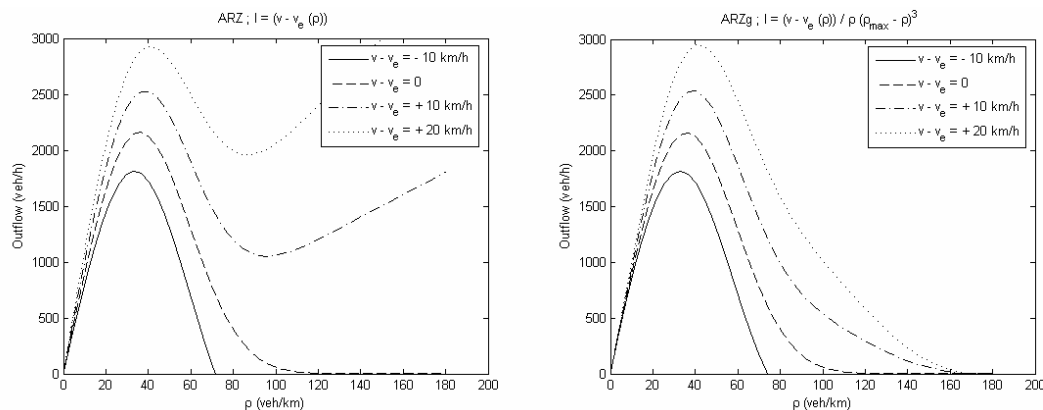
Fig. 2. Translated exponential fundamental diagram(s) by the ARZ model and its generalization.

When using the delayed acceleration model (see fig. 3), the invariant identifies to the inverse of the maximum density, i.e. the vehicle length ρ_x^{-1}. To avoid confusion ρ_{max} denotes the mean maximum density and ρ_x denotes the maximum density associated to some vehicle class. Such a modeling yields pattern the width of which increases with the density. So an alternate modeling can be used: considering the expression $I = \rho^{-1} - v\tau$, given that τ is a constant and $I = \rho_x^{-1}$, the expression $\tau = \left(\rho^{-1} - \rho_x^{-1}\right)v^{-1}$ is equivalent. Thus a new second equation comes from the property $d_t\tau = 0$. Though mathematical expressions are different, the new equation is equivalent to the previous one, and gives us a different parametrization of the fundamental diagram.
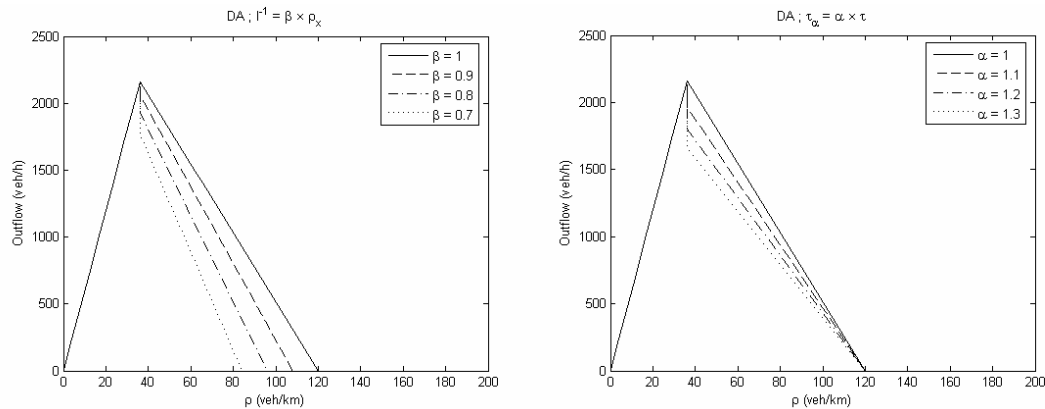
Fig. 3. Fundamental diagram(s) of the delayed acceleration model.

References

[1] J.H. Banks, "An investigation of some characteristics of congested flow", *Transportation Research Record*, 1678, 128-134 (1999).

[2] M.H. Lighthill and G.B. Whitham, "On kinematic waves II: a theory of traffic flow on long crowded corridors", *Proceedings of the Royal Society* A229, 317-345 (1955).

[3] A. Aw and M. Rascle, "Resurrection of second-order models of traffic flow", *SIAM Journal of Applied Mathematics*, 60 (3), 916-938 (2000).

[4] H.M. Zhang, "A non equilibrium traffic model devoid of gas-like behavior", *Transportation Research Part B* 36, 275-290 (2002).

[5] R. M. Colombo, "A 2 × 2 Hyperbolic Traffic Flow Model", *Mathematical and Computer Modelling* 35, 683-688 (2002).

[6] J.P. Lebacque, S. Mammar and H. Haj-Salem, "Generic second-order traffic flow modeling", *Proceedings of the 17th International Symposium on Transportation and Traffic Theory*, 749-770 (2007).

[7] B. Schnetzler and X. Louis, "A multiphasic traffic flow model", New Trends in Model Coupling, Paris, September 2009.

The "Adaptive Smoothing Method" with Spatially Varying Kernels: ASM-svK

Thomas Schreiter (corresponding author),

Yufei Yuan, Hans van Lint and Serge Hoogendoorn

Transport and Planning, Civil Engineering and Geosciences

Delft University of Technology, The Netherlands

1 Introduction

This paper presents an offline traffic state filter, which estimates macroscopic quantities like speeds and flows, given raw traffic data provided by dual-loop detectors, for instance. The basis of this method is the "Adaptive Smoothing Method" (ASM) by Treiber and Helbing [1]. The ASM smooths the raw data over space and time to remove measurement noise and to estimate traffic quantities between the observation points. The output is thus a continuous speed and a continuous flow map.

The rationale behind the ASM is founded in kinematic wave theory [2, 3]; empirical observations show that the characteristics of traffic, like flow, speed and density, travel along the traffic. Following from kinematic wave theory and the fundamental diagram, if the traffic is in congestion, then the characteristics travel upstream, usually with a speed of about $c_{\mathrm{cong}} = -20\,\mathrm{km/h}$. In the ASM, the traffic data are therefore smoothed along a rhomboid-shaped area

$$\phi(x) = \exp\left(-\frac{|x|}{\sigma} - \frac{|t - \frac{x}{c_{\mathrm{cong}}}|}{\tau}\right)\,, \tag{1}$$

with the main axes parallel to this characteristic congested wave speed c_{cong}, as shown in Figure 1. The larger the distance between a detector location and the filter point, the smaller is the share of the detector information in the filter result. There is a similar smoothing kernel for the case of free-flow traffic; the characteristics travel downstream with a speed of about $c_{\mathrm{free}} = 80\,\mathrm{km/h}$. The ASM filter calculates a weighted average of the congested and free-flow filter estimates, in which the weight is derived from the speed data. Further details are described in [1].

In order to filter the raw data, the ASM uses smoothing kernels of fixed size. The ASM does not take the underlying road conditions into account, especially geographic road discontinuities like on-ramps. This can lead to misestimations of the flow. Since vehicles enter the highway at this on-ramp, the flow downstream of it is higher than upstream of it. The ASM, however, smooths the
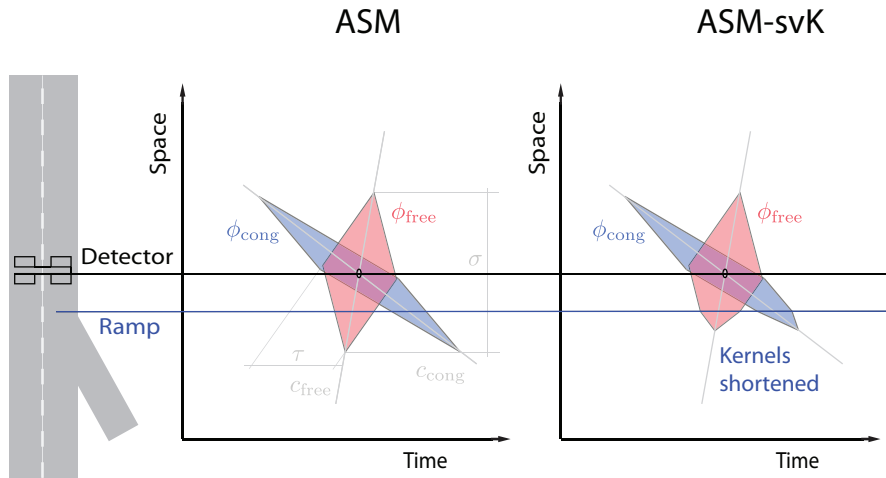
Figure 1: Comparison of the ASM [1] with the proposed ASM-svK

information observed upstream of this on-ramp into the part downstream of it. The traffic flow downstream is thereby underestimated, and the traffic flow upstream is overestimated.

To solve this problem, we propose in this paper to adapt the kernel functions ϕ (1) of the Adaptive Smoothing Method. In this method, called the ASM-svK, the discontinuities influence the shape of the kernel functions, so that the information from the far side of a discontinuity have a lesser impact, as illustrated in Figure 1. The ASM-svK estimates speeds and flows in the vicinity of road discontinuities more accurately than the original ASM. In addition, this method can be applied to temporal discontinuities as well, for example at accidents or bridge openings.

The remainder of this extended abstract explains the proposed ASM-svK in Section 2 and gives an experimental setup and its results in Section 3.

2 Spatially Varying Kernels

We propose to change the kernel functions (1) in such a way that road discontinuities influence the filter result, as shown in Figure 1. To every filter point x_{filt}, a specific kernel function

$$\phi_{x_{\text{filt}}}(x) = \alpha_{x_{\text{filt}}}(x - x_{\text{filt}}) \cdot \phi(x) \tag{2}$$

is assigned, where $\alpha_{x_{\text{filt}}}$ is a road geometry specific weighting function.

Consider, one wishes to estimate the traffic state (e.g. the speed) upstream of a discontinuity h, such as an on ramp. Clearly, observations upstream this on-ramp should be weighted heavier than observations downstream of this on-ramp. Accordingly, the weighting function $\alpha_{x_{\text{filt}}}$ in (2) should increase up to location of the discontinuity x_h and decrease fast downstream of it. This can be

achieved as follows:

$$\alpha_{x_{\text{filt}}}(x) = \begin{cases} \psi_h(x) & \text{if } x > x_h \\ 1 & \text{else} \end{cases} \tag{3}$$

with the discontinuity function

$$\psi_h(x) = \exp\left(\frac{-|x - x_h|}{\sigma}\right) . \tag{4}$$

In case of multiple discontinuities, the weighting functions (3) are generalized to

$$\alpha_{x_{\text{filt}}}(x) = \prod_h \begin{cases} \psi_h(x) & \text{if } \left[(x_h > x_{\text{filt}}) \leftrightarrow (x > x_h)\right] \\ 1 & \text{else} \end{cases} . \tag{5}$$
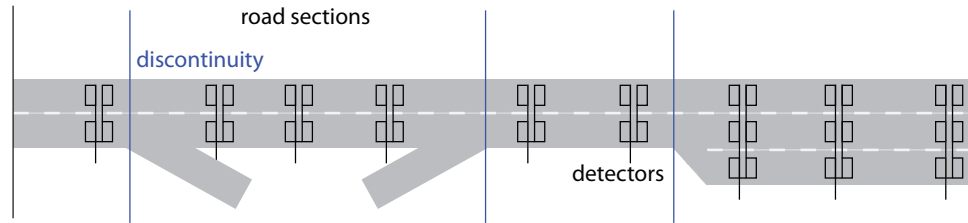
(a) Discontinuities caused by ramps and a lane opening, dividing the road into sections
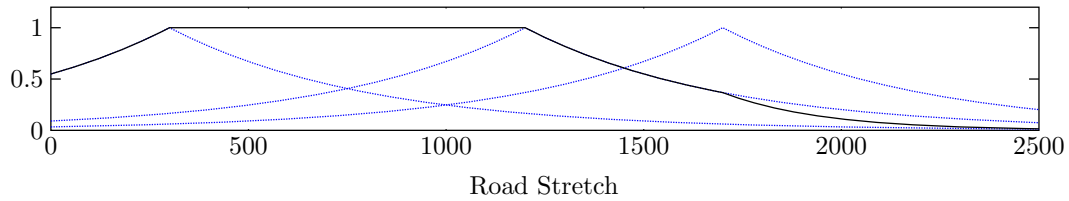
(b) Discontinuity kernels ψ (blue) and kernel factor $\alpha_{x_{\text{filt}}}$ (black) for filter point $x_{\text{filt}} = 1000$
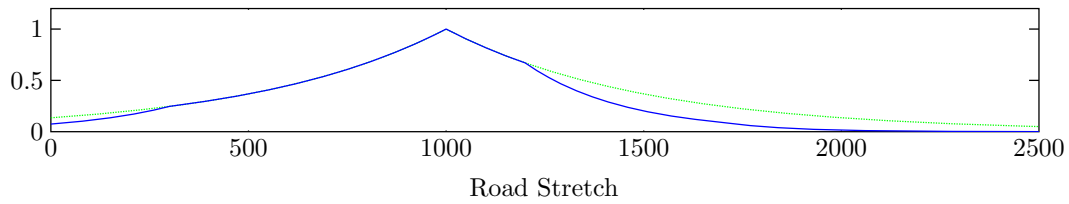
(c) ASM-svK Smoothing Kernel $\phi_{x_{\text{filt}}}$ (blue) for filter point $x_{\text{filt}} = 1000$ in comparison with ASM Kernel ϕ (green)

Figure 2: Spatially dependent kernel of the ASM-svK, caused by road discontinuities

Figure 2 shows an example of a road stretch with discontinuities at locations $x_1 = 300$, $x_2 = 1200$ and $x_3 = 1700$ (Figure 2a). Let the filter point be at $x_{\text{filt}} = 1000$. Figure 2b shows the three discontinuity functions ψ_h (4) and the kernel weighting function $\alpha_{x_{\text{filt}}}$ (5). The resulting kernel function $\phi_{x_{\text{filt}}}$ (2) (Figure 2c) is equal to or less than the kernel function ϕ of the original ASM. If the filter point and the detector lie within the same road section, then the kernel is not changed. In contrast, if there is a discontinuity between the filter point and the detector, then the
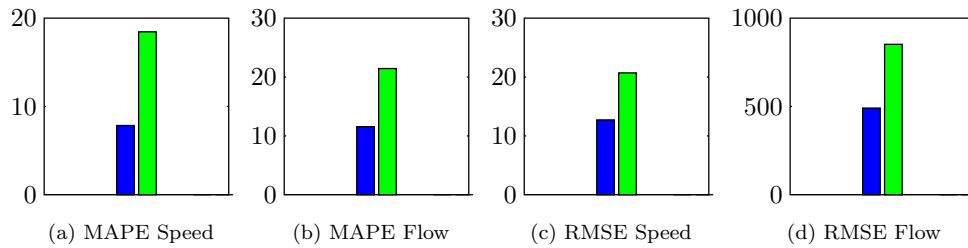
Figure 3: Error measures: ASM-svK (blue) against original ASM (green)

information from the detector is weighted less. The effect is that information from the same road section is valued higher than information from the neighboring road section, which is disrupted by the discontinuity.

3   Experimental Setup and Results

The ASM-svK is compared against the original ASM with simulated data from FOSIM. It microscopically simulates a highway stretch containing an on-ramp with a significant inflow, which constitutes a large discontinuity in the road geometry. This road stretch is monitored by dual-loop detectors every 500 m and every 60 sec. The dual-loop detector closest to the on-ramp is hidden. The speeds and flows estimated by the AMK-svK and the ASM at the position of this hidden detector are then compared with the actual observations of this detector. Figure 3 shows the Root Mean Squared Errors (RMSE) and the Mean Absolute Percentage Errors (MAPE) between the values estimated and values observed, for a simulation time of 2 h, averaged over 100 simulation runs.

As Figure 3 indicates, the ASM-svK estimates speeds and flows more accurately than the ASM, especially near road discontinuities. This method is thus suitable for travel time estimation. In addition, with this method the traffic state between an off- and an on-ramp is estimated, enabling the estimation of inflows, outflows and turn fractions of a highway.

References

[1] M. Treiber and D. Helbing. Reconstructing the Spatio-Temporal Traffic Dynamics from Stationary Detector Data. *Cooper@tive Tr@nsport@tion Dyn@mics*, 1(3):3.1–3.21, 2002.

[2] MJ Lighthill and GB Whitham. On Kinematic Waves. II. A Theory of Traffic Flow on Long Crowded Roads. *Proceedings of the Royal Society of London. Series A, Mathematical and Physical Sciences (1934-1990)*, 229(1178):317–345, 1955.

[3] P.I. Richards. Shock Waves on the Highway. *Operations research*, 4(1):42–51, 1956.

Controlling the level of robustness in timetabling and scheduling: a bicriteria approach

Anita Schöbel

Institut für Numerische und Angewandte Mathematik

Georg-August-Universität Göttingen, Göttingen, Germany

1 Introduction

Finding robust solutions of an optimization problem is an important issue in practice. In particular, in timetabling and scheduling one aims to have a solution which is still OK in case of small disturbances or delays. Various concepts on how to define the robustness of an algorithm or of a solution have been suggested, see Section 2.

However, there is always a trade-off between the best possible solution and a robust solution, called the price of robustness, see [3]. Most of the robustness approaches hence determine the level of robustness beforehand and seek for the solution with best objective value with at least this minimum level of robustness.

In this paper, we analyze this trade-off using a bicriteria approach. We treat an optimization problem as a bicriteria problem adding the robustness of its solution as a second goal to the original given objective function.

For defining the robustness, any of the established robustness concepts can be used. Formally, let an *uncertain optimization problem*

$$(\text{Opt}(\xi)) \quad \min\{f(x, \xi) : x \in F(\xi)\}, \ \ \xi \in \mathcal{U}$$

be given with its nominal objective f and its feasible set $F(\xi) \subseteq \mathcal{F}$. The dependency of the optimization problem on the input data is indicated by ξ. Roughly speaking, a solution is robust if it is still suitable for all scenarios $\xi \in \mathcal{U}$, where $\mathcal{U}$ is called the *uncertainty set*. Note that the definition of "suitable" depends on the robustness concept used: In strict robustness, a solution is "suitable" for ξ if $x \in F(\xi)$, in light robustness x only needs to satisfy some relaxed constraints and in recovery robustness "suitable" means that a recovery algorithm exists which is able to update the solution x to a feasible one. For all robustness definitions, one can define a function $R(x, \xi)$ evaluating the level of robustness that a solution x has. It is usually assumed that the robustness

of a solution is large if it is "suitable" for many scenarios, i.e. if the uncertainty set $\mathcal{U}$ is large. This level of robustness is in most cases defined beforehand, often implicitly through the definition of the uncertainty set. In our analysis we will extend the nominal problem by adding the robustness as a second objective, i.e. we consider a vector optimization problem of the type

$$\left\{ \begin{pmatrix} \min f(x,\xi) \\ \max R(x,\xi) \end{pmatrix} \;\; \text{s.t.} \;\; x \in F(\xi) \right\}. \tag{1}$$

We are mainly interested for which definitions of R Pareto solutions can be found without increasing the time-complexity of the original problem $(\text{Opt}(\xi))$. To this end, we will analyze the additional constraints to be added when using e.g. the ϵ-constraint method. Depending on their structure the problem may be solved similar to the original problem (Opt).

2 Robustness concepts

In robust optimization, the objective – in contrast to stochastic programming – is purely deterministic. In the concept of *strict robustness* ([1]), the solution has to be feasible for all likely scenarios. The solution gained by this approach can then be fixed since by construction it needs not be changed when disturbances occur. However, as the solution is fixed independently of the actual scenario, strictly robust optimization leads to solutions that are too conservative in many applications. Possible approaches to overcome this problem concern *adjustable robustness* ([2]) in which dependent variables may be adjusted if the scenario is known, *light robustness* ([5]) where the constraints are relaxed by adding slack to them, and *recoverable robustness* ([6, 4]). The latter concept starts from the practical point of view that a solution is robust if it can be recovered easily in case of a disturbance. This means the solution has no longer to be feasible for all possible scenarios, but a recovery phase is allowed in which a recovery algorithm is applied to turn an infeasible solution into a feasible one. To obtain a good solution, some limitations on the recovery phase have to be taken into account. For example, the recovery should be quick enough and the quality of the recovered solution should not be too bad.

3 Controlling the level of robustness in timetabling

Roughly speaking, the robustness of a timetable evaluates its sensitivity to unforeseen delays. It is clear that the concept of *strict* robustness makes not much sense: If a delay occurs it is usually not possible to keep all departure and arrival times as planned. The question is rather, how to recover the timetable to allow the passengers to reach their goals with smallest possible delay. This question is also known as *delay management problem* [8] and the special strategy used is important for defining the robustness level of a timetable.
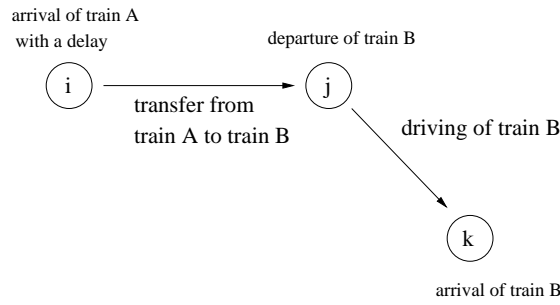
Figure 1: Train 1 arrives at station A with a delay. Should train 2 wait or depart on time?

We hence first specify how a timetable is updated in case a delay occurs.

Let i be an arrival event of train 1 and let $a = (i, j)$ be a transfer activity to train 2. Furthermore, let $\tilde{a} = (j, k)$ be the next driving activity of train 2, see Figure 1 for an illustration. Assume that train 1 arrives at i with a delay of y_i and let us denote the slack times of activities a and $\tilde{a}$ by s_a and $s_{\tilde{a}}$, respectively. The following three rules can be used to determine if train 2 should wait for train 1 or depart on time.

WTR1: Train 2 is not allowed to have a delay at its *next* station. Hence the maximal allowed waiting time at event j is given by the slack time $s_{\tilde{a}}$ of its next driving activity $\tilde{a} = (j, k)$. The transfer is maintained if and only if $y_i \leq s_a + s_{\tilde{a}}$.

WTR2: The maximal allowed waiting time at event j is n minutes where n is fixed beforehand. The transfer is maintained if and only if $y_i \leq s_a + n$.

WTR3: Train 2 is not allowed to have a delay of more than m (minutes) at its *next* station. Hence the maximal allowed waiting time at event j is given by m plus the slack time $s_{\tilde{a}}$ of its next driving activity. The transfer is maintained if and only if $y_i \leq s_a + s_{\tilde{a}} + m$.

An intuitive definition of robustness hence is the following. Let a fixed waiting time rule (according to WTR 1,2, or 3 above) be given as well as a set of source-delayed events $\mathcal{E}_{del} \subseteq \mathcal{E}$. A timetable (given by its slack values $s \in \mathbb{R}^{|\mathcal{A}|}$) has the robustness $R(s)$ if all its transfers are maintained whenever all source delays are smaller than or equal to R, i.e. the propagation of delays will not cause a transfer to fail.

Analyzing the bicriteria problem (1) w.r.t this definition we will show that Pareto solutions are in some sense extreme solutions:

Theorem 3.1 *Let $\tilde{a} \in \mathcal{A}^{drive}$. Let $prec(\tilde{a})$ be the set of its directly preceding activities. Let s be a Pareto solution. Then for all three waiting time rules s satisfies:*

$s_{\tilde{a}} = 0$ *or* $s_a = m_a$ *for some* $a \in prec(\tilde{a})$,

where m_a is a given upper bound on the slack time of activity a.

We will furthermore present a solution approach which shows that solutions with a given level of robustness can be calculated within the same (polynomial) time complexity as for the usual aperiodic timetabling problem. This is achieved by adding virtual activities which ensure that the propagation of delays is kept small.

4 Other timetabling and scheduling problems

We will show that robust project planning can be interpreted as a special case of robust aperiodic timetabling and hence can also be solved efficiently for any given level of robustness. We will furthermore derive special properties of the Pareto set for project scheduling based on the analysis of the critical path and the resulting buffer times for the non-critical activities. As an extension we will also model the periodic robust timetabling problem in a bicriteria setting.

References

[1] A. Ben-Tal and A. Nemirovski, Robust convex optimization, in *Mathematics of Operations Research* 23:4, pp. 769-805, 1998

[2] A. Ben-Tal and A. Goryashko and E. Guslitzer and A. Nemirovski, Adjustable robust solutions of uncertain linear programs, in *Math. Programming* 99, pp. 351-376, 2004

[3] D. Bertsimas and M. Sim, The Price of Robustness, in *Operations Research*, 52:1, pp. 35-53, 2004

[4] A.L. Erera and J.C. Morales and M. Savelsbergh, Robust Optimization for empty Repositioning Problems, in *Operations Research* 57:2, pp. 468-483, 2009

[5] M. Fischetti and M. Monaci, Light robustness, in *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*, LNCS, Vol. 5868, pp. 61-84, 2009

[6] C. Liebchen and M. Lübbecke and R. H. Möhring and S. Stiller, The concept of recoverable robustness, in *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*, LNCS, Vol. 5868, pp. 1-27, 2009

[7] A. Schöbel and A. Kratz, A bicriteria approach for robust timetabling, in *Robust and Online Large-Scale Optimization: Models and Techniques for Transportation Systems*, LNCS, Vol. 5868, pp. 119-144, 2009

[8] A. Schöbel, Capacity constraints in delay management, in *Public Transport* 1:2, pp. 135-154, 2009

Dynamic Capacity Control for Flexible Products: An Application to Resource Allocation in Transportation Logistics

Herbert Kopfer

Chair of Logistics

University of Bremen, Wilhelm-Herbst-Straße 5, 28359 Bremen, Germany

Email: kopfer@uni-bremen.de

Jörn Schönberger

Chair of Logistics

University of Bremen, Wilhelm-Herbst-Straße 5, 28359 Bremen, Germany

Email: jsb@uni-bremen.de

The deregulation and harmonization of the international trade, especially within Europe, have had significant impact on the business strategy and market positioning of road-based freight carrier companies. For them, the floor is prepared to act actively on the market by adjusting their products and services as a reaction to demand changes. However, systems and decision tools for supporting a profit-oriented allocation of capacity (container, full truck loads or less-than-truckload) are hardly available because a one-to-one transfer of those technologies from other service industries is not possible. Due to the quite complicated interdependencies between process (routes, shipments) and customer orders (coupling effects) the determination of lower bounds for prices for particularly demanded transport services is not possible today [1].

Capacity control is part of the toolbox used in operational revenue management [2]. It is used to support the decision about the acceptance or rejection of requests that requires the allocation of scarce resources or resources that are likely to become scare during the booking phase for resources. Capacity control originates from civil airline industry applications but today capacity control techniques are used in various service industry applications [3] as well as in production contexts [4]. The major goal that drives capacity control is to exploit the available resources at highest efficiency. Typically, capacity control is compromised by missing knowledge about the exact future demand for the controlled resources. Therefore, concatenated acceptance/rejection decisions must be made consecutively whenever additional knowledge about the demand (e.g. additional requests for capacity) appear (dynamic capacity control, [5])

Traditionally, road-based freight carriage has been considered as a pure service provider for other value creation stages in a value creation system. The ability of freight carriage to create value

was ignored. However, since 1 and a half decade, important freight carriage companies have formed on the market that earned money offering innovation service products that often comprise additional services accompanying the pure transport service. Today, these companies are in a position that enables them to select the most profitable requests from the spot market only. However, an important prerequisite for such a request acquisition strategy is the availability of longer term contracts that ensures the carrier sufficiently high average capacity utilization. If this precondition is fulfilled then the freight carrier company tries to allocate the residual capacities for the most profitable requests found in the spot-market.

In this contribution, we investigate the dynamic acceptance/rejection decision problem that a fleet disposition manager of a freight carrier has to solve. The decision support challenge in such a situation is two-fold. At first, a stream of incoming requests for capacity allocation on fixed service routes must be managed dynamically by the dispatcher. The dispatcher has to decide reactively about the initial allocation of resources for an arriving request, e.g. it is to decide if there is a vehicle that is able to fulfill the request profitably (external acceptance). Secondly, transport services in a network are often so called flexible products [6], [7] which means, that the carrier has several alternatives to fulfill a once accepted request. Internal re-assignments of requests to another vehicle become necessary in order to enable the dispatcher to utilize the available overall capacity at highest profitability. The application of capacity control is a rarely investigated subject in the context of road-based freight-carriage and the application to road-based freight transportation has not received significant interest so far. Nevertheless, there is empirical evidence that the consideration of the specific requirements of flexible products enables the realization of additional profits [8]. An explicit control of internal re-assignments in order to exploit the potentials of flexible products is not investigated.

We start with the proof that the major application preconditions for a successful capacity control to the previously outlined decision situation are fulfilled [9]. At first, the operational flexibility of the resource capacity is low, e.g. it is not possible to extend the fleet spontaneously at reasonable efforts. Secondly, we have to integrate an external factor that is the unknown customer demand. Thirdly, the demand is heterogeneous and finally, the offered services can be standardized.

To prepare the development of an automatic decision support tool for dynamic capacity control, we model the carrier's acceptance problem as on online optimization Problem. Therefore, we deploy a resource scheme introduced in [10] and extend this scheme from the one-vehicle-application to the multi-vehicle-application in which a flexible product refers to different paths a package can use to travel from its pickup to its delivery location through a given transport network. If we use this scheme then we can formulated the combined request acceptance as well as the request re-assignment problem as a linear program. To solve an instance of this model, we can apply standard solvers based on the simplex-scheme and from the optimal simplex-tableau, we get the shadow-prices for the maintained resources. Using these shadow prices, we calculate the bid-prices which are lower bounds of the revenues associated with the incoming requests.

We have developed a rolling-horizon planning system framework. In each cycle of the framework a new capacity control model is determined and solved. From the solution of the model we can derive the bid-prices of the resources as well as the required request re-assignment decisions. We use the resource-specific bid-prices to calculate least revenues for additionally arriving requests and the re-assignment information for re-allocating capacities of the resources for already accepted requests. If additional requests are accepted (because the associated revenues are larger than their bid-prices) then we allocate resources on one of the vehicles of the available fleet. Otherwise, we reject the request. In addition, we explicitly check if internal re-assignments of already accepted requests among vehicles are useful in order to increase the performance of the capacity control approach

The evaluation of the proposed decision support approach is reported. We perform extensive computation simulation experiments in which we deploy the online capacity control model. We propose parameterizable test cases and record several performance indicators during the simulation experiments.

References

[1] M.A. Figliozzi, H. Mahmassani, P. Jaillet, "Traveling Pricing in Dynamic Vehicle Routing Problems", *Transportation Science* 41, 302-318 (2007).

[2] R. Klein and C. Steinhardt, *Revenue Management*, Springer-Verlag, Berlin-Heidelberg (2008)

[3] A. Ingold, I. Yeoman, U. McMahon, *Yield Management: Strategies for the Service Industries,* Thomson Learning, London, 2nd Edition (2007).

[4] T. Spengler, S. Rehkopf, T. Volling "Revenue management in make-to-order manufacturing – an application to the iron and steel industry", *OR Spectrum* 29, 157-171 (2007).

[5] M. Koenig and J. Meissner, "Risk Management Policies for Dynamic Capacity Control", Working Paper (available at http://www.meiss.com), Lancaster University management School (2009).

[6] M. Müller-Bungart, *Revenue Management with Flexible Products,* Springer-Verlag, Berlin-Heidelberg (2007).

[7] G. Gallego and R. Phillips, "Revenue Management of Flexible Products", *Manufacturing & Service Operations Management* 6, 321-337 (2004).

[8] A. Petrick, C. Steinhardt, J. Gönsch and R. Klein, Using flexible products to cope with demand uncertainty in revenue management, *accepted for publication in OR Spectrum.*

[9] A. Kimms and R. Klein, "Revenue Management im Branchenvergleich", *Zeitschrift für Betriebswirtschaftslehre,* Ergänzungsheft 1 "Revenue Management", S. 1-30 (2005)

[10] J. Schönberger and H. Kopfer, "Kapazitätssteuerung in der Transportlogistik", accepted for publication in Proceedings of Multi-Konferenz Wirtschaftsinformatik 2010, Göttingen, Februar 2010

A heuristic based on clustering for the vehicle routing problem: a case study on spare parts distribution

Mehdi Sharifyazdi[*]

Rotterdam School of Management, Erasmus Reseach Institute of Management

Erasmus University, Rotterdam, The Netherlands

Email: sharifyazdi@ese.eur.nl

Matin Bagherpour

Department of Industrial Engineering

University of Science and Culture, Tehran, Iran

Introduction

In this paper, a case is considered where a distribution center (warehouse of an auto spare parts company) receives orders regularly. Warehouse management is interested in assigning available vehicles to picked orders in such a way that lead time remains lower than a threshold, and transportation cost per unit (money) of received orders is minimized. Since the company receives orders dynamically and arrival of new orders can provide it with the opportunity to improve existing decided distribution paths, the problem better be solved several times a day in a dynamic manner.

We will propose an event-oriented (dynamic) algorithm for this problem. That is, the algorithm is called whenever specific events happen in the system. These events include arrival of an order and end of picking an order. Regarding the fact that many orders are received on a daily basis, the algorithm must be called frequently. So, runtime of the algorithm, as well as quality of solutions are of great importance. Performance of proposed algorithm is evaluated in a real-world case in an automobile spare parts distribution company in Tehran.

The mentioned problem can be classified in the category of Vehicle Routin Problems (VRP), but has several differences with the classical VRP [1] in the assumptions. There are different types of limited-capacity vehicles, time limits for delivery, different customers which must be served by different types of vehicles. Also, the most important of all, the problem has to be solved repeatedly, several times (per each order's arrival, completion and meet of dispatching deadline). Each node can also be visited several times by several vehicles. Although this problem has something in common with PVRP (periodic VRP) [2] and CVRP (capacitated VRP) [3], VRPTW (VRP with time windows) [3], it has its own specific characteristics which will be explained in the following sections.

---

[*] Corresponding Author

Probelm Definition and Formulation

The main purpose to solve this problem is to determine the orders to be distributed together, the distribution route, and the vehicle to use. The case is when there are some orders which their routes and vehicles have been currently assigned. Now, one (or more) new order arrives at the distribution center. Then, the objective is to modify previous routes with assigning new orders to the previous primary routes or creating some new routes in such a way that cover newly received orders. This is a dynamic and continuous decision process. Based on the observations made in the real case, the following assumptions are made:

(1) Information about received orders is available real-time; (2) Each order has a maximum acceptable delivery time (based on the type of customer as well as its distance to the distribution center); (3) There is always inventory available to satisfy the received orders in the distribution center; (4) The number of vehicles is infinite (since the vehicles are provided by a relativley big number of logistic service providers), so there is no need to worry about vehicle shortage when deciding on distribution routes and vehicle assignments; (5) All the vehicles start/end their route from/to the distribution center; (6) Total travel time of a vehicle (outside Tehran) is equal to the summation of travel time between cities, delivery, and pick up time. In other owrds, travel times within cities are negligible; (7) Always (if possible) a bigger vehicle is economically preferable to a smaller one. It means that the ratio of transportation cost to the volume of transported load (if the vehicle is used with full capacity), in all of the routes, is cheaper for big vehicles in comparison with small ones. In our studied case, six types of vehicles were available. According to the transportation contracts, this assumption is validated; (8) It is always preferable to use two vehicles in order to provide service to each city (cluster) than each vehicle covers a portion of the demand of each city; (9) To guarantee the existence of feasible solution for the problem, it is assumed that there is no order bigger than the capacity of the biggest available vehicle (trailer). Sometimes, this assumption may be violated. However, those big orders can simply be partitioned into smaller orders in such a way that none of them is bigger than the capacity of a trailer (the biggest vahicle available). Each of these partitioned orders is considered as a single order in the distribution system; (10) Violation from delivery time is not allowed; (11) Orders are collected according to their arrival time.

In this problem, vehicle capacity constraints, vehicle weight constraints, delivery time constraints and customer assignment constraints are considered.

Having such information as, customers' addresses, volume and weight capacity of the vehicles, volume and weight of orders, deadline of orders, travel times/expenses between customers and promised lead times, an integer programming model is made to determine the assignment of orders to routes and vehicles as well as dispatching times. With a special parameter setting, this model can be reduced to classical VRP and hence it is NP-Complete.

The Algorithm

Since the algorithm must be executed several times a day, its time efficiency is vital. To generate proper routes at the least possible time, we used a pre-structured framework of paths. To use this framework, cities (customer locations) are classified into groups namely "clusters". Each cluster includes the cities that can normally be served with a common vehicle (in a single route). After

construction, the clusters will remain fixed during routine and operational decision makings like routing and assignment of customers to vehicles. However, if new cities are added to the distribution network, the clusters will be updated.

Hence, the algorithm has two phases: (1) Static phase: Clustering of the cities; (2) Dynamic phase: Routing through fixed clusters. Order processing and routing can be performed either in batches or one-by-one. In one-by-one processing, routing is performed per each new order arrival and the newly arrived order will be assigned to a vehicle. Ideally, all the previously constructed routes must be updated when a new order arrives. In batch processing, received orders are processed and routses are updated periodically (for instance, once an hour).

For clustering, firstly, we built up a minimum spanning tree over the graph consisting of all of the cities (where at least one customer is located) as vertices and direct roads as edges. In this tree, each node (city) has an ancestor. The ancestor of a city is the city through which the city is connected to Tehran. In fact, Tehran is the root of this tree. To construct the clusters, first, each of the main roads branched from Tehran are considered as a main cluster and the cities on these roads are assigned to these main clusters. Then, at the points that each of these roads is branched to some other roads, each of these branches is considered as a secondary cluster. This process is repeated for the secondary clusters. At the end, other cities which have not been assigned to any cluster, are assigned to the existing cluster with the minimum distance to them. Then, the whole tree is juxtaposed versus the pathes which experienced drivers suggested to reach each city from the root (Tehran). Based on this comparison, some modifications has been made to the clustering tree to make the pathes practically feasible.

The second phase of the algorithm (routing and vehicle assignment) will be executed any time one of the following events happens: (1) Arrival of an order, (2) End of picking of an order, (3) Being prepared to be sent (for an order) and (4) Reaching the deadline for dispatching an order. The algorithm for the event of order arrival is designed for batch processing. However, it can also be easily used for real-time processing regarding the fact that it is a special case of batch processing (batch size = 1). In this algorithm, at the first stage, any newly arrived order or any other order already in the system which still has not been assigned to a finalized route, will be assigned to a separate route containing only one city. The routes where the volume of their corresponding order fills a minimum portion of a trailer (here 85%) will be finalized and planned for dispatching. Other routes (orders) are considered as incomplete one-city routes and will be merged with other routes or the next orders which will arrive later. Incomplete routes are merged only when it is time to dispatch one of the orders in an incomplete route. Meanwhile, in order to build economic routes, it is preferred to group the orders such a way that the total volume of the orders assigned to a route be close to the volume of a trailer. In order to merge orders (incomplete routes) within preferably the largest vehicles possible, firstly, the routes related to the same city will be combined (in a descending order with respect to the volume). The total volume of orders grouped in a route must not exceed the capacity of a trailer. Every new route where the total volume of the orders is more than the capacity of the largest vehicle will be finalized and assigned to a trailer. If even after merging, the total volume of the orders from the same city assigned to the same route is less than 85% of volume of a trailer, then, if it is possible, they will be merged

with the incomplete routes in the same cluster as theirs. If the trailer is not still full, the incomplete route will be combined to other incomplete routes of the child secondary clusters at first and if needed, to those of the parent clusters. Eventually, if a trailer cannot be filled in this manner, a treshold time, which shows the latest possible dispatching time (the earliest latest dispatching time among the orders assigned to the route), will be calculated for the incomplete route. If by the treshold time, the total volume of the orders cannot fill a trailer, then the smallest vehicle which can carry them, will be used.

When an order is picked completely, the number of orders in the queue decreases by one (if there is a queue), the status of the picked order changes to "being packed", and the status of the first order in the queue changes to "being picked". When packing of an order finishes, status of that order changes to "ready for shipment". If the route assigned to this order is finalized, and all of the orders in the route are ready to be shipped, then the proper vehicle will be called to ship it.

If a one-city route is "complete", its corresponding vehicle will be dispatched at the earliest possible time, and it is the best case with respect to cost, because the vehicle visits only one city and also the most economic and favorable vehicle (trailer) is used. However, if a route is "incomplete" at its latest possible dispatching time, it should be merged with other routes in order to make it more economic, in such a way that the promised delivery deadline is not violated. The reason that merging of the incomplete routes is postponed until the latest possible time, is to take advantage of the future orders to construct better routes.

Any time the algorithm is executed, several routes (preferably complete) are constructed, from which only one is used. It is the route with an order whose dispatching deadline is reached. The reason that not merely a single route is constructed for the order which should be shipped, and instead, other orders (which are not due to be sent at that time), are also assigned to routes (these routes will be split in future) is that if the algorithm only focuses on one order to build up a proper route for it, then it is likely that the opportunity to construct proper routes for other orders may be lost. It means that the route may be good for one order, while totally, the whole set of routes constructed for all of the orders is not satisfactory. In this situation, the algorithm would be a greedy algorithm which is normally very fast, but the solutions are not very good.

The Results

The algorithm has been coded and implemented in connection with the current integrated information and planning system of ISACO company. However, some further modifications and exceptions were needed to make it applicable. In a 6-month period between February and July 2009, the transportation cost per unit order has been decreased by 11.8% in comparison with the same period in 2008. Also, the percentage of the delayed orders, has been reduced from 2.6% to 0.06%.

References

[1] Dantzig, G.B., Fulkerson, D.R., Minimizing the number of tankers to meet a fixed schedule, *Naval Research Logistics*, q. 1, pp. 217-222, 1954.
[2] Pasaraftis, H.N., "Dynamic Vehicle Routeing Problems", in *Vehicle Routing: Methods and Studies*, B.L. Golden and A.A. Assad (eds), 223-248, North-Holland, Amsterdam, 1988.

[3] Toth, P., Vigo, D., The Vehicle Routing Problem, SIAM, 2002.

A large neighbourhood search heuristic for a periodic supply vessel planning problem arising in offshore oil and gas operations

Aliaksandr Shyshou

Molde University College

Kjetil Fagerholt

MARINTEK and The Norwegian University of Science and Technology

Irina Gribkovskaia

Molde University College, Postbox 2110, 6402, Molde, Norway

Email: Irina.Gribkovskaia@himolde.no

Gilbert Laporte

Canada Research Chair in Distribution Management

HEC Montréal

1 Problem description and literature review

Oil and gas operators use supply vessels to service offshore installations from an onshore base. Supply vessel expenses constitute a significant cost component in the area of upstream logistics as their daily charter rates may be as high as 80,000 US dollars. Our study is based on a real-world problem faced by Statoil, the largest Norwegian offshore oil and gas operator. However, the model and algorithms we provide are of wide applicability. We refer to the problem as the periodic supply vessel planning problem (PSVPP).

The PSVPP consists of simultaneously determining a repetitive weekly sailing plan, made up of scheduled voyages, and the fleet configuration required to perform these voyages. Each voyage is defined by a start day and a sequence of installations to visit. The objective is to minimize the sum of the vessel charter costs and of the voyage sailing and service costs.

A number of practical constraints have to be respected. Voyage duration has to lie within certain limits, i.e. there are minimum and maximum durations which are measured in days. A practical restriction is that a voyage should last two or three days. This implies that a vessel

can perform two to three voyages per week. There are also restrictions on the minimum and maximum number of installation visits per voyage. In addition, vessel deck capacity should not be exceeded. In reality vessels have two types of capacity: one for deck cargo and one for bulk products. However, in the vast majority of the cases, deck capacity is the constraining factor since bulk capacity is much larger compared to the demand.

Each installation has a weekly demand that must be fulfilled. Additionally, the installations need to be visited a certain number of times per week. It is assumed that weekly demands are uniformly distributed among the visits. The departures should also be fairly evenly spread throughout the week. It should be noted that the capacity of the onshore base is limited, i.e. the number of vessel departures on a given day is limited. Additionally, all vessels depart from the base at 16:00 out of practical considerations as this is the end of the working day when all loading operations are finished. Some installations are closed at night, while others may be visited at any time. Another practical restriction is that there should be no vessel departures on Sundays.

In Figure 1 we depict a weekly sailing plan involving three vessels (Star, Symphony, Foresight) and five offshore installations (BID, BRA, DSD, HDA, GRA). The number in each cell of the upper row corresponds to the end of an eight-hour slot. In this example each vessel performs two voyages during the week. The voyages have a bold outline in the figure. Each voyage is preceded by an eight-hour period at the onshore base, when loading and unloading operations take place. In the figure these periods are shaded with diagonal stripes. It can be seen that vessel Foresight starts its second voyage on Saturday at 16:00 (136 hours after the beginning of the week) and finishes it on Monday of the following week.

	Monday			Tuesday			Wednesday			Thursday			Friday			Saturday			Sunday		
	8	16	24	32	40	48	56	64	72	80	88	96	104	112	120	128	136	144	152	160	168
Star		░GRA			BID	DSD		BRA			░GRA			BID		BRA					
Symphony					░GRA	HDA		BID	DSD BRA					░	BID	DSD BRA					
Foresight	DSD BRA						░GRA			BID		DSD				░	GRA	HDA			BID

The PSVPP can be classified as a periodic vessel routing and fleet sizing problem with time windows and multiple use of vessels. A number of problem aspects stand out from the classification, namely periodic routing, fleet sizing and mix, and multiple use of vessels during the time horizon. Relaxed versions of the PSVPP have been studied by [1] and [2].

Two exact methods for the PSVPP are proposed in [3]. The first uses an arc flow model, while the second is based on the pregeneration of all cheapest feasible voyages which are then used as an input to a set covering model with numerous side constraints. The authors demonstrate that the second method, referred to as the voyage-based formulation, is computationally superior to the first one.

2 Large neighborhood search heuristic and computational results summary

We have developed a large neighbourhood search (LNS) heuristic for the PSVPP. The algorithm is applied for a number of restarts. At each restart we randomly generate initial feasible solution and then perform a number of LNS iterations. An LNS iteration consists of three parts: 1) Removal loop (randomly remove several visits from several voyages and put them in a bank of uninserted visits S). 2) Insertion loop (reinsert visits from S back into voyages using regret insertion heuristics). 3) If $S = \emptyset$ apply local improvement procedures which mostly involve reassignments of voyages to other vessels and visit relocations.

We evaluated our heuristic on the instances from [3], which are based on actual data provided by Statoil. There are 22 instances in total with the smallest involving three installations and the largest involving 14 installations. Computational results are summarized in Table 1. Column "**Gap (%)**" gives the percentage gap between the value of the best found solution and either the value of the optimal solution ($*$) or that of the smallest valid lower bound ($\dagger$). The optimal solution values and the lower bound values are those of [3]. Column "**Seconds**" gives the CPU time in seconds. The columns "**Vessels**" and "**Voyages**" report the number of vessels used and voyages performed respectively.

Heuristic solutions are generally very close to optimal ones, with an average gap of 0.05% for the instances with up to twelve installations. Moreover, LNS computation times are quite stable. This heuristic is generally slower on the easier instances and faster on the more difficult ones. As a result, LNS can be applied when the exact method ceases to be practical.

References

[1] K. Fagerholt and H. Lindstad, "Optimal policies for maintaining a supply service in the Norwegian Sea", *Omega* 28, 269–275 (2000).

[2] I. Gribkovskaia, G. Laporte and A. Shlopak, "A tabu search heuristic for a routing problem arising in servicing of offshore oil and gas platforms", *Journal of the Operational Research Society* 59, 1449–1459 (2008).

[3] E. E. Halvorsen-Weare, K. Fagerholt, L.M. Nonås and B.E. Asbjørnslett, "Fleet size and mix and periodic routing of offshore supply vessels", Submitted for publication (2009).

Instance	Gap (%)		Seconds	Vessels	Voyages
3-5-16-0	0.00	(∗)	5.2	2	6
4-5-21-0	0.00	(∗)	13.9	3	6
5-5-23-0	0.00	(∗)	21.9	3	6
5-5-23-1	0.00	(∗)	23.1	3	6
6-5-25-0	0.00	(∗)	30.0	3	6
6-5-25-2	0.03	(∗)	32.4	3	6
7-5-30-0	0.00	(∗)	55.6	3	6
7-5-30-2	0.06	(∗)	65.3	3	6
8-5-36-0	0.00	(∗)	109.2	3	6
8-5-36-2	0.02	(∗)	114.3	3	6
9-5-42-0	0.01	(∗)	695.2	3	6
9-5-42-2	0.05	(∗)	663.9	3	6
10-5-43-0	0.03	(∗)	657.5	3	7
10-5-43-3	0.01	(∗)	671.0	3	7
11-5-47-0	0.18	(∗)	1 696.4	3	8
11-5-47-3	0.17	(∗)	1 927.4	3	8
12-5-51-0	0.13	(∗)	1 994.7	3	8
12-5-51-3	0.19	(∗)	2 253.5	3	8
13-5-55-0	13.91	(†)	2 248.4	4	9
13-5-55-3	1.09	(†)	2 384.5	4	9
14-5-59-0	1.02	(†)	3 376.4	4	9
14-5-59-3	0.97	(†)	2 857.7	4	9
Average	**0.81**		**995.3**	**3.14**	**7.00**

Table 1: Computational results for the large neighborhood search heuristic

A methodology for locating link count sensors taking into account the reliability of prior o-d matrix estimates

Fulvio Simonelli

Dipartimento di Ingegneria dei Trasporti "L. Tocchetti"

Università di Napoli "Federico II" – Italy

Vittorio Marzano

Dipartimento di Ingegneria dei Trasporti "L. Tocchetti"

Università di Napoli "Federico II" – Italy

corresponding author Email: vmarzano@unina.it

Andrea Papola

Dipartimento di Ingegneria dei Trasporti "L. Tocchetti"

Università di Napoli "Federico II" – Italy

Roberta Vitillo

Dipartimento di Ingegneria dei Trasporti "L. Tocchetti"

Università di Napoli "Federico II" – Italy

The problem of selecting the optimal locations of link count sections for o-d matrix estimation has received much interest in transport engineering. Various methods have been proposed in the literature to date, following different approaches and objectives, either finding the minimal set for inferring all link flows in a network from the subset of counted flows, such as in [1] and [2], or updating/correcting prior deterministic o-d matrix estimates, e.g. through maximum flow or maximum coverage methods [3], or topological heuristics, e.g. the screen-line method [4]. Notably, none of the mentioned methods takes into account explicitly the degree of reliability of the prior o-d matrix estimate. Rather, such reliability is normally expressed either in terms of the reliability of the estimator (e.g. the MPRE measure [5]) or by means of other ex-post measures (e.g. the total demand scale [6]).

This paper proposes a different approach, based on the availability of prior information about a space of feasible o-d matrices, i.e. the joint probability distribution of o-d demand rather than only a prior deterministic estimate. Notably, the distribution and its domain can be easily built on the basis of observable data, for instance upper and lower bounds of the domain may be linked to residential and

labour densities of each zone, and the functional form of the distribution may be related to the methodologies adopted for obtaining the prior o-d estimate (source and/or model). In this framework, equations represented by link counts allow for a reduction of the space of the feasible o-d matrices, in terms both of dimension and dispersion, the latter measured accordingly with the metric related to the chosen reliability measure. The research aims at showing that this approach is expected to give new insight on the link count location problem. By way of an example, in this abstract a toy network application is presented, in order to show how an heterogeneous level of knowledge across o-d pairs may lead to the choice of counting sections different from those resulting from the commonly adopted procedures. This is particularly true in presence of heterogeneous prior information, which has been addressed to date only in the choice of the estimator (e.g. [7]-[9]). In the final version of the paper, applications to real networks will be performed as well.

To show the rationale of the approach, let us assume a static framework, i.e. linear relationship between the demand vector $\mathbf{d}$ and the link flows vector $\mathbf{f}$ through the assignment matrix $\mathbf{M}$. Let also $\mathbf{y}$ be the subset of available link counts and $\mathbf{M^*}$ the related submatrix of $\mathbf{M}$; in the following, both $\mathbf{M}$ and $\mathbf{y}$ will be assumed error-free. As stated in the introduction, a joint distribution $f_d(\mathbf{d})$ with feasibility domain S_d can be hypothesized for $\mathbf{d}$, based on prior knowledge. Consequently, by means of the usual equations $\mathbf{M^*} \cdot \mathbf{d} = \mathbf{y}$ expressing link counts, a joint distribution $f_y(\mathbf{y})$ with feasibility domain S_y can be stated as well, based on $f_d(\mathbf{d})$ and S_d. The equations coming from link counts with their information lead to a new feasibility domain S^y_d and to a new joint distribution for the true o-d demand $f_d(\mathbf{d}|\mathbf{y})$ conditioned on $\mathbf{y}$.

Therefore, a measure of the reliability of the correction procedure may be defined on the basis of S^y_d and $f_d(\mathbf{d}|\mathbf{y})$. Apart from the volume of the set S^y_d (which is not practical due to the difficulties in calculating volumes of polytopes), an example may be a global dispersion measure based on distributions $f_d(\mathbf{d}|\mathbf{y})$ and $f_y(\mathbf{y})$. In addition, an estimator can be taken into account, e.g. the barycentre of the distribution $f_d(\mathbf{d}|\mathbf{y})$, or the projection of the prior distribution $f_d(\mathbf{d})$ on S^y_d. In general, given an estimator $\mathbf{d^*}$, another measure of reliability may be the mean of the distance between the true o-d matrix and the chosen estimator, that is:

$$MErr = \int_{S_y} \left[\int_{S_{d(y)}} Err(\boldsymbol{d}, \boldsymbol{d^*}) f_d(\boldsymbol{d}|\boldsymbol{y}) \, \delta \boldsymbol{d} \right] f_y(\boldsymbol{y}) \, \delta \boldsymbol{y} \tag{1}$$

where Err($\mathbf{d}$,$\mathbf{d^*}$) is the error distance function between the true demand and the chosen estimator. Accordingly, equation (1) can be introduced as objective function in the traditional formulation of the problem of optimal link count sections locations:

$$\boldsymbol{z} = argmin \{MeanError(\boldsymbol{z})\} \qquad \text{s.t.} \sum_{i=1}^{i=Nl} \boldsymbol{z_i} \leq Z_{max}$$

where Nl is the number of links in the network, $z_i=1$ if link i is a count section and Z_{max} is the maximum number of sensors (budget constraint).

In order to show how the preceding formulation may lead to a choice of count locations significantly different from the outcomes of the common methods, let us consider the following toy network (left side of the figure below) with 3 links, two o-d pairs (A-C and B-C), no route choice and naïve assignment matrix $\mathbf{M} \equiv \{(1,0);(0,1);(1,1)\}$. Let also be $Z_{max}=1$.
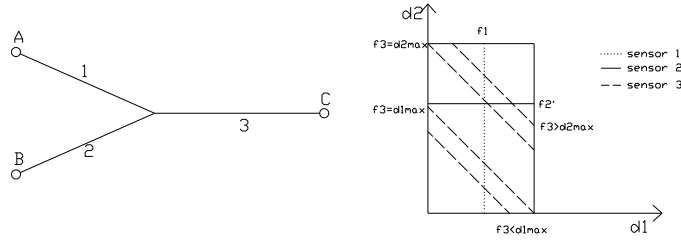
In a first step, the joint distribution of the true demand $\mathbf{d}=(d_{AC}, d_{BC})=(d_1, d_2)=(f_1, f_2)$ may be hypothesized to be uniform in the ranges $[0,d_{1max}]$ e $[0,d_{2max}]$ respectively, with independent marginals. This means having only prior knowledge about S_d but not specific knowledge about $f_d(\mathbf{d})$. Clearly, the maximum coverage and the maximum flow interception methods lead to the choice of link 3 as sensor location. However, the right side of the figure above depicts the feasibility set S^y_d of demand flows for all the three possible sensor locations. Notably, S^y_d is always a straight line whatever link is chosen for sensor location; however, due the specific nature of the example, S^y_d depends on the (unknown) value of the counted flow only if count section is located on link 3.

From the hypotheses above, $f_y(\mathbf{y})$ in the case of $\mathbf{y} \equiv \{f_3\}$ becomes a trapezoidal distribution and the equation (1) can be applied for the calculation of the mean error. For this aim, the estimator is assumed to be the barycentre of S^y_d with $\mathbf{y} \equiv \{f_3\}$ and the metric is the usual metric in $\mathbb{R}^2$. The results are (under the assumption $d_{2max} \geq d_{1max}$):

$MErr_1 = 0.25 \cdot d_{2max}$; $MErr_2 = 0.25 \cdot d_{1max}$; $MErr_3 = 0.353 \cdot d_{1max} \cdot [1 - d1_{max}/(3 \cdot d_{2max})]$

where $MErr_i$ is the solution of equation (1) when link i is the count section. As a consequence, $MErr_3$ depends on the heterogeneity of the prior estimate and, for decreasing value of the ratio d_{1max}/d_{2max} the optimal link count section changes from link 3 to link 2 (see following table).

d_{1max}/d_{2max}	$MErr_1$	$MErr_2$	$MErr_3$	Best Location
1	$0.250 \cdot d_{1max}$	$0.250 \cdot d_{1max}$	**$0.237 \cdot d_{1max}$**	3
0.9	$0.278 \cdot d_{1max}$	$0.250 \cdot d_{1max}$	**$0.249 \cdot d_{1max}$**	3
0.8	$0.321 \cdot d_{1max}$	**$0.250 \cdot d_{1max}$**	$0.260 \cdot d_{1max}$	2
0.1	$2.500 \cdot d_{1max}$	**$0.250 \cdot d_{1max}$**	$0.342 \cdot d_{1max}$	2

Alternatively, an independent normal distribution may be defined for both $d_1 = N(\mu_1, \sigma_1^2)$ and $d_2 = N(\mu_2, \sigma_2^2)$, leading to a normal bivariate vector $\mathbf{d}$ with diagonal covariance matrix. Therefore, if link 1 or 2 are chosen as count section, one component of $\mathbf{d}$ is fixed and the inaccuracy of the estimate can be measured through the dispersion (i.e. variance) of the other unknown component. If link 3 is chosen instead, a change of basis can be performed adopting an horizontal axis parallel to the equation (see

figure above), through the rotation matrix **B**. As a consequence, **d** will be expressed in the current reference system as $\{d/\!/,d\perp\}=$ **B·d** with mean **B·μ** ($\mu\equiv\{\mu_1, \mu_2\}$) being and dispersion matrix (no longer diagonal) $\Sigma'=$**B^{t}ΣB**. Therefore, fixing f_3 equals fixing the component $d\perp$ and the inaccuracy can be related to the dispersion of the component $d/\!/$ whose conditioned distribution $f(d/\!/\mid d\perp=\frac{\sqrt{2}}{2}f_3)$ is still normal distributed with mean and standard deviation respectively given by:

$$\mu_{d/\!/}|\tfrac{\sqrt{2}}{2}f_3 = \mu_{d/\!/} + \tfrac{cov(d/\!/,d\perp)}{var(d\perp)}(\tfrac{\sqrt{2}}{2}f_3 - \mu_{d\perp}) \quad \text{and} \quad \sigma_{d/\!/}|\tfrac{\sqrt{2}}{2}f_3 = \sigma_{d/\!/} - \tfrac{cov(d/\!/,d\perp)^2}{var(d\perp)}$$

With the same calculation of above it follows:

$$MErr_1=\sigma_2^2 \quad ; \quad MErr_2=\sigma_1^2 \quad ; \quad MErr_3=2\sigma_2^2\sigma_1^2 / (\sigma_2^2 + \sigma_1^2)$$

and the corresponding best location is reported in the following table for different values of the ratio σ_2^2/σ_1^2 (σ_1^2 equal to 1 for sake of simplicity).

σ_2^2/σ_1^2	Err_1	Err_2	Err_3	Best Location
1	1	1	1	1-2-3
1.2	1.2	1	1.09	2
1.4	1.4	1	1.17	2
1.6	1.6	1	1.23	2
1.8	1.8	1	1.31	2

References

[1] Hu, S. H., Peeta, S., and Chu, C. Identification of vehicle sensor locations for link-based network traffic applications. Transportation research Part B 2009.

[2] Viti, F., W. Verbeke, and C.M.J. Tampère. Sensor Locations for Reliable Travel Time Prediction and Dynamic Management of Traffic Networks, Tansportation Research Record vol 2049/2008

[3] H. Yang, J. Zhou, Optimal traffic counting locations for origin-destination matrix estimation, Transportation research Part B vol. 32 n.2 1998

[4] Cascetta E. and Postorino M.N. (2001). Fixed point models for the estimation of O-D matrices using traffic counts on congested networks. Transportation Science 35(2).

[5] H. Yang, Y. Iida, T. Sasaki, An analysis of the reliability of origin-destination matrices from link traffic counts on congested networks, Transportation research Part B vol. 25 1991

[6] M. Bierlaire, The total demand scale: a new measure of quality for static and dynamic origin destination trip tables Transportation research Part B vol. 25 2002

[7] . Cascetta, E. Estimation of trip matrices from traffic counts and survey data: a generalized least squares estimator. *Transportation Research, Part B*, 18:289–299 1984.

[8] Cascetta, E. and Nguyen, S. A unified framework for estimating or updating origin/ destination matrices from traffic counts. *Transportation Research, Part B*, 22:437–455 1988

[9] Doblas, J. and Benitez, F. G. An approach to estimating and updating origin-destination matrices based upon traffic counts preserving the prior structure of a survey matrix *Transportation Research, Part B* , 39 2005

Workforce management in periodic delivery operations

Karen Smilowitz

Industrial Engineering and Management Sciences

Northwestern University, Evanston, IL, USA 60208

e.mail: ksmilowitz@northwestern.edu

Maciek Nowak

Information Systems and Operations Management Department

Loyola University, Chicago, IL, USA 60611

e.mail: mnowak4@luc.edu

Tingting Jiang

Industrial Engineering and Management Sciences

Northwestern University, Evanston, IL, USA 60208

e.mail: tingting-jiang@northwestern.edu

1   Extended Abstract

This paper explores the interplay between workforce management and routing decisions in the Period Vehicle Routing Problem (PVRP). The PVRP, introduced in [2] and [9], is an extension of the classic vehicle routing problem (VRP), with vehicle routes constructed to service customers according to preset visit frequencies over an established period of time; see [6] for a review of the PVRP. The objective of the PVRP is to create a set of tours for each vehicle on each day in the period to minimize the routing costs, while satisfying operational constraints such as vehicle capacity and customer visit frequency.

When routes are constructed over multiple days, issues of workforce management arise. In [5], the authors consider workforce management in terms of the operational complexity of a solution, which is defined as the difficulty of implementing a PVRP solution for both service providers and customers. The authors develop several metrics for quantifying operational complexity, including crewsize, which measures the number of different drivers visiting a customer over the period. The

1

metrics are calculated *a posteriori* to evaluate the complexity of periodic routing solutions obtained with objectives of minimizing travel time and maximizing visit frequency. In [8], the authors introduce the Consistent Vehicle Routing Problem (CVRP). Workforce management is considered in the modeling and solution phases of a periodic routing problem by adding a constraint for customer service. A customer must be visited by the same driver throughout the service period.

Importantly, the work in [8], as well as related work in [13], is motivated by the express package delivery industry, where trucks visit sets of customers over the course of a week [12]. Workforce management is critical in this industry. For example, customers may prefer to be serviced by the same driver over the course of the week. Further, the company may wish to send the same driver to a customer repeatedly in order to take advantage of the familiarity the driver establishes with the customer or with the geographic region, see [13]. This familiarity is taken into consideration by [11] when constructing daily routes for the courier delivery problem with uncertainty; a driver is able to visit more customers on a route given route familiarity. Considering workforce management when routing vehicles has an effect on the solution to the PVRP, creating a problem in which the optimal solution is not necessarily found through the minimization of travel cost.

In this paper, we summarize the results found in [10]. This work incorporates workforce management into the modeling of the PVRP by adding several metrics to the objective function:

- Driver consistency (DC): the objective term includes a function that increases cost for every additional driver that visits a customer;

- Customer familiarity (CF): the objective term includes a cost function that reduces the cost per customer visit for a driver as the frequency of visits to that customer increases for that driver;

- Region familiarity (RF): the objective term includes a cost function that reduces the cost per visit to a geographic region for a driver as the frequency of visits to that region increases for that driver.

Several multi-objective models are developed and a general comparison of the various models is conducted, identifying the operational characteristics of each model. For example, the DC and CF models are similar in their goals; however, CF places a higher value on an increased frequency of visits, rather than simply reducing the number of drivers visiting a customer. We also analyze the differences between models focused on customer familiarity and the model focused on region familiarity. As with most VRP literature, much of the PVRP related literature has focused on heuristic solution methods (e.g., [1], [3], and [4]). We present a Tabu Search heuristic approach to solving this problem, adapted from [7], which is modified to account for operational complexity. Several parameters associated with workforce management objectives are evaluated, including the frequency of customer requests and the balance between travel costs and workforce management

metrics. Further, we evaluate multiple models of driver learning behavior.

We find that solving the traditional PVRP to minimize travel cost can lead to solutions that are less desirable from the perspective of workforce management. This is true even when a post-processing phase is introduced to improve the workforce metrics of the cost-minimizing solutions. However, by adding workforce management metrics to the objective function when initially designing routes, an appropriate balance can be obtained between travel cost and workforce management goals. It is shown that with the proper parameters in place, workforce management principles may be successfully applied without sacrificing other operational objectives.

References

[1] E. Angelelli and M. G. Speranza. The periodic vehicle routing problem with intermediate facilities. *European Journal of Operational Research*, 137:233–247, 2002.

[2] E. Beltrami and L. Bodin. Networks and vehicle routing for municipal waste collection. *Networks*, 4:65–94, 1974.

[3] I.-M. Chao, B. Golden, and E. Wasil. An improved heuristic for the period vehicle routing problem. *Networks*, 26:25–44, 1995.

[4] J.-F. Cordeau, M. Gendreau, and G. Laporte. A tabu search heuristic for periodic and multi-depot vehicle routing problems. *Networks*, 30:105–119, 1997.

[5] P. Francis, K. Smilowitz, and M. Tzur. Flexibility and complexity in periodic distribution problems. *Naval Research Logistics*, 54:136 – 150, 2007.

[6] P. Francis, K. Smilowitz, and M. Tzur. *The Vehicle Routing Problem: Latest Advances and New Challenges*, chapter The period vehicle routing problem and its extensions, pages 73–102. Springer, 2008.

[7] M. Gendreau, A. Hertz, and G. Laporte. A tabu search heuristic for the vehicle routing problem. *Management Science*, 40:1276–1290, 1994.

[8] C. Groer, B. Golden, and E. Wasil. The consistent vehicle routing problem. Accepted in *Manufacturing and Service Operations Management*.

[9] R. Russell and W. Igo. An assignment routing problem. *Networks*, 9:1–17, 1979.

[10] K. Smilowitz, M. Nowak, and T. Jiang. Workforce management in periodic delivery operations. IEMS Working paper # 09-04, November 2009.

[11] I. Sungur, F. Ordonez, M. M. Dessouky, Y. Ren, and H. Zhong. A model and algorithm for the courier delivery problem with uncertainty. Accepted in *Transportation Science*.

[12] UPS Corp. Driving success: Why the UPS model for managing 112,450 drivers is a competitive advantage. http://pressroom.ups.com/mediakits/popups/factsheet/0,1889,1201,00.html, December 2008.

[13] H. Zhong, R. W. Hall, and M. Dessouky. Territory planning and vehicle dispatching with driver learning. *Transportation Science*, 41:74–89, 2007.

An optimizing heuristic for managing traffic flow at choke points in river transportation systems

L. Douglas Smith (Corresponding Author)
Robert M. Nauss
College of Business Administration
University of Missouri-St. Louis
ldsmith@umsl.edu

Jian Li
Dirk Christian Mattfeld
Carl-Friedrich-Gauss Faculty
Universität Braunschweig
d.mattfeld@tu-bs.de

1 Introduction

River transportation systems often contain series of choke points that require traffic regulation to help relieve congestion. The choke points may be caused by narrow navigation channels, sections of a river with swift currents, or locks that connect successive river pools with different elevations. In extreme cases, only a single powered vessel can navigate a channel or fit in a lock in the upstream or downstream direction. Vessels travelling upstream move with slower velocity at a given power setting, but may be easier to maneuver under power against the current. The time for a vessel to pass a choke point and clear it for the next vessel in the opposite (or same) direction varies according to the characteristics of the vessel itself, the direction of travel, the river conditions at the time of transit, and the specific sequence of operations. The mix and intensity of river traffic usually varies by season of the year, day of the week and time of day. **River navigation systems with choke points can be represented as a series of unique interdependent bi-directional servers with time-varying traffic levels and operational characteristics, stochastically determined itineraries, and multiple queues with restricted queueing disciplines to impose maneuvering constraints.** We have employed such a structure to address seasonal bottlenecks in a congested section of the Upper Mississippi River (UMR) navigation system. In the UMR, the expected locking time for a commercial vessel can vary from 15 to 110 minutes, depending on the number of barges being pushed by the vessel, their physical configuration in the locking process, and the direction of travel (upstream or downstream). The delay before a lockage operation can begin depends on the vessel's direction of travel and also on whether successive lockages involve vessels travelling in the same direction (with a recycling or "turnback" of the lock) or in opposite directions (with an exchange of vessels in the chamber as currently configured after the departing vessel clears the area).

In this paper, we discuss the development and application of a scheduling heuristic that minimizes total waiting times of vessels at a lock while respecting a restrictive tandem queueing discipline and employing a priority shifting mechanism that prevents serious inequities (relative to a FIFO solution) in the pursuit of operational efficiency. To test the heuristic, we compare its solutions for sets of randomly generated test problems against solutions from a nonlinear integer programming model for the same problems. After demonstrating the efficacy of the heuristic in a deterministic

context, we embedded it (as a C++ routine) into an Arena simulation model of the UMR waterway to show the potential benefits of employing an optimizing procedure for regulating the lockage operations of commercial barge traffic. Similar heuristics may be employed to minimize waiting times in other situations (such as repair shops or freight yards) where tandem queueing structures are required to position the next entities for processing, and where processing delays (setup times) depend on whether sequential entities are chosen from the same queue or different queues.

2. Optimal scheduling of lockage operations for commercial vessels

Optimization of operations at locks in river transportation systems can occur with consideration of several competing objectives. The Panama Canal, authority, for example, recognizes that schedules might consider fresh water usage in lockage operations, pilot availability, priority bookings, liability for refund of transit tolls if priority passage is not accomplished according to schedule, total throughput of vessels, and total revenue from tolls. Nauss [1] created an IP model to demonstrate the possibility of considering a variety of factors (urgency of cargo shipments, crew's experience, vessel equipment, etc.) when scheduling operations at locks in the UMR. Government and barge-industry representatives, however, reacted coolly to scheduling criteria beyond the waiting times of individual vessels. They saw first-come-first served (FCFS or FIFO) as the most equitable schedule but recognized the need to use other scheduling mechanisms to promote efficient use of the resources, especially when there were backlogs of vessels waiting at a lock. Accordingly, our further research on alternative scheduling regimes focused on the tradeoffs between equity and efficiency as reflected by the average waiting times for vessels. Smith et al. [2] developed an Arena discrete-event simulation model of the UMR and used it to study system performance under a variety of scheduling rules and infrastructure changes.

The scheduling of lockage operations must occur with consideration of the limited maneuvering space in the immediate vicinity of the lock (upstream and downstream). We deal with this problem by employing a tandem queueing structure for traffic in both directions. As illustrated in Figure 1, the first position at the head of the queue is designated as the "mooring buoy" from which the next vessel to lock from that direction must be chosen. When a vessel at the mooring buoy is cleared to enter the lock, any vessel queued behind may be chosen to occupy the mooring buoy and thus be designated as the next vessel to lock from that direction. Generally, the waiting times are minimized if vessels are locked in order of their processing times. The maneuvering limitations (tandem queueing structure) and equity considerations interfere with this solution. The vessel that could be locked most quickly in a given direction may, for example, may have arrived after a vessel with very long processing time had been positioned at the mooring buoy.
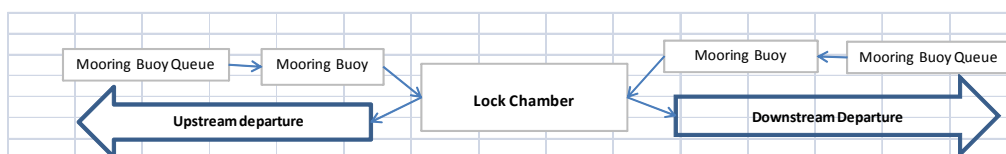
Figure 1. Schematic model of a lock and its queues for commercial vessels.

Taking this into consideration, a new IP model (Smith and Nauss, [3]) was created that would minimize total (or average) waiting times for all vessels, while (1) limiting the delay experienced by an individual vessel to a specified maximum delay beyond the delay it would experience in a FIFO solution and (2) respecting the restricted queueing discipline. With the new IP model, we were able to demonstrate how greater efficiencies could be achieved when there is more diversity in the traffic mix and when longer delays relative to the FIFO solution are allowed for individual vessels. To assess the potential effects of using the optimizing procedure in practice, however, we needed to simulate its usage over many years while re-solving the lock clearing problem at each lock after each departure. The solutions times for the IP procedure were not excessive (approx. 20 seconds for 20 queued vessels) for operational purposes, but a more efficient solution procedure was needed for deployment in the simulation model. A more efficient procedure would also be required in other applications where the processing time for queued entities is much shorter.

The scheduling heuristic addresses the problem in two phases. In the first phase, it strives to create a processing sequence that is close to the "fastest processing time sequence" considering both setup and processing times, while forcing the first vessel to be locked in a given direction to be the vessel currently at the respective mooring buoy. In this phase, it tests the consequence of shifting the vessel at the mooring buoy and blocks of faster locking vessels behind to an earlier position in the locking sequence. It makes the shift only if the total waiting time for all vessels would be reduced. In the second phase, it explores whether interleafing vessels from different queues (changing lockages from turnbacks to exchanges or vice versa) has a net beneficial effect. It continues until no such change generates an improvement. To impose equity, vessels are shifted into higher priority classes with the passage of a stated time interval and the first vessel to lock must be chosen from the highest priority class or from a mooring buoy that is blocking a vessel in the highest priority class.

We compare solutions from the IP model with solutions from the heuristic for a battery of test problems (using expected values of processing times and setup times) and find that the solutions are identical in the majority of cases. Where the solutions differ, the heuristic is found to generate improvements over FIFO that are very close to the improvements generated by the IP. Accepting the heuristic as a good surrogate for the optimizing IP, we then use it in the Arena model of the system and compare the results against FIFO and other scheduling rules to illustrate how stochastic phenomena affect the benefits achievable from "optimizing" schedules with different constraints to impose equity.

References

[1]   R. M. Nauss, Optimal Sequencing in the Presence of Setup Times for Tow/Barge Traffic Through a River Lock, *European Journal of Operational Research,* 187 (3), 1268-1281, 2008.

[2]   Smith, L.D., Sweeney D.C. II, and Campbell, J.F. (2009) Simulation of Alternative Approaches to Relieving Congestion at Locks in a River Transportation System, *Journal of the Operational Research Society*, 60, 519-533.

[3]   Smith, L.D., R.M. Nauss, Investigating Strategic Alternatives for Improving Service in an Inland Waterway Transportation System (2010), *International Journal of Strategic Decision Sciences*, in press.

1. Midnight speaker:
Marius Solomon

Title
"Those magnificent researchers and their flying benchmark problems"

Abstract
I will illustrate why the longevity of benchmark problems from different fields is not the same.

The Multi–Constraint Team Orienteering Problem with Multiple Time Windows

Wouter Souffriau

Information Technology, KaHo St.-Lieven, Belgium

Pieter Vansteenwegen

Centre for Industrial Management, Katholieke Universiteit Leuven, Belgium

Greet Vanden Berghe

Information Technology, KaHo St.-Lieven, Belgium

Dirk Van Oudheusden

Centre for Industrial Management, Katholieke Universiteit Leuven, Belgium

Email: wouter.souffriau@kahosl.be

The Orienteering Problem (OP) is a combinatorial routing problem of which the goal is to find a tour that maximises the total score earned by visiting vertices. A set of vertices is given, determined by their coordinates and a score. The pairwise travel times between the vertices are known. The total travel time should not exceed a predetermined time budget. Each vertex can be visited at most once.

The OP serves as the starting point for modelling tourist trip design problems [2, 4]: a tourist who wants to visit a city or region is limited in time, and cannot visit every tourist attraction. Therefore, the tourist has to make a selection of the most interesting places to visit, within the limited time frame. The score represents the estimated personal interest of the tourist in the location. The time budget obviously represents the maximal amount of time the tourist has available. Solving the OP results in a personal trip for the tourist. However, while executing the planned trip, unexpected events often make the solution infeasible. Therefore, a fast algorithm is needed in order to dynamically recalculate the plan. Well known OP extensions are the Team OP (TOP), and the OP with Time Windows (TW). The former allows modelling trip planning problems for multiple days, the latter allows defining a period for each vertex in which the visit has to take place.

The scope of this paper is the Multi–Constraint TOP with Multiple TWs (MCTOPMTW), in

which each location is extended with Z attributes for each day. Z additional knapsack constraints are defined, which limit the selection of locations, together with the time constraints. In the envisioned tourist application (http://www.citytripplanner.com), the additional constraints will involve budget limitations for entrance fees and "max-n type" for each day and for the whole trip (e.g. a maximum number of museums to visit on the first day). In addition, the proposed model addresses the main drawbacks of the TOPTW model of Vansteenwegen et al. [4]. The new model allows defining different TWs on different days and more than one TW per day. The (T)OP with multiple (and different) TWs was recently discussed and tackled by Tricoire et al. [3], without extra constraints.

This abstract defines the MCTOPMTW mathematically, proposes a fast local search based metaheuristic algorithm and presents promising experimental results.

1 Mathematical Model

The MCTOPMTW can be formulated as an integer program: given are M tours and N locations with a non-negative score S_i, Z attributes and W TWs; location 1 is the starting location, location N is the end location; the shortest path between location i and location j requires time t_{ij}, the Euclidean distance between them; $x_{ijm} = 1$ if, in tour m, a visit to location i is followed by a visit to location j, 0 otherwise; $y_{iwm} = 1$ if location i is visited during TW w in tour m, 0 otherwise; s_{im} is the start of the visit at location i in tour m; O_{iwm} and C_{iwm} are the opening and closing times of TW w of vertex i in tour m; e_{imz} is the cost associated with knapsack constraint z for location i in tour m; E_z is the cost budget of knapsack constraint z; L is a large constant. The total score of the selected visits has to be maximised.

$$Max \sum_{m=1}^{M} \sum_{w=1}^{W} \sum_{i=2}^{N-1} S_i y_{iwm} \tag{1}$$

Subject To:

$$\sum_{m=1}^{M} \sum_{j=2}^{N} x_{1jm} = \sum_{m=1}^{M} \sum_{i=1}^{N-1} x_{inm} = M \tag{2}$$

$$\sum_{i=1}^{N-1} x_{ikm} = \sum_{j=2}^{N} x_{kjm} = \sum_{w=1}^{W} y_{kwm}; \forall k = 2, ..., N-1; \forall m = 1, ..., M \tag{3}$$

$$s_{im} + t_{ij} - s_{jm} \leq L(1 - x_{ijm}); \forall i, j = 1, ..., N; \forall m = 1, ..., M \tag{4}$$

$$\sum_{m=1}^{M} \sum_{w=1}^{W} y_{iwm} \leq 1; \forall i = 1, ..., N \tag{5}$$

$$\sum_{m=1}^{M}\sum_{w=1}^{W}\sum_{i=1}^{N}e_{imz}y_{iwm} \leq E_z; \forall z = 1, ..., Z \tag{6}$$

$$\exists w \in 1, ..., W : O_{iwm} \leq s_{im} \leq C_{iwm}; \forall i = 1, ..., N; \forall m = 1, ..., M \tag{7}$$

$$x_{ijm}, y_{iwm} \in \{0, 1\}; \forall i, j = 1, ..., N; \forall w = 1, ..., W; \forall m = 1, ..., M \tag{8}$$

The objective function (1) maximises the total collected score. Constraint (2) guarantees that all tours start in vertex 1 and end in vertex N. Constraints (3) and (4) determine the connectivity and time line of each tour. Constraints (5) guarantee that every vertex is visited at most once. Knapsack constraints (6) limit the selection by constraining attributes of the vertices, used to define budget and max-n type constraints. Constraints (7) restrict the start of the visit to one of the time windows. Note that the model also enables max-n type and budget constraints to be defined per day, e.g. visit maximum one church on the first day, or spend at most 100\$ on the second day. In this case, an extra knapsack constraint is added for that particular day. Moreover, constraints (5) can also be expressed by a special case of general knapsack constraints (6).

2 Algorithm

The algorithm for tackling the MCTOPMTW is based on the Iterated Local Search (ILS) for the TOPTW of Vansteenwegen et al. [4]. In order to add diversification, the ILS approach is hybridised with GRASP (Algorithm Listing 1), as GRASP has proven to work well for the TOP [2].

for *greed=0.89; greed>0.59; greed-=0.01* **do**

    while *NumberOfIterationsNoImprovement < 100* **do**

        Solution = GRASP(greed);

        if *Solution > BestFound* **then**

            BestFound=Solution;

            NumberOfIterationsNoImprovement=0;

        else

            NumberOfTimesNoImprovement++;

        Shake();

Return BestFound;

Algorithm 1: Hybrid ILS–GRASP for the MCTOPMTW

The GRASP procedure iteratively adds visits to the current solution, allowing only feasible solutions. This procedure is governed by a greediness parameter, which varies from 0.89 (close to best–improving local search) to 0.60 (more randomness). In order to escape from local optima, the shake procedure removes R visits from the current solution, starting from vertex S. R and S

are dynamically updated to steer diversification. More details about this updating process can be found in [4].

As the ILS–GRASP algorithm iteratively adds visits to and removes visits from the current solution, an efficient mechanism is designed to evaluate the knapsack constraints. Efficient value propagation dynamically maintains a neighbourhood structure of possible visits for each tour. Also, the knapsack constraints' slack values serve as a basis for calculating the heuristic value of an insertion of a vertex in a tour, which is used by the GRASP procedure.

3   Experimental Results

Garcia et al. [1] designed MCTOPTW test instances based on available test sets for the TOPTW [4]. We extended the MCTOPTW instances with one money budget constraint, ten max-n type constraints and with m ranging from 1 to 4 instead of 2. The instances are designed in such a way that high quality TOPTW solutions are also feasible high quality solutions of the new MCTOPTW instances. Different time windows in different tours are not included in these instances, but that would have no influence on the performance of this algorithm. Multiple time windows are also not included, but these can be modelled by copying the location, giving each copy one time window and adding an extra knapsack constraint.

The algorithm is allowed to run 10 times on a 2.5GHz Intel Xeon processor with 4 GB of RAM. The results are very good: one average run has a score gap (with the known high quality solutions) of only 3.26%, using 1 second of computation time. When the best solution of 10 runs is used, the initial high quality solution was found in 27% of the test instances and this solution was improved in 9% of the instances. More extensive experimental results indicate that hybridising ILS with GRASP significantly increases the performance of the algorithm.

References

[1] A.Garcia, P. Vansteenwegen, W. Souffriau, M. Liniza and O. Arbelaitz, "Solving multi-constrained team orienteering problems to generate tourist routes", *Discrete Optimization*, under review.

[2] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe and D. Van Oudheusden, "A Path Relinking Approach for the Team Orienteering Problem", *Computers & O.R.*, in press.

[3] F. Tricoire, M. Romauch, K.F. Doerner, R.F. Hartl, "Heuristics for the Multi-Period Orienteering Problem with Multiple Time Windows", *Computers & O.R.* 37, 351-367 (2010).

[4] P. Vansteenwegen, W. Souffriau, G. Vanden Berghe and D. Van Oudheusden, "Iterated Local Search for the Team Orienteering Problem with Time Windows", *Computers & O.R.* 36, 3281-3290 (2009).

Integrated Crew Pairing and Crew Assignment by Dynamic Constraint Aggregation

François Soumis, Mohammed Saddoune, Issmail Elhallaoui, Guy Desaulniers

École Polytechnique de Montréal and GERAD
Case postale 6079, Succ. Centre-ville
Montréal, Canada, H3C 3A7
Corresponding author email: francois.soumis@gerad.ca

1 Introduction

The *airline crew scheduling problem* is one of the most important planning problems faced by the airlines because the total crew cost is considered, next to the fuel cost, the largest single expense of an airline. Given a schedule of flights to be operated by the same aircraft fleet, it consists of determining, for the available crew members, least-cost schedules that cover all flights and respect various safety and collective agreement rules. For large fleets, this problem is usually addressed using a two-stage sequential solution approach. In the first stage, least-cost crew pairings are built to cover each flight by a crew. A pairing is a sequence of one or more duties separated by rest periods and a duty is a sequence of flights separated by connections forming a work day. A pairing must start and end at the same crew base and must respect various feasibility rules. It can contain deadhead flights that are used for repositioning purposes. This first stage problem is called the *crew pairing problem*. Given the computed pairings, the second stage problem, called the *crew assignment problem*, consists of constructing monthly schedules for the available crew members at each base. A schedule is a sequence of pairings interspersed by rest periods that may contain days off. Additional crew members expected to be in reserve can also be scheduled at a high penalty cost. In this paper, we consider the construction of anonymous pilot schedules (called bidlines) that are later assigned to the pilots according to their preferences and seniority. We propose a model and a method for solving the integrated crew scheduling (ICS) problem.

The crew pairing and the crew assignment problems have been widely studied separately as mentioned in the recent surveys [1] and [2]. For the last two decades, column generation has been the leading solution methodology for the crew pairing problem. The literature on the crew assignment problem is more heterogeneous as the problem definition often differs from one paper to another. Metaheuristics and mathematical-programming based solution methods, including column generation, have been developed for this problem. To our knowledge, the ICS problem was addressed only in [3]. The authors proposed two duty-based integer linear programming models which rely on the assumptions that all duties can be generated a priori and that deadheads can be introduced as needed in the schedule without any additional costs. Using a heuristic branch-and-bound algorithm, the authors succeeded to solve small-sized instances (up to 210 flights or 40 crew members).

2 Model and solution method

When building pilot schedules, the ICS problem can be formulated as a set partitioning problem using the following notation. *F:* set of flights to cover*; B:* set of crew bases; q_b: number of pilots available at base *b* (excluding those expected to be in reserve)*; β:* penalty cost for each extra pilot; S^b : set of feasible schedules for pilots at base *b*; c_s : cost of schedule *s*; a_{fs} : equal to 1if schedule *s* covers flight *f* and 0 otherwise*;* x_s : binary variable indicating whether or not schedule *s* is chosen; y_b : surplus variable indicating the number of extra pilots required at base *b*. The proposed model is:

$$Minimize \quad \sum_{b \in B} \sum_{s \in S^b} c_s x_s + \beta \sum_{b \in B} y_b \tag{1}$$

$$\text{subject to:} \quad \sum_{b \in B} \sum_{s \in S^b} a_{fs} x_s = 1, \quad \forall f \in F \tag{2}$$

$$\sum_{s \in S^b} x_s - y_b \le q_b, \quad \forall b \in B \tag{3}$$

$$x_s \in \{0, 1\}, \quad y_b \ge 0, \quad \forall b \in B, \ s \in S^b. \tag{4}$$

The objective function (1) minimizes the sum of the schedule costs and the penalty costs for the additional pilots. Flight coverage is imposed by the set partitioning constraints (2), whereas soft pilot availability per base is ensured by (3).

For very small-sized instances, model (1)-(4) can be tackled using column generation embedded into a variable fixing procedure. In such a solution approach, column generation is used for solving the linear relaxation of (1)-(4), which is then called the master problem. At each iteration, this method solves the master problem restricted to a subset of its variables, called the restricted master problem (RMP), and several subproblems (one per crew base) that identify negative reduced cost columns (variables) to add to the current RMP. The column generation process stops when all subproblems fail to generate a negative reduced cost column. For the ICS problem, the subproblems correspond to resource constrained shortest path problems that are defined on time-space networks. Such a network allows the construction of all feasible schedules for a base, including the construction of their pairings and duties. It involves 11 arc types, namely, flight, deadhead, rest, day off and waiting arcs among others. Any feasible schedule for the corresponding base corresponds to a path from a source node to a sink node in this network. However, not all paths represent a feasible schedule. Resource constraints are, thus, used to restrict path feasibility. For the ICS problem considered, nine resources are required to take into account all feasibility rules for the duties, the pairings, and the schedules themselves. To derive an integer solution, column generation is embedded into a variable fixing procedure that sets to 1 any x_s variable with a fractional value larger than a given threshold or, as a second option, pairs of flights to be covered consecutively by the same pilot.

For practical size instances, column generation becomes inefficient for solving the ICS problem because the number of set partitioning constraints (2) is large and the number of flights per schedule can easily exceed 30, yielding a highly degenerate master problem. To overcome this difficulty, we propose to combine the column generation method with a bi-dynamic constraint aggregation (BDCA) method (see [4] and [5]). A BDCA method uses an aggregated restricted master problem (ARMP) that is obtained by aggregating clusters of the RMP set partitioning constraints (2) and keeping one representative constraint for each cluster. This constraint aggregation can vary throughout the solution process. Since each constraint (2) is associated with a flight, a cluster corresponds to a non-empty subset of flights and an aggregation is performed according to a partition Q of the flights into clusters. For the ICS problem, we start the solution process using an initial cluster for each pairing of a heuristic solution computed for the crew pairing problem. A variable x_s is said to be compatible with partition Q if the set of flights covered by the corresponding schedule is the union of some clusters in Q. Otherwise, it is declared incompatible. The ARMP only contains compatible x_s variables and all y_b variables. Once solved, it provides a primal and an aggregated dual solution. To allow pricing all (compatible and incompatible) x_s variables, this dual solution is disaggregated using a repetitive shortest path procedure. A newly generated variable x_s compatible with the current partition Q can be added to the ARMP without modifying this partition. At the opposite, an incompatible variable cannot be added to the ARMP without modifying it. By working with a reduced sized master problem, DCA reduces the impact of degeneracy and speeds up the computational time per column generation iteration.

To maintain a higher level of aggregation during the solution process, a partial pricing strategy favoring the generation of compatible or slightly incompatible columns is used. This strategy associates with each variable x_s a number of incompatibilities with respect to the current partition and uses a sequence of phases that gradually allows the pricing of variables with a higher number of incompatibilities. To further speed up the solution process, the size of the subproblem networks is reduced according to partition Q. The reduction procedure selects a subset of clusters according to their corresponding aggregated dual values and removes all the arcs in the networks that would yield a disaggregation of these clusters if they were used. When no columns can be generated using the reduced networks, the subproblems are solved again using the complete networks. In this paper, we propose to enhance this reduction procedure by using a neighborhood (defined by a time slice) that restricts cluster selection and increases the chances of generating columns that are complementary. Such a neighborhood is kept for several column generation iterations before switching to another one. Finally, the overall BDCA/column generation method is made heuristic by stopping prematurely the column generation process when the decrease in the objective value realized in a predetermined number of iterations is deemed insufficient.

3 Computational results

Computational experiments were performed to evaluate the savings that can be obtained by solving the ICS problem instead of solving sequentially the crew pairing and the crew assignment problems. For these experiments, we considered seven instances involving between 1011 and 7527 flights over one month, and three crew bases each. All tests were conducted on a Linux PC machine clocked at 2.8 GHz.

Table 1 reports the test results. We observe that, compared to the sequential approach, the integrated approach yields significant savings: on average, 4.02% on the total cost and 5.51% on the number of schedules. However, the computational times are on average 3.02 times longer with the integrated approach. Thus, we conclude that integrating crew pairing and crew assignment can be highly profitable, but this requires much longer computational times.

Instance	Flights	Sequential approach			Integrated approach					
		CPU (min)	Total cost	No. scheds	CPU (min)	CPU INT/SEQ	Total cost	Svgs (%)	No. scheds	Svgs (%)
I-1	1011	4.0	767754	33	6.4	1.73	723684	5.74	31	6.06
I-2	1463	5.8	957989	34	14.7	2.53	923426	3.60	31	8.82
I-3	1793	11.4	1313391	47	34.7	3.04	1272972	3.07	43	8.51
I-4	5466	522.6	3502527	145	966.3	1.84	3382494	3.42	137	5.51
I-5	5639	231.9	4835090	247	1401.7	6.04	4637323	4.09	241	2.42
I-6	5755	260.0	5144122	223	783.0	3.01	4796863	6.75	209	6.27
I-7	7527	507.6	6536094	305	1518.2	2.99	6437594	1.50	302	0.98
Average						**3.02**		**4.02**		**5.51**

Table 1: Results obtained by the sequential and the integrated approaches

References

[1] B. Gopalakrishnan and E.L. Johnson, "Airline crew scheduling: State-of-the-art", *Annals of Operations Research* 140(1), 305-337 (2005).

[2] D. Klabjan, "Large-scale models in the airline industry", in *Column generation*, G. Desaulniers, J. Desrosiers, and M.M. Solomon (eds), 33-65, Springer, New York, 2005.

[3] F.-M. Zeghal and M. Minoux, "Modeling and solving a crew assignment problem in air transportation", *European Journal of Operational Research* 175(1), 187–209 (2006).

[4] I. Elhallaoui, D. Villeneuve, F. Soumis and G. Desaulniers, "Dynamic Aggregation of Set Partitioning Constraints in Column Generation", *Operations Research* 53, 632–645 (2005).

[5] I. Elhallaoui, G. Desaulniers, A. Metrane and F. Soumis, "Bi-dynamic constraint aggregation and subproblem reduction", *Computers and Operations Research* 35, 1713-1724 (2008).

Inventory routing problems

M.Grazia Speranza

Department of Quantitative Methods

University of Brescia

Email: speranza@eco.unibs.it

Extended abstract

The class of Inventory–Routing Problems (IRPs) includes a variety of different optimization problems that, though often very different from each other, all consider a routing and an inventory component of an optimization problem. Time may be discrete or continuous, demand may be deterministic or stochastic, inventory holding costs may be accounted for in the objective function or not. When the inventory cost is not included in the objective function, an inventory capacity at the customers is defined. IRPs have received little attention, if compared to vehicle routing problems. However, the interest in this class of problems has been increasing from the beginning of the eighties. Some pioneering papers appeared in the eighties, while several papers appeared in the last two decades and some surveys (see, e.g., [4, 5, 7, 8]) summarize the state of the art.

In this talk the class of inventory routing problems will be presented. After a review of the literature, with motivations to study this class of problems, the talk will focus on a class of discrete time IRPs that include in the objective function transportation and inventory costs. Contributions in this area will be reviewed, starting from the simplest models to the most complex ones.

The simple case of one origin and one destination will be presented first. Even in this simple case the problem to minimize the sum of transportation and inventory costs is NP-hard. This proves that the complexity of IRPs may come from the routing side of the problems but may also come from other sides. While simple shipping policies, such as shipping full loads, may perform poorly, simple yet effective frequency based policies will be presented.

Then, the case of a general distribution problem will be considered. A product is distributed from a common supplier to a set of retailers over a time horizon. At each discrete time a quantity is produced or made available at the supplier and a quantity is consumed at each retailer. A starting inventory level at the supplier is given. Each retailer has an inventory capacity and a starting inventory level. If the deterministic order-up-to level management policy is applied, when a retailer is visited, then the quantity shipped to the retailer is such that the inventory level reaches

its maximum level, that is the inventory capacity. If instead the maximum level policy is applied, when a retailer is visited, then the quantity shipped to the retailer is less constrained, the only constraint being that the capacity must not be exceeded. The inventory cost is charged both at the supplier and at the retailers. Shipments from the supplier to the retailers are performed by vehicles of given capacity. Each vehicle route visits retailers that are served at the same time. The transportation cost to travel directly from a retailer to another retailer is known. The goal is to determine for each time the quantity to ship to each retailer, and the routes visiting all the retailers served at that time at minimum cost.

This problem, as most IRPs, is very complex to solve because it combines the complexity of time-dependent decisions with the complexity of the traditional vehicle routing problems. However, compared with most of the studied IRPs, it has a relatively simple structure. To investigate the problem characteristics in depth and to understand which are the most appropriate exact and heuristic techniques, the case of one vehicle will be addressed. A branch-and-cut algorithm for this case was proposed in [2].

Firstly, a hybrid solution approach will be presented (see [1]). A tabu search heuristic for the above described problem is shown to perform well. However, while the tabu search is not sufficient to guarantee really high quality solutions, the quality of the solutions can be improved by means of ad hoc designed MILP models, embedded in the tabu search framework, that explore in depth some promising parts of the solution space. Computational results on a set of benchmark instances show the excellent performance of this hybrid heuristic.

Then, the exact solution of the problem by means of a branch-and-price-and-cut method will be addressed (see [3]). A mixed integer linear programming formulation for this problem was proposed in [2]. A new different mixed integer linear programming formulation will be presented and it will be shown that the new one has a stronger relaxation. For any retailer, the sequence of replenishments that occur at the retailer has the structure of the well-known (single-item single-level) dynamic lot-sizing problem [9]. For this problem, it is known that a formulation with separate variables for setups, that is decisions on the periods of replenishment, and production/order quantities is weak and that a path formulation over the time periods is stronger. In the stronger formulation binary variables indicate that a retailer has successive replenishments at two given times t and t' and that no replenishment takes place between t and t'. Some families of valid inequalities will also be presented to strengthen further the formulation. Branch-and-price methods have been successfully applied for other IRPs, the first being due to Christiansen and Nygren [6]). A branch-and-price-and-cut algorithm for the solution of this problem will be presented, based on the stronger formulation, and compared with the branch-and-cut algorithm proposed in [2].

References

[1] C. Archetti, L. Bertazzi, A. Hertz and M.G. Speranza, A Hybrid Heuristic for an Inventory-Routing Problem, Working paper 317, Department of Quantitative Methods of the University of Brescia, 2009 (submitted).

[2] C. Archetti, L. Bertazzi, G. Laporte and M.G. Speranza, "A Branch-and-Cut Algorithm for a Vendor Managed Inventory Routing Problem", *Transportation Science* 41, 382-391 (2007).

[3] C. Archetti, N. Bianchessi, S. Irnich and M.G. Speranza, "A Branch-and-Price-and-Cut Algorithm for an Inventory Routing Problem", work in progress (2009).

[4] L. Bertazzi, M.W.P. Savelsbergh and M.G. Speranza, "Inventory Routing", in: *Vehicle routing: Latest advances and new challenges*, B. Golden, R. Raghavan, and E. Wasil (eds), 49-72, Springer, 2008.

[5] A.M. Campbell, L. Clarke, A. Kleywegt and M.W.P. Savelsbergh, "The Inventory Routing Problem", in: *Fleet Management and Logistics*, T.G. Crainic and G. Laporte (eds), 95-113, Kluwer, Boston, 1998.

[6] M. Christiansen and B. Nygreen, "A Method for Solving Ship Routing Problems with Inventory Constraints", *Annals of Operations Research* 81, 357-378 (1998).

[7] J.-F. Cordeau, G. Laporte, M.W.P. Savelsbergh and D. Vigo, "Short-Haul Routing", in: *Handbooks in Operations Research and Management Science: Transportation*, G. Laporte and C. Barnhart (eds), Vol. 14, Elsevier, Amsterdam, 2007.

[8] A. Federgruen and D. Simchi–Levi, "Analysis of Vehicle Routing and Inventory-Routing Problems", in: *Handbooks in Operations Research and Management Science*, M.O. Ball, T.L. Magnanti, C.L. Monma and G.L. Nemhauser (eds), Vol. 8, 297-373, North-Holland, Amsterdam, 1995.

[9] Y. Pochet and L.A. Wolsey, *Production Planning by Mixed Integer Programming*, Springer Series in Operations Research and Financial Engineering, Springer, New York (2006).

Robust Optimization in Distribution Networks: The Vehicle Rescheduling Problem

Remy Spliet

Econometric Institute

Erasmus University Rotterdam

P.O. Box 1738, 3000 DR Rotterdam, The Netherlands

Email: Spliet@ese.eur.nl

Adriana F. Gabor

Econometric Institute

Erasmus University Rotterdam

Rommert Dekker

Econometric Institute

Erasmus University Rotterdam

1   Introduction

The capacitated vehicle routing problem (CVRP) is a classical problem in operations research. Consider a depot where goods are stored and a set of locations which have nonnegative demand for the goods. A set of vehicles of finite capacity is available to transport the goods from the depot to the customers. The vehicles start and end their routes at the depot. Costs are incurred for traveling from one location to another. The CVRP is to find a routing schedule that describes the sequence of locations that is visited by every vehicle, in such a way that the total traveling costs are minimized. The CVRP is known to be an NP-hard problem.

In situations of frequent periodic deliveries, it is beneficial for operational processes to determine the moment of delivery before the orders are placed. It is for instance very costly, if at all possible, to roster delivery handling personnel one day before they are needed. It is therefore very common to determine a long term schedule, henceforward *master schedule*, that serves as a schedule for every periodic delivery over a certain period of time in which multiple deliveries are made. In the classical CVRP, demand is deterministic and known. A situation that often occurs in practice is that demand varies per periodic delivery and only becomes apparent at a late moment. For

example, in the retail industry it is very common that the orders of the individual stores are placed only a few days, sometimes even just one day, before delivery. Designing a robust master schedule is the main goal of our research project.

Such a master schedule is made before demand realizations become apparent. As a result the master schedule will not always be feasible as for instance high demand might cause the capacity of a vehicle to be insufficient to make deliveries to all locations on its route planned in the master schedule. In such cases the master schedule needs to be deviated from. Moreover, low demand may lead to inefficient use of vehicle capacity, such that lower traveling costs might be obtained by deviating from the master schedule. The construction of a new schedule when demand realizations become known, will be referred to as *rescheduling*. The main focus of our paper [4] is to model and solve the vehicle rescheduling problem (VRSP).

2 Problem Description of the VRSP

Consider a directed complete graph $G = (V, E)$. The set of nodes $V = \{0, 1, ..., n\}$ correspond to a single depot 0 and the customers $V' = \{1, ..., n\}$. For every edge $(i, j) \in E$ traveling costs $c_{ij} \geq 0$ are given that satisfy the triangle inequality. We suppose that an unlimited number of vehicles of capacity $Q \geq 0$ is at our disposal. Furthermore, for every location $i \in V'$ the demand q_i is given such that $Q \geq q_i > 0$. The vehicles will be used to supply demand.

A *route* $r \subset E$ is defined as a cycle in G including the depot and is called feasible if the total demand of the locations visited on that route does not exceed the capacity Q. A routing schedule S is a collection of edge-disjoint routes such that all customers are included in exactly one route. A schedule is called feasible when all routes it includes are feasible. The set of all feasible schedules is $\mathbb{S}$.

Assume that a master schedule S_M is available. Note that this master schedule need not be feasible as capacity restrictions might be violated by demand realizations. Next a *deviation* is defined per location. We say that the new schedule does not deviate for location l when all locations visited prior to l on the route in both the master and the new schedule are the same and that it deviates otherwise. To be more precise, suppose location l is visited on route r_M in the master schedule S_M, Let r_M be represented by the set of locations: $\{i_1^M, ..., i_v^M, l, ..., i_k^M\}$. In the new schedule S_R, l is visited on route r_R, which is represented by $\{i_1^R, ..., i_w^R, l, ..., i_m^R\}$. When $v = w$ and $i_1^M = i_1^R, ..., i_v^M = i_w^R$, the new schedule does not deviate for location l; otherwise it does deviate. Therefore, if location l deviates, it immediately follows that all subsequent locations on the same route also deviate. As an example, suppose $r_M' = \{1, 2, 3, 4, 5, 6\}$ and $r_R' = \{1, 2, 4, 5, 7, 6\}$. The new schedule does not deviate for locations 1 and 2, but it does deviate for all locations 3

through 7 (we know that location 3 is moved to another route and 7 is moved from another route).

Whenever a new schedule deviates for location $i \in V'$, costs $u_i \geq 0$ are incurred. Let us therefore define the following function describing the incurred deviation costs for location i given an master and a new schedule, S_M and S_R respectively:

$$U(S_M, S_R, i) = \begin{cases} u_i, & \text{if } S_R \text{ deviates from } S_M \text{ for location } i; \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

It is now possible to fully define the VRSP as finding a feasible schedule S_R such that it minimizes the total traveling and deviation costs for a given master schedule S_M:

$$\text{(VRSP)} \min_{S_R \in \mathbb{S}} \Big[\sum_{(i,j) \in S_R} c_{ij} + \sum_{i \in V'} U(S_M, S_R, i) \Big] \tag{2}$$

Note that the CVRP is a particular instance of the VRSP when $u_i = 0 \ \forall i \in V'$. As CVRP is NP-hard, so is VRSP.

3 Solution Methods

The VRSP can be modeled as an mixed integer linear programming problem. As the VRSP is closely related to the CVRP, many MIP formulations of the CVRP can be extended to a VRSP formulation. As stated in [3], the most successfully implemented formulation of the CVRP is the two commodity flow formulation, introduced in [1]. It lends itself to be solved using advanced branch-and-cut methods as is done in [1]. We have extended this formulation to a two commodity flow formulation of the VRSP. Randomly generated instances of the VRSP of up to 30 customer locations have been solved to optimality by direct implementation of this model into ILOG CPLEX 10.1.

To solve larger instances of the VRSP, the two-phase heuristic is introduced. The main idea behind the two-phase heuristic is to start with the possibly infeasible master schedule S_M and modify it to make it feasible. In the first phase of the heuristic, customers are removed from infeasible routes starting with the last location on a particular route. This is continued until the remaining locations do not exceed the capacity of the vehicle. Next, what remains of the master schedule will be completed again in the second phase by adding the removed locations to existing routes or constructing new routes, such that all locations are visited and the resulting schedule is feasible. The problem of adding the removed locations to the schedule can be modeled as a CVRP and solved accordingly.

The two-phased heuristic has some nice properties. We have proven that this heuristic always generates a schedule with the minimal number of deviation locations. Moreover, this enables us to prove that when the deviation costs are above a certain value that is solely dependent on the

traveling costs, the two-phase heuristic always generates the optimal schedule. A tight example shows that this bound cannot be improved. Finally an analytical bound can be given on the relative difference between the costs of using the schedule produced by the two-phase heuristic and the costs of using the optimal schedule. Again it is shown that this bound cannot be improved by means of a tight example.

A second heuristic is also introduced, the modified savings heuristic. This heuristic is based on the savings heuristic introduced in [2]. The modified savings heuristic can easily be extended to incorporate features that are relevant in practical situations, like for instance time-window constraints.

Over 1000 test cases have been generated with varying parameters. The two-phase heuristic generates schedules that are closer to the optimal schedule than the schedules produced by the modified savings algorithm in most test cases. However, as the number of customers present in a test instance increase, the computation time of the two-phase heuristic increases exponentially while the computation time of the modified savings heuristic increases linearly.

References

[1] R. Baldacci, E. Hadjiconstantinou and A. Mingozzi, "An Exact Algorithm for the Capacitated Vehicle Routing Problem Based on a Two-Commodity Network Flow Formulation", *Operations Research*, 52(5), 723-738 (2004).

[2] G. Clarke and J.W. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points",*Operations Research*, 12(4), 568-581(1964).

[3] G. Laporte, "What you should know about the vehicle routing problem", *Naval research Logistics*, 54(8), 811-819 (2007).

[4] R. Spliet, A.F. Gabor and R. Dekker, "The Vehicle Rescheduling Problem", Report Econometric Institute EI2009-43, Econometric Institute, Erasmus University Rotterdam, 2009.

Template-based Tabu Search Algorithms for the Multi-Period Vehicle Routing Problem with Consistent Service Constraints

Foteini Stavropoulou

Center for Operations Research & Decision Systems

Department of Management Science and Technology

Athens University of Economics and Business

Panagiotis P. Repoussis

Center for Operations Research & Decision Systems

Department of Management Science and Technology

Athens University of Economics and Business

Christos D. Tarantilis

Center for Operations Research & Decision Systems

Department of Management Science and Technology

Athens University of Economics and Business

Email: tarantil@aueb.gr

1 Introduction

Discrete combinatorial optimization problems of high computational complexity appear in a multitude of real-world applications, such as vehicle routing, assignment, scheduling, network design and many other fields of utmost economic, industrial and scientific importance. Taking into account that these problems in practice are usually large-scale, the significance and the challenge of developing efficient and effective solution approaches is obvious [1, 2, 3]. The main focus in this paper is given on the multi-period vehicle routing problem with consistent service constraints (ConVRP). The ConVRP is introduced recently in [4] and can be used to model a variety of real-life applications, such as parcel deliveries and collection services.

The ConVRP is a NP-hard combinatorial optimization problem and deals with planning and

managing a fleet of homogeneous capacitated vehicles to meet customer needs or demand for services. It involves the design of a set of minimum cost vehicle routes to service a set of customers with known demands over multiple days. Customers may receive service either once or with a predefined frequency; however frequent customers must receive consistent service throughout the planning period, such that the maximum service difference between the earliest and latest service times over multiple days does not exceed a maximum time limit. The goal is to minimize the total distance traveled by the vehicles such that all customers are served by exactly one vehicle without violating capacity, route duration and consistent service constraints.

2 Solution Frameworks

This paper presents two new template-based Tabu Search (TS)[6] metaheuristic algorithms for solving the ConVRP. Both solution approaches utilize the rationale of template schedules-routes introduced in [4]. Based on the consistent service constraint template routes are constructed considering only the frequent customers. The main effort is to ensure that their visiting sequence remains relatively the same throughout the planning period. This precedence principle is believed to adhere to the consistency constraints. Following this principle the ConVRP can be decomposed into two sub-problems. The master sub-problem seeks to design a template route-schedule in order to determine the sequence of frequent customers, while the slave sub-problem seeks to find the actual daily service schedules for both frequent and non-frequent customers on the basis of the master template schedule. In a manner similar the proposed template-based Tabu Search algorithms operate on a dual mode basis. The master mode refers to the application of the TS algorithm on the template, while the slave mode refers to the application of the TS algorithm on the actual daily schedules. Regarding the latter, apart from vehicle capacities and route duration restrictions, the visit requirements and the customers' precedence constraints, as dictated by the corresponding template schedule, are also considered.

The main difference between the proposed solution frameworks lies in the way the template schedule is constructed and manipulated. The first solution framework initially consider that a complete template schedule is constructed using the well-known savings construction heuristic of Clarke and Wright [5]. Subsequently, a TS algorithm is repeatedly applied to improve either the current template schedule (master mode) or the appropriately re-constructed daily schedules after each change of the template schedule (slave mode). On the other hand, the second framework builds the template schedule in a sequential fashion. In particular, a partial template schedule is gradually constructed for each day of the planning period and TS is applied sequentially at the end of each construction phase (master mode). After each successful cycle the template schedule is updated, and the oscillations between construction and improvement phases are repeated until the

template schedule is complete. To this end, the daily schedules are determined using an insertion-based construction heuristic and TS is applied to improve each daily schedule (slave mode).

In both cases, the proposed TS algorithm operates on the basis of simple edge-exchange neighborhood structures (i.e. 2-Opt, 1-1 Exchange and CROSS-Exchange) using a direct gain oriented neighborhood evaluation scheme. In an effort to provide a balance between diversification and intensification, an auxiliary augmented objective function is used to guide the search process. More specifically, a long term memory structure is introduced that keeps track the appearance frequency of particular solution attributes, combined with a simple penalization scheme. The main goal is to penalize edges (pairs of customers) that frequently change state (added and/or deleted) whenever the search is confined during the exploration of the solution space. Finally, a reactive mechanism is introduced that dynamically updates and controls the size of tabu lists.

3 Computational Results

The benchmark data sets with up to 199 customers introduced in [4] are used as the baseline for the evaluation of the proposed template-based Tabu Search algorithms, abbreviated hereafter as SF1 and SF2. Table 1 summarizes the computational results obtained (TD and NV stands for traveling distance and number of vehicles respectively). For all problem instances, SF1 and SF2 yield high quality solutions and proved to be highly competitive with very reasonable computational time requirements (maximum running time 600 seconds). Compared to the state-of-the-art Record-to-Record travel metaheuristic algorithm (ConRTR) of Groër et al [4], SF1 and SF2 improved the average distance traveled by 4.18% and 2.14%, while further reductions in terms of number of vehicles utilized are also obtained.

Table 1: Comperative analysis

Instance	ConRTR		SF1			SF2		
	TD	NV	TD	NV	Gap%	TD	NV	Gap %
Problem 1	2282.14	5	2210.56	4	-3.14	2245.08	5	-1.62
Problem 2	3872.86	11	3605.68	9	-6.90	3720.33	9	-3.94
Problem 3	3628.22	7	3476.88	6	-4.17	3589.41	6	-1.07
Problem 4	4952.91	12	4600.73	10	-7.11	4804.28	10	-3.00
Problem 5	6416.77	16	5756.23	14	-10.29	5917.75	13	-7.78
Problem 6	4084.24	5	4096.87	5	0.31	4096.86	5	0.31
Problem 7	7126.07	12	6905.56	11	-3.09	6817.08	10	-4.34
Problem 8	7456.19	9	7331.11	8	-1.68	7301.12	8	-2.08
Problem 9	11033.54	14	10556.4	13	-4.32	10453.36	12	-5.26
Problem 10	13916.8	18	13304.50	17	-4.40	13425.77	16	-3.53
Problem 11	4753.89	7	4810.95	6	.1.20	5127.15	6	7.85
Problem 12	3861.35	10	3607.88	8	-6.56	3690.25	8	-4.43
Average	6115.415	10.5	5855.28	9.25	-4.18	5932.37	9	-2.41

4 Conclusions

This paper presents template-based Tabu Search algorithms for solving the multi-period vehicle routing problem with consistent service constraints. The architecture of both solution frameworks utilize the concept of template schedules, introduced by Groër et al [4], on a dual mode basis for producing high quality solutions. Computational experiments on benchmark data sets of the literature validate the effectiveness and the efficiency of the proposed solution approaches.

References

[1] P.P. Repoussis, C. D. Tarantilis, and G. Ioannou, "An Arc-Guided Evolutionary Algorithm for the Vehicle Routing Problem with Time Windows". *IEEE Transactions on Evolutionary Computation* 13(3), 624-647 (2009)

[2] C.D. Tarantilis, "Solving the vehicle routing problem with adaptive memory programming methodology", *Computers and Operations Research* 32, 2309-2327 (2005).

[3] C.D. Tarantilis and C.T. Kiranoudis, "Boneroute: An adaptive memory-based method for effective fleet management", *Annals of Operations Research* 115, 227-241 (2002).

[4] C. Groër, B.L. Golden and E. Wasil, "The Consistent Vehicle Routing Problem", *Manufacturing & Service Operations Management*, Articles in Advance, 1-14 (2008).

[5] G. Clarke and J.W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points", *Operations Research* 12, 568-581 (1964).

[6] F. Glover, "Future paths for integer programming and links to artificial intelligence", *Computers & Operations Research* 13, 5, 533-549 (1986).

Mathematical models and tabu search heuristic for two-echelon location-routing problem in freight distribution

Claudio Sterle

Department of Computer Science and Systems

University of Naples "Federico II", Via Claudio 21, Naples, Italy

Email: claudio.sterle@unina.it

Maurizio Boccia

Department of Engineering,

University of Sannio, Corso Garibaldi 104, Benevento, Italy

Teodor Gabriel Crainic

NSERC Industrial Research Chair in Logistics Management

ESG - Université du Quebec à Montreal

Antonio Sforza

Department of Computer Science and Systems

University of Naples "Federico II", Via Claudio 21, Naples, Italy

1. Two-echelon freight distribution system

In the last years great interest has been addressed to the freight distribution systems in the context of City Logistics problems.

The more relevant aspect of the City Logistics operations can be identified in the plants for the consolidation of the flows entering and leaving the urban areas, the so called platforms or City Distribution Centers (CDC), in a single-echelon system. Their function is the rationalization of the movements in the urban areas, consolidating in a single point the freights to and from the city. They are basically devoted to reduce the fragmentation of all the movements that do not pass through other platforms or warehousing point. Their main targets are the increasing of the loading factor of the vehicles and the improvement of the coordination among the different subjects. CDCs are in general stand-alone facilities situated close to the access or ring highways, or they may be part of air, rail or navigation terminals.

The CDCs certainly have improved the freight distribution in urban areas in the last years, but the initial success of the related single echelon system has showed some deficiencies for what concerns

its application in big cities, where the freight flows have increased significantly and the trend is not going to change. The reasons at the base of this situation are:

- CDCs located rather far from the center. If the aim is to minimize the number of trucks in the urban areas, then heavy trucks should be used in order to consolidate on the same vehicle as many orders as possible. This implies that there will be large trucks moving within the urban areas, performing long routes to serve all the final customers, with difficulty in respecting the delivery time-windows.

- The particular structure of city center of big cities. Big cities are very constrained areas not only for what concerns the density of population and land use, but especially for the road network, characterized by a wide variety of streets of different width, one way streets, few and limited zones for parking, interdicted zones to the trucks etc..

.

For these reasons in the last years new structures for freight distribution systems have been proposed, based on more intermediate facilities. Two-echelon systems have been recently proposed in Crainic et al. [1], Crainic et al. [2], Gragnani et al. [3].

The two-echelon City Logistics concept builds on and expands the CDC idea. City Distribution Centers form the first level of the system and are located on the outskirts of the urban zone. The second echelon is constituted of satellite platforms (satellites) where the freight coming from the CDCs may be consolidated into vehicles suitable for dense city zones. Satellites perform limited activities. In this way at satellites no special infrastructures and functions have to be built, but existing facilities can be used, like for example underground parking slots or municipal bus depots, or spaces like city squares and therefore no high additional costs have to be sustained (Crainic et al. [1]) for satellite activities.

Two types of vehicles are involved in a two-echelon City Logistics system, urban-trucks and city-freighters, and both are supposed to be environmentally friendly. Urban-trucks move freight to one or more satellites, whereas city-freighters are vehicles of small capacity that can travel along any street in the city-center to perform the final distribution.

2. Exact models and tabu search heuristic for two-echelon location-routing problem

The design of a two-echelon freight distribution system is a strategical and tactical decisional problem. The aim is to define the location and the number of the two different kinds of facilities, the assignment of customers to each open secondary facility and of satellites to platforms, the size of two different vehicle fleets (urban trucks and city freighters) and the related routes on each echelon. The problem has been modeled as a two-echelon (multilevel) location-routing problem (2E-LRP). This problem is NP-hard since it arises from the combination of two NP-hard problems, facility location

(FLP) and vehicle routing (VRP). To the best of our knowledge, multi-level location-routing problems have not yet been addressed either with exact or heuristic methods.

Several models for the two-echelon location-routing problem are presented. Three of them derive directly from the classical formulations proposed in Toth and Vigo [4] for the VRP. Another one is based on a multi-depot vehicle-routing formulation (MDVRP) proposed by Dondo and Cerdà [5], which uses assignment and sequencing variables.

The hardness of 2E-LRP allows to solve previous models just on small instances (2 platforms, 10 satellites and 25 customers) with the usage of commercial solver Xpress 7.0. For this reason a tabu search (TS) heuristic approach is proposed to solve 2E-LRP on large instances.

This heuristic is based on the decomposition of the problem in its two main components, i.e. two location-routing problems. Each component, in turn, is decomposed in its two composing sub-problems, i.e. the capacitated facility location problem and the multi-depot vehicle routing problem.

The heuristic builds on the two-phase iterative approach, proposed by Tuzun and Burke [6], and on the nested approach of Nagy and Salhy [7], hence it can be defined as an "iterative-nested approach".

The TS starts with an initial feasible solution, obtained solving the arising multi-level capacitated facility location problem and tries to improve it in two phases:

1. Location phase: a tabu search is performed on the location variables in order to determine a good configuration of facilities to be used in the distribution system. The passage from a configuration to another is obtained through the usage of add and swap moves. The two moves are performed sequentially, first swap moves and then add moves. The swap moves keep the number of facilities unchanged but locations change. Swap moves are performed until a maximum number of iterations without improvement is reached. Then an add move is performed, until a stopping criterion is satisfied.

2. Routing phase: for each location solution determined during the location-phase, a tabu search is performed on the routing variables. The initial routes are built with Clarke and Wright algorithm and then improved by local search. Finally a tabu search based on insert and swap moves is performed.

The key element of the proposed heuristic resides in the combination and integration of the location and routing solutions on each echelon and of the location-routing solutions of the two echelons, in order to obtain a solution that is globally good.

Concerning the combination of single echelon solutions, in the location phase of the algorithm, a TS is performed on the location variables, starting from the configuration with the minimum number of open facilities. For each of the location configurations, another TS is run on the routing variables in order to obtain a good routing for the given configuration. Therefore each time a

move is performed on the location phase, the routing phase is started in order to update the routing according to the new configuration.

Concerning instead the combination of location and routing solutions of the two echelons, each time a change of the demand assigned to a set of open satellites occurs and a pre-defined condition on facility and vehicle capacity is satisfied, then the location-routing problem of the first echelon should be re-solved in order to find the best location and routing solution to serve the new demand of the satellites.

Tabu Search heuristic has been tested on several sets of small, medium and large instances (up to 5 platforms, 20 satellites and 200 customers). The sets differ for the spatial distribution of facilities and customers. Each instance has been solved with different settings of the tabu search parameters. Tabu Search results have been compared with bounds obtained solving small instances of the 2E-LRP by commercial solver Xpress 7.0 and medium and large instances by a simple decomposition approach. The obtained results show that the proposed Tabu Search is able to find good solutions with limited computation time.

References

[1] CRAINIC T.G., RICCIARDI, N., STORCHI, G.: Advanced freight transportation systems for congested urban areas. Transportation Research Part C: Emerging Technologies, 12, 2, 119-137, (2004).

[2] CRAINIC T.G., RICCIARDI, N., STORCHI, G.: Planning models for city logistics Operation. CIRRELT Technical report, (2007).

[3] GRAGNANI, S., VALENTI, G., VALENTINI, M.: City Logistics in Italy: A National Project. In Taniguchi, E. and Thompson, R.G., editors, Logistics Systems for Sustainable Cities, 279293, Elsevier, Amsterdam, (2004).

[4] TOTH, P., VIGO, D.: The Vehicle Routing Problem. SIAM, Philadelphia, (2002).

[5] DONDO, R., CERDA, J.: A cluster-based optimization approach for the multi-depot heterogeneous fleet vehicle routing problem with time windows. European Journal of Operational Research, 176, 14781507, (2007).

[6] TUZUN, D., BURKE, L.I.: A two-phase tabu search approach to the location routing problem, European Journal of Operational Research, 116, 8799, (1999).

[7] NAGY, G., SALHI, S.: Nested Heuristic Methods for the Location-Routeing Problem. The Journal of the Operational Research Society, 47, 9, 1166-1174, 1996.

A Branch-and-Price-and-Cut method for Tramp Ship Routing and Scheduling

Magnus Stålhane, **Henrik Andersson**, **Marielle Christiansen**

Norwegian University of Science and Technology

E-mail: magnus.stalhane@iot.ntnu.no

Jean-Francois Cordeau

Canada Research Chair in Logistics and Transportation and HEC Montreal

Guy Desaulniers

GERAD and Ecole Polytehnique de Montreal

1 Introduction

In a global economy, transportation of goods between the continents is continuously increasing. For most goods the only real option is to transport it by sea. There are huge investments involved in purchasing and maintaining a fleet of ships. A new bulk ship (Panamax-size) costs approximately 35 million USD to purchase, and has daily operating expenses of several thousand USD. Thus it is important for the shipping companies to have as high utilization as possible of their ships, in order to make a reasonable profit.

The shipping segment of trade can roughly be divided into three modes of transportation - industrial, tramp and liner, see [7]. Industrial shipping is characterized by the cargo owner also handling the shipping. Thus the objective is to transport a set of cargoes as cheaply as possible, using the owner's own fleet. In tramp shipping the ships act like taxis, being paid for transporting a specific cargo between an origin and a destination. At any given point in the planning horizon a tramp shipping company typically have one set of cargoes it has already committed itself to carry, as well as one set of optional cargoes it might choose to carry if the fleet has sufficient capacity and it is profitable to do so. In liner shipping the ships act like buses, sailing a pre-determined route and follow a published itinerary, giving the visited ports and the corresponding time of the visit. Customers may then book cargo space on the ship between two given ports on the ship's route.

Previously the tramp shipping market was dominated by many smaller companies, each owning a few ships and only operating within a small geographical region, or in long-hauling between

certain geographical regions. However, in the last couple of decades there have been many mergers and acquisitions in the tramp shipping segment, and we now have a new situation where the market is dominated by fewer, but larger, shipping companies, see [2] for more information. Whereas 20 years ago a scheduler in a tramp shipping company had about 5 ships to schedule, he now has 20, 30, or maybe 50 ships. At the same time the development in communication technology over the past 20 years has made the market for cargoes more transparent. Consequently the schedulers now have greater choice in which cargoes to bid on, but at the same time greater competition which squeezes the profit margins. All these changes in the tramp shipping industry have made routing and scheduling decisions much harder, and thus there is an increased need for decision support systems to help the shipping companies make better decisions.

In this work a typical problem faced by a tramp shipping company is studied. The aim is to select a portfolio of cargoes that maximizes the tramp shipping company's profit. This problem is a maritime version of the well-studied pick-up and delivery problem with time windows (PDPTW), that originates from land-based transportation. For a thorough description of the PDPTW we refer to [3].

In recent years some work on the maritime-PDPTW has surfaced in the literature, both on exact solution approaches and heuristics. [4] and [5] solve variations of tramp shipping problems using path based formulations. The paths are pre-generated and used as input in a set partitioning formulation. [1] and [6] describe heuristics for the maritime-PDPTW. Both papers use a neighborhood structure to search the solution space in order to find good solutions to the problem.

The purpose of this paper is to present a tailor-made branch-and-price-and-cut algorithm for tramp ship routing and scheduling. Solution methods are based on PDPTW literature from road-based problems, but adapted to fit in with the maritime environment and its special features. We also look at extensions of the PDPTW that are unique for tramp shipping problems, and discuss how the solution approach can be modified to accommodate them.

2 Problem description

The problem considered corresponds to a multiple vehicle pick-up and delivery problem with time windows and capacity constraints. The fleet of ships is heterogeneous, and may vary in cargo capacity, speed and cost structure. The ships have different initial positions, and may be positioned anywhere at the end of the planning horizon. During the planning horizon the shipping company has a number of mandatory and optional cargoes to transport. Each cargo has a given size, and is to be picked up in one port within a given time window, and delivered to another port within another time window. For this work the company receives a lump sum, and the aim is to select a portfolio of cargoes that maximizes the total profit of the shipping company.

There are many possible extensions to the maritime-PDPTW described above, reflecting common practices in tramp shipping. In shipping the cargoes are relatively large compared to the size of the ships, and splitting a cargo between two ships may be possible in order to increase the utilization of the ships. Another common practice in shipping is to charter in additional ships in busy periods in order to handle more cargoes. In previous papers, this has been modelled on a per-cargo basis (e.g in [1] and [6]). However, it is more realistic for a shipping company to charter in one ship and use it to transport several cargoes, thus incurring the fixed part of the chartering cost only once. In a sub-segment of tramp shipping, called project shipping, it is also usual for cargoes to be coupled. This means that the ship have to either transport all the coupled cargoes, or none of them. Mostly these linked cargoes share destination (origin), but have different origins (destinations). Sometimes the cargo owner also demands that the different cargoes are delivered (picked up) together, by the same ship. These extensions, and their impact on the maritime-PDPTW model will be presented.

3   Solution method

To solve this problem we have implemented a branch-and-price-and-cut algorithm, based on the work presented in [8]. The problem is decomposed into a master problem, and one subproblem for each ship. The subproblems are solved as shortest path problems with resource constraints, using dynamic programming. We use non-elementary paths to solve the subproblems, because in maritime transportation the travel times are long compared to the width of the time windows, and consequently the possibilities of cycles occurring in the paths are limited. Thus 2-cycle-elimination constraints in the sub-problem are sufficient to remove most of the cycles. Domination criteria are used to remove sub-optimal paths from the subproblem at the earliest opportunity.

The master problem is solved as a set packing problem using the simplex algorithm. Cuts are added to the master problem in order to strengthen the formulation. Several branching schemes are incorporated into the algorithm, and the branching is done by altering the structure of the subproblems, or modifying the master problem.

For the extensions of the maritime-PDPTW presented above, we have further modified the solution approach. For split loads additional constraints have to be added to the master problem to insure that the entire cargo is transported, and quantity variables are introduced to decide how much of each cargo goes on a given ship. In the subproblem cargo quantity can no longer be a constraint used to limit the number of cargoes on-board, as we no longer know the quantity.

In the extension with project shipping, constraints have to be added to the sub-problem making sure that if one coupled cargo is picked up by the ship, then the cargoes to which it is coupled must also be picked up by the same ship. No modifications are necessary in the master problem.

Chartering in additional ships is solved by adding one new subproblem for each possible ship type to charter in. In the master problem these ships are treated as the operator's own fleet, but the subproblem for these chartered ships are more complex. To capture the daily charter cost, the subproblem needs to be modified to have start day and end day as decision variables and to have this time specific cost added to both arcs, nodes and waiting time. The fixed part of the chartering cost is added to all arcs leaving the origin node to pick up a cargo.

4   Computational Results

Computational results based on test cases from the tramp shipping industry will be presented. The effect of different branching-rules, node selection policies, and cuts will also be tested and compared.

References

[1] G. Brønmo, M. Christiansen and K. Fagerholt and B. Nygreen, "A multi-start local search heuristic for ship scheduling - a computational study", *Computers & Operations Research* 34, 900-917 (2007).

[2] M. Christiansen, K. Fagerholt and D. Ronen, "Ship Routing and Scheduling: Status and Perspectives", *Transportation Science* 38, 1-18 (2004).

[3] J. Desrosiers, Y. Dumas, M.M. Solomon and F. Soumis, "Time constrained routing and scheduling", in *Handbooks in Operations Research and Management Science: Network Routing*, M.O. Ball and T.L. Magnanti and C.L. Monma and G.L. Nemhauser, (eds.), 35-139, North-Holland, Amsterdam, (1995).

[4] K. Fagerholt and M. Christiansen, "A combined ship scheduling and allocation problem", *Journal of the Operational Research Society* 51, 834-842 (2000).

[5] K. Fagerholt, "Ship scheduling with soft time windows: An optimisation based approach", *European Journal of Operational Research* 131, 559-571 (2001).

[6] J.E. Korsvik, K. Fagerholt and G. Laporte, "A tabu search heuristic for ship routing and scheduling", *Forthcoming in Journal of the Operational Research Society, 21 pages.*, (2009).

[7] S.A. Lawrence, *International sea transport: the years ahead*, Lexington, MA (1972).

[8] S. Røpke and J-F. Cordeau, "Branch-and-Cut-and-Price for the Pickup and Delivery Problem with Time Windows", *Transportation Science* 43, 267-286 (2008).

Distance-based path relinking for the vehicle routing problem

Kenneth Sörensen

Faculty of Applied Economics

University of Antwerp

Prinsstraat 13, 2000, Antwerp, Belgium

Email: kenneth.sorensen@ua.ac.be

Marc Sevaux

Lab-STICC

Université Bretagne-Sud

F-56321 Lorient, France

Email:marc.sevaux@univ-ubs.fr

1 Path relinking for the vehicle routing problem

Path relinking is a relatively new metaheuristic technique for combinatorial optimization, proposed by Glover (see e.g. [3]). Path relinking attempts to find new good solutions by examining the solutions that are on a *path* from an initial (incumbent) to a final (guiding) solution. By definition, each move on the path makes the solution more different from the initiating solution and more similar to the guiding solution. Moving on the path is done by a *neighbourhood operator*, just like in any local search algorithm. The technical difference with ordinary local search is that the neighbourhood search *strategy* that decides which move to execute is not based on the quality of the resulting solution, but on the *distance in the solution space* between the resulting solution and the guiding solution. A move that takes the solution closer to the guiding solution will be preferred over one that takes it further away, regardless of the quality of the resulting solution. Constructing a path relinking procedure therefore requires us to select a move operator to use and a distance measure *in the solution space* between two solutions. The distance measure than can be used to determine whether a move brings a solution closer to the guiding solution and whether the resulting solution can be considered to be "on a path from incumbent to guiding solution".

Usually, path relinking is not used as a standalone solution method, but combined with other metaheuristics, such as tabu search or GRASP, see e.g. [1,7,9,10].

Although the capacitated vehicle routing problem (CVRP) is one of the best-known combinatorial optimization problems, path relinking approaches for this problem are few and far between. This is partially due to the fact that a VRP solution is most naturally represented as a *set of permutations of customers*, each member of the set representing a tour. Contrary to problems that have a natural binary or vector representation, it is not immediately obvious what is meant by

"moving along a path from incumbent to guiding solution". To the best of our knowledge, the only application of path relinking to the CVRP is due to Ho and Gendreau [6]. Application of path relinking to other routing problems can be found in [5] (for the vehicle routing problem with time windows), [8] (for the capacitated arc routing problem with time windows), [13] (for the team orienteering problem), and [2] (for the multi-compartment vehicle routing problem).

2   Distance-based path relinking

Glover and Laguna [4] state that path relinking algorithms should move from an initial (incumbent) to a guiding solution by progressively introducing attributes contributed by the guiding solution. In a sense, path relinking *transforms* the incumbent solution in the guiding solution, one step at a time. This can be illustrated by considering the specific case of scatter search, essentially a restricted version of path relinking for continuous problems. In scatter search, new solutions are found by taking linear combinations of existing solutions. Any linear combination of two solutions lies on the "shortest path" (i.e. the straight line) connecting these two solutions. The key property is that any convex linear combination x of solutions a and b satisfies the equality that the sum of its distance to the initial and guiding solutions is equal to the distance between these two solutions.

$$d(a, x) + d(x, b) = d(a, b). \tag{1}$$

For permutation problems such as the CVP, the "shortest path" between two solutions is not a universally agreed-upon notion. Existing path relinking approaches for vehicle routing problems have therefore been based on ad-hoc procedures that do not have any relationship with the neighbourhood search operators used in the other parts of the algorithm. However, given any local search operator, a distance between any two solutions can be calculated, corresponding to the minimal number of moves required to transform the first solution into the second one. A distance measure therefore gives rise to a list of moves that, when executed in a specific order, result in a set of intermediate solutions on the "shortest path" between guiding and incumbent solution. For each of the intermediate solutions, equation (1) holds.

For the CVRP, many different move types have been defined in the literature (swap, remove–insert, 2-opt, and many more). Each of these move types gives rise to an associated distance measure. Some of these distance measures are easy to calculate, whereas others are NP-hard (see [11] for an overview). The aim of this (ongoing) research is to investigate different move types and their associated distance measures to see how we can construct an efficient path relinking procedure. Other research questions that need to be answered include:

1. How do we choose the initiating and the guiding solution?

2. Given the list of solutions that transforms the incumbent into the guiding solution, in which order do we execute this list of moves?

3. Which solution from the path do we return upon termination?

4. Can a path-relinking approach be used as a stand-alone optimizer or do we need to integrate it in a (local search) procedure?

3   Previous work and experiments

A *proof-of-concept* procedure was developed and published in [12]. In this contribution, we developed a path relinking procedure based on the "add-remove" edit distance, which corresponds to the minimal number of "add-remove" moves that have to be made to transform a solution into another one. One of the goals of this research is to generalize this procedure and perform the same analysis for more move types and their corresponding distance measures, resulting in procedure like the one depicted in Algorithm 1.

Algorithm 1: Path relinking for the VRP pseudo code

Input: Two solutions s and t

Output: A solution u on a path from s to t

Calculate $n = d(s, t)$ and determine the list $M = \{m_1, \ldots, m_{n-1}\}$ of moves that transforms s into t;

Set $u_0 = s$;

for $i = 1$ *to* $n - 1$ **do**
$\quad \lfloor$ Perform move $m_i \in M$ $(u_{i-1} \rightarrow u_i)$;
Choose solution u to return from the set of generated solutions $\{u_0, \ldots, u_{n-1}\}$;

Although some promising results have already been obtained, this research is ongoing. Detailed results will be presented at the conference.

References

[1] L. Alfandari, A. Plateau, and P. Tolla. A path relinking algorithm for the generalized assignment problem. In M.G.C. Resende and J.P. De Sousa, editors, *Metaheuristics: Computer Decision-Making*, pages 1–18. Kluwer, Boston, 2004.

[2] A. El Fallahi, C. Prins, and R. Wolfler Calvo. A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem. *Computers & Operations Research*, 35(5):1725–1741, 2008.

[3] F. Glover. Scatter search and path relinking. In D. Corne, M. Dorigo, and F. Glover, editors, *New Ideas in Optimization*, pages 297–316. McGraw–Hill, 1999.

[4] F. Glover and M. Laguna. Fundamentals of scatter search and path relinking. *Control and Cybernetics*, 29(3):653–684, 2000.

[5] H. Hashimoto and M. Yagiura. A path relinking approach with an adaptive mechanism to control parameters for the vehicle routing problem with time windows. In J. Van Hemert and C. Cotta, editors, *Evolutionary Computation in Combinatorial Optimization*, pages 254–265. Springer, 2008.

[6] S.C. Ho and M. Gendreau. Path relinking for the vehicle routing problem. *Journal of Heuristics*, 12:55–72, 2006.

[7] C.A.S. Oliveira, P.M. Pardalos, and M.G.C. Resende. GRASP with Path-Relinking for the quadratic assignment problem. In *Experimental and Efficient Algorithms*, pages 356–368. Springer, 2004.

[8] M. Reghioui, C. Prins, and N. Labadi. GRASP with path relinking for the capacitated arc routing problem with time windows. In *Applications of Evolutionary Computing*, pages 722–731. Springer, 2007.

[9] M.G.C. Resende and C.C. Ribeiro. GRASP and path relinking: recent advances and applications. Technical report, AT&T Labs Research, April 6 2003.

[10] M.G.C. Resende and R.F. Werneck. A hybrid multistart heuristic for the uncapacitated facility location problem. Technical Report TD-5RELRR, AT&T Labs Research, 2003.

[11] K. Sörensen, M. Reimann, and C. Prins. Permutation distance measures for memetic algorithms with population management. In *Proceedings of MIC 2005, the sixth Metaheuristics International Conference*, Vienna, Austria, August 22–26, 2005.

[12] K. Sörensen and M. Sevaux. Path relinking for the vehicle routing problem using the edit distance. In *Proceedings of MIC 2005, the sixth Metaheuristics International Conference*, Vienna, Austria, August 22–26, 2005.

[13] W. Souffriau, P. Vansteenwegen, G. Vanden Berghe, and D. Van Oudheusden. A path relinking approach for the team orienteering problem. *Computers & Operations Research*, To appear.

Link Transmission Model: an efficient dynamic network loading algorithm with realistic node and link behaviour

Chris M.J. Tampère, Ruben Corthout, Francesco Viti & Dirk Cattrysse

Katholieke Universiteit Leuven

Center for Industrial Management, Traffic & Infrastructure

Celestijnenlaan 300A, PO Box 2422

B-3001 Leuven, Belgium

E-mail: chris.tampere@cib.kuleuven.be

1   Introduction

Dynamic Traffic Assignment (DTA) is a set of criteria through which the demand for mobility is distributed over time and space on a transport network. The role of DTA is, in essence, to provide with a functional relationship between the demand for mobility and the network supply. As such, it is crucial for any DTA model to dispose of a realistic model for the supply behavior: the dynamic network loading (DNL) component. Because usually the DNL simulation needs to be evaluated multiple times – for instance in iterations towards system or user optimal equilibrium or in day-to-day models – not only the degree of realism, but also the computational speed of a DNL is of paramount importance, especially for large (e.g. regional) networks.

The function of a dynamic network loading model is to propagate traffic from origin to destination over the links and nodes of a traffic network, assuming that the route flows are known. It then calculates as a function of time the generalized costs for traveling each route. In a DTA framework, these costs can lead to adaptation of departure time choice, route choice and/or mode choice or the choice to travel at all (elastic demand).

Many different DNL models have been proposed in literature. In this paper, the Link Transmission Model (LTM) is presented. It is one of few models combining the following properties in a computationally efficient way:

- link model consistent with traffic flow theory
- link model accounts for delays at intersections, also in undersaturated conditions
- node model consistent with all constraints imposed by the link model
- node model imposing node capacity constraints.

Moreover, the LTM is a multi-commodity network flow model. This implies that split rates at each node are consistent with the route flows passing through the node. It reflects the fact that traffic flow

consists of drivers that are not invariant to their direction after the current link. This is in contrast to single-commodity flow like for instance water where the molecules at a valve do not care whether they turn left or right. The multi-commodity property of LTM can be obtained in two distinct ways: (i) either in one network loading iteration by disaggregating flow over links and nodes with respect to the route travelled, or (ii) by iteration towards convergence of a number of single-commodity runs with route flow dependent split rates at each node. The former option is the faster one when the number of routes in the network is limited, so that memory usage and lookup operations remain within bounds. The latter option is more appropriate in larger networks with numerous origin-destination pairs and routes.

2   Link model

The link model describes the flow over a link, given the boundary constraints at the up- and downstream ends of the link. The evolution of flow in space and time is described by traffic flow theoretical models. These models describe for instance the emergence of congestion and the occurrence and propagation of shock waves and expansion fans at the up- respectively downstream interfaces of a traffic jam with the free flow phase. For most network analyses, a first order approximation of these phenomena is sufficient as described in kinematic wave theory (Lighthill & Whitham, 1955). Such models capture the emergence of congestion and the delay imposed on vehicles passing the jams. The original formulation is in terms of average flow, speed and density, the dynamics of which are governed by a continuity equation and an empirical relation (fundamental diagram) $q = f(k)$ between flow q and density k.

An efficient numerical implementation of this theory is based on the cumulative formulation (line integration) of the continuity law and a simplified piecewise linear fundamental diagram after Newell (1993). In essence, this formulation states that the cumulative in- and outflow of a link are maximized under certain constraints. This maximization can be understood as the tendency of drivers to move forward whenever possible, i.e. as long as there is a sufficient stock of vehicles available, there is sufficient space, and it can be done in a safe way (i.e. in safe combinations of flow and density as allowed by the fundamental diagram). For the cumulative inflow, the constraints are:

- maximum inflow allowed by the capacity of the link (capacity $q^c = \max f(k)$ is a property of the fundamental diagram);
- maximum inflow allowed by the upstream node;
- constraint equal to the cumulative outflow backward time earlier, increased by the maximum number of vehicles that can be stored inside the link (backward time and max number of vehicles are both properties of the fundamental diagram and of the link length).

For the cumulative outflow, similar constraints hold:

- maximum outflow allowed by the capacity of the link;

- maximum outflow allowed by the upstream node;
- constraint equal to the cumulative inflow forward time earlier (forward is a property of the fundamental diagram and of the link length).

For fast numerical evaluation, the link model has the advantage that it can be discretized with time steps equal to the minimum of the forward and backward times. These are in the order of ten(s) of seconds, which is substantially larger than the more commonly applied numerical scheme of Daganzo (1995). Moreover, Yperman et al. (2006) show that the numerical error is also smaller.

A potential drawback of using larger calculation time steps is that traffic signal control cycles cannot be explicitly modeled, since this would require reducing the time step again. In such cases, Yperman et al. (2007) show how a flow dependent delay can be introduced into the cumulative link model formulation by Newell (1993). The idea is to add a point queue at the downstream end of each link that vehicles have to cross before entering the node. The size of the point queue is adapted dynamically in order to generate the required undersaturated delay. The same procedure can be applied to model delay at undersaturated priority junctions or to model non-linear free flow branches of the fundamental diagram (i.e. delay incurred by busy, undersaturated traffic on motorways, prior to the activation of bottlenecks).

However, if larger calculation time steps are not really required and the details of traffic signals are relevant for the application, the Link Transmission Model can also be evaluated with smaller time steps in order to explicitly model traffic signal control cycles.

3   Node model

Node models for macroscopic simulation have attracted relatively little attention in literature. Nevertheless, in DNL models for congested road networks, node models are as important as the extensively studied link models. Node models have two functions in DNL models. The first is to seek consistency between the in- and outflows of the incoming and outgoing links respectively; the second to impose supplementary constraints on the outflow of each incoming link (limited supply of the node itself or node supply constraints).

Just like the link model, the node model allows maximum flow over the node under a series of constraints (note that the index i is used for incoming links and j for outgoing links):

$$\max\left(\sum_i \sum_j q_{ij}\right)$$

s.t.

$$
\begin{aligned}
&non-negativity: & &q_{ij} \geq 0 & &\forall i, j \\
&demand\ constraints: & &q_i = \sum_j q_{ij} \leq S_i & &\forall i \\
&supply\ constraints: & &q_j = \sum_i q_{ij} \leq R_j & &\forall j \\
&CTF\ constraints: & &p_{ij} = \frac{S_{ij}}{S_i} = \frac{q_{ij}}{q_i} & &\forall i, j \\
&invariance\ principle: & &if\ \exists i \mid q_i < S_i & &q_i\ invariant\ to\ S_i \rightarrow q_i^c \\
&internal\ node\ supply\ constraints \\
&prevailing\ supply\ constraint & &interaction\ rule
\end{aligned}
\tag{1}
$$

The first three constraints are relatively straightforward: flows must be positive, the total flow q_i coming from an incoming link i cannot exceed the stock of vehicles available at the downstream end of link i (denoted as sending flow S_i); neither can the total flow q_j going to an outgoing link j exceed the space available at the upstream end of link j (denoted as receiving flow R_i). The fourth constraint stems from the multi-commodity character of traffic flow. It states that the turning fractions p_{ij} as found in traffic ready to enter the node from the incoming link i, must also be preserved in the actual flows q_{ij} over the node.

So far, the proposed node model does not differ from those available in literature. However, none of the existing node models simultaneously maximize flow, while at the same time respecting the fifth, sixth and seventh constraint: the invariance principle after Lebacque and Khoshyaran (2005), a series of internal node supply constraints and the prevailing supply constraint interaction rule (Tampère et al., forthcoming). All three specific aspects are explained in the paper.

The internal node supply constraints relate to some internal limited capacity inside the node. For instance, the node may contain conflict points that cannot be simultaneously occupied by two (or more) conflicting flows. Alternatively, a stop line can be an internal supply, the capacity of which is intentionally limited by some traffic signal that only allows traffic to pass during a fraction of total time (i.e. during the green phase).

Finally, the supply constraint interaction rule defines for the case where some receiving flow or internal node supply is the active constraint for at least two incoming links, how this limited supply is distributed over all links competing for the available space, and how this distribution interacts with the first four constraints.

Whereas the explicitly written constraints in (1) are generic for all node models, the internal node supply constraints and the interaction rule are specific for different types of intersections or various driving cultures. Indeed, these are the parts of the model that reflect aggregate driver behavior (taking/giving away priority, gap acceptance, reversed priority, competition for space in congested outgoing links, entering blocked intersections,...).

For relatively simple instances of the generic node model formulation (1), a proof of exactness and a fast, exact solution algorithm is proposed in Tampère et al. (forthcoming). More refined models for common intersection types like priority junctions and roundabouts are currently under development.

4   Conclusion

The Link Transmission Model is a powerful dynamic network loading model suitable to support dynamic traffic assignment models or other network analyses. It offers physically accurate link and node models based on the paradigm of flow maximization under various applicable constraints. To the authors' knowledge, it is the only truly macroscopic model that obeys all traffic flow theoretical requirements and constraints, while still being capable of simulating multi-commodity traffic flows in realistically sized mixed urban and motorway networks.

References

[1] Daganzo, C.F. (1995). The cell transmission model part II: Network traffic. Transportation Research Part B 29, pp. 79-93.

[2] Lebacque, J.P. & Khoshyaran, M.M. (2005). First-order macroscopic traffic flow models: intersection modeling, network modeling. Proceedings of the 16th International Symposium on Transportation and Traffic Theory (ISTTT), pp. 365-386.

[3] Lighthill, M. J., Whitham, G.B. (1955). On kinematic waves II: a theory of traffic flow on long crowded roads. Proceedings of the Royal Society of London.

[4] Newell G.F. (1993) A simplified theory of kinematic waves in highway traffic, Part I: General theory, Part II: Queuing at freeway bottlenecks, Part III: Multi-destination flows. Transportation Research 27B, 281-313.

[5] Tampère, C.M.J. , Corthout, R., D. Cattrysse & L.H. Immers (forthcoming), A generic class of first order node models for dynamic macroscopic simulation of traffic flows, submitted to Transportation Research B

[6] Yperman, I., C.M.J. Tampère, and L.H. Immers (2007), A Kinematic Wave Dynamic Network Loading Model Including Intersection Delays. Proceedings of the 86th Transportation Research Board Annual Meeting, Washington, D.C., U.S.A., January 2007

[7] Yperman, I., S. Logghe, C.M.J. Tampère, and L. H. Immers (2006), The multi-commodity Link Transmission Model for Dynamic Network Loading. Proceedings of the 85th Transportation Research Board Annual Meeting, Washington, D.C., U.S.A., January 2006

Route choice behavior based on license plate observations in the Dutch city of Enschede

Tom Thomas

Centre of Transport Studies

University of Twente

Bas Tutert

Centre of Transport Studies

University of Twente

Tom Thomas

Centre of Transport Studies

University of Twente, p.o.box 217, Enschede, The Netherlands

Email: t.thomas@utwente.nl

1 Introduction

Traffic assignment models consists of two main components: routeset generation and route choice behavior [1]. Traditionally, a routeset is generated in advance. Alternative routes are often chosen by Monte Carlo simulations, in which link resistances are changed randomly [2]. However, routeset generation actually depends on route choice behavior. When travelers choose alternative routes, these routes should be included in the route set.

The problem is that route choice is not sufficiently modeled in traditional assignments. The models are often theoretical, and sometimes calibrated by stated preference surveys, but they are seldom validated by oberved route choices. Studies that link observed route choice behavior to underlying attributes, e.g. [3], [4] and [5], are rare. These studies, however, are useful, because they show to what extend real choices depend on 'objective' attributes, like travel time, and to what extend these choices are based on individual preferences of travelers.

When travelers make choices based on individual preferences rather than economics, this will have consequences for traffic loads on the network. In this study, we use license plate data from the Dutch city of Enschede to analyze empirical route choice behavior We generate a route set of many alternative routes and show how the observed distribution of routes depends on the travel times along these routes. From this we can also estimate the effects of individual preferences on route choice and how this influences the traffic loads on the network.

2 Method

This study is based on the registration of license plates at observation posts along all main roads of the city of Enschede during the off peak (14.00 – 16.00h), and evening rush hour (16.00 – 18.00h) on a Tuesday, and during a Saturday afternoon (13.00 – 15.00h). In total, about 26000 observed cases, evenly distributed over these three periods, were used in this analysis.

We defined a link as an imaginary line between two successive (chronological) observations, and a route as a chain of links. When a license plate, for example, is first registered at observation post A, and last registered at B, then the origin-destination (OD) pair for that case is AB. If the car is also registered at C, the route would consist of the links AC and CB.

Journey times were also registered. By aggregating all measurements from a link, average travel times for that link were estimated. These average travel times are quite accurate, because of the large number of cases used in their estimates. The average travel time of a route was obtained by simply adding all the average link travel times of that route.

Our route set contains all observed routes, but also routes that were not observed. In this case, we avoid a bias, because we also include routes that could have been chosen, but which are not in the observed sample by accident. The route set generation was done as follows. The links themselves form the first set of routes and OD pairs. A new route is generated when a new link connects to the previous (sequence of) link(s). By chaining links, the number of routes and OD pairs is extended. This process is stopped until no new routes are formed, or when a route becomes circular (i.e. when one observation post occurs twice in the sequence), or when a (part of a) route is more than 20 minutes longer than the fastest route between the same posts. In total, we generated about 80 routes on average per OD pair.

In theory, we can obtain the frequency distribution of routes per OD pair. However, because frequencies are small for individual OD pairs, we aggregated OD pairs in groups. We distinguish different travel time classes (4 – 7 min, 7 – 12 min, 12 – 17 min, and 17 – 25 minutes) and grouped OD pairs based on the travel time along the fastest route. We discard the class with very short travel times, because this class only contains few OD pairs for which the route choice may very well depend on other attributes. These OD pairs are therefore not representative.

We then estimated the travel time difference between the fastest route and every other route per OD pair. Based on the travel time difference, we grouped the routes in travel time difference classes (0 – 1 min, 1 – 2 min, 2 – 4 min, 4 – 7 min, 7 – 12 min and 12 – 20 minutes difference). Per class, n_1 is the sum of freqeuncies along the fastest routes (the frequencies for all comparisons are added, i.e. if a fastest route is compared with two other routes, its freqeuncy is added twice), and n_2 is the sum of frequencies along the other routes. Similarly, T_1 is the average travel time along the fastest routes, and T_2 is the average travel time along the other routes in the same class. The aggregated frequencies (n_1 and n_2), and average travel times (T_1 and T_2) are the principal parameters, considered in this study.

3 Results

Route choice may depend on many attributes. Their influence is often hardly known, and if known from for example stated preference surveys, most of these attributes, like for example travel time reliability, cannot be estimated in a straightforward way. In applied assignment models, it is therefore often assumed that travelers choose the fastest route, which will lead to an user equilibrium [6]. However, without impicitly modeling individual preferences, their aggregated effect can be taken into account by estimating the relation between the distribution of observed frequencies and real travel times. We did this, and find the following tight relation for the Enschede data.

$$\frac{n_2}{n_1} = \exp[-1.33(T_2 - T_1)^{0.7} + 1.33(0.5)^{0.7}] \qquad \text{voor } T_2 - T_1 > 0.5 \text{ min} \quad \text{(1a)}$$

$$\frac{n_2}{n_1} = 1 \qquad \text{voor } T_2 - T_1 \leq 0.5 \text{ min} \quad \text{(1b)}$$

This relation is more or less valid for all travel time classes. We therefore conclude that the route choice probablity depends on absolute rather than relative travel time difference. Equations (1a) and (1b) are also valid for the different periods (off peak, rush hour and Saturday afternoon), although the average travel times are different, e.g. the average travel time is on average 10% larger during rush hour than during the off peak.

The calculation of the choice probabilities is now straightforward. If for example an OD pair has two alternatives that are 1 and 3 minutes longer than the fastest route, then according to equation (1a), $n_2/n_1 = 0.60$ and $n_3/n_1 = 0.13$. Thus, in that case, the assigned fractions over routes 1, 2 and 3 are 58%, 35% and 7% respectively.

The probability that a longer route is chosen, declines rapidly with travel time difference. However, because there are many alternatives, a significant fraction of 25% (of the observed cases) does not follow the fastest route. Regarding the network performance, the detour time over the alternative routes is an even more important parameter. This 'occupancy measure' indicates how much of extra load the network has to process due to detours. We find that the average detour time (a combination of number of alternative routes, route choice probability and travel time difference) is maximal for alternatives with a travel time difference of about 5 minutes with respect to the fastest route.

The average detour time (aggregated over all routes) is 0.57 minutes. This is about 8% of the travel time (the average travel time along the fastest route was 7.5 min), which implies that the network has to process 8% of extra load compared to a traditional equilibrium assignment.

This study is based on observed route choice behavior in Enschede. It would be useful to compare these results with observations from other cities, and with observed route choices on highways. Because the number of alternatives is lower for a highway network, we expect that detour factors should also be lower, and thus that traditional assignments will probably show more reliable results for highways. However, this can only be validated by route choice and travel time observations from highways

As mentioned before, most other attributes cannot be estimated in a straightforward way. The hierarchy of roads may be one of the few measurable attributes that has an effect on route choice. Small roads with speed bumps are less comfortable than highways (without congestion). Thus, given similar travel times, it is quite likely that travelers prefer larger roads. Another factor that can be taken into account, is the rate of overlap between routes, e.g. two almost identical routes may be seen as one route by the traveler. The hierarchy of roads and the overlap of routes are to be analyzed in a follow-up study.

Including other attributes, such as those mentioned above, may significantly improve route choice predictions for individual cases. However, it is not likely that this will have an effect on equation (1). In fact, we do not expect that we have introduced a systematic bias by not including other attributes, because they are implicitly in the observed choices, and also not correlated with travel times of different alternative routes. We therefore think that our simple route choice model can already be used to improve traffic assignments in a structural way.

References

[1] P.H.L. Bovy, "On Modelling Route Choice Sets in Transportation Networks: A Synthesis", *Transport Reviews, 29*(*1*), 43-68 (2009)

[2] Y. Sheffi and W.B. Powell, "An algorithm, for the Equilibrium Assignment Problem with Random Link Times", *Networks I2*(*2*), 191-207 (1982)

[3] D. Papinski and D.M. Scott, "GIS Toolkit design and application for route choice analysis: a comparison of observed routes and shortest paths", in TRB proceedings (2009)

[4] R.P. Batley and R.G. Clegg, "Driver Route and Departure Time Choices: The Evidence and the Models", Proceedings of the 33rd Annula UTSG conference (2001)

[5] J.R. Duffell and A. Kalombaris, "Empirical Studies of Car Drive Route Choice in Hertfordshire", *Traffic Engineering and Control, 29* (*7/8*), 398 – 408, 1988

[6] J.G. Wardrop, "Some theoretical aspects of road traffic research, *Proceedings Institute of Civil Engineers*, *part II* (*vol. 1*), 325-378 (1952)

Exact Methods for the Multi-Trip Vehicle Routing Problem

A. Mingozzi

Department of Mathematics

University of Bologna

R. Roberti

DEIS

University of Bologna

P. Toth

DEIS

University of Bologna

Email: paolo.toth@unibo.it

1   Introduction

The *Multi-Trip Vehicle Routing Problem* (MTVRP) is an extension of the Capacitated Vehicle Routing Problem (CVRP) where each vehicle is allowed to perform more routes (trips) during its working period.

The MTVRP is defined on an undirected graph $G = (V', E)$, where $V' = V \cup \{0\}$. $V = \{1, 2, \ldots, n\}$ represents a set of n customers, and 0 represents the depot. With each edge $\{i, j\} \in E$ are associated a travel cost c_{ij} and a travel time t_{ij}. Each customer $i \in V$ requires q_i units of goods from the depot. A fleet M of m identical vehicles is used to fulfill the requests. Each vehicle $k \in M$ has a *capacity Q* and a *maximum working time T*.

A *route* is a simple circuit visiting the depot and some customers and such that the total request of the customers served does not exceed Q. The cost (working time) of a route is defined as the sum of the travel costs (travel times) of the edges traversed. A *schedule* for a vehicle is a subset of routes (*i*) having total working time not exceeding T, and (*ii*) visiting each customer at most once. The cost of a schedule is the sum of the costs of its routes.

The objective of the MTVRP is to design a set of m schedules, one for each vehicle, of minimum total cost and such that all the customers are visited exactly once. The MTVRP appeared first in Fleischmann [4].

In spite of the practical relevance of the problem, to our knowledge, no exact algorithm has been presented in the literature for the MTVRP so far. Several heuristic methods have been proposed

by different authors.

In this work, we present two exact methods for the MTVRP using two different formulations. Both exact methods are based on the solution framework presented in Baldacci et al. [1]. The computational results show that instances with up to 100 customers can be consistently solved to optimality within acceptable comuting times.

2   Mathematical Formulations

Our exact methods use two set-partitioning based formulations of the MTVRP.

Let $\mathscr{R}$ be the index set of all the feasible routes, and let $\mathscr{R}_i \subseteq \mathscr{R}$ be the subset of routes serving customer $i \in V$. Let c_ℓ and τ_ℓ be, respectively, the cost and the working time of route $\ell \in \mathscr{R}$. Let ξ_ℓ^k be a (0-1) binary variable equal to 1 if and only if route $\ell \in \mathscr{R}$ is performed by vehicle $k \in M$. The first formulation we propose is the following:

$$(F1) \quad z(F1) = \min \sum_{\ell \in \mathscr{R}} c_\ell \sum_{k \in M} \xi_\ell^k \tag{1}$$

$$s.t. \sum_{\ell \in \mathscr{R}_i} \sum_{k \in M} \xi_\ell^k = 1, \quad \forall i \in V, \tag{2}$$

$$\sum_{\ell \in \mathscr{R}} \tau_\ell \xi_\ell^k \leqslant T, \quad \forall k \in M, \tag{3}$$

$$\xi_\ell^k \in \{0,1\}, \quad \forall \ell \in \mathscr{R}, \forall k \in M. \tag{4}$$

Constraints (2) impose that each customer is visited exacly once. Constraints (3) specify that the total working time of the routes performed by each vehicle cannot exceed T.

Let $\mathscr{S}$ be the index set of all the feasible schedules, and let $\mathscr{S}_i \subseteq \mathscr{S}$ be the subset of schedules serving customer $i \in V$. Let d_ℓ be the cost of schedule $\ell \in \mathscr{S}$. Let y_ℓ be a (0-1) binary variable equal to 1 if and only if schedule $\ell \in \mathscr{S}$ is in the solution (0 otherwise). The second formulation we propose is the following:

$$(F2) \quad z(F2) = \min \sum_{\ell \in \mathscr{S}} d_\ell y_\ell \tag{5}$$

$$s.t. \sum_{\ell \in \mathscr{S}_i} y_\ell = 1, \quad \forall i \in V, \tag{6}$$

$$\sum_{\ell \in \mathscr{S}} y_\ell = m, \tag{7}$$

$$y_\ell \in \{0,1\}, \quad \forall \ell \in \mathscr{S}. \tag{8}$$

Constraints (6) specify that each customer $i \in V$ must be visited exactly once. Constraint (7) imposes that m schedules are selected.

3   The Exact Methods

Formulations $F1$ and $F2$ are used to derive two exact methods for solving the MTVRP. Both methods are based on the same exact solution framework, whose key components are dual ascent heuristics and exact cut-and-column generation procedures that compute a near-optimal dual solution of the LP-relaxations of $F1$ and $F2$ enforced by valid inequalities. The final dual solution achieved is used to generate two reduced problems $\widehat{F1}$ and $\widehat{F2}$ containing only the variables (routes in $\widehat{F1}$ and schedules in $\widehat{F2}$) having reduced cost smaller than the gap between a known upper bound and the lower bound achieved. The resulting reduced integer problems $\widehat{F1}$ and $\widehat{F2}$ are then solved by a general purpose integer programming solver.

The key components of this solution methods are (i) the dual ascent heuristics that do not require the a-priori generation of all variables, (ii) the original state space relaxation method of the route set $\mathscr{R}$ and schedule set $\mathscr{S}$, (iii) the bounding functions and the multiple dual feasible solutions that reduce the state space graph when solving the pricing problem in the cut-and-column generation procedures, and (iv) the new valid inequalities that enforce the LP-relaxation of $F1$.

The computational results show that the proposed exact methods can solve to optimality several instances with up to 100 customers in reasonable amount of computing time.

References

[1] R. Baldacci, E. Bartolini, A. Mingozzi, and R. Roberti, "An Exact Solution Framework for a Broad Class of Vehicle Routing Problems", *Computational Management Science* accepted for publication (2009).

[2] R. Baldacci, N. Christofides, and A. Mingozzi, "An Exact Algorithm for the Vehicle Routing Problem based on the Set Partitioning Formulation with Additional Cuts", *Mathematical Programming* 115 (2), 351-385 (2008).

[3] R. Baldacci, A. Mingozzi, and R. Roberti, "Solving the Vehicle Routing Problem with Time Windows using New State Space Relaxation and Pricing Strategies", *Submitted for publication* (2009).

[4] B. Fleischmann, "The Vehicle Routing Problem with Multiple Use of Vehicles", Working Paper, Facbereich Wirtschaftswissenschafte Universitat Hamburg, 1990.

[5] E. Taillard, G. Laporte, and M. Gendreau, "The Vehicle Routing Problem with Multiple Use of Vehicles", *Journal of the Operational Research Society* 47, 1065-1070 (1996).

Short-Term Traffic Flow Forecasting by means of Wavepack Analysis and Multi-population Genetic Programming

Athanasios Tsakonas[1]

Department of Financial and Management Engineering

University of the Aegean, Kountouriotou 41, 82100 Chios, Greece

Email: tsakonas@stt.aegean.gr

Matthew G. Karlaftis

Department of Transportation Planning and Engineering

National Technical University of Athens

1 Extended Abstract

Accurate short-term predictions of traffic variables such as volume and occupancy is of outmost importance in transportation research because of the increased demand for reliable Advanced Traffic Management and Traveler Information Systems in urban areas. Literature indicates that real-time short-term prediction models of traffic variables have to be accurate and effective for a given forecasting horizon [1]. Empirical evidence has shown that prediction accuracy is best accomplished by data-driven approaches that construct the underlying rules of complex traffic datasets rather than working based on pre-determined mathematical rules [2]; these models can be parametric (such as ARIMA models) or non-parametric (such as non-parametric regression and neural networks). The essential difference is that, in parametric models, a specific - usually predefined - functional form connects inputs with outputs. In non-parametric models, a functional form is not defined or required, while the algorithms seek for data-specific structures in connecting inputs with outputs (a review of the literature, methodologies and approaches used can be found in Vlahogianni [3]).

Among non-parametric approaches, non parametric regression, Kalman filtering and neural networks have been proven to be most effective in forecasting traffic flow variables because of their propensity to account for a large range of traffic conditions and provide more accurate predictions than classical statistical forecasting algorithms [4], [5]. However, previous research has indicated some shortcomings regarding their efficiency in terms of prediction accuracy; difficulties in modeling the variability observed in freeway traffic flow, particularly in cases of extreme traffic conditions (congestion), has been emphasized both for freeways and urban arterials [6], [7], [8], [9].

---

[1] Corresponding author

When considering short-term prediction systems that operate in real-time and in an "intelligent" technology-based environment, the effectiveness depends, mostly, on predicting traffic information in a timely manner [4]. Real-time system effectiveness depends both on the results and on the time in which these are produced [10]. The computational time to produce a prediction mainly depends on the functional form of the prediction system; empirical results show that data-driven prediction systems that include recursive data-search algorithms exhibit 'best' prediction accuracy, but need extensive computational time for convergence at acceptable results [4]. Moreover, the real-time system's software implementation is crucial; software must be structured into adaptive and configurable modules, enabling application-dependent tradeoffs for timeliness, precision, and accuracy to be negotiated in response to changes in operational parameters [11].

The present paper proposes a flexible and adaptive real-time system for short-term traffic flow prediction. The system is based on the principles of computational intelligence and wavelet packet analysis and is able to develop the proper model for prediction based on prevailing traffic flow conditions. The term wavelet is used to describe families of basis functions having special features. Due to their structure, wavelets have various fundamental properties rendering them highly useful in signal analysis [9]. In order to perform the wavelet decomposition in this system, we use the fast wavelet transform. In electrical engineering this algorithm is referred as subband filtering, and the filters are known as quadrature mirror filters [19]. The principle behind the noise reduction using wavelets is that noise contributes into many coefficients, while the trend contributes to only a few coefficients. Hence, by setting the smaller coefficients to zero, we can nearly optimally eliminate noise while preserving the underlying trend. Wavelet packet -or wavepacket- analysis is an advance regarding removal of the noise content in a signal since they provide a richer signal analysis. Training with time-series data after effectively removing the noise often carries better regression and forecasting results, as the noise content is a stochastic parameter of the signal which by definition is impossible to predict [15].

Genetic programming (GP) [17] belongs to the computational intelligence family of the evolutionary methods, and it has been applied nowadays in a wide range of real-world problems. An extension to the standard GP is to apply a multi-population model. Multi-population models have been used in other forms of evolutionary population, where they have been shown superior performance as compared to single-population models [18]. A multi-population model is a paradigm of a distributed population model in which the subpopulations are kept isolated during the application of genetic operations. The subpopulations focus on the evolutionary process and a migration phase is usually applied at predetermined times in which the subpopulations sent and receive individuals (immigrants) to and from other subpopulations.

The data under examination is taken by a major urban signalized arterial in Athens, Greece, in order to model and forecast the traffic flow. It consists of measurements of the traffic flow every 3 minutes between January and May of 2000. The total length of the area under study is 1.5 km, and there are four traffic measurements (namely L101, L103, L106 and L108). The task is consisted of the

calculation of the value L101 at time t, when the values of L101, L103, L106 and L108 are known at times t-1, t-2, t-3 and t-4. To estimate the noise content, the entropy analysis method S.U.R.E. (Stein's Unbiased Estimate of Risk) [12] was used. For the decomposition and the reconstruction we applied biorthogonal wavelets with 2 vanishing points. To remove the noise content we used soft-thresholding [16] for all signals (L101, L103, L106 and L108). As fitness measure for the genetic programming algorithm the mean magnitude relative error (MMRE) was selected. The system output carried the following results: MMRE in test set: 9.93215, Pred(25): 92.1053 % and Pred(30): 94.0443 %.

The proposed methodology proved to be competitive to previous research. Also, the system automatically selected these input variables that contribute the most to the output, potentially offering knowledge discovery. In addition, the solution form is a practical expression, containing only the elementary mathematical operations, a property that contributes to its portability. Finally, it is worth to note that the system output is in accordance to previous research results, while it promoted specific time-delayed values of the L103, L106 and L108 signals that already are considered as important in the literature. Further research may include training with other genetic programming systems (e.g. GP for the production of neural networks), training of the de-noised signal with other computational intelligence systems, noise removal attempts using other wavelet/wavepacket setups and the application of the methodology to more transportation data.

References

[1] H. C. Lieu, "Traffic estimation and prediction system," TR News, vol. 208, pp. 3-6, May 2000.

[2] G.-L. Chang and C.-C. Su, "Predicting intersection queue with neural network models," Transportation Research Part C, vol. 3, no. 3, pp. 175–191, June (1995).

[3] E. I. Vlahogianni, J. C. Golias and M. G. Karlaftis, "Short-Term Traffic Forecasting: Overview of Objectives and Methods," Transport Reviews, vol. 24, no. 5, pp. 533-557, Sept. 2004.

[4] B. L. Smith and R. K. Oswald, "Meeting Real-Time Requirements with Imprecise Computations: A Case Study in Traffic Flow Forecasting," Computer Aided Civil and Infrastructure Engineering. vol. 18, no. 3, pp. 201-213, May 2003.

[5] Y. Wang, M. Papageorgiou, and A. Messmer, "A Real-Time Freeway Network Traffic Surveillance Tool," IEEE Transactions on Control Systems Technology, vol.14, no. 1, pp. 18-31, Jan. 2006.

[6] B. L. Smith, B. M. Williams and K. R. Oswald, "Comparison of parametric and non-parametric models for traffic flow forecasting," Transportation Research Part C, vol.10, no 4, pp. 303-321, Aug. 2002.

[7] S. Ishak and C. Alecsandru, "Optimizing traffic prediction performance of Neural networks under various topological, input and traffic condition settings," Journal of Transportation Engineering, vol.130, no. 4, pp. 452-465, July 2004.

[8] A. Stathopoulos and M. G., Karlaftis, "A multivariate state-space approach for urban traffic flow modelling and prediction," Transportation Research Part C, vol.11, no. 2, pp. 121-135, April 2003.

[9] E. I. Vlahogianni, M. G. Karlaftis and J. C. Golias, "Optimized and meta-optimized neural networks for short-term traffic flow prediction: A genetic approach," Transportation Research C, vol.13, no. 3, pp. 211-234, June 2005.

[10] K. Shin and P. Ramanathan, "Real-time computing: A new discipline of computer science and engineering," Proc. IEEE, vol. 82, pp. 6-24, Jan. 1994.

[11] J. A. Stankovic, "Misconceptions About Real-Time Computing: A Serious Problem for Next-Generation Systems," Computer, vol.21, no. 10, pp. 10-19, Oct. 1988.

[12] Coifman, R.R.; M.V Wickerhauser, "Entropy-based algorithms for best basis selection," IEEE Trans. on Inf. Theory, vol. 38, 2, pp. 713-718, 1992 .

[13] Mallat, S. "A wavelet tour of signal processing", Academic Press, 1998.

[14] Ling Gao, Shouxin Ren, "Combining direct orthogonal signal correction and wavelet packet transform with partial least squares to analyze overlapping voltammograms of nitroaniline isomers", Electrochimica Acta, Vol. 54 (11), pp 3161-3168, Apr. 2009

[15] D.L. Donoho, I.M. Johnstone, "Ideal Spatial Adaptation by Wavelet Shrinkage," Biometrika, vol 81, 1994.

[16] Angkoon Phinyomark, Chusak Limsakul, Pornchai Phukpattaranont, "A Comparative Study of Wavelet Denoising for Multifunction Myoelectric Control", Int'l Conf. on Comp. and Aut. Eng. (ICCAE 2009), pp. 21-25, 2009.

[17] Koza J.R., "Genetic Programming: On the Programming of Computers by Means of Natural Selection", Cambridge, MA, MIT Press., 1992

[18] F. Fernandez, M. Tomassini, and L. Vanneschi. "An empirical study of multipopulation genetic programming", Genetic Programming and Evolvable Machines, 4(1):21–51, March 2003.

[19] Abramovich F., Baily C.T., Sapatinas T., "Wavelet analysis and its statistical applications", The Statistician, Vol. 49, 1-29, 1999

Robust fleet sizing and allocation in industrial ocean shipping organisations

Panagiotis Tsilingiris, José Fernando Alvarez, Nikolaos M.P. Kakalis[*]

Det Norske Veritas Research & Innovation, Piraeus, Greece, and Høvik, Norway

[*]Email: Nikolaos.Kakalis@dnv.com

Maritime transport is the major transport mode of international trade. In 2007, seaborne trade reached 8.02 billion tons, which accounted for over 80% of world merchandise trade by volume ([1]). Two inherent characteristics of the maritime industry are *strong cyclicality* and *high volatility*, the latter introducing significant uncertainty in maritime strategic decision making. Cash flows in maritime organisations accrue from trading in four distinct markets: the *newbuilding* market, where new ships are ordered at shipyards; the *freight* market, which includes ship chartering and forward freight agreements (FFAs); the *sale and purchase* (S&P) market for second-hand ships; and the *demolition* market, where ships are sold for scrap ([2]).

Fleet Sizing and *Allocation* (FS&A) constitutes a central and complex decision problem met in the strategic and tactical planning of *industrial* shipping organisations, which mainly transport the cargo of the parent company. Industrial shipping organisations use their own vessels and complement their fleet with spot and time charters. In the case of excess fleet capacity, industrial shipping companies can charter out their vessels to others. Therefore, the status of fleet structure and allocation changes over time via several decisions, including the type and number of vessels to build, purchase, sell, charter in, charter out, lay up, or scrap. Moreover, re-allocation of vessels between geographic markets is possible during each planning period. The goal of the fleet manager is to determine the fleet sizing and allocation that maximises the net present value (NPV) of the shipping unit, while accommodating the demand for transport of the parent company. This task is complicated by the significant uncertainty in vessel prices, freight rates, and transport demand that can be observed in the shipping industry over the strategic planning period.

The majority of relevant publications ([3]-[14]) on the maritime FS&A problem do not include cash flows from all four shipping markets, omit uncertainty from their analysis (with the exception of [8]), use single-period models (with the exception of [10] and [14]), and focus on tactical and operational considerations, rather than strategic planning.

In this paper, we propose a robust optimisation approach to the fleet sizing and allocation problem. We factor in the inherent uncertainty and multi-period nature of the problem as well as cash flows from the four shipping markets. Our model accounts for the particularities of industrial carriers' operations, such as supplementing the capacity of the owned fleet by chartering in third-party vessels and chartering out owned vessels during periods of excess capacity.

We derive a mixed integer programming (MIP) formulation of the FS&A problem, as well as its robust counterpart. The objective of the nominal MIP is to maximize the NPV of cash flows accrued from operation on all four shipping markets. We formulate our model over a finite planning horizon, which is in turn subdivided into discrete time periods such as quarters or months. In order to represent the continuation of economic activities beyond the planning horizon, we also include a final *sunset* period. The shipping organisation accrues a credit for each vessel that remains in the fleet in the sunset period, corresponding to the residual value of the vessel and its utilisation over its remaining lifetime. Our model represents the different contract types that are generally used in the maritime sector, including voyage charters, contracts of affreightment, and time charters. In order to limit the combinatorial complexity of the proposed MIP, we group vessels with similar characteristics into a small group of vessel families. We also predetermine a subset of vessel families that are most appropriate to a particular market, based for instance on draft restrictions as well as the ability of the vessels to carry specific types of merchandise.

The core of the proposed model can be viewed as a network flow model, where vessels originate at the shipyard, the second-hand market, the spot market, or the time-charter market. The vessels then flow to different intermediate states, such as deployment to a market, or lay-up. Finally, the vessels leave the network when they are sold, demolished, or returned to their owner. An additional state is included to reduce the dimensionality of our model: the company vessel pool. When a vessel undergoes a transition between two states (for instance, re-deployment from one geographical market to another), it has to pass through the intermediate state. Most of the constraints in the nominal MIP simply ensure the balance of vessels at each node in the network. An important issue regarding the balance of vessels is that the ownership of each vessel must be correctly represented, even as chartered vessels enter and leave the focal fleet.

Additional constraints ensure that demand from the parent company is satisfied and that financial constraints on the fleet owner are properly represented. In particular, we have limited the total investments that can be made by the fleet manager to represent real-world constraints on the credit lines of a fleet owner.

Market conditions that are beyond the control of the fleet manager, such as vessel prices and operational expenses, are exogenous to the model. We assume that the planner has a base-line expectation of how uncertain market prices and charter rates will evolve for each time period in the planning horizon. Based on historical data, the planner has also established the degree to which each uncertain parameter may deviate from the base-line expectation. More precisely, we assume that the uncertain exogenous parameters are independent random variables with density functions that are symmetric and bounded.

We derive the robust counterpart to the nominal MIP model following Bertsimas and Sim [15]. In the robust model, each constraint that contains uncertain parameters is modified by the introduction of slack variables. The amount of slack added to each constraint is dependent on the

potential number of uncertain parameters and the extent to which they might vary from the base-line expectation.

We note that a wide range of attitudes towards risk can be observed in the maritime transport sector. This motivates our usage of the formulation by Bertsimas and Sim [15], which allows us to represent the degree of conservatism of the planner. In this sense, a conservative planner is willing to settle for a lower expected net profit in exchange for a stronger guarantee that a given fleet configuration will remain feasible (i.e. comply with all transport demand and financial constraints). We represent the degree of conservatism of the planner as the number of parameter deviations against which the planner requires guaranteed protection.

We present results from a realistic case study of an industrial ocean shipping company transporting liquid bulk cargo. The contracts examined include time contracts of different durations and spot (voyage) contracts. The initial conditions in the case study resemble the actual market conditions observed within 2009. In the case study we represent the decisions of different fleet managers, each with a different attitude towards financial and operational risk. Having obtained an optimal solution to the robust optimisation problem corresponding to each level of risk, we conduct a post-optimisation analysis using Monte Carlo simulation. Each optimal solution is repeatedly tested against random realisations of the exogenous parameters. Following Bertsimas and Sim [16], we then estimate the *price of robustness* for the focal company. That is, we compare the expected NPV realized by each planner against the number of constraint violations that each experiences. We demonstrate the nature of the insights that a fleet manager might derive by using the proposed model.

References

[1] United Nations Conference on Trade and Development, *Review of maritime transport*, United Nations, New York and Geneva (2008).

[2] M. Stopford, *Maritime Economics*, 3rd edition, Routledge, London and New York (2009).

[3] R. Agarwal and Ö. Ergun, "Ship scheduling and network design for cargo routing in liner shipping", *Transportation Science* 42:2, 175-196 (2008).

[4] J.F. Alvarez, "Joint routing and deployment of a fleet of container vessels", *Maritime Economics and Logistics* 11, 186-208 (2009).

[5] H. Bendall and A. Stent, "A scheduling model for a high speed containership service: a hub and spoke short-sea application", *International Journal of Maritime Economics* 3:3, 262-277 (2001).

[6] S. C. Cho and A. Perakis, "Optimal liner routeing strategies", *Maritime Policy and Management* 23:3, 246-259 (1996).

[7] G. B. Dantzig and D. R. Fulkerson, "Minimizing the number of tankers to meet a fixed schedule", *Naval Research Logistics Quarterly* 1, 217-222 (1954).

[8] H.-S. Hwang, S. Visoldikokpun, and J Rosenberger, "A branch-and-price-and-cut methods for ship scheduling with limited risk", *Transportation Science* 42, 336-351 (2008).

[9] D. Jaramillo and A. Perakis, "Fleet deployment optimization for liner shipping. Part 2: Implementation and results", *Maritime Policy and Management* 18, 235-262 (1991).

[10] A. Mehrez, M.S. Hung, and B.H. Ahn, "An industrial ocean-cargo shipping problem", *Decision Sciences* 26:3, 395-423 (1995).

[11] A. Perakis and D. Jaramillo, "Fleet deployment optimization for liner shipping. Part 1: Background, problem formulation and solution approaches", *Maritime Policy and Management* 18, 183-200 (1991).

[12] B. Powell and A. Perakis, "Fleet deployment optimization for liner shipping: an integer programming model", *Maritime Policy and Management* 24, 183-192 (1997).

[13] R. Pesenti. "Hierarchical resource planning for shipping companies", *European Journal of Operational Research* 86, 91-102 (1995).

[14] X. Xinlian, W. Tengfei, and C. Daisong, "A dynamic model and algorithm for fleet planning", *Maritime Policy and Management* 27:1, 53-63 (2000).

[15] D. Bertsimas and M. Sim, "Robust discrete optimization and network flows", *Mathematical Programming, Ser B* 98. 49-71 (2003).

[16] D. Bertsimas and M. Sim, "The price of robustness", *Operations Research* 52:1, 35-53 (2004).

Multi-Vehicle One-to-One Pickup and Delivery Problem with Split Loads

Gizem Çavuşlar, **Güvenç Şahin**, **Mustafa Şahin**

Faculty of Engineering and Natural Sciences, Industrial Engineering Program

Sabancı University

Temel Öncan

Department of Industrial Engineering

Galatasaray University

Dilek Tüzün

Department of Systems Engineering

Yeditepe University, Kadıköy, Istanbul, Turkey

Email: dtuzun@yeditepe.edu.tr

1   Introduction

The multi-vehicle one-to-one pickup and delivery problem determines a set of least cost vehicle routes in order to satisfy a set of pickup and delivery requests between location pairs, subject to some side constraints. In the conventional vehicle routing and pickup and delivery problems, delivery request from a pickup location to its corresponding delivery location is satisfied with a single service by a single vehicle. In the Pick-up and Delivery Problem with Split Loads (PDPSL), which was first introduced by Nowak et al. [1], splitting loads is permitted, i.e. a delivery request may be satisfied with more than a single service and possibly by more than a single vehicle.

PDPSL is defined on a directed graph $G = (V, A)$ where V is the vertex set and A is the arc set. The vertex set is partitioned as $V = \{P, D, \{0, 2n + 1\}\}$. For a given set of n pickup-delivery pairs, $P = \{1, 2, \ldots, n\}$ is the set of pickup vertices and $D = \{n + 1, n + 2, \ldots, 2n\}$ is the set of delivery vertices where i and $n + i$ represent a pickup-delivery pair. $\{0, 2n + 1\}$ includes the two copies of the depot location representing respectively the starting and ending locations of the vehicle routes. The arc set is defined as $A = \{(i, j) : i = 0, j \in P\} \cup \{(i, j) : i, j \in P \cup D, i \neq j, i \neq n + j\} \cup \{(i, j) : i \in D, j = 2n + 1\}$. $K = \{1, 2, \ldots, m\}$ denote the set of available vehicles. For each vertex $i \in V$, q_i denotes the pickup or delivery quantity where $q_i > 0$ for $i \in P$, $q_i = -q_{i-n}$

for $i \in D$, and $q_0 = q_{2n+1} = 0$. Each vehicle $k \in K$ has a capacity of Q_k. d_{ij} is the travel distance associated with arc $(i, j) \in A$, and D is the maximum travel distance of a vehicle route. We assume that any pickup-delivery load can be provided with a single service by a single vehicle, i.e. $q_i \leq \max_{k \in K}\{Q_k\}$.

Nowak et al. [1] study the single vehicle variant of the PDPSL, where all loads are serviced on a single route. They develop a tabu search heuristic that uses ideas from the classical savings algorithm Clark and Wright [2] with common local search methods to solve the problem. The heuristic is tested using a new set of random problem instances of varying characteristics. Their results indicate that the savings achieved by load splitting is highest when the load sizes are just over half of the vehicle capacity. In Nowak et al. [3], they empirically investigate the effect of problem characteristics on the magnitude of benefit obtained by splitting loads among multiple trips. The problem characteristics considered in the study include mean load size and variance, number of origins relative to the number of destinations, the percentage of origin-destination pairs with a load requiring service, and the clustering of origin and destination locations. To the best of our knowledge, there have not been any other studies in the literature that address the PDPSL. However, there is a large body of literature on the classical PDP. One may refer to the extensive surveys conducted recently on the problem by Cordeau et al. [4] and Parragh et al. [5] for classification of pick-up and delivery problems and a review of the exact and heuristic solution approaches in each problem category. Due to the complexity of the problem, heuristic and metaheuristic approaches dominate the PDP literature. Although the issue of split loads has not received much attention in the PDP literature, a significant amount of literature exists on the vehicle routing problem with split loads (SDVRP). For an overview of the theoretical results on the SDVRP and review both exact and heuristic approaches to the problem, we refer the reader to a recent survey by Archetti and Speranza [6].

2   Tabu Search Algorithm for the PDPSL

As stated in Nowak et al. [1], PDPSL is an NP-Hard problem; therefore, solving large instances optimally is often intractable. We propose a tabu search heuristic to solve the PDPSL. In [1], they propose a tabu search algorithm for the single vehicle version of PDPSL and also test the algorithm on multiple vehicle problems. Our algorithm is designed for the multiple vehicle case and uses a different strategy that creates moves with and without split simultaneously. The algorithm also makes use of an optimality condition of PDPSL to improve the resulting routes. Below, we first describe the savings heuristic that is used to initialize the algorithm and then introduce the details of the tabu search procedure.

We adapt the well-known savings algorithm due to Clark and Wright [2] to obtain an initial

solution to the PDPSL. In parallel with the original algorithm, each pickup-delivery pair is initially served by a separate vehicle route, i.e. each route is of the form $(0, i, i + n, 0)$. A savings value is then calculated for every pair of pickup point i and delivery point j such that $j \neq i + n$. The saving value is the difference in total distance travelled when the routes that serve the two points are combined into a single route. The pairs are then sorted in non-increasing order of their savings. Starting from the first pair on the list, the algorithm merges the routes associated with the pairs with positive savings while ensuring the feasibility of the resulting routes. Finally, an improvement step is carried out where the pickup point i and delivery point j are moved forward and backward respectively in the merged route if such a move results in further savings.

Starting with the initial solution from the savings algorithm which contains no split loads, the tabu search algorithm searches for better solutions using moves that create split pickups and deliveries. The algorithm searches for an improvement in route length by considering split and insert moves for a (possibly partial) pickup-delivery pair. At each step of the tabu search algorithm, all pickup-delivery pairs are considered for insertion at all possible positions of the existing routes. The cost and the capacity of each option is calculated. Insertion cost is the difference in the total route length due to the insertion. The insertion capacity is the maximum amount of load that can be inserted, which is dependent on the residual capacity between the new pickup and delivery positions. Based on the amount of insertion capacity, we can perform two moves:

- *Insert*: If the insertion capacity is sufficient to handle the entire load, then an insert move is feasible for the pickup-delivery pair.

- *Split*: If the insertion capacity is not sufficient, the algorithm searches for feasible two-way splits, where the load is divided between two pickup-delivery positions.

Among all feasible insert and split moves, the move that results in the largest reduction in route length is applied and future moves that yield the same route length, number of stops, and number of routes combination are declared tabu for a number of iterations. In performing a split move, we may use different splitting strategies to allocate the load among the two the pickup-delivery locations. For instance, the smaller capacity location is filled up to capacity and the remaining load is assigned to the other pickup-delivery location pair. Implementing either an insert or a split move, the algorithm checks whether any of the following optimality conditions are violated:

Condition 1. In a PDPSL where the distance matrix d_{ij} satisfies the triangular inequality, an optimal route does not contain multiple deliveries of a load without a pickup of the same load in between.

Condition 2. In a PDPSL where the distance matrix d_{ij} satisfies the triangular inequality, an optimal route does not contain multiple pickups of a load without a delivery of the same load in between.

In case of *Condition 1*, the route can be improved by making all deliveries at the first delivery point. If *Condition 2* holds, then the route can be improved by making all pickups at the last pickup point. These merge operations are carried out before we proceed to the next iteration. The algorithm continues until a predetermined number of iterations have been performed without an improvement in the objective function value.

3 Results

The tabu search algorithm is tested on the data sets given in Nowak et al. [1] and Ropke and Pisinger [7]. Based on the preliminary results, the algorithm is effective in providing good solutions in reasonable computational time. In our experimental study, we explore the impact of different splitting strategies (of the split move) on the algorithm performance. We also perform further experimentation with the tabu search parameters.

References

[1] M. Nowak, Ö. Ergun and C. C. White III, "Pickup and Delivery with Split Loads", *Transportation Science* 42, 32-43 (2008).

[2] G. Clarke and J. W. Wright, "Scheduling of Vehicles from a Central Depot to a Number of Delivery Points", *Operations Research* 12, 568-581, (1964)

[3] M. Nowak, Ö. Ergun and C. C. White III, "An Empirical Study on the Benefit of Split Loads with the Pickup and Delivery Problem", *European Journal of Operational Research* 198, 734-740 (2009).

[4] J. F. Cordeau, G. Laporte, S. Ropke, "Recent Models and Algorithms for One-to-One Pickup and Delivery Problems", in *The Vehicle Routing Problem: Latest Advances and Challenges*, B.L. Golden, S. Raghavan, E. Wasil (eds), Springer, New York, NY, USA, 2008.

[5] S. N. Parragh, K. F. Doerner, R. F. Hartl, "A Survey on Pickup and Delivery Problems Part II: Transportation Between Pickup and Delivery Locations", *Journal für Betriebswirtschaft* 58, 81117 (2008).

[6] C. Archetti and M. G. Speranza, "The Split Delivery Vehicle Routing Problem: A Survey", in *The Vehicle Routing Problem: Latest Advances and Challenges*, B.L. Golden, S. Raghavan, E. Wasil (eds), Springer, New York, NY, USA, 2008.

[7] S. Ropke and D. Pisinger, "An Adaptive Large Neighborhood Search Heuristic for the Pickup and Delivery Problem with Time Windows", *Transportation Science* 40, 455-472 (2006).

A 0-1 Mixed-Integer Linear Programming Model for the Distribution Planning of Bulk Lubricants

M. Furkan Uzar

Faculty of Engineering and Natural Sciences

Sabanci University

Bülent Çatay

Faculty of Engineering and Natural Sciences

Sabanci University, Istanbul, 34956, Turkey

Email: catay@sabanciuniv.edu

1 Introduction

We address the distribution planning problem of bulk lubricants at BP Turkey. With its specific characteristics and elements of the distribution system the problem differs from many of the transportation problems addressed in the literature. Although the oil industry has been a major source of applications, white papers and reports on those applications and the academic research in the field are rather scant [1]. [2] addresses the transportation problem of gasoline from a single bulk terminal to customers. They design and implement a centralized dispatching system where the objective is to minimize the transportation costs while maintaining equitable man and equipment workload, safety standards, and customer service. [3] extends this work by considering multiple sources. [4] develops a rule-based decision support system for a regional oil company. The algorithm finds the schedule of the drivers and the dispatching of the tank trucks for a single day and is implemented as a semi-automated system. [5] considers the distribution problem of bulk and packaged lube oil in Mobil Oil Corporation using a heterogeneous fleet. [6] proposes a variable neighborhood search heuristic to dispatch the tank trucks with multiple compartments in the delivery of fuel. Vehicles with multiple compartments are also used in the transportation of food and grocery items ([7], [8], and [9]).

Our study considers the distribution of bulk lubes from a lube production plant to industrial customers. In our problem, the fleet is heterogeneous and consists of multi-compartment vehicles, i.e., tank trucks, where each compartment can only be assigned to a single product. The objective is to find a minimum cost transportation plan. The problem basically consists of loading the customer orders to tank trucks and determining the routes of the assigned tank trucks.

2 Problem Description

The problem is a multi-product, multi-period, heterogeneous fleet management problem that involves the assignment of customer orders to tank trucks and routing of tank trucks. The elements of the distribution system can be classified into four categories: (i) the fleet, which consists of multi-compartment tank trucks; (ii) the distribution network, which includes the plant where the trucks are loaded and the cities where the customers are located; (iii) the products with their specific properties; and (iv) the scheduling system, which has different constraints and flexibilities. In what follows, we provide further details on these elements of the problem and then formulate the mathematical model.

The tank trucks have 4 or 5 compartments (tanks) with different capacities. The company does not have its own fleet and uses a third party logistics (3PL) contractor for the distribution of the lubes. It estimates the fleet type and size and makes an annual contract with the 3PL contractor based on a pre-determined fleet dedicated to its delivery services. In the case the capacity is insufficient in any day the company hires trucks from the spot market at an additional cost. Hence, the truck capacity can be considered as a loose constraint in that sense. The trucks have different load restrictions and tank capacities, which makes the problem a heterogeneous fleet type distribution problem. In addition, the trucks in the fleet are classified as big- and small-sized trucks. Small trucks are used to serve the customers whose unloading area is not large enough to accommodate the big-size trucks.

The distribution network consists of one plant in Bursa (northwest region) and 180 customers dispersed in 28 cities located in different regions of Turkey. Trucks are loaded at the plant according to the planned deliveries and visit the customers using a route such that the total distance until the last customer on the route is the minimum. The routing is only made for the city-to-city network and the distances between the customers located in the same city are not accounted for because the company is charged for long distance trips on a kilometer basis and pays a fixed cost for each additional customer served in the same city. At the end of its trip, the truck returns to the plant. The company does not pay for the return trip of the truck to the depot, which makes the problem an open vehicle routing problem.

The company produces and distributes 130 different products in total. There are 8 basic product families and each product family consists of product groups. Since the products are in liquid form two different products cannot be loaded within the same tank. In addition, the tank may require a cleaning operation depending on the type of lube oil last loaded in the tank. The cleaning is not product-dependent and its time (cost) is same for all product groups.

The orders are received on a daily basis and assigned with an estimated delivery date. However, the planned delivery date is finalized after an advanced payment from the customer has been confirmed. The company has flexibility in determining the delivery date for consolidation purposes. For instance, an order can be delivered 2 days before or after its planned delivery date. In this study, we refer to the latest day that the demand must be delivered as the due date of the order. That is, a demand with due date 5 can be satisfied in any of the days 1, 2, 3, 4 or 5. Therefore, the distribution problem is a multi-period problem which is solved on a rolling horizon basis.

3 Solution Approach and Results

The problem is formulated as a 0-1 mixed integer linear programming model. Since the model is intractable for real-life industrial environment we propose two heuristic approaches to solve it efficiently. The first approach is a linear programming (LP) relaxation-based heuristic (LPH) while the second is a threshold accepting heuristic (TAH). We propose two variants of the latter heuristic: the first (TAH1) uses the distance priority whereas the second (TAH2) has a due date priority. Since the distribution plan is made daily and the plan of the following day is implemented the proposed algorithms are also designed to finalize the delivery schedule of the next day by iteratively solving them every day.

LP relaxation basically relaxes the binary variables by allowing them to take values between 0 and 1. The proposed LPH utilizes the LP relaxation with some rounding techniques and tries to find a good feasible solution for the original problem. The aim is to satisfy the demands of the first day and then to assign the remaining orders to the available tanks of the partially loaded trucks to efficiently utilize their capacities. Once the demands are assigned to tank trucks, the route can be easily determined since the problem reduces to finding a Hamiltonian path originating from the plant. We have observed that the nearest neighbor algorithm (NN) is usually able to find the optimal routes because the cities to be visited are lined up in one direction and it is rarely the case that a tank truck visits 3 cities or more. Hence, we implemented NN in the routing phase of the algorithm.

TAH1 aims at assigning the demands of small customers first. It starts with the customer farthest to the plant and having a due date 1 and continues with the remaining small customers with other due dates. A threshold parameter is used for controlling the insertion of a new customer demand into an existing tour. When all small customers have been served the algorithm assigns the demands of the large customers in the same way. Similar to TAH1, TAH2 assigns the demands of the small customers first and satisfies the demands of large customers next; however, the priority is given to customers having a due date 1. Once the loads are determined, the routes are obtained using NN.

Table 1. Weekly cost figures in preliminary experiments

Week	TAH1	TAH2	LPH	CPLEX
1	31044	27222	36212	37177
2	22360	25798	26364	34951
3	25310	21174	20887	38807

The proposed algorithms were tested on two different real data sets. The numerical results of the preliminary experiments on a monthly data have shown that threshold-accepting heuristics are very efficient in terms of both the computational time and the solution quality whereas the LPH is time inefficient with inferior solution quality. The weekly costs are summarized in Table 1. CPLEX upper bounds are obtained by setting the global time limit to 2000 seconds. In the cases when CPLEX fails to find a feasible solution within this time limit it is extended to 3000 seconds. Note that these figures are in "Monetary Units (MU)" that are kept fictitious for confidentiality reasons.

Table 2. Monthly cost figures in the extended experiments

Month	TAH1	TAH2	Current System
1	51570	53335	57618
2	62022	66003	62899
3	51270	56700	61047

To further investigate the performances of TAH1 and TAH2 and compare them with the current system in practice, we perform an extended computational study on quarterly data. To better evaluate both algorithms fairly, we freeze the time horizon at the end of 13th week, i.e. the demands due thereafter are not considered. The monthly costs are summarized in Table 2. We observe that TAH1 and TAH2 outperform the current system by 10.1% and 3.1%, respectively. Furthermore, the performance of TAH1 is 6.8% better than that of TAH2. These results are promising in the sense that both of the proposed TAHs are capable of improving the current distribution costs of the company.

References

[1] D. Ronen, "Dispatching Petroleum Products", *Operations Research* 43, 379–387 (1995).

[2] G. Brown and G. Graves, "Real-time Dispatch of Petroleum Tank Trucks", *Management Science*. 27, 19-32 (1981).

[3] G. Brown, C. Ellis, G. Graves and D. Ronen, "Real-time Wide Area Dispatch of Mobil Tank Trucks", *Interfaces*. 17, 107-120 (1987).

[4] L.S. Franz and J. Woodmanse, "Computer-aided Truck Dispatching Under Conditions of Product Price Variance with Limited Supply", *Journal of Business and Logistics* 11, 127-139 (1990).

[5] D.O. Bausch, G. Brown and D. Ronen, "Consolidating and Dispatching Truck Shipments of Mobil Heavy Petroleum Products", *Interfaces*. 25, 1-17 (1995).

[6] F. Ben Abdelaziz, C. Roucairol and C. Bacha, "Deliveries of Liquid Fuels to SNDP Gas Stations Using Vehicles with Multiple Compartments", in *IEEE International Conference on Systems, Man and Cybernetics*, 478-483, 2002.

[7] E.D. Chajakis and M. Guignard, "Scheduling Deliveries in Vehicles with Multiple Compartments", *Journal of Global Optimization* 26, 43–78 (2003).

[8] B. Bilgen and I. Ozkarahan, "A Mixed-Integer Linear Programming Model for Bulk Grain Blending and Shipping", *International Journal of Production Economics* 107, 555-571 (2007).

[9] A.E. Fallahi, C. Prins and R. Calvo, "A Memetic Algorithm and a Tabu Search for the Multi-Compartment Vehicle Routing Problem", *Computers and Operations Research* .35, 1725-1741 (2008).

Recursive column generation for the Tactical Berth Allocation Problem

Ilaria Vacca   **Matteo Salani**   **Michel Bierlaire**

Transport and Mobility Laboratory

Ecole Polytechnique Fédérale de Lausanne, Switzerland

Email: ilaria.vacca@epfl.ch

1   Introduction

Container terminal operations have received increasing interest in the scientific literature over the last years and operations research techniques are nowadays used to improve terminal's efficiency and productivity. A promising research trend is represented by the simultaneous optimization of decision problems that are usually solved hierarchically by terminal's planners. In particular, the integration of the berth allocation problem and the quay crane assignment problem has been recently tackled from several angles, as reported in [1].

The Tactical Berth Allocation Problem (TBAP), introduced by [2], aims to schedule incoming ships over a time horizon, assigning them a berthing position and a certain *quay crane profile* (i.e., number of quay cranes per working shift). These decisions are strictly correlated, since the number of quay cranes assigned to a ship affects its expected handling time, and thus has impact on the scheduling in the berth allocation plan. Furthermore, housekeeping costs generated by the berth assignement are taken into account by a quadratic term in the objective function.

In this work, we tackle the computational complexity of TBAP by exploiting its structure, developing ad-hoc optimization techniques. In particular, we present a reformulation based on Dantzig-Wolfe decomposition and an exact solution approach based on column generation. A new framework, called two-stage column generation, is illustrated and discussed.

2   Column generation for TBAP

Compact formulation   In the following we refer to the mixed integer formulation for TBAP introduced by [2]. Let N be the set of vessels and M the set of berths. Given $n = |N|$ ships with time windows on the arrival time at the terminal, $m = |M|$ berths with time windows on availability,

a planning time horizon discretized in $|H|$ time steps, a set P_i of feasible QC assignment profiles defined for every ship $i \in N$ with associated value, and the maximum number of quay cranes available in the terminal Q, the objective of TBAP is to find a feasible assignment of ships to berths, a feasible scheduling of ships in every berth and to assign a quay crane profile to every ship, in order to maximize the total value of selected profiles as well as minimize the housekeeping costs generated by the berth assignment. In particular, the objective function was defined as:

$$\max \sum_{i \in N} \sum_{p \in P_i} v_i^p x_i^p - \frac{1}{2} \sum_{i \in N} \sum_{j \in N} \sum_{k \in M} \sum_{w \in M} y_i^k y_j^w f_{ij} d_{kw} \tag{1}$$

where v_i^p is the monetary value assigned to profile p, f_{ij} is the number of containers exchanged between ships i and j, d_{kw} is the housekeeping cost to transfer one container from the yard slot corresponding to berth k to the yard slot corresponding to berth w. The involved decision variables are x_i^p, binary, equal to 1 if profile p is assigned to ship i, 0 otherwise and y_i^k, binary, equal to 1 if ship i is assigned to berth k, 0 otherwise. The authors further linearize the quadratic term in the objective function by introducing decision variables z_{ij}^{kw}, binary, equal to 1 if ship i is assigned to berth k and ship j is assigned to berth w, 0 otherwise.

Extensive formulation Our reformulation is based on the concept of *berth sequence*, a sequentially ordered subset of ships in a berth with an assigned quay crane profile. Let Ω^k be the set of all feasible sequences of berth $k \in M$. For every $r \in \Omega^k$, the binary decision variable λ_r is equal to 1 if sequence r is chosen, 0 otherwise. We reformulate TBAP via Dantzig-Wolfe, obtaining the so called *extensive formulation*:

$$\max \sum_{k \in M} \sum_{r \in \Omega^k} v_r \lambda_r - \frac{1}{2} \sum_{i \in N} \sum_{j \in N} \sum_{k \in M} \sum_{w \in M} z_{ij}^{kw} f_{ij} d_{kw} \tag{2}$$

$$\text{s.t.} \sum_{k \in M} \sum_{r \in \Omega^k} \alpha_r^i \lambda_r = 1 \qquad \forall i \in N, \tag{3}$$

$$\sum_{k \in M} \sum_{r \in \Omega^k} q_r^h \lambda_r \le Q^h \qquad \forall h \in H, \tag{4}$$

$$\sum_{r \in \Omega^k} \lambda_r \le 1 \qquad \forall k \in M, \tag{5}$$

$$\sum_{k \in M} \sum_{w \in M} z_{ij}^{kw} = g_{ij} \qquad \forall i \in N, j \in N, \tag{6}$$

$$z_{ij}^{kw} \le \sum_{r \in \Omega^k} a_r^i \lambda_r \qquad \forall i \in N, j \in N, k \in M, w \in M, \tag{7}$$

$$z_{ij}^{kw} \le \sum_{r \in \Omega^k} a_r^j \lambda_r \qquad \forall i \in N, j \in N, k \in M, w \in M, \tag{8}$$

$$z_{ij}^{kw} \in \{0,1\} \qquad \forall i \in N, j \in N, k \in M, w \in M, \tag{9}$$

$$\lambda_r \in \{0,1\} \qquad \forall r \in \Omega^k, k \in M, \tag{10}$$

where α_r^i is a binary parameter equal to 1 if ship i in assigned to sequence r and 0 otherwise, q_r^h is the number of quay cranes used by sequence r at time step h, Q^h is the number of quay cranes available at time step h and g_{ij} is a binary parameter equal to 1 if $f_{ij} > 0$ and 0 otherwise. The total value v_r of a sequence r is obtained by summing up the values of the quay crane profiles used by the ships in the sequence, i.e., $v_r = \sum_{i \in N} \sum_{p \in P_i} \beta_r^{ip} v_i^p$, where β_r^{ip} is a binary parameter equal to 1 if ship i is in sequence r and is assigned profile p, 0 otherwise.

The objective function (2) maximizes the total value of sequences, i.e., the total value of selected profiles, while minimizing the total housekeeping cost generated by the berth allocation plan. Constraints (3) state that every ship is assigned to exactly one sequence, and thus to one berth, while constraints (4) ensures that the quay crane capacity is not violated. Constraints (5) selects at most one sequence for each berth, while constraints (6)–(9) are due to the linearization of the objective function and link z_{ij}^{kw} variables to decision variables λ_r, constrained to be binary by constraints (10).

Master problem and column generation Let Ω^k be the set of all feasible berth sequences for berth k. We apply standard column generation to the restricted master problem defined by (2)–(8) over a subset $\Omega' \subset \Omega$, where $\Omega = \bigcup_{k \in M} \Omega^k$.

Let $[\pi, \mu, \pi_0, \theta, \eta]$ be an optimal dual solution to a restricted master problem, where π, μ, π_0, θ and η are the dual vector associated to constraints (3), (4), (5), (7) and (8), respectively. The reduced cost of sequence $r \in \Omega^k$ is given by:

$$\tilde{v}_r = v_r - \sum_{i \in N} \pi^i \alpha_r^i - \sum_{h \in H} \mu^h q_r^h - \pi_0^k + \sum_{i \in N} \sum_{j \in N} \sum_{w \in M} \theta_{ij}^{kw} a_r^i + \sum_{i \in N} \sum_{j \in N} \sum_{w \in M} \eta_{ij}^{kw} a_r^j. \tag{11}$$

At each iteration of column generation, we solve m subproblems, one for every berth $k \in M$. In particular, the *pricing subproblem* identifies, for every Ω^k, the column r_k^* with the maximum reduced cost. If $\tilde{v}_{r_k^*} > 0$ for some k, we add column r_k^* to the formulation and we iterate the process, otherwise we stop, since the current solution of the master problem is proven to be optimal.

The pricing subproblem is solved as an Elementary Shortest Path Problem with Resource Constraints (ESPPPRC). The underlying network $G(\tilde{N}, A)$ has one node for every ship $i \in N$, for every profile $p \in P_i$ and for every time step $h \in H$, and transit time on arcs equal to the length of profile p assigned to customer i. The size of this network grows polynomially with the number of vessels, time steps and quay crane profiles.

We implemented dynamic programming with standard accelerating techniques, such as decremental state space relaxation, to solve the pricing problem as well as dual space stabilization.

Two-stage column generation In order to tackle the complexity of TBAP and exploit its structure, we propose a new framework, called *two-stage column generation*. The basic idea is to apply column generation recursively, in two stages. We start considering only a subset $P_i' \subset P_i$

of quay crane profiles for every ship $i \in N$. We reformulate the problem via Dantzig-Wolfe and we apply standard column generation. At the end of the process, we try to identify profitable profiles $p \in P_i \setminus P_i'$ to be added to the compact formulation, by estimating their potential contribution to the master problem, in the same spirit of standard column generation. In this sense the column generation process has two stages: firstly, berth sequences are generated considering a restricted subset of quay crane profiles (inner column generation); subsequently, promising profiles are identified and added to the compact formulation, if any (outer column generation).

From a computational point of view, by considering only a subset of profiles, we are able to significantly reduce the size of the ESPPRC underlying network, so that the pricing subproblem can be solved more efficiently. However, the estimation of the potential impact of compact formulation variables on the master problem represented the main challenge in our approach. We addressed this issue adapting the method proposed by [3] to our framework: an upper bound to the profiles' reduced cost is obtained using sequences' reduced costs. Analytically, the reduced cost $\tilde{v}_i^p$ of profile $p \in P_i$, $i \in N$, is computed as:

$$\tilde{v}_i^p = \max_{r \in \Omega_i^p} \tilde{v}_r \tag{12}$$

where Ω_i^p represents the set of all sequences $r \in \Omega$ where ship i is visited and assigned profile p.

Computational tests, performed on the same set of instances of [2], compare our two-stage column generation approach to standard column generation, in the context of a branch-and-price algorithm. Results will be presented and discussed, in order to outline current issues and future research tracks.

References

[1] C. Bierwirth and F. Meisel,"A survey of berth allocation and quay crane scheduling problems in container terminals", *European Journal of Operational Research*, 202(3):615-627 (2010).

[2] G. Giallombardo, L. Moccia, M. Salani and I. Vacca, "Modeling and solving the Tactical Berth Allocation Problem", *Transportation Research Part B: Methodological*, 44(2):232-245 (2010).

[3] S. Irnich, G. Desaulniers, J. Desrosiers and A. Hadjar, "Path-Reduced Costs for Eliminating Arcs in Routing and Scheduling", *INFORMS Journal of Computing*, Articles in Advance pp. 1-17 (2009).

Online estimation of Kalman Filter parameters for traffic state estimation

Ir. C.P.IJ. van Hinsbergen [1,2] (corresponding author)

Email: c.p.i.j.vanhinsbergen@tudelft.nl

Dipl. Inform. T. Schreiter [1,2]

Dr. ir. J.W.C. van Lint [1]

Prof. dr. ir. S.P. Hoogendoorn [1]

Prof. dr. H.J. van Zuylen [1,3]

[1] Department of Transport & Planning, Faculty of Civil Engineering and Geosciences, Delft University of Technology

[2] TRAIL Research School

[3] Hunan University, College Civil Engineering

1 Introduction

Online traffic simulation models have widely been applied in advanced traffic information systems (ATIS) or dynamic traffic management (DTM) in recent years [1, 2, 3]. Online models use real-time traffic data to make an accurate estimate of the current state of traffic (in terms of density, flow or speed). The Extended Kalman Filter (EKF) has been applied successfully to optimally combine data with a traffic simulation model [3, 4]. However, one of the great difficulties in applying the EKF is that the parameters describing the measurement and process noise distributions (the covariance matrices) are seldomly known or even observable, and that assumptions on these noise parameters are usually made on the basis of trial and error, rather than theory or empirical evidence. The main contribution of this paper is a consistent methodology to continuously adapt the EKF parameters in an online traffic state estimation system. In a case study it is shown that these continuously adapted parameters lead to good state estimates, independent of their initial values.

2 Methodology: Bayesian estimation of the noise parameters

Define the state-space equation that describes the traffic state vector $\mathbf{x}_k$ of size N x 1 as a function of the previous state $\mathbf{x}_{k-1}$ and a noise vector $\mathbf{w}_k$:

$$\mathbf{x}_k = f\left(\mathbf{x}_{k-1}\right) + \mathbf{w}_k \tag{1}$$

In this paper f equals the *first order traffic model* [5, 6], solved by the Godunov scheme as the numerical solution [7], all implemented in the JDSMART software package [8]. The state vector $\mathbf{x}_k$ in

this case consists of the *densities* in all cells at time *k*. A measurement equation describing the measurement vector $\mathbf{z}_k$ of size *M* x 1 as a function of $\mathbf{x}_k$ with measurement noise $\mathbf{v}_k$ is given by

$$\mathbf{z}_k = h(\mathbf{x}_k) + \mathbf{v}_k \tag{2}$$

where $\mathbf{z}_k$ consists of all measurements, which are related to densities through the fundamental diagram described by the function *h*. The process noise $\mathbf{w}_k$ and measurement noise $\mathbf{v}_k$ are assumed zero mean white Gaussian noise with covariance matrix $\mathbf{Q}_k$ and $\mathbf{R}_k$ respectively and are assumed to be independent. Each element of the state vector is assumed to be drawn from a single distribution with variance $1/\alpha_k$ such that $\mathbf{Q}_k = 1/\alpha_k \cdot \mathbf{I}_N$, in which $\mathbf{I}_N$ is the identity matrix of size *N*x*N*, and each measurement is assumed to be drawn from a distribution with variance $1/\beta_k$ such that $\mathbf{R}_k = 1/\beta_k \cdot \mathbf{I}_M$ in which $\mathbf{I}_M$ the identity matrix of size *M*x*M*.

From a Bayesian perspective, the EKF provides a recursive methodology for finding the Maximum A Posteriori (MAP) value of $\mathbf{x}_k$, i.e. the EKF maximizes $P(\mathbf{x}_k|\mathbf{Z}_k)$ [9, 10], where $\mathbf{Z}_k = (\mathbf{z}_1, \mathbf{z}_2, ..., \mathbf{z}_k)$ is the set of all data vectors observed so far. In this paper it is shown that values for the Kalman filter parameters α_k and β_k can be also chosen using the same Bayesian inference approach. First consider the fact that only the *relative* magnitude of α_k and β_k matter in the EKF as the Kalman filter uses the two noise terms to balance the model predictions with data. Therefore, in this study it is chosen only to adapt β_k during simulation; α_k is held constant. To obtain the MAP estimate for β_k, Bayes rule is applied:

$$P(\beta_k \mid \mathbf{z}_k, \mathbf{Z}_{k-1}) = \frac{P(\mathbf{z}_k \mid \mathbf{Z}_{k-1}, \beta_k) P(\beta_k \mid \mathbf{Z}_{k-1})}{P(\mathbf{z}_k \mid \mathbf{Z}_{k-1})} \tag{3}$$

The prior $P(\beta_k|\mathbf{Z}_{k-1})$ is chosen *non-informative*, to represent the fact that there is normally very little knowledge of suitable values for the EKF parameters. Furthermore, it can be seen that the denominator of (3) is independent of β_k. Thus, when maximizing the posterior $P(\beta_k|\mathbf{z}_k, \mathbf{Z}_{k-1})$, *only the likelihood term $P(\mathbf{z}_k|\mathbf{Z}_{k-1}, \beta_k)$ needs to be maximized*. It can be shown that, after some involved algebra, the MAP estimate for β_k, given the chosen constant value of α_k, equals:

$$\hat{\beta}_k = \frac{M}{\left(\mathbf{z}_k - h(\hat{\mathbf{x}}_k)\right)^T \left(\mathbf{z}_k - h(\hat{\mathbf{x}}_k)\right) + Tr\left(\mathbf{B}^{-1}\mathbf{H}^T\mathbf{H}\right)} \tag{4}$$

where $\hat{\mathbf{x}}_k$ is the state estimate that the EKF produces, *M* is the number of measurements, $\mathbf{z}_k$ and *h* were given in (2), *Tr* is the trace operator, and **B** is the Hessian $\mathbf{B} = \left(\hat{\mathbf{P}}_k^-\right)^{-1} + \beta_k \mathbf{H}_k^T \mathbf{H}_k$. $\mathbf{H}_k$ is the Jacobian $\nabla_\mathbf{x} h\big|_{\hat{\mathbf{x}}_k^-}$ with $\hat{\mathbf{x}}_k^-$ the state estimate prior to performing the EKF. Finally, $\hat{\mathbf{P}}_k^- = \mathbf{A}_{k-1}\hat{\mathbf{P}}_{k-1}\mathbf{A}_{k-1}^T + \mathbf{Q}_k$ with $\hat{\mathbf{P}}_{k-1}$ the estimate of the covariance matrix of the state at time *k*-1 and $\mathbf{A}_k$ the Jacobian $\nabla_\mathbf{x} f\big|_{\hat{\mathbf{x}}_k}$. Here it is chosen to use the MAP estimate in the next time step, so $\beta_{k+1} = \hat{\beta}_k$ to prevent iterative simulations.

Equation (4) provides a solution for the problem of finding appropriate values for β_k and thus for $\mathbf{Q}_k$ and $\mathbf{R}_k$ during simulation. It can be given a very intuitive interpretation as follows. The term ($\mathbf{z}_k - h(\hat{\mathbf{x}}_k)$) represents the difference between measurement and model *after* the state has been corrected. As the estimate $\hat{\mathbf{x}}_k$ is an optimal estimate in the case of a Kalman Filter, and the best estimate we can find in the case of an Extended Kalman Filter, this difference can be seen as an indicator of the noise in

the data. A larger value for $(\mathbf{z}_k - h(\hat{\mathbf{x}}_k))$ indicates that the data is more noisy, and that lower trust should be placed on the data; in those cases β_k and thus the state corrections will become smaller.

3 Experiment

To illustrate the impact of the Bayesian choice for the EKF parameters, a small-scale case study is performed. The traffic network as shown in Figure 1 is simulated with JDSMART with a time step of two seconds with link capacities as shown in Figure 1a. A total of 600 time steps are simulated, with four different demand levels at the two origins O_1 and O_2 and four different turn fractions at the node A. Each time step the speeds in all cells are stored as the ground truth. Then, the network is simulated again with the same demands and turn fraction, but with random changes applied to the capacities of the links as shown in Figure 1b; this represents the presence of *process noise*. The speeds at four different cells, indicated by the arrows in Figure 1a, are then used as measurements to correct the state in the altered network. Zero mean Gaussian noise is added to these measurements, representing *measurement noise*. The states in the 'noisy' network are then corrected using the EKF every five time steps. The resulting speeds in all cells are compared to the cell speeds in the original network.
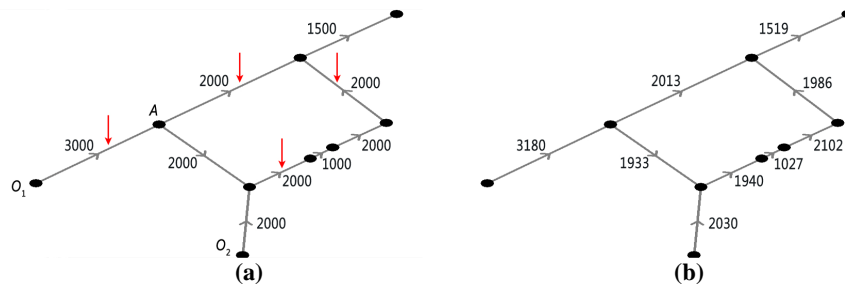
Figure 1 The ground truth network (a) and the network with 'process noise' (b). Numbers indicate the link capacities in veh/hr and arrows indicate measurement locations.

For all simulations $1/\alpha_k$ was set to 4 veh^2/km^2 $\forall\ k$, while the initial value $1/\beta_0$ was varied from 0.01 to 20 km^2/u^2, both with the Bayesian adaptation scheme as well as without. Figure 2 shows the resulting Mean Absolute Percentage Error (MAPE) that was calculated for all cell speeds for all time steps. It can be seen from Figure 2 that for constant $1/\beta_k$, the error shows a clear minimum. Left of the minimum the noise of the measurements is hardly filtered, while right of it the measurements are hardly used at all. In the case of the Bayesian choice for $1/\beta_k$ very little variation can be seen for different initial values $1/\beta_0$. Moreover, in this case the error for the Bayesian parameters it is nearly equal to the minimal possible error for constant parameters.
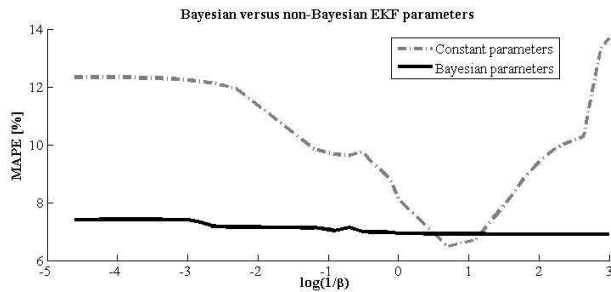
Figure 2 The errors for both constant β_k and continyyuously adapted Bayesian β_k.

4 Discussion and conclusions

This paper has proposed a method for setting values for the EKF parameters (measurement and process covariances) for online traffic state estimation, using a two-stage Bayesian inference framework. From a Bayesian perspective, the EKF is equal to Maximum A Posteriori (MAP) approach. The Bayesian MAP approach can also be applied to the Kalman filtering parameters, leading to a recursive method for choosing values for the measurement covariance matrix $\mathbf{R}_k$. As only the relative value of the two covariances matters in determining the Kalman gain, $\mathbf{Q}_k$ is held constant, leading to optimal choices for $\mathbf{R}_k$ for a given fixed value of $\mathbf{Q}_k$.

In the case study that is shown, the Bayesian choice for $\mathbf{R}_k$ leads to an error that is almost as low as the best possible constant settings, for any initial value $\mathbf{R}_0$. It can therefore be concluded that the Bayesian choice for the EKF parameters is very robust with respect to a high or low initial choice of the measurement noise covariance, whereas with constant EKF parameters the results are very sensitive to changes in the initial values. Future work will need to show if this holds for all cases, for example with different settings for the constant value of $\mathbf{Q}_k$, for larger networks and for real-world examples.

References

[1] Ben-Akiva, M., Bierlaire, M., Burton, D., Koutsopoulos, H. N. & Mishalani, R., "Network state estimation and prediction for real-time traffic management", *Networks and spatial economics,* 1**,** 293-318 (2001).

[2] Mahmassani, H. S., "Dynamic network traffic assignment and simulation methodology for advanced system management applications", *Networks and spatial economics,* 1**,** 267-292 (2001).

[3] Wang, Y. & Papageorgiou, M., "Real-time freeway state estimation based on extended Kalman filter: a general approach", *Transportation research B,* 39**,** 141-167 (2005).

[4] Tampère, C. M. J. & Immers, L. H., "An Extended Kalman Filter Application for Traffic State Estimation Using CTM with Implicit Mode Switching and Dynamic Parameters", 10th IEEE Intelligent Transportation Systems Conference, Seattle, Washington, USA, 2007.

[5] Lighthill, M. J. & Whitham, G. B., "On Kinematic waves (II): A theory of traffic flow on long crowded roads. ", *Proceedings of the Royal Society* A**,** 281-345 (1955).

[6] Richards, P. I., "Shockwaves on the highway", *Operations Research,* 4**,** 42-51 (1956).

[7] Lebacque, J. P., "The Godunov scheme and what it means for first order traffic flow models", Proceedings of the 13th international symposium of transportation and traffic theory, Lyon, 1996.

[8] Zuurbier, F. S., Hegyi, A. & Hoogendoorn, S. P., "A modeling framework for RG++: Route guidance as a traffic management measure", DTA Symposium, Leuven, Belgium, 2008.

[9] Ho, Y. C. & Lee, R. C. K., "A Bayesian approach to problems in stochastic estimation and control", *IEEE Transactions on Automatic Control,* 9**,** 333-339 (1964).

[10] Chen, Z., "Bayesian filtering: from Kalman filters to particle filters, and beyond", Hamilton, ON, Canada, Adaptive Syst. Lab., McMaster University, 2003.

Railway Crew Rescheduling with Retiming

Lucas P. Veelenturf

Department of Decision & Information Sciences and ECOPT

Rotterdam School of Management, Erasmus University Rotterdam

Burgemeester Oudlaan 50, 3062 PA Rotterdam, The Netherlands

Email: lveelenturf@rsm.nl, Phone: +31(0)10-4081567

Daniel Potthoff

Econometric Institute and ECOPT

Erasmus School of Economics, Erasmus University Rotterdam

Dennis Huisman

Econometric Institute and ECOPT

Erasmus School of Economics, Erasmus University Rotterdam

Department of Logistics, Netherlands Railways

Leo G. Kroon

Department of Decision & Information Sciences and ECOPT

Rotterdam School of Management, Erasmus University Rotterdam

Department of Logistics, Netherlands Railways

1 Introduction

Passenger railway operations often face unforeseen events like infrastructure malfunctions, accidents or rolling stock breakdowns. As a consequence, parts of the railway infrastructure may become temporarily unavailable. Therefore, it may not be possible to operate the timetable as planned. Within minutes or, even better, seconds, new schedules must be constructed which is called real-time planning. In [1] the disruption management process is described as the accomplishment of three interconnected steps: (*i*) Timetable adjustment, (*ii*) rolling stock rescheduling and (*iii*) crew rescheduling. Due to the complexity of the process and the limited time available for decision making, these steps are carried out sequentially in practice. First, an adjusted timetable is constructed by canceling, delaying or rerouting trains. Thereafter modified resource (rolling stock

and crew) schedules are constructed. If no matching resource schedule can be found, the timetable is adjusted again.

The iterations may lead to cancelations of trains in addition to the inevitable cancelations caused by the disruption, for example if no crew can be found to drive a certain train. Sometimes such additional cancelations could be avoided by delaying the departures of some trains by just a couple of minutes. It is quite clear that up to 1,000 passengers waiting for a train on a busy station during peak hours would prefer a delayed train over a canceled one. In this paper, we propose a crew rescheduling approach which may change the timetable as well.

Recently, Operations Research based models have been developed for real-time resource rescheduling in railways. For example, [2] deals with rolling stock rescheduling while [3] and [4] present models and solution approaches for railway crew rescheduling. However, these models do not allow to change the timetable. Constructing the timetable and resource schedules at the same time leads in general to better or equal solutions than the iterative procedure.

In this paper, we consider an extension of the crew rescheduling problem where timetabling decisions are integrated into crew rescheduling: The departure of trains may be delayed, which is called retiming. This gives additional flexibility to Step (*iii*) of the disruption management process and may avoid undesired iterating of that process. Moreover, this new approach may be able to provide high quality solutions from a service level point of view. The approach is tested on real-life train driver data instances of Netherlands Railways (NS). Therefore, the mathematical model and the solution approach are specified to train drivers.

In [5] a model is presented that does timetabling and crew rescheduling at the same time for relatively small problems. However, because of the limited computation time that is available and the high detail of the timetabling decisions, integration seems reaching too far for large scale problems at this point in time.

In airline crew rescheduling, [6], [7], [8] and [9] are papers that integrated timetable and crew rescheduling. Moreover, [9] also integrated aircraft rescheduling. Some of these approaches could be applicable in railways after significant modifications. However, we decided to work further on the approach of [4] since we do not know about the performance of those airline approaches on railway instances and [4] already has shown to perform well for railway crew rescheduling without retiming.

2 Mathematical model

We assume that the timetable and rolling stock schedule have been rescheduled already. In this new timetable some trains have been canceled and some new trains have been added to deal with the disruption. The original crew schedule has become infeasible for the new timetable. Drivers

who had to run a now canceled train have to perform other trains. Moreover, crew must be found for the new trains. The operator often has some stand-by crew available to deal with those canceled and new trains.

The *operational crew rescheduling problem with retiming* has as objective to find a driver for as many trains as possible: If no driver can be found, the train will be canceled. The problem is formulated as a set covering problem with side constraints for the retiming. We use multiple copies of trains to represent the retiming possibilities. The copies differ from each other in their departure and arrival times. Using copies of trains limits the retiming possibilities since the departure time cannot be chosen continuously and the retiming possibilities of a train must be determined beforehand. One set of the side constraints requires that for every train only one departure time is used in the crew schedule. The remaining side constraints assure that delays caused by retiming will be absorbed or propagated, depending on the dwell times of the trains.

3   Solution approach

Our aim is to provide solutions of good quality within a couple of minutes of computation time. Therefore, we use a heuristic approach in which not all drivers and trains of the original crew schedule are considered. We extract core problems containing only a subset of them.

We start with computing a solution for a core problem without retiming possibilities. If all trains are supplied with a crew member, we stop. Otherwise we iterate over the trains without a driver. For each such train a new small core problem is defined. The new core problems are constructed with a neighborhood definition and contain some retiming possibilities. After each iteration, the list of uncovered trains is updated. The core problems are solved by a Lagrangian heuristic embedded in a column generation scheme very similar to the one proposed by [4]. The approach of [4] has been extended such that it allows timetable changes.

4   Computational results and conclusions

We tested our approach with retiming on six real-life disruption scenarios of NS. In order to evaluate the benefits of retiming, we compare our method with the approach of [4].

The approach of [4] came up with solutions with up to 2 canceled trains. Our approach generated equal or better solutions. In 2 out of the 6 cases, our approach could not improve the solutions. However, by using our approach, in 3 cases one train less had to be canceled and once two trains less had to be canceled. Moreover, the observed delays that were introduced into the timetable are very small. This makes it likely that those solutions can indeed be implemented in practice. The computation times of our approach are less than 5 minutes, which should make it applicable within a decision support system for disruption management.

So far, we have limited ourselves to considering train drivers only. However, in a disrupted situation conductors need to be rescheduled as well. Our goal is to extend the approach with conductors.

Moreover, in our future work conflicts between trains due to retiming decisions will be taken into account as well. We believe that the presented model and solution approaches could be extended into that direction without sacrificing computation time too much.

References

[1] J. Jespersen-Groth, D. Potthoff, J. Clausen, D. Huisman, L.G. Kroon, G. Maróti and M. Ny-have Nielsen, "Disruption Management in Passenger Railway Transportation", Technical Report EI 2007-05, Econometric Institute, Erasmus University Rotterdam, 2007.

[2] L.K. Nielsen, "A Decision Support Framework for Rolling Stock Rescheduling", Technical Report ARRIVAL-TR-0158, ARRIVAL Project, 2008.

[3] N.J. Rezanova and D.M. Ryan, The train driver recovery problem – A set partitioning based model and solution method, *Computers & Operations Research* (2009).

[4] D. Potthoff, D. Huisman and G. Desaulniers, "Column generation with dynamic duty selection for railway crew rescheduling", Technical Report EI 2008-28, Econometric Institute, Erasmus University Rotterdam, 2008.

[5] C.G. Walker, J.N. Snowdon and D.M. Ryan, "Simultaneous disruption recovery of a train timetable and crew roster in real time", *Computers & Operations Research* 32, 2077-2049 (2005).

[6] M. Stojković and F. Soumis, "An Optimization Model for the Simultaneous Operational Flight and Pilot Scheduling Problem", *Management Science* 47, 1290-1350 (2001).

[7] A. Abdelghany, G. Ekollu, R. Narasimhan and K. Abdelghany, "A Proactive Crew Recovery Decision Support Tool for Commercial Airlines during Irregular Operations", *Annals of Operations Research* 127, 309-331 (2004).

[8] M. Stojković and F. Soumis, "The operational flight and multi-crew scheduling", *Yugoslav Journal of Operations Research* 15, 25-48 (2005).

[9] K.F. Abdelghany, A.F. Abdelghany and G. Ekollu, "An integrated decision support tool for airline schedule recovery during irregular operations", *European Journal of Operational Research* 185, 825-848 (2008).

GRASP/VND with Path Relinking for the Truck and Trailer Routing Problem

Juan G. Villegas[1,2], Christian Prins[1], Caroline Prodhon[1], Andrés L. Medaglia[2] and Nubia Velasco[2]

[1] Laboratoire d'Optimisation des Systèmes Industriels, Institut Charles Delaunay, Université de Technologie de Troyes BP 2060, 10010 Troyes Cedex, France

[2] Centro para la Optimización y Probabilidad Aplicada, Departamento de Ingeniería Industrial, Universidad de los Andes. A.A. 4976, Bogotá D.C., Colombia

Email: jg.villegas64@uniandes.edu.co, juan_guillermo.villegas@utt.fr

1. Problem definition and literature review

The Truck and Trailer Routing Problem (TTRP) is an extension of the well known vehicle routing problem. In the TTRP an heterogeneous fleet composed of m_t trucks and m_r trailers ($m_r < m_t$) is used to serve a set of customers $N = \{1, \ldots, n\}$ from a central depot, denoted with 0. Each customer $i \in N$ has a demand q_i; the capacities of the trucks and the trailers are Q_t and Q_r, respectively; and the distance c_{ij} between any two points $i, j \in N \cup \{0\}$ is known. The existence of accessibility constraints at some customers creates a partition of N into two subsets: the subset of *truck customers* N_t accessible only by truck; and the subset of *vehicle customers* N_v accessible either by truck or by a complete vehicle (i.e., a truck pulling a trailer). Due to the heterogeneity of the fleet and the accessibility constraints, a solution of the TTRP may have three types of routes: *pure truck routes* performed by a truck visiting customers in N_v and N_t; *pure vehicle routes* performed by a complete vehicle serving only customers in N_v; and finally *vehicle routes with subtours* performed by a complete vehicle. The latter type of route includes the case in which a trailer is detached at a vehicle customer in N_v to perform a subtour just with the truck visiting one or more customers in N_t (or even in N_v). The objective of the TTRP is to find a set of routes of minimum total distance such that: each customer is visited by a compatible vehicle exactly once; the total demand of the customers visited in a route or subtour does not exceed its capacity; and the number of required trucks and trailers is not greater than m_t and m_r, respectively.

The TTRP was introduced by Chao [4] and has been tackled using tabu search [4][9], simulated annealing [6], and a mathematical programming based heuristic [3]. Most of the methods ([3],[4],[9]) use a natural cluster-first, route-second approach. In this work, we show that a route-first, cluster-second procedure embedded within a hybrid metaheuristic based on Greedy Randomized

Adaptive Search Procedure (GRASP), Variable Neighbourhood Descent (VND) and Path Relinking (PR) is an effective approach to solve the TTRP. A description of the components of the hybrid metaheuristic follows.

2. Solution approach

GRASP is a two-phase iterative method: first, a feasible solution is built by a greedy randomized heuristic; second, the solution is improved by local search. As highlighted in [8], the performance of GRASP can be enhanced using multiple neighbourhoods and path relinking. Accordingly, we replaced the local search by an iterated VND, and used PR in various strategies.

Greedy Randomized Construction: originally proposed by Beasley [1] route-first, cluster-second methods provide a flexible and effective framework for the solution of arc and node routing problems [7]. Then, the greedy randomized construction of the proposed solution approach is performed by such a method. A giant tour $T = (0, t_1, \ldots, t_k, \ldots t_n)$ that visits all the customers in N is found using a randomized nearest neighbour heuristic with a restricted candidate list of size r. Then, a solution S of the TTRP is derived from T by means of a tour splitting procedure. The tour splitting procedure constructs one auxiliary acyclic graph $H = (X, U, W)$, where the set of nodes X contains a dummy node 0 and n nodes numbered 1 through n, and node k represents the customer in the $k - th$ position of T (i.e., t_k). The arc set U contains one arc $(k-1, l)$ if and only if the subsequence $(t_k, \ldots t_l)$ can be served by a feasible route. Finally, the weight of the arc $(k-1, l)$ in W is the total distance of the corresponding route. To derive S it is necessary to find the shortest path between 0 and n in H. The cost of the shortest path corresponds to the total distance of S and the arcs in the shortest path represent the routes of S.

To adapt the tour splitting procedure for the solution of the TTRP it is necessary to take into account the heterogeneous fixed fleet. Thus, to obtain S, a resource-constrained shortest path problem is solved, where the resources are the available trucks and trailers. Moreover, if the arc $(k-1, l)$ represents a vehicle route with subtours its cost is found with a dynamic programming method that solves a restricted version of the Single Truck and Trailer Routing Problem with Satellite Depots [10]. Some preliminary experiments have shown that it may be difficult to find feasible solutions with the tour splitting procedure in problems with a tight ratio between the total demand and the total capacity. Therefore, if the solution of the resource-constrained shortest path problem fails to find a feasible solution, an unfeasible "solution" is obtained solving an unrestricted shortest path problem.

Iterated Variable Neighbourhood Descent: The improvement phase of the proposed method is performed with an iterated VND [5]. One main loop of the iterated VND takes S as initial solution and performs three steps: (1) randomly exchange p pairs of customers from its giant tour T to obtain a new giant tour T'; (2) derive a new solution S' by applying the tour splitting procedure to T'; and (3) apply VND to S'. The latter VND step uses five neighbourhoods in the following order: Or-opt

(in single routes and subtours), node exchange, 2-opt, node relocation (in single routes/subtours and between pairs of routes/subtours), and finally, for each subtour it applies the root refining procedure of [4]. The exploration of each neighbourhood uses a best-improvement strategy, the iterated VND procedure repeats during ni iterations, and the value of p is controlled dynamically between 1 and p_{max}. Since unfeasible solutions are accepted as initial solutions and also during the search of VND, the incumbent solution of VND is replaced by S' if $f(S') < f(S)$ and its unfeasibility

$$\mu(S') = \max\left\{0, \frac{nt(S')}{m_t} - 1\right\} + \max\left\{0, \frac{nr(S')}{m_r} - 1\right\}$$ does not exceed a given limit τ, where $f(\cdot)$

denotes the objective function, and $nt(S')$ and $nr(S')$ the number of trucks and trailers used in S'. Every time a feasible solution is found, the best solution of the iterated VND is checked for an update. At each call of the iterated VND τ is initialized at τ_{max}, and updated at each iteration with

$$\tau = \tau - \tau_{max} / ni$$.

Path Relinking: GRASP with PR maintains a pool of elite solutions (ES). To be included in ES a solution S must be better than the worst solution of the pool; but to preserve its diversity, the distance between S and the pool ($d(ES, S)$) must be greater than a given threshold δ, where $d(ES, S) = \min_{S' \in ES} d(S', S)$, unless it is simply better than the best solution of ES. In this work the distance between any two solutions $d(S, S')$ is the distance for R-permutations [1] between their corresponding giant tours T and T'. The solutions in the pool are ordered according to a lexicographic comparator that gives priority to feasible solutions, among feasible solutions to those with smaller distances, and among unfeasible solutions to those with smaller unfeasibility. To transform the starting solution S_0 into the target solution S_f, the PR operator works in their giant tours, repairing from left to right the broken pairs of T_0 to create a path of giant tours with non-increasing distance to T_f. All the giant tours in the path are split and the resulting solutions are improved with VND, finally all the resulting solutions are tested for insertion in the pool. The PR operator uses the back and forward scheme [8], exploring the path from S_0 to S_f, and also the path from S_f to S_0. Due to the fact that GRASP and PR can be hybridized in different ways (see [8]), we tested PR: (i) as a post-optimization procedure; (ii) as an intensification mechanism; and (iii) in Evolutionary Path Relinking (EvPR).

3. Computational Results

The proposed method has been implemented in Java. All the variants were run for 60 GRASP iterations with $r = 2$, $ni = 200$, $p_{max} = 6$, and $\tau_{max} = 0.75$, for PR $|ES| = 5$ and $\delta = \max(10, m_t + m_r)$, and EvPR is run every 20 GRASP iterations. Table 1 shows the average

results over the 21 instances described in [4]. In summary, all GRASP/VND with PR variants outperform the previous competing methods.

Table 1. Results for the 21 test instances of the TTRP. (BKS: Best known solution)

Method	Avg. Dev. BKS	Avg. Time (min)	Computer
GRASP/VND with EvPR	0.72%	46.39	Pentium D 3.4 GHz
GRASP/VND with PR (Post-optimization)	0.95%	28.77	Pentium D 3.4 GHz
GRASP/VND with PR (Intensification)	0.98%	37.33	Pentium D 3.4 GHz
Simulated annealing [6]	1.47%	39.56	Pentium IV 1.5 GHz
Math. Programming Heuristic [3]	1.70%	Not Available	Pentium IV 2.8 GHz
Tabu search [9]	1.71%	47.32	Pentium IV 1.5 GHz
Tabu search [4]	7.51%	14.51	Pentium II 350 MHz

References

[1] J.E. Beasley, "Route-first cluster-second methods for vehicle routing". Omega 11, 403–408, 1983.

[2] V. Campos, M. Laguna and R. Martí, "Context-Independent Scatter and Tabu Search for Permutation Problems", *INFORMS Journal on Computing* 17, 111-122 (2005).

[3] M. Caramia and F. Guerriero, "A heuristic approach for the truck and trailer routing problem". *Journal of the Operational Research Society*. Advance online publication; doi:10.1057/jors.2009.59 (2009).

[4] I.-M. Chao, "A tabu search method for the truck and trailer routing problem", *Computers & Operations Research* 29, 33-51 (2002).

[5] P. Hansen and N. Mladenovic, "Variable neighbourhood search: Principles and applications", *European Journal of Operational Research* 130, 449-467 (2001).

[6] S.-W. Lin, V. F. Yu, and S.-Y. Chou, "Solving the truck and trailer routing problem based on a simulated annealing heuristic", *Computers & Operations Research* 36, 1683-1692 (2009).

[7] C. Prins, N. Labadi, and M. Reghioui, "Tour splitting algorithms for vehicle routing problems", *International Journal of Production Research* 47, 507-535(2009).

[8] M.G.C. Resende and C.C. Ribeiro, "Greedy Randomized Adaptive Search Procedures: advances and applications", in *Handbook of Metaheuristics, 2nd Ed.* M. Gendreau and J.-Y. Potvin (eds.), Springer, (To appear).

[9] S. Scheuerer, "A tabu search heuristic for the truck and trailer routing problem", *Computers & Operations Research* 33, 894-909 (2006).

[10] J.G. Villegas, C. Prins, C. Prodhon, A.L. Medaglia and N. Velasco. GRASP/VND and multi-start evolutionary local search for the single truck and trailer routing problem with satellite depots. *Engineering Applications of Artificial Intelligence,* Forthcoming (2010).

Improving the logistics of moving empty containers – Can new concepts avoid a collapse in container transportation?

Stefan Voß[1] and Robert Stahlbock[12]

[1] Institute of Information Systems, University of Hamburg,

Von-Melle-Park 5, 20146 Hamburg, Germany

e-mail: stefan.voss@uni-hamburg.de, stahlbock@econ.uni-hamburg.de

[2] Lecturer at FOM University of Applied Sciences, Essen/Hamburg, Germany

1 Introduction

Since the introduction of the maritime container in the mid 1950's, liner shipping groups have migrated from inefficient traditional cargo handling techniques to large cellular vessels seen at any of the world's major ports today. Container use improved intermodal productivity and allows for shorter point to point transit times. In addition, cargo damage is reduced. Besides an enormous benefit, shippers and carriers are faced with increased operational complexity as well as a multitude of variable and fixed costs. Managing these costs is important in particular in a situation with intense competition. Total seaborne trade has nearly quadrupled over the past four decades. During 2005, the world container population grew by 9.0% to reach 21.6 million TEU[1][2]. In 2007, global container trade was estimated at 143 million TEU, a 10.8% increase over 2006. In tonnage terms, container trade is estimated at 1.24 billion tons, accounting for about one quarter of total dry cargo loaded [8]. Carrier focus groups estimated in 2001 that approximately $ 17 billion is spent each year to deal with inefficiencies caused by repositioning of ECs[2] amongst others. Constant increase of the container population as well as increasing trade imbalance resulted in accumulation of ECs in some major port areas – and therefore in container shortage in other regions.

More sophisticated logistics may result in a reduction of EC movements. However, since 2000 the percentage of EC movements is stable around a level of 21%. The main reason for the ratio are the pronounced trade imbalances between Asia and Europe as well as North America. This

---

[1] TEU: twenty-foot equivalent unit; a standard size of a container, typically used for denoting the output or capacity of container terminals as well as for defining the container carrying capacity or loading of vessels.

[2] EC: empty container.

imbalance has persisted, and a declining trend to a level lower than 20% or even 19% that was evident prior to 1998 is unlikely to re-emerge.[3] There is no sign of a reversal of the imbalance in the Asian transatlantic and transpacific trades. On the contrary, the proportion of ECs is expected to be nearly 23% in 2015. The problem is influenced not only by trade imbalances, but by uncertainty (demands, handling, transportation), dynamic environment, and blind spots in the transport chain. Hence, repositioning ECs is a major problem and huge burden for ocean carriers now and in the future due to the difficulty of redeeming the incurred costs. Carriers are still aiming at achieving a lower level of the empty share of container movements and to improve the logistics with respect to ECs. The high level of the empty share cannot be significantly reduced by further improvements in managing equipment or by employing additional handling equipment.

This paper addresses the problem of EC management. We briefly describe the global chain of maritime container transport and related costs for different actors. Most importantly, we provide a review of literature regarding EC management and discuss some approaches including promising decision support systems for reducing EC transportation. Moreover, we discuss solution concepts to overcome parts of this situation which are not popular in practice (like, e.g., the use of foldable containers). Economical and technical conditions for promising usage are considered together with ideas for improving them. Ecological issues related to reverse logistics (e.g., scrap, waste paper/recycle) are taken into account and we propose the idea of pooling (i.e., interchange of containers on different scales, between shipping companies, owners etc.). Finally, we discuss entry and exit of containers with respect to the transport chain.

2 Literature Review

Scientific literature regarding the management of empty equipment is abundant but it is primarily focused on the optimisation of equipment transportation and on particular areas of the distribution cycle. The dynamic allocation, distribution and reuse of empty equipment for balancing demand and supply among terminals is extensively discussed, even in the context of network design problems. Furthermore, research on the effect of the planning horizon length is done as well as on finding an optimal amount of storage space in a yard. The EC management problem is also treated as an equilibrium inventory problem. Some studies analyze stakeholder operational activities. Almost all studies assume that the destination ports of ECs have to be determined before they are loaded onto vessels. One research gap until now seems to be a comprehensive investigation of a strategy using flexible destination ports.

According to the systems considered, the literature can be classified into studies focusing on

---

[3]The imbalance between the eastward and westward traffics seems to have levelled off in 2007, e.g., with the Asia-United States cargo flows exceeding those in the reverse direction by 10.5 million TEU, compared to 10.3 million in 2006 and 8 million TEU in 2005 [8].

deterministic systems and systems under uncertainty such as inland EC allocation or port-to-port container repositioning. With respect to adopted techniques, studies using mathematical programming can be distinguished from those applying parameterised control policies. The former focus on, e. g., the selection of an appropriate planning horizon, the latter investigate, e. g., characteristics of different empty repositioning policies.

Along these lines, before entering specific ideas for improvement and research issues, we conducted a very comprehensive literature survey which is classified as follows: (a) Allocation and distribution of empty equipment, (b) planning horizon, (c) dynamic equipment allocation and reuse problem, (d) EC balancing strategies within the context of a network design problem, and (e) problem of optimal amount of storage space in container terminals

3   Research Issues

The literature review shows that EC accumulation is well recognised in science and industry. Even if most papers either focus on isolated components of the problem and try to solve them separately or treat this problem as a side issue, focusing on separate aspects of the entire problem helps to gain insight and therefore supports the development of an approach considering all components and solution strategies for this integrated problem. In the sequel we list two out of quite a few research issues that we have put together and tried to develop further in this research.

Firstly, some papers discuss options for commercial application of foldable containers. It is shown that foldable containers can result in substantial net benefits in the total container transport chain due to their potential for cost savings. One of the crucial assumptions in the available studies refers to the additional costs caused by foldable containers. Obviously the costs of folding and unfolding are considered. Additional costs for transport to places where folding and unfolding can take place are mentioned, too. However, no extra location for (un)folding operations is necessary if foldable containers are designed in a way that they can be folded (or maybe even dismantled) on the spot. A basic idea is that the spreader of the crane folds the container automatically ('on the fly') when locating it in the vessel's body (or similar). This might be an interesting challenge for developers/engineers and a related cost calculation is provided assuming that this is possible.

Secondly, we consider the application of data mining techniques for gaining insight into processes and problems of container handling with a special focus on EC management. Container terminals perform similar functions, but the processes, technology, and labor requirements differ at each terminal. Therefore, terminals are faced with different bottlenecks. Due to the complexity of terminal operations it is difficult to identify bottlenecks within a process. The large amount of transaction data and the large number of potential, interdependent factors make an analysis challenging. However, having a large database is a prerequisite for using promising and well

established data mining methods, such as decision trees, neural networks, support vector machines, or association rule analysis in order to derive knowledge from data. Data mining methods and algorithms are useful for analysing current data reflecting current processes, identifying problems, interpreting solutions as well as for forecasting. For example, forecasting the movements of full and ECs on a terminal-wise, regional or even more aggregated level can be helpful for equipment planning purposes. Classifying time series of equipment demand or container flows by means of an ABC/XYZ analysis as well as taking calendar effects on a detailed regional level into account can provide useful knowledge for forecasting equipment demand. Decision tree based approaches can help to interpret data and identify causes for inefficiently performed processes (e.g., abnormal high truck dwell times within a terminal; see, e.g., [3]). Although there is a lot of scientific literature on either container logistics or data mining, there are only very few publications which combine both research fields (e.g., [1, 4, 5, 6, 7]), but they are not focused on EC management.

References

[1] G. Costa, G. Manco, R. Ortale, D. Sacca, A. D'Atri, and S. Za. Logistics management in a mobile environment: A decision support system based on trajectory mining. In *Proc. of Second Intern. Conf. on Systems 2007 (ICONS'07), Sainte-Luce, Martinique*, pp. 34–39, 2007.

[2] Dynamar. Dyna liners trade review. e-mail newsletter, 2005. Dynamar BV, Alkmaar.

[3] N. Huynh and N. Hutson. Mining the sources of delay for dray trucks at container terminals. *Transportation Research Record: Journal of the Transportation Research Board*, 2066:41–49, 2008.

[4] J. Kang, K. R. Ryu, and K. H. Kim. Determination of storage locations for incoming containers of uncertain weight. *Lecture Notes in Computer Science*, 4031: 1159–1168, 2006.

[5] W. H. K. Lam, P. L. P. Ng, W. Seabrooke, and E. C. M. Hui. Forecasts and reliability analysis of port cargo throughput in Hong Kong. *Journal of Urban Planning & Development*, 130(3):133–144, 2004.

[6] W. Li, B. Li, and Y. Zhang. Container terminal scheduling and decision-making using simulation based optimization and business intelligence. In *Proc. of IEEE Intern. Conf. on Mechatronics and Automation 2008 (ICMA 2008), Takamatsu, Japan*, pp. 1053–1058, 2008.

[7] W.-Y. Peng and C.-W. Chu. A comparison of univariate methods for forecasting container throughput volumes. *Mathematical and Computer Modelling*, 50(7-8):1045–1057, 2009.

[8] United Nations Conference on Trade and Development (UNCTAD) – secretariat. Review of maritime transport, 2008.

Single-commodity stochastic network design

Stein W. Wallace

Department of Management Science

Lancaster University Management School, Lancaster LA1 4YX, UK

Email: stein.w.wallace@lancaster.ac.uk

Teodor G. Crainic

University of Quebec at Montreal

Michal Kaut

Norwegian University of Science and Technology

Biju K. Thapalia

Molde University College, Norway

1 Background

There are many real-life problems that can be described as network flow problems and for most (if not all) of them there is an underlying design problem. The purpose of this paper is twofold. Firstly, we study the relationship between the stochastic and the deterministic single commodity network design problems, studying both random demand and random arc capacities. Secondly, we characterize the stochastic designs directly, without a specific reference to the deterministic case. Although connected, these two problems provide different perspectives on stochastic network design.

The traditional approach to network design is to formulate deterministic models. The demand is usually set to its expected value or sometimes some other, somewhat higher, value, to cater for "normal variation". In almost all cases, it is understood that the demand is actually stochastic, but the handling of stochasticity is deferred to the operational planning level. The reasons for doing so can be many: Computational complexity even of the deterministic network design model; a view that modelling wise, we know too little about demand while still being at the network design level of the planning; or simply that it is appropriate to postpone such details of the plan. After all, the goal is to set up the network, not decide how to route the flow.

The question we ask here is: how much do we lose by not taking stochasticity in demand into

account already at the design level? Could it be that the design coming from a model which is explicitly told that the future demand and arc capacities are uncertain is substantially better than a design not based on this knowledge? *Given* the distributional information used in the stochastic formulation, the design coming from the stochastic model will by definition be better (measured by the objective function) than the design from any deterministic model. Of course, if the distributional information is substantially incorrect, a deterministic design might (by chance) behave better in the real world. That, however, is not the focus of this work. Rather, what we are interested in is, given distributional information, how much better is the stochastic design, and even more importantly: In what ways do the stochastic designs differ from their deterministic counterparts, that is, *what* is it that makes one design better than the other? We know it is related to investment in flexibility, see [4] for a discussion in the framework of option theory, but we would like to know rather precisely what this investment in flexibility consists of. And conversely, we are also interested to see if some structures from the deterministic design actually carry over to the stochastic counterpart, so that, in fact, the deterministic solution contains useful information.

We thus study the structural difference between the deterministic and stochastic formulation, as well as try to charaterize the stochastic design directly, so as to better understand the phenomenon of investing in flexibility. We also hope to use the results to develop algorithms to solve the problem approximately (for large cases) or potentially to optimality (for moderate cases).

Our work is related to that of [2]. They study the multi-commodity problem. They identify two major structural differences: In the stochastic solution it is valuable to have several paths for each commodity and each of these paths should be shared with other commodities. Sharing is particularly useful in the case of negative correlations between demands. Without enforcing consolidation, their networks end up as consolidation networks, often hub-and-spoke. Contrary to conventional deterministic design, consolidation is in this case a hedging device, not a volume related undertaking. Hence, they identify structures that can be seen as investments in flexibility, that is, options, along the lines of [4]. Deterministic models would not produce such results.

2 Questions and Tests

In order to check the quality of the deterministic designs, as well as comparing them to the stochastic ones, we have set up three tests, named *comparisons*. Whenever a comparison is performed, we take the deterministic and stochastic designs – or parts thereof – (i.e. the first-stage solutions) and evaluate them using reference trees – in our case trees with 1000 scenarios, to make sure we have good approximations of the true distributions. The costs from the design and evaluation phase are added up, making the reported costs comparable across all tests. This out-of-sample evaluation means that there is no principal guarantee that a stochastic solution (being based on

for example 100 scenarios) should be better than a deterministic one. Indeed, we see cases where the deterministic solution is slightly (less than 1%) better.

The three comparisons are:

A The classical test where the whole first-stage solution is evaluated out-of-sample. This amounts to solving a 1000-scenario stochastic program with all first-stage variables (designs and capacities) fixed, so in fact this equals the solution of 1000 independent second-stage problems. Since the second stage does not involve any integer variables, this is very fast.

B Only edge information is imported from the first stage. So, in a 1000-scenario stochastic program, all discrete variables describing opened and closed edges—we call it a *skeleton*— are fixed and the stochastic program is run. So the model is allowed to install any capacity on the opened edges (also lower than in the deterministic case), but not to open new ones.

C The whole design (both the skeleton and its capacities) is taken as input to the 1000-scenario stochastic program. The stochastic program can then add new capacities on already opened edges (paying only variable setup costs) and new edges (paying both fixed and variable setup costs). Hence, all capacities opened in the deterministic case add cost to the objective function, even if these are not needed in the final design.

The purpose of Comparisons B and C is to check if the design from the deterministic solution really is good for the stochastic case, and if it bad, in what way it is bad. By making edges from the deterministic case "free" in two different ways, the stochastic programs (as defined in the comparisons) are guided towards the deterministic solution. This way we compare if stochastic programs solved with input from the deterministic solutions behave much worse than stochastic programs which have no deterministic input.

So, Comparison A is the classical test of the quality of the deterministic solution. Comparison B, on the other hands, checks if we can use a deterministic method to determine the skeleton and then a stochastic *linear* program to set the capacities. If Comparison B comes out with good results, it points to an alternative solution procedure that avoids solving a stochastic mixed integer program: First use a deterministic method to find the skeleton, then a stochastic linear program to set capacities. This represents a severe saving in computation (if it works well, of course).

Comparison C can be seen as testing what happens if we first solve the deterministic design problem and implement the solution, but then discover that it is not very good, and wish to update it. If Comparison C comes out well, a deterministic design can be corrected and become almost optimal for the stochastic case, provided setup costs must not be paid again on opened edges. If Comparison C comes out badly, the costs of updating a deterministic design in light of uncertainty in demand will be high. Note that Comparison C is itself a stochastic mixed integer program, so in most cases it does not represent an alternative solution approach.

3 Discussion

In the talk we shall show how the deterministic and stochastic designs relate to each other. We shall see that in some cases Comparison B yields good results, implying that the deterministic design actually has a good structure, though might be off in terms of capacities installed. In other cases, the comparison is in line with what is expected: The structure itself is too limited. Comparison C is generally rather good for these problems, indicating that the deterministic solution can be updated to become good if setup costs need not be paid again for those edges opened in the deterministic solution.

References

[1] J. L. Higle and S. W. Wallace. Sensitivity analysis and uncertainty in linear programming. *Interfaces*, 33:53–60, 2003.

[2] A.-G. Lium, T. G. Crainic, and S. W. Wallace. A study of demand stochasticity in stochastic network design. *Transportation Science*, 43(2):144–157, 2009.

[3] Stein W. Wallace. Decision making under uncertainty: Is sensitivity analysis of any use? *Operations Research*, 48(1):20–25, 2000.

[4] Stein W. Wallace. Stochastic programming and the option of doing it differently. *Annals of Operations Research*, page to appear, 2009.

The vehicle routing problem with driver assignment

Min Wen

Department of Transport

Technical University of Denmark, Kgs. Lyngby, Denmark

Email: mw@transport.dtu.dk

Emil Krapper, Jesper Larsen and Thomas K. Stidsen

Department of Management Engineering

Technical University of Denmark

1   Introduction

In this work, we consider a *Vehicle Routing problem with driver assignment* arising at the largest fresh meat producer in Denmark, Danish Crown. Danish Crown delivers the fresh meat from its distribution terminals to the supermarkets all over Denmark. The supermarkets place their orders with specified demand for different days of the week before the week starts. The distributor then makes a weekly delivery plan for the drivers and vehicles so that the orders are fulfilled, the drivers' working regulations are respected and the total travel cost is minimized.

2   Problem description

The problem is to determine delivery routes for a fleet of heterogeneous vehicles and a number of drivers with predefined working regulations over a one-week planning horizon.

A number of practical constraints need to be considered regarding the delivery. First of all, each customer orders a different amount of meat every day and each vehicle has a limited capacity. Secondly, each customer has a certain time window for receiving its order. These time windows are based on numerous factors such as working hours of the employees in the supermarket, city traffic etc. Lastly, certain special customers have requirements on the vehicle size. This is usually because of small roads or limited parking lot sizes. If an inappropriate vehicle type is used to serve such a customer, the driver usually needs to park some distance from the supermarket. This results in additional service time, which is proportional to the number of pallets ordered.

There are two kinds of drivers hired to carry out the delivery: internal and external drivers. The internal drivers work on the predefined workdays and for no more than a maximum weekly working duration (37 hours) over a week. Both the internal and external drivers start from given starting times and finish before given latest ending times. The drivers cannot drive for more than 4.5 hours without a 45-minute break according to the EU driving legislation.

Several different types of costs are considered in this problem. We assume that the internal drivers have regular salaries according to their contracts. Hence only the fuel cost of the routes taken by the internal drivers are considered, which depends on the distance travelled and the cost per kilometer. The external drivers are paid at a fixed price every hour, which covers both the salary for the driver and the vehicle cost. Therefore, the cost of the external routes is calculated by multiplying the route duration and the cost factor.

The objective of this problem can therefore be translated to minimize the fuel cost of the internal routes and the cost of the external routes over the planning horizon in such a way that each order must be served by one vehicle within its time window, vehicle capacities are not exceeded, each driver starts working at a predefined time and finishes before a given time on every workday, the internal drivers work for no more than a maximum weekly duration over the planning horizon, and the break rule regarding the driving legislation is respected.

3 Solution method

We propose to solve this problem using a heuristic. Firstly, the problem is NP-hard and secondly we foresee that the size of the problems that needs to be solved makes an exact approach prohibitive. The proposed method is named Multi-Level Variable Neighborhood Search heuristic (MLVNS) and illustrated in Figure 1.

The MLVNS consists of three levels. The first level (Level I) reduces the problem size through a node aggregation procedure which combines several nodes (customers) into a single supernode ([3]). The aggregation is based on the fact that several supermarkets may be located very closed to each other, for example in a big shopping center, and therefore it is very likely that these supermarkets should be visited by the same vehicle if it is feasible. Hence, the nodes are selected to be aggregated by analyzing their time windows, demands, and the travel times between them.

The second level (Level II) constructs the solution to the aggregated problem. Note that the orders on different days are fixed. The only constraint connecting the routes on different days is the maximum 37 weekly hours for the internal drivers, which implies that a certain driving schedule for an internal driver on one day will affect the maximum duration of the driver on the remaining days. Without this constraint, this weekly planning problem can be viewed as several independent daily planning problems, each of which considers the vehicle routing and driver assignment on a
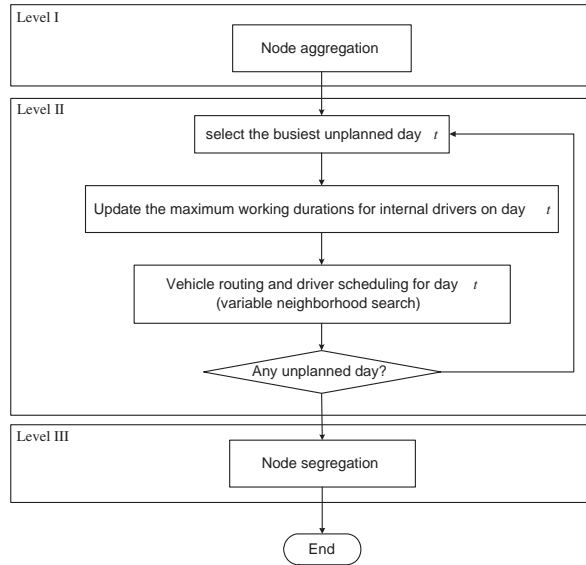
Figure 1: The flowchart of the MLVNS.

single day. To reduce the computational overhead, we decompose the weekly planning problem into six daily planning problems, which are then solved sequentially in a given order. Before a specific daily problem is solved, the maximum daily duration of each internal driver is updated based on the 37 week-hour constraints and the workload that has been assigned to the driver on the previously planned days. Given the updated information on the internal drivers, the daily distribution plan is determined by means of a variable neighborhood search ([2]). The proposed VNS consists of three components: initialization, a shaking phase, and a local search. An initial solution is constructed and improved iteratively. At each iteration, several neighborhoods are used in the shaking phase, and the Unified Tabu Search ([1]) is then applied in order to find good local optima. At the last level (Level III), the solution of the aggregated problem is expanded to a solution for the original problem. The visiting time at each customer is determined according to the sequence of the customers in each route.

4 Computational experiments

The proposed method is tested on real-life data provided by Danish Crown. The planning horizon consists of six days in a week, from Monday to Saturday. The data involves over 800 supermarkets and more than 2000 orders over a week. The total amount of meat delivered varies from day to day, ranging from 80 tons to 170 tons. The time windows of the supermarkets range from 1 hour to 24 hours. Approximately 10% of the supermarkets have requirements on the vehicle size. Three types of vehicles with different sizes and capacities are available for the delivery. Approximately

11 internal drivers and at most 14 external drivers are available every day.

A number of computational experiments are carried out to analyze the sensitivity of the parameters used in the algorithm. In the first experiment, we investigate the effectiveness of the node aggregation. Three different degrees of aggregations, which reduce the data size by 25%, 35% and 50%, are tested. The results show that solution converges faster with a more aggressive aggregation. A good trade-off between the running time and solution quality is obtained when the problem size is reduced by 25%. In the second experiment, we create two different scenarios. In one scenario, there is no supermarket that has requirement on the vehicle size. In the other, there are 20% supermarkets that have special requirements on the vehicle size. It is shown that increasing the portion of these supermarkets has a major effect on the total travel cost and duration, but only a minor effect on the number of vehicles required. This is because the special requirements on the vehicles do not change the overall capacity needed and are treated as soft constraints. The violations of these soft constraints are compensated by additional duration and cost.

We also compared our solution with the route plan used by Danish Crown. Since the only accessible information about the real-life plan is the list of customers served in every route on every day, whereas the exact order in which and the time at which each customer is visited are not available. We therefore calculated a TSP lower bound on the travel distance for each route. Compared to the real-life plan, our method yields an improvement between 13% and 26% in terms of the travel distance. Moreover, the number of vehicles required is also reduced by 20% on average.

5   Conclusion

We have addressed a route planning problem, in which a weekly routing plan has to be made for a fleet of heterogeneous vehicles and for a number of drivers to deliver the fresh meat to the supermarkets according to their demands and preferences. A multi-level variable neighborhood search based heuristic is proposed and tested on real-life data.

References

[1] Cordeau, J., Laporte, G., and Mercier, A. (2001). A unified tabu search heuristic for vehicle routing problems with time windows. *Journal of the Operational Research Society*, 52(8):928–936.

[2] Mladenovic, N., and Hansen, P. (1997). Variable neighborhood search. *Computers & Operations Research*, 24(11):1097–1100.

[3] Oppen, J., and Lokketangen, A. (2006). Arc routing in a node routing environment. *Computers & Operations Research*, 33(4):1033-1055.

Encouraging efficient usage of railway infrastructure through pricing mechanisms

Adrian S. Werner (Corresponding Author)

Department of Applied Economics and Operations Research

SINTEF Technology and Society, 7465 Trondheim, Norway

Email: AdrianTobias.Werner@sintef.no

Arnt-Gunnar Lium

Department of Applied Economics and Operations Research

SINTEF Technology and Society

1   Introduction

Efficient and reliable movement of freight and passengers is a key component for economical growth in today's society. For this reason, significant resources have been invested in building transportation infrastructure. Railroad infrastructure is very costly to develop. At the same time, it enables the movement of large quantities of goods and passengers. Even small improvements in its utilization have a significant impact on the return on investment. Consequently, removing bottlenecks and shifting traffic from peak periods to non-peak periods has a strong positive effect on the return on investment and on the environment as it stimulates the migration of passengers and freight transport from roads to rail. This will reduce emission of green house gases, noise, or accidents, and limit the need to invest in additional road infrastructure.

We discuss the development of methods encouraging efficient utilization of railroad infrastructure, thereby making rail transportation more competitive in comparison to other modes of transportation. We propose a methodology to enable pricing of the railroad infrastructure such that it is used in an economically optimal way, benefiting the society as a whole. Obviously, the underlying mathematical framework will be complex and difficult to tackle even for small, isolated cases. We meet this by combining socio-economic modeling concepts with Operations Research methods, in particular approaches from bilevel programming and stochastic service network design.

In this talk, we will set the stage for developing such a framework. We focus on describing the problem background and set-up, highlighting challenges related to this approach. Such challenges

may arise from different goals and decision processes of the involved actors but may also be of a more general and practical nature. Sketching the structure of a modeling framework which addresses these complexities, we point out the further direction of our research. An additional goal of our talk is to, hopefully, stimulate some discussion around this complex topic.

2   Problem background

Several EU Commission White Papers lay out the Commission's view on a European transport policy. Special attention is paid to further introduction of free market structures and demonopolization in the railroad sector. During the past decades, this reorganization process lead to a vertical separation of the sector into traffic operations and railroad infrastructure divisions. In such a structure, various independent Railroad Operators (ROs) can provide train services, either in direct competition or in different market segments. It is important to notice that different ROs can have different objectives and requirements when creating schedules. Track infrastructure including stations, installations etc. is controlled by an Infrastructure Manager (IM) such as Jernbaneverket. One of this entitys tasks is to assign rights to access the infrastructure to the single ROs in a best possible way.

Often, the process of allocating network access rights (as, e.g., outlined in Jernbaneverket's Network Statements [4]) is based on strategic and political considerations. Conflicts are resolved through an administrative committee and only a limited number of alternatives are evaluated at a time. The allocation does, therefore, not reflect the utility or relative importance of the allocated infrastructure to the IM or the RO, and it is controlled by the users requests rather than the IMs requirements. Moreover, the reliance on administrative and political mechanisms does not guarantee efficiency of the assignment process or of the resulting schedules.

Currently, ROs pay for using railroad infrastructure by way of charges which, in theory, shall be based on short-term socio-economic costs [3, 4]. Generally, the charges do not differentiate with respect to track segment or time of the day [4]. Only a few special charges apply for specific track segments or train types. However, price differentiation becomes more and more important as key for the sector's competitiveness.

Hence, it seems comparatively easy to extend already existing charging principles to a more detailed process, rather than introducing a completely new layer through additional mechanisms. Of course, a process allocating usage rights should not rely solely on charging to resolve potential conflicts between ROs' preferences or to achieve the IM's goals. Instead, the charges may be used to guide the ROs' behavior, in conjunction with other political or strategic tools.

3 Modeling framework

The allocation of infrastructure access rights is closely connected with the process of constructing efficient time tables for the single operators. Moreover, when studying complex traffic scheduling problems attention must also be paid on network effects such as "ripple effects" which may be difficult to identify immediately. This emphasizes the need for models studying the infrastructure network in a larger perspective rather than separate small instances. However, mathematical models become quite complex already for small networks and most modeling and solution approaches tend to focus on only a few aspects at a time.

So far, mathematical models for the allocation of access rights often relied on auctioning approaches. For example, Nilsson [5] uses auctioning to extract knowledge about the ROs' preferences, but the method required that only few isolated conflicts occur and the track network is not overly complex. Other approaches [1] combine the time tabling process and the allocation of access rights, strongly focusing on the first issue. Utilizing combinatorial auctions, robust schedules on a network of consecutive track segments can be constructed. The utilization of auctioning processes for allocating access rights to network capacity has been debated, but few alternative approaches are known.

Our goal is to construct plans that take into account the entire network, the time-space dimension as well as the charges for using the infrastructure. Hence, we aim at exploiting model properties like network structures, time slots and the interrelations of ROs and the IM. In this regard, especially two fields of mathematical programming become important, bilevel programming and stochastic service network design.

Bilevel programming problems (Stackelberg games) describe the interplay between two or more independent decision makers in a hierarchical relation. Each decision maker solves its own optimization problem; the IM does not take into account the ROs objectives and vice versa. They interact only through submitting selected solutions which are then used as parameters (in the lower hierarchy level) or responses on the parameters (in the higher levels). Consequently, each RO can come up with an optimal response for a given infrastructure charge while utilizing its assets in an optimal way, whatever its objectives are. The IM, on the other hand, can use the charges to influence the ROs' decision process.

The framework of bilevel programming comprises auctions and other game-theoretic mechanisms but draws upon a more general field of theory. Due to their generic nature, the models are well suited to take into account several aspects affecting the ROs' decisions such as the opportunity to switch to road transport. The derived charges can directly reflect the costs of operating the infrastructure. Auctioning mechanisms tend to mask these actual costs and rather show the ROs' valuation of infrastructure usage. Therefore, bilevel programming concepts also contribute toward usage-efficient allocation and the realization of political and/ or administrative goals as

these aspects are within the realm of the IM rather than the ROs.

While the upper level of a bilevel problem addresses what the optimal price for using the underlying infrastructure is, the sub-problems address how the actors should plan their operations in the best possible way for given prices in the upper level problem. To solve the lower-level problem we formulate this as a time-dependent *stochastic service network design problem*. Service network design is an extension of network design and deals with consolidating transportation systems to provide decision support on issues such as the selection and scheduling of offered services, routing of freight and passengers or terminal utilization and how to deal with potential congestion. Service network design typically focuses on maximizing profit or minimizing the total operational system cost for the carrier while maintaining a usually pre-specified customer service level, and the models can be used for different modes and multimodal transportation [2].

4   Conclusions

We discuss methods encouraging efficient utilization of railroad infrastructure, thereby making rail transportation more competitive in comparison to other modes of transportation. The situation as it currently exists in practice is illustrated through an example involving several ROs on the Norwegian railway network and the IM, Jernbaneverket. The underlying mathematical framework will be complex and difficult to tackle even for small, isolated cases. We highlight therefore some major research challenges for the mathematical modeling of the situation and for solution approaches.

References

[1] R. Borndörfer, M. Grötschel, S. Lukac et al, "An auctioning approach to railroad slot allocation", *Competition and Regulation in Network Industries*, 1, 163-196 (2006).

[2] T. G. Crainic, K. H. Kim, C. Barnhart and G. Laporte. "Intermodal transportation" in *Handbooks in Operations Research and Management, Transportation, vol. 14*, North-Holland, Amsterdam, 467-537, 2007.

[3] Directive 2001/14/EC of the European Parliament and of the Council, "On the allocation of railroad infrastructure capacity and the levying of charges for the use of railroad infrastructure and safety certification", 2001.

[4] Jernbaneverket, "Network Statement 2010. 7th Edition, 1 Dec. 2008. Validity period 13 Dec. 2009 - 11 Dec. 2010", 2010.

[5] J.E. Nilsson, "Towards a welfare enhancing process to manage railroad infrastructure access", *Transportation Research Part A*, 36, 419-436 (2002).

Multi-objective Network Design Problem: minimizing externalities using dynamic traffic management measures

Luc J.J. Wismans (corresponding author)

Centre for Transport Studies

University of Twente, Enschede the Netherlands

Email: L.J.J.Wismans@ctw.utwente.nl

Eric C. Van Berkum

Centre for Transport Studies

University of Twente

Michiel C.J. Bliemer

Department of Transport & planning

Delft University of Technolgy

1 Introduction

Optimization of a road transport system is often viewed as a problem to find the best way to expand or improve an existing network. This type of problem is generally referred to as the Network Design Problem (NDP). One specific example of this is to optimize a network through the implementation of dynamic traffic management (DTM) measures which can influence the supply of infrastructure dynamically (e.g. traffic signals, ramp metering and rush hour lanes). Traditionally, this type of optimization is focused on improving accessibility, subject to some conditions regarding externalities as traffic safety or livability (set by law). However, due to the increasing attention for these type of externalities, it may no longer suffice to view a transport system as feasible when it meets these conditions. Therefor we will view de NDP as an optimization problem with multiple objectives, where externalities are incorporated in the objective functions [1].

The NDP is usually formulated as a bi-level problem in which the lower level describes the behavior of road users that optimize their own objectives. Usually this is operationalized as a user equilibrium. The upper level consists of the objectives that have to be optimized for solving the NDP. Because of the non convexity of the problem (e.g. [2]), several global solution approaches are used, including genetic algorithms and simulated annealing. There are several studies that formulated multi-

objective (MO) NDP in which for example the budget constraint is formulated as a second minimization problem and optimization studies in which externalities, mainly air quality, are used as objective (e.g. [3], [4], [5], [6]). In the bi-level optimization studies the solution approach using genetic algorithms has been proven successful, but still requires many function evaluations. These studies, however, have been limited to considering only a few externalities and often focused on local optimization. Further, in the lower level mainly a static user equilibrium was used. Although this choice is understandable, because of the many function evaluations needed, Dynamic Traffic Assignment (DTA) models are more suitable to assess the effects of DTM measures. Different researches have shown that there is a proven relation between the traffic dynamics and external effects like emissions of pollutants and traffic safety. High speeds and speed variation (accelerating, braking) have for example a negative effect on traffic safety and emissions of pollutants [7].

2 Model framework and methodology

Within this study, the optimization of externalities using DTM measures is formulated as a bi-level optimization problem. In our case we focus on strategic DTM measures optimizing the objectives on the long term. In the lower level road users optimize their own objective (travel time). This is operationalized by solving the Dynamic User Equilibrium problem using the INDY traffic model, which is a DTA model with dynamic queueing and spillback. Output of this model are speeds and flows on all links of the network as a function of time. From this, the level of service of all network elements can be determined as a function of time.

The upper level consists of the optimization of the objectives of the road management authorities concerning accessibility, air quality, climate, traffic safety and noise by applying available DTM measures. Based on an extensive literature review [8] for each objective an objective function is defined, where the input stems from a DTA model [9]. Accessibility is defined as the total travel time in the network, which is a direct result of the DTA. Air quality is defined as the total weighted emission of PM_{10} (or NO_x). The weights are related to the level of urbanization, and the emissions are determined based on a traffic situation based emission model, which means depend on the level of service of the traffic flows. Climate is defined as the total emission of CO_2 and is determined based on an average speed based emission model. Traffic safety is defined as the total number of injuries and is determined based on a accident risk based model. Finally noise is defined as the average weighted sound power level is calculated, in which the weights of emissions as for air quality depend on the level of urbanization, and emissions are based on a load and speed dependent emission function.

A DTM measure is modelled as a measure that influences the fundamental diagram on the links where the measure is implemented. The impact of the measure depends on the actual settings, e.g. the green time for a certain direction on a signalized intersection. Time and settings of the DTM measures are discretized, so the upper level then becomes a discrete optimization problem where for each time period a certain DTM measure with a certain setting is implemented or not.

The upper level optimization was solved using metaheuristics, in particular we applied genetic programming methods. We tested three solution approaches in which specific attention was paid to the determination of the initial set of solutions and restriction of the solution space to accelerate the search. For this some form of pre optimization was used where the inital set and rectristriction of the solution space is a result of a static optimization.

3 Application and conclustions

A case study on a small hypothesized road network consisting of one OD pair, three routes and four DTM measures (11 possible settings, 6 time periods) is conducted, to show the feasibilty of the solution approaches. Although the network was small, it did incorporate the major elements like urban and non-urban routes when using DTM measures to optimize accessibility and externalities. Moreover, these objectives were modeled in a realistic manner incorporating traffic dynamics.
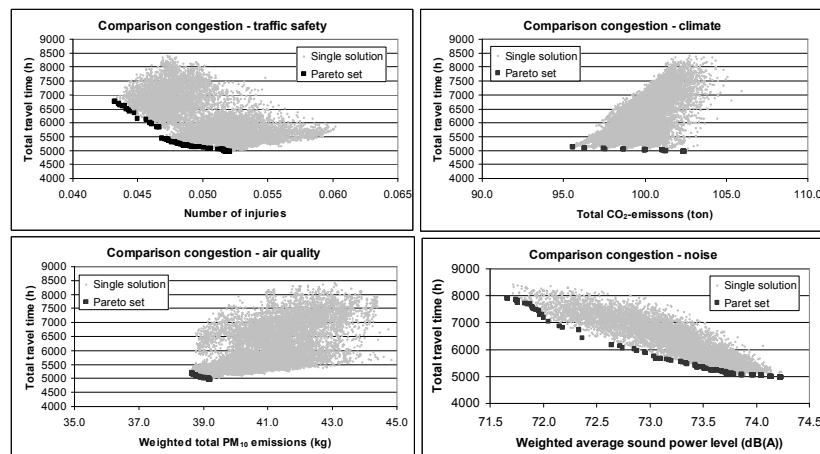
FIGURE 1 Objective functions solutions related to congestion

The results show that in this testcase the objectives for congestion, traffic safety, emissions and noise show different optimal solutions, which means there is not a combination of measures resulting in an optimal situation for a combination of all objectives. However, it was found by investigating the Pareto optimal sets that the objectives concerning air quality and congestion are aligned and that these objectives are opposite to noise and traffic safety (see figure 1). This can be explained, because optimizing congestion aims at avoiding congestion using full capacity of the available routes. Optimizing traffic safety aims at maximizing the use of the relatively safe highway route and avoiding use of the urban route. Optimizing emissions aims at avoiding congestion and high speeds and searches for the best trade of between minimizing traffic using the urban roads and the level of congestion on the highway. Optimizing noise aims at lowering the driving speeds as much as possible and avoiding traffic using the urban routes. Concerning the application of the three solution approaches we found that all approaches were able to find improvements and there is no reason to

assume that reducing the solution space as we did results in sub optimal solutions. However, the results also show that pre optimization does not enhance the optimization process. The optimization requires even for this small network a substantial number of function evaluations. Because we are mainly interested in finding improvements and not necesseraly the exact Pareto optimal set, the approaches are scalable. However, since the lower level requires in these approaches, especially for larger networks, a substantial amount of CPU time, more research is needed to enhance the efficiency of possible solution approaches. These enhancements can be achieved by incorporating more knowledge of road transport systems to reduce the solution space more effectively or to optimize the solution approach (e.g. by using function approximation) in order to reduce the number of time consuming function evaluations.

References

[1] Ministerie Verkeer en Waterstaat. *Nota mobiliteit. Dutch policy document*. Den Haag, 2004.

[2] Gao, Z., Wu, J. and Sun, H. Solution algorithm for the bi-level discrete network design problem. Transportation Research B, Vol. 39, 2005, pp. 479-495.

[3] Sharma, S., Ukkusuri, S.V. and Mathew, T. A Pareto optimal multi-objective optimization for the robust transportation Network Design Problem. Presented at the 88th Annual Meeting of the Transportation Research Board, Washington DC, USA, 2009.

[4] Lu, Z. L., Fan, B.Q., Liu, J.J. and Zhou, Y.H. Bi-level multi-objective programming model for the ramp control and pollution control on urban expressway networks. Kongzhi yu Juece/Control and Decision Vol. 21(1), 2006.

[5] Schmöcker, J.-D., Ahuja S. and Bell, M.G.H. Multi-objective signal control of urban junctions - Framework and a London case study. Transportation Research Part C: Emerging Technologies, Vol. 16(4) pp. 454-470, 2008.

[6] Cantarella, G.E., Pavone, G. And Vitetta, A. Heuristics for urban road network design: Lane layout and signal settings. European Journal of Operational Research, Vol 175, pp. 1682–1695, 2006.

[7] Beek, W., Derriks, H., Wilbers, P., Morsink, P., Wismans, L.J.J., Beek, P.. The effects of speed measures on air pollution and traffic safety. In proceedings European Transport Conference 2007.

[8] Wismans, L.J.J., Van Berkum, E.C. and Bliemer, M.C.J. Modeling externalities using dynamic traffic assignment models: A review. Submitted to the 89th Annual Meeting of the Transportation Research Board, 2010.

[9] Wismans, L.J.J., Van Berkum, E.C. and Bliemer, M.C.J. Dynamic trafic management measures to optimize externalities: effects of single objective optimization. Working document, 2009

Calibration of Automatic Pedestrian Counter Data Using A Nonparametric Statistical Method

Hong Yang (Corresponding Author)

Kaan Ozbay

Bekir Bartin

Rutgers Intelligent Transportation Systems (RITS) Laboratory

Department of Civil and Environmental Engineering

Rutgers, The State University of New Jersey, USA

Corresponding Email: yanghong@eden.rutgers.edu

1 Introduction

Pedestrian traffic data are one of the important sources of information for pedestrian planning studies. Compared to vehicular count data, existing pedestrian counts are in general not accurate enough to be directly used for planning studies. This is beacuse the methods or technologies currently employed to collect pedestrian data were not as advanced as the ones used for vehicluar data collection. Moreover, pedestrian traffic patterns are much more complex than vehicular traffic making it even more difficult to collect accurate data. Although practioners have been looking for more advanced ways to efficiently and accurately collect pedestrian data, the majority still relied on traditional methods such as, videotapes, and manual counting with click boards. Factors such as human error, and labor cost have limited the use of these traditional methods for long-term pedestrian data collection.

Rather than relying on these costly and error-prone data collection methods, recent advances in sensor technologies can be used for long-term pedestrian data collection. However, even the output of most advanced sensors should be carefully used because in several recent studies most of the automatic counters were found to be less than 100 percent accurate [1], [2]. These studies clearly showed that every autmatic pedestrian sensor more or less suffered from specific error factors that need to be incorporated into the data processing procedure.

2 Problem

Among the commerically available technologies, infrared counters are one of the most widely used sensors for monitoring pedestrian traffic. It is easy to find examples of their use in indoor settings such as shopping malls and visitor centers. Apllications of infrared counters in outdoor settings such as sidewalks or trails, however, are less common due to accuracy concern. Since infrared counters require single pedestrian passing to achieve maximum accuracy, they will systematically undercount when

people walk side by side or in groups [3]. **Figure 1** shows an example when an undercount will occur. In this example, the counter will count only one pedestrian in all three cases shown in **Figure 1**.

(a)Pattern 1: Truth=1 (b)Pattern 2: Truth=2 (c) Pattern 3: Truth=3

Figure 1. Example of possible passing patterns given counter output is one

Regarding the counter errors, researchers have been developing calibration methods to adjust original counter outputs [4], [5], [6]. They developed regression models using counter outputs as the predictor variable to estimate actual counts. Essentially, all of these regression models attempted to estimate adjustment factors to correct raw sensor counts. However, there was no universal adjustment factor. The transferability of these regression models was problematic because the correction factor may vary from site to site, and time to time. It is difficult to propose a single calibration factor as it depends on various conditions, for instance, how pedestrians use the facility (single line or walking side by side), how busy the facility is, geometric characteristics of the facility, and so on. Moreover, it is laborious to collect large enough sample of counter outputs and the corresponding ground-truth counts to build reliable and robust regression models.

Instead of using regerssion models, the primary objective of this study was to develop a new method for adjusting the infrared counter outputs. By comparing the counter outputs and the associated errors, a novel statistical modeling procedure was proposed to estimate the actual counts using the raw counter ouputs.

3 Proposed Methodology

A bivariate bootstrap sampling procedure was proposed to estimate the hourly pedestrian counts using the raw counter outputs of 15-minute interval. In this proposed method, we still needed the ground-truth data. However, use of the boothrap sampling procedure made it possible to build a large synthetic dataset from the limited field observations. This allowed us to achieve better calibration results with a relatively small original dataset. The procedure was summarized as follows:

Step 1: Let (X_i, Y_i) be a pair of counter output and corresponding actual (ground truth) pedestrian volume at the ith 15-minute interval, i=1, 2, …, n.

Step 2: Randomly sample 4 pairs of(X_1', Y_1'), (X_2', Y_2'), (X_3', Y_3'), and (X_4', Y_4') from (X_i, Y_i) with replacement. Define (C_j, M_j) as the jth pair of hourly counter output and actual volume, and caculate the hourly counting error rate as follows:

$$C_j = \sum_{j=1}^{4} X_j^{'} \tag{1}$$

$$M_j = \sum_{j=1}^{4} Y_j^{'} \tag{2}$$

$$\varepsilon_j = \frac{C_j - M_j}{M_j} \times 100\% \tag{3}$$

Step 3: Repeat steps 2 and 3 B times (B is a large number), and list $(C_j, M_j, \varepsilon_j)$ in a lookup table, where j=1, 2, …, B.

Step 4: Given a new hourly counter observation Q, define an interval $A = [Q - \delta, Q + \delta]$, where δ is a small value. Construct $(C_j, M_j, \varepsilon_j)$ vectors in lookup table where $C_j \epsilon A$, and denote them as $(C_k^{'}, M_k^{'}, \varepsilon_k^{'})$, where k=1 to the total number of subsets (assumed to be K). The aim is to find reference counter outputs which are close to or equal to Q.

Step 5: The correction factor for Q is determined as $E(\varepsilon_k^{'})$, where $E(\varepsilon_k^{'})$ is the expectation of $\varepsilon_k^{'}$ in the subset. The calibrated counter output is then calculated as:

$$\hat{Q} = \frac{Q}{1 + E(\varepsilon_k^{'})} \tag{4}$$

Step 6: If necessary, construct percentile confidence interval for $\hat{Q}$: order $\varepsilon_k^{'}$ from smallest to largest. Identify $\varepsilon_L = [(\frac{\alpha}{2} \times 100\%) \times K]^{th}$ and $\varepsilon_U = [(1 - \frac{\alpha}{2}) \times 100\% \times K]^{th}$ values of the ordered values. These values represent the lower and upper limits for the $(1 - \alpha) \times 100\%$ confidence interval of correction factor. Then use them to calculate the lower and upper limits for $\hat{Q}$:

$$\hat{Q}_L = \frac{Q}{1 + \varepsilon_U} \tag{5}$$

$$\hat{Q}_U = \frac{Q}{1 + \varepsilon_L} \tag{6}$$

4 Emperical Tests

To validate the proposed calibration method, a dual-sensor pyroelectric infrared counter developed by EcoCounter was tested and calibrated as a case study. Training dataset consisted of field data including counter ouputs and actual counts collected at a trail on Rutgers University Busch campus on October 19 and 26, 2009. Lookup table was created according to the proposed procedure using the data of these two days. Test datasets include dataset collected on April 10 at another trail on the same campus and another dataset collected on October 12 at a sidewalk in downtown of New Brunswick, NJ. They were used to validate the performance of the proposed calibration method.

The calibration results for the two test datasets are shown in **Figure 2**. From the figure we can see that the infrared counter obviously undercounted pedestrians at these sites. For the dataset collected on April 10, the original overall error rate of the infrared counter was -20.7 percent. By using the calibration method, the overall estimated counts were only 1.2 percent more than the actual counts. Similarly, the overall counter error rate was -14.3 percent at the sidewalk on October 12 and the overall error of the calibrated counts was successfully reduced to -4.4 percent.
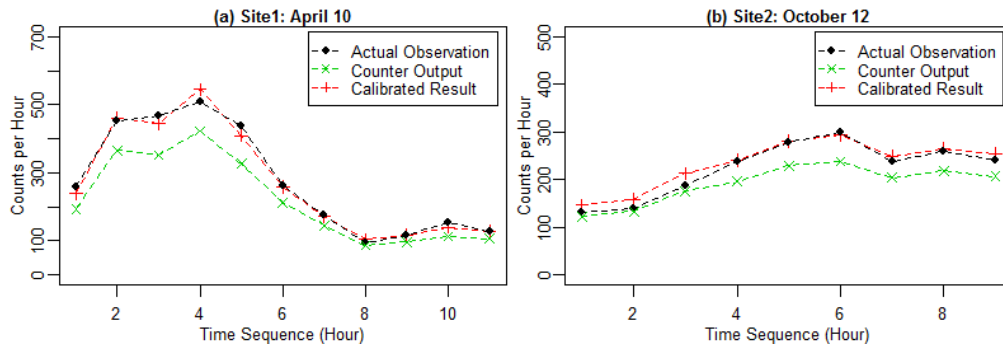
Figure 2. Calibration results at two sites

5 Conclusions

Automatic pedestrian counting method is a potential alternative for long-term pedestrian data collection. The large differences between the actual counts and counter outputs observed in this and other studies raises the need to adjust the raw counter outputs. Using a single correction factor is an impractical approach since it may vary by time and locations, etc. This study proposed a nonparametric statistical method to adjust the raw counts. The validation tests showed that the proposed method performed well and can be easily transfered to other sites even though the trained lookup table was not built using data from these sites.

References

[1] SRF Consulting Group, Inc. Bicycle and Pedestrian Detection. Minnesota Department of Transportation Research Report, 2003.

[2] S. Turner, D. Middleton, R. Longmire, M. Brewer, and R. Eurek. Testing and Evaluation of Pedestrian Sensors: Texas Transportation Institute, Texas A&M University, September 2007.

[3] K. Ozbay, H. Yang, and B. Bartin. "Calibration of An Infrared-based Automatic Counting System for Pedestrian Traffic Flow Data Collection". Presented at 89th Annual Meeting of the Transportation Research Board, Washington, D.C., 2010.

[4] G. K. Kuah. "Calibration of an Infrared Pedestrian Counting System for Shopping Malls", *Journal of the Transportation Research Board*, No.1210, 31-34 (1989).

[5] G. Lindsey and D.B.L. Nguyen. "Use of Greenway Trails in Indiana", *Journal of Urban Planning and Development*, ASCE.Vol.130, No.4, 213-217(2004).

[6] P. Lindsey and G. Lindsey. Using Pedestrian Count Models to Estimate Urban Trail Traffic. *Journal of Regional Analysis & Policy*, Vol.34, No.1, 51-68 (2004).

Dynamic Pricing, Heterogeneous Users and Perception Error: Bi-Criterion Dynamic Stochastic User Equilibrium Assignment

Kuilin Zhang

Transportation Center
Northwestern University

Hani S. Mahmassani

(corresponding author)
Transportation Center
Northwestern University, Evanston, IL, U.S.A.
Email: masmah@northwestern.edu

Chung-Cheng Lu

Institue of Information and Logistics Management
National Taipei University of Technology

1 Introduction

This paper presents a probit-based bi-criterion dynamic stochastic user equilibrium (BDSUE) model to address stochastic choice behavior of heterogeneous users with different value of time (VOT) preferences and different perceptions of travel costs. The model is motivated by the need to evaluate the impact of time-varying pricing schemes on network performance, which entails capturing heterogeneous travelers' choice behavior in response to such pricing policies [1]. In previous work, bi-criterion user equilibrium traffic assignment models, in which the value of time (VOT) is distributed across the user population, have been developed for both the static case (BUE) [2], [3], [4], and more recently for the case where flows and prices vary with time [5]. In particular, Lu et al. [5] introduced a continuous random VOT in the dynamic traffic assignment (DTA) context and proposed a parametric analysis method to solve the resulting bi-criterion dynamic user equilibrium (BDUE) problem.

Both aforementioned BUE and BDUE models were developed in the (deterministic) user equilibrium (UE) path choice framework, in which travel costs are known precisely to travelers. This assumption may not always be realistic, since users may also have different perceptions of travel costs. Accordingly, the probit-based BDSUE model introduced in this study can capture path choice behavior of heterogeneous users with both distinct VOT preferences and different perceptions of travel costs. In particular, across the population of travelers, the VOT is represented by a continuously distributed random variable, and the perception errors of travel costs in a choice set are multivariate normally distributed. The BDSUE problem is formulated as a fixed point problem in the infinite dimensional space, and solved by a column generation solution framework which embeds (i) a parametric analysis

method (PAM, [6]) to transform the continuous problem to the finite dimensional space by finding breakpoints that partition the entire range of VOT into subintervals and define a multi-class dynamic stochastic user equilibrium problem (MDSUE); (ii) path (column) generation algorithm to augment a feasible path set for each user class; (iii) a probit-based stochastic path flow updating scheme solving a Restricted MDSUE problem defined by the set of feasible paths; and (vi) dynamic network loading using a particle-based traffic simulator [7] to capture traffic dynamics and determine experienced travel times for a given path flow pattern.

2 Problem statement and assumptions

Consider a time-varying network $G = (N, A)$, where N is a finite set of nodes and A is a finite set of directed links; the time period of interest (planning horizon) is discretized into a set of small time intervals, $H = \{t_0, t_0 + \Delta t, t_0 + 2\Delta t, \ldots, t_0 + |T|\Delta t\}$, where t_0 is the earliest possible departure time from any origin node, Δt is a small time interval during which no perceptible changes in traffic conditions and/or travel cost occur, and $|T|$ is the cardinality of the set T of time intervals, such that the intervals from t_0 to $t_0 + |T|\Delta t$ cover the planning horizon H. The time-varying OD demands, $q_{wt}, \forall w \in W, t \in T$ (W is a set of OD pairs) for the entire planning horizon are assumed to be known *a priori*. Travelers have different VOT, α, and are subject to perception errors in selecting the best path from a path choice set $K_{wt}(\alpha)$. No en-route path-switching is allowed after departure from origins. Without loss of generality, associated with each link a and time interval τ are two time-varying attributes: travel time TT_a^τ and travel monetary cost TC_a^τ, which are required to traverse link a when departing at time interval $\tau \in T$ from upstream node of link a. By assuming path travel disutilities are additive in terms of their respective link travel disutilities, we define the path travel time and path travel money cost of a path k as $TT_k(\mathbf{x}) = \sum_{a \in A(k)} TT_a^\tau$ and $TC_k(\mathbf{x}) = \sum_{a \in A(k)} TC_a^\tau$ respectively. The experienced generalized cost or disutility for travelers of OD pair w with VOT α departing at time t along path k is defined as $GC_k(\alpha) = TC_k + \alpha \times TT_k$. To reflect the heterogeneity of the population, the VOT in this study is treated as a continuous random variable distributed across the population of travelers, with a density function $\varphi(\alpha) > 0, \forall \alpha \in [\alpha^{min}, \alpha^{max}]$, and $\int_{\alpha^{min}}^{\alpha^{max}} \varphi(\alpha) d\alpha = 1$, where the feasible range of VOT is determined by a given closed interval $[\alpha^{min}, \alpha^{max}]$. In the probit-based choice model framework, the systematic path disutility is given by the negative of the generalized cost as $\mathbf{V} = (-1) \times \mathbf{GC} = (-1) \times (\mathbf{TC} + \alpha \times \mathbf{TT})$, and the perceived path disutility is defined as $\mathbf{U} = \mathbf{V} + \boldsymbol{\varepsilon}$, where the random error vector follows a multivariate normal distribution, $\boldsymbol{\varepsilon} \sim \mathbf{MVN}(\mathbf{0}, \boldsymbol{\Sigma}_\varepsilon)$. Assuming that each traveler chooses a path that maximizes his/her perceived utility, the choice probability of each path $k \in K_{wt}(\alpha)$ for travelers with VOT α can be determined as

$p_k = p_k(\mathbf{x}) = \Pr[U_k(\mathbf{x}, \alpha) = \max_{k' \in K_{wt}(\alpha)} \{U_{k'}(\mathbf{x}, \alpha)\}]$. Hence a stochastic path choice approach is employed to represent both travel time perception errors and heterogeneity of VOT.

Under the above assumptions, the bi-criterion dynamic stochastic user equilibrium (BDSUE) conditions extend in a natural way the usual static stochastic user equilibrium (SUE) conditions [8]. The BDSUE implies that each traveler is assigned to a path with least perceived travel disutility with respect to his/her own VOT and perception error. This paper presents a mathematical formulation for the above-defined BDSUE network assignment problem, and develops and applies a solution procedure to find the time-dependent path and link flow patterns under a given time-dependent road pricing scheme in a general network.

3 Bi-criterion dynamic stochastic user equilibrium model

Typically, the static SUE conditions are defined based on the weak law of large and formulated as a fixed point problem [8]. Zhang et al. [9] extended the static SUE conditions to the dynamic context and proposed a fixed point formulation. Accordingly, considering VOT α as a continuous random variable, the BDSUE conditions can be stated mathematically as in Eq. (1):

$$x_k(\alpha) = q_{wt}(\alpha) \times p_k(\mathbf{x}), \forall k \in K_{wt}(\alpha), w \in W, t \in T, \alpha \in \left[\alpha^{\min}, \alpha^{\max}\right] \tag{1}$$

Define a map $\mathbf{F}[\mathbf{x}(\alpha)] = \mathbf{q}(\alpha) \times \mathbf{p}[\mathbf{x}(\alpha)]$, where $\mathbf{F}: \mathbf{R}_+^{K(\alpha)} \to \mathbf{R}_+^{K(\alpha)}$, $\mathbf{x}(\alpha) \in \Omega(\alpha) \subseteq \mathbf{R}_+^{K(\alpha)}$, and $\forall \alpha \in \left[\alpha^{\min}, \alpha^{\max}\right]$. The BDSUE problem of interest can be formulated as the following infinite dimensional fixed point (FP) problem in Eq. (2):

$$\mathbf{x}^*(\alpha) \in \Omega(\alpha), \text{ satisfying } \mathbf{x}^*(\alpha) = \mathbf{F}[\mathbf{x}^*(\alpha)], \forall \alpha \in \left[\alpha^{\min}, \alpha^{\max}\right] \tag{2}$$

Solving the above infinite dimensional FP problem will give the path flow vector $\mathbf{x}^*(\alpha) = \left\{x_k^*(\alpha) \mid \forall k \in K_{wt}(\alpha), w \in W, t \in T\right\}$, $\forall \alpha \in \left[\alpha^{\min}, \alpha^{\max}\right]$, which is also the solution of the BDSUE problem, i.e. $\mathbf{x}*$ would satisfy the BDSUE conditions given in Eq. (1).

4 Simulation-based column generation solution framework

The simulation-based column generation solution framework for solving the BDSUE problem includes three main steps: (i) input and initialization, (ii) parametric analysis of VOT α and path (column) generation, and (iii) solving the RMDSUE (restricted multi-class dynamic stochastic user equilibrium) problem. These steps are summarized below.

Step I: Input and initialization.

Step I.1: Input. Input a time-dependent OD demand matrix for the entire feasible range of VOT over the planning horizon, q_{wt}, a time-dependent link toll scheme, network topology, and VOT distribution.

Step I.2: VOT generation. Generate the VOT for each vehicle based on the given VOT distribution to obtain heterogeneous time-dependent OD demands (vehicles), $q_{wt}(\alpha)$.

Step I.3: Initial simulation-assignment. Set outer loop counter $m = 0$, and perform a dynamic network loading to obtain initial time-varying feasible path set $\widetilde{\mathbf{K}}^{(0)}$, experienced link and path travel time $\mathbf{TT}^{(0)}$ and monetary cost $\mathbf{TC}^{(0)}$, and path flow pattern, $\mathbf{x}^{(0)}$, from the traffic simulator.

Step II: Parametric analysis of VOT α and path (column) generation.

Step II.1: Bi-criterion dynamic shortest path calculation. Apply the PAM based bi-criterion time-dependent least generalized cost path (BTDLGCP) algorithm [6] to find a complete set of time-dependent extreme efficient paths and the corresponding set of breakpoints, $\boldsymbol{\alpha}^{(m)} = \left\{ \alpha_0, ..., \alpha_i^{(m)}, ..., \alpha_I^{(m)} \mid \alpha^{\min} = \alpha_0 < ... < \alpha_i^{(m)} < ... < \alpha_I^{(m)} = \alpha^{\max} \right\}$ that partitions the entire closed interval of VOT, $\left[\alpha^{\min}, \alpha^{\max} \right]$, and defines the multiple classes of travelers, $\mathbf{q}_{wt}\left(\boldsymbol{\alpha}^{(m)} \right) = \left\{ q_{wt}\left(\alpha_i^{(m)} \right) \mid \forall i = 1, ..., I \right\}$, each class $\alpha_i^{(m)}$ of which covers a subinterval of VOT, $\left[\alpha_{i-1}, \alpha_i \right), \forall i = 1, ..., I$. Starting from the lowest bound of VOT, $\alpha^{\min}$, the BTDLGCP algorithm continuously solves for a time-dependent least generalized cost path tree (TDLGC) for a given VOT subinterval and determines a upper bound of that VOT subinterval, $\alpha_i^{(m)}$, for which the TDLGC path tree remains optimal, until reaching the highest bound of VOT, $\alpha^{\max}$. The algorithm divides a continuous distributed VOT closed interval into a set of VOT subintervals (i.e., user classes) by parametrically analyzing the VOT, α, and simultaneously generating optimal paths (columns) and augments the restricted feasible path set, $\widetilde{\mathbf{K}}\left(\boldsymbol{\alpha}^{(m)} \right)$, for each VOT subinterval (user class).

Step II.2: Convergence checking for the outer loop. If no new path is found or *m = mMax*, then stop; otherwise go to Step III to solve the RMDSUE problem defined by the given restricted feasible path set, $\widetilde{\mathbf{K}}\left(\boldsymbol{\alpha}^{(m)} \right)$.

Step III: Solving the RMDSUE problem.

Step III.1 Initialization. Set inner loop counter *n* = 1, and prepare the restricted feasible path choice set, $\widetilde{\mathbf{K}}\left(\boldsymbol{\alpha}^{(m)} \right)$, from Step II.

Step III.2 Multi-class probit-based stochastic path assignment. Determine multi-class probit-based path assignment, $\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right)$, by the multi-class probit-based path flow updating/equilibrating scheme for the given reduced feasible path choice set.

Step III.3 Multi-class dynamic network loading. Perform a multi-class dynamic network loading to evaluate the multi-class path assignment, $\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right)$, and obtain time-varying travel costs (i.e., $\mathbf{TT}^{(n+1)}$ and $\mathbf{TC}^{(n+1)}$) and link flow pattern from the traffic simulator.

Step III.4 Convergence checking for the inner loop. If $g\left(\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right) \right) <= \epsilon$ (ϵ is a predefined convergent threshold) or *n = nMax* (maximum number of inner iterations), then go to Step II, and set *m = m* + 1; otherwise set *n = n* + 1, and return to Step III.2. $g\left(\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right) \right)$ is a gap measure defined as in Eq.(3), which is the sum of square of difference between assigned path flow (i.e., number of travelers assigned to a path) $x_k^{(n+1)}\left(\alpha_i^{(m)} \right)$ and expected path flow $q_{wt}\left(\alpha_i^{(m)} \right) \times p_k^{n+1}\left(\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right) \right)$ for each user class, OD pair, and departure time interval.

$$g\left(\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right) \right) = \frac{1}{2} \times \sum_w \sum_t \sum_k \sum_i \left[x_k^{(n+1)}\left(\alpha_i^{(m)} \right) - q_{wt}\left(\alpha_i^{(m)} \right) \times p_k^{n+1}\left(\mathbf{x}^{(n+1)}\left(\boldsymbol{\alpha}^{(m)} \right) \right) \right] \qquad (3)$$

5 Numerical results

The proposed BDSUE algorithm is implemented and tested on the Irvine (California, USA) network, consisting of 326 nodes (70 of them signalized), 626 links, and 61 zones. A two hour (7:00AM-9:00AM) morning peak time-dependent OD demand table is loaded to the network, with 35,304 vehicles in the observation period (7:10AM-8:50AM). A hypothetical toll station is created on a portion (about 1 mile) of the I-405 westbound freeway, where three of the five lanes are converted to toll lanes. The OD demand assignment interval (or departure interval) is set to 1 minute. The resolution (aggregation interval) of the time-dependent shortest path tree calculation is set to 6 seconds, the simulation interval of the traffic simulator. Table 1 lists the three dynamic pricing scenarios tested in the experiments, and Figure 1 shows corresponding time-varying toll road usage for each scenario.

Table 1 Dynamic road pricing scenarios

Pricing Scenario	Period 1 (7:00-7:30AM)	Period 2 (7:30-8:00AM)	Period 3 (8:00-8:30AM)	Period 4 (8:30-9:00AM)
Low	$0.10	$0.20	$0.30	$0.15
Middle	$0.15	$0.25	$0.35	$0.20
High	$0.20	$0.30	$0.40	$0.25
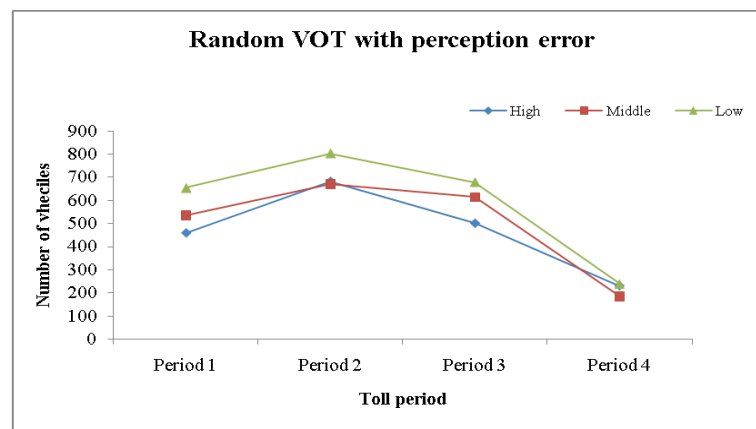
Figure 1 Time-varying toll road volume for different pricing levels in Irvine network

References

[1] W. S. Vickery, "Congestion Theory and Transport Investment", *American Economics Review* 72 (3), 467-479 (1669).

[2] F. Leurent, "Cost Versus Time Equilibrium over a Network", *European Journal of Operational Research* 71 (2), 205-221 (1993).

[3] R. B. Dial, "Bicriterion Traffic Assignment: Basic Theory and Elementary Algorithms", *Transportation Science* 30 (2), 93-110 (1996).

[4] P. Marcotte and D. L. Zhu, "Equilibria with Infinitely Many Differential Classes of Customers", in: *Complementarity and Variational Problems. State of the Art, Proceedings of the 13th International Conference on Complementarity Problems SIAM*, J.-S. Pang and M. Ferris (eds), 234-258, Philadelphia, PA 1997.

[5] C.-C. Lu, H. S. Mahmassani and X. Zhou, "A Bi-Criterion Dynamic User Equilibrium Traffic Assignment Model and Solution Algorithm for Evaluating Dynamic Road Pricing Strategies", *Transportation Research Part C* 16(4), 371-389 (2008).

[6] H. S. Mahmassani, X. Zhou and C.-C. Lu, "Toll Pricing and Heterogeneous Users: Approximation Algorithms for Finding Bi-Criterion Time-Dependent Efficient Paths in Large-Scale Traffic Networks", *Transportation Research Record: Journal of Transportation Research Board* 1923, 28-36 (2005).

[7] R. Jayakrishnan, H. S. Mahmassani and T.-Y. Hu, "An Evaluation Tool for Advanced Traffic Information and Management Systems in Urban Network", *Transportation Research Part C* 2 (3), 129-147 (1994).

[8] C. F. Daganzo and Y. Sheffi, "On Stochastic Models of Traffic Assignment", *Transportation Science* 11, 253-274 (1977).

[9] K. Zhang, H. S. Mahmassani, and C.-C. Lu, "Probit-Based Time-Dependent Stochastic User Equilibrium Traffic Assignment Model", *Transportation Research Record: Journal of the Transportation Research Board* 2085, 86–94 (2008).

An MIP Reverse Logistics Network Model for Product Returns

Nizar Zaarour

Department of Mechanical and Industrial Engineering

Northeastern University

Emanuel Melachrinoudis

Department of Mechanical and Industrial Engineering

Northeastern University

Hokey Min

Department of Management

Bowling Green State University

Marius M. Solomon

Department of Information, Operations, and Analysis

Northeastern University, Huntington Avenue, Boston, USA

E-mail: m.solomon@neu.edu

1 Introduction

The logistics of handling returned products accounts for nearly 1% of the total U.S. gross domestic product ([1]). During the holiday season of 2006, an estimated $13.2 billion in holiday gifts were returned to retailers – more than a third of the $36 billion reverse logistics market in the U.S.

Regardless of product return type, the reverse logistics for product returns often presents unique challenges. Products are returned to initial collection points (e.g., retail stores and designated take-back sites including local convenience stores) in small quantities and thus product returns increase per unit shipping cost due to lack of freight discount opportunities. To create volume, returned products need to be aggregated into larger shipments. However, such aggregation increases product holding time at the initial collection point or a centralized return center which in turn increases inventory carrying costs.

Given this logistics dilemma, our research objectives are to: determine the optimal location of initial collection points (ICPs) and direct customers to designated ICPs in such a way that customer inconvenience is minimized; determine the optimal number of days of holding time for consolidation at each ICP in such a way that total inventory carrying costs are minimized; and determine the optimal location of a centralized return center (CRC) in such a way that total shipping costs (including transshipment cost) are minimized. We will place our work in the context of the ever growing literature on reverse logistics and discuss its contribution.

2 Model Design

In [2], the authors present a nonlinear integer program for solving the multi-echelon reverse logistics problem for product returns. We describe the model below.

Indices:

i = index for customers; $i \in I$

j = index for initial collection points; $j \in J$

k = index for centralized return centers; $k \in K$

Decision Variables:

X_{jk} = volume of products returned from initial collection point j to centralized return center k

$$Y_{ij} = \begin{cases} 1, & \text{if customer } i \text{ is allocated to initial collection point } j \\ 0, & \text{otherwise} \end{cases}$$

$$Z_j = \begin{cases} 1, & \text{if an initial collection point is established at site } j \\ 0, & \text{otherwise} \end{cases}$$

T = length of a collection period (in days) at each initial collection point

$$G_k = \begin{cases} 1, & \text{if a centralized return center is established at site } k, \ (k \in K) \\ 0, & \text{otherwise} \end{cases}$$

Parameters:

a_j = annual cost of renting initial collection point j

b = daily inventory carrying cost per unit

w = annual working days

r_i = volume of products returned by customer i per day

h_j = handling cost of unit product at initial collection point j

c_k = annual cost of establishing and maintaining centralized return center k

m_k = maximum processing capacity of centralized return center k in new returns per day

d_{ij} = distance from customer i to initial collection point j

d_{jk} = distance from collection point j to centralized return center k

l = maximum allowable distance from a given customer to an initial collection point

$\overline{T}$ = maximum length of a collection period (in days) at an ICP. This upper bound is necessary to assure that the return lead time is not too long for the customers

$C_i = \{j \mid d_{ij} \leq l\}$ set of initial collection points that are within distance l from customer i

$D_j = \{i \mid d_{ij} \leq l\}$ set of customers that are within distance l from initial collection point j

$f(X_{jk}, d_{jk}) = E\alpha_{jk}\beta_{jk}$ unit transportation cost between collection point j and return center k where E is the standard freight rate (\$/unit), α_{jk} is the freight discount rate according to the volume of shipment between initial collection point j and centralized return center k, and β_{jk} is the penalty rate applied for the distance between initial collection point j and centralized return center k

$$\alpha_{jk} = \begin{cases} 1 & for \quad X_{jk} \leq P_1 \\ \alpha_1 & for \quad P_1 < X_{jk} \leq P_2 \\ \alpha_2 & for \quad X_{jk} > P_2 \end{cases} \qquad \beta_{jk} = \begin{cases} 1 & for \quad d_{jk} \leq Q_1 \\ \beta_1 & for \quad Q_1 < d_{jk} \leq Q_2 \\ \beta_2 & for \quad d_{jk} > Q_2 \end{cases}$$

Mathematical Formulation:

Minimize

$$\sum_{j \in J} a_j Z_j + bw\left(\frac{T+1}{2}\right)\sum_{i \in I} r_i + w\sum_{i \in I}\sum_{j \in C_i} r_i h_j Y_{ij} + \sum_{k \in K} c_k G_k + \frac{w}{T}\sum_{k \in K}\left(\sum_{j \in J} X_{jk} f(X_{jk}, d_{jk})\right) \quad (1)$$

subject to:

$$\sum_{j \in C_i} Y_{ij} = 1, \qquad \forall i \in I \qquad (2)$$

$$\sum_{i \in D_j} Y_{ij} \leq M\, Z_j, \qquad \forall j \in J \qquad (3)$$

$$T\sum_{i \in D_j} r_i Y_{ij} = \sum_k X_{jk}, \qquad \forall j \in J \qquad (4)$$

$$\sum_j X_{jk} \leq T m_k G_k, \qquad \forall k \in K \qquad (5)$$

$$X_{jk} \geq 0, \qquad \forall j \in J, \ \forall k \in K \qquad (6)$$

$$Y_{ij} \in \{0,1\}, \qquad \forall i \in I, \ \forall j \in J \qquad (7)$$

$$Z_j \in \{0,1\}, \qquad \forall_j \in J \tag{8}$$

$$G_k \in \{0,1\}, \qquad \forall_k \in K \tag{9}$$

The objective function (1) minimizes the total reverse logistics costs, which are comprised of five annual cost components: the cost of renting the ICPs, the cost of establishing and maintaining the CRCs, the handling costs at the ICPs, the inventory carrying cost, and the transportation cost. Constraint (2) assures that a customer is assigned to a single initial collection point. Constraint (3) prevents any return flows from customers to be collected at a closed ICP (M is an arbitrarily set big number). Constraint (4) makes the incoming flow equal to the outgoing flow at each initial collection point. Constraint (5) ensures that the total volume of products shipped from initial collection points to a centralized return center does not exceed the maximum capacity of the centralized return center. Constraint (6) preserves the non-negativity of decision variables X_{jk}. Constraints (7) – (9) declare decision variables Y_{ij}, Z_j and G_k as binary.

Given the inherent computational complexity of the non-linear program, the authors utilized a genetic algorithm to solve small-sized problems. To overcome these shortcomings and those of other prior studies, we will show how to linearize the objective function and solve much larger problems optimally. We will also report on the extensive computational results we performed and sensitivity analyses we conducted by varying return rates and assessing their impacts on optimal collection periods and the total reverse logistics costs. For certain special structures we determine the optimal tradeoff between inventory and shipping costs and develop closed form solutions.

References

[1] R. Gecker, "Industry best practices in reverse logistics", Unpublished White Paper, Aberdeen Group, 2007.

[2] H. Min, H.K. Ko, and C.S Ko, "A genetic algorithm approach to developing the multi-echelon reverse logistics network for product returns," *Omega*, 34, 56-59 (2006).

TRISTAN VII – Author index