The route-first cluster-second principle in vehicle routing

Christian PRINS

Institute Charles Delaunay,
University of Technology of Troyes (UTT)
France

christian.prins@utt.fr

VIP 2008, Oslo, 12-14/06/2008

Outline

1. The route-first cluster-second principle
2. A few examples of constructive heuristics
3. Use in metaheuristics (mainly GA)
4. Two less obvious applications

Part 1

The route-first cluster-second principle

Two strategies for VRP heuristics

Cluster-first route-second heuristics are well known:

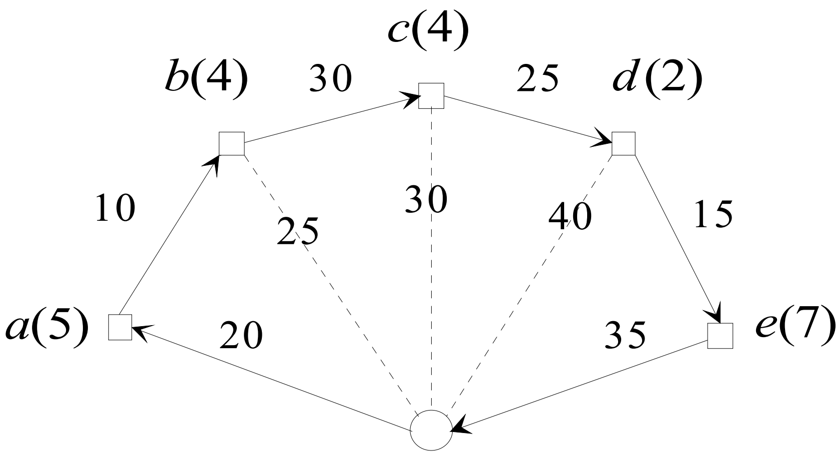
- build clusters of clients and solve one TSP per cluster
- sweep heuristic, Gillett and Miller (1974)
- heuristic of Fisher and Jaikumar (1984).

Route-first cluster-second methods are seldom used:

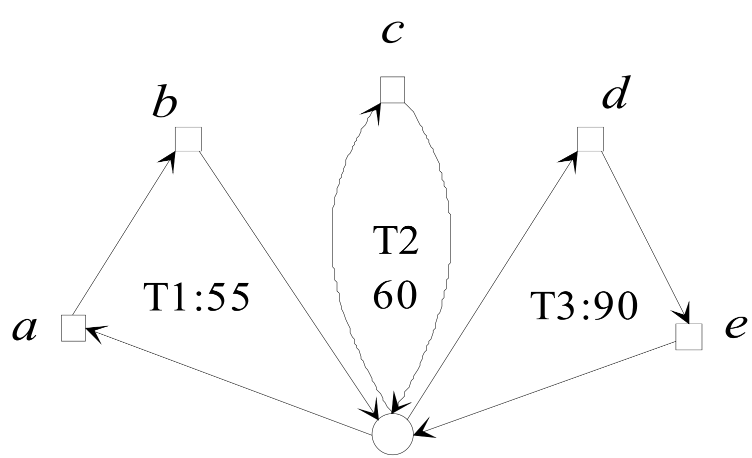
- relax vehicle capacity to build a "giant tour" (TSP tour)
- then split the giant tour into feasible trips
- proposed by Beasley (1983), without numerical results
- used by Ulusoy (1985) for the CARP, on one instance.

No comparison with other VRP or CARP heuristics.

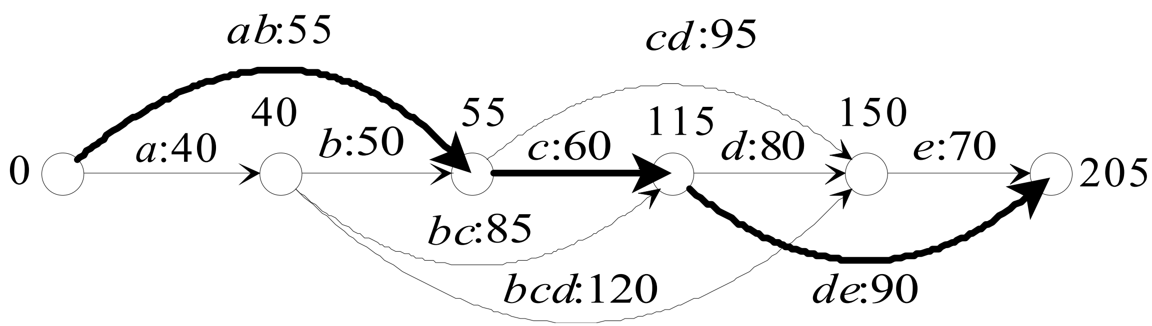
Basic splitting procedure (Split)



Giant tour T = (a,b,c,d,e)



Optimal splitting, cost 205



Auxiliary graph of possible trips for W=10 and shortest path in boldface

Remarks

Properties:

- Split is optimal, subject to the order defined by *T*.
- $O(np)$, p average length of feasible subsequences.

Examples of use:

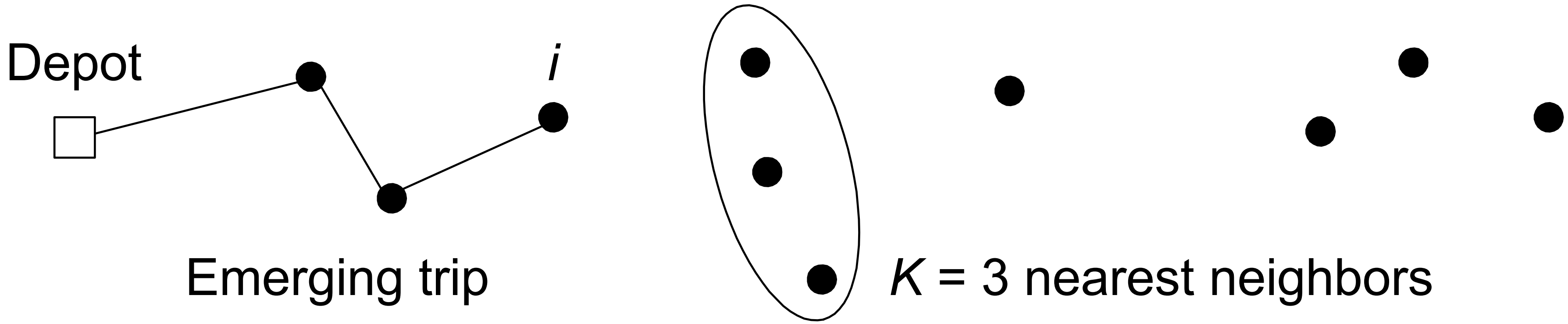
- constructive heuristics: run any algorithm for the TSP or Rural Postman Problem and apply Split.
- randomized heuristics: randomize giant tour construction to get several tours and split them.
- metaheuristics: search the space of giant tours and evaluate them using Split.

Part 2

A few examples of route-first cluster-second heuristics

Randomized giant tour 1/2

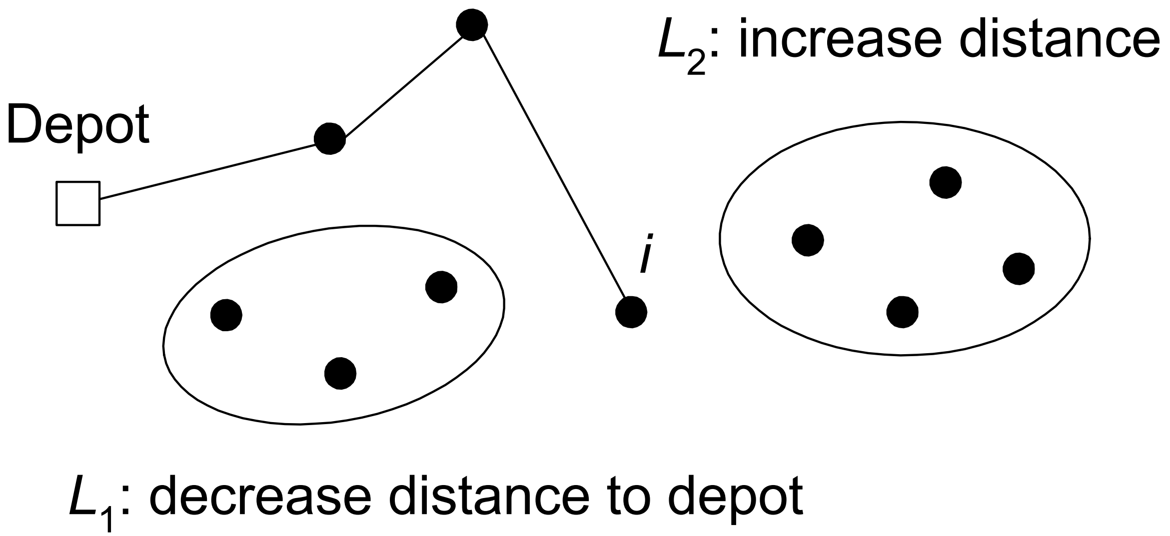
Nearest Neighbor (NN) randomized:

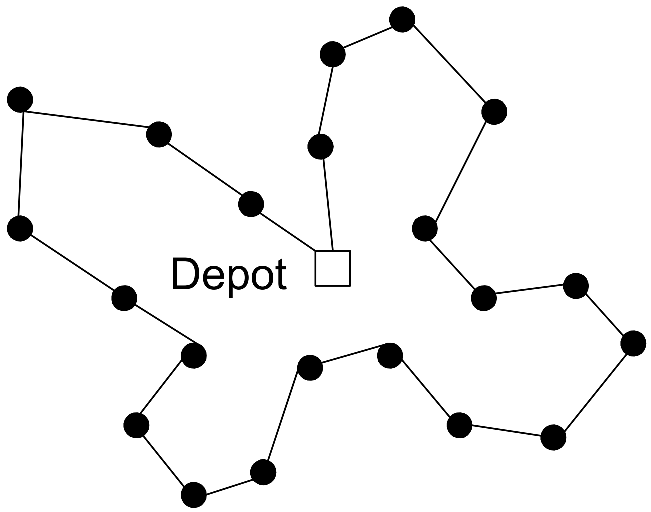


Draw the next customer among the *K* nearest customers.

Randomized giant tour 2/2

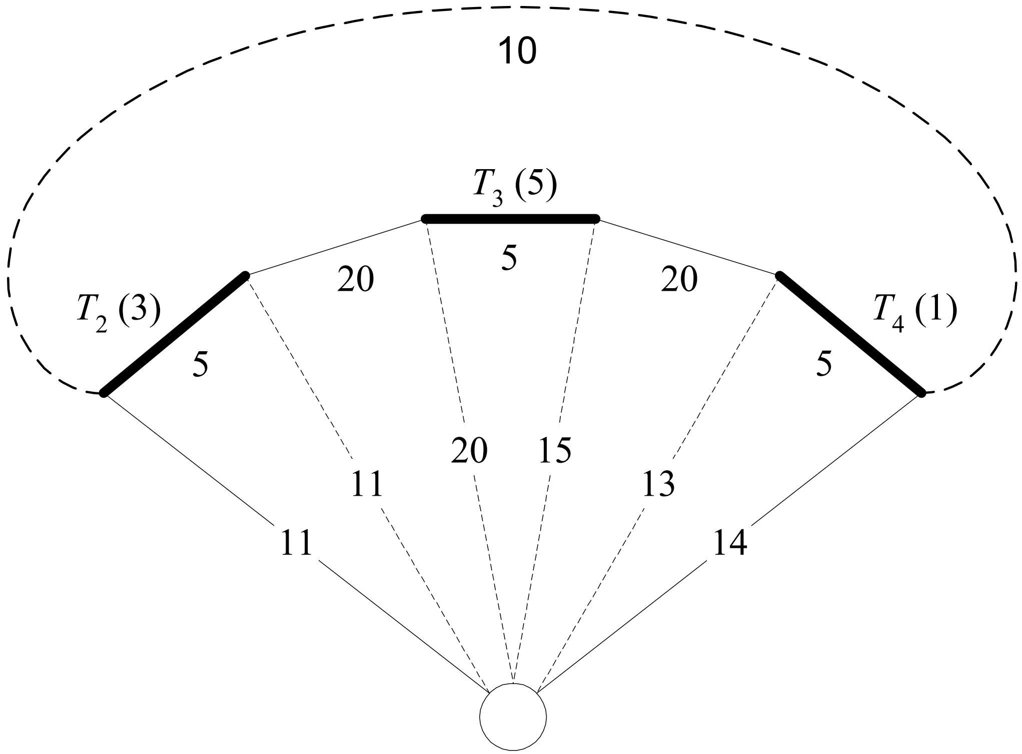
Nearest Neighbor randomized, "Flower" version (NNF):





If *load* $\in [k \cdot Q,(k+0.5) \cdot Q]$ then draw in L_2 else in L_1.
Higher probability to cut T when close to depot.

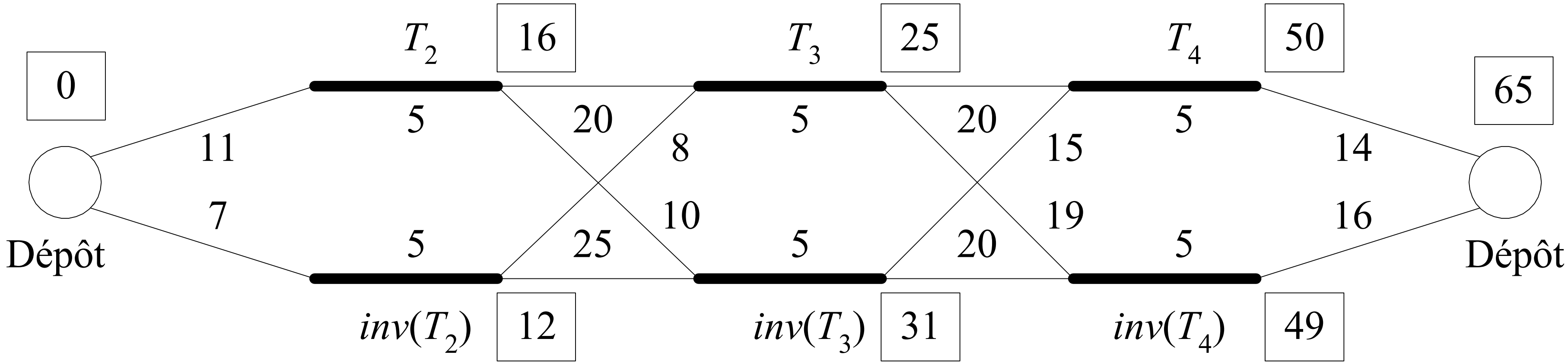
Split with shifts (rotations) or Split-S



(T_2, T_3, T_4): cost 80, (T_3, T_4, T_2): cost 76, (T_4, T_2, T_3): cost 73.
The cost of arc (1,4) in the auxiliary graph is 73.
If (1,4) is on the shortest path, the trip will be (T_4, T_2, T_3).

Split with flips or Split-F

Find the best edge directions for each subsequence:



Inv(T_k) is the other direction for edge T_k.
(T_2, T_3, T_4): cost 80, (*inv*(T_2), T_3, *inv*(T_4)): cost 65.

Other improved versions of Split

Split-S and Split-F are in $O(nv)$ like the basic Split.
It is possible to combine shifts and flips: Split-SF.

Iterative versions of Split-S, F and SF are also possible.
Example for Split-SI:

$S \leftarrow$ *Split-S* (T)
repeat
    concatenate the trips of S (with rotations), giving T'
    $S' \leftarrow$ *Split-S* (T')
    if *cost* $(S') <$ *cost* (S) then $S := S'$ endif
until *cost* $(S') \geq$ *cost* (S).

Examples on CARP (to appear in IJPR)

23 gdb instances, 20 random tours, times < 0.01s at 1.8 GHz.
PS: Path-Scanning, Golden et al. (1983)
AM: Augment-Merge, Golden & Wong (1981)

Giant tour NN	PS	AM	Version of Split			
			Basic	Shift	Flip	SFI
Avg. dev. opt %	10.8	7.2	4.4	3.8	4.1	2.9
Worst dev. %	33.1	24.2	17.2	15.5	17.2	13.2
Nb of optima	2	2	3	3	3	7

Giant tour NNF	PS	AM	Version of Split			
			Basic	Shift	Flip	SFI
Avg. dev. opt %	10.8	7.2	3.5	2.7	3.3	2.3
Worst dev. %	33.1	24.2	14.2	13.9	13.2	11.2
Nb of optima	2	2	4	5	5	8

Part 3

Use of splitting procedures in metaheuristics

Use in memetic algorithms 1/2

Principle:

- each chromosome is encoded as a giant tour *T*.
- Split extracts the best VRP solution, subject to *T*.

Advantages:

- classical crossovers for the TSP can be reused.
- no repair procedure.

No loss of information:

- the MA explores the smaller space of giant tours
- Split evaluates each giant tour optimally
- *there exists one optimal giant tour.*

Use in memetic algorithms 2/2

Problem	Reference
VRP	Prins, *Comput. Oper. Res.*, 2004
CARP	Lacomme, Prins, Ramdane-Chérif, *Annals of OR*, 2004
Mixed CARP	Belenguer, Benavent, Lacomme, Prins, *Comput. Oper. Res.*, 2006
Capacitated GRP: required nodes, arcs and edges	Prins, Bouchenoua, *Recent advances in memetic algorithms*, Springer, 2004

Additional constraints 1/2

Adding constraints can affect 3 steps in Split:

let T be a giant tour with n customers
for each subsequence $(T_i, T_{i+1}, \ldots, T_k)$ *do*
    if feasible *then*
        add arc $(i\text{-}1,k)$ to the auxiliary graph H
        compute its cost $Z_{i\text{-}1,k}$
    endif
endfor
compute a shortest path from node 0 to node n in H

The shortest path computation is rarely affected.
In general, the complexity can be preserved.

Additional constraints 2/2

VRP. Feasibility: discard trips with loads > Q.

Distance constraint. Feasibility: discard trips of length > L.

VRP with Time Windows (VRPTW):

- feasibility: discard trips which violate time windows.
- arc costs : add waiting times.

Vehicle Fleet Mix Problem (VFMP):

- p vehicle types, type t has a capacity Q_t and a fixed cost F_t
- feasibility: discard trips with loads > Q_{max}
- arc costs : add F_k (k cheapest type with enough capacity).

Limited fleet size K:

- shortest path: compute a shortest path with at most K arcs.

Use in other metaheuristics

GRASP:

- generate giant tours using a randomized heuristic,
- apply Split and then a local search to the solution.

Tests on the CARP, 1000 iterations (to appear in IJPR):

- simpler than existing metaheuristics,
- not better than the MA of Lacomme et al. (2004),
- but better than the tabu search of Hertz et al. (2000),
- and 10 times faster.

Good results on the CARP with Time Windows (CARPTW):

to appear in "*Advances in evolutionary computation for trans-portation and logistics*", A. Fink & F. Rothlauf (eds), Springer.

Use in other metaheuristics

New Iterated Local Search (ILS) for the VRP (simplified):

compute one initial giant tour T
$S \leftarrow Split\ (T)$
for iter := 1 to maxiter do
    $T' \leftarrow Mutate\ (T)$
    $S' \leftarrow Split\ (T')$
    $Local\_Search\ (S')$
    if $cost(S') < cost(S)$ then
        $S \leftarrow S'$
        $T \leftarrow Concat\_Trips\ (S)$
    endif
endfor

Alternation between giant tours and complete solutions!

Results on Christofides instances 1/2

14 instances with 50 to 199 customers.

In Cordeau et al., "New heuristics for the VRP" (2005),
4 methods < 0.3% to best-known solutions with one run:

- AGES "best" and "fast", Mester & Bräysy (2007).
- Bone Route, Tarantilis and Kiranoudis (2002).
- SEPAS, Tarantilis, 2005.
- MA, Lacomme et al., 2004.

Methods with 10 runs (discarded):

- Reimann et al. (2004), 0.15% but 0.48% if one run.
- Pisinger & Röpke (2007), 0.11% but 0.31% if one run.

Instances of Christofides et al. 2/2

Method	AGES best	ILS	AGES fast	Bone Route	SEPAS	MA
Dev. BKS%	0.027	0.071	0.084	0.183	0.196	0.236
BKS found	13	10	10	11	9	8
Time (s)	163	16	3	62	67	154

Times scaled for a 2.8 GHz PC.

To appear in: "*Bio-inspired algorithms for the VRP*",
F. Pereira and J. Tavares (eds), Springer.

Instances of Golden et al.

20 instances with 200 to 483 customers.
3 methods < 1% to best-known solutions, with one run.

Method	AGES best	ILS	SEPAS	AGES fast
Dev. BKS%	0.013	0.315	0.615	0.914
BKS found	17	4+1	2	1
Time (s)	1461	436	538	13
Parameters	12	4	8	12

This ILS becomes the second best metaheuristic for the VRP.
Using several settings of parameters, it improves 2 BKS.

Part 4

Two less obvious applications

The Periodic CARP 1/3

Data. One CARP instance plus:

- planning horizon *H* of *p* days
- for each task *e*, demand *q*(*e*), frequency *f*(*e*),
- set of allowed day combinations *comb*(*e*).

Goal (hierarchical bi-objective function):

- select *f*(*e*) days for each task *e*,
- solve one CARP per day
- main objective: minimize fleet size
- secondary objective: total duration of trips over *H*

The PCARP 2/3

Chromosome *T*:

- *p* sublists *T*(1)…*T*(*p*): one giant tour per day
- edge *e* occurs *f*(*e*) times, using one day combination
- node *e* occurs at most once in each giant tour *T*(*k*).

e	*f*(*e*)	*comb*(*e*)	*e*	*f*(*e*)	*comb*(*e*)
1	3	{1,3,4},{2,3,4}	5	1	{1},{2}
2	2	{1,3},{2,4}	6	3	{2,3,4},{1,2,4},{1,2,3}
3	1	{1},{4}	7	2	{2,4},{1,3}
4	1	{1}	8	1	{2},{4}

Mon				**Tue**			**Wed**			**Thu**			
1	2	3	4	5	6	7	1	2	6	1	6	7	8

The PCARP 3/3

Hierarchical bi-objective function *Z=M.nvu+tcost*, where:

- *nvu*, number of vehicles used (fleet size required)
- *tcost*, total duration of trips over horizon *H*

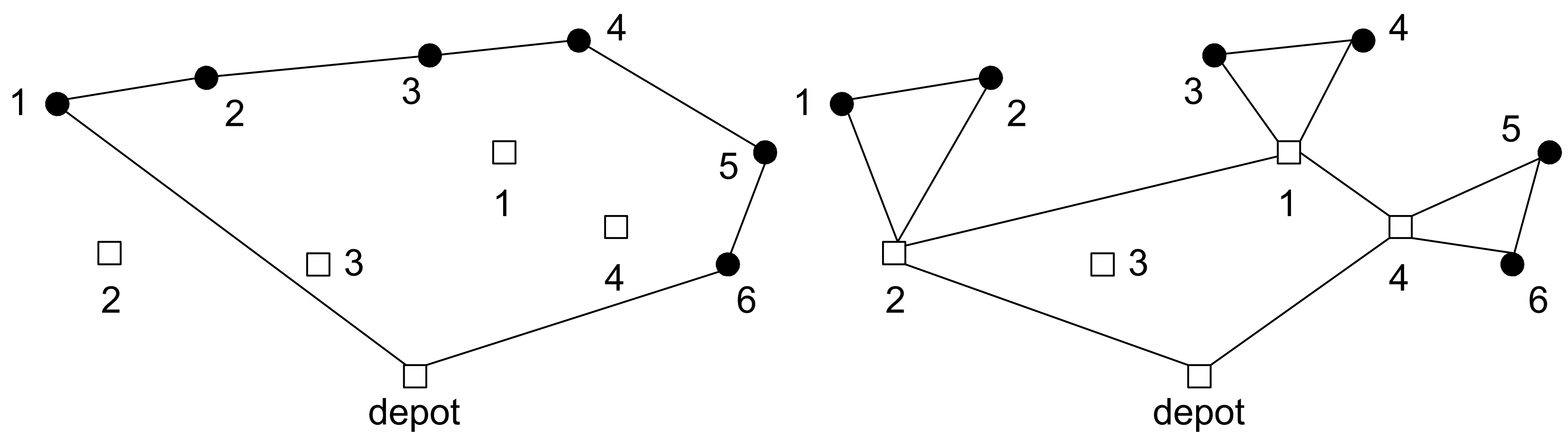
Optimal chromosome evaluation:

1. Split in each day *p* with arc costs = 1 in auxiliary graph → minimum nb of vehicles per day *nv*(*p*).
2. Minimum fleet size *nvu* = maximum of the *nv*(*p*).
3. Split on each day *p* using actual trip costs and at most *nvu* arcs → minimum cost per day *cost*(*p*)
4. Return *Z* = *M.nvu* + sum of the *cost*(*p*).

A truck and trailer problem 1/3

Problem met in milk collection, Argentina.
A truck with a tank + a trailer with another tank is used.
Most farms cannot be reached with the trailer.

Data: a giant tour for *n* farms + *p* trailer-sites (squares).
Goal: insert trailer-sites to get an optimal solution.



A truck and trailer problem 2/3

Consider a giant tour $T=(1,2,\ldots,n)$, w.l.o.g.

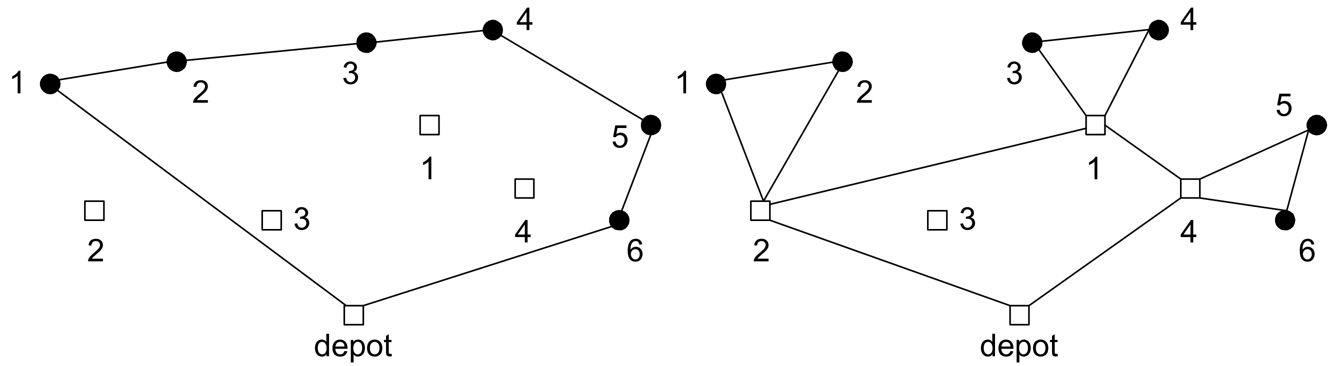
Splitting graph H with $np+2$ nodes and np^2v arcs:

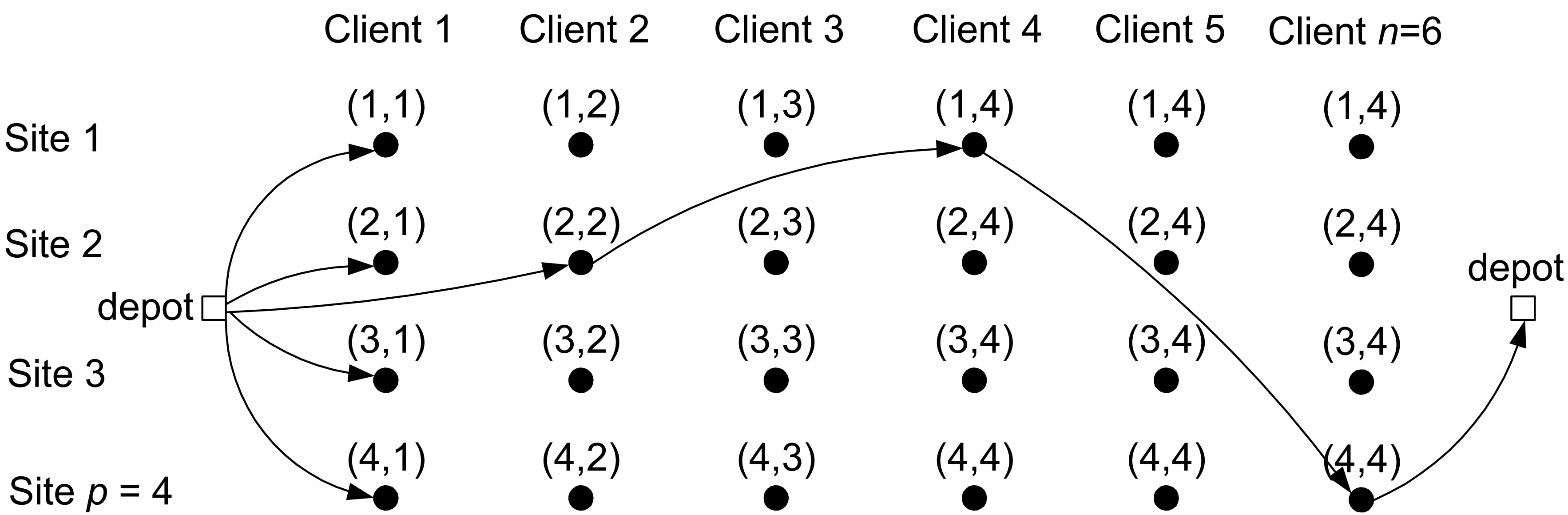
- node (i,j): models a trip ending with customer j and site i.
- arc $((i,j),(a,b))$: take trailer at site i, go to site a, detach trailer, visit farms $j+1, j+2, \ldots b$ and return to site a.
- arc cost:

$$d_{ia} + d_{a,j+1} + \sum_{k=j+1}^{b-1} d_{k,k+1} + d_{ba}$$

- compute a shortest path in $O(np^2v)$.

A truck and trailer problem 3/3





Concluding remarks

The route-first cluster-second principle is general and flexible.

It can be used to design fast and effective constructive heuristics and metaheuristics.

However, it reaches its limits when the underlying shortest path problem is no longer polynomial, e.g., in the Heterogeneous Fleet VRP.