Synchronized vehicle routing

David Bredström & Mikael Rönnqvist

Norges Handelshøyskole
Norwegian School of Economics and Business Administration

Literature reference

- This presentation:
  - D. Bredström and M. Rönnqvist, Routing and scheduling with synchronization constraint, *European Journal of Operational Research*, Vol. 191, pp. 19-29, 2008.
  - D. Bredström and M. Rönnqvist, A Branch and Price Algorithm for the Combined Vehicle Routing and Scheduling Problem With Synchronization Constraints, *Scandinavian Working Papers in Economics*, *NHH Discussion Paper 07/2007, 2007*.
- Application – home care:
  - P. Eveborn, M. Rönnqvist, M. Almroth, M. Eklund, H. Einarsdóttir and K. Lidèn, Operations Research (O.R.) Improves Quality and Efficiency in Home Care, to appear in special issue in *Interfaces* from Franz Edelman finalists

Outline

- Applications with synchronization restrictions
- Standard VRP approach and extension with synchronization
- Heuristic solution method and experiments
- Set partitioning approach, Branch & Price method and experiments
- Concluding remarks

Two applications with synchronization constraints

- - Home care routing/ scheduling
- - Harvest & forward operations

Home Care in Sweden

- By law, the local authorities have to provide visiting services to allow older people to continue living independently at home
- Wide range of services, from cleaning to medical care
- Sector employs 80,000 people, about 2% of Sweden’s total workforce
- Fast growing sector due to ageing population

Daily planning problem

- *Address (location)*
- *Gender*
- *Language*

Assignment (scheduling & routing)

Visit

- *Service (medical etc..*
- *Care time*
- *Time windows*

- *Availability*
- *Working hours*
- *Competence/ skills*

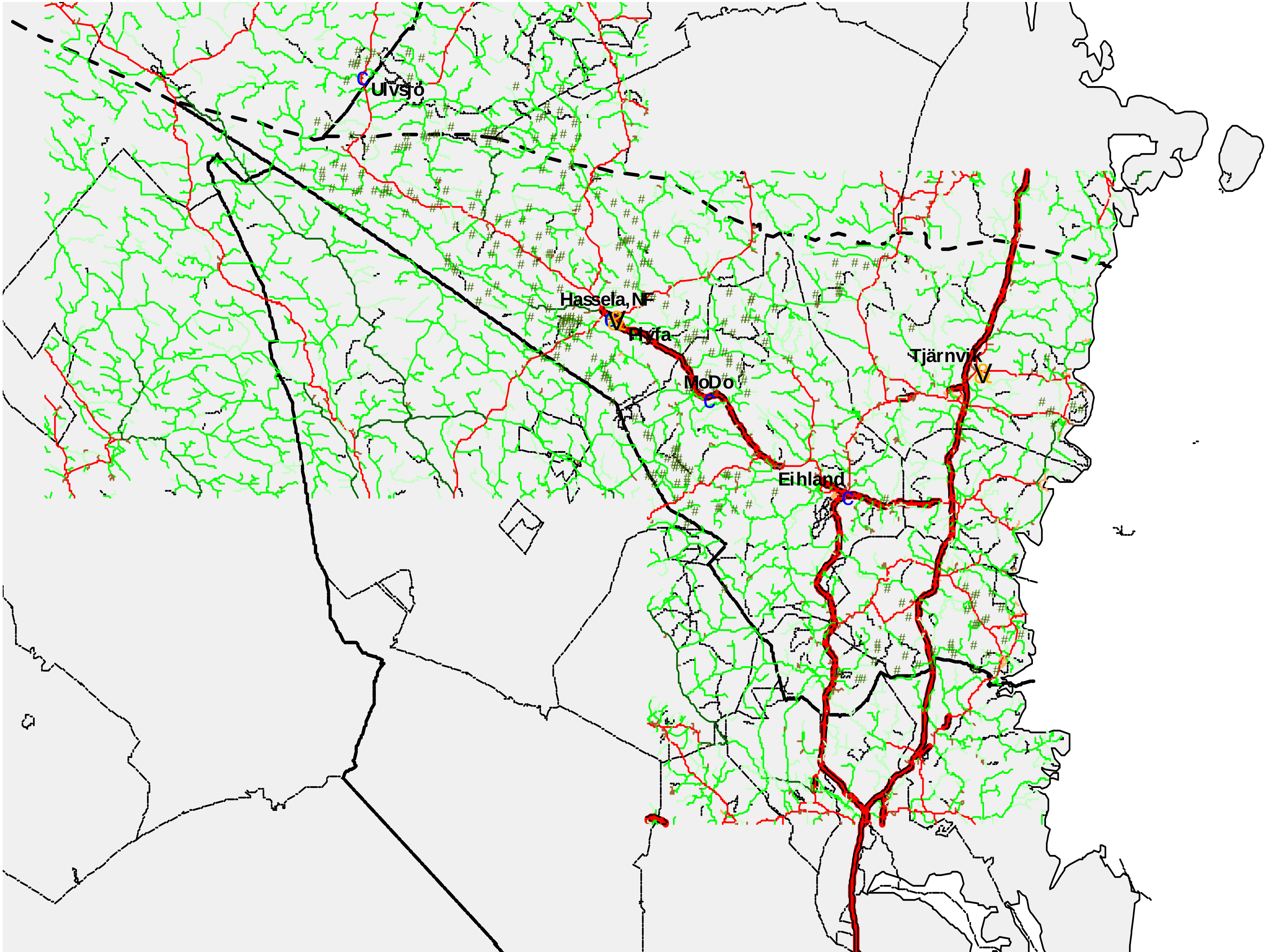
Problem in OR terms

- Decisions
  - Allocation of visits to home care workers
  - Routing of workers
- Constraints
  - Skills, Time windows (short and wide time windows)
  - Working hours, travel time/ breaks
  - Synchronisation
    - *Synchronized visits (double staffing)*
    - *Precedence relations of visits (at the same elderly)*
- Objective
  - Short and long term continuity, Route cost/ time,
  - Fairness, Preferences

Laps Care system in City of Stockholm

- In practice locally since 2003
- Full scale implementation 2008
  - 800 Planning Officers are involved
  - All Home Care Units, about 15000 workers participate
  - 40 000 Elderly Customers enjoy the benefit
- Large scale solutions
  - E-learning programs
  - Centralized database
  - Interconnected systems to ensure information flow

Harvest and forwarding units



Ulvsjö
Hassela, NF
Plyfa
MoDo
Tjärnvik
Eihland

Harvest, forward and harward units

Standard VRP approach

Problem formulation

K : set of vehicles

$G(\overline{N}, A)$: directed graph

N : set of nodes to be visited

$\overline{N}$: set of nodes to be visited + depot

A : set of arcs

D_i : duration for visit i

$[a_i, b_i]$: time window for visit node i

$[a_i^k, b_i^k]$: time window for vehicle k (depot start, depot end)

T_{ij} : Travelling time between node i and j

$P^{sync} \subset NxN$: pairwise synchronized visits

$P^{prec} \subset NxN$: pairwise precedence constraints (S_{ij}: off set)

MIP formulation – variables

$$x_{ijk} = \begin{cases} 1, \text{ if vehicle } k \in K \text{ uses arc } (i,j) \in A \\ 0, \text{ otherwise} \end{cases}$$

$$t_{ik} = \text{time when vehicle } k \text{ arrives to node } i$$

$$(0 \text{ if no visit})$$

$$\sum_{k\in K}\sum_{j:(i,j)\in A} x_{ijk} = 1 \quad \forall i \in N, \tag{1}$$

$$\sum_{j:(o,j)\in A} x_{ojk} = \sum_{j:(j,d)\in A} x_{jdk} = 1 \quad \forall k \in K, \tag{2}$$

$$\sum_{j:(i,j)\in A} x_{ijk} - \sum_{j:(j,i)\in A} x_{jik} = 0 \quad \forall i \in N \;\; \forall k \in K, \tag{3}$$

$$t_{ik} + (T_{ij} + D_i)x_{ijk} \leqslant t_{jk} + b_i(1 - x_{ijk}) \quad \forall k \in K$$
$$\forall (i,j) \in A, \tag{4}$$

$$a_i \sum_{j:(i,j)\in A} x_{ijk} \leqslant t_{ik} \leqslant b_i \sum_{j:(i,j)\in A} x_{ijk} \quad \forall k \in K \;\; \forall i \in N, \tag{5}$$

$$a_i^k \leqslant t_{ik} \leqslant b_i^k \quad \forall k \in K \;\; \forall i \in \{o, d\}. \tag{6}$$

Additional synchronization constraints

$$\sum_{k \in K} t_{ik} = \sum_{k \in K} t_{jk} \quad \forall (i, j) \in P^{\mathrm{sync}}, \tag{7}$$

$$\sum_{k \in K} t_{ik} \leqslant S_{ij} + \sum_{k \in K} t_{jk} \quad \forall (i, j) \in P^{\mathrm{prec}}. \tag{8}$$

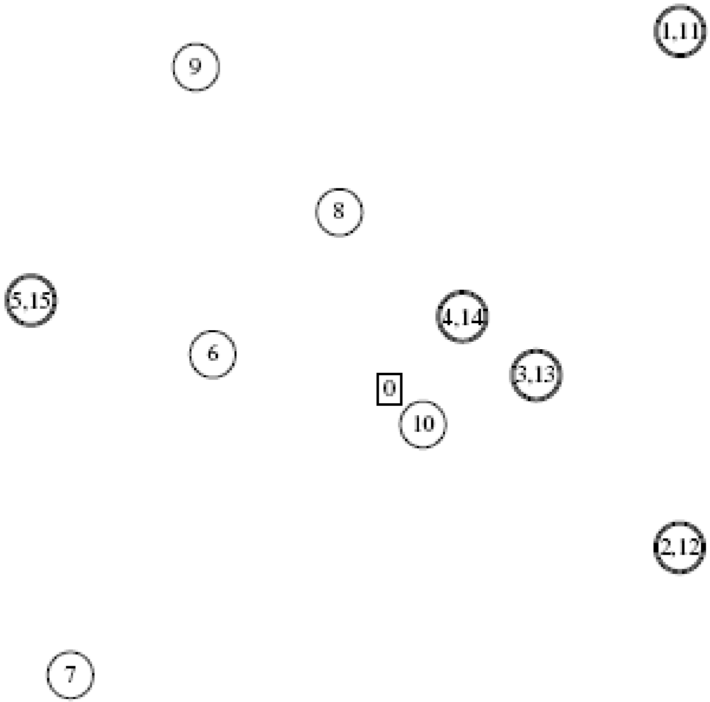
Objective function

Measuring difference between pair of vehicles

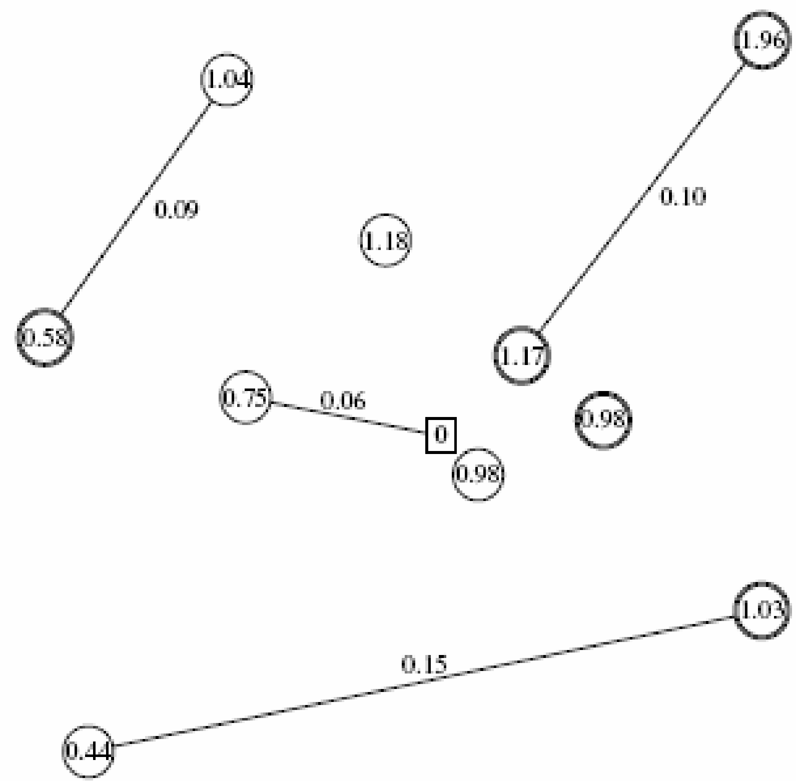
$$\sum_{(i,j)\in A} W_{ijk_1} x_{ijk_1} - \sum_{(i,j)\in A} W_{ijk_2} x_{ijk_2} \leqslant w \quad \forall k_1 \in K \ \forall k_2 \in K \setminus \{k_1\}. \tag{9}$$

Balance between preference, travel time and balancing

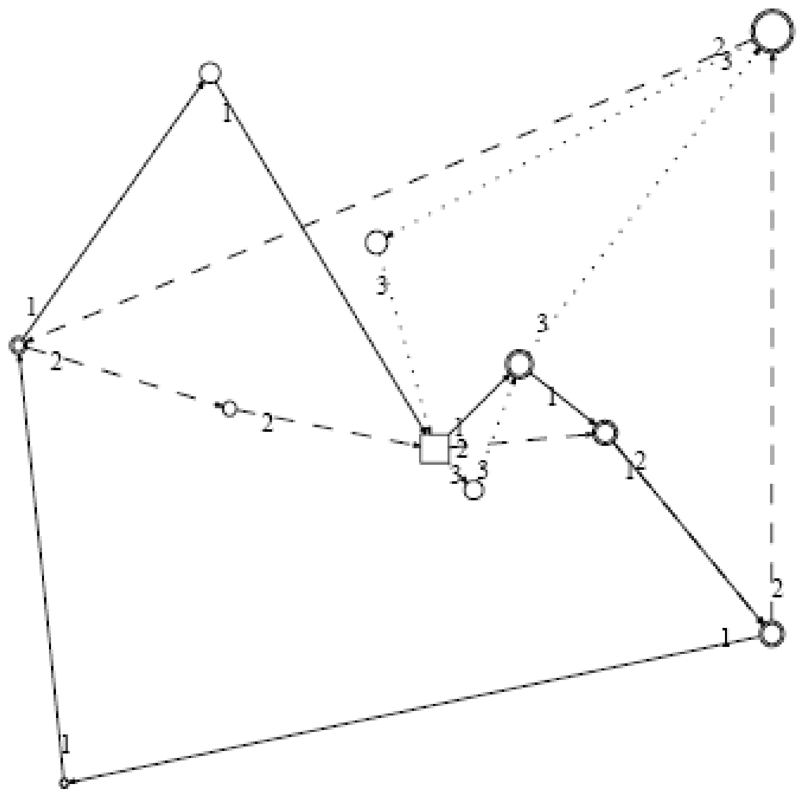
$$\begin{aligned} \min \quad & \alpha_P \sum_{k\in K} \sum_{(i,j)\in A} c_{ik} x_{ijk} + \alpha_T \sum_{k\in K} \sum_{(i,j)\in A} T_{ij} x_{ijk} + \alpha_B w, \\ \text{s.t.} \quad & (1)\text{–}(9), \end{aligned}$$



(a) The network.



(b) Service durations.



(a) The optimal solution when minimizing the sum of traveling distance and work load for 3 vehicles.

Schedules for the three vehicles in the first example

	Arrival	Duration	Traveling	Waiting
Vehicle 1				
Depot	0.00	0.00	0.04	0.99
4, 14	1.03	1.17	0.04	0.00
3, 13	2.24	0.98	0.06	0.00
2, 12	3.28	1.03	0.15	2.14
7	6.60	0.44	0.13	0.00
5, 15	7.17	0.58	0.09	0.00
9	7.84	1.04	0.11	0.00
Depot	9.00	0.00		
	Sum	5.25	0.62	3.13
Vehicle 2				
Depot	0.00	0.00	0.05	2.19
3, 13	2.24	0.98	0.06	0.00
2, 12	3.28	1.03	0.14	0.55
1, 11	5.00	1.96	0.22	0.00
5, 15	7.17	0.58	0.06	0.37
6	8.19	0.75	0.06	0.00
Depot	9.00	0.00		
	Sum	5.30	0.60	3.10
Vehicle 3				
Depot	0.00	0.00	0.01	0.00
10	0.01	0.98	0.04	0.00
4, 14	1.03	1.17	0.10	2.69
1, 11	5.00	1.96	0.11	0.70
8	7.77	1.18	0.05	0.00
Depot	9.00	0.00		
	Sum	5.29	0.92	3.39

Test instances

Instance	$\lvert N\rvert$	$\lvert K\rvert$	$\lvert P^{sync}\rvert$	$\sum D_i/\lvert K\rvert$	AvTD	Time windows (hour)				
						F	S	M	L	A
1	20	4	2	4.9	0.22	0	1.5	2.1	2.9	9
2	20	4	2	4.2	0.20	0	1.7	2.2	3.0	9
3	20	4	2	5.3	0.21	0	1.5	2.4	3.0	9
4	20	4	2	5.9	0.29	0	1.8	2.9	3.9	9
5	20	4	2	5.0	0.21	0	1.3	2.1	3.2	9
6	50	10	5	4.7	0.25	0	1.4	2.3	3.1	9
7	50	10	5	5.0	0.23	0	1.6	2.5	3.4	9
8	50	10	5	6.2	0.23	0	1.5	2.4	3.2	9
9	80	16	8	6.1	0.21	0	1.5	2.3	2.9	9
10	80	16	8	5.1	0.17	0	1.6	2.6	3.6	9

Time windows: F:fixed, S:small, M: medium, L: large, A: no restriction

Instance 1-5: 1,900 variables, 2,100 constraints
Instance 6-8: 27,000 variables, 28,000 constraints
Instance 9-10: 106,000 variables, 109,000 constraints

Heuristic approach – idea: keep MIP small to reduce B&B tree

- Step 1: Decide Association *Y*
  - *Y*: vehicles *k* allowed to visit node *i*
- Step 2: Solve LP-relaxation with variables defined through *Y* → arc set <u>*A*</u> used
- Step 3: Solve MIP over *Y* and <u>*A*</u>
- Step 4: Repeat the following step for fixed time
  - Every r iteration, reduce *Y* and <u>*A*</u>
  - Randomly extend *Y* and <u>*A*</u>
  - Solve MIP over *Y* and <u>*A*</u>

Solutions proved optimal are marked in bold

Instance	TW	Preferences		Traveling time		Fairness	
		BK (hour)	H 2 min (hour)	BK (hour)	H 2 min (hour)	BK (hour)	H 2 min (hour)
1	F	**−96.45**	**−96.45**	**5.13**	**5.13**	**0.117**	**0.117**
1	S	**−114.03**	**−114.03**	**3.55**	**3.55**	0.026	0.052
1	M	**−117.80**	**−117.80**	3.55	3.55	0.026	0.026
1	L	**−118.51**	**−118.51**	3.44	3.39	0.026	0.026
1	A	−118.51	−116.37	3.16	3.69	**0.000**	0.026
2	F	**−85.26**	**−85.26**	**4.98**	**4.98**	**0.037**	**0.037**
2	S	**−92.09**	**−92.09**	**4.27**	**4.27**	0.025	0.025
2	M	−104.81	−102.63	3.58	3.58	0.025	0.025
2	L	−104.81	−106.06	3.58	3.42	0.012	0.025
2	A	−117.24	−117.24	3.58	3.34	0.012	0.012
3	F	**−56.70**	**−56.70**	**5.19**	**5.19**	**0.154**	**0.154**
3	S	**−99.49**	**−99.49**	**3.63**	**3.63**	0.064	0.064
3	M	**−106.59**	**−106.59**	3.41	3.33	0.038	0.064
3	L	**−107.87**	−104.72	3.29	3.29	0.038	0.013
3	A	−111.29	−92.22	3.1	3.28	0.038	0.026
4	F	**−63.08**	**−63.08**	**7.21**	**7.21**	**0.942**	**0.942**
4	S	**−100.00**	−99.43	**6.14**	6.69	0.130	0.162
4	M	−105.42	−105.42	5.91	5.75	0.130	0.049
4	L	−105.42	−96.96	5.83	5.3	0.081	0.032
4	A	−105.42	−92.78	5.23	4.91	0.032	0.065
5	F	**−62.59**	**−62.59**	**5.37**	**5.37**	**0.201**	**0.201**
5	S	**−76.29**	**−76.29**	**3.93**	**3.93**	0.063	0.038
5	M	**−76.29**	**−76.29**	**3.53**	**3.53**	0.038	0.063
5	L	**−84.21**	**−84.21**	3.43	3.34	0.025	0.025
5	A	−84.21	−43.74	3.26	3.45	0.025	0.038

Heuristic vs optimization

The average, maximum and minimum objective function values from 20 runs of the heuristic (H 2 min) compared to the solution obtained from OPT after 2, 10, 30 and 60 minutes

	Instance				
	1	2	3	4	5
2 min	4.70	5.02	5.49	–	4.08
10 min	4.47	4.30	3.63	6.61	3.70
30 min	4.40	4.30	3.63	6.61	3.70
60 min	4.04	4.20	3.63	6.61	3.70
H 2 min, aver	4.29	4.09	3.94	6.41	3.83
H 2 min, max	4.88	4.39	4.38	7.77	4.33
H 2 min, min	4.01	3.96	3.63	5.88	3.70

Impact of synchronization

Solutions when the synchronized visits are fixed to time windows' midpoints, synchronized with time windows, and where the synchronization constraints are relaxed

Instance	Fix		Sync		No Sync	
	Obj (hour)	AvWT (%)	Obj (hour)	AvWT (%)	Obj (hour)	AvWT (%)
6	11.97	64	11.87	64	10.59	62
7	14.16	71	11.52	68	12.97	69
8	–	–	15.16	84	13.78	83
9	–	–	20.68	81	19.29	80
10	17.69	68	17.61	68	16.35	67

Impact of time window size

Solutions for the different time windows

Instance	S		M		L		A	
	Obj	AvWT	Obj	AvWT	Obj	AvWT	Obj	AvWT
6	13.69	66	12.80	65	11.87	64	11.88	64
7	15.06	72	13.45	70	11.52	68	12.41	69
8	–	–	–	–	15.16	84	13.01	82
9	–	–	–	–	20.68	81	22.89	81
10	16.24	67	15.33	67	17.61	68	17.59	67

Set partitioning approach with Branch & Price algorithm

$$[P] \quad \min \sum_{k \in K} \sum_{(i,j) \in A} (C_{ik}^{Prefs} + C_{ij}^{Time}) x_{ijk} \tag{1}$$

$s.t.$

$$\sum_{k \in K} \sum_{j:(i,j) \in A} x_{ijk} = 1 \quad \forall i \in N \tag{2}$$

$$\sum_{j:(o,j) \in A} x_{ojk} = \sum_{j:(j,d) \in A} x_{jdk} = 1 \quad \forall k \in K \tag{3}$$

$$\sum_{j:(i,j) \in A} x_{ijk} - \sum_{j:(j,i) \in A} x_{jik} = 0 \quad i \in N \; \forall k \in K \tag{4}$$

$$t_{ik} + (T_{ij} + D_i) x_{ijk} \leq t_{jk} + b_i (1 - x_{ijk}) \quad \forall k \in K \; \forall (i,j) \in A \tag{5}$$

$$a_i \sum_{j:(i,j) \in A} x_{ijk} \leq t_{ik} \leq b_i \sum_{j:(i,j) \in A} x_{ijk} \quad \forall k \in K \; \forall i \in N \tag{6}$$

$$a_i^k \leq t_{ik} \leq b_i^k \quad \forall k \in K \; \forall i \in \{o, d\} \tag{7}$$

$$\sum_{k \in K} t_{i_1 k} = \sum_{k \in K} t_{i_2 k} \quad \forall (i_1, i_2) \in P^{sync} \tag{8}$$

$$x_{ijk} \in \{0, 1\} \quad \forall k \in K \; \forall (i,j) \in A \tag{9}$$

We let $(A_{ij}, R_{ij})_{i \in N}$ denote a schedule $j \in J$, where $A_{ij} = 1$ when schedule j includes a service of customer i at the time R_{ij}, and $A_{ij} = R_{ij} = 0$ otherwise. If J_k is the subset of schedules feasible for vehicle k at the cost c_{jk}, the side constrained set partitioning formulation of [P] is as follows.

$$[SCSP] \quad f^*_{SCSP} = \min \sum_{k \in K} \sum_{j \in J_k} c_{kj} z_{kj}$$

$$s.t.$$

$$\sum_{j \in J_k} z_{kj} = 1 \quad \forall k \in K$$

$$\sum_{k \in K} \sum_{j \in J_k} A_{ij} z_{kj} = 1 \quad \forall i \in N$$

$$\sum_{k \in K} \sum_{j \in J_k} (R_{i_1 j} - R_{i_2 j}) z_{kj} = 0 \quad (i_1, i_2) \in P^{sync}$$

$$z_{kj} \in \{0, 1\} \quad \forall k \in K \, \forall j \in J_k$$

Solution approach

- SCSP: Side constrained set partitioning
- SP: relaxation of SCCP with constraint (13) relaxed
- We aim to solve SCSP with a branch & price algorithm using the LP relaxation of SP as master problem.
- The feasibility with respect to the synchronization constraint (13) is treated in the branching strategies
- We do not need to use multiple columns. Instead we change the arrival times.
- Motivation:
  - With synchronization constraints relaxed, the SP is solvable with a wide range of established methods
- The columns are generated by solving a constrained shortest path problem with time windows.

optimal solution $(\bar{z}_{kj})$ to SP_{LP} function value $\bar{f}_{LP}$

$$V = \{(i_1, i_2) \in P^{sync} \,|\, \exists (k_1, j_1) \in Z, \exists (k_2, j_2) \in Z : |R_{i_1 j_1} - R_{i_2 j_2}| > 0\}$$

$$W = \{i \in N \,|\, \exists (k_1, j_1) \in Z, \exists (k_2, j_2) \in Z : |R_{i j_1} - R_{i j_2}| > 0\}$$

In words, V is the set of synchronization pairs for which there are schedules used in the solution with a deviation in arrival time, and W is the set of customers for which there are schedules in the solution with a deviation in arrival time. With this notation, the solution possesses one of the following properties.

P1 $(\bar{z}_{kj})$ is feasible in $SCSP$, that is, $Z = I$ and $V = \emptyset$.

P2 $(\bar{z}_{kj})$ is feasible in SP, but not in $SCSP$, that is, $Z = I$ and $V \neq \emptyset$.

P3 $(\bar{z}_{kj})$ is fractional and $V \neq \emptyset$ and $W \neq \emptyset$.

P4 $(\bar{z}_{kj})$ is fractional and $V \neq \emptyset$ and $W = \emptyset$.

P5 $(\bar{z}_{kj})$ is fractional and $V = \emptyset$ and $W \neq \emptyset$.

P6 $(\bar{z}_{kj})$ is fractional and $V = W = \emptyset$.

BR1: Branching on a time window for a customer i.
This rule is applicable when $W \neq 0$.

BR2: Branching on time windows for synchronized customers. This rule is applicable when $V \neq 0$.

BR3: Branching on the vehicle / customer pair.
This rule is applicable for P3 - P6 when we have a a fractional solution.

Test problems

Table 1: Problem instances overview.

Instance	$\|N\|$	$\|K\|$	$\|P^{sync}\|$	$\sum D_i/\|K\|$ (h)	AvTD (h)	S (h)	M (h)	L (h)
1	20	4	2	4.9	0.22	1.5	2.1	2.9
2	20	4	2	4.2	0.20	1.7	2.2	3.0
3	20	4	2	5.3	0.21	1.5	2.4	3.0
4	20	4	2	5.9	0.29	1.8	2.9	3.9
5	20	4	2	5.0	0.21	1.3	2.1	3.2
6	50	10	5	4.7	0.25	1.4	2.3	3.1
7	50	10	5	5.0	0.23	1.6	2.5	3.4
8	50	10	5	6.2	0.23	1.5	2.4	3.2
9	80	16	8	6.1	0.21	1.5	2.3	2.9
10	80	16	8	5.1	0.17	1.6	2.6	3.6

characteristics

Integrality gap

Problem	Preferences		Time	
	VRP	Sync	VRP	Sync
1S	0.00	3.64	0.00	0.00
1M	0.00	5.13	2.11	2.11
1L	0.00	4.56	8.86	10.12
2S	0.00	0.00	0.00	0.00
2M	0.00	2.18	0.14	0.83
2L	0.00	1.46	0.19	6.74
3S	0.00	1.20	0.00	2.91
3M	0.00	3.15	1.29	1.29
3L	0.00	5.19	2.10	2.10
4S	0.00	5.87	0.80	1.07
4M	0.00	3.23	0.42	5.25
4L	0.00	2.21	1.72	8.97
5S	0.00	0.00	1.12	5.51
5M	0.00	0.97	1.08	1.08
5L	0.00	0.53	0.00	5.14
6S	0.00	0.58	0.58	0.75
6M	0.09	1.80	0.79	4.42
6L	0.00	2.88	0.54	4.87
7S	0.00	0.92	0.54	2.05
7M	0.00	0.00	1.08	*
7L	0.00	1.75	0.67	*
8S	0.00	0.76	0.80	4.01
8M	0.00	0.57	0.76	4.84
8L	0.00	0.72	*	*

preferences

Prob.	LP	LBD	UBD	nodes	FC	SUB tot.	SUB root	Solve	Total
1S	-118.34		-114.03	4	1	228	152	0.73	1.27
1M	-124.17		-117.80	7	2	263	155	0.98	1.68
1L	-124.17		-118.51	8	1	306	178	1.50	2.55
2S	-92.09		-92.09	1	0	153	153	0.34	0.60
2M	-107.15		-104.81	8	2	282	170	1.36	2.30
2L	-109.24		-107.64	21	2	508	164	3.84	6.44
3S	-100.69		-99.49	13	1	242	124	1.00	1.66
3M	-110.06		-106.59	5	1	280	161	1.19	2.01
3L	-113.78		-107.87	9	1	329	162	1.64	2.63
4S	-106.24		-100.00	8	2	261	124	1.00	1.72
4M	-110.28		-106.72	9	2	300	152	1.42	2.36
4L	-111.74		-109.27	18	1	459	158	2.95	5.04
5S	-76.29		-76.29	1	1	139	139	0.34	0.64
5M	-77.03		-76.29	3	2	225	180	0.72	1.28
5L	-84.66		-84.21	6	1	275	164	1.30	2.21
6S	-372.21		-370.06	51	1	1578	864	78.45	150.63
6M	-386.86		-379.88	101	2	1861	875	126.48	247.88
6L	-398.70		-387.20	154	5	2495	818	244.94	474.15
7S	-404.85		-401.11	103	4	2066	775	146.26	291.29
7M	-406.17		-406.17	22	1	1240	721	49.53	86.70
7L	-414.76		-407.48	264	4	3030	731	368.32	714.62
8S	-383.66		-380.76	112	1	1223	718	68.34	135.39
8M	-405.87		-403.57	166	2	1851	796	148.11	290.77
8L	-410.43		-407.48	206	2	2032	731	197.17	362.18
9S	-639.26	-633.91	-552.65	414	0	5773	1922	1892.92	*
9M	-664.08	-661.77	-463.82	412	0	6026	1808	1955.00	*
9L	-675.61	-673.09	-663.47	368	3	5164	1901	1856.07	*
10S	-680.53	-676.14	-675.81	796	1	3744	2167	1732.41	*
10M	-694.97	-687.90	-685.31	328	9	5341	2091	1807.09	*
10L	-700.03	-693.98	-691.34	433	3	4764	1973	1807.72	*

Traveling time

Prob.	LP	LBD	UBD	nodes	FC	SUB tot.	SUB root	Solve	Total
1S	3.548		3.548	17	1	159	83	1.01	1.96
1M	3.475		3.548	2192	0	595	83	103.85	221.93
1L	3.081		3.393	1059	1	568	96	50.27	107.41
2S	4.266		4.266	17	0	299	99	2.02	3.28
2M	3.546		3.575	56	1	417	96	4.64	8.12
2L	3.199		3.415	9	1	210	100	1.50	2.72
3S	3.526		3.628	87	2	533	75	7.80	14.17
3M	3.291		3.333	151	4	537	85	9.49	17.57
3L	3.227		3.295	420	4	768	89	22.46	42.78
4S	6.076		6.141	152	1	356	61	7.04	14.02
4M	5.387		5.670	214	4	530	74	13.88	27.53
4L	4.711		5.134	66	1	380	85	5.04	9.74
5S	3.724		3.929	18	1	209	80	1.56	2.84
5M	3.490		3.527	550	0	506	89	26.86	57.04
5L	3.176		3.339	66	2	352	88	4.78	9.11
6S	8.083	8.129	8.143	2887	2	1244	280	2500.79	*
6M	7.375	7.674	7.714	1250	0	1955	304	2468.43	*
6L	6.807		7.138	916	3	2242	341	1874.08	3279.48
7S	8.223		8.392	525	0	1782	287	805.25	1472.39
7M	7.112	7.355	7.673	1089	1	2229	278	2376.01	*
7L	6.537	6.871	6.885	804	2	2720	304	2104.61	*
8S	9.169		9.537	260	4	1656	266	487.74	931.95
8M	8.145	8.527	8.540	917	1	2342	263	2231.23	*
8L	7.624	7.913	8.616	894	0	2168	266	2377.05	*
9S	11.498	11.695	-	437	0	2252	281	1991.10	*
9M	10.635	10.792	11.745	403	0	2378	477	1931.61	*
9L	10.159	10.366	11.106	362	0	2686	471	1938.34	*
10S	8.204	8.398	-	304	0	2548	560	1840.07	*
10M	7.392	7.502	8.538	630	0	2626	565	1606.39	*
10L	6.876	7.054	-	215	0	3025	600	1592.40	*

BR3 first vs BR3 last

- BR3 first:
  - No solution found
  - LBD= 8.145 after
  - 8,998 subproblem calls and 152 B&B nodes
- BR3 last:
  - Solution found with UBD=8,540
  - LBD= 8,527 after
  - 2342 subproblem calls and 197 B&B nodes

Concluding remarks

- New model for synchronized VRP
  - Generalization of standard VRP
  - Including constraint has a positive effect on planning (compared to make simplifiactions)
- Heuristic method
  - Finds good solutions in short time
- Set partitioning & Branch and price
  - Solution method dependent on branching strategy
  - Time window branching is better than constraint branching as long as time window branches can be found