Reactive Optimization Methods for a Field Service Routing Problem

F. Semet, S. Binart, P. Dejax, M. Gendreau

Route 2014

Presentation outline

1. Problem description
2. State of the art
3. Solution approach
   - General approach
   - Planning Methods
   - Execution methods
   - Simulations
4. Computational results
5. Conclusions

Presentation outline

Problem description

Industrial context

Service companies have to plan routes for their employees

Number of technicians is limited

Two types of interventions :

- preventive maintenance : control operations
- curative maintenance : repair operations

Service and travel times are stochastic

⇒ Planned routes have to be adjusted in real-time

Problem description

Vehicles

- Limited fleet
- Vehicle without capacity restriction
- Multi-skilled technicians
- Several depots
- Hard time-window constraints at the depots

Customers

- Mandatory customers with hard time-window restrictions
- Optional customers with profits

Other assumptions:

- One period
- Stochastic travel times and service times

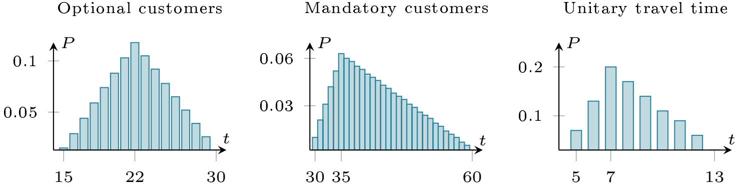
Problem description

Stochastic part

Assumptions

- Discrete times
- Minimum, modal and maximum values known a priori
- Distances satisfy the triangle inequality

Probability laws



Problem description

Objectives and constraints

Problem = design a set of routes

Constraints

- visit mandatory customers
- satisfy time-window restrictions
- one route per vehicle
- departure from the vehicle depot
- arrival at the vehicle depot

Objectives

- maximize the total profit
- minimize the estimated total time

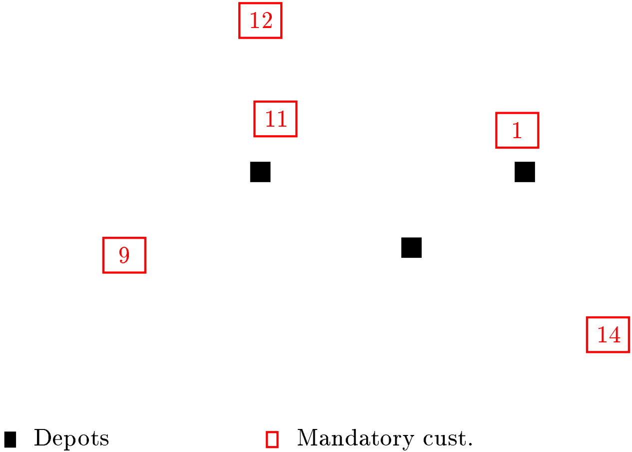
Problem description

Illustration

■ Depots

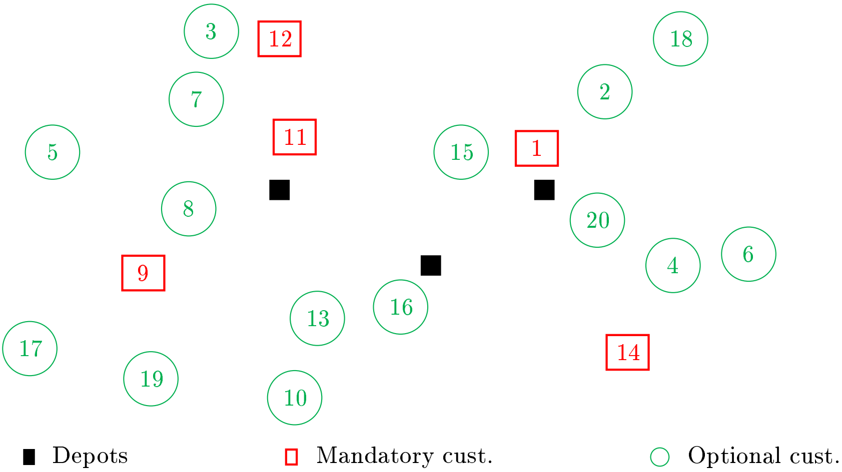
Problem description

Illustration



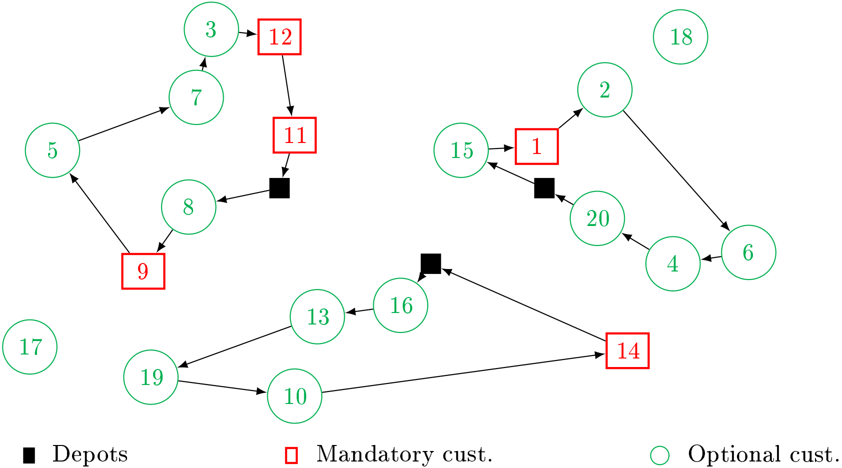
Problem description

Illustration



Problem description

Illustration



Presentation outline

Field service routing problem

Vehicles

- Limited fleet
- Vehicle without capacity restrictions
- Technicians with specific skills
- Several depots
- Hard time-window constraint at the depots

Customers

- Mandatory customers with hard time-windows
- « Optional » customers with service time-window

Others

- Multi periods
- ~~Stochastic service times and traveling times~~

Field service routing problem

State of the art

Deterministic case

Dugardin (2006) : Rules to react to different events
Tricoire (2006), Bostel *et al.* (2008) : Memetic alg. and column generation over a rolling horizon
Alsheddy *et al.* (2011) : Guided local search
Petrakis *et al.* (2012) : Heuristics (insertion, assignment) — Dynamic, Soft TW

Tricoire *et al.* (2011) : Column generation (exact and heuristic)
Rasmussen *et al.* (2012) : Column generation (exact) — 1 period

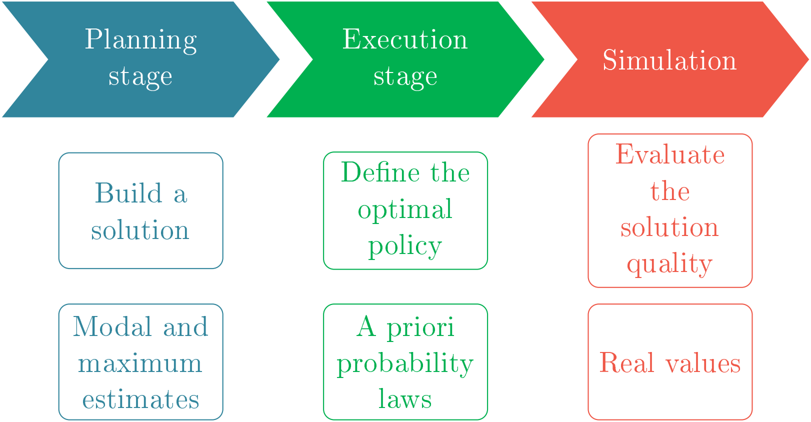
Stochastic service times (one period problem)

Borenstein *et al.* (2009) : Multi-step heuristic — Dynamic
Delage (2010) : Dynamic programming ; Monte Carlo + tabu search

Presentation outline

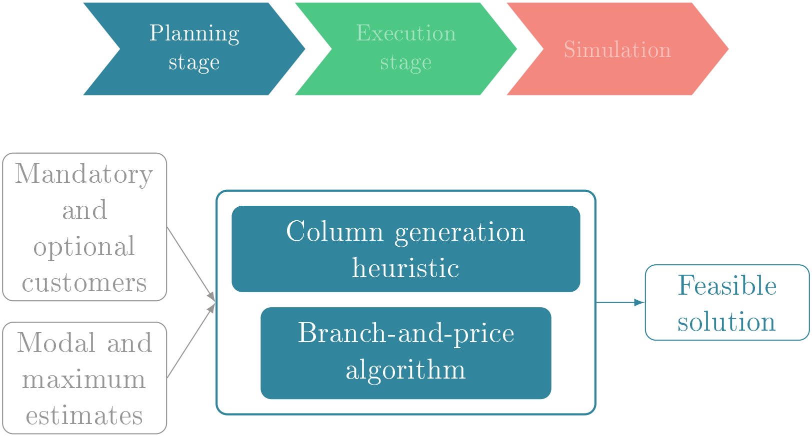
General solution approach

Description



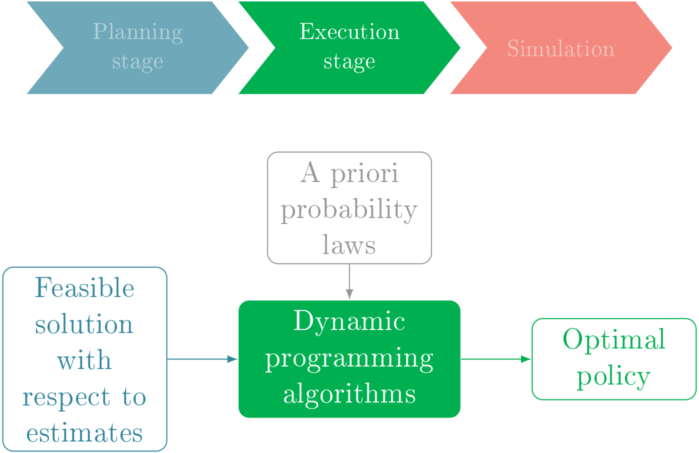
General solution approach

Planning stage



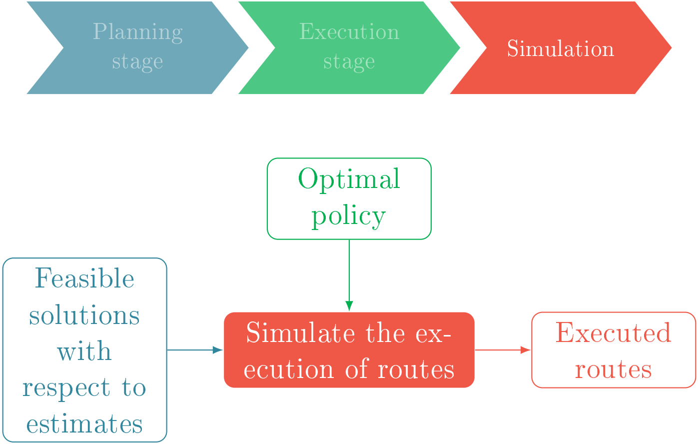
General solution approach

Execution stage



General solution approach

Simulation



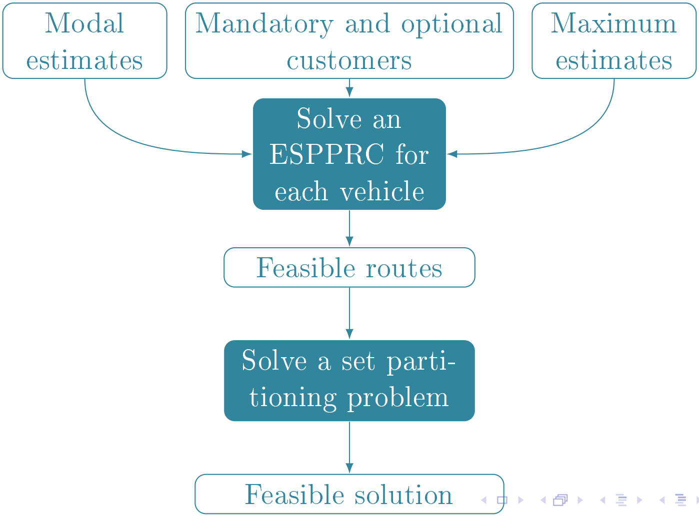
Planning Methods

Two methods for the planning stage :

- Column generation based heuristic
- Branch and price algorithm

Column generation based heuristic

General description



Column generation based heuristic

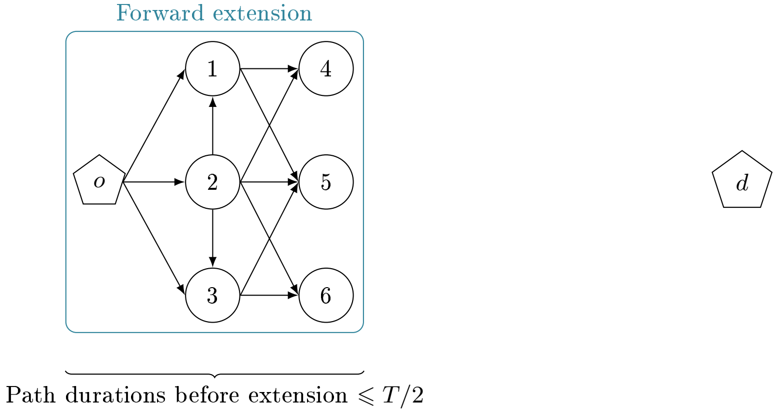
ESPPRC solution

Bounded bidirectional dynamic programming algorithm (Righini et Salani, 2006)

Column generation based heuristic

ESPPRC solution

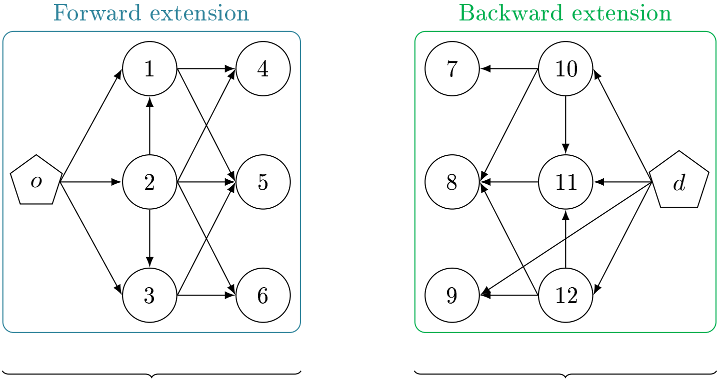
Bounded bidirectional dynamic programming algorithm (Righini et Salani, 2006)



Column generation based heuristic

ESPPRC solution

Bounded bidirectional dynamic programming algorithm (Righini et Salani, 2006)

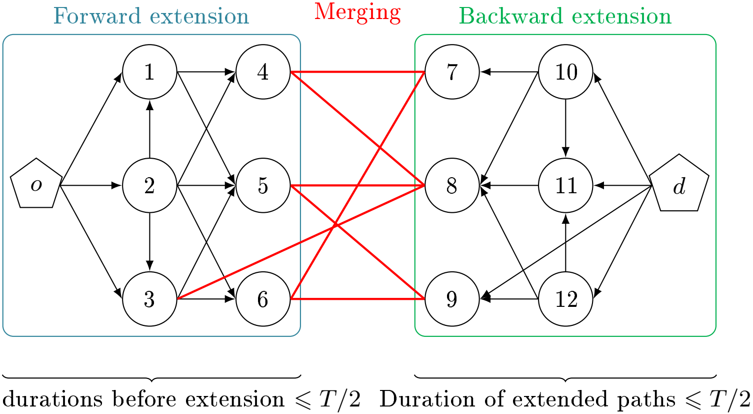


Path durations before extension $\leqslant T/2$ Duration of extended paths $\leqslant T/2$

Column generation based heuristic

ESPPRC solution

Bounded bidirectional dynamic programming algorithm (Righini et Salani, 2006)



Column generation based heuristic

Algorithm implemented by Salani

Label associated with a path : (S, t, r, u, i)

Extension rules

$$\begin{cases} S' = S \text{ and } S'[j] = 1 \\ t' = \max(e, t + \sigma_i + \tau_{ij}) \\ r' = r + p_j \\ u' = u + 1 + |U'_j| \end{cases}$$

Dominance rules

$$\begin{cases} S_1 \leqslant S_2 \\ t_1 \leqslant t_2 \\ r_1 \geqslant r_2 \\ u_1 \leqslant u_2 \\ \text{at least one inequality strict} \end{cases}$$

Feasible extension if

$$\begin{cases} S[j] = 0 \\ t' \leqslant l \\ t' \leqslant T/2 & \text{backward} \\ \boxed{t \leqslant T/2} & \text{forward} \end{cases}$$

Half-time limit may be exceeded

Merging rules

$$\begin{cases} S^{fw}[c] + S^{bw}[c] \leqslant 1 & \forall c \in N \\ t^{fw} + \sigma_i + \tau_{ij} + t^{bw} \leqslant T \end{cases}$$

Column generation based heuristic

The extended algorithm

Label associated with a path: $(S, t, \bar{t}, r, d, u, i)$ or $(S, s, t, \bar{t}, r, d, u, i)$

Extension rules

$$\begin{cases} S' = S \text{ and } S'[j] = 1 \\ t' = \max(e, t + \sigma_i + \tau_{ij}) \\ r' = r + p_j \\ u' = u + 1 + |U'_j| \\ d' = d + d_{ij} \\ \bar{t}' = \bar{t} & j \text{ opt.} \\ \bar{t}' = \max(e, \bar{t} + \bar{\sigma}_i + \bar{\tau}_{ij}) & j \text{ mand.} \end{cases}$$

Dominance rules

$$\begin{cases} S_1 \leqslant S_2 \text{ or } s_1 \leqslant s_2 \\ t_1 \leqslant t_2 \\ r_1 \geqslant r_2 \\ u_1 \leqslant u_2 \\ d_1 \leqslant d_2 \\ \bar{t}_1 \leqslant \bar{t}_2 \\ \text{at least one inequality strict} \end{cases}$$

Extension feasible if

$$\begin{cases} S[j] = 0 \\ t' \leqslant l \quad \bar{t}' \leqslant l \\ t' \leqslant T/2 \quad \bar{t}' \leqslant T/2 \end{cases}$$

Half-time limit satisfied

Merging rules

$$\begin{cases} S^{fw}[c] + S^{bw}[c] \leqslant 1 & \forall c \in N \\ t^{fw} + \sigma_i + \tau_{ij} + t^{bw} \leqslant T \\ \bar{t}^{fw} + \bar{\sigma}_i + \bar{\tau}_{ij} + \bar{t}^{bw} \leqslant T \end{cases}$$

Column generation based heuristic

OPTW instances

Extended version of the algorithm implemented by Salani

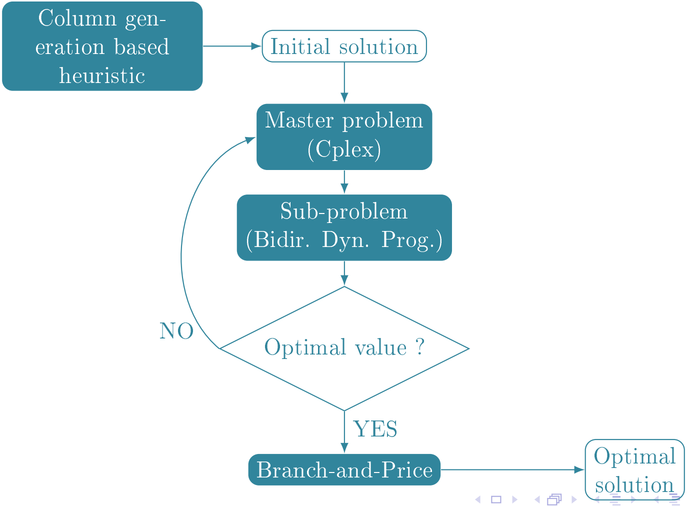
- Algorithm implemented in `C++`
- Improvement of the test sequence
- Forward labels satisfy the half-time limit

Computational results on OPTW instances

- Reduced computation time (from 30 min to 30s, from > 2h to less than 1h)
- Optimal solutions for 7 unsolved instances

Branch-and-Price Algorithm

General description



Branch and price algorithm

Main features

Branching strategy

- Branching on the assignment of customers to vehicles
  - Dense support graph for each vehicle
  - Heterogeneous fleet
- Feasibility management when branching
  - Branching on mandatory customers in priority
  - Feasibility of routes restricted to mandatory customers

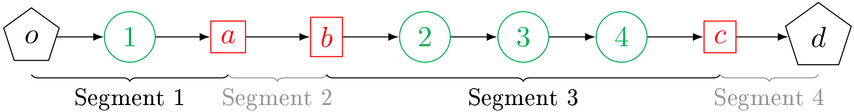
Subproblem solution

Exact and approximate variants of the bounded bidirectional dynamic programming algorithm

Execution algorithms

General description

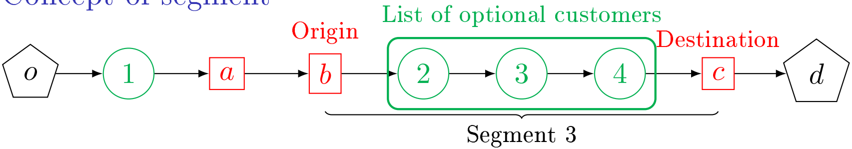
Concept of segment



Execution algorithms

General description

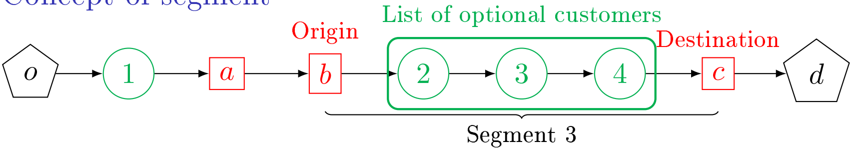
Concept of segment



Execution algorithms

General description

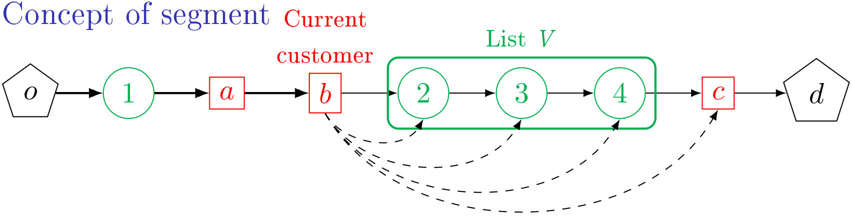
Concept of segment



Execution algorithms

General description

Concept of segment



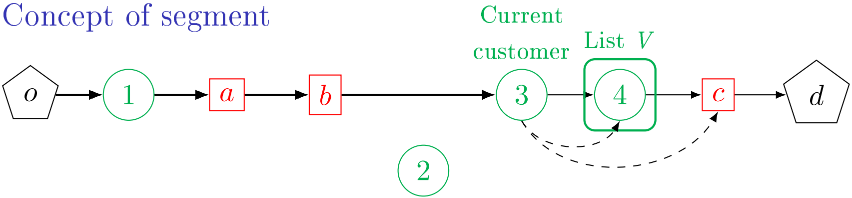
Dynamic Programming

- Step : End of service at a customer
- State : time t, customer v, list of optional unvisited customers V
- Decision : Next customer served
- Strategy : One segment or whole route

Execution algorithms

General description

Concept of segment

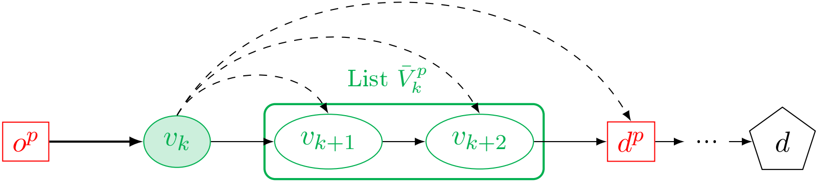


Dynamic Programming

- Step : End of service at a customer
- State : time t, customer v, list of optional unvisited customers V
- Decision : Next customer served
- Strategy : One segment or whole route

Execution algorithms

Algorithm based one segment (OS)

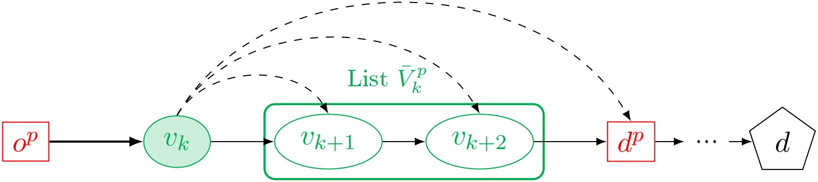


$$f(v_k, t_k, \bar{V}_k^p) = \max \begin{cases} E[f(d^p, t_k + \tau_{v_k d^p}, \emptyset)] \\ \max_{\bar{v} \in \bar{V}_k^p} \left(p_{\bar{v}} + E[f(\bar{v}, t_k + \tau_{v_k \bar{v}} + \sigma_{\bar{v}}, \bar{V}_k^p \setminus \{\bar{v}\})] \right) \end{cases}$$

with $f(d^p, t, \emptyset) = \quad -\Gamma_{d^p} \max(t - l_{d^p}, 0)$

Execution algorithms

Algorithm based one segment (OS)



d^p

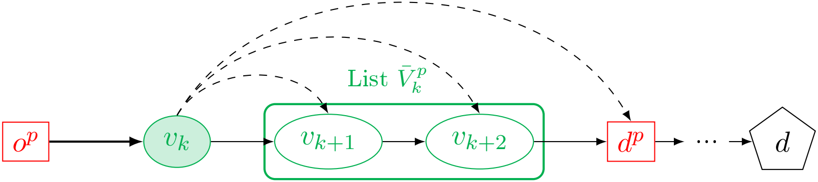
$\bar{V}_k^p$

$$f(v_k, t_k, \bar{V}_k^p) = \max \begin{cases} E[f(d^p, t_k + \tau_{v_k d^p}, \emptyset)] \\ \max_{\bar{v} \in \bar{V}_k^p} \left(p_{\bar{v}} + E[f(\bar{v}, t_k + \tau_{v_k \bar{v}} + \sigma_{\bar{v}}, \bar{V}_k^p \setminus \{\bar{v}\})] \right) \end{cases}$$

$$\text{with } f(d^p, t, \emptyset) = \quad -\Gamma_{d^p} \max(t - l_{d^p}, 0)$$

Execution algorithms

Algorithm based one segment (OS)

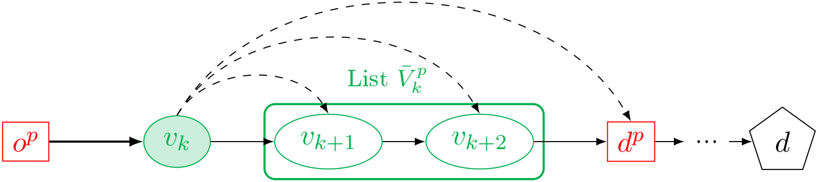


$$f(v_k, t_k, \bar{V}_k^p) = \max \begin{cases} E[f(d^p, t_k + \tau_{v_k d^p}, \emptyset)] \\ \max_{\bar{v} \in \bar{V}_k^p} \left(\underbrace{p_{\bar{v}}}_{\text{Profit in } \bar{v}} + \underbrace{E[f(\bar{v}, t_k + \tau_{v_k \bar{v}} + \sigma_{\bar{v}}, \bar{V}_k^p \backslash \{\bar{v}\})]}_{\text{Expected revenue once } \bar{v} \text{ visited}} \right) \end{cases}$$

with $f(d^p, t, \emptyset) = \quad -\Gamma_{d^p} \max(t - l_{d^p}, 0)$

Execution algorithms

Algorithm based one segment (OS)



$$f(v_k, t_k, \bar{V}_k^p) = \max \begin{cases} E[f(d^p, t_k + \tau_{v_k d^p}, \emptyset)] \\ \max_{\bar{v} \in \bar{V}_k^p} \left(p_{\bar{v}} + E[f(\bar{v}, t_k + \tau_{v_k \bar{v}} + \sigma_{\bar{v}}, \bar{V}_k^p \setminus \{\bar{v}\})] \right) \end{cases}$$

with $f(d^p, t, \emptyset) = -\Gamma_{d^p} \max(t - l_{d^p}, 0)$ Delay penalty

Simulations

Parameters

- 100 simulations per instance
- Minimal speed = 15 km/h (estimate = 20 km/h)
- Maximal service time at mandatory customers = 90 min (estimate = 60 min)

Presentation outline

Computational results

Instances

Instances based on Tricoire instances (2006)

Parameters

$\alpha = 1$	travel time weight
$p_i = 100$	profit associated with optional customers
$30 \leqslant \sigma_i \leqslant 60$	estimate of the service time for mandatory customers
$15 \leqslant \sigma_i \leqslant 30$	estimate of the service time for optional customers
$20 \leqslant v \leqslant 50$	estimate of speed (km/h)

Experimental environment

- Linear programs solved with Cplex 12.4
- 100 simulations per instance
- Computer with 4CPU, 2.8Ghz, 30Go of RAM

Computational results

Before simulation

# cust.	# mand. cust.	Heuristic based on customer priorities			Column generation heuristic			Branch and price algorithm		
		CPU (s)	non served	Dist. (km)	CPU (s)	non served	Dist. (km)	CPU (s)	non served	Dist. (km)
30	5	43	1,4	220	22	0,0	200	**3588**	**0,0**	**196**
	6	66	2,8	232	37	0,0	214	**4740**	**0,0**	**199**
	7	53	2,4	230	34	0,0	219	**4020**	**0,0**	**202**
	8	34	3,4	216	25	0,0	214	**1995**	**0,0**	**201**
	9	24	3	217	29	0,0	217	**1788**	**0,0**	**205**
40	5	3277	7,2	202	97	1,4	238	2652	0,0	237
	6	962	10,2	217	120	1,8	250	3828	0,0	244
	7	1451	10	211	122	2,2	254	2580	0,2	250
	8	1880	10	210	98	2,0	242	3408	0,2	237
	9	1053	10	206	120	2,4	240	5325	1,2	236
50	5	1227	15,4	192	184	8,0	226	7200	7,2	215
	6	3064	17,6	197	212	8,6	222	5832	7,4	218
	7	2844	18	192	240	9,2	221	7200	8,6	218
	8	2323	18,6	200	218	9,2	215	5568	8,2	212
	9	1408	19,2	188	261	9,4	221	5856	8,8	213

Computational results

Before simulation

# cust.	# mand. cust.	Heuristic based on customer priorities			Column generation heuristic			Branch and price algorithm		
		CPU (s)	non served	Dist. (km)	CPU (s)	non served	Dist. (km)	CPU (s)	non served	Dist. (km)
30	5	43	1,4	220	22	0,0	200	**3588**	**0,0**	**196**
	6	66	2,8	232	37	0,0	214	**4740**	**0,0**	**199**
	7	53	2,4	230	34	0,0	219	**4020**	**0,0**	**202**
	8	34	3,4	216	25	0,0	214	**1995**	**0,0**	**201**
	9	24	3	217	29	0,0	217	**1788**	**0,0**	**205**
40	5	3277	7,2	202	97	1,4	238	2652	0,0	237
	6	962	10,2	217	120	1,8	250	3828	0,0	244
	7	1451	10	211	122	2,2	254	2580	0,2	250
	8	1880	10	210	98	2,0	242	3408	0,2	237
	9	1053	10	206	120	2,4	240	5325	1,2	236
50	5	1227	15,4	192	184	8,0	226	7200	7,2	215
	6	3064	17,6	197	212	8,6	222	5832	7,4	218
	7	2844	18	192	240	9,2	221	7200	8,6	218
	8	2323	18,6	200	218	9,2	215	5568	8,2	212
	9	1408	19,2	188	261	9,4	221	5856	8,8	213

Computational results

Before simulation

# cust.	# mand. cust.	Heuristic based on customer priorities CPU (s)	non served	Dist. (km)	Column generation heuristic CPU (s)	non served	Dist. (km)	Branch and price algorithm CPU (s)	non served	Dist. (km)
30	5	43	1,4	220	22	0,0	200	**3588**	**0,0**	**196**
	6	66	2,8	232	37	0,0	214	**4740**	**0,0**	**199**
	7	53	2,4	230	34	0,0	219	**4020**	**0,0**	**202**
	8	34	3,4	216	25	0,0	214	**1995**	**0,0**	**201**
	9	24	3	217	29	0,0	217	**1788**	**0,0**	**205**
40	5	3277	7,2	202	97	1,4	238	2652	0,0	237
	6	962	10,2	217	120	1,8	250	3828	0,0	244
	7	1451	10	211	122	2,2	254	2580	0,2	250
	8	1880	10	210	98	2,0	242	3408	0,2	237
	9	1053	10	206	120	2,4	240	5325	1,2	236
50	5	1227	15,4	192	184	8,0	226	7200	7,2	215
	6	3064	17,6	197	212	8,6	222	5832	7,4	218
	7	2844	18	192	240	9,2	221	7200	8,6	218
	8	2323	18,6	200	218	9,2	215	5568	8,2	212
	9	1408	19,2	188	261	9,4	221	5856	8,8	213

Computational results for the 2 execution strategy

Computational results after simulation

		Unserved				Distance				Delay			
	# clients oblig.	Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.	
		WR	OS	WR	OS	WR	OS	WR	OS	WR	OS	WR	OS
30	5	1,2	0,7	2,3	1,9	197	199	187	189	0,6	7,4	0,2	5,8
	6	3,8	1,9	4,2	2,5	199	208	181	190	1,1	19,2	1,2	36,3
	7	4	2,8	4,3	2,4	206	213	183	193	2,1	30,8	1,6	25,8
	8	3,2	2,2	4,7	3,7	201	208	180	188	2,4	29,3	1,8	30,8
	9	4,5	2	4,9	3,0	203	214	184	198	4,8	62,1	3,6	51,0
40	5	7,8	6,4	7,7	6,3	206	220	202	216	1,8	44,7	1,3	47,4
	6	9,9	7,9	8,7	7,0	214	229	196	211	2,2	59,1	1,6	44,0
	7	11,7	8,8	10,8	7,8	221	241	207	226	2,7	78,2	3,1	80,0
	8	11,1	7,9	10,5	6,7	201	232	194	216	2,7	93	2,7	116,7
	9	12,3	7,8	11,8	8,0	198	232	201	222	8,2	135	7,1	101,5
50	5	16,3	14,6	16,0	14,6	195	214	191	200	1,8	68,9	1,6	49,1
	6	17,5	15,5	17,5	15,2	193	208	192	200	1,5	48,9	1,1	43,7
	7	19,1	16,7	19,0	15,8	194	208	192	204	2,9	62,8	3,2	68,4
	8	19,7	16,4	19,2	16,6	188	205	182	198	3,9	82,2	2,8	64,4
	9	21,3	16,8	20,8	16,1	187	210	186	202	3,5	102,6	3,9	100,3

Computational results for the 2 execution strategy

Computational results after simulation

		Unserved				Distance				Delay			
	# clients oblig.	Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.	
		WR	OS	WR	OS	WR	OS	WR	OS	WR	OS	WR	OS
30	5	1,2	0,7	2,3	1,9	197	199	187	189	0,6	7,4	0,2	5,8
	6	3,8	1,9	4,2	2,5	199	208	181	190	1,1	19,2	1,2	36,3
	7	4	2,8	4,3	2,4	206	213	183	193	2,1	30,8	1,6	25,8
	8	3,2	2,2	4,7	3,7	201	208	180	188	2,4	29,3	1,8	30,8
	9	4,5	2	4,9	3,0	203	214	184	198	4,8	62,1	3,6	51,0
40	5	7,8	6,4	7,7	6,3	206	220	202	216	1,8	44,7	1,3	47,4
	6	9,9	7,9	8,7	7,0	214	229	196	211	2,2	59,1	1,6	44,0
	7	11,7	8,8	10,8	7,8	221	241	207	226	2,7	78,2	3,1	80,0
	8	11,1	7,9	10,5	6,7	201	232	194	216	2,7	93	2,7	116,7
	9	12,3	7,8	11,8	8,0	198	232	201	222	8,2	135	7,1	101,5
50	5	16,3	14,6	16,0	14,6	195	214	191	200	1,8	68,9	1,6	49,1
	6	17,5	15,5	17,5	15,2	193	208	192	200	1,5	48,9	1,1	43,7
	7	19,1	16,7	19,0	15,8	194	208	192	204	2,9	62,8	3,2	68,4
	8	19,7	16,4	19,2	16,6	188	205	182	198	3,9	82,2	2,8	64,4
	9	21,3	16,8	20,8	16,1	187	210	186	202	3,5	102,6	3,9	100,3

Computational results for the 2 execution strategy

Computational results after simulation

		Unserved				Distance				Delay			
	# clients oblig.	Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.	
		WR	OS	WR	OS	WR	OS	WR	OS	WR	OS	WR	OS
30	5	1,2	0,7	2,3	1,9	197	199	187	189	0,6	7,4	0,2	5,8
	6	3,8	1,9	4,2	2,5	199	208	181	190	1,1	19,2	1,2	36,3
	7	4	2,8	4,3	2,4	206	213	183	193	2,1	30,8	1,6	25,8
	8	3,2	2,2	4,7	3,7	201	208	180	188	2,4	29,3	1,8	30,8
	9	4,5	2	4,9	3,0	203	214	184	198	4,8	62,1	3,6	51,0
40	5	7,8	6,4	7,7	6,3	206	220	202	216	1,8	44,7	1,3	47,4
	6	9,9	7,9	8,7	7,0	214	229	196	211	2,2	59,1	1,6	44,0
	7	11,7	8,8	10,8	7,8	221	241	207	226	2,7	78,2	3,1	80,0
	8	11,1	7,9	10,5	6,7	201	232	194	216	2,7	93	2,7	116,7
	9	12,3	7,8	11,8	8,0	198	232	201	222	8,2	135	7,1	101,5
50	5	16,3	14,6	16,0	14,6	195	214	191	200	1,8	68,9	1,6	49,1
	6	17,5	15,5	17,5	15,2	193	208	192	200	1,5	48,9	1,1	43,7
	7	19,1	16,7	19,0	15,8	194	208	192	204	2,9	62,8	3,2	68,4
	8	19,7	16,4	19,2	16,6	188	205	182	198	3,9	82,2	2,8	64,4
	9	21,3	16,8	20,8	16,1	187	210	186	202	3,5	102,6	3,9	100,3

Computational results for the 2 execution strategy

Computational results after simulation

		Unserved				**Distance**				**Delay**			
	# clients oblig.	Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.		Column generation heuristic		Branch and price alg.	
		WR	OS	WR	OS	WR	OS	WR	OS	WR	OS	WR	OS
30	5	1,2	0,7	2,3	1,9	197	199	187	189	0,6	7,4	0,2	5,8
	6	3,8	1,9	4,2	2,5	199	208	181	190	1,1	19,2	1,2	36,3
	7	4	2,8	4,3	2,4	206	213	183	193	2,1	30,8	1,6	25,8
	8	3,2	2,2	4,7	3,7	201	208	180	188	2,4	29,3	1,8	30,8
	9	4,5	2	4,9	3,0	203	214	184	198	4,8	62,1	3,6	51,0
40	5	7,8	6,4	7,7	6,3	206	220	202	216	1,8	44,7	1,3	47,4
	6	9,9	7,9	8,7	7,0	214	229	196	211	2,2	59,1	1,6	44,0
	7	11,7	8,8	10,8	7,8	221	241	207	226	2,7	78,2	3,1	80,0
	8	11,1	7,9	10,5	6,7	201	232	194	216	2,7	93	2,7	116,7
	9	12,3	7,8	11,8	8,0	198	232	201	222	8,2	135	7,1	101,5
50	5	16,3	14,6	16,0	14,6	195	214	191	200	1,8	68,9	1,6	49,1
	6	17,5	15,5	17,5	15,2	193	208	192	200	1,5	48,9	1,1	43,7
	7	19,1	16,7	19,0	15,8	194	208	192	204	2,9	62,8	3,2	68,4
	8	19,7	16,4	19,2	16,6	188	205	182	198	3,9	82,2	2,8	64,4
	9	21,3	16,8	20,8	16,1	187	210	186	202	3,5	102,6	3,9	100,3

Presentation outline

Conclusions

Comparison of planning methods

- Column generation based heuristic
  - + Small computation times
  - + Good quality solutions
- Branch and price algorithm
  - + Optimal solutions for small-sized instances
  - + Very good quality solutions
  - – Larger computation times

Conclusions

Execution strategies

- OS strategy with respect to **the number of unserved customers**
- WR strategy with respect to the **distance** and to the **delay**

Perspectives

- Address multi-period case
- Improve the branch-and-price algorithm