Advances in Exact Algorithms for Vehicle Routing

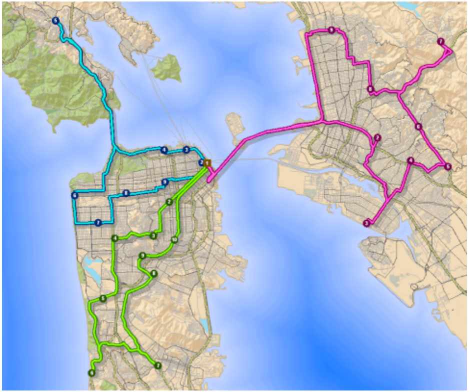
Eduardo Uchoa

Departamento de Engenharia de Produção
Universidade Federal Fluminense, Brazil
INRIA International Chair 2022-2026, Bordeaux

Vehicle Routing Problem (VRP)

One of the most widely studied in Combinatorial Optimization:

- +6,000 works published only in 2021 (Google Scholar), mostly heuristics
- Direct application in the real systems that distribute goods and provide services

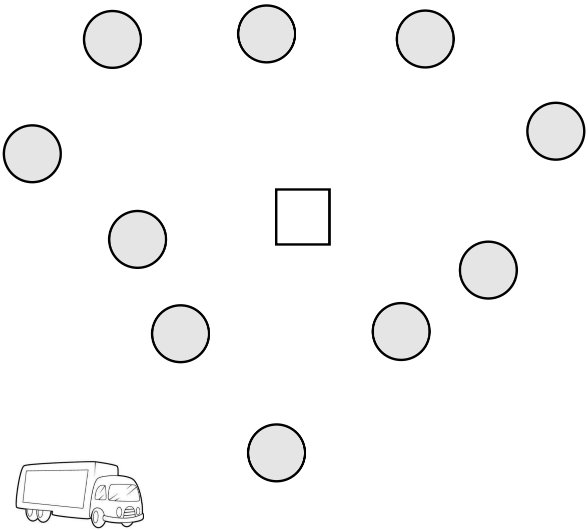


Vehicle Routing Problem (VRP)

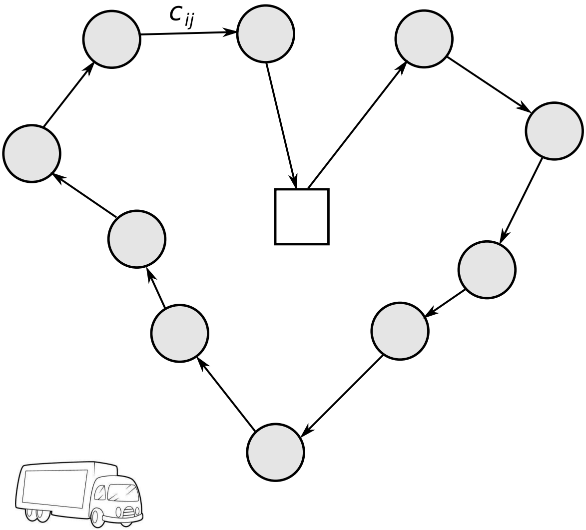
Reflecting the variety of real transportation systems, VRP literature is spread into hundreds of variants. For example, there are variants that consider:

- Vehicle capacities,
- Time windows,
- Heterogeneous fleets,
- Multiple depots,
- Split delivery, pickup and delivery, backhauling,
- Arc routing (Ex: garbage collection),
- etc, etc.

Traveling Salesman Problem (TSP)

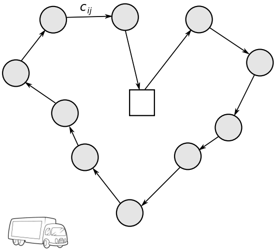


Traveling Salesman Problem (TSP)

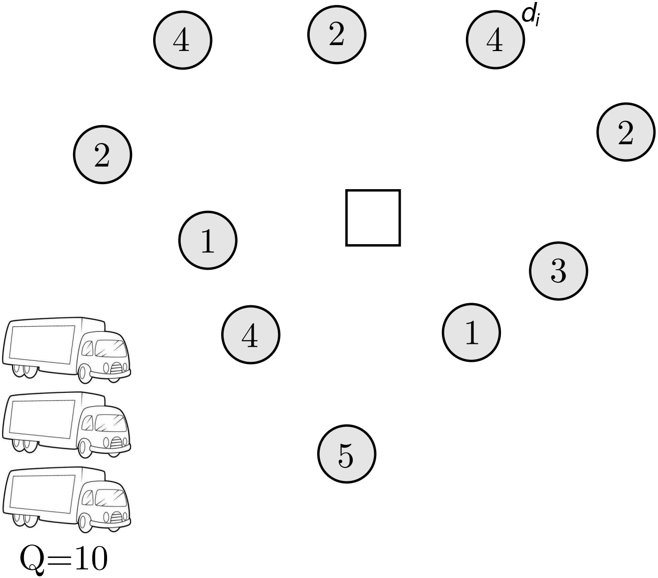


Traveling Salesman Problem (TSP)

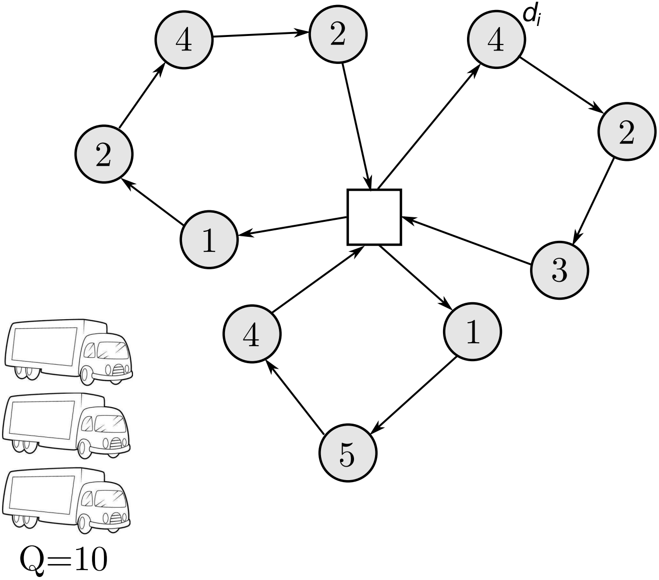
Simplest routing problem: single vehicle, no constraints to the route



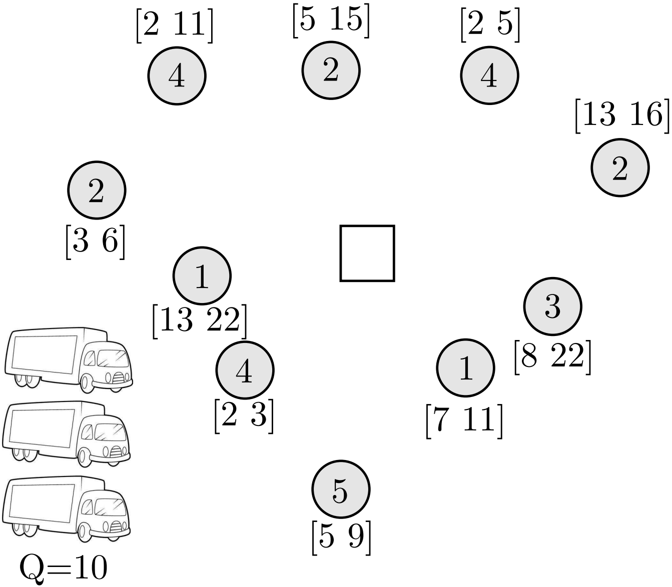
Largest solved instance (Concorde): 85,900 points!! Instances with one thousand points often solved in one minute


4
2
4
d_i
2
2
1
3
4
1
5
Q=10

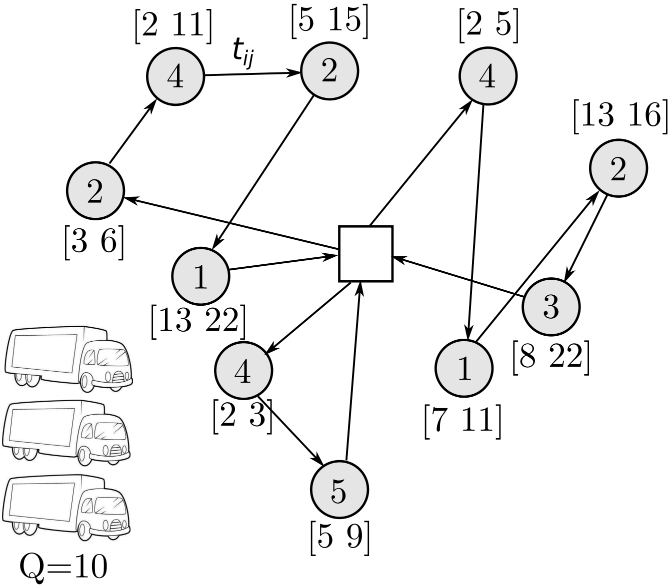
Capacitated Vehicle Routing Problem (CVRP)

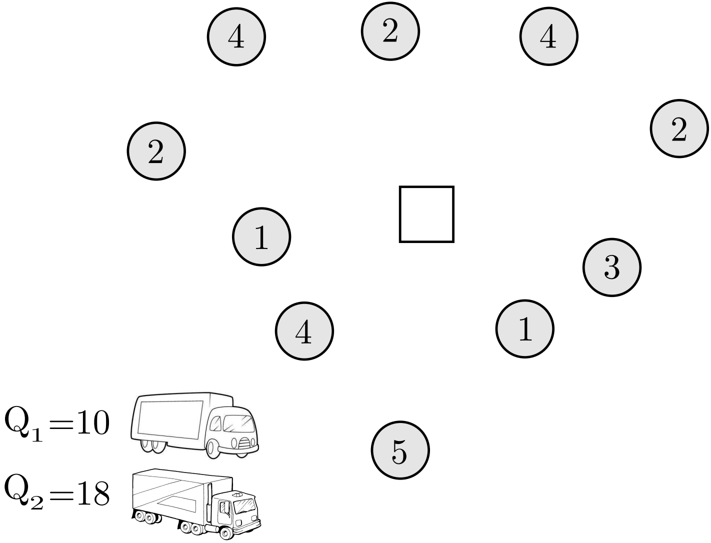


VRP with Time Windows (VRPTW)

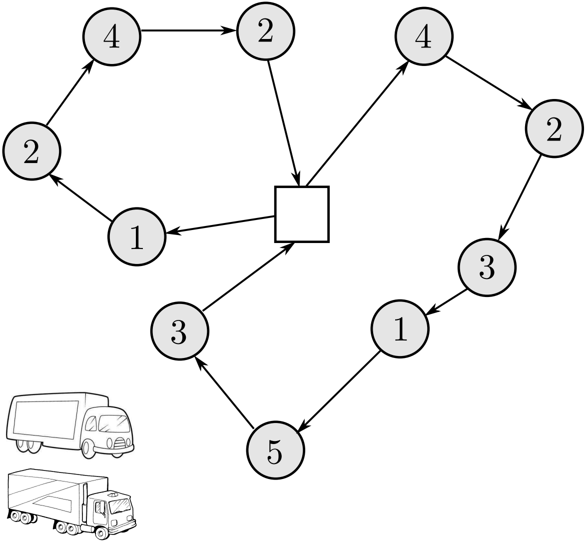


VRP with Time Windows (VRPTW)




$Q_1 = 10$
$Q_2 = 18$

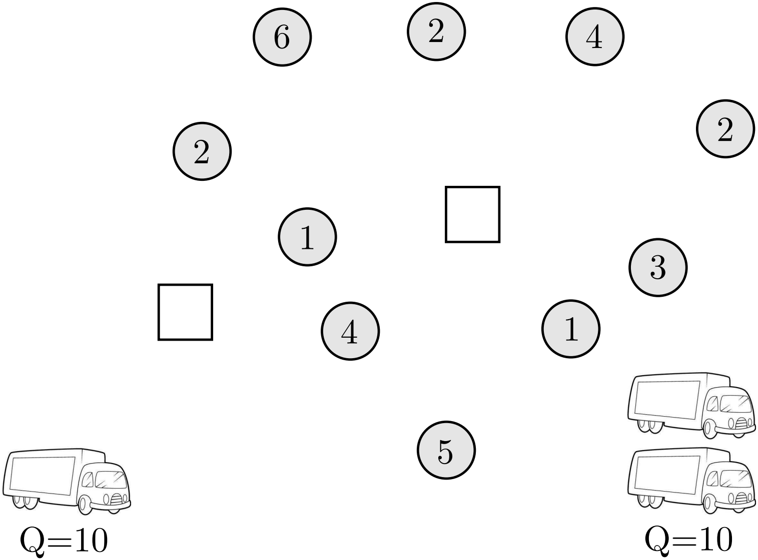
Heterogeneous Fleet VRP (HFVRP)



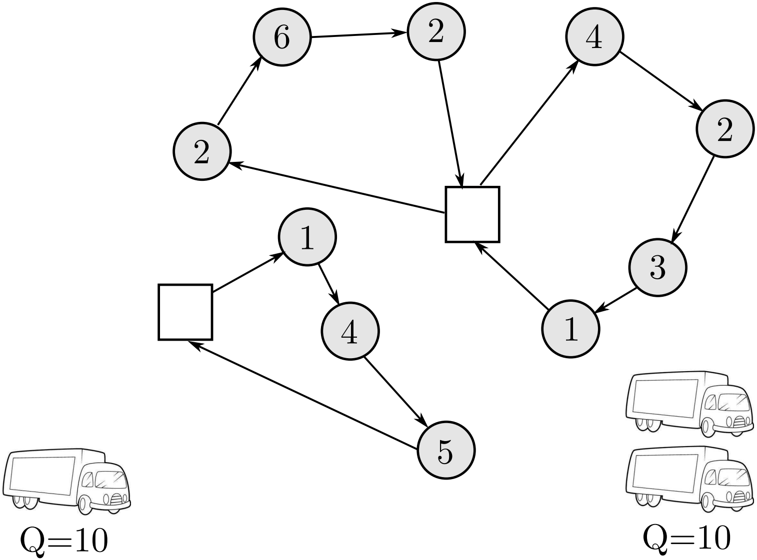
$Q_1 = 10$

$Q_2 = 18$

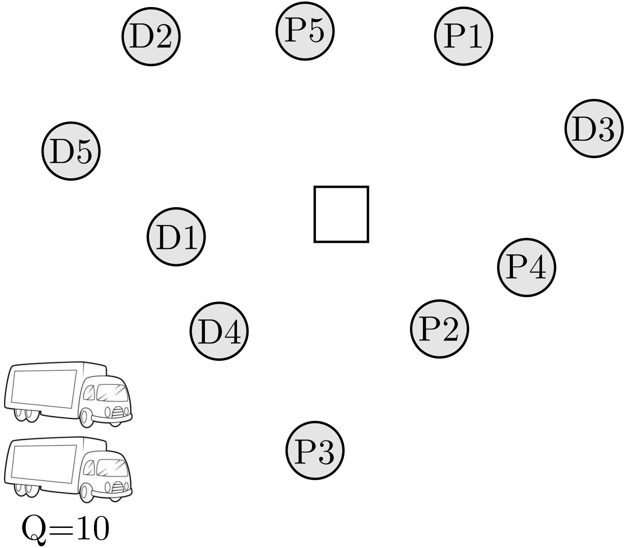
Multi-Depot VRP (MDVRP)



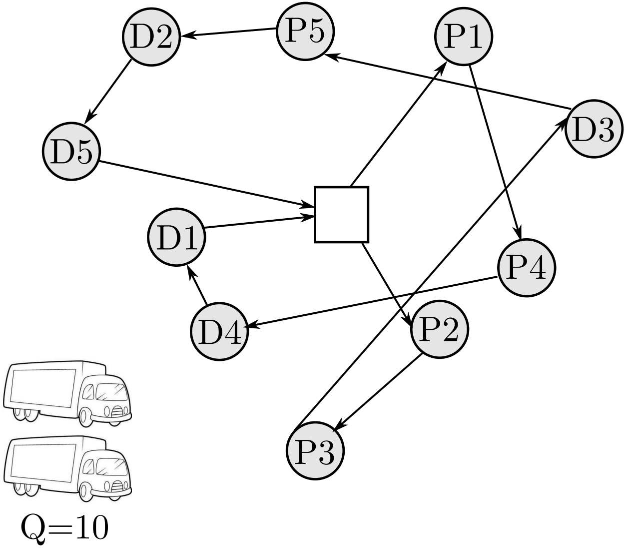
Multi-Depot VRP (MDVRP)



Pickup and Delivery VRP (PDVRP)



Pickup and Delivery VRP (PDVRP)



"Emerging" variants

Routing of electrical vehicles:

- Limited autonomy. Recharge is only available at a few points and is slow

Dynamic Routing:

- New demands appear during the day, ongoing routes may have to be changed due to unexpected events...

Outline of the presentation

Part I - Advances on Exact CVRP algorithms

- Review of the advances in the last 15 years
- A key advance: cuts with limited memory

Part II - From CVRP to other classic VRP variants

Part III - A Generic Exact VRP Solver

- A generic VRP model
- Computational results
- Downloading and using the code

Conclusion: perspectives on the use of exact algorithms in practice

Part I - Advances on Exact CVRP Algorithms

Capacitated Vehicle Routing Problem (CVRP)

First (Dantzig and Ramser [1959]) and **most basic variant**:

Instance: Complete graph $G = (V, E)$ with $V = \{0, \ldots, n\}$; 0 is the depot, $N = \{1, \ldots, n\}$ is the set of customers. Each edge $e \in E$ costs c_e. Each $i \in N$ demands d_i units. Homogeneous fleet of vehicles with capacity Q.

Solution: Routes from the depot, respecting the capacities and visiting all customers once; minimizing the total cost.

Edge Formulation (Laporte and Nobert [1983])

Variable x_e indicates how many times e is used.

$$\min \sum_{e \in E} c_e x_e \tag{1}$$

$$\text{S.t.} \quad \sum_{e \in \delta(i)} x_e = 2 \qquad \forall\, i \in N, \tag{2}$$

$$\sum_{e \in \delta(S)} x_e \geq 2 \left\lceil \sum_{i \in S} d_i / Q \right\rceil \qquad \forall\, S \subseteq N, \tag{3}$$

$$x_e \in \{0, 1\} \qquad \forall\, e \in E \setminus \delta(0), \tag{4}$$

$$x_e \in \{0, 1, 2\} \qquad \forall\, e \in \delta(0). \tag{5}$$

Constraints (3) are *Rounded Capacity Cuts*

Branch-and-Cut (BC) Algorithms for CVRP

Extensive research on families of cuts:

- Framed Capacity, Strengthened Comb, Multistar, Extended Hypotour, etc.

Dominant approach (Naddef and Rinaldi [2002]) until early 2000's:

- Araque, Kudva, Morin, and Pekny [1994]
- Augerat, Belenguer, Benavent, Corberán, Naddef, and Rinaldi [1995]
- Blasum and Hochstättler [2000]
- Ralphs, Kopman, Pulleyblank, and Trotter Jr. [2003]
- Achuthan, Caccetta, and Hill [2003]
- Ralphs [2003]
- Wenger [2004]
- Lysgaard, Letchford, and Eglese [2004]

Best BC results

Class	Size	#Ins	LLE04 #Unsolved	Root gap	Avg. Time(s)
A	36-79	22	7	2.06	6638
B	36-79	20	1	0.61	8178
E-M	50-199	12	9	2.10	39592
F	44-134	3	0	0.06	1016
P	14-100	24	8	2.26	11219
Total		81	25		
Processor			Intel Celeron 700MHz		

Smallest unsolved instance: 49 customers

J. Lysgaard, A. Letchford, and R. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100:423–445, 2004

Why BC is so much better on TSP than on CVRP? One reason may be that the original Dantzig et al. [1954] formulation is much stronger than the original Laporte and Nobert [1983] formulation

Best BC results

Class	Size	#Ins	LLE04 #Unsolved	LLE04 Root gap	LLE04 Avg. Time(s)
A	36-79	22	7	2.06	6638
B	36-79	20	1	0.61	8178
E-M	50-199	12	9	2.10	39592
F	44-134	3	0	0.06	1016
P	14-100	24	8	2.26	11219
Total		81	25		
Processor			Intel Celeron 700MHz		

Smallest unsolved instance: 49 customers

J. Lysgaard, A. Letchford, and R. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100:423–445, 2004

Why BC is so much better on TSP than on CVRP? One reason may be that the original Dantzig et al. [1954] formulation is much stronger than the original Laporte and Nobert [1983] formulation

Set Partitioning Formulation – SPF (Balinski and Quandt [1964])

Ω is the set of routes, route r costs c_r, coefficient a_{ir} indicates how many times r visits customer i

$$\min \sum_{r\in\Omega} c_r\lambda_r \quad (6)$$

$$\text{S.t.} \quad \sum_{r\in\Omega} a_{ir}\lambda_r = 1 \qquad \forall\ i \in N, \quad (7)$$

$$\lambda_r \in \{0,1\} \qquad \forall\ r \in \Omega. \quad (8)$$

- Exponential number of variables $\Longrightarrow$ Column generation / Branch-and-Price (BP) algorithms
- Pricing elementary routes is strongly NP-hard $\Longrightarrow$ Relax Ω including non-elementary q-routes (Christofides et al. [1979])

Combining Column Generation and Cut Separation

SPF linear relaxation is very good for VRPTW *with tight windows* (Desrosiers et al. [1984]) but **is quite weak for CVRP**:

- Typical root gaps $>$**3%** (even if only elementary routes are priced!), worse than **2%** of BC root gap

Fukasawa et al. [2006] combined both methods. A cut over edge variables

$$\sum_{e \in E} \alpha_e x_e \geq b,$$

is translated to

$$\sum_{r \in \Omega} \left(\sum_{e \in E} \alpha_e a_{er} \right) \lambda_r \geq b,$$

where a_{er} is the number of times that e is used in route r.

The combination of column generation with robust cuts defined over edges yields root gaps around **1%**.

A crucial point in combining column generation with cuts is the effect of the new dual variables in the pricing:

- A cut is **robust** when its dual variable can be translated into costs in the pricing. The subproblem structure does not change.
- **non-robust** cuts change the pricing, each additional cut makes it harder.

Robust BCP results in FLL+06

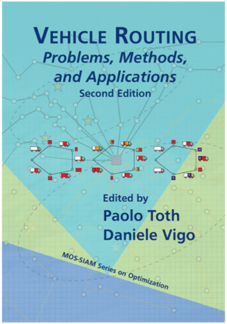
		LLE04			FLL+06		
Class	#Ins	NS	Gap	T(s)	NS	Gap	T(s)
A	22	7	2.06	6638	0	0.81	1961
B	20	1	0.61	8178	0	0.47	4763
E-M	12	9	2.10	39592	3	1.19	126987
F	3	0	0.06	1016	0	0.06	2398
P	24	8	2.26	11219	0	0.76	2892
Total	81	25			3		
Processor		Intel Celeron 700MHz			Pentium 4 2.4GHz		

Robust BCP solved all literature instances with up to 134 customers. Three larger instances remained open: M-n151-k12, M-n200-k16 e M-n200-k17.

R. Fukasawa, H. Longo, J. Lysgaard, M. Poggi de Aragão, M. Reis, Uchoa. E., and R.F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, 106: 491–511, 2006

New exact CVRP methods

After Fukasawa et al. [2006], all the proposed exact CVRP algorithms combine column generation and cuts. Two surveys are:



M. Poggi and Uchoa. E. New exact approaches for the capacitated VRP. In P. Toth and D. Vigo, editors, *Vehicle Routing: Problems, Methods, and Applications*, chapter 3, pages 59–86. SIAM, second edition, 2014

L. Costa, C. Contardo, and G. Desaulniers. Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science*, 53:946–985, 2019

- Uses **non-robust cuts**: Strengthened Capacity and Clique.

Root gaps were significantly reduced. Several tricks to keep pricing reasonably tractable.

Important new idea:

- Instead of branching, algorithm finishes by **enumerating** all routes with reduced cost smaller than the gap. The SPF with only those routes was solved by CPLEX. This saves a lot of time in some instances

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

Introduces *ng*-**routes**, an effective elementarity relaxation better than *q*-routes:

- For each $i \in N$, $NG(i) \subseteq N$ contains the *ng* closest customers. An *ng*-route can only revisit i if it passes first by a customer j such that $i \notin NG(j)$
- $ng = 8$ does not make pricing too hard and, in practice, eliminates most cycles

Non-robust Subset Row Cuts (Chvátal-Gomory Cuts of Rank 1 over the set partitioning constraints [Jepsen et al., 2008]) replace Cliques, smaller impact on pricing

R. Baldacci, A. Mingozzi, and R. Roberti. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, 59:1269–1283, 2011a

Røpke [2012]

Back to Robust BCP, but already using *ng*-routes.

- Proposes a sophisticated and aggressive **strong branching**, reducing a lot the branch-and-bound trees
- Before this work, BCP algorithms only applied SB in a timid way, wrongly believing that aggressive SB would not pay in that context

M-n151-k12 solved in 5 days!

S. Røpke. Branching decisions in branch-and-cut-and-price algorithms for vehicle routing problems. *Presentation in Column Generation 2012*, 2012

Contardo and Martinelli [2014]

- Uses Subset Row Cuts and *ng*-routes
- **Enumeration to a pool** with up to several million routes can be performed. After that, pricing is done by inspection in the pool.
    - Non-robusts cuts can be freely separated
    - As lower bounds improve, fixing by reduced costs reduce pool size
    - The problem is finished by a MIP solver only when pool size is much reduced

M-n151-k12 solved in less than 3 hours!

C. Contardo and R. Martinelli. A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. *Discrete Optimization*, 12:129–146, 2014a

Pecin et al. [2014]

A complex BCP algorithm incorporating elements from **all** previously mentioned works and presenting some new ideas.

D. Pecin, A. Pessoa, M. Poggi, and Uchoa. E. Improved branch-cut-and-price for capacitated vehicle routing. In *Proceedings of the 17th IPCO*, pages 393–403. Springer, 2014

Pecin et al. [2014]: Cuts

- Robust Cuts (separation using code by J.Lysgaard [2004])
  - Rounded Capacity
  - Strengthened Comb
- Non-Robust Cuts
  - Subset Row
- Post-Enumeration Cuts
  - Cliques (separation using code by H.Santos [2012])

Pecin et al. [2014]: Pricing

Most critical part of the BCP: dynamic programming label setting algorithm, handling:

- *ng*-routes
- The modifications induced by Subset Row Cuts

Features:

- Bi-directional search (Righini and Salani [2006]), using balanced mid-point
- Completion Bounds with under-evaluation (Contardo [2012])
- Fixing by reduced cost considering arc accumulated load (Pessoa et al. [2010])

Pecin et al. [2014]: Non-Robustness Control

Even with all care, non-robust cuts are indeed "non-robust":

- Pricing may be handling hundreds of Subset Row Cuts well. Then, the separation of a few dozen additional cuts makes the pricing 100x or even 1000x slower!
- When such a situation is detected, the algorithm **rolls back**, removing the "bad" cuts.

Pecin et al. [2014]: Miscellanea

Hybrid search strategy, combining branching and enumeration:

- Strong Branching (SB):
  - Hierarchical, 3 levels
  - Keeps full history to guide future decisions
  - Up to 200 candidates per node.
- Route enumeration:
  - Generates pool with up to 20M routes
  - Branching can be done over enumerated node
  - Uses MIP solver (CPLEX) only when pool is reduced to less than 20K routes

Dual stabilization by smoothing (Wentges [1997])

Results in PPPU14

Class	#Ins	BMR11 US	BMR11 Gap	BMR11 T(s)	Rop12 US	Rop12 Gap	Rop12 T(s)	CM14 US	CM14 Gap	CM14 T(s)
A	22	0	0.13	30	0	0.57	53	0	0.09	59
B	20	0	0.06	67	0	0.25	208	0	0.08	34
E-M	12	3	0.49	303	2	0.96	44295	2	0.27	1548
F	3	1	0.11	164	0	0.25	2163	0	0.03	27772
P	24	0	0.23	85	0	0.69	280	0	0.18	240
Total	81	4			2			2		
Processor		Xeon X7350 2.93GHz			Core i7-2620M 2.7GHz			Xeon E5462 2.8GHz		

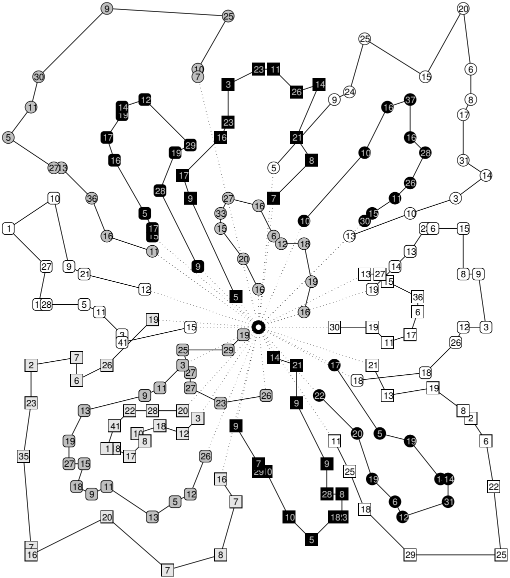
Class	#Ins	PPPU14 UnSolved	PPPU14 Gap	PPPU14 T(s)
A	22	0	0.03	5.6
B	20	0	0.04	6.2
E-M	12	0	0.19	3669
F	3	0	0.00	3679
P	24	0	0.07	33
Total	81	0		
Processor		Core i7-3770 3.4GHz		

Detailed Results: M-n200-k16

Algo	Root LB	Final LB	Total Time (s)
BMR11	1256.6	1256.6	319
Ropke12	1253.0	1258.2	7200
CM14	1266.9	1266.9	432000
PPPU14	1266.5	**1274**	39869

Previous best known solution: 1278.

Optimal solution M-n200-k16, cost: 1274



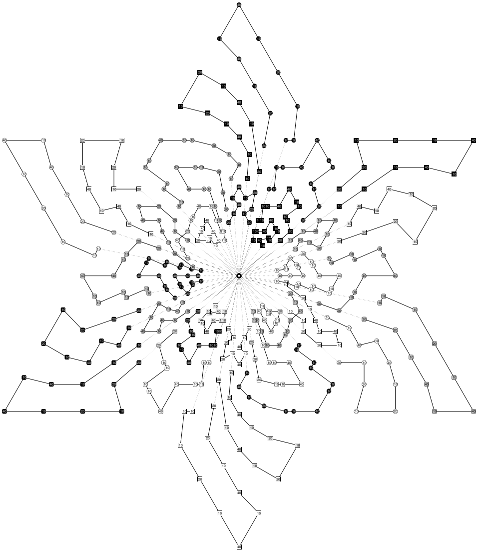
Golden Instances

Golden, Wasil, Kelly and Chao [1998] proposed 12 CVRP instances, having from 240 to 483 customers.

- Frequent in the heuristic literature
- Considered "out of reach" of exact algorithms

6 instances could be solved, with 240, 252, 300, 320, 360 and 420 customers.

Optimal solution Golden_20 (420 customers), 7 days CPU time, cost 1817.59; best heuristic 1817.86



The breakthrough in Pecin et al.[2014]: Limited Memory Cuts

The concept of limited memory cuts was pivotal for those improvements.

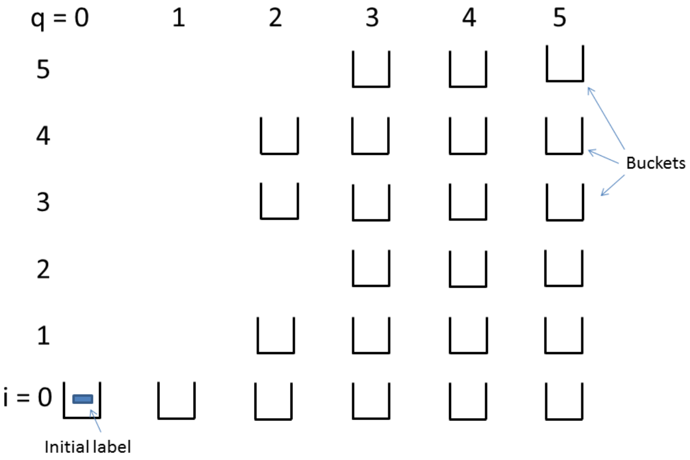
In order to understand the concept, it is necessary to see how non-robust cuts interfere with the pricing.

The Labeling Algorithm

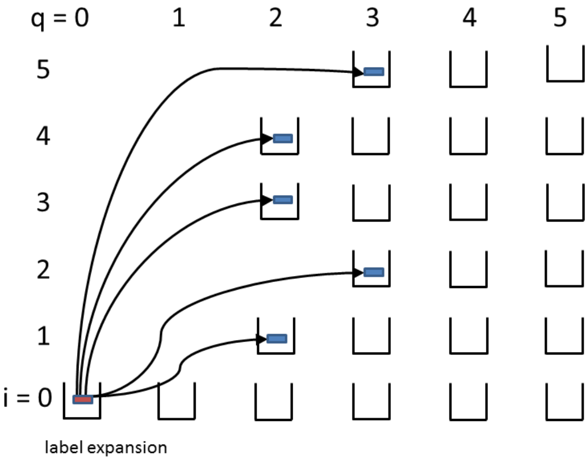
The pricing is done by a labeling dynamic programming algorithm.

- $B(i, q), i \in V, q \in \{d_i, \dots, Q\}$ are **buckets**
- A **label** $L(P)$ represents a partial path P, with cost $\bar{c}(P)$. All labels corresponding to paths ending in i with load q are kept in bucket $B(i, q)$.
- An initial label representing a null path is put in $B(0, 0)$.
- Labels are **expanded** producing other labels.
- Dominated labels should be removed along the algorithm.

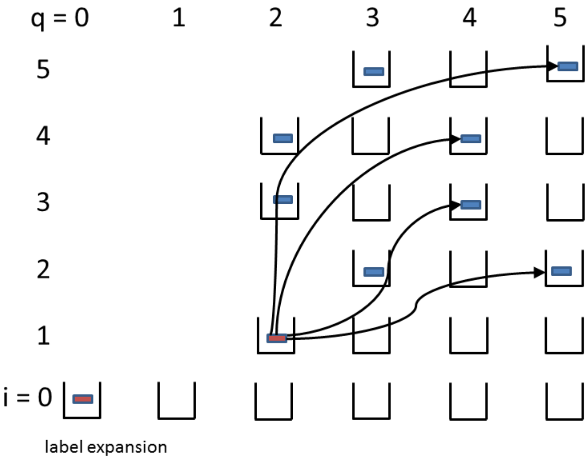
The Labeling Algorithm



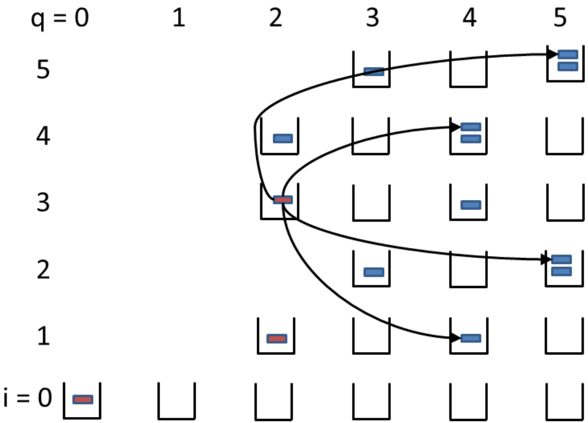
The Labeling Algorithm



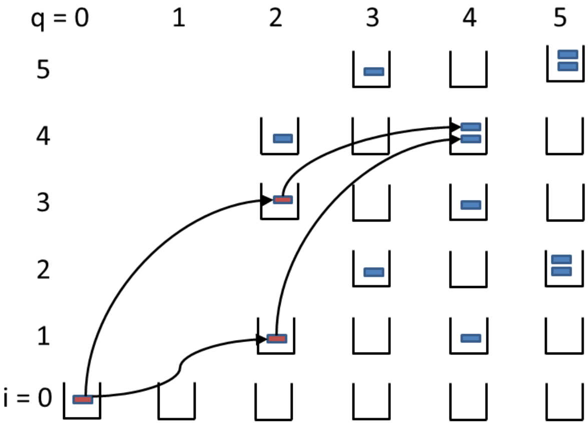
The Labeling Algorithm



The Labeling Algorithm

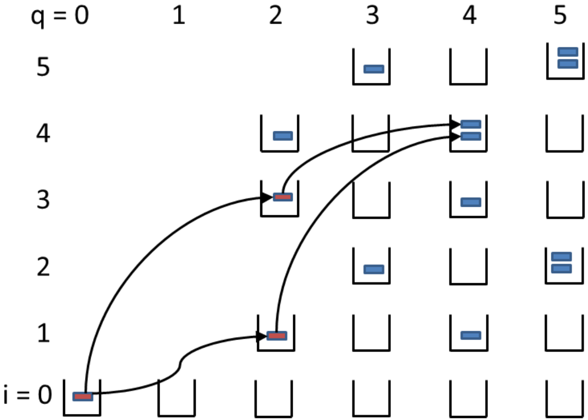


Do both labels need to be kept in bucket (4,4)?



The labels represent partial paths 0-1-4 and 0-3-4

The Labeling Algorithm



Do both labels need to be kept in bucket (4,4)? Depends on the definition of Ω

Domination in the Labeling Algorithm

Dominance Rule

A label $L(P_1)$ dominates another label $L(P_2)$ in the same bucket if $\bar{c}(P_1) \leq \bar{c}(P_2)$ and every valid completion of P_2 is also a valid completion for P_1.

- Ω allows cycles: at most one non-dominated label per bucket. Labeling runs in $O(n^2.Q)$ time (pseudo-polynomial).
- Only elementary routes in Ω: In the worst case, exponential number of non-dominated labels per bucket.
- ng-routes: Maximum of 2^{ng-1} labels per bucket. $ng = 8$ is a safe choice to avoid combinatorial explosion.

Subset Row Cuts (SRCs)

Let $C \subseteq N$ be a set of 3 customers. At most one route that passes by 2 or more customers in C can be used. So, the following 3-SRC is valid:

$$\sum_{r \in \Omega : |r \cap C| \geq 2} \lambda_r \leq 1$$

3-SRCs (and other SRCs) can reduce gaps a lot. However, they can also make the pricing intractable.

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

How 3-SRCs interfere with the Labeling Algorithm?

- Suppose that m 3-SRCs are added, $\sigma_s < 0$ is the dual variable of 3-SRC s, $C(s)$ its base set.
- Each label needs m additional binary dimensions, $S(P)[s]$ is the **parity** of the number of times that P visited customers in $C(s)$.
- When a label $L(P)$ with $S(P)[s] = 1$ is expanded to a vertex in $C(s)$, new label $L(P')$ has its reduced cost penalized by σ_s.

How 3-SRCs interfere with the Labeling Algorithm?

Dominance Rule with 3-SRCs

A label $L(P_1)$ dominates another label $L(P_2)$ in the same bucket if $\bar{c}(P_1) \leq \bar{c}(P_2) + \sum_{1 \leq s \leq m : S(P_1)[s] > S(P_2)[s]} \sigma_s$ and every valid completion of P_2 is also a valid completion for P_1.

- More 3-SRCs, weaker dominance
- In many instances, 100 3-SRCs are enough for a combinatorial explosion in the number of non-dominated labels

Limited Memory 3-Subset Row Cuts (lm-3-SRCs)

The 3-SRC with base-set $C \subseteq N$ is assigned to a memory-set M, $C \subseteq M \subseteq N$. The lm-3-SRC is:

$$\sum_{r \in \Omega} \alpha(C, M, r)\lambda_r \leq 1, \qquad (9)$$

where coefficient of route r is given by:

function $\alpha(C,\ M\ r)$
$\quad coeff \leftarrow 0$, $state \leftarrow 0$
for every vertex $i \in r$ (in order) **do**
$\quad$ **if** $i \notin M$ **then**
$\quad\quad state \leftarrow 0$
$\quad$ **else if** $i \in C$ **then**
$\quad\quad state \leftarrow state + 1/2$
$\quad\quad$ **if** $state \geq 1$ **then**
$\quad\quad\quad coeff \leftarrow coeff + 1$, $state \leftarrow state - 1$
return $coeff$

Limited Memory 3-Subset Row Cuts (lm-3-SRCs)

```
function α(C, M, r)
coeff ← 0, state ← 0
for every vertex i ∈ r (in order) do
    if i ∉ M then
        state ← 0
    else if i ∈ C then
        state ← state + 1/2
        if state ≥ 1 then
            coeff ← coeff + 1, state ← state − 1
return coeff
```

- If memory-set M is equal to N, the lm-3-SRC is equivalent to an 3-SRC. Otherwise, **it is weaker**.
- However, **by adjusting dynamically the memory-sets** M, it is possible to obtain exactly the same bounds obtainable by regular 3-SRCs.
- If the final M is small, as usually happens, the gains in pricing time are large.

Why limited memory reduces impact in the pricing?

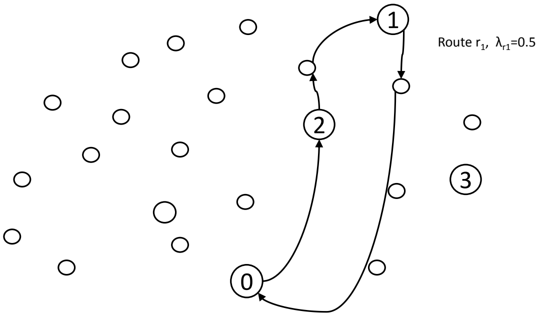
Dominance Rule with lm-3-SRCs

A label $L(P_1)$ dominates another label $L(P_2)$ in the same bucket if $\bar{c}(P_1) \leq \bar{c}(P_2) + \sum_{1 \leq s \leq m: S(P_1)[s] > S(P_2)[s]} \sigma_s$ and every valid completion of P_2 is also a valid completion for P_1.

Suppose a lm-3-SRC s with memory set $M(s)$ and dual variable σ_s. The dimension $S(P)[s]$ is set to zero whenever P visits a vertex not in $M(s)$ (so, the previous visits to $C(s)$ are "forgotten").

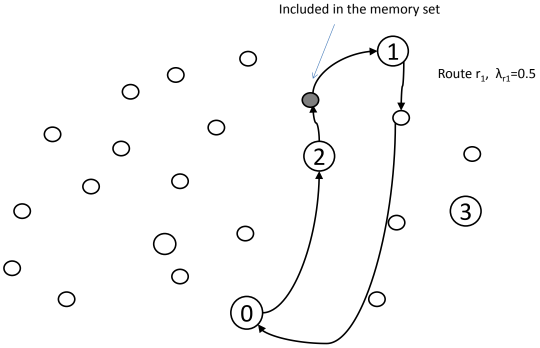
- Many more SRCs can be separated before pricing becomes intractable

Separation of lm-SRCs



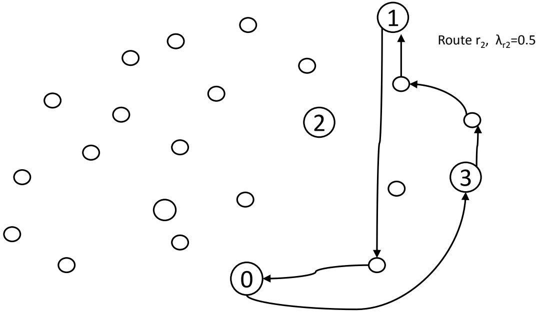
λ_{r1} has coefficient 1 in the 3-SRC with $C = \{1, 2, 3\}$

Separation of lm-SRCs



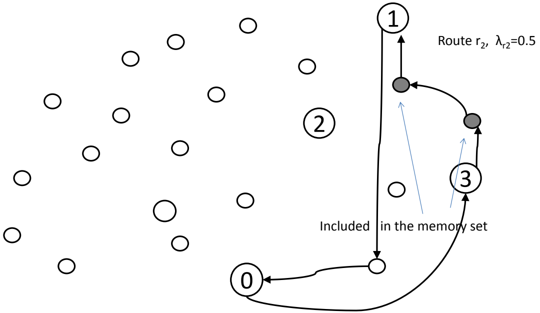
λ_{r1} still has coefficient 1 in the lm 3-SRC

Separation of lm-SRCs



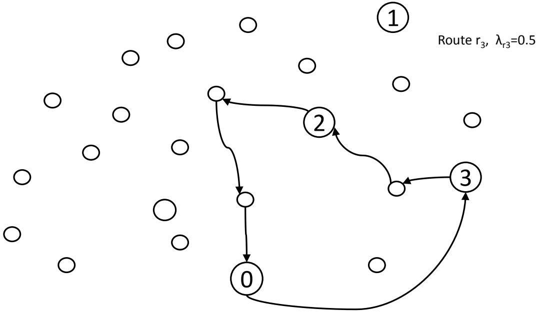
λ_{r2} has coefficient 1 in the 3-SRC with $C = \{1, 2, 3\}$

Separation of lm-SRCs



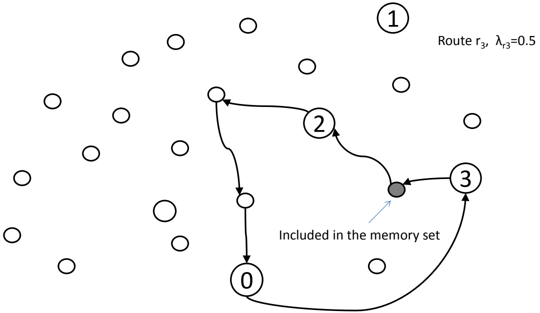
λ_{r2} still has coefficient 1 in the lm 3-SRC

Separation of lm-SRCs



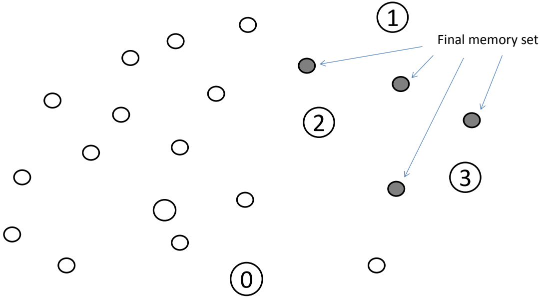
λ_{r3} has coefficient 1 in the 3-SRC with $C = \{1, 2, 3\}$

Separation of lm-SRCs

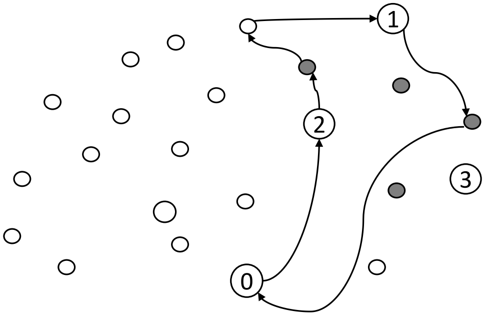


λ_{r3} still has coefficient 1 in the lm 3-SRC

Separation of lm-SRCs

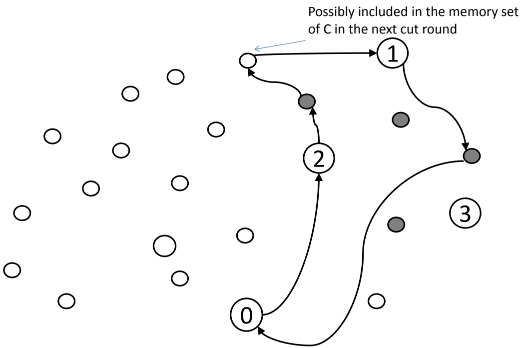


Separation of lm-SRCs



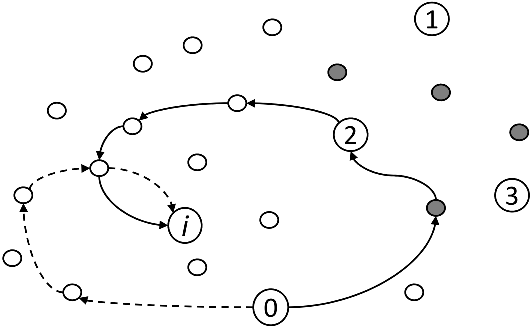
The next pricing iteration may generate routes that dodge the memory!

Separation of lm-SRCs



No problem, memory will be adjusted in the next cut round!

Why it is good to reduce the set M as much as possible?



Solid path may only dominate the dashed path because the lm 3-SRC $\{1, 2, 3\}$ is already forgotten at i.

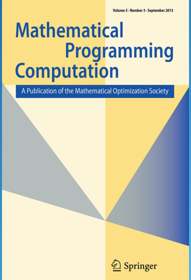
Advanced Branch-Cut-and-Price Algorithms

The advanced state-of-the-art BCPAs are those where the non-robust cuts and the pricing algorithm are jointly and symbiotically designed, in such a way that the pricing can handle a large number of very tailored non-robust cuts without becoming too inefficient.

- The presented **limited memory technique is good for the labeling algorithm**. If the pricing was being solved by another method (for example, by MIP), they would actually make the pricing harder!

Springer springer.com

This certificate is awarded at the 23rd International Symposium on Mathematical Programming, Bordeaux, France, July 2018



MPC Best Paper in 2017

The editorial board of MPC has chosen

Improved Branch-Cut-and-Price for Capacitated Vehicle Routing

by **Diego Pecin, Artur Pessoa, Marcus Poggi** and **Eduardo Uchoa**
MPC, volume 9, pp. 61-100, March 2017

Part II - From CVRP to other classic variants

Facets of the Set Partitioning polyhedron with Chvátal-Gomory rank 1.
Generalize and strengthen Subset Row Cuts.

D. Pecin, A. Pessoa, M. Poggi, Uchoa. E., and Haroldo Santos. Limited memory rank-1 cuts for vehicle routing problems. *Operations Research Letters*, 45(3):206–209, 2017b

T. Bulhoes, A. Pessoa, F. Protti, and Uchoa. E. On the complete set packing and set partitioning polytopes: Properties and rank 1 facets. *Operations Research Letters*, 46(4):389–392, 2018a

VRP with Time Windows (VRPTW)

Presents the concept of **memory cut over arcs**

Results on the classic Solomon instances (100 customers)

- All solved, 55/56 at the root node

Results on Gehring-Homberger instances (200 customers)

- 51/60 solved, 27 for the first time

D. Pecin, C. Contardo, G. Desaulniers, and Uchoa. E. New enhancements for exactly solving the vehicle routing problem with time windows. *INFORMS Journal on Computing*, 29:489–502, 2017a

Presents the concept of **memory cut depending on subproblem**

- Solves most instances with up to 200 clientes, two times more than previous methods

A. Pessoa, R. Sadykov, and Uchoa. E. Enhanced branch-cut-and-price algorithm for heterogeneous fleet vehicle routing problems. *European Journal of Operational Research*, 270:530–543, 2018

The classic multi-vehicle arc routing

Solves instances two times larger than previous methods.

- 23/24 Eglese instances, 11 for the first time
- 134/135 other instances from the literature

Diego Pecin and Eduardo Uchoa. Comparative analysis of capacitated arc routing formulations for designing a new branch-cut-and-price algorithm. *Transportation Science*, 53(6):1673–1694, 2019

Generic Exact VRPSolver

The difficulty of creating state-of-the-art algorithms for new VRP variants

Those BCP algorithms are very complex:

- Each of the previously mentioned algorithms took a lot of time to be coded, even when they adapt an already existing code for another variant
- There are intricate conceptual issues for adapting some techniques for more complex variants

One would like to have a generic algorithm that could be easily customized to many variants

Previous work (Desaulniers et al. [1998])

The SPF is quite generic, by changing the definition of Ω several classic variants can be modeled. Desaulniers et al. [1998] proposed a framework where many VRPs could be solved by the same BP algorithm. The sets Ω were associated to the solutions of **Resource Constrained Shortest Path (RCSP) Problems**.

$$\min \sum_{r\in\Omega} c_r \lambda_r \tag{10}$$

$$\text{S.t.} \quad \sum_{r\in\Omega} a_{ir}\lambda_r = 1, \qquad \forall\ i \in N, \tag{11}$$

$$\lambda_r \in \{0,1\} \qquad \forall\ r \in \Omega. \tag{12}$$

G. Desaulniers, J. Desrosiers, I. Ioachim, M. Solomon, F. Soumis, and D. Villeneuve. A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In *Fleet management and logistics*, pages 57–93. Springer, 1998

A BCP solver for a generic model that encompasses a wide class of VRPs and even some other kinds of problems

Incorporates almost all advanced elements found in the best recent VRP algorithms,

A. Pessoa, R. Sadykov, E. Uchoa, and F. Vanderbeck. A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, 183(1):483–523, 2020

Computational Experiments

- VRPSolver algorithms coded in C++ over BaPCod package (Vanderbeck et al. [2018])
- IBM CPLEX 12.8 used as LP solver
- Models are defined using a Julia–JuMP (Dunning et al. [2017]) based interface.

Tests over 13 problems: CVRP, VRPTW, HFVRP, Multi-Depot VRP (MDVRP), (Capacitated) Team Orienteering Problem (CTOP/TOP), Capacitated Profitable Tour Problem (CPTP), VRP with Service Level constraints (VRPSL), GAP, Vector Packing Problem (VPP), Bin Packing Problem (BPP) and CARP.

Computational results

Problem	Data set	#	T.L.	VRPSolver	Best Published		2nd Best Published	
CVRP	E-M	12	10h	12 (61s)	**12 (49s)**	Pecin et al. [2017c]	10 (432s)	Contardo et al. [2014]
	X	58	60h	**36 (147m)**	34 (209m)	Uchoa et al. [2017]	—	
VRPTW	Sol Hard	14	1h	**14 (5m)**	13 (17m)	Pecin et al. [2017a]	9 (39m)	Baldacci et al. [2011a]
	Hom 200	60	30h	**56 (21m)**	50 (70m)	Pecin et al. [2017a]	7 (-)	Kallehauge et al. [2006]
HFVRP	Golden	40	1h	**40 (144s)**	39 (287s)	Pessoa et al. [2018]	34 (855s)	Baldacci et al. [2009]
MDVRP	Cordeau	11	1h	**11 (6m)**	11 (7m)	Pessoa et al. [2018]	9 (25m)	Contardo et al. [2014]
PDPTW	RC	40	1h	**40 (5m)**	33 (17m)	Gschwind et al. [2018]	32 (14m)	Baldacci et al. [2011b]
	LiLim	30	1h	3 (56m)	**23 (20m)**	Baldacci et al. [2011b]	18 (27m)	Gschwind et al. [2018]
TOP	Chao 4	60	1h	**55 (8m)**	39 (15m)	Bianchessi et al. [2018]	30 (-)	El-Hajj et al. [2016]
CTOP	Archetti	14	1h	**13 (7m)**	7 (34m)	Archetti et al. [2013]	6 (35m)	Archetti et al. [2009]
CPTP	Archetti	28	1h	**24 (9m)**	0 (1h)	Bulhoes et al. [2018b]	0 (1h)	Archetti et al. [2013]
VRPSL	Bulhoes	180	2h	**159 (16m)**	49 (90m)	Bulhoes et al. [2018b]	—	
GAP	OR-Lib D	6	2h	5 (40m)	**5 (30m)**	Posta et al. [2012]	5 (46m)	Avella et al. [2010]
	Nauss	30	1h	**25 (23m)**	1 (58m)	Gurobi [2017]	0 (1h)	Nauss [2003]
VPP	1,4,5,9	40	1h	**38 (8m)**	13 (50m)	Heßler et al. [2018]	10 (53m)	Brandão et al. [2016]
BPP	Falk T	80	10m	80 (16s)	**80 (1s)**	Brandão et al. [2016]	80 (24s)	Belov et al. [2006,16]
	Hard28	28	10m	28 (17s)	**28 (7s)**	Belov et al. [2006,16]	26 (14s)	Brandão et al. [2016]
	AI	250	1h	**160 (25m)**	116 (35m)	Belov et al. [2006,16]	100 (40m)	Brandão et al. [2016]
	ANI	250	1h	**103 (35m)**	97 (40m)	Wei et al. [2019]	67 (45m)	Belov et al. [2006,16]
CARP	Eglese	24	30h	**22 (36m)**	22 (43m)	Pecin et al. [2019]	10 (237m)	Bartolini et al. [2013]

Table: VRPSolver vs best specific solvers on 13 problems.

Class X with 100 instances, ranging between 100 and 1000 customers:

- Designed to mimic a wide diversity of characteristics found in real applications
- Available at CVRPLIB (http://vrp.atd-lab.inf.puc-rio.br/index.php/en/)

E. Uchoa, D. Pecin, A. Pessoa, M. Poggi, T. Vidal, and A. Subramanian. New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research*, 257(3):845–858, 2017

VRPSolver on CVRP

53 out of 100 instances could be solved, sometimes with very special parameterization and very long runs (up to one month):

- $100 \leq n < 200$: 22/22 (100%)
- $200 \leq n < 300$: 19/21 (90%)
- $300 \leq n < 500$: 8/25 (32%)
- $500 \leq n \leq 1000$: 4/32 (12%)

Smallest unsolved: X-n280-k17

Largest solved: X-n856-k95

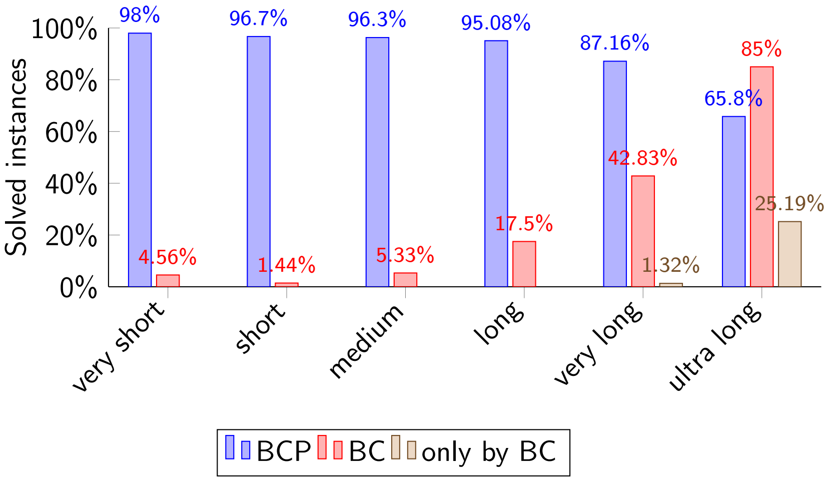
Optimal solution X-856-k95 (unitary demands), 10 days of CPU time, cost 88,965

A new benchmark of 10,000 instances with 100 customers

- CVRPLIB did not have any small/medium instances with > 25 customers/route. Realizing that limitation (after Amazon Last Mile Routing Challenge!), this recent benchmark included instances with "ultra-long" routes.
- VRPSolver could find all optimal solutions with a median time of 1 minute. Yet, 10.1% of instances could not be solved in 30 minutes using default settings. Solving them required special strategies and/or long runs (up to 80 hours).
- But we also reimplemented the best BC over CPLEX 20

E. Queiroga, R. Sadykov, Uchoa. E., and T. Vidal. 10,000 optimal CVRP solutions for testing machine learning based heuristics. *AAAI-22 Workshop on Machine Learning for Operations Research (ML4OR)*, 2022.

BC vs BCP (VRPSolver) by route size, TL 30 mins



BC for CVRP is not dead! However, BCs are hard to generalize.
For example, there is no reasonable BC for VRPTW

Downloading and using the VRP Solver

VRP Solver available over Julia–JuMP

The VRP solver is available for academic use (vrpsolver.math.u-bordeaux.fr):

- Algorithms are bundled in a single pre-compiled docker (runs in every OS)
- There is a no-docker version for Linux
- Julia–JuMP user interface for modeling, including several demos

JUMP

1. Modeling a typical VRP variant requires around 100 lines of Julia code (not counting input/output). An experienced user can build a working solver for a new variant in 1 day
2. Many computer experiments and parameter tuning may be needed for an improved performance
3. In some cases, separation routines for problem-specific (robust) cuts are needed for a better performance

```julia
function build_model(data::DataCVRP)
   E = edges(data)
   n = nb_customers(data)
   V = [i for i in 0:n]
   V⁺ = [i for i in 1:n]
   Q = veh_capacity(data)
   cvrp = VrpModel()
   @variable(cvrp.formulation, x[e in E], Int)
   @objective(cvrp.formulation, Min, sum(c(data,e) * x[e] for e in E))
   @constraint(cvrp.formulation, deg[i in V⁺], sum(x[e] for e in δ(data, i)) == 2.0)
   function build_graph()
      v_source = v_sink = 0
      G = VrpGraph(cvrp, V, v_source, v_sink, (0, n))
      cap_res_id = add_resource(G, main = true)
      for i in V
         set_resource_bounds(G, i, cap_res_id, 0, Q)
      end
      for (i,j) in E
         arc_id = add_arc(G, i, j, [x[(i,j)]])
         set_arc_consumption(G, arc_id, cap_res_id, d(data, j))
         arc_id = add_arc(G, j, i, [x[(i,j)]])
         set_arc_consumption(G, arc_id, cap_res_id, d(data, i))
      end
      return G
   end
   G = build_graph()
   add_graph(cvrp, G)
   set_vertex_packing_sets(cvrp, [[(G,i)] for i in V⁺])
   define_packing_sets_distance_matrix(cvrp, [[dist(data, (i, j)) for j in V⁺] for i in V⁺])
   add_capacity_cut_separator(cvrp, [ ( [(G,i)], d(data, i) ) for i in V⁺], Q)
   set_branching_priority(cvrp, "x", 1)
   return (cvrp, x)
end
```

Room for Creative Modeling

We believe users may find original ways (transformations) of fitting new problems in the proposed model

- Not only VRP variants, possibly also problems from scheduling, network design, etc.

Since VRP solving technology is quite advanced, there is a chance of obtaining better-than-existing-methods performance

Some works that used VRPSolver with creative modeling

T. Bulhões, R. Sadykov, A. Subramanian, and E. Uchoa. On the exact solution of a large class of parallel machine scheduling problems. *Journal of Scheduling*, 23(4):411–429, 2020

A. Pessoa, M. Poss, R. Sadykov, and F. Vanderbeck. Branch-cut-and-price for the robust capacitated vehicle routing problem with knapsack uncertainty. *Operations Research*, 69(3):739–754, 2021a

A. Pessoa, R. Sadykov, and E. Uchoa. Solving bin packing problems using VRPSolver models. *Operations Research Forum*, 2(2):1–25, 2021b

I. Mohamed, W. Klibi, R. Sadykov, H. Şen, and F. Vanderbeck. The two-echelon stochastic multi-period capacitated location-routing problem. *European Journal of Operational Research*, 2022

Some works that used VRPSolver with creative modeling

T. Adamo, G. Ghiani, P. Greco, and E. Guerriero. Properties and bounds for the single-vehicle capacitated routing problem with time-dependent travel times and multiple trips. In *10th Conference on Operations Research and Enterprise Systems (ICORES)*, pages 82–87, 2021

A. Subramanyam, T. Cokyasar, J. Larson, and M. Stinson. Joint routing of conventional and range-extended electric vehicles in a large metropolitan network. *Transportation Research Part C: Emerging Technologies*, 144:103830, 2022

C. Damião, J.M. Silva, and E. Uchoa. A branch-cut-and-price algorithm for the cumulative capacitated vehicle routing problem. *4OR*, pages 1–25, 2021

G. Volte, E. Bourreaua, R. Giroudeau, and O. Naud. Using VRPSolver to efficiently solve the differential harvest problem. *Computers & Operations Research*, 2022. to appear

Disclaimer on current VRPSolver

VRPSolver is being maintained by a tiny group of people working on spare time

- Documentation is poor, it is quite difficult to understand how to change parameters
- Docker version is stuck in old Julia and JuMP versions

We would like to have a community of users

Conclusions and Perspectives

Impact of Exact VRP algorithms in practice

Historically, exact solvers were rarely used in practical routing

1. Existing algorithms could not solve realistic-sized instances in reasonable times
   - Now many instances of the most classic VRPs with up to 200 customers can be solved
   - More importantly, instances with up to 100 customers can often be solved in a few minutes
2. The real problems seldom correspond exactly to one of the classic variants. Creating a good exact code for a new variant is a hard task
   - Highly customizable codes with state-of-the-art performance are now available

We expect that exact algorithms will be much more used by practitioners, at least for benchmarking their heuristics

Thank you

N. Achuthan, L. Caccetta, and S. Hill. An improved branch-and-cut algorithm for the capacitated vehicle routing problem. *Transportation Science*, 37:153–169, 2003.

T. Adamo, G. Ghiani, P. Greco, and E. Guerriero. Properties and bounds for the single-vehicle capacitated routing problem with time-dependent travel times and multiple trips. In *10th Conference on Operations Research and Enterprise Systems (ICORES)*, pages 82–87, 2021.

J. Araque, G. Kudva, T. Morin, and J. Pekny. A branch-and-cut algorithm for the vehicle routing problem. *Annals of Operations Research*, 50:37–59, 1994.

C. Archetti, D. Feillet, A. Hertz, and M G Speranza. The capacitated team orienteering and profitable tour problems. *Journal of the Operational Research Society*, 60(6):831–842, Jun 2009.

C. Archetti, N. Bianchessi, and M.G. Speranza. Optimal solutions for routing problems with profits. *Discrete Applied Mathematics*, 161(4–5):547–557, 2013.

P. Augerat, J. Belenguer, E. Benavent, A. Corberán, D. Naddef, and G. Rinaldi. Computational results with a branch and cut code for the capacitated vehicle routing problem. Technical Report 949-M, Université Joseph Fourier, Grenoble, France, 1995.

Pasquale Avella, Maurizio Boccia, and Igor Vasilyev. A computational study of exact knapsack separation for the generalized assignment problem. *Computational Optimization and Applications*, 45(3):543–555, 2010.

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.

R. Baldacci, A. Mingozzi, and R. Roberti. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*, 59:1269–1283, 2011a.

Roberto Baldacci and Aristide Mingozzi. A unified exact method for solving different classes of vehicle routing problems. *Mathematical Programming*, 120(2):347–380, 2009.

Roberto Baldacci, Enrico Bartolini, and Aristide Mingozzi. An exact algorithm for the pickup and delivery problem with time windows. *Operations Research*, 59(2):414–426, 2011b.

M.L. Balinski and R.E. Quandt. On an integer program for a delivery problem. *Operations Research*, 12(2):300–304, 1964.

Enrico Bartolini, Jean-François Cordeau, and Gilbert Laporte. Improved lower bounds and exact algorithm for the capacitated arc routing problem. *Mathematical Programming*, 137(1): 409–452, Feb 2013.

G. Belov and G. Scheithauer. A branch-and-cut-and-price algorithm for one-dimensional stock cutting and two-dimensional two-stage cutting. *European Journal of Operational Research*, 171(1):85 – 106, 2006.

Nicola Bianchessi, Renata Mansini, and M. Grazia Speranza. A branch-and-cut algorithm for the team orienteering problem. *International Transactions in Operational Research*, 25(2): 627–635, 2018.

U. Blasum and W. Hochstättler. Application of the branch and cut method to the vehicle routing problem. Technical Report ZPR2000-386, Zentrum fur Angewandte Informatik Köln, 2000.

Filipe Brandão and João Pedro Pedroso. Bin packing and related problems: General arc-flow formulation with graph compression. *Computers & Operations Research*, 69:56 – 67, 2016.

T. Bulhoes, A. Pessoa, F. Protti, and Uchoa. E. On the complete set packing and set partitioning polytopes: Properties and rank 1 facets. *Operations Research Letters*, 46(4):389–392, 2018a.

T. Bulhões, R. Sadykov, A. Subramanian, and E. Uchoa. On the exact solution of a large class of parallel machine scheduling problems. *Journal of Scheduling*, 23(4):411–429, 2020.

Teobaldo Bulhoes, Minh Hoàng Hà, Rafael Martinelli, and Thibaut Vidal. The vehicle routing problem with service level constraints. *European Journal of Operational Research*, 265(2):544 – 558, 2018b.

N. Christofides, A. Mingozzi, and P. Toth. The vehicle routing problem. In N. Christofides, A. Mingozzi, P. Toth, and C. Sandi, editors, *Combinatorial Optimization, A Wiley-Interscience Publication, Based on a series of lectures, given at the Summer School in Combinatorial Optimization, held in Sogesta, Italy, May 30th-June 11th, 1977, Chichester: Wiley, 1979, edited by Christofides, Nicos*, volume 1, pages 315–338, 1979.

C. Contardo. A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. Technical report, Archipel-UQAM 5078, Université du Québec à Montréal, Canada, 2012.

C. Contardo and R. Martinelli. A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. *Discrete Optimization*, 12:129–146, 2014a.

Claudio Contardo and Rafael Martinelli. A new exact algorithm for the multi-depot vehicle routing problem under capacity and route length constraints. *Discrete Optimization*, 12:129–146, 2014b.

L. Costa, C. Contardo, and G. Desaulniers. Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science*, 53:946–985, 2019.

C. Damião, J.M. Silva, and E. Uchoa. A branch-cut-and-price algorithm for the cumulative capacitated vehicle routing problem. *4OR*, pages 1–25, 2021.

G. Dantzig and R. Ramser. The truck dispatching problem. *Management Science*, 6:80–91, 1959.

George Dantzig, Ray Fulkerson, and Selmer Johnson. Solution of a large-scale traveling-salesman problem. *Journal of the operations research society of America*, 2(4):393–410, 1954.

G. Desaulniers, J. Desrosiers, I. Ioachim, M. Solomon, F. Soumis, and D. Villeneuve. A unified framework for deterministic time constrained vehicle routing and crew scheduling problems. In *Fleet management and logistics*, pages 57–93. Springer, 1998.

J. Desrosiers, F. Soumis, and M. Desrochers. Routing with time windows by column generation. *Networks*, 14(4):545–565, 1984.

Iain Dunning, Joey Huchette, and Miles Lubin. JuMP: A modeling language for mathematical optimization. *SIAM Review*, 59(2): 295–320, 2017.

Racha El-Hajj, Duc-Cuong Dang, and Aziz Moukrim. Solving the team orienteering problem with cutting planes. *Computers & Operations Research*, 74:21 – 30, 2016.

R. Fukasawa, H. Longo, J. Lysgaard, M. Poggi de Aragão, M. Reis, Uchoa. E., and R.F. Werneck. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming*, 106:491–511, 2006.

Timo Gschwind, Stefan Irnich, Ann-Kathrin Rothenbächer, and Christian Tilk. Bidirectional labeling in column-generation algorithms for pickup-and-delivery problems. *European Journal of Operational Research*, 266(2):521 – 530, 2018.

Gurobi. Gurobi optimizer reference manual, version 7.5, 2017. URL `http://www.gurobi.com`.

Katrin Heßler, Timo Gschwind, and Stefan Irnich. Stabilized branch-and-price algorithms for vector packing problems. *European Journal of Operational Research*, 271(2):401 – 419, 2018.

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.

B. Kallehauge, J. Larsen, and O.B.G. Madsen. Lagrangian duality applied to the vehicle routing problem with time windows. 33 (5):1464–1487, 2006.

G. Laporte and Y. Nobert. A branch and bound algorithm for the capacitated vehicle routing problem. *OR Spectrum*, 5(2):77–85, 1983.

J. Lysgaard, A. Letchford, and R. Eglese. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming*, 100:423–445, 2004.

I. Mohamed, W. Klibi, R. Sadykov, H. Şen, and F. Vanderbeck. The two-echelon stochastic multi-period capacitated location-routing problem. *European Journal of Operational Research*, 2022.

Denis Naddef and Giovanni Rinaldi. Branch-and-cut algorithms for the capacitated vrp. In *The vehicle routing problem*, pages 53–84. SIAM, 2002.

Robert M. Nauss. Solving the generalized assignment problem: An optimizing and heuristic approach. *INFORMS Journal on Computing*, 15(3):249–266, 2003.

D. Pecin and Uchoa. E. Comparative analysis of capacitated arc routing formulations for designing a new branch-cut-and-price algorithm. *Transportation Science*, (Forthcoming), 2019.

D. Pecin, A. Pessoa, M. Poggi, and Uchoa. E. Improved branch-cut-and-price for capacitated vehicle routing. In *Proceedings of the 17th IPCO*, pages 393–403. Springer, 2014.

D. Pecin, C. Contardo, G. Desaulniers, and Uchoa. E. New enhancements for exactly solving the vehicle routing problem with time windows. *INFORMS Journal on Computing*, 29: 489–502, 2017a.

D. Pecin, A. Pessoa, M. Poggi, Uchoa. E., and Haroldo Santos. Limited memory rank-1 cuts for vehicle routing problems. *Operations Research Letters*, 45(3):206–209, 2017b.

D. Pecin, A. Pessoa, M. Poggi, and E. Uchoa. Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation*, 9(1):61–100, 2017c.

Diego Pecin and Eduardo Uchoa. Comparative analysis of capacitated arc routing formulations for designing a new branch-cut-and-price algorithm. *Transportation Science*, 53(6): 1673–1694, 2019.

A. Pessoa, R. Sadykov, and Uchoa. E. Enhanced branch-cut-and-price algorithm for heterogeneous fleet vehicle routing problems. *European Journal of Operational Research*, 270:530–543, 2018.

A. Pessoa, R. Sadykov, E. Uchoa, and F. Vanderbeck. A generic exact solver for vehicle routing and related problems. *Mathematical Programming*, 183(1):483–523, 2020.

A. Pessoa, M. Poss, R. Sadykov, and F. Vanderbeck. Branch-cut-and-price for the robust capacitated vehicle routing problem with knapsack uncertainty. *Operations Research*, 69(3): 739–754, 2021a.

A. Pessoa, R. Sadykov, and E. Uchoa. Solving bin packing problems using VRPSolver models. *Operations Research Forum*, 2(2):1–25, 2021b.

M. Poggi and Uchoa. E. New exact approaches for the capacitated VRP. In P. Toth and D. Vigo, editors, *Vehicle Routing: Problems, Methods, and Applications*, chapter 3, pages 59–86. SIAM, second edition, 2014.

Marius Posta, Jacques A. Ferland, and Philippe Michelon. An exact method with variable fixing for solving the generalized assignment problem. *Computational Optimization and Applications*, 52:629–644, 2012.

T. Ralphs. Parallel branch and cut for capacitated vehicle routing. *Parallel Computing*, 29:607–629, 2003.

T. Ralphs, L. Kopman, W. Pulleyblank, and L. Trotter Jr. On the capacitated vehicle routing problem. *Mathematical Programming*, 94:343–359, 2003.

G. Righini and M. Salani. Symmetry helps: bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization*, 3(3):255–273, 2006.

S. Røpke. Branching decisions in branch-and-cut-and-price algorithms for vehicle routing problems. *Presentation in Column Generation 2012*, 2012.

A. Subramanyam, T. Cokyasar, J. Larson, and M. Stinson. Joint routing of conventional and range-extended electric vehicles in a large metropolitan network. *Transportation Research Part C: Emerging Technologies*, 144:103830, 2022.

E. Uchoa, D. Pecin, A. Pessoa, M. Poggi, T. Vidal, and A. Subramanian. New benchmark instances for the capacitated vehicle routing problem. *European Journal of Operational Research*, 257(3):845–858, 2017.

François Vanderbeck, Ruslan Sadykov, and Issam Tahiri. BaPCod — a generic Branch-And-Price Code, 2018. URL `https://realopt.bordeaux.inria.fr/?page_id=2`.

G. Volte, E. Bourreaua, R. Giroudeau, and O. Naud. Using VRPSolver to efficiently solve the differential harvest problem. *Computers & Operations Research*, 2022. to appear.

Laguna Wei, Zhixing Luo, Roberto Baldacci, and Andrew Lim. A new branch-and-price-and-cut algorithm for one-dimensional bin-packing problems. *INFORMS Journal on Computing*, Forthcoming, 2019.

Paul Wentges. Weighted dantzig–wolfe decomposition for linear mixed-integer programming. *International Transactions in Operational Research*, 4(2):151–162, 1997.