



Invited Review

Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints

Roberto Baldacci^a, Aristide Mingozzi^{b,*}, Roberto Roberti^c^a DEIS, University of Bologna, Via Venezia 52, 47521 Cesena, Italy^b Department of Mathematics, University of Bologna, Via Sacchi 3, 47521 Cesena, Italy^c DEIS, University of Bologna, Viale Risorgimento 2, 40136 Bologna, Italy

ARTICLE INFO

Article history:

Received 19 January 2011

Accepted 22 July 2011

Available online 6 August 2011

Keywords:

Vehicle routing

Exact algorithms

Survey

ABSTRACT

This paper provides a review of the recent developments that had a major impact on the current state-of-the-art exact algorithms for the vehicle routing problem (VRP). The paper reviews mathematical formulations, relaxations and recent exact methods for two of the most important variants of the VRP: the capacitated VRP (CVRP) and the VRP with time windows (VRPTW). The paper also reports a comparison of the computational performances of the different exact algorithms for the CVRP and VRPTW.

© 2011 Elsevier B.V. All rights reserved.

1. Introduction

Vehicle routing problem (VRP) is a generic name given to a whole class of problems involving the design of optimal routes for a fleet of vehicles to service a set of customers subject to side constraints. The VRP is a central problem in the physical delivery of goods and services. In practice, several variants of the VRP exist, depending on the nature of the transported goods, the quality of service required, and the characteristics of customers and vehicles. Some typical complications are heterogeneous vehicles located at different depots, customers incompatible with certain vehicle types, customers accepting delivery within specified time windows, multiple-day planning horizons and vehicles performing multiple routes. In all cases, the objective is to supply the customers at minimum cost.

The simplest and most studied member of the VRP family is the *capacitated VRP* (CVRP). In the CVRP, a fleet of identical vehicles located at a central depot has to be optimally routed to supply a set of customers with known demands. Each vehicle can perform at most one route and the total demand of the customers visited by a route cannot exceed the vehicle capacity. Another important variant of the VRP is the *VRP with time windows* (VRPTW) that generalizes the CVRP by imposing that each customer is visited within a specified time interval, called *time window*.

Several heuristics have been proposed for the CVRP and its variants in the literature. For surveys of both exact and heuristic methods for these VRPs up to 2002, the reader is referred to [Toth and Vigo \(2002\)](#).

In this paper, we survey the most effective exact methods proposed for the CVRP and the VRPTW in the literature in the last ten years. Section 2 reviews exact methods for the CVRP while Section 3 deals with exact methods for the VRPTW. Conclusions are reported in Section 4.

2. The capacitated vehicle routing problem (CVRP)

The CVRP can be described as follows. An undirected graph $G = (V, E)$ is given where $V = \{0, 1, \dots, n\}$ is the set of $n + 1$ vertices and E is the set of edges. Vertex 0 represents the depot, and the vertex set $V = V \setminus \{0\}$ corresponds to n customers. A nonnegative cost d_{ij} is associated with each edge $\{i, j\} \in E$. Each customer $i \in V$ requires a supply of q_i units of goods from depot 0 (we assume $q_0 = 0$). A set of m identical vehicles of capacity Q is stationed at the depot and must be used to supply the customers. A route is defined as a least-cost elementary cycle $R = (0, i_1, \dots, i_h, 0)$ of graph G passing through the depot and such that the total demand of the customers visited does not exceed the vehicle capacity. The cost $c(R)$ of a route is equal to the cost of the solution to the Traveling Salesman Problem (TSP) defined by the set R of vertices.

The CVRP is to design m routes of minimum cost, one for each vehicle, so that all customers are visited exactly once; the CVRP is \mathcal{NP} -hard as it is a natural generalization of the TSP.

Many formulations have been proposed for the CVRP (see [Achuathan and Caccetta, 1991](#); [Toth and Vigo, 2002](#); [Kara et al., 2004](#)), but not all of them have been used to derive exact algorithms. Currently, the most successful exact methods for the CVRP are based on the *two-index flow formulation*, the *two-commodity flow formulation*

* Corresponding author.

E-mail addresses: r.baldacci@unibo.it (R. Baldacci), mingozzi@csr.unibo.it (A. Mingozzi), roberto.roberti@unibo.it (R. Roberti).

proposed by Baldacci et al. (2004), and the *set partitioning formulation* proposed by Balinski and Quandt (1964). These algorithms are described in Sections 2.1 and 2.2. In Section 2.3, a comparison of the computational performances of these algorithms is reported.

2.1. Branch-and-cut algorithms

The branch-and-cut (BC) algorithms for the CVRP are based either on the two-index flow or the two-commodity flow formulation.

Let $S = \{S : S \subseteq V, |S| \geq 2\}$, and let $q(S) = \sum_{i \in S} q_i$ be the total demand of customers in $S \in \mathcal{S}$ and $k(S)$ the minimum number of vehicles of capacity Q needed to service all customers in S . Moreover, let $\delta(S)$ be the *cutset* defined by S (i.e., $\delta(S) = \{i, j\} \in E : i \in S, j \notin S$ or $i \notin S, j \in S$).

Let x_{ij} be an integer variable that takes value in $\{0, 1\}$, for all $\{i, j\} \in E \setminus \{0, j\} : j \in V$, and value in $\{0, 1, 2\}$, for all $\{0, j\}, j \in V$, with $x_{ij} = 1$ when edge $\{i, j\}$ is traversed and $x_{0j} = 2$ when the single customer route $(0, j, 0)$ is in the solution.

The two-index vehicle flow formulation of the CVRP is as follows.

$$(F) \quad z(F) = \min \sum_{\{i, j\} \in E} d_{ij} x_{ij} \quad (1)$$

$$\text{s.t.} \quad \sum_{\{i, j\} \in \delta(\{h\})} x_{ij} = 2, \quad \forall h \in V, \quad (2)$$

$$\sum_{\{i, j\} \in \delta(S)} x_{ij} \geq 2k(S), \quad \forall S \in \mathcal{S}, \quad (3)$$

$$\sum_{j \in V} x_{0j} = 2m, \quad (4)$$

$$x_{ij} \in \{0, 1\}, \quad \forall \{i, j\} \in E \setminus \{0, j\} : j \in V, \quad (5)$$

$$x_{0j} \in \{0, 1, 2\}, \quad \forall \{0, j\}, j \in V. \quad (6)$$

Constraints (2) are the degree constraints for each customer, and constraints (3) are the *capacity constraints*. Whenever $k(S) = q(S)/Q$, constraints (3) are called *fractional capacity inequalities*. Constraint (4) states that m vehicles must leave and return to the depot.

The LP-relaxation of F can be strengthened if $k(S)$ is computed as $k(S) = \lceil q(S)/Q \rceil$, where $\lceil x \rceil$ denotes the smallest integer not less than x . In this case, constraints (3) are called *rounded capacity constraints*. Several authors have proposed valid inequalities to reinforce the lower bound achieved by the LP-relaxation of F . Some of these inequalities are derived by extending, to formulation F , the successful results of polyhedral combinatorics developed for the TSP (see Grötschel and Padberg, 1985).

Augerat et al. (1995) were the first to describe an exact BC algorithm for the CVRP based on formulation F strengthened by valid inequalities such as the *generalized capacity constraints*, *hypotour inequalities*, *comb inequalities*, *path-bin inequalities*. The BC algorithm of Augerat et al. (1995) was able to solve, for the first time, a CVRP instance involving 135 customers. An improved version of the BC algorithm of Augerat et al. (1995) is described in Naddef and Rinaldi (2002).

Ralphs et al. (2003) described a BC algorithm based on the two-index formulation and on the addition of rounded capacity constraints in a cutting plane fashion.

Lygaard et al. (2004) described a BC algorithm based on formulation F , strengthened by valid inequalities, including the *rounded capacity*, *generalized capacity*, *framed capacity*, *strengthened comb*, *multistar*, *partial multistar*, *extended hypotour inequalities*, and *Gomory mixed integer cuts*. Their BC algorithm solved several instances not solved by Augerat et al. (1995).

Baldacci et al. (2004) proposed a two-commodity flow formulation of the CVRP which extends the TSP model introduced by Finke et al. (1984). This formulation requires, for each edge $\{i, j\} \in E$, a binary variable x_{ij} that is equal to 1 if the edge is in the solution,

and two variables y_{ij} and y_{ji} representing the vehicle load and the empty vehicle space (i.e., $y_{ji} = Q - y_{ij}$) on edge $\{i, j\}$, whenever $x_{ij} = 1$. The resulting model involves $(2n + |E| + 3)$ constraints. It can be shown that (i) any solution (x, y) of the LP-relaxation of this model is also a feasible solution of the LP-relaxation of F when $k(S) = q(S)/Q$ is used in constraints (3), and (ii) (x, y) satisfies the generalized multistar inequalities not implied by F .

Baldacci et al. (2004) described a BC algorithm based on this model using only rounded capacity inequalities and reported computational results showing that their BC algorithm is competitive with the algorithm of Naddef and Rinaldi (2002).

2.2. Algorithms based on the set partitioning formulation

The Set Partitioning (SP) formulation of the CVRP was originally proposed by Balinski and Quandt (1964) and associates a binary variable with each feasible route. The Balinski and Quandt formulation is the following.

Let \mathcal{R} be the index set of all routes. Let $a_{i\ell}$ be a binary coefficient equal to 1 if vertex $i \in V$ belongs to route $\ell \in \mathcal{R}$ and 0 otherwise (note that $a_{0\ell} = 1, \forall \ell \in \mathcal{R}$). Each route $\ell \in \mathcal{R}$ has an associated cost c_ℓ which is the sum of the costs of the edges traversed. Let ξ_ℓ be a binary variable that is equal to 1 if route $\ell \in \mathcal{R}$ is in the optimal solution and 0 otherwise.

Model SP is as following:

$$(SP) \quad z(SP) = \min \sum_{\ell \in \mathcal{R}} c_\ell \xi_\ell \quad (7)$$

$$\text{s.t.} \quad \sum_{\ell \in \mathcal{R}} \xi_\ell \leq m, \quad (8)$$

$$\sum_{\ell \in \mathcal{R}} a_{i\ell} \xi_\ell = 1, \quad \forall i \in V, \quad (9)$$

$$\xi_\ell \in \{0, 1\}, \quad \forall \ell \in \mathcal{R}. \quad (10)$$

Constraint (8) requires that at most m routes are selected, and constraints (9) specify that each customer $i \in V$ must be covered by one route.

In the following, we use R_ℓ and $E(R_\ell)$ to indicate the subset of vertices of graph G visited and the edges traversed by route $\ell \in \mathcal{R}$, respectively.

Formulation SP cannot be used directly to solve nontrivial CVRP instances because of the large number of potential routes. Model SP is very general and can take into account several route constraints (e.g., time windows) because the route feasibility is implicitly considered in the definition of the route set \mathcal{R} .

The optimal solution cost of the LP-relaxation of SP, called *LSP*, dominates the lower bounds provided by the LP-relaxation of formulation F when $k(S) = q(S)/Q$ (see Baldacci et al., 2004) because any LSP solution implicitly satisfies a variety of valid inequalities that are not implied by the LP-relaxation of F . In particular, Baldacci et al. (2004) have shown that LSP implies *fractional capacity*, *generalized large multistar* and *knapsack large multistar* inequalities. Letchford and Salazar González (2006) have shown that LSP implies some *hypotour-like inequalities* by projection.

Moreover, Baldacci et al. (2004) have shown that any SP solution ξ can be transformed into an F solution x by setting:

$$x_{ij} = \sum_{\ell \in \mathcal{R}} \eta_{ij}^\ell \xi_\ell, \quad \forall \{i, j\} \in E, \quad (11)$$

where the coefficients η_{ij}^ℓ are defined as follows: (i) if ℓ is a single customer route $(0, h, 0)$ then $\eta_{0h}^\ell = 2$ and $\eta_{ij}^\ell = 0, \forall \{i, j\} \in E \setminus \{0, h\}$; (ii) if ℓ is not a single customer route, then $\eta_{ij}^\ell = 1, \forall \{i, j\} \in E(R_\ell)$ and $\eta_{ij}^\ell = 0, \forall \{i, j\} \in E \setminus E(R_\ell)$.

Let \mathcal{F} be the family of valid inequalities for model F that can be expressed in a general form as:

$$\sum_{(i,j) \in E} \alpha_{ij}^t x_{ij} \geq \beta^t, \quad t \in \mathcal{F}. \quad (12)$$

Thus, using Eq. (11), inequalities (12) become the following inequalities for the SP model:

$$\sum_{(i,j) \in E} \alpha_{ij}^t \eta_{ij}^t \zeta_\ell \geq \beta^t, \quad t \in \mathcal{F}. \quad (13)$$

2.2.1. The exact algorithm of Fukasawa et al. (2006)

Fukasawa et al. (2006) described an exact algorithm based on the SP model where the variables correspond to the set of q -routes, introduced by Christofides et al. (1981), while the constraints correspond to the set partitioning constraints (8), (9) and valid inequalities (13), such as *rounded capacity inequalities*, *framed capacity*, *strengthened comb*, *multistar*, *partial multistar*, *generalized large multistar* and *hypotour inequalities*, all presented in Lysgaard et al. (2004) for formulation F .

A (q, i) -path is a nonnecessarily elementary path, starting from the depot, visiting a set of vertices (without 2-vertex loops) of total demand equal to q , and ending at vertex i . Let $f(q, i)$ be the cost of a least-cost (q, i) -path; functions $f(q, i)$ can be computed by using dynamic programming (DP) (see Christofides et al., 1981). A q -route is a $(q, 0)$ -path of cost $f(q, 0)$.

Because the resulting formulation has an exponential number of both columns and rows, Fukasawa et al. used a column-and-cut generation (CCG) method to compute the lower bound and a branch-and-cut-and-price (BCP) algorithm to solve the CVRP. Their exact method decides at the root node, according to the best balance between running time and bound quality, to solve the CVRP either using the BC method of Lysgaard et al. (2004) or their BCP algorithm.

2.2.2. The exact algorithm of Baldacci et al. (2008)

Baldacci et al. (2008) proposed an exact algorithm based on model SP, strengthened by the following two types of valid inequalities.

2.2.2.1. Strengthened capacity inequalities. These inequalities are obtained by lifting the rounded capacity constraints of F expressed in the form of inequalities (13). Let $\mathcal{R}(S)$ be the index subset of routes visiting at least one customer of set $S \in \mathcal{S}$. The strengthened capacity inequalities are:

$$\sum_{\ell \in \mathcal{R}(S)} \zeta_\ell \geq \lceil q(S)/Q \rceil, \quad \forall S \in \mathcal{S}. \quad (14)$$

2.2.2.2. Clique inequalities. Let $H = (\mathcal{R}, \mathcal{E})$ be the *conflict graph* where each vertex corresponds to a route and the edge set \mathcal{E} contains an edge $\{\ell, \ell'\}$, $\forall \ell, \ell' \in \mathcal{R}$, $\ell < \ell'$, such that $R_\ell \cap R_{\ell'} \neq \{0\}$. Let \mathcal{C} be the set of all cliques of H .

Clique inequalities are:

$$\sum_{\ell \in \mathcal{C}} \zeta_\ell \leq 1, \quad \forall \mathcal{C} \in \mathcal{C}. \quad (15)$$

Let SP denote the problem obtained by adding strengthened capacity and clique inequalities to problem SP , and let LSP denote the LP-relaxation of SP . Let $\mathbf{u} = (u_0, u_1, \dots, u_n)$ be the vector of dual variables associated with constraints (8) and (9), where $u_0 \leq 0$ and $u_i \in \mathcal{R}$, $i \in V$. Moreover, let $v_S \geq 0$, $S \in \mathcal{S}$, and $g_C \leq 0$, $C \in \mathcal{C}$, be the dual variables of constraints (14) and (15), respectively.

The exact method of Baldacci et al. (2008) can be described as follows.

1. Solve relaxation LSP , and let $(\mathbf{u}', \mathbf{v}', \mathbf{g}')$ be an optimal dual solution of LSP of cost z' .
2. Define the reduced problem \widehat{SP} resulting from SP as follows: (i) replace the route set \mathcal{R} with the largest subset $\widehat{\mathcal{R}} \subseteq \mathcal{R}$ such that $c'_\ell < z_{UB} - z'$, $\ell \in \widehat{\mathcal{R}}$, where c'_ℓ is the reduced cost of route ℓ with

respect to $(\mathbf{u}', \mathbf{v}', \mathbf{g}')$ and z_{UB} is a valid upper bound on the CVRP; (ii) add all constraints (14) and (15) whose associated slacks are not in the optimal basis of LSP .

3. Solve problem \widehat{SP} using a general purpose integer programming solver.

The effectiveness of this method relies on the quality of the dual solution $(\mathbf{u}', \mathbf{v}', \mathbf{g}')$ achieved as the size of subset $\widehat{\mathcal{R}}$ depends on the gap $z_{UB} - z'$.

The core of the algorithm of Baldacci et al. (2008) is the method for solving SP . They propose to solve LSP using three CCG procedures, called H^1 , H^2 and H^3 , that produce three lower bounds LB_1 , LB_2 and LB_3 (such that $LB_1 \leq LB_2 \leq LB_3$) corresponding to the costs of three different dual solutions of LSP . The three procedures are executed in sequence, and the dual solution produced by procedure H^k is used to generate the master problem of procedure H^{k+1} , $k = 1, 2$.

The first two procedures H^1 and H^2 ignore clique inequalities (15) and use rounded capacity constraints instead of strengthened capacity inequalities (14). H^1 replaces the route set \mathcal{R} with the set of all q -routes whereas H^2 uses elementary routes instead of q -routes. The CCG method used by H^1 and H^2 differs from standard CCG algorithms based on the simplex as it uses a dual ascent heuristic to find a near-optimal dual solution of the master problem. H^3 is a CCG method based on the simplex to solve LSP including both inequalities (14) and (15), that inherits the master problem from H^2 . In practice, H^3 is fast as it requires few iterations to converge to an optimal dual solution of LSP .

A key component of the algorithm of Baldacci et al. (2008) is the procedure, called GENROUTE, that solves the pricing problem in H^2 and H^3 and generates the final reduced problem \widehat{SP} .

Let \mathcal{P} be the set of all elementary paths of minimum cost from the depot such that $q(P) \leq \lceil Q/2 \rceil + q_{\sigma(P)}$, $\forall P \in \mathcal{P}$, where $q(P) = \sum_{i \in V(P)} q_i$, $\sigma(P)$ and $V(P)$ represent the load, the terminal customer and the subset of visited customers of path P , respectively.

GENROUTE is a two-phase procedure based on the observation that every feasible route can be obtained by combining a pair of paths $P, \bar{P} \in \mathcal{P}$ such that: (a) $\sigma(P) = \sigma(\bar{P})$; (b) $V(P) \cap V(\bar{P}) = \{0, \sigma(P)\}$; (c) $q(P) + q(\bar{P}) \leq Q + q_{\sigma(P)}$. GENROUTE uses bounding functions, based on the q -path relaxation, to limit the size of the path set \mathcal{P} , by avoiding the generation of paths that cannot be in any optimal CVRP solution and an algorithm for generating the routes which avoids the enumeration of all path pairs.

2.2.3. The exact algorithm of Baldacci et al. (in press)

Baldacci et al. (in press) extended the exact method of Baldacci et al. (2008) to solve the CVRP and VRPTW by means of the following improvements.

1. **ng-route relaxation.** The *ng*-route relaxation is a new state-space relaxation that improves the q -path and t -path relaxations proposed for the CVRP and VRPTW, respectively. This relaxation provides *ng*-routes, that are used as an alternative to q -routes in procedure H^1 and in computing better bounding functions used in procedure GENROUTE. The *ng*-path relaxation consists of partitioning the set of all possible q -paths ending at vertex i according to a mapping function that associates with each q -path a subset NG of the vertices visited that depends on the order in which they are visited. Let $N_i \subseteq V$ be a set of selected customers for vertex i (according to some criterion) such that $N_i \ni i$ and $|N_i| \leq \Delta(N_i)$, where $\Delta(N_i)$ is a parameter. For a given path $P = (0, i_1, \dots, i_k)$, let $\Pi(P)$ be the subset of $V(P)$ containing customer i_k and every customer i_r , $r = 1, \dots, k-1$, of P that belongs to all sets $N_{i_{r+1}}, \dots, N_{i_k}$. An *ng*-path (NG, q, i) is a nonnecessarily elementary path $P = (0, i_1, \dots, i_{k-1}, i_k)$ starting from the depot, visiting a subset of

customers of total demand equal to q such that $NG = \Pi(P)$, ending at customer i_k , and such that $i_k \notin \Pi(P')$, where $P' = (0, i_1, \dots, i_{k-1})$. Let $f(NG, q, i)$ the cost of a least-cost ng -path (NG, q, i) . Any $(NG, q, 0)$ -path is called ng -route. Functions $f(NG, q, i)$ can be computed using DP recursions.

2. **Subset-Row (SR3) inequalities.** Jepsen et al. (2008) introduced Subset-Row (SR3) inequalities for the VRPTW. SR3s represent a subset of clique inequalities. Let $C \subseteq \{C \subseteq V : |C| = 3\}$ be the subset of all customer triplets and $\mathcal{R}(C)$ be the subset of routes serving at least two customers in C . Subset-Row (SR3) inequalities are:

$$\sum_{\ell \in \mathcal{R}(C)} \xi_{\ell} \leq 1, \quad \forall C \in \mathcal{C}. \quad (16)$$

3. **Weak Subset-Row (WSR3) inequalities.** WSR3s are a weaker version of SR3s. Given $C \in \mathcal{C}$, the route set $\mathcal{R}(C)$ contains only those routes traversing at least an edge $\{i, j\}$ with $i, j \in C$. WSR3 dual variables can be easily considered in solving the pricing problem and in computing the bounding functions used by GENROUTE. The SR3s and, thus, the WSR3s are separated by complete enumeration. Both SR3s and WSR3s are used instead of clique inequalities.
4. **A new pricing strategy involving multiple dual solutions.** Baldacci et al. (in press) propose a new algorithm, more efficient than GENROUTE, to generate routes of negative reduced costs in H^3 that also uses the dual solution achieved by H^2 to eliminate routes that cannot be in any optimal solution. Such improvement stabilizes the CCG procedure based on the simplex and improves the final lower bound achieved by H^3 .

2.3. Computational comparison of exact algorithms for the CVRP

In this section, we report a computational comparison of the results obtained by Lysgaard et al. (2004) (hereafter called LLE), Fukasawa et al. (2006) (FLL) and the two algorithms of Baldacci et al. (2008) (BCM) and Baldacci et al. (in press) (BMR), on six well-known classes of CVRP instances from the literature, called A, B, E, F, M and P, and available at <http://branchandcut.org/VRP/data>.

The algorithm of BMR was run on an IBM Intel Xeon X7350 Server (2.93 GHz–16 GB of RAM). According to SPEC (<http://www.spec.org/benchmarks.html>), BMR machine is three times faster than the Pentium 4 2.6 GHz PC of BCM and the Pentium 4 2.4 GHz PC of FLL and 10 times faster than the Intel Celeron 700 MHz PC of LLE.

Table 1 reports, for each class, the name (Class), the number of instances (NP) and, for each exact method, the number of instances solved to optimality (Opt), the average percentage deviation of the lower bound (%LB) and the average computing time in seconds (Time). For FLL, column Opt_{BC} reports the number of instances solved to optimality by using the BC of Lysgaard et al., and column Opt_{BCP} reports the number of instances solved by their BCP. The last two lines of Table 1 report the total number of instances solved by each method and the averages of lower bounds and computing times.

BCM was not able to solve to optimality 3 instances solved by FLL and by BMR. Table 1 indicates that the lower bounds of BCM and BMR are on average superior to the lower bounds of FLL in all classes of instances considered. Notice that FLL solved 78 instances to optimality, but 29 of them were solved using the BC of Lysgaard et al. (2004). Table 1 indicates that BMR is, on average faster than the other methods.

3. The vehicle routing problem with time windows (VRPTW)

The VRPTW is defined on a complete digraph $G = (V, A)$, where $V = \{0, 1, \dots, n\}$ is the vertex set and A is the arc set. Associated with each arc (i, j) included in A is a travel cost d_{ij} and a travel time $t_{ij} > 0$, where t_{ij} includes the service time at vertex i . It is assumed that matrices d_{ij} and t_{ij} satisfy the triangle inequality. Associated with each vertex i included in V is a demand q_i and a time window $[e_i, l_i]$, where e_i and l_i represent the earliest and latest time to visit vertex i . The time windows are assumed hard.

A fleet of m identical vehicles of capacity Q stationed at the depot has to fulfill customer demands. A vehicle route $R = (0, i_1, \dots, i_r, 0)$, with $r \geq 1$, is a simple circuit in G , passing through the depot, visiting vertices $V(R) = \{0, i_1, \dots, i_r\}$, $V(R) \subseteq V$, and such that the following holds:

- The total demand of visited customers does not exceed the vehicle capacity Q .
- The vehicle leaves the depot 0 at time e_0 , visits each customer in $V(R)$ within its time window, and returns to the depot before l_0 .
- If the vehicle arrives at $i \in V(R)$ before e_i , the service is delayed to time e_i .

The cost of route R is equal to the sum of the travel costs of the arc set, $A(R)$, traversed by route R .

The VRPTW consists of designing at most m routes of minimum total cost such that each customer is visited exactly once. As the VRPTW reduces to the CVRP if $e_i = 0$ and $l_i = \infty$, $\forall i \in V$, the VRPTW is \mathcal{NP} -hard. Indeed, it is strongly \mathcal{NP} -complete to find a feasible solution for the VRPTW if $m < n$.

Several exact algorithms have been presented for the VRPTW. A review of the exact methods up to 2002 is reported in Cordeau et al. (2002) and Kallehauge (2008). The best exact methods recently published on the VRPTW are based on model SP , described in Section 2.2, where the route set \mathcal{R} contains any least-cost route satisfying time windows constraints.

Model SP can be strengthened by any valid inequality studied for the CVRP, as discussed in Section 2, and by the k -path inequalities introduced by Kohl et al. (1999). The Kohl et al. strategy can be defined as follows.

Let $k(S)$ be the minimum number of routes needed to service the customer subset $S \in \mathcal{S}$. Kohl et al. (1999) called k -path inequalities the following generalization of the rounded capacity constraints:

Table 1
Computational comparison of the exact methods for the CVRP.

Class	BMR				BCM			FLL				LLE			
	NP	Opt	%LB	Time	Opt	%LB	Time	Opt	Opt _{BCP}	Opt _{BC}	%LB	Time	Opt	%LB	Time
A	22	22	99.9	30	22	99.8	118	22	20	2	99.2	1961	15	97.9	6638
B	20	20	99.9	67	20	99.8	417	20	6	14	99.5	4763	19	99.4	8178
E-M	12	9	99.8	303	8	99.4	1025	9	7	2	98.9	126,987	3	97.7	39,592
F	3	2	100.0	164				3	0	3	99.9	2398	3	99.9	1046
P	24	24	99.8	85	22	99.7	187	24	16	8	99.2	2892	16	97.7	11,219
Avg			99.9	92		99.7	323				99.3	17,409		98.4	9935
Tot	81	77			72			78	49	29			56		

Table 2

Computational comparison of the exact methods for the VRPTW.

Class	n	NP	Solved			Time		
			BMR	JPSP	DHL	BMR	JPSP	DHL
C2	50	8	8	7	n.a.	8	79	n.a.
RC2	50	8	8	7	n.a.	27	268	n.a.
R2	50	11	11	9	n.a.	124	7086	n.a.
C1	100	9	9	9	9	25	468	18
RC1	100	8	8	8	8	276	11,004	2150
R1	100	12	12	12	12	251	27,412	2327
C2	100	8	8	7	8	40	2795	2093
RC2	100	8	8	5	6	3767	3204	15,394
R2	100	11	10	4	8	28,680	35,292	63,068

$$\sum_{\ell \in \mathcal{R}} b_{\ell}(S) \xi_{\ell} \geq k(S), \quad \forall S \in \mathcal{S}, \quad (17)$$

where $b_{\ell}(S) = |\{(i, j) \in A(R_{\ell}) : i \in V \setminus S, j \in S\}|$. These inequalities can be lifted by defining $b_{\ell}(S) = 1$ if $V(R_{\ell}) \cap S \neq \emptyset$, and $b_{\ell}(S) = 0$ otherwise.

The separation algorithm consists of solving a VRPTW where only the customers in S need to be serviced and the objective is to minimize the number of vehicles used. For this reason, most researchers considered the 2-path inequalities that consist of finding any subset $S \in \mathcal{S}$, such that $\lceil q(S)/Q \rceil = 1$ and in solving a TSP with time windows (TSPTW) for the set $S \cup \{0\}$. When the TSPTW is infeasible, then $k(S) = 2$ provides a valid 2-path inequality for S . Other valid inequalities, related to k -path inequalities and called *Reachability cuts*, have been investigated by [Lygaard \(2006\)](#).

The exact algorithms for solving the resulting SP model of the VRPTW use CCG methods for computing the lower bound and either branch-and-price (BP) or BCP algorithms to find an optimal integer solution. The key component of these algorithms is the method for solving the pricing problem. This algorithm consists of finding a number of VRPTW routes of negative reduced cost with respect to the duals of the SP constraints (8), (9) and of the different inequalities separated during the previous iterations. This problem is solved using different DP strategies to find either non-elementary or elementary routes.

The main difficulty faced by both BP and BCP algorithms is how to take into account the duals of the different cuts in solving the pricing problem. The duals of inequalities (8), (9) (say u_0 and $u_i \in \mathcal{R}$, $i \in V$) and the duals of inequalities defined according to expressions (13) (say, w_t , $t \in \mathcal{F}$) can be associated with the arcs of a route. Thus, it is easy to observe that the reduced cost c'_{ℓ} of a route $\ell \in \mathcal{R}$ is equal to $\sum_{(i,j) \in A(R_{\ell})} d'_{ij}$ by defining the modified arc costs d'_{ij} as $d'_{ij} = d_{ij} - \frac{1}{2}(u_i + u_j) - \sum_{t \in \mathcal{F}} \alpha_{ij}^t w_t$.

It is much more complex to solve the pricing problem when considering set partitioning inequalities such as clique, subset row, lifted versions of rounded capacity and k -path inequalities. A possible way is to associate a binary state variable for each of these inequalities in the DP algorithm where each variable represents the dual of the associated inequality in the reduced cost of the emerging path. Unfortunately, the size of the resulting state-space graph grows and the resulting DP algorithm becomes computationally slow.

[Kohl et al. \(1999\)](#) described a BP algorithm which improves the BP of [Desrochers et al. \(1992\)](#) by adding 2-path inequalities to LP-relaxation of the SP formulation and by using nonelementary routes without 2-vertex loops in solving the pricing problem. [Irnich and Villeneuve \(2006\)](#) improved the BP of [Kohl et al. \(1999\)](#) by using a sophisticated labelling algorithm to forbid routes having k -vertex loops with $k \geq 3$. The numerical experiments on some hard VRPTW instances generated by [Solomon \(1987\)](#) show that k -cycle elimination, with $k \geq 3$, can substantially improve the lower bounds. The new labelling algorithm has proven to be a key ingredient for solving to optimality more than 15 unsolved Solo-

mon instances with 25, 50, and 100 customers. Nonetheless, this method fails to solve several instances with 25 customers.

Algorithms based on the computation of elementary routes were proposed by [Feillet et al. \(2004\)](#), [Danna and Le Pape \(2005\)](#) and [Chabrier \(2006\)](#). [Righini et al. \(2006\)](#) proposed a DP method, called *decremental state-space* algorithm. This algorithm solves the pricing problem by forcing the routes to visit the customers of a selected subset $\bar{V} \subseteq V$ at most once. The least-cost route is computed using a label-setting method that expands both forward and backward paths from the depot and connects routes in the middle.

A significant contribution has been given by [Jepsen et al. \(2008\)](#). They extended the BCP framework by adding the SR3 inequalities (described in Section 2.2.3) to the SP master problem. The SR3 inequalities provide better lower bounds but increase the complexity of the pricing problem. The pricing problem is solved to optimality by using standard label setting techniques where each SR3 inequality is represented by an additional state variable. To improve the performance of the label-setting algorithm, they introduced a modified dominance criterion that handles the reduced cost calculation in a reasonable way. Moreover, to reduce the computing time, they attempted to solve the pricing problem heuristically. If no route with negative reduced cost is found, then the pricing is solved to optimality. The computational results indicate that the algorithm of [Jepsen et al. \(2008\)](#) outperforms those of [Irnich and Villeneuve \(2006\)](#) and [Chabrier \(2006\)](#) and solves eight previously unsolved instances. However, the algorithm did not succeed in solving four instances previously solved by different authors.

The BCP of [Jepsen et al. \(2008\)](#) was improved by [Desaulniers et al. \(2008\)](#) by adding both SR3 and generalized k -path inequalities and using a tabu search heuristic, before using DP, to rapidly generate negative reduced cost routes. Their method outperforms all other algorithms, remarkably decreasing the computational time on Solomon instances with 100 customers, and solving 5 of the 10 open Solomon instances. The authors do not report any evidence of the performance of their method in solving the Solomon instances with 25 and 50 customers.

The exact method of [Baldacci et al. \(in press\)](#) described in Section 2.2.3 for the CVRP also solves the VRPTW by simply using different route relaxations to consider the time window constraints. These route relaxations are based on the (t, i) -path and the (NG, t, i) -path relaxations.

A (t, i) -path is a nonnecessarily elementary path, without 2-vertex loops, starting from the depot at time e_0 , visiting a set of customers (even more than once) within their time windows, and ending at vertex i at time $e_i \leq t \leq l_i$.

Let $f(t, i)$ be the cost of a least-cost (t, i) -path. A t -route, for a given $e_0 \leq t \leq l_0$, is defined as the $(t, 0)$ -path of cost $f(t, 0)$. Functions $f(t, i)$ can be computed using DP (see [Christofides et al., 1981](#)).

A (NG, t, i) -path is a nonnecessarily elementary path $P = (0, i_1, \dots, i_{k-1}, i_k)$ starting from the depot at time e_0 , visiting a subset of customers (even more than once) within their time windows such that $NG = \Pi(P)$, ending at customer i_k at time $e_{i_k} \leq t \leq l_{i_k}$, and such that $i_k \notin \Pi(P')$, where $P' = (0, i_1, \dots, i_{k-1})$.

Let $f(NG, t, i)$ be the cost of a least-cost (NG, t, i) -path. The ng -routes for the VRPTW corresponds to the $(NG, t, 0)$ -path $e_0 \leq t \leq l_0$, $NG \subseteq V'$, of cost $f(NG, t, 0)$. Capacity constraints are ignored in both (t, i) -path and $(NG, t, 0)$ -path relaxations.

3.1. Computational comparison of exact algorithms for the VRPTW

In this section, we report the computational comparison of the exact methods of [Jepsen et al. \(2008\)](#) (hereafter called JPSP), [Desaulniers et al. \(2008\)](#) (DHL) and [Baldacci et al. \(in press\)](#) (BMR) on six classes of instances (C1, RC1, R1, C2, RC2 and R2) (see [Solomon, 1987](#)).

BMR was run on an IBM Intel Xeon X7350 Server (2.93 GHz–16 GB of RAM). According to SPEC (<http://www.spec.org/benchmarks.html>), the machine used by BMR is three times faster than the Intel Pentium 4 3.0-GHz PC of JPSP and twice as fast as the Linux PC Dual Core AMD Opteron at 2.6 GHz of DHL. For each class, Table 2 reports the class name (*Class*), the number of customers (*n*), the number of instances (*NP*), the number of instances solved by each of the three methods (*Solved*) and the average computing time in seconds (*Time*) (*n.a.* means data is not available).

The table shows that BMR solved all but one Solomon instance and closed four open instances. Moreover, BMR outperforms JPSP and DHL: all instances solved by the other methods were solved by BMR and the average time is significantly lower. Instances of classes C2, RC2 and R2 involving 100 customers are more difficult than instances of classes C1, RC1 and R1 of the same dimension as they feature wide time windows.

4. Conclusions

In the last decade, some innovative exact approaches for vehicle routing problems have been proposed, producing a significant improvement on the size of the instances that can be solved to optimality. Indeed, these algorithms have brought more than one hundred the number of customers that may be handled. The key factor of the success of these approaches is the effective combination of the set partitioning formulation with families of cuts into column generation based algorithms. This approach has significantly improved the quality of the lower bounds that are now very close to the optimal solution values. Furthermore, set partitioning based approaches proved quite general as they are easily able to incorporate additional constraints arising in practical applications.

References

- Achuthan, N.R., Caccetta, L., 1991. Integer linear programming formulation for a vehicle routing problem. *European Journal of Operational Research* 52 (1), 86–89.
- Augerat, P., Belenguer, J.M., Benavent, E., Corberán, A., Naddef, D., Rinaldi, G., 1995. Computational results with a branch and cut code for the capacitated vehicle routing problem. Tech. Rep. 1 RR949-M, ARTEMIS-IMAG, Grenoble France.
- Baldacci, R., Hadjiconstantinou, E., Mingozzi, A., 2004. An exact algorithm for the capacitated vehicle routing problem based on a two-commodity network flow formulation. *Operations Research* 52, 723–738.
- Baldacci, R., Christofides, N., Mingozzi, A., 2008. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Mathematical Programming Series A* 115 (2), 351–385.
- Baldacci, R., Mingozzi, A., Roberti, R., in press. New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research*.
- Balinski, M., Quandt, R., 1964. On an integer program for a delivery problem. *Operations Research* 12, 300–304.
- Chabrier, A., 2006. Vehicle routing problem with elementary shortest path based column generation. *Computers & Operations Research* 33, 2972–2990.
- Christofides, N., Mingozzi, A., Toth, P., 1981. Exact algorithms for the vehicle routing problem based on spanning tree and shortest path relaxations. *Mathematical Programming* 20 (1), 255–282.
- Cordeau, J.-F., Desaulniers, G., Desrosiers, J., Solomon, M., Soumis, F., 2002. The vrp with time windows. In: Toth, P., Vigo, D. (Eds.), *The Vehicle Routing Problem*, vol. 9. SIAM Monographs on Discrete Mathematics and Applications, Philadelphia, pp. 157–193.
- Danna, E., Le Pape, C., 2005. Column generation. In: Desaulniers, G., Desrosiers, J., Solomon, M.M. (Eds.), *Accelerating Branch-and-price with Local Search: A Case Study on the Vehicle Routing Problem with Time Windows*. Springer, New York, pp. 90–130.
- Desaulniers, G., Lessard, F., Hadjar, A., 2008. Tabu search, partial elementarity, and generalized k-path inequalities for the vehicle routing problem with time windows. *Transportation Science* 42 (3), 387–404.
- Desrochers, M., Desrosiers, J., Solomon, M., 1992. A new optimization algorithm for the vehicle-routing problem with time windows. *Operations Research* 40 (2), 342–354.
- Feillet, D., Dejax, P., Gendreau, M., Gueguen, C., 2004. An exact algorithm for the elementary shortest path problem with resource constraints: Application to some vehicle routing problems. *Networks* 44, 216–229.
- Finke, G., Claus, A., Gunn, E., 1984. A two-commodity network flow approach to the traveling salesman problem. *Congress. Numerantium* 41, 167–178.
- Fukasawa, R., Longo, H., Lysgaard, J., Poggi de Aragão, M., Reis, M., Uchoa, E., Werneck, R., 2006. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Mathematical Programming Series A* 106, 491–511.
- Grötschel, M., Padberg, M., 1985. Polyhedral theory. In: Lawler, E., Lenstra, J., Rinnooy Kan, A., Shmoys, D. (Eds.), *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. John Wiley & Sons Ltd., Chichester, pp. 231–305.
- Irnich, S., Villeneuve, D., 2006. The Shortest-Path problem with resource constraints and k-Cycle elimination for $k \geq 3$. *INFORMS Journal on Computing* 18 (3), 391–406.
- Jepsen, M., Petersen, B., Spoorendonk, S., Pisinger, D., 2008. Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research* 56 (2), 497–511.
- Kallehauge, B., 2008. Formulations and exact algorithms for the vehicle routing problem with time windows. *Computers & Operations Research* 35 (7), 2307–2330.
- Kara, I., Laporte, G., Bektas, T., 2004. A note on the lifted Miller–Tucker–Zemlin subtour elimination constraints for the capacitated vehicle routing problem. *European Journal of Operational Research* 158 (3), 793–795.
- Kohl, N., Desrosiers, J., Madsen, O.B.G., Solomon, M.M., Soumis, F., 1999. 2-path cuts for the vehicle routing problem with time windows. *Transportation Science* 33 (1), 101–116.
- Letchford, A., Salazar González, J., 2006. Projection results for vehicle routing. *Mathematical Programming Series B* 105 (2–3), 251–274.
- Lysgaard, J., 2006. Reachability cuts for the vehicle routing problem with time windows. *European Journal of Operational Research* 175 (1), 210–223.
- Lysgaard, J., Letchford, A.N., Eglese, R.W., 2004. A new branch-and-cut algorithm for the capacitated vehicle routing problem. *Mathematical Programming Series A* 100, 423–445.
- Naddef, D., Rinaldi, G., 2002. Branch-and-cut algorithms for the capacitated VRP. In: Toth, P., Vigo, D. (Eds.), *The Vehicle Routing Problem*, vol. 9. SIAM Monographs on Discrete Mathematics and Applications, pp. 53–81.
- Ralphs, T., Kopman, L., Pulleyblank, W., Trotter, L., 2003. On the capacitated vehicle routing problem. *Mathematical Programming Series B* 94, 343–359.
- Righini, G., Salani, M., 2006. Symmetry helps: Bounded bi-directional dynamic programming for the elementary shortest path problem with resource constraints. *Discrete Optimization* 3 (3), 255–273.
- Solomon, M.M., 1987. Algorithms for the vehicle routing and scheduling problem with time window constraints. *Operations Research* 35, 234–265.
- Toth, P., Vigo, D. (Eds.), 2002. *The Vehicle Routing Problem*. Monographs on Discrete Mathematics and Applications. SIAM, Philadelphia, PA.