Citation for published version:
Salani, M & Battarra, M 2018, 'The opportunity cost of time window violations', Euro Journal on Transportation and Logistics, vol. 7, no. 4, pp. 343-361. https://doi.org/10.1007/s13676-018-0121-3

DOI:
10.1007/s13676-018-0121-3

Publication date:
2018

Document Version
Peer reviewed version

Noname manuscript No.
(will be inserted by the editor)

The opportunity cost of time window violations

Matteo Salani · **Maria Battarra**

Received: date / Accepted: date

Abstract This paper studies a variant of the Vehicle Routing Problem with Soft Time Windows (VRPSTW), inspired by real world distribution problems. In applications, violations of the prescribed delivery time are commonly accepted. Customers' inconvenience due to early or late arrival is typically modelled as a penalty cost included in the VRPSTW objective function, added to the routing costs. However, weighting routing costs against customer inconvenience is not straightforward for practitioners. In our problem definition, practitioners evaluate solutions by comparison with the hard time windows solution (referred as nominal solution). The desired routing cost saving is set by the practitioners as a percentage of the nominal solution's routing costs. The objective function minimises the time window violations, or the customer inconvenience, with respect to the nominal solution. This allows practitioners to quantify the opportunity cost (i.e., the customer inconvenience), when a target routing cost saving is imposed. We propose two exact algorithms: the first is based on a standard branch-and-cut-and-price, the second is a branch-and-cut-and-price nested in a bi-section algorithm. Computational results demonstrate that the second algorithm outperforms the standard implementation. Solutions obtained with the opportunity cost interpretation of soft time windows are then compared with solutions obtained using both hard time windows and the standard interpretation of soft time windows.

Keywords Vehicle Routing · Soft Time Windows · Branch-and-cut-and-price · Oppourtunity cost

M. Salani
Istituto dalle Molle di studi deull'Intelligenza Artificale (IDSIA), Scuola Universitaria professionale della Svizzera Italiana (SUPSI), Università della Svizzera Italiana (USI).
E-mail: matteo.salani@idsia.ch

M.Battarra
School of Management, University of Bath, BA2 7AY, UK.
Tel.: +44 (0)1225 384731
E-mail: m.battarra@bath.ac.uk

1 Introduction

Route planning is a critical task in the logistic industry and it has been estimated to account for up to 20% of the overall logistic cost (Toth and Vigo, 2002). The Vehicle Routing Problem (VRP) was proposed more than 50 years ago (Dantzig and Ramser, 1959) and it is a challenging combinatorial optimization problem. “Rich” VRP variants have been proposed to respond to the variety of operational constraints arising in the distribution industry (i.e., multiple depots, heterogeneous fleet, multi-trips, ...). Academics propose effective exact and heuristic algorithms responding to the needs of the distribution industry (Toth and Vigo, 2002; Golden et al, 2008; Vidal et al, 2012).

One of the real-world features attracting a remarkable attention from researchers and practitioners are the so called *time windows constraints* (Schrage, 1981; Gendreau and Tarantilis, 2010). Customers receiving goods often demand delivery within a time interval or *time window*. Time windows are classified as *hard*, if customers must be visited within the specified time interval, and *soft*, if time windows can be violated at the expense of customer inconvenience. In the first case, the problem is usually refereed as the Vehicle Routing Problem with Hard Time Windows (VRPHTW), in the latter the Vehicle Routing Problem with Soft Time Windows (VRPSTW).

Time window violations are often accepted in practical applications do to the potential routing cost saving. Figure 1 depicts an instance where the cost of traversing arcs $(1,3)$ and $(2,4)$ is 2 units, whereas all other arcs have cost 1 unit. The vehicle departs from the depot (the square vertex in the figure) at time zero and should visit each customer within the corresponding time window (minimum and maximum arrival times at a customer are reported in square brackets). We assume zero service time at the customers. The time window of customer 2 imposes visits at earliest at time 4 and this forces the vehicle to a costly routing solution, when hard time windows constraints are imposed (figure on the left). The same instance may result in a much more convenient routing solution, if customer 2 is visited two units before the desired time window. Practitioners may favour the soft time windows solution (given two units routing cost saving) or the hard time window solution (if inconvenience at customer 2 is not acceptable).

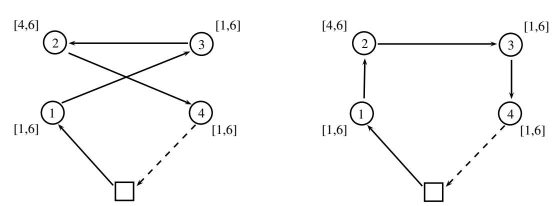

Fig. 1 Time windows violation at customer 2.

The VRPHTW was first proposed by Pullen and Webb (1967) and since then it has been widely studied. An interested reader may refer to the surveys of Cordeau et al (2002), Bräysy and Gendreau (2005a,b), Kallehauge (2008), Gendrau and Tarantilis (2010), Desaulniers et al (2010), Baldacci et al (2012) and Desaulniers et al

(2014) for comprehensive literature reviews on the VRPHTW. The VRPHTW has been solved effectively both by exact and heuristic algorithms and high quality solutions are achieved in relatively limited computing times.

A survey and critical discussion of the VRPSTW literature will be presented in Section 2; however the VRPSTW received less attention and there is not a unique interpretation of time windows violations. Customer inconvenience for being visited too late and (possibly) too early with respect to the desired time window is quantified and the VRPSTW's objective function is typically modelled as a weighted combination of routing costs and a measure of the customer inconvenience. However, how to measure customer inconvenience and how to quantify the relative weight of routing and customer inconvenience are still open research questions.

Quantifying customer inconvenience as opposed to routing costs is not a simple task for practitioners. Our previous collaborations with logistic companies made us realize that VRPSTW objective functions often do not reflect the standard practice in handling soft time windows constraints. In Ruinelli et al (2012), the handmade routing plans presented a large number of time windows violations and practitioners privileged routing cost optimization versus customer satisfaction. Moreover the experience of human planners and their knowledge about customer needs allowed them to discriminate among sensible time-windows violations and routing benefits associated with each violation. Finally, regarding the practice of objective function weighting in multi-objective optimization, the planner is often "not aware of which weights are the most appropriate to retrieve a satisfactorily solution, he/she does not know in general how to change weights to consistently change the solution" Caramia and Dell' Olmo (2008).

In this paper, we model soft time windows constraints trying to overcome the difficulties of practitioners in comparing routing costs and customer's inconvenience. A base of comparison is established, by setting as benchmark the optimal solution of the hard time windows problem. The planner sets a desired mileage saving with respect to the nominal solution. The exact algorithm minimizes the time window violations to achieve this goal. Therefore we model the opportunity cost of solutions with smaller routing cost, that is the incurred customer inconvenience due to time windows violations. This new variant makes the use of VRPSTW software more intuitive for planners. The parameter they are asked to define is simply a measure of desired mileage saving. Moreover, the iterative use of the software allows practitioners to generate solutions with the desired routing costs and to compare scenarios in which customer satisfaction is privileged with respect to routing costs and vice-versa.

The exact algorithms we propose are both based on branch-and-cut-and-price. The first is a standard implementation, the second is a branch-and-cut-and-price nested in a bisection framework. The solutions obtained are compared with those produced by similar algorithms for the VRPHTW and the VRPSTW (as defined in Liberatore et al (2011)).

The reminder of the paper is as follows. Section 2 presents a comprehensive literature review on the VRPSTW. In Section 3, we provide a mathematical model for our problem variant. We propose an exact algorithm based on branch-and-price-and-cut in Section 4. Section 5 presents computational experiments performed on a test set derived from the well known Solomon's data set for the VRPTW Desaulniers et al (1997). Final remarks and future research directions are reported in Section 6.

2 VRPSTW: Literature review

Early VRPSTWs have been presented in the pioneering articles by Sexton and Bodin (1985a,b) and Sexton and Choi (1986). Ferland and Fortin (1989) proposed a heuristic adjusting time windows of pairs of customers so to reduce the overall cost. Min (1991) modelled the problem of goods distribution among libraries using a multi-objective mixed integer linear programming model. In these article, extra mileage with respect to the ideal TSP for each route should be counterbalanced by a suitable reduction in the inconvenience for late arrival or early departure at the customers. More precisely, customer inconvenience is a included in the objective function as a linear weighted penalization. Dumas et al (1990) modelled customer inconvenience as a convex function and proposed an algorithm for optimizing departure and arrival time at each customer in a single route. The complexity of the resulting scheduling problems when costs are convex, linear and quadratic is discussed.

Koskosidis et al (1992) presented a cluster-first route-second heuristic for the VRPSTW in which linear penalizations apply for early and late arrival at customer locations. Balakrishnan (1993) studied a variant in which the service at a customer can start early and tardy, but within an outer time window. Linear penalizations apply when the visit is early/tardy with respect to the inner time window. If the vehicle is early with respect to the outer time window, waiting time at a customer is allowed but bounded. Fast constructive heuristics are proposed, namely a nearest neighbour, a saving heuristic and a space-time heuristic.

In Chang and Russell (2004), a Tabu Search is developed for the same problem studied in Balakrishnan (1993). Taillard et al (1997) also proposed a Tabu Search for a VRPSTW, in which linear penalizations are applied only to late visits. Calvete et al (2004) applies goal programming to the VRPSTW with linear penalizations for early and tardy visits.

A ship scheduling pickup and delivery problem with soft time windows is presented in Fagerholt (2001), in which the concept of maximum violation of a time window is introduced. Customer visits are allowed both before and after an "inner" time window, however earliness and tardiness are bounded within an outer time window (comprising the inner window). Customer visits falling within the outer time window are penalized according to a linear, quadratic or constant function. The problem is solved by first enumerating a large set of feasible routes and than picking the most suitable subset using a set partitioning formulation.

The same problem with linear penalizations is studied in Ioannou et al (2003), in which an iterative heuristic is developed. At each iteration, a percentage of the soft time windows is retained and the remaining windows are assumed to be hard. The resulting problem is solved by means of a nearest-neighbour heuristic.

Ibaraki et al (2005) extend the concept of time window and measure customer's inconvenience as a non-convex, piecewise linear and time dependent function. The authors propose a dynamic programming algorithm to optimise the arrival time at each customer in each route and devise three local search based metaheuristics (multi-start local search, iterated local search, adaptive multi-start local search). A faster algorithm is developed in Ibaraki et al (2008) by assuming the inconvenience measure to be a nonnegative, convex, piecewise linear, time dependent function.

Fu et al (2008) acknowledge the need of a unified approach that model penalties associated to time windows. The variants modelled in Taillard et al (1997), Koskosidis et al (1992), Balakrishnan (1993), Chang and Russell (2004), and Fagerholt (2001) are all solved by a single Tabu Search heuristic. Figliozzi (2010) proposed an iterative route construction and improvement algorithm for the same variant and compares its results with Balakrishnan (1993), Fu et al (2008) and Chang and Russell (2004).

Among the exact algorithms for VRPSTWs, it is worth referring to Qureshi et al (2009). In this article, the authors solve by column generation a problem with semi soft time windows, such as the problem in which penalizations are applied only for late arrival at the customers and the maximum delay is bounded. Bhusiri et al (2014) extend this algorithm to the variant in which both early and late arrival at a customer are penalized, but the arrival time is bounded in an outer time window. Liberatore et al (2011) solves the VRPSTW with unbounded penalization of early and late arrival at the customers using a branch-and-cut-and-price technique.

Figure 2 presents a summary of the penalization functions presented in the literature. Light grey lines identify the (inner) time window and dark grey lines the maximum allowed earliness/tardiness (or the outer time window). When a single light grey line is present, only tardy arrival is admitted and penalized. Waiting time is represented with a dashed line (as in Balakrishnan, 1993, Chang and Russell, 2004, Fu et al, 2008, Figliozzi, 2010).

Penalizations have been modelled with a variety of functions in the literature to represent customer's inconvenience. The most sophisticated penalizations involve 5 parameters per customer (i.e., Balakrishnan, 1993, Chang and Russell, 2004, Fu et al, 2008, Figliozzi, 2010) or even more for piecewise linear or convex functions (i.e., Dumas et al, 1990, Ibaraki et al, 2005, Ibaraki et al, 2008). These models are more flexible and rich. On the other hand, defining the penalization parameters for each customer and setting the relative weight of customer inconvenience is challenging for practitioners. Moreover, objective functions weighting routing costs and customer inconvenience suffer of typical drawbacks of weighted-sum multi objective optimization problems. Adjusting the weighting parameter between routing costs and customers inconvenience is often a time consuming task and this parameter often requires to be separately tuned for different instances.

3 Branch-and-cut-and-price algorithms

In what follows, we define the Opportunity Cost VRPSTW (OC-VRPSTW), by first introducing notation and recalling the definitions of the VRPSTW and of the VRPHTW.

A graph $G(V,A)$ is given, where the set $V = N \cup \{0\}$ is composed of a special vertex 0 representing the depot and a set of N customers. Non-negative weights t_{ij}

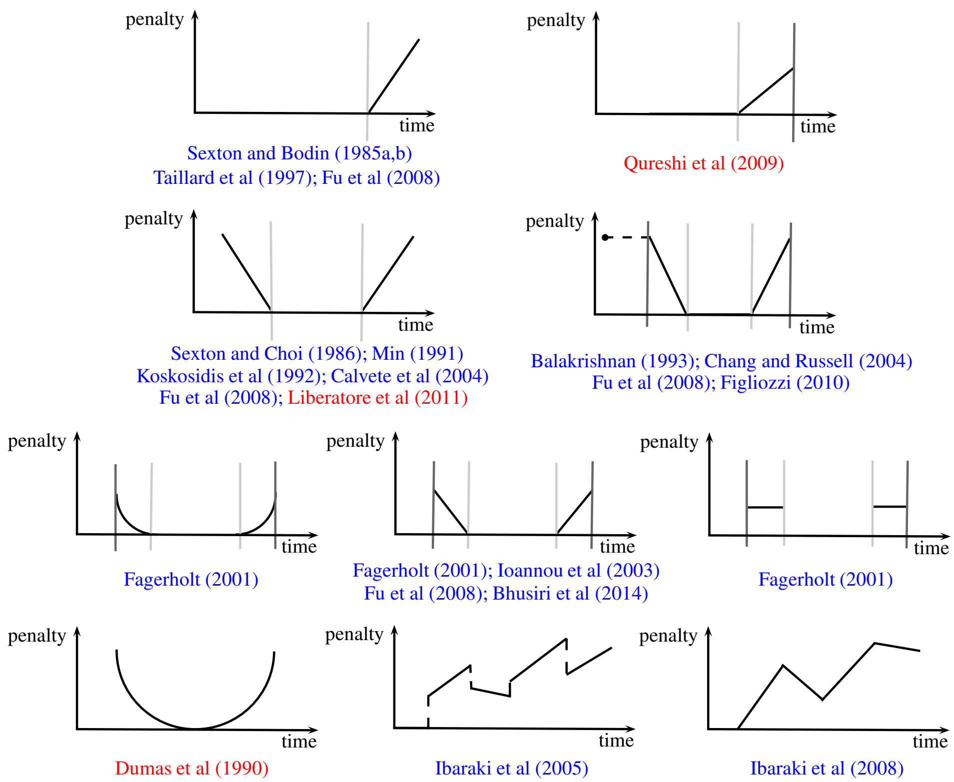

Fig. 2 Penalization functions

and c_{ij} are associated with each arc $(i, j) \in A$ representing the traveling time and the transportation cost, respectively. Traveling times satisfy the triangle inequality. A positive integer demand d_i is associated with each vertex $i \in N$ and Q is the capacity of each vehicle. The fleet is composed of K vehicles. A non-negative integer service time s_i and a time window $[a_i, b_i]$, defined by two non-negative integers, are also associated with each vertex $i \in N$.

The VRTSTW asks to find a set of routes with cardinality at most $|K|$, visiting all customers exactly once and respecting time windows and vehicles' capacity constraints. The objective is to minimize a combination of routing costs and customer inconvenience. In the VRPHTW, the vehicle has to wait until the opening of the time window a_i, in case of early arrival at customer's i location. However, both in the VRPHTW and in the VRPSTW, vehicles are allowed to wait at no cost before servicing the customer.

In the OC-VRPSTW, the model assumes that the optimal routing cost of the underlying VRPHTW, z^*, is known and is strictly positive. A cost saving is imposed by the planner as a maximum percentage $\beta < 1$ of z^*. The objective is to minimize the overall time windows violation. The model reads as follows:

$$g_\Theta = minimize \sum_{r\in\Theta} v^r x^r \tag{1}$$

$$s.t. \sum_{r\in\Theta} f_i^r x^r \geq 1 \qquad \forall i \in N \tag{2}$$

$$\sum_{r\in\Theta} c^r x^r \leq \beta \cdot z^* \tag{3}$$

$$\sum_{r\in\Theta} x^r \leq |K| \tag{4}$$

$$x^r \in \{0,1\} \qquad \forall r \in \Theta \tag{5}$$

where Θ is the set of feasible routes in which the vehicle's capacity is not exceeded and v^r is the overall time window violation of route r. Constraints (2) impose that all customer are visited at least once, here f_i^r represents the number of times route r visits customer i. Constraint (3) states that the routing cost must be not greater than a fraction of the cost of the optimal VRPHTW solution. Constraint (4) imposes that no more than $|K|$ vehicles are used.

The routing cost improvement β is the only parameter required. Planners are likely to be comfortable defining parameter β as it directly relates to monetary savings, and alternative scenarios allow for a direct comparison between solutions in which customer convenience is more or less sacrificed in favour of routing cost savings.

4 Exact algorithms for the OC-VRPSTW

Model (1)-(5) may contain a number of variables which grows exponentially with the size of the instance and cannot be dealt with explicitly. Therefore, to compute valid lower bounds, we solve the linear relaxation of the model recurring to a column generation procedure. To obtain feasible integer solutions we embed the column generation bounding procedure into an enumeration tree (Desaulniers et al, 2005; Vanderbeck and Wolsey, 1996).

At each column generation iteration, the linear relaxation of the Restricted Master Problem (RMP, i.e., the model (1)-(5) where a subset of variables is considered) is solved. We search for new columns with a negative reduced cost:

$$\overline{v}^r = v^r - \sum_{i\in N} f_i^r \pi_i - c^r \rho - \gamma \tag{6}$$

where π_i is the nonegative dual variable associated to the ith constraint of the set (2), ρ is the nonpositive dual variable associated with the threshold constraint (3) and γ is the nonpositive dual variable associated with constraint (4). The pricing problem can be modeled as a resource constrained elementary shortest path problem (RCESPP). In our implementation, we extend the algorithms presented in Righini and Salani (2006, 2008), Liberatore et al (2011) and make use of some acceleration techniques presented in Salani and Vacca (2011).

After some preliminary computational experiments, we observed that the solution of model (1)-(5) required a substantially greater amount of computational time in comparison with the time required to solve the original counterpart without time windows violation. In some instances the time required was of the order of several magnitudes higher. The reasons for this increased computational effort are the following:

- The set of feasible columns is larger than in the corresponding VRPHTW, as it contains also routes violating time windows constraints.
- The exact solution of the pricing problem is harder because two new non-dominated states are generated at each label extension. Therefore the overall number of generated labels is greater.
- For all routes r feasible for the corresponding VRPHTW instance $v_r = 0$. This increases both the complexity of the column generation procedure and the pricing algorithm. In the dynamic programming algorithm, much more labels are non dominated.
- Proving the infeasibility of a node in the search tree, in particular with respect to constraint (3) is hard and requires additional effort. Indeed, a linear relaxation of the RMP may satisfy constraint (3), but no integral solution does.

Therefore, we propose an alternative formulation and an alternative exact solution algorithm based on bisection search inspired by the ε-constraint method for multi-objective optimization.

At each iteration of the bisection search, we solve the model (7)-(11) which prescribes the minimization of the overall routing cost subject to a maximal permitted time windows violation. In the bisection search algorithm, the permitted time windows violation is then updated according to the value of the optimal solution of the model. Briefly: when the routing costs satisfy the savings prescribed by the planner, then the permitted time windows violation is reduced. When the routing costs do not satisfy the savings prescribed by the planner, then the permitted time windows violation is increased.

The model is solved with branch-and-cut-and-price and reads as follows:

$$h_{\Theta} = minimize \sum_{r \in \Theta} c^r y^r \tag{7}$$

$$s.t. \sum_{r \in \Theta} f_i^r y^r \geq 1 \qquad \forall i \in N \tag{8}$$

$$\sum_{r \in \Theta} v^r y^r \leq g_{max} \tag{9}$$

$$\sum_{r \in \Theta} y^r \leq |K| \tag{10}$$

$$y^r \in \{0,1\} \qquad \forall r \in \Theta \tag{11}$$

where g_{max} represents the maximal permitted time windows violation. Note that we model the overall time windows violation as the sum of violations of each selected route in constraint (9).

Problem (7)-(11) is defined over the same set of feasible routes Θ as problem (1)-(5) and is solved with a branch-and-cut-and-price algorithm. In the column generation process, the pricing problem searches for columns minimizing the following reduced cost:

$$\overline{c}^r = c^r - \sum_{i \in N} f_i^r \pi_i - v^r \psi - \gamma \tag{12}$$

where π_i is the nonegative dual variable associated to the ith constraint of the set (8), ψ is the nonpositive dual variable associated with the time windows violation (9) and γ is the nonpositive dual variable associated with constraint (10). The pricing problem associated to this formulation is equivalent to that studied by Liberatore et al (2011), where the linear penalty for earliness and tardiness is adjusted by means of the dual variable of constraint (9). We choose, therefore, to exploit algorithms presented in Liberatore et al (2011) and Salani and Vacca (2011).

The overall exact algorithm is based on a bisection search on the value of the permitted violation g_{max}. The key finding that allows us to devise an efficient algorithm is that at each iteration g_{max} represents either an upper or a lower bound to the optimal value of model (1)-(5), g^*_Θ.

Let $y^*(g_{max})$ be an optimal solution of (7) - (11) for a given value of g_{max} and $h^*_\Theta(g_{max})$ the corresponding value of the objective function. The bisection algorithm exploits the following two properties:

1. If $h^*_\Theta(g_{max}) > \beta \cdot z^*_\Omega$, then g_{max} is a valid lower bound to g^*_Θ, indeed, as the set of routes Θ is the same for problems (1)-(5) and (7)-(11), then any feasible solution of (1)-(5) would incur a time windows violation strictly greater than g_{max}.
2. If $h^*_\Theta(g_{max}) \leq \beta \cdot z^*_\Omega$, then $\sum_{r \in \Theta} v^r y^r$ is a valid upper bound to g^*_Θ. Trivially, a feasible solution to (1)-(5) is provided by $y^*(g_{max})$.

The algorithm requires the existence of a feasible solution and the value of an upper bound g_{UB} to g^*_Θ. In order to prove the existence of a feasible solution, the associated Capacitated Vehicle Routing Problem (CVRP), in which time windows are neglected, is solved. Assume that z^*_{CVRP} is the optimal solution of the associated CVRP, if $\beta \cdot z^* \geq z^*_{CVRP}$ then a feasible solution to (1)-(5) exist. Moreover, in the bisection algorithm, g_{UB} can be set to the violation incurred by the optimal solution of the associated CVRP.

We report in Algorithm 1 the pseudo code of the bisection algorithm.

Algorithm 1 Bisection search

Require: β, z^*_Ω, g_{UB}
 $it := 0$; $g^{it}_{UB,\Theta} := g_{UB}$; $g^{it}_{LB,\Theta} := 0$;
 while $\left(g^{it}_{UB,\Theta} - g^{it}_{LB,\Theta} > \varepsilon\right)$ **do**
  $g_{max} := \left(g^{it}_{UB,\Theta} + g^{it}_{LB,\Theta}\right)/2$;
  $h^{*,it}_\Theta$:= Solve (7) - (11);
  if $h^{*,it}_\Theta > \beta \cdot z^*_\Omega$ **then**
   $g^{it+1}_{LB,\Theta} := g_{max}$; $g^{it+1}_{UB,\Theta} := g^{it}_{UB,\Theta}$;
  else
   $g^{it+1}_{UB,\Theta} := \sum_{r\in\Theta} v^r y^r$; $g^{it+1}_{lB,\Theta} := g^{it}_{LB,\Theta}$;
  end if
  $it := it + 1$;
 end while

The algorithm is initialized with a given value for g_{UB}. At each iteration, the range of possible values for the violation of time windows $(g^{it}_{LB,\Theta}, g^{it}_{UB,\Theta})$ is halved. The algorithm stops when the gap between the lower and the upper bounds is less than ε. The value ε is strictly positive and is determined using the instance data.

5 Computational results

We performed our experiments using the well-known Solomon's data set (Solomon, 1983). For 17 instances of classes R1 and RC1 we considered the first $n = 25$ customers. For each instance, we required a percentage improvement with respect to the nominal solution of 1%, 5% and 10% (i.e., β equal to $0.99, 0.95, 0.90$, respectively). The overall number of runs amounts therefore to 51. A time limit of one hour was imposed on all runs.

All tests were performed on a PC equipped with an Intel Core i7 2.67 GHz 2 Cores processor with 3 GB RAM. The branch-and-price-and-cut is coded in ANSI-C and the linear relaxation solver is IBM-Cplex 12.0

Table 5 illustrates the exact solution of both models (1)-(5) and (7)-(5) and it is organized as follows. Each row is dedicated to one instance and the name of the instance is given in the first column; three groups of three columns follow. Each group provides the results associated with a specific value of β. Column g^* contains the opportunity cost (i.e. time windows violation) associated with the set cost saving. An asterisk $(*)$ means that no feasible solution exists for the instance and the desired value of β. Column $t(s)$ contains the overall computational time required by the standard branch-and-cut-and-price algorithm, whereas column $t'(s)$ provides information on the overall computational time required by the bisection algorithm.

A dash line $(-)$ in the cell related to the computational time means that the corresponding procedure did not converge within the time limit.

	OC (β = 0.99)			OC (β = 0.95)			OC (β = 0.90)		
	g^*	$t(s)$	$t'(s)$	g^*	$t(s)$	$t'(s)$	g^*	$t(s)$	$t'(s)$
R101	7.0	0.9	0.2	21.0	7.0	16.6	41.4	0.2	7.9
R102	2.4	1.2	0.1	2.4	1.2	0.1	29.4	3.8	18.5
R103	7.0	84.1	0.4	23.6	148.0	31.0	*	0.4	1.1
R104	6.7	1841.9	0.2	43.3	-	501.4	86.2	-	0.9
R105	2.2	1.7	0.1	19.4	2.6	4.8	47.7	17.4	16.0
R106	11.0	634.8	11.1	19.1	60.6	9.8	56.9	-	1.6
R107	6.7	-	25.2	47.9	-	28.0	*	0.9	1.0
R108	2.3	-	1.9	25.7	-	1938.9	*	0.4	3.6
R109	20.3	1491.7	14.1	68.8	881.9	0.5	*	0.4	0.6
RC101	3.4	195.8	853.9	4.3	2207.0	1237.7	4.3	38.5	2.2
RC102	0.3	519.8	0.8	16.3	-	46.6	*	0.6	2.5
RC103	2.0	-	0.8	37.1	-	122.2	*	0.9	10.3
RC104	14.3	-	15.1	*	0.7	2.0	*	0.7	44.7
RC105	3.4	3125.5	0.6	8.0	13.2	15.1	8.0	12.9	20.6
RC106	3.0	2758.9	2.4	11.1	2264.9	14.8	73.2	-	3.72
RC107	*	1.1	6.1	*	1.1	9.3	*	1.1	4.7
RC108	*	2.4	2.4	*	2.4	22.8	*	2.4	2.2

Table 1 Computational comparison between two exact algorithms for the OC-VRPSTW

Table 5 illustrates that the bisection algorithm produced an optimal solution for all instances in which such a solution exists or proved that an optimal solution does not exist.

The results in Table 5 show that the bisection algorithm was able to find an optimal solution or prove that no one exists for all instances, while the standard branch-and-price-and-cut procedure failed on 12 instances out of 51. or the instances solved by both algorithms, the bisection algorithm is in average faster than the standard-branch-and-price-and-cut. In many cases the improvement is of two orders of magnitude. In few instances, the standard branch-and-price-and-cut is faster, but the order of magnitude of computational time is the same for the two algorithms. Our final observation is that infeasible instances are quickly detected by both algorithms.

5.1 Comparison with VRPSTW

We compare our results with those obtained with an exact method for the VRPSTW. We recall that in the VRPSTW the violation of time windows is permitted and penalized in the objective function. Each time unit of violation is penalized by a constant factor. For the comparison, we executed the exact branch-and-price procedure by Liberatore et al (2011) over the same set of instances.

We are interested in the values of time windows violation and the corresponding value of the primary objective when a method based on soft-constraints is used, i.e. a linear combination of primary objective and time windows violation. To perform our comparison, we set the penalty for time windows violation equal to 1 as done in the tests reported by Liberatore et al (2011).

Results are summarised in Table 2 which is organized as follows: the first column contains the instance name followed by the optimal value without time windows violation z^*_Ω; the following two columns are dedicated to the optimal solution of VRPSTW and they contain the value of the violation of time windows, g_Θ and the primary objective value, z_Θ. We recall that their sum is optimal for the model with soft time windows. Subsequent columns illustrate the minimal time windows violation, g^*_Θ, and the value of the routing cost objective corresponding to the optimal solution, $z(g^*_\Theta)$, for the opportunity cost approach (OC) with requested cost saving of 1%, 5% and 10%. As above, when column g^*_Θ contains an asterisk $(*)$, it means that the corresponding instance is infeasible.

We observe that the soft constraints method always produces a solution. For 3 out of 17 instances, the solution is the same as for the VRPHTW. For some other instances (e.g., r101 and r102), the time windows violation is larger than that necessary to obtain a 10% improvement over the optimal VRPHTW solution.

We recall that the reported solutions have been obtained by setting the same penalty to all customers. Different penalties would indeed lead to different solutions. This fluctuating behaviour of the algorithm is undesirable for planners. They cannot rely on these results as the same settings for the parameters lead to substantially different solutions in different structurally similar instances.

		Soft TW		OC ($\beta = 0.99$)		OC ($\beta = 0.95$)		OC ($\beta = 0.90$)	
Instance	z^*_Ω	g_Θ	z_Θ	g^*_Θ	$z(g^*_\Theta)$	g^*_Θ	$z(g^*_\Theta)$	g^*_Θ	$z(g^*_\Theta)$
R101	617.1	46.7	538.0	7.0	605.4	21.0	583.4	41.4	553.4
R102	547.1	35.2	475.8	2.4	515.5	2.4	515.5	29.4	483.3
R103	454.6	9.9	439.8	7.0	448.1	23.6	430.8	*	*
R104	416.9	6.7	410.1	6.7	410.1	43.3	389.2	86.2	375.1
R105	530.5	40.2	481.3	2.2	523.0	19.4	503.7	47.7	474.4
R106	465.4	21.0	440.8	11.0	453.7	19.1	440.8	56.9	415.0
R107	424.3	6.7	413.6	6.7	418.7	47.9	402.0	*	*
R108	397.3	2.3	393.0	2.3	389.2	25.7	373.9	*	*
R109	441.3	0.0	441.3	20.3	435.4	68.9	418.0	*	*
RC101	461.1	12.1	359.6	3.4	455.9	4.3	413.0	4.3	413.0
RC102	351.8	3.3	338.8	0.3	346.0	16.3	333.5	*	*
RC103	332.8	2.0	329.4	2.0	329.4	37.1	316.0	*	*
RC104	306.6	0.0	306.6	14.3	299.7	*	*	*	*
RC105	411.3	14.8	346.6	3.4	405.0	8.0	358.0	8.0	358.0
RC106	345.5	11.1	327.6	3.0	341.0	11.1	327.6	73.2	310.5
RC107	298.3	1.0	296.3	*	*	*	*	*	*
RC108	294.5	0.0	294.5	*	*	*	*	*	*

Table 2 Comparison of VRPSTW with penalty 1 and OC-VRPSTW with β equal 99%, 95% and 90%.

6 Conclusions

We introduced a new variant of the VRPSTW, in which practitioners are allowed to set a desired routing cost saving with respect to the VRPHTW solution. The opportunity cost of cheaper solutions is quantified by minimising the customer inconvenience due to time windows violations. This problem definition does not require practitioners to define a weighting coefficient between routing cost and time window violations, and allows for the analysis of alternative scenarios with increasing routing savings and decreasing customer satisfaction. Furthermore, customer dissatisfaction can be quantified with alternative measures (i.e., minimization of the maximum time window violations, minimization of the number of time window violations).

We propose two branch-and-cut-and-price algorithms. The second algorithm is embedded within a bi-section framework and takes advantage of an easier pricing algorithm. Despite its iterative nature, this algorithm outperforms the first and it is capable of generating a larger number of optimal solutions.

Our computational results showcase that scenarios of decreasing routing cost and increasing time window violations can be easily obtained using smaller values of β. The opportunity cost solutions obtained by iteratively decrementing β allow for an overview of the possible alternative solutions that can be obtained by prioritising routing cost versus customer inconvenience.

References

Balakrishnan N (1993) Simple heuristics for the vehicle routeing problem with soft time windows. Journal of the Operational Research Society 44:279–287

Baldacci R, Mingozzi A, Roberti R (2012) Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints. European Journal of Operational Research 218:1–6

Bhusiri N, Qureshi A, Taniguchi E (2014) The trade-off between fixed vehicle costs and time-dependent arrival penalties in a routing problem. Transportation Research Part E: Logistics and Transportation Review 62:1–22

Bräysy O, Gendreau M (2005a) Vehicle routing problem with time windows, part i: Route construction and local search algorithms. Transportation Science 39:104–118

Bräysy O, Gendreau M (2005b) Vehicle routing problem with time windows, part ii: Metaheuristics. Transportation Science 39:119–139

Calvete H, Gal C, Oliveros M, Sánchez-Valverde B (2004) Vehicle routing problems with soft time windows: an optimization based approach. Monograf'ias del Seminario Matemático García de Galdeano 31:295–304

Caramia M, Dell' Olmo P (eds) (2008) Multi-objective Management in Freight Logistics. Springer London, London

Chang WC, Russell R (2004) A metaheuristic for the vehicle-routeing problem with soft time windows. Journal of the Operational Research Society 55:1298–1310

Cordeau JF, Desaulniers G, Desrosiers J, Solomon M, Soumis F (2002) VRP with time windows. In: Toth P, Vigo D (eds) The vehicle routing problem, SIAM, chap 7, pp 157–193

Dantzig GB, Ramser JH (1959) The Truck Dispatching Problem. Management Science 6:80–91

Desaulniers G, Desrosiers J, Dumas Y, Solomon M, Soumis F (1997) Daily aircraft routing and scheduling. Management Science 43(6):841–855

Desaulniers G, Desrosiers J, Solomon M (eds) (2005) Column Generation. GERAD 25th Anniversary Series, Springer

Desaulniers G, Desrosiers J, Spoorendonk S (2010) The Vehicle Routing Problem with Time Windows: State-of-the-Art Exact Solution Methods, John Wiley & Sons, Inc.

Desaulniers G, Madsen O, Ropke S (2014) Vrp with time windows. In: Toth P, Vigo D (eds) Vehicle Routing: Problems, Methods, and Applications, SIAM, Philadelphia, US

Dumas Y, Soumis F, Desrosiers J (1990) Technical noteoptimizing the schedule for a fixed vehicle path with convex inconvenience costs. Transportation Science 24:145–152

Fagerholt K (2001) Ship scheduling with soft time windows: An optimisation based approach. European Journal of Operational Research 131:559–571

Ferland J, Fortin L (1989) Vehicles scheduling with sliding time windows. European Journal of Operational Research 38:213–226

Figliozzi M (2010) An iterative route construction and improvement algorithm for the vehicle routing problem with soft time windows. Transportation Research Part C: Emerging Technologies 18:668–679

Fu Z, Eglese R, Li L (2008) A unified tabu search algorithm for vehicle routing problems with soft time windows. Journal of the Operational Research Society 59:663–673

Gendrau M, Tarantilis C (2010) Solving large-scale vehicle routing problems with time windows: The state-of-the-art. Tech. Rep. 2010-04, CIRRELT

Gendreau M, Tarantilis CT (2010) Solving large-scale vehicle routing problems with time windows: state-of-the-art. Tech. Rep. 2010-04, CIRRELT

Golden B, Raghavan S, Wasil EA (2008) The vehicle routing problem : latest advances and new challenges. Operations research/Computer science interfaces series, 43, Springer

Ibaraki T, Imahori S, Kubo M, Masuda T, Uno T, Yagiura M (2005) Effective local search algorithms for routing and scheduling problems with general time-window constraints. Transportation Science 39:206–232

Ibaraki T, Imahori S, Nonobe K, Sobue K, Uno T, Yagiura M (2008) An iterated local search algorithm for the vehicle routing problem with convex time penalty functions. Discrete Applied Mathematics 156:2050–2069

Ioannou G, Kritikos M, Prastacos G (2003) A problem generator-solver heuristic for vehicle routing with soft time windows. Omega 31:41–53

Kallehauge B (2008) Formulations and exact algorithms for the vehicle routing problem with time windows. Computers & Operations Research 35:2307–2330

Koskosidis Y, Powell W, Solomon M (1992) An optimization-based heuristic for vehicle routing and scheduling with soft time window constraints. Transportation Science 26:69–85

Liberatore F, Righini G, Salani M (2011) A column generation algorithm for the vehicle routing problem with soft time windows. 4OR: A Quarterly Journal of Operations Research 9:49–82

Min H (1991) A multiobjective vehicle routing problem with soft time windows: the case of a public library distribution system. Socio-Economic Planning Sciences 25:179–188

Pullen H, Webb M (1967) A computer application to a transport scheduling problem. The Computer Journal 10:10–13

Qureshi A, Taniguchi E, Yamada T (2009) An exact solution approach for vehicle routing and scheduling problems with soft time windows. Transportation Research

Part E: Logistics and Transportation Review 45:960–977
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
Righini G, Salani M (2008) New dynamic programming algorithms for the resource constrained shortest path problem. Networks 51(3):155–170, DOI 10.1002/net.20212
Ruinelli L, Salani M, Gambardella LM (2012) Hybrid column generation-based approach for vrp with simultaneous distribution, collection, pickup-and-delivery and real-world side constraints. In: Proceedings of ICORES 2012, pp 247–255
Salani M, Vacca I (2011) Branch and price for the vehicle routing problem with discrete split deliveries and time windows. European Journal of Operational Research 213(3):470–477
Schrage L (1981) Formulation and structure of more complex/realistic routing and scheduling problems. Networks 11:229–232
Sexton T, Bodin L (1985a) Optimizing single vehicle many-to-many operations with desired delivery times: I. scheduling. Transportation Science 19:378–410
Sexton T, Bodin L (1985b) Optimizing single vehicle many-to-many operations with desired delivery times: Ii. routing. Transportation Science 19:411–435
Sexton T, Choi YM (1986) Pickup and delivery of partial loads with "soft" time windows. American Journal of Mathematical and Management Sciences 6:369–398
Solomon M (1983) Vehicle routing and scheduling with time windows constraints: Models and algorithms. PhD thesis, University of Pennsylvania
Taillard E, Badeau P, Gendreau M, Guertin F, Potvin JY (1997) A tabu search heuristic for the vehicle routing problem with soft time windows. Transportation Science 31:170–186
Toth P, Vigo D (2002) The vehicle routing problem. SIAM monographs on discrete mathematics and applications, Society for Industrial and Applied Mathematics
Vanderbeck F, Wolsey L (1996) An exact algorithm for ip column generation. Operations Research Letters 19:151–159
Vidal T, Crainic T, M G, Prins C (2012) Heuristics for multi-attribute vehicle routing problems: A survey and synthesis. Tech. Rep. 2012-05, CIRRELT