TU/e EINDHOVEN UNIVERSITY OF TECHNOLOGY

Vehicle routing problem with stochastic travel times including soft time windows and service costs

Citation for published version (APA):
Tas, D., Dellaert, N. P., Woensel, van, T., & Kok, de, A. G. (2013). Vehicle routing problem with stochastic travel times including soft time windows and service costs. *Computers & Operations Research*, *40*(1), 214-224. https://doi.org/10.1016/j.cor.2012.06.008

DOI:
10.1016/j.cor.2012.06.008

Document status and date:
Published: 01/01/2013

Document Version:
Publisher's PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the "Taverne" license above, please follow below link for the End User Agreement:

www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:

openaccess@tue.nl

providing details and we will investigate your claim.

Download date: 07. Nov. 2026

Computers & Operations Research 40 (2013) 214–224

Contents lists available at SciVerse ScienceDirect

Computers & Operations Research

journal homepage: www.elsevier.com/locate/caor

Vehicle routing problem with stochastic travel times including soft time windows and service costs

Duygu Taş *, Nico Dellaert, Tom van Woensel, Ton de Kok

School of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, The Netherlands

ARTICLE INFO

Available online 27 June 2012

Keywords:
Vehicle routing
Time windows
Stochastic travel times
Service costs

ABSTRACT

This paper studies a vehicle routing problem with soft time windows and stochastic travel times. A model is developed that considers both transportation costs (total distance traveled, number of vehicles used and drivers' total expected overtime) and service costs (early and late arrivals). We propose a Tabu Search method to solve this model. An initialization algorithm is developed to construct feasible routes by taking into account the travel time stochasticity. Solutions provided by the Tabu Search algorithm are further improved by a post-optimization method. We conduct our computational experiments for well-known problem instances. Results show that our Tabu Search method performs well by obtaining very good final solutions in a reasonable amount of time.

© 2012 Elsevier Ltd. All rights reserved.

1. Introduction

Traditionally, the Vehicle Routing Problem with Time Windows (VRPTW) aims to route vehicles such that all customers are served within their respective time windows. Mathematically, this is translated into a deterministic arrival time moment of being in the time window or not. The latter is usually penalized depending on whether the vehicle is early or late. However, solutions of the deterministic routing models deteriorate once applied in real-life problems where (especially) the travel times are stochastic (see [1] for a review). Using stochastic travel times allows for a much richer set of (stochastic) measures to evaluate whether the vehicle arrives in the time windows or not. The definition and incorporation of these stochastic measures in the VRPTW is the subject of this paper.

In practice, carrier companies and their customers have different concerns. From the perspective of the carrier companies, the goal is to deliver the goods to different customers as efficiently as possible. From the customers' point of view, the main concern is to reliably receive the deliveries on-time. In this paper, our routing problem has soft time windows and stochastic travel times, leading to stochastic arrival times. The latter are used to calculate the service costs, defined as the measure of delivery reliability.

The described problem extends the classical Vehicle Routing Problem (VRP) by considering stochastic travel times and soft time windows. The VRP belongs to the class of NP-hard combinatorial optimization problems (see [2] for an explanation). Small-sized instances of such problems are usually handled by exact algorithms. However, metaheuristic algorithms are widely used to solve medium- to large-sized instances. In this paper, we solve the Solomon's problem instances [3] effectively by a solution procedure based on Tabu Search.

The two main contributions of this paper are described below.

1. Our paper proposes the first model that distinguishes between the transportation costs and the service costs. Stochasticity in the travel times plays a role in the calculations of both cost components. The transportation costs are the true costs that the carrier company pays. On the other hand, the intangible service costs are included to provide reliability to the customers by limiting early and late arrivals. The service cost component can be thought of as a surrogate for customer service. Applying the generated model enables us to obtain meaningful combinations of the two cost components, leading to different solution options to the carrier companies to meet their priorities. A comprehensive analysis is performed to examine the behavior and the particular features of the solutions found.
2. We propose a solution approach that comprises three phases. In the first phase, an initial solution is constructed. A number of heuristic methods have been presented by Solomon [3] to build the initial routes for the VRPTW. We extend insertion heuristic I1 by including a criterion related to the penalties resulting from the time window violations. This solution is then improved with respect to the total transportation cost, leading to the initial feasible solution. In the second phase, the given initial solution is improved by a Tabu Search metaheuristic. The algorithm given by Cordeau et al. [4] constitutes the

* Corresponding author. Tel.: +31 40 2474388; fax: +31 40 2465949.
E-mail address: d.tas@tue.nl (D. Taş).

0305-0548/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cor.2012.06.008

base structure of our Tabu Search method. The soft time windows for the deliveries enable us to handle the time window violations either by the objective function or by the constraints. In our model, these violations are taken care of directly in the objective function. Therefore, we have a different cost function from that given in [4] to evaluate the solutions. As another difference from [4], we include a medium-term memory application in our Tabu Search method. This application improves the quality of the solutions by providing intensification in the promising parts of the neighborhood. In the third phase, a post-optimization method is applied to improve the solution obtained by the Tabu Search algorithm. The post-optimization method adjusts the departure time of each allocated vehicle from the depot to reduce the total service cost of the corresponding route.

The remainder of this paper is organized as follows. In Section 2, we present a literature review which deals with the stochastic versions of the VRP and the VRPTW. In Section 3, we describe our model and motivations for the VRPTW with stochastic travel times and soft time windows, and we discuss the issues connected with the model. In Section 4, we explain the methods used in the three phases of our solution approach. In Section 5, we present the results of the Solomon's problem instances solved using the methods developed. Finally, we end the paper with conclusions and with suggestions for future research.

2. Literature review

There is a wide range of literature on the VRP as it is a highly relevant, yet complicated problem. We refer to Laporte [2,5] for exact, heuristic and metaheuristic algorithms, and to Baldacci et al. [6] for recent exact algorithms applied to the VRP. A related problem also frequently seen in practice and studied is the VRPTW. In his seminal paper, Solomon [3] extended a number of VRP heuristic methods for the VRPTW. The interested reader is referred to Bräysy and Gendreau [7,8] and Desrosiers et al. [9] for an overview of various solution methods applied to the VRPTW. Most of the models developed for the VRP and the VRPTW in the literature considered deterministic parameters such as deterministic travel times, demands and service times.

A comprehensive survey on the stochastic VRP can be found in Gendreau et al. [1]. The authors argued that uncertainty can be seen in various components of the VRP: stochastic travel times, stochastic demands and/or stochastic customers. In Laporte et al. [10], the VRP with stochastic travel and service times was considered. The authors introduced three distinct formulations based on stochastic programming and developed a branch-and-cut approach. Kenyon and Morton [11] developed a solution procedure by inserting a branch-and-cut algorithm into a Monte Carlo solution approach to solve large-sized problems effectively. Van Woensel et al. [12] studied the VRP with the travel times resulting from a stochastic process due to the traffic congestion. The developed queueing models were solved by means of a Tabu Search metaheuristic. Stewart and Golden [13] studied the stochastic VRP where uncertain customer demands were considered.

Stochastic versions of the VRPTW are introduced more recently. Ando and Taniguchi [14] considered the VRPTW with uncertain travel times. The objective was to minimize the total cost which included the penalty costs due to the early and late arrivals, the operation costs and the fixed cost of vehicles used. A genetic algorithm was proposed to solve the described problem. Russell and Urban [15] also studied the VRPTW where the travel times were random variables with a known probability distribution. The number of vehicles used and the total distance traveled were minimized along with the penalties due to arrivals outside the time windows. The authors developed a Tabu Search method. The VRPTW with stochastic travel and service times was studied by Li et al. [16]. Two formulations based on stochastic programming were proposed. A heuristic algorithm based on Tabu Search was developed to obtain the results effectively. The models in these studies placed emphasis on the customers and considered all cost components together regardless of their relations and differences. Since efficiency plays an important role in operations, we separate out the cost components into transportation costs and service costs. We develop a one-stage model which enables different combinations of these two cost components with respect to the company preferences. Additionally, in our model the time window violations and the overtime of the drivers are handled by the objective function. A technique similar to that applied in Stewart and Golden [13] is used in our study to calculate the penalties incurred for early and late servicing. Our model takes into account penalties proportional to the expected duration of the earliness and lateness derived from the arrival time distributions.

The classical routing problems and their stochastic versions have been widely solved by applying the Tabu Search metaheuristic to obtain good solutions within a reasonable time. The interested reader is referred to Glover [17,18] for the details about this metaheuristic. Gendreau et al. [19,20] implemented the Tabu Search method for the VRP. Rochat and Taillard [21] proposed the adaptive memory, which turned out to be very effective for Tabu Search applications in the VRP. For the VRPTW, some implementations of the Tabu Search method come from Cordeau et al. [4], Garcia et al. [22], and Taillard et al. [23]. Our Tabu Search implementation is based on the algorithm given in [4]. We however apply a different function in the local search algorithm to evaluate the solutions since the violations of the time windows are handled by the objective function. In the process of evaluating the solutions, the stochasticity of the problem is taken into consideration. Furthermore, the Tabu Search algorithm is improved by adding a medium-term memory which focuses the search on the promising solutions in the neighborhood. This intensification mechanism operates by restarting from the best feasible solution, and searching its neighborhood effectively by means of a list which includes the moves previously applied from that solution. This structure makes our medium-term memory application different from the intensification approaches in which solutions are generated by extracting good routes from high-quality solutions on-hand (see [21]).

The interested reader is referred to Cordeau et al. [4], Rochat and Taillard [21], Russell and Urban [15], and Taillard et al. [23] for a post-optimization heuristic applied in the VRPTW. In [4,23], the authors used a heuristic method developed by Gendreau et al. [24] for the traveling salesman problem with time windows by modifying the GENIUS procedure (see [25]). In the post-optimization phase of their heuristic, each node of a route was successively removed and re-inserted to improve the solution on-hand. As a post-optimization method, Rochat and Taillard [21] solved a set partitioning model at the end of the diversification and intensification techniques to improve the solution by using the routes already generated. In Russell and Urban [15], a post-optimization method was applied to optimize the waiting times at each customer by using a generalized reduced gradient method. In this paper, we improve the solution obtained by Tabu Search by applying a post-optimization method which is based on adjusting the departure time of each route from the depot, and is thus different from the post-optimization methods given in the literature. In our approach, we do not change the nodes, but the

departure time of a route from the depot is repeatedly shifted until the method does not provide any improvement in the total service cost of that route.

3. Problem statement and model formulation

A connected digraph $G=(N,A)$ denotes the network in which $N=\{0,1,\ldots,n\}$ is the set of nodes and $A=\{(i,j)|i,j\in N, i\neq j\}$ is the set of arcs. The depot is represented by node 0 and each node in $N\backslash\{0\}$ corresponds to a distinct customer. Each customer has a known demand ($q_i\geq 0$), a fixed service duration ($s_i\geq 0$) and a soft time window ($[l_i,u_i]$, $l_i\geq 0$, $u_i\geq 0$). The time window at the depot, $[l_0,u_0]$, corresponds to the scheduling horizon. We assume that the service at the customers can start before or after the time windows. If a vehicle arrives early at a customer, waiting until the customer time window opens is not considered as an option. If a customer is served outside its time window, then the company incurs penalties for early or late servicing. Each arc $(i,j)\in A$ has a weight d_{ij} which is the distance of that arc. In addition, we assume that the probability distribution function of the travel time on each arc (i,j) is known. The base location of the vehicles is the depot and each vehicle $v\in V$ is assumed to have the same capacity (Q). The aim is to construct a set of vehicle routes at the minimum total cost by fulfilling the following requirements:

- Each route is operated by a single vehicle and each customer is served by one vehicle exactly once.
- Each vehicle route starts from the depot and ends at the depot.
- The total demand of the customers assigned to a vehicle route cannot exceed the vehicle's capacity.

We first define the notations used in the mathematical formulation of the described problem. The decision variable x_{ijv} takes the value 1 if arc (i,j) is covered by vehicle v and 0, otherwise. The vector $\mathbf{x}$, where $\mathbf{x}=\{x_{ijv}|i,j\in N, v\in V\}$, is used to denote the assignments of the vehicles and the sequences of the customers in these assignments (vehicle routes). We have two functions in the service cost component described as $D_{jv}(\mathbf{x})$ and $E_{jv}(\mathbf{x})$. These are the *expected* delay and the *expected* earliness at node j in case it is visited by vehicle v, respectively. $O_v(\mathbf{x})$ is the *expected* overtime of the driver working on the route of the vehicle v. The calculations of the expected values are directly linked with the routing decisions. These calculations are described in Section 3.2 for a specific probability distribution.

The coefficients c_d and c_e can be thought of as the costs that the company is charged for one unit of delay and for one unit of earliness, respectively. Note that these coefficients are needed to balance late and early servicing. The parameters c_t and c_o are the costs that the carrier company has to pay for one unit of distance and for one unit of overtime, respectively. Furthermore, a fixed cost c_f has to be paid by the company for each vehicle used.

The mathematical formulation is as follows:

$$\min \quad \rho\frac{1}{C_1}\left(c_d\sum_{j\in N}\sum_{v\in V}D_{jv}(\mathbf{x})+c_e\sum_{j\in N}\sum_{v\in V}E_{jv}(\mathbf{x})\right)+(1-\rho)\frac{1}{C_2}\left(c_t\sum_{i\in N}\sum_{j\in N}\sum_{v\in V}d_{ij}x_{ijv}+c_f\sum_{j\in\{N\backslash 0\}}\sum_{v\in V}x_{0jv}+c_o\sum_{v\in V}O_v(\mathbf{x})\right) \tag{1}$$

$$\text{s.t.} \quad \sum_{i\in N}x_{ikv}-\sum_{j\in N}x_{kjv}=0, \quad k\in N\backslash\{0\},\ v\in V \tag{2}$$

$$\sum_{j\in N}\sum_{v\in V}x_{ijv}=1, \quad i\in N\backslash\{0\}, \tag{3}$$

$$\sum_{j\in N}x_{0jv}=1, \quad v\in V, \tag{4}$$

$$\sum_{i\in N}x_{i0v}=1, \quad v\in V, \tag{5}$$

$$\sum_{i\in N\backslash\{0\}}q_i\sum_{j\in N}x_{ijv}\leq Q, \quad v\in V, \tag{6}$$

$$\sum_{i\in S}\sum_{j\in S}x_{ijv}\leq |S|-1, \quad S\subseteq N\backslash\{0\},\ v\in V, \tag{7}$$

$$x_{ijv}\in\{0,1\}, \quad i\in N,\ j\in N,\ v\in V. \tag{8}$$

The objective (1) is to minimize the total weighted cost. The set of the constraints (2) ensures the conservation of flow for each vehicle at each customer. The set of the constraints (3) states that each customer is visited exactly once. The sets of the constraints (4) and (5) ensure that each vehicle route originates from the depot and ends at the depot. The set of the constraints (6) states that the total demand of the customers served by a vehicle cannot exceed the capacity of the vehicle. The set of constraints (7) eliminates subtours and the set of the constraints (8) is needed as we cannot have partial services at the customers.

In this model, the parameter ρ is used to obtain alternative combinations of the cost of servicing and the cost of transportation. As these cost components (potentially) have different scales, we include two parameters (C_1 and C_2) in the objective function to scale their values. The solution generated by the initialization algorithm is improved by the Tabu Search method with respect to only the total transportation cost, leading to the initial feasible solution. Then, the scaling parameters for the total service cost (C_1) and the total transportation cost (C_2) are calculated as follows:

$$C_1=c_d\sum_{j\in N}\sum_{v\in V}D_{jv}(\tilde{\mathbf{x}})+c_e\sum_{j\in N}\sum_{v\in V}E_{jv}(\tilde{\mathbf{x}}) \quad \text{and} \tag{9}$$

$$C_2=c_t\sum_{i\in N}\sum_{j\in N}\sum_{v\in V}d_{ij}\tilde{x}_{ijv}+c_f\sum_{j\in\{N\backslash 0\}}\sum_{v\in V}\tilde{x}_{0jv}+c_o\sum_{v\in V}O_v(\tilde{\mathbf{x}}), \tag{10}$$

where $\tilde{\mathbf{x}}$ represents the assignments of the vehicles and the sequences of the customers in the initial feasible solution. The variable $\tilde{x}_{ijv}$ takes the value 1 if arc (i,j) is covered by vehicle v in this solution and 0, otherwise. By using C_1 and C_2 parameters, we can solve all VRPTW instances with a fixed set of ρ values.

3.1. Properties of the arrival times

We consider stochastic travel times with a known probability distribution. Suppose that T_{ij} represents the time needed for traveling from node i to node j by traversing the arc (i,j). As we have no waiting, the arrival time of vehicle v at node j, denoted by Y_{jv}, is described as follows:

$$Y_{jv}=\sum_{(l,k)\in A_{jv}}T_{lk}, \tag{11}$$

where A_{jv} represents the set of arcs which are covered by vehicle v until node j. This calculation does not include the service times. As we have deterministic service times, the time window at node j is shifted to the left on the time horizon by the amount of the cumulative service time. The latter (s_{jv}) is the total time spent by the vehicle v for servicing at the nodes before visiting node j. The mean and the variance of the arrival time at node j which is

visited by vehicle v immediately after node i are calculated by:

$$E[Y_{jv}] = E[Y_{iv}] + E[T_{ij}] \quad \text{and} \tag{12}$$

$$\mathrm{Var}(Y_{jv}) = \mathrm{Var}(Y_{iv}) + \mathrm{Var}(T_{ij}), \tag{13}$$

respectively.

3.2. Calculations with gamma distribution

The distributions of the travel times most commonly applied so far are normal, log-normal, shifted gamma and gamma distributions (see [26,27,16,15]). Assume that T is the random travel time spent for traversing one unit of distance and that T is Gamma distributed with shape parameter α and scale parameter λ. Then, the probability density function f and the cumulative distribution function F are given as follows:

$$f(t) = \frac{(e^{-t/\lambda})(t)^{\alpha-1}}{\Gamma(\alpha)\lambda^{\alpha}}, \tag{14}$$

$$F(\delta) = \mathrm{Prob}\{t \le \delta\} = \Gamma_{\alpha,\lambda}(\delta) = \int_0^{\delta} \frac{(e^{-z/\lambda})(z)^{\alpha-1}}{\Gamma(\alpha)\lambda^{\alpha}}\, dz, \tag{15}$$

where $t \ge 0$, $\delta \ge 0$ and $\Gamma(\alpha) = \int_0^{\infty} e^{-r} r^{\alpha-1}\, dr$. For other distribution types, similar expressions can be derived. Note that we obtain different Coefficient of Variation (CV) values of the travel time per unit distance by using different values of α and λ parameters. Since α and λ are the parameters associated with T, T_{ij} is Gamma distributed with parameters αd_{ij} and λ obtained by scaling T with respect to the distance of the arc (i,j). The mean and the variance of T_{ij} are calculated accordingly as follows:

$$E[T_{ij}] = \alpha\lambda d_{ij}, \tag{16}$$

$$\mathrm{Var}(T_{ij}) = \alpha\lambda^2 d_{ij}. \tag{17}$$

By defining the arrival times as in Eq. (11), we obtain Gamma distributed arrival times. The shape and the scale parameters of Y_{jv} are then given as follows:

$$\alpha_{jv} = \alpha \sum_{(l,k)\in A_{jv}} d_{lk}, \tag{18}$$

$$\lambda_{jv} = \lambda. \tag{19}$$

The expected delay $D_{jv}(\mathbf{x})$ is calculated as follows with a similar procedure to that given in Dellaert et al. [28]:

$$\begin{aligned} D_{jv}(\mathbf{x}) &= \int_{u_j'}^{\infty} (z-u_j') \frac{(e^{-z/\lambda_{jv}})(z)^{\alpha_{jv}-1}}{\Gamma(\alpha_{jv})(\lambda_{jv})^{\alpha_{jv}}}\, dz, \\ &= \int_{u_j'}^{\infty} \frac{(e^{-z/\lambda_{jv}})(z)^{\alpha_{jv}}}{\Gamma(\alpha_{jv})(\lambda_{jv})^{\alpha_{jv}}}\, dz - u_j' \int_{u_j'}^{\infty} \frac{(e^{-z/\lambda_{jv}})(z)^{\alpha_{jv}-1}}{\Gamma(\alpha_{jv})(\lambda_{jv})^{\alpha_{jv}}}\, dz, \\ &= \alpha_{jv}\lambda_{jv}(1-\Gamma_{\alpha_{jv}+1,\lambda_{jv}}(u_j')) - u_j'(1-\Gamma_{\alpha_{jv},\lambda_{jv}}(u_j')), \end{aligned} \tag{20}$$

where u_j' is the upper bound of the shifted time window at node j. Similarly, the expected earliness, $E_{jv}(\mathbf{x})$, is calculated by

$$E_{jv}(\mathbf{x}) = l_j'\Gamma_{\alpha_{jv},\lambda_{jv}}(l_j') - \alpha_{jv}\lambda_{jv}\Gamma_{\alpha_{jv}+1,\lambda_{jv}}(l_j'), \tag{21}$$

where l_j' is the lower bound of the shifted time window at node j. The expected overtime of the driver working on the route of vehicle v is calculated with respect to the arrival time of that vehicle at the depot:

$$O_v(\mathbf{x}) = \alpha_{0v}\lambda_{0v}(1-\Gamma_{\alpha_{0v}+1,\lambda_{0v}}(w')) - w'(1-\Gamma_{\alpha_{0v},\lambda_{0v}}(w')), \tag{22}$$

where w' is the agreed labor shift time (w) less the total service time spent by the vehicle v for servicing at all nodes on its route (s_{0v}). The adjustment to the value of w is needed due to the arrival time calculation.

If $w \le s_{0v}$, then $O_v(\mathbf{x})$ is calculated by:

$$O_v(\mathbf{x}) = E[Y_{0v}] + s_{0v} - w. \tag{23}$$

We have similar conditions for the bounds of the time windows at customers. If $l_j \le s_{jv}$ at customer j, then $E_{jv}(\mathbf{x})$ will be equal to 0 since it is impossible to be early for that customer. However, if $u_j \le s_{jv}$ at customer j, then $D_{jv}(\mathbf{x})$ is calculated as follows:

$$D_{jv}(\mathbf{x}) = E[Y_{jv}] + s_{jv} - u_j. \tag{24}$$

4. Solution methods

We develop a solution method based on a Tabu Search metaheuristic. Algorithm 1 gives the structured overview of our solution approach.

Algorithm 1. Structured overview of the solution approach.

1. Obtain the Initial Feasible Solution (IFS):
   (a) Construct a solution by using the initialization algorithm
   (b) Improve this solution through the Tabu Search method with respect to the total transportation cost
2. Calculate C_1 and C_2 according to IFS and use these values in following steps
3. Improve IFS with respect to the total weighted cost by using the Tabu Search method
4. Apply the post-optimization method to the generated solution

4.1. Initialization algorithm

We extend the Solomon's insertion heuristic I1 [3] to construct a feasible solution by considering the expected violations of the time windows.

Let C denote the set of customers not yet covered by any route. Note that the method starts from a set $C = N\backslash\{0\}$. The criterion applied for the route initialization is the furthermost customer $c \in C$ from the depot. At each iteration a new customer is inserted into the route. Let $m_1(i,k,j)$ and $m_2(i,k,j)$ denote the measures used to evaluate the insertion of customer k between adjacent customers i and j on the current route. The calculations of these measures are as follows:

$$m_1(i,k,j) = \beta_1 m_{11}(i,k,j) + \beta_2 m_{12}(i,k,j) + \beta_3 m_{13}(i,k,j), \tag{25}$$

$$m_2(i,k,j) = \eta E[T_{0k}] - m_1(i,k,j), \tag{26}$$

where

$$m_{11}(i,k,j) = d_{ik} + d_{kj} - \gamma d_{ij}, \tag{27}$$

$$m_{12}(i,k,j) = E[b_{j_k}] - E[b_j], \tag{28}$$

$$m_{13}(i,k,j) = c_d\left(\sum_{h\in H}(D_{hv}(\mathbf{r}_k) - D_{hv}(\mathbf{r}))\right) + c_e\left(\sum_{h\in H}(E_{hv}(\mathbf{r}_k) - E_{hv}(\mathbf{r}))\right), \tag{29}$$

where $\beta_1 \ge 0$, $\beta_2 \ge 0$, $\beta_3 \ge 0$, $\eta \ge 0$, $\gamma \ge 0$ and $\beta_1 + \beta_2 + \beta_3 = 1$. $E[T_{0k}]$ is the expected travel time from depot to customer k. c_d and c_e are the coefficients used by the service cost component in the proposed model (see Section 3). $E[b_j]$ is the expected time to begin service at customer j which is calculated with respect to the

sequence of the customers in the current route. $E[b_{j_k}]$ is the new expected time to begin service at customer j, given that customer k is on the route. H is the set of customers which are visited after customer i on the current route. $\mathbf{r}$ and $\mathbf{r}_k$ are the vectors of the customer sequence in the route before customer k is inserted and after customer k is inserted, respectively. The vehicle that covers the current route is denoted by v. Note that the expected time window violations due to the insertion of customer k are taken into consideration in $m_{13}(i,k,j)$. Furthermore, the criteria proposed in the heuristic I1 which include time aspects are modified in our procedure in accordance with the stochastic nature of the problem. These adjustments can be seen in the measures given by Eqs. (26) and (28).

The best insertion place of customer k is the one that minimizes the value of $m_1(i,j,k)$ over all feasible insertion places. This means that the weighted combination of the extra distance, extra time and extra penalties (incurred due to the insertion of customer k) is minimized. Different sets of weight values (β_1,β_2 and β_3) are used to construct different combinations. The best feasible customer for the current route is the one that maximizes the value of $m_2(i,k,j)$ over all feasible customers. In this way, the benefit gained by serving a customer on the current route instead of serving this customer by a single vehicle is maximized.

The available vehicle capacity is checked to indicate the customers feasible to be inserted into the current route. The time feasibility conditions given in Solomon [3] are checked to determine the feasible insertion places for each indicated feasible customer. Although the latter feasibility check is no longer necessary in our model formulation, we keep this part in the initialization procedure as numerical results show that it contributes to a good solution quality.

4.2. Tabu Search algorithm

The structure of the Tabu Search method is based on the procedure given by Cordeau et al. [4]. We modify this procedure with respect to the characteristics of our problem and extend it in terms of intensification mechanisms. In the following, we first introduce the generic features of the developed method. The differences of the method with the procedure given in [4] are highlighted.

Our Tabu Search algorithm always starts with a feasible solution. Therefore, it is guaranteed that the algorithm will end with a feasible solution. These conditions make the algorithm different from the procedure given in [4] in terms of the starting technique and thus the feasibility situation of the solution obtained at the termination. At each iteration, a neighborhood of the current solution is constructed. In this neighborhood, a solution is selected as the new current solution in accordance with some criteria. Then, the algorithm continues from this new solution. Note that both feasible solutions and infeasible solutions with respect to the capacity constraint are taken into consideration during the search in the neighborhood.

A solution y is a set of p routes which may be infeasible with respect to the vehicle capacity constraint. The total load of its routes in excess of the vehicle capacity is represented by $q(y)$. Let $z(y)$ denote the objective function value of the solution y. This value corresponds to the total transportation cost in Step (1b), and the total weighted cost in Step (3) of Algorithm 1. Then, the cost function which is used to evaluate the solutions is defined as follows:

$$c(y) = z(y) + vq(y), \tag{30}$$

where v is a positive parameter. This parameter is modified at each iteration. The total violation of the time windows and the drivers' work hours are included in the cost function by means of $z(y)$. Therefore, solutions are evaluated by a cost function which has a different relaxation mechanism from the technique given in [4].

The neighborhood $g(y)$ of the solution y is constructed by employing two types of relocation operators defined as follows:

- Relocate a customer by changing its location within the route.
- Relocate a customer by deleting it from a route and inserting it into another route.

Suppose that in the current iteration a solution generated by relocating the customer i is selected as the new current solution. The customer i is then added to the tabu list to prevent its relocation for the next ϑ iterations. However, the tabu status of a customer is overridden if the aspiration criterion is satisfied: a solution which is generated by relocating a tabu customer can be selected by the algorithm if this solution has a better cost value than the best cost value obtained up to the current iteration.

As a diversification mechanism, a supplementary cost component is used during the search in the neighborhood. Suppose that $y' \in g(y)$ and $c(y') \geq c(y)$. Under these conditions, the additional cost of the solution y' is calculated by a function with respect to the features of that solution. This supplementary cost is then added to $c(y')$ to diversify the search. We use a similar function to that given in [4] to calculate the supplementary costs. In this function, the intensity of the diversification is adjusted by a constant parameter (μ). Note that for the diversification mechanism, we take into consideration the customers relocated during the search instead of the added attributes to the solution since we have a different tabu list structure.

The process outlined in the following is repeated at most θ times. At each iteration, our algorithm selects a non-tabu solution in the neighborhood of the current solution which has a better cost function value than the cost function value of the current solution. If such a solution cannot be found by the algorithm, then the best non-tabu solution in this neighborhood is chosen. Note that we apply the first selection criterion to have an effective Tabu Search algorithm. The solution selected as the new current solution is then checked for the best feasible solution criteria. In case these criteria are satisfied, the algorithm updates the best feasible solution obtained so far. In the algorithm, the parameter τ is used as a secondary terminating criterion. If the best feasible solution is not updated for τ iterations where $\tau < \theta$, the algorithm is terminated. In our Tabu Search procedure, a medium-term memory is applied as an intensification mechanism. If the best feasible solution is not updated for a specific number of iterations, it becomes the new current solution. The previous moves applied from the best feasible solution have been recorded by a list. By means of these recorded moves, the search can now be directed from the non-promising regions to the promising regions in the neighborhood. We conducted a number of preliminary tests to measure the effect of our medium-term memory. Results indicate that this mechanism improves the solution quality by 1–2% on average, over carrying out Tabu Search without medium-term memory, with a modest increase in the solution time. Our Tabu Search algorithm with the applications of the secondary terminating criterion and the intensification mechanism extends the procedure given in [4]. The steps of our Tabu Search procedure are summarized in Algorithm 2.

Algorithm 2. Tabu Search algorithm.

Set y as the given initial feasible solution
Set $y^* := y$ and $z(y^*) := z(y)$
Set $\kappa := 1$, $stop := 0$
while $\kappa \leq \theta$ and $stop = 0$ **do**

Choose the first solution $y' \in g(y)$ that satisfies $c(y') < c(y)$ and is not tabu or satisfies the aspiration criterion
if *Such a solution cannot be found* **then**
  Choose a solution $y' \in g(y)$ that minimizes $c(y')$ value and is not tabu
end
if y' *is feasible and* $z(y') < z(y^*)$ **then**
  Set $y^* := y'$ and $z(y^*) := z(y')$
end
if y^* *is not updated for* $\sqrt{\kappa}$ *iterations* **then**
  Set $y := y^*$ and $c(y) := c(y^*)$
  Update the tabu list accordingly
end
else
  Set $y := y'$ and $c(y) := c(y')$
end
if $q(y) \neq 0$ **then**
  Set $v := v^*(1+\varphi)$
end
else if $q(y) = 0$ **then**
  Set $v := v/(1+\varphi)$
end
if y^* *is not updated for* τ *iterations* **then**
  Set *stop* := 1
end
Set $\kappa := \kappa + 1$
end

In this algorithm, the parameter φ is used to modify the value of the parameter v at each iteration. The current solution is represented by y; y^* and $z(y^*)$ are used to denote the best feasible solution found by the algorithm and its corresponding objective function value, respectively.

4.3. Post-optimization method

In the last step of Algorithm 1, a post-optimization method is applied. Initially, all vehicle routes in the given solution start from the depot at time 0. In our post-optimization method, the departure time of each vehicle route from the depot is shifted iteratively by the amount of M minutes until no improvement in the total weighted cost of that route is seen. In this way, the balance between early and late servicing is improved. In addition, a reduction is provided in the total service cost component in the objective function. According to the results of preliminary tests, we observe that our post-optimization method reduces the total service cost with a ρ value of 0.5 by approximately 21% on average. This reduction leads to an improvement in the objective function value by approximately 1.3% on average.

5. Computational results

We experiment with sets from Solomon [3]. Each VRPTW instance contains one depot and 100 customers. Capacity of all vehicles (Q) is taken from [3]. For each instance, we apply different CV values of the travel time per unit distance to compare solutions with respect to the variability. In all experiments, the expected travel times are equal to the corresponding Euclidean distances.

We set $w=480$, $\rho = 0.00, 0.25, 0.50, 0.75, 1.00$, $M=15$, and (c_d, c_e, c_t, c_f, c_o) are equal to (1.00, 0.10, 1.00, 400, 5/6), respectively.

Table 1
Parameters used by initialization algorithm to construct starting solutions.

β_1	1.00	0.00	0.50	0.80	0.00	0.40	0.60	0.00	0.30	0.40	0.00
β_2	0.00	1.00	0.50	0.00	0.80	0.40	0.00	0.60	0.30	0.00	0.40
β_3	0.00	0.00	0.00	0.20	0.20	0.20	0.40	0.40	0.40	0.60	0.60
β_1	0.20	0.20	0.00	0.10	0.10	0.00	0.05	0.00	γ	1.00	1.00
β_2	0.20	0.00	0.20	0.10	0.00	0.10	0.05	0.00	η	1.00	2.00
β_3	0.60	0.80	0.80	0.80	0.90	0.90	0.90	1.00	CV	1.00	1.00

The algorithms are coded in JAVA and all experiments are run on an Intel Core Duo with 2.93 GHz and 4 GB of RAM.

5.1. Constructing initial feasible solutions

In our initialization algorithm, we use the parameters given by Table 1, which leads to 38 runs in total. For the CV value given in this table, the corresponding parameters (α, λ) are set to (1.00, 1.00). In Step (1a) of Algorithm 1, the initialization algorithm selects the solution with the minimum total transportation cost among 38 solutions to construct IFS. In addition to IFS, two more types of solutions are used as an alternative starting point in the second phase. One type of these solutions ($AIFS_1$) is generated by our initialization algorithm by selecting the solution with the minimum total weighted cost among 38 solutions. The second type of the alternative initial solutions ($AIFS_2$) is based on the deterministic optimal/best-known solutions. The latter solutions, which have been reported in the literature, are the optimal/best-known solutions provided for the well-known VRPTW instances with deterministic parameters (see [29,30] for optimal solutions of nine previously open instances). These solutions are evaluated under travel time stochasticity by having soft time windows to calculate their cost components.

5.2. Parameter calibration in Tabu Search method

We conduct a number of preliminary tests to determine the most appropriate values of the parameters used in Tabu Search method. To calibrate our parameters, we follow an approach similar to Cordeau et al. [31]. Experiments are carried out successively where different values of one parameter are tested by leaving the values of other parameters unchanged. We obtain three sets of results by testing μ, φ, and ϑ over the intervals [0.005,0.025], [0.25,1.25], and $[5\log_{10}|N|, 15\log_{10}|N|]$, respectively. In each set, we observe that using different values of the parameter tested does not lead to significant variability in the results. Moreover, the most appropriate values found by Cordeau et al. [4] provide very good final solutions in a reasonable amount of time in our setting as well. Therefore, we set the values of μ, φ, and ϑ to 0.015, 0.5, and the nearest integer to $7.5\log_{10}|N|$, respectively (following Cordeau et al. [4]).

Recall that the parameter v is dynamically adjusted at each iteration. We set its initial value to 1 which is a reasonable cost to charge one unit of capacity violation.

In Step (1b) of Algorithm 1, (θ, τ) are (500, 100) and (α, λ) are (1.00, 1.00). In Step (3), (θ, τ) are (2000, 500) and (α, λ) are (16, 0.0625), (1.00, 1.00) and (0.0625, 16). Three different CV values are used in the latter step. Accordingly, the objective function value of each initial solution is calculated with respect to the applied CV value.

5.3. Results

Table 2 provides the details of the initial feasible solutions, solutions generated by Tabu Search and final solutions obtained by post-optimization method for all sets where $\rho = 0.50$ and

Table 2
Details of solutions for all sets with $\rho = 0.50$ and CV = 1.00.

Set	Initial solution					Solution of Tabu Search				Final solution		Imp. % in SC component
	Type	TC	SC	OFV	CPU	TC	SC	OFV	CPU	SC	OFV	
RC1	IFS	4934.35	1298.89	1.00	348.50	5144.95	219.63	0.61	1171.63	219.52	0.61	0.05
RC1	$AIFS_1$	6350.16	145.99	0.70	30.63	6180.57	54.66	0.65	1076.13	53.36	0.65	2.38
RC1	$AIFS_2$	6338.13	91.87	0.68	0.88	5982.91	49.00	0.62	1045.63	47.69	0.62	2.68
RC2	IFS	2291.92	1956.71	1.00	391.38	2690.60	547.69	0.71	884.50	483.81	0.69	11.66
RC2	$AIFS_1$	3226.93	557.19	0.85	139.75	2813.39	344.71	0.69	1023.38	313.20	0.68	9.14
RC2	$AIFS_2$	3522.90	1322.66	1.10	0.75	2927.68	657.99	0.79	969.38	512.83	0.75	22.06
R1	IFS	4559.86	1779.04	1.00	376.83	4735.04	362.93	0.61	1115.08	362.93	0.61	0.00
R1	$AIFS_1$	6034.00	155.08	0.71	32.67	5817.22	78.83	0.66	1057.42	77.61	0.66	1.55
R1	$AIFS_2$	6478.46	124.59	0.74	0.58	5891.12	62.66	0.66	1167.92	61.24	0.66	2.25
R2	IFS	2233.84	1897.48	1.00	362.00	2573.10	316.82	0.64	925.45	272.43	0.63	14.01
R2	$AIFS_1$	2923.98	533.05	0.79	162.55	2583.00	279.72	0.64	1077.64	258.52	0.63	7.58
R2	$AIFS_2$	3096.19	1167.87	0.99	0.55	2718.87	472.26	0.72	1102.18	381.35	0.69	19.25
C1	IFS	9322.88	4656.92	1.00	359.67	9721.57	137.79	0.53	934.56	94.80	0.53	31.20
C1	$AIFS_1$	9832.69	322.10	0.56	46.11	9583.57	140.46	0.53	985.22	75.97	0.52	45.92
C1	$AIFS_2$	9018.69	17.89	0.49	0.56	9031.96	7.69	0.49	695.78	6.88	0.49	10.55
C2	IFS	8787.75	11 626.05	1.00	346.75	9044.28	219.01	0.52	894.50	123.18	0.52	43.75
C2	$AIFS_1$	9285.86	1876.27	0.61	168.63	9082.55	324.11	0.53	963.75	230.87	0.53	28.77
C2	$AIFS_2$	8581.41	12.66	0.49	0.63	8581.63	9.88	0.49	634.50	9.66	0.49	2.25

Table 3
Superiority of IFS over $AIFS_1$ and $AIFS_2$.

Set	# of best sol. starting with IFS	Sup. of IFS over $AIFS_1$			Sup. of IFS over $AIFS_2$		
		Best	Worst	Ave.	Best	Worst	Ave.
RC1	5	0.10	0.04	0.07	0.07	0.01	0.04
RC2	13	0.02	0.01	0.01	0.33	0.03	0.14
R1	9	0.11	0.01	0.05	0.12	0.03	0.07
R2	14	0.02	0.01	0.01	0.25	0.01	0.10

CV = 1.00. Note that the values of Transportation Cost (TC) and Service Cost (SC), Objective Function Values (OFV), CPU times, and percentages of improvement in SC component provided by post-optimization method represent the average values calculated over all instances in the set considered. CPU times are reported in seconds. We do not report the computational time spent by post-optimization method to improve the solution found by Tabu Search method since this value is next to 0 for each instance.

These results indicate that our Tabu Search method performs well in different network structures and obtains very good solutions in a reasonable amount of time. Moreover, post-optimization method provides very significant improvements in SC component. With respect to the solutions generated by Tabu Search method, all best results of RC1, RC2, R1 and R2 sets can be found by starting with IFS. With respect to the final solutions obtained by post-optimization method, all best results of RC1, R1 and R2 can be found by starting with IFS. For C1 and C2 sets, all best results can be generated by starting with $AIFS_2$. The reason behind this situation is explained in Section 5.3.4.

In what follows, we discuss different aspects in detail.

5.3.1. *Effects of initial feasible solutions*

Within each set, the final solutions obtained are assessed with respect to the average values calculated for each of the 3 CV and 5 ρ values, leading to 15 results for each set. Table 3 provides the number of best results found by starting with IFS, and its superiority over $AIFS_1$ and $AIFS_2$ with respect to the average objective function values (average weighted costs) of final solutions.

According to the average objective function values, all best results out of 15 final solutions of the C1 set and all best results out of 15 final solutions of the C2 set can be generated by starting with $AIFS_2$. Comparing final solutions according to the starting points, we observe that the average weighted costs of the final solutions obtained by starting with IFS are higher than the best results, 3% on average both in C1 problem instances and in C2 problem instances. Additionally, we may not have the deterministic optimal solutions of real-life problems. Therefore, we conclude that it is reasonable to start with IFS in our Tabu Search.

5.3.2. *Effects of CV and ρ values*

The following figures represent the average service and average transportation costs of the final solutions of instances in RC, R, and C sets where Tabu Search algorithm starts with IFS. Note that in all these figures, within a CV value, ρ values are increasing along the axis of the average transportation costs. In other words, for all sets the average transportation costs are increasing in all cases as the value of ρ increases within the same CV value.

In the RC1 set (Fig. 1(a)), the average service costs are increasing in all cases as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in 23 cases out of 30. In the RC2 set (Fig. 1(b)), the average service costs are increasing in 13 cases out of 15 as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in all cases.

In the R1 set (Fig. 2(a)), the average service costs are increasing in all cases as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in 20 cases out of 30. In the R2 set (Fig. 2(b)), the average service costs are increasing in 13 cases out of 15 as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in all cases.

In the C1 set (Fig. 3(a)), the average service costs are increasing in 14 cases out of 15 as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in 27 cases out of 30. In the

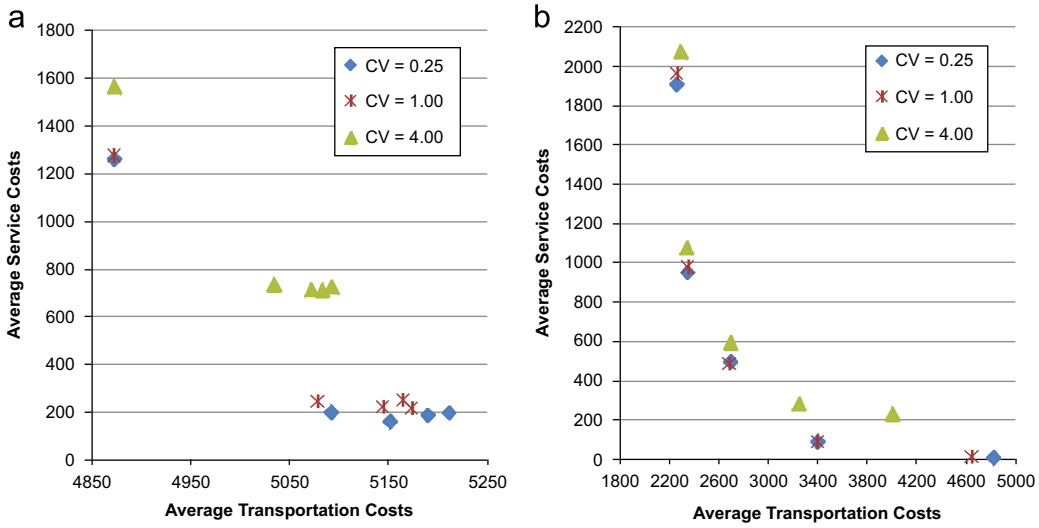

Fig. 1. Average cost values of the final solutions of RC sets obtained by starting with IFS. (a) Average cost values of the RC1 set and (b) average cost values of the RC2 set.

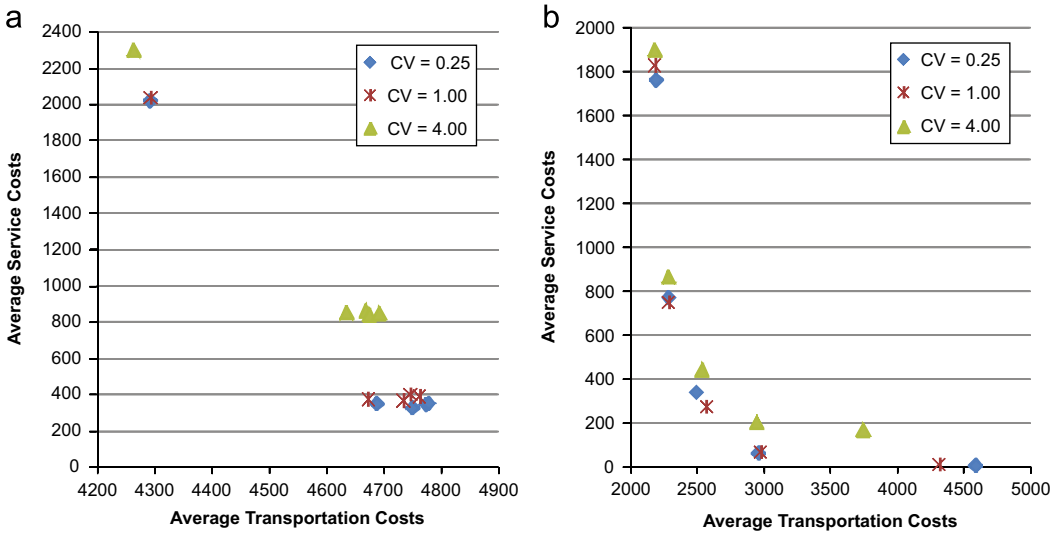

Fig. 2. Average cost values of the final solutions of R sets obtained by starting with IFS. (a) Average cost values of the R1 set and (b) average cost values of the R2 set.

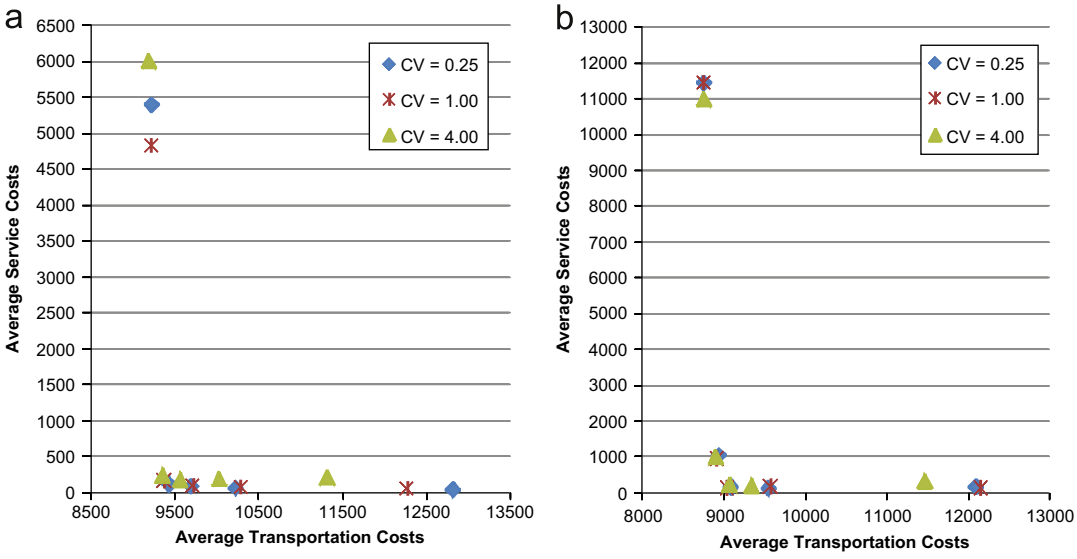

Fig. 3. Average cost values of the final solutions of C sets obtained by starting with IFS. (a) Average cost values of the C1 set and (b) average cost values of the C2 set.

C2 set (Fig. 3(b)), the average service costs are increasing in 9 cases out of 15 as the value of CV increases within the same ρ value. As the value of ρ increases within the same CV value, the average service costs are decreasing in 25 cases out of 30.

5.3.3. Effects of considering stochastic travel times

We compare our final solutions obtained by starting with IFS with the deterministic optimal/best-known solutions that correspond to $AIFS_2$ in our solution approach. We focus on the cases where $\rho = 0.50$ and CV = 4.00. Figs. 4(a), 5(a) and 6(a) represent the comparison of the customers' expected arrival times for the instances RC103, R201 and RC202, respectively. Moreover, Figs. 4(b), 5(b) and 6(b) provide the comparison of the customers' sequences for the instances RC103, R201 and RC202, respectively.

From these figures, we observe some extreme situations in which customers are served much later (or earlier) in our final solutions than in $AIFS_2$. Some examples of the customers that are visited later in our solutions are customers 61, 12 and 81 in Figs. 4(a), 5(a) and 6(a), respectively. These customers are served by $AIFS_2$ too early which either leads to very high expected earliness values or prevents serving other customers in their routes reliably and efficiently. In our final solutions, these exemplified customers are visited within their time windows. Thus, we have reasonable expected delay and expected earliness values. In addition, shifting these customers in routes allows us to construct the final solutions which have lower total weighted costs than $AIFS_2$. Some examples of the customers that are visited earlier in our solutions are customers 90 and 94 in Figs. 4(a) and 6(a), respectively. These customers have only due dates for deliveries because of the fact that the lower bound of their time windows is 0. Therefore, giving precedence to such customers in routes leads to very low expected delay values.

Note that Figs. 4(b), 5(b) and 6(b) include the segments which are parallel to the 45-degree line. These segments correspond to parts of routes (sub-routes) that are constructed both by our final solutions and by $AIFS_2$. These sub-routes are visited by our solutions earlier (or later) to have reliable and efficient routes.

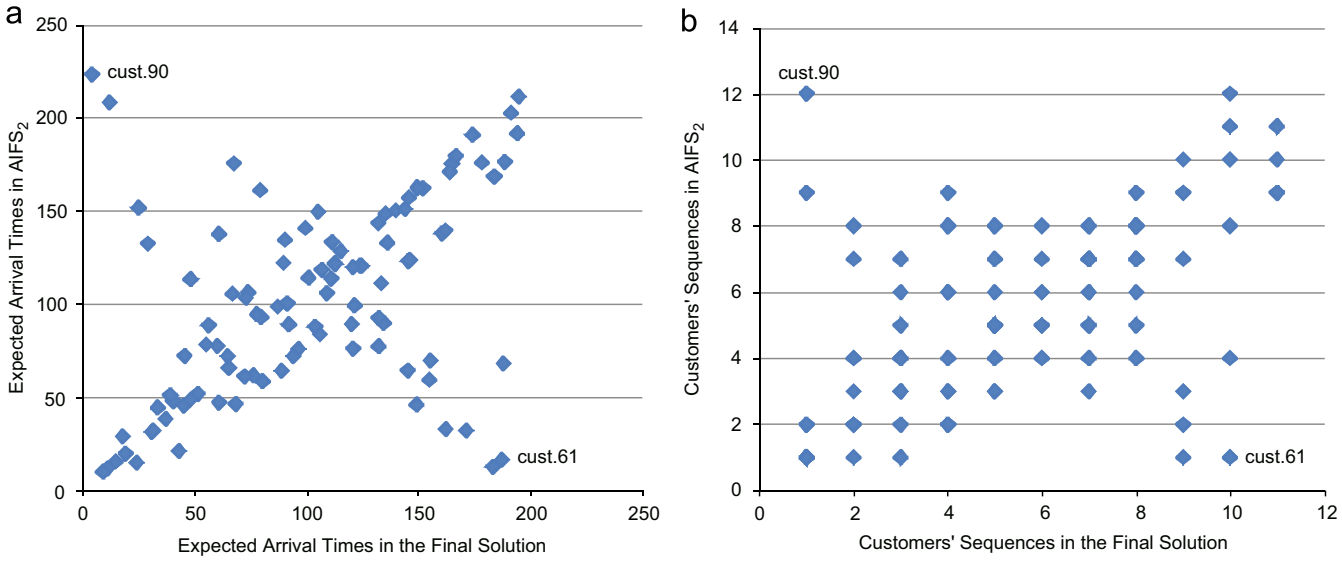

Fig. 4. Comparison of the final solutions with $AIFS_2$ for the instance RC103. (a) Comparison of the customers' expected arrival times for RC103 and (b) comparison of the customers' sequences for RC103.

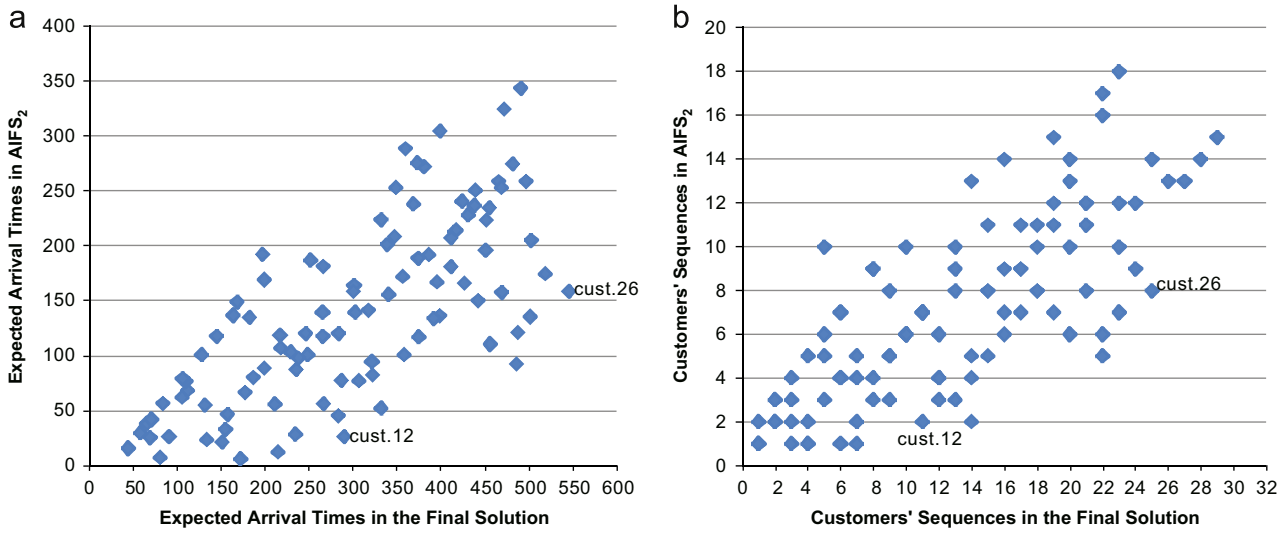

Fig. 5. Comparison of the final solutions with $AIFS_2$ for the instance R201. (a) Comparison of the customers' expected arrival times for R201 and (b) comparison of the customers' sequences for R201.

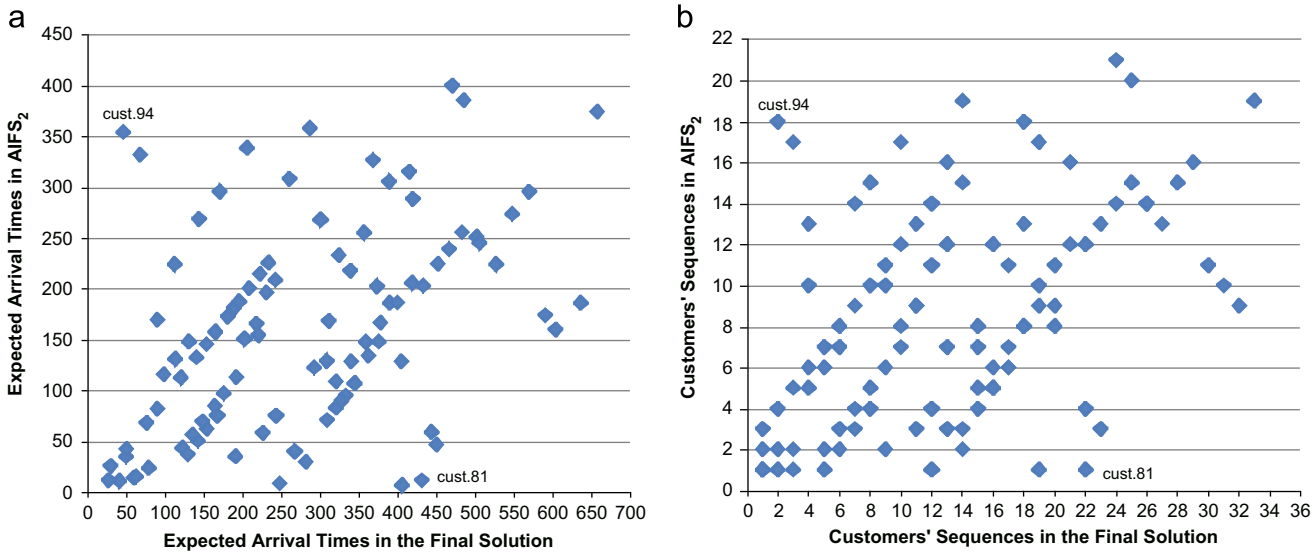

Fig. 6. Comparison of the final solutions with $AIFS_2$ for the instance RC202. (a) Comparison of the customers' expected arrival times for RC202 and (b) comparison of the customers' sequences for RC202.

5.3.4. *Effects of network structures*

Recall that $AIFS_2$ provides the best results for all instances in C1 and C2 sets with respect to the average weighted costs. This is due to the fact that the time windows in these instances have been determined according to the arrival times at the customers (see [3]). Therefore, the deterministic optimal results have very low service costs, leading to very low service costs in the final solutions (except the case with $\rho = 0$ where the service costs do not play a role in the objective function). The deterministic optimal solutions have low total distances and low number of vehicles as well since their aim is to minimize the total distance for the VRPTW and the customers appear in clusters in C sets.

The service time of each customer is given as 90 time units in C sets whereas it is equal to 10 time units both in R and in RC sets. Accordingly, the total expected overtime values are very high in all initial feasible solutions and in their final solutions of the instances in C sets. The effect of this situation can be seen in Fig. 3(a) and (b) in the average transportation cost values.

RC2, R2, and C2 sets have instances with wide time windows compared to the time windows of instances in RC1, R1, and C1 sets. Due to this structural property, the average service and transportation costs of RC2, R2, and C2 sets are less sensitive to the variability in the travel times than those of RC1, R1, and C1 sets, respectively (see Figs. 1(a), (b), 2(a), (b), 3(a), and (b) for effects of different network structures).

5.4. *Managerial insights*

The VRP studied in this paper originates by the fact that the stochasticity in the travel times should be taken into account to have reliable routing decisions. Having a small increase in the total transportation cost, we observe a significant decrease in the total service cost. For example, focusing on cases where ρ takes the value 0.25 from 0.00 within CV=0.25, we see that a 4.37% increase in the total transportation cost on average leads to a 77.08% decrease in the total service cost on average over all problem instances. A manager might take the advantage of a significant service cost reduction by having a very small increase in the transportation cost with the same number of vehicles.

6. Conclusions and future research

Considering reliability and efficiency in a stochastic way in the VRPTW is important as it extends the current body of knowledge closer to the real-life environment. Clearly, assuming a deterministic and static world leads to suboptimal solutions which need to be further improved in the real-life applications. In this paper, we focus on a vehicle routing problem with soft time windows and stochastic travel times. Our aim is to construct reliable and efficient routes by means of the stochastic measures defined in this study. We propose a model that obtains meaningful combinations of the transportation costs and the service costs with respect to the carrier companies' priorities. Additionally, we propose a solution approach with three main phases to solve our model. In the first phase, an initial feasible solution is constructed in two stages (IFS). This solution is improved by applying a Tabu Search method in the second phase. Finally, a further improvement is provided by a post-optimization method.

We apply our solution approach to the experiments conducted for well-known problem instances which have different structures. We test our Tabu Search algorithm by starting with distinct initial solutions. Results show that the Tabu Search method performs well in each network structure and provides very good final results in a reasonable amount of time. We find that most of the best results are obtained by improving the IFS. The network structure has a significant effect on the performance of the initial solutions with respect to their corresponding final solutions. Finally, we analyze the effect of variability on the service costs and the effect of the priority of reliability on two main cost components. We observe that the variability in the travel time per unit distance has a direct effect on the cost of servicing. In addition, our model is very successful to create various solution options with respect to the company preferences.

Future research includes the time-dependency of the travel times for the same VRPTW setting. In real-life applications, it is important to consider time-dependent travel times since speeds vary throughout the day due to the events like accidents or congestion during the rush hours. By including this property, we have both stochastic and dynamic travel times. This structure requires adjustments in the algorithms with respect to the distributions of the arrival times.

Acknowledgments

The authors would like to thank the anonymous referee and the editor for their careful consideration and valuable comments. They also acknowledge Guy Desaulniers (École Polytechnique de Montréal, Canada) and Roberto Baldacci (University of Bologna, Italy) for providing the deterministic optimal/best-known solutions.

References

[1] Gendreau M, Laporte G, Séguin R. Stochastic vehicle routing. European Journal of Operational Research 1996;88:3–12.
[2] Laporte G. What you should know about the vehicle routing problem. Naval Research Logistics 2007;54:811–9.
[3] Solomon MM. Algorithms for the vehicle routing and scheduling problems with time window constraints. Operations Research 1987;35(2):254–65.
[4] Cordeau J, Laporte G, Mercier A. A unified tabu search heuristic for vehicle routing problems with time windows. Journal of the Operational Research Society 2001;52:928–36.
[5] Laporte G. The vehicle routing problem: an overview of exact and approximate algorithms. European Journal of Operational Research 1992;59: 345–58.
[6] Baldacci R, Toth P, Vigo D. Exact algorithms for routing problems under vehicle capacity constraints. Annals of Operations Research 2010;175(1): 213–45.
[7] Bräysy O, Gendreau M. Vehicle routing problem with time windows, part I: route construction and local search algorithms. Transportation Science 2005;39(1):104–18.
[8] Bräysy O, Gendreau M. Vehicle routing problem with time windows, part II: metaheuristics. Transportation Science 2005;39(1):119–39.
[9] Desrosiers J, Dumas Y, Solomon MM, Soumis F. Time constrained routing and scheduling. In: Ball M, Magnanti T, Monma C, Nemhauser G, editors. Network Routing, Handbooks in Operations Research and Management Science, vol. 8. Amsterdam: North-Holland; 1995. p. 35–139.
[10] Laporte G, Louveaux F, Mercure H. The vehicle routing problem with stochastic travel times. Transportation Science 1992;26(3):161–70.
[11] Kenyon A, Morton D. Stochastic vehicle routing with random travel times. Transportation Science 2003;37(1):69–82.
[12] Van Woensel T, Kerbache L, Peremans H, Vandaele N. Vehicle routing with dynamic travel times: a queueing approach. European Journal of Operational Research 2008;186(3):990–1007.
[13] Stewart W, Golden B. Stochastic vehicle routing: a comprehensive approach. European Journal of Operational Research 1983;14:371–85.
[14] Ando N, Taniguchi E. Travel time reliability in vehicle routing and scheduling with time windows. Networks and Spatial Economics 2006;6:293–311.
[15] Russell RA, Urban TL. Vehicle routing with soft time windows and erlang travel times. Journal of the Operational Research Society 2008;59:1220–8.
[16] Li X, Tian P, Leung S. Vehicle routing problems with time windows and stochastic travel and service times: models and algorithm. International Journal of Production Economics 2010;125:137–45.
[17] Glover F. Tabu search, part I. ORSA Journal on Computing 1989;1(3):190–206.
[18] Glover F. Tabu search, part II. ORSA Journal on Computing 1990;2(1):4–32.
[19] Gendreau M, Hertz A, Laporte G. A tabu search heuristic for the vehicle routing problem. Management Science 1994;40(10):1276–90.
[20] Gendreau M, Laporte G, Séguin R. A tabu search heuristic for the vehicle routing problem with stochastic demands and customers. Operations Research 1996;44(3):469–77.
[21] Rochat Y, Taillard E. Probabilistic diversification and intensification in local search for vehicle routing. Journal of Heuristics 1995;1:147–67.
[22] Garcia B, Potvin J, Rousseau J. A parallel implementation of the tabu search heuristic for vehicle routing problems with time window constraints. Computers and Operations Research 1994;21(9):1025–33.
[23] Taillard E, Badeau P, Gendreau M, Guertin F, Potvin J. A tabu search heuristic for the vehicle routing problems with soft time windows. Computers and Operations Research 1997;31(2):170–86.
[24] Gendreau M, Hertz A, Laporte G, Stan M. A generalized insertion heuristic for the traveling salesman problem with time windows. Operations Research 1998;43(3):330–5.
[25] Gendreau M, Hertz A, Laporte G. New insertion and postoptimization procedures for the traveling salesman problem. Operations Research 1992;40(6):1086–94.
[26] Fan Y, Kalaba R, Moore J. Arriving on time. Journal of Optimization Theory and Applications 2005;127(3):497–513.
[27] Kaparias I, Bell M, Belzner H. A new measure of travel time reliability for in-vehicle navigation systems. Journal of Intelligent Transportation Systems 2008;12(4):202–11.
[28] Dellaert NP, de Kok A, Wei W. Push and pull strategies in multi-stage assembly systems. Statistica Neerlandica 2000;54(2):175–89.
[29] Desaulniers G, Lessard F, Hadjar A. Tabu search, partial elementarity, and generalized *k*-path inequalities for the vehicle routing problem with time windows. Transportation Science 2008;42(3):387–404.
[30] Baldacci R, Mingozzi A, Roberti R. New route relaxation and pricing strategies for the vehicle routing problem. Operations Research 2011;59(5):1269–83.
[31] Cordeau J, Gendreau M, Laporte G. A tabu search heuristic for periodic and multi-depot vehicle routing problems. Networks 1997;30:105–19.