Vehicle routing with soft time windows and stochastic travel times : a column generation and branch-and-price solution approach

Document status and date:
Published: 01/01/2012

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

Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation and Branch-and-Price Solution Approach

D. Tas, M. Gendreau, N. Dellaert, T. van Woensel,
A.G. de Kok

Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation and Branch-and-Price Solution Approach

D. Taş[a,*], M. Gendreau[b], N. Dellaert[a], T. van Woensel[a], A.G. de Kok[a]

[a]*School of Industrial Engineering and Innovation Sciences, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, The Netherlands*
[b]*CIRRELT and MAGI, École Polytechnique, C.P. 6079, Succursale Centre-ville, Montréal, Québec, Canada H3C 3A7*

Abstract

We study a vehicle routing problem with soft time windows and stochastic travel times. In this problem, we consider stochastic travel times to obtain routes which are both efficient and reliable. In our problem setting, soft time windows allow early and late servicing at customers by incurring some penalty costs. The objective is to minimize the sum of transportation costs and service costs. Transportation costs result from three elements which are the total distance traveled, the number of vehicles used and the total expected overtime of the drivers. Service costs are incurred for early and late arrivals; these correspond to time-window violations at the customers. We apply a column generation procedure to solve this problem. The master problem can be modeled as a classical set partitioning problem. The pricing subproblem, for each vehicle, corresponds to an elementary shortest path problem with resource constraints. To generate an integer solution, we embed our column generation procedure within a branch-and-price method. Computational results obtained by experimenting with well-known problem instances are reported.

Keywords:
Routing, Stochastic travel times, Column generation, Branch-and-Price

*Corresponding author
Email address:* `d.tas@tue.nl` (D. Taş)

1. Introduction

The Vehicle Routing Problem (VRP), sometimes referred to as capacitated VRP, aims to find a set of feasible routes that start and end at the depot to serve a set of customers. Each customer, given with a known demand, is visited exactly once by one vehicle. Each route has a total demand that cannot exceed the vehicle capacity. The objective is to minimize the total cost, traditionally constructed by the sum of distances traveled or the number of vehicles used or a combination of these. The interested reader is referred to Toth and Vigo [31], and Laporte [21, 22] for comprehensive literature surveys about the VRP. This problem is extended by considering different customer service aspects such as starting the service at each customer within a given time interval, called the Vehicle Routing Problem with Time Windows (VRPTW). Time windows are called soft when they can be violated with some penalty costs. They are called hard when violations are not permitted, i.e., vehicles are allowed to wait with no cost if they arrive early and they are prohibited to serve if they arrive late. For reviews on the VRPTW the reader is referred to Bräysy and Gendreau [4, 5], Gendreau and Tarantilis [17], and Kallehauge [20].

In the classical formulation of the VRP, all problem elements are deterministic. However, carrier companies have to deal with various types of uncertainty in real-life applications. The quality of the service becomes quite poor if uncertainties are disregarded at the planning level. To overcome the inefficiency incurred at the operational level, stochastic variants of the VRP have been introduced (see Gendreau et al. [16] for a review on stochastic routing problems). Common parameters considered in these variants are stochastic demands, stochastic customers and stochastic travel times. In this research, we study a version of the VRP where we focus on stochastic travel times with a known probability distribution. Using stochastic travel times enables us to construct both reliable and efficient routes. In addition to the cost effectiveness, we also consider customer service aspects where each customer has a soft time window that allows early and late servicing.

For our problem, we consider the formulation introduced in Taş et al. [29] where the authors focus on modeling aspects and on solving the problem effectively with a new solution procedure based on metaheuristics. The study conducted in [29] extends existing models, which are generated for stochastic routing problems, by proposing a one-stage formulation in which the objective function copes with time-window violations and expected over-

time, and all constraints are linear (see Ando and Taniguchi [1], Russell and Urban [26], and Li et al. [23] for existing models). In this formulation, the objective is to minimize the sum of transportation costs and service costs. Transportation costs result from three elements which are the total distance traveled, the number of vehicles used and the total expected overtime of the drivers. Service costs are incurred for early and late arrivals; these correspond to time-window violations at the customers. Various solution options can be provided to carrier companies by generating different combinations of two main cost components. In our study, we optimally solve this model which provides solutions by placing importance both on efficiency and on reliability. The interested reader is referred to Taş et al. [29] for an in-depth discussion about the framework of the model.

To obtain the optimal solution, we apply a column generation procedure to solve the model described above (see Lübbecke and Desrosiers [24], and Desaulniers et al. [9] for comprehensive surveys on column generation). In our procedure, the master problem can be modeled as a classical set partitioning problem. The pricing subproblem, for each vehicle, corresponds to an elementary shortest path problem with resource constraints. This column generation procedure is embedded within a branch-and-price method to obtain integer solutions. The branch-and-price method is very successful among the exact methods recently applied to the deterministic and stochastic variants of the routing problems. Some applications can be seen in Fischetti et al. [15], Desrochers et al. [11], Chabrier [6], Irnich and Villeneuve [19], and Christiansen and Lysgaard [7]. To our knowledge, no research has studied exact methods to solve the VRP with soft time windows and stochastic travel times. Our paper extends the related literature by obtaining the optimal solution to the described problem.

The paper is organized as follows. The problem and the formulation used in this paper are introduced in Section 2. The column generation procedure is explained in Section 3. The pricing problem with a dominance relation newly introduced in this study is reported in Section 4. Then, we describe the framework of branch-and-price algorithm in Section 5 and report numerical results on Solomon's problem instances [28] in Section 6. Finally, we present our conclusions in Section 7.

2. Problem Description and Formulation

Let $G = (N, A)$ be a connected digraph where $N = \{0, 1, ..., n\}$ is the set of nodes and A is the set of arcs. In this graph, node 0 denotes the depot, and nodes 1 to n represent customers. A distance d_{ij}, and a travel time T_{ij} with a known probability distribution are defined for each arc (i, j), where $i \neq j$. With each customer $i \in N \setminus 0$, is associated a positive demand q_i, a positive service time s_i, and a soft time window $[l_i, u_i]$ where l_i and u_i are non-negative parameters. Soft time windows enable to serve customers outside their time windows, which incurs some penalty costs to the company for early and late servicing. The scheduling horizon for the problem is represented by $[l_0, u_0]$ that is the time window given for the depot. Furthermore, a homogeneous fleet of vehicles of equal capacity (Q) is located at the depot. These vehicles, given in set V, are not allowed to wait at customer locations in case of arriving early; service must take place immediately.

In this paper, we focus on the mathematical formulation introduced in Taş et al. [29]. We first summarize the notation used in this formulation in Table 1.

Table 1: The notation used in the mathematical model

x_{ijv}	:	equal to 1 if vehicle v covers arc (i, j), 0 otherwise
$\mathbf{x}$	:	vector of vehicle assignments and customer sequences in these vehicle routes, where $\mathbf{x} = \{x_{ijv} \mid i, j \in N, v \in V\}$
$D_{jv}(\mathbf{x})$	:	*expected* delay at node j when it is served by vehicle v
$E_{jv}(\mathbf{x})$	:	*expected* earliness at node j when it is served by vehicle v
$O_v(\mathbf{x})$	:	*expected* overtime of the driver working on route of vehicle v
c_d	:	penalty cost paid for one unit of delay
c_e	:	penalty cost paid for one unit of earliness
c_t	:	cost paid for one unit of distance
c_o	:	cost paid for one unit of overtime
c_f	:	fixed cost paid for each vehicle used for servicing

The model, which is solved by applying exact algorithms in our paper, can then be stated as follows:

$$\min \quad \sum_{v \in V} \left[\rho \frac{1}{C_1} \left(c_d \sum_{j \in N} D_{jv}(\mathbf{x}) + c_e \sum_{j \in N} E_{jv}(\mathbf{x}) \right) \right.$$

$$\left. + (1 - \rho) \frac{1}{C_2} \left(c_t \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ijv} + c_f \sum_{j \in \{N \backslash 0\}} x_{0jv} + c_o O_v(\mathbf{x}) \right) \right] \quad (1)$$

$$\text{subject to} \sum_{j \in N} \sum_{v \in V} x_{ijv} = 1, \qquad\qquad i \in N \setminus \{0\}, \quad (2)$$

$$\sum_{i \in N} x_{ikv} - \sum_{j \in N} x_{kjv} = 0, \qquad k \in N \setminus \{0\}, v \in V, \quad (3)$$

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

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

$$\sum_{i \in N \setminus \{0\}} q_i \sum_{j \in N} x_{ijv} \leq Q, \qquad\qquad v \in V, \quad (6)$$

$$\sum_{i \in B} \sum_{j \in B} x_{ijv} \leq |B| - 1, \qquad B \subseteq N \setminus \{0\}, v \in V, \quad (7)$$

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

The objective function (1) minimizes the total weighted cost which has two main components, service costs and transportation costs. The constraints (2) ensure that each customer is visited exactly once. The constraints (3) satisfy the conservation of flow at each customer for each vehicle. The constraints (4) and (5) indicate that every vehicle route must start from the depot and end at the depot. The constraints (6) state that each vehicle can be loaded up to its capacity. The constraints (7) ensure the subtour elimination, and (8) are the integrality constraints. Parameter ρ is needed to obtain various combinations of the two main cost components by adjusting their values with scaling parameters C_1 and C_2. The interested reader is referred to Taş et al. [29] for the details about these parameters and their calculations.

To make the problem tractable, it is assumed that the random traversal time spent for one unit of distance follows a suitable Gamma distribution

with shape parameter α and scale parameter λ. This approach leads to Gamma distributed arc traversal times where shape parameters are obtained by scaling α with respect to the distance of the corresponding arc. Since vehicles do not wait at customer locations, the arrival time of a vehicle at a node along its route can be defined as the sum of travel times on arcs covered by the vehicle until that node. The latter calculation requires an adjustment to the time window at the visited node by taking into account the cumulative service time. Gamma distributed arrival times are then derived where shape parameters are obtained by scaling α with respect to the total distance of the arcs covered. These definitions enable us to calculate expected delay, expected earliness and expected overtime values exactly by using an approach similar to that given in Dellaert et al. [8]. Note that expected delay and expected earliness values, and thus the total service cost of a route are computed with respect to the optimal starting time of that route from the depot (see Section 4.1 for the related calculations).

3. Column Generation

The interested reader is referred to Desrosiers et al. [12] for the details about the column generation method. In the following, we present the master problem and the pricing subproblem of the column generation method proposed for the model formulation explained in Section 2.

The Master Problem: The master problem, which corresponds to the constraints (2) in the original model formulation, can be modeled as a set partitioning problem as follows:

$$\min \quad \sum_{p \in P} K_p y_p \tag{9}$$

$$\text{subject to} \sum_{p \in P} a_{ip} y_p = 1, \qquad i \in N \setminus \{0\}, \tag{10}$$

$$y_p \in \{0, 1\}, \qquad p \in P, \tag{11}$$

where P is the set of all feasible vehicle routes that start from the depot and end at the depot. Here, K_p is the total weighted cost of route p, and a_{ip} is 1 if customer i is served by route p and 0, otherwise. The decision variable y_p is 1 if route p is selected by the solution and 0, otherwise.

The Pricing Subproblem: The pricing subproblem for each vehicle v, which corresponds to the constraints (3)-(8) in the original model formulation, is an Elementary Shortest Path Problem with Resource Constraints (ESPPRC). Note that in our problem, we have merely the capacity of the vehicles as the resource. Then, the subproblem for a given vehicle v can be modeled as follows:

$$\min \quad \overline{K}_p \tag{12}$$

$$\text{subject to} \quad (3) - (8), \tag{13}$$

where p corresponds to the route of vehicle v and $\overline{K}_p$ is the reduced cost of route p. The latter is calculated by:

$$
\begin{aligned}
\overline{K}_p = & K_p - \sum_{i \in \{N \setminus 0\}} a_{ip} u_i \\
= & \rho \frac{1}{C_1} \left(c_d \sum_{j \in N} D_{jv}(\mathbf{x}) + c_e \sum_{j \in N} E_{jv}(\mathbf{x}) \right) \\
& + (1 - \rho) \frac{1}{C_2} \left(c_t \sum_{i \in N} \sum_{j \in N} d_{ij} x_{ijv} + c_f \sum_{j \in \{N \setminus 0\}} x_{0jv} + c_o O_v(\mathbf{x}) \right) - \sum_{i \in \{N \setminus 0\}} a_{ip} u_i,
\end{aligned}
\tag{14}
$$

where u_i, $i \in \{N \setminus 0\}$ is the dual price associated with the constraints (10).

In column generation algorithm, we start with solving a Restricted Linear Programming Master Problem (RLPMP) in which the constraints (11) are relaxed and only the vehicle routes of an initial feasible solution are included. These initial routes constitute a subset of all feasible vehicle routes in the formulation (9) - (11). We then solve our pricing subproblem by using the optimal dual values obtained by solving the RLPMP. If a new vehicle route with negative reduced cost is found by the pricing subproblem, it is added to the RLPMP and this problem is re-optimized to obtain new optimal dual values. Otherwise, we terminate the algorithm since the optimal solution to the linear programming relaxation of the formulation (9) - (11) is reached. For the first step where the algorithm is initiated, we construct an initial feasible solution by using the initialization algorithm introduced in Taş et al. [29]. This algorithm modifies the insertion heuristic I1 given in Solomon [28] by taking into account the expected violations of the time windows calculated with respect to the stochastic travel times. The interested reader is referred to [29] for the details about this initialization algorithm.

7

4. Elementary Shortest Path Problem with Resource Constraints

We solve our pricing subproblem with the algorithm of Feillet et al. [13] by applying the state space augmentation (decremental state space relaxation) technique of Boland et al. [3], and Righini and Salani [25]. The algorithm proposed in [13] to solve the ESPPRC is based on the label correcting reaching algorithm of Desrochers [10]. In the latter algorithm, labels are used to denote the paths on nodes. Each label on a node represents a path from the depot to that node by specifying the cost of the path and the consumption of the resources along the path. The label correcting reaching algorithm repeatedly treats the nodes in which each new label on the treated node is extended to each possible successor node. If the algorithm cannot generate any new labels, then it is terminated. Feillet et al. [13] extend this classical label correcting algorithm, which had been developed for the non-elementary shortest path problem with resource constraints, by including node resources to solve the ESPPRC optimally. Beasley and Christofides [2] propose the idea of adding a binary resource for each node in the graph but they do not conduct any computational experiments for that. Feillet et al. [13] solve the ESPPRC on a full-dimensional state space, and thus they apply the idea of using unreachable nodes in the labels to have an efficient algorithm. A node is unreachable for a path if it cannot be visited by that path either because it has already been visited or because visiting that node would violate at least one resource constraint.

In the state space augmentation algorithm, the problem is relaxed where multiple visits are forbidden for the nodes in a given set $S \subseteq N \setminus \{0\}$. If the optimal solution of the relaxed form of the ESPPRC is elementary, then it is also optimal for the ESPPRC. Otherwise, the state space is augmented by adding some nodes to the set S which are selected with respect to the optimal solution of the relaxed problem.

In our problem, a state $(W_p^1, ..., W_p^R, a_p^S, \mathbf{V}_p^S)$ is associated with each path p from the depot to node i. In that state, $(W_p^1, ..., W_p^R)$ represents the consumption of each of the R resources along the path p. a_p^S denotes the number of nodes in S which are unreachable by path p. $\mathbf{V}_p^S$ is the vector of unreachable nodes in S which is defined by $V_p^b = 1$ if node $b \in S$ is unreachable by path p and 0, otherwise. We represent each path p by a label $(L_p, \overline{K}_p)$ where $L_p = (W_p^1, ..., W_p^R, a_p^S, \mathbf{V}_p^S)$ and $\overline{K}_p$ is the reduced cost of path p which is calculated by Equation (14) with respect to the optimal starting time of path p from the depot (see Section 4.1 for the calculation of the optimal departure

time). Let p and p^* be two distinct paths from the depot to node i where each path starts from the depot at the optimal departure time of its corresponding vehicle. In addition, suppose that these two paths arrive at node i at different times (different expected arrival times). In such a case, for the dominance relation, the starting time of one path (path p) from the depot is adjusted to make this path arrive at node i at the same time as the other path (path p^*). The reduced cost of path p calculated with respect to this adjusted starting time from the depot is denoted by $\overline{K}_{p_{p^*}}$. The dominance relation is then defined as follows:

Definition 4.1. *If p and p^* are two distinct paths from the depot to node i with labels $(L_p, \overline{K}_p)$ and $(L_{p^*}, \overline{K}_{p^*})$, respectively, then path p dominates path p^* if and only if $W_p^r \leq W_{p^*}^r$ for $r = 1,...,R$, $a_p^S \leq a_{p^*}^S$, $V_p^b \leq V_{p^*}^b$ for all $b \in S$, $\overline{K}_p \leq \overline{K}_{p^*}$, $\overline{K}_{p_{p^*}} \leq \overline{K}_{p^*}$ and $(L_p, \overline{K}_p) \neq (L_{p^*}, \overline{K}_{p^*})$.*

A path is called efficient if its corresponding label is non-dominated. The method applied to solve our pricing subproblem is described in Algorithm 1. In that algorithm, Π_i, I, and H_{ij} denote the list of labels on node i, the list of nodes that will be treated and the set of labels extended from node i to node j, respectively. Moreover, $EFF(\Pi)$ is the procedure that removes the dominated labels and keeps only the non-dominated ones in Π. In our problem, we have only one resource constraint which is the capacity of the vehicles (Q). Therefore, we have only W_p^1 in the labels which corresponds to the consumption of the capacity resource along the path p. In addition, w_{ij}^1 represents the consumption of the capacity along the arc (i, j) which is equal to the demand value at node j (q_j). Formally, the extension of a label on node i to node j is defined in Algorithm 2. In this algorithm, $\pi_{p'}$ represents the resulting label obtained by extending label π_p from node i to node j.

Note that in Algorithm 1, the procedure $EFF(\Pi)$ is applied to the list of paths on the ending depot after all nodes are treated. At this step, the elementary paths on the ending depot which are efficient ones with non-negative reduced costs and dominated ones regardless of their reduced costs are kept in an Intermediate Column Pool (ICP), which is similar to the application of the buffer column pool proposed in Savelsbergh and Sol [27]. Instead of solving the ESPPRC immediately after re-optimizing the RLPMP, we first search the ICP with respect to new optimal dual values. The columns with negative reduced costs are then sent from the ICP to the RLPMP. The ESPPRC is solved if we cannot find any such columns in the ICP. At each iteration, we check the size of the ICP to clean it if it is needed. Cleaning

$S \leftarrow \emptyset$
$S' \leftarrow \emptyset$
repeat
    $S \leftarrow S \cup S'$
    $S' \leftarrow \emptyset$
    $\Pi_0 = \{(0,0,\mathbf{0},0)\}$
    forall the $i \in N \setminus \{0\}$ **do**
        $\Pi_i \leftarrow \emptyset$
    end
    $I = \{0\}$
    repeat
        Choose $i \in I$
        forall the $(i,j) \in A$ **do**
            $H_{ij} \leftarrow \emptyset$
            forall the $\pi_p = (W_p^1, a_p^S, \mathbf{V}_p^S, \overline{K}_p) \in \Pi_i$ **do**
                if *(j ∉ S)* or *(j ∈ S and $V_p^j = 0$)* **then**
                    if Extend*(i, π_p, j)* $\neq$ FALSE **then**
                        $H_{ij} \leftarrow H_{ij} \cup \{\text{Extend}(i, \pi_p, j)\}$
                    end
                end
            end
            if $j \in N \setminus \{0\}$ **then**
                $\Pi_j \leftarrow EFF(\Pi_j \cup H_{ij})$
                if Π_j has changed and $j \notin I$ **then**
                    $I \leftarrow I \cup \{j\}$
                end
            end
        end
        $I \leftarrow I \setminus \{i\}$
    until $I = \emptyset$;
    $\Pi_0 \leftarrow EFF(\Pi_0)$
    if There is at least one elementary path on the depot with negative reduced cost **then**
        Send such paths to the RLPMP
    else
        if The minimum reduced cost is negative **then**
            Select the customer with the highest multiplicity in the solution with the minimum reduced cost
            $S' \leftarrow \{\text{selected customer}\}$
        end
    end
until $S' = \emptyset$;

Algorithm 1: Algorithm with state space augmentation technique to solve the ESPPRC

> **if** $W_p^1 + w_{ij}^1 > Q$ **then**
> | return FALSE
> **else**
> > calculate $W_{p'}^1$ and $\overline{K}_{p'}$
> > $a_{p'}^S \leftarrow a_p^S$
> > $\mathbf{V}_{p'}^S \leftarrow \mathbf{V}_p^S$
> > **if** $j \in S$ **then**
> > > $a_{p'}^S \leftarrow a_{p'}^S + 1$
> > > $V_{p'}^j \leftarrow 1$
> >
> > **end**
> > **foreach** $b \in S$ and $(j, b) \in A$ such that $W_{p'}^1 + w_{jb}^1 > Q$ **do**
> > > $a_{p'}^S \leftarrow a_{p'}^S + 1$
> > > $V_{p'}^b \leftarrow 1$
> >
> > **end**
> > return $\pi_{p'} = (L_{p'}, \overline{K}_{p'})$
> **end**

Algorithm 2: Extend(i, π_p, j)

takes place in case the number of columns in the ICP is larger than a threshold value. In this situation, all columns which have been kept for more than a pre-determined number of iterations are removed from the ICP. By this way, we efficiently search the ICP to find the columns with negative reduced costs.

In addition to the strategy described above, we apply one more accelerating method which is related to the pricing subproblem in our solution procedure. At each time we extend a label to the ending depot, we determine the number of efficient elementary paths with negative reduced costs by means of the dominance relation in the comparison of the recently extended path with all other efficient paths on the ending depot. If this number is larger than a threshold value, we stop the ESPPRC and then send all these paths on the ending depot, which are efficient elementary ones with negative reduced costs, to the RLPMP. The interested reader is referred to Feillet et al. [13] and Feillet et al. [14] for similar implementations of this technique.

4.1. Service Cost Component

For the expected values considered in service cost component, we summarize the related notation in Table 2. The equations given in this table are

constructed with respect to the random traversal time which is Gamma distributed with shape parameter α and scale parameter λ. Recall that we have Gamma distributed arc traversal times and Gamma distributed arrival times by means of the definitions explained in Section 2. The interested reader is referred to Taş et al. [29] for explanations about the definitions in detail.

Table 2: The notation used for the calculation of service cost component

$\Gamma(\alpha)$	:	Gamma function, where $\Gamma(\alpha) = \int_0^\infty e^{-r} r^{\alpha-1} dr$
$\Gamma_{\alpha,\lambda}(\delta)$	:	cumulative distribution function, where $\delta \geq 0$ and
		$\Gamma_{\alpha,\lambda}(\delta) = \int_0^\delta \frac{(e^{-z/\lambda})(z)^{\alpha-1}}{\Gamma(\alpha)\lambda^\alpha} dz$
Y_{jv}	:	arrival time of vehicle v at node j
A_{jv}	:	set of arcs covered by vehicle v before visiting node j
α_{jv}	:	shape parameter of Y_{jv}, where $\alpha_{jv} = \alpha \sum_{(l,k) \in A_{jv}} d_{lk}$
λ_{jv}	:	scale parameter of Y_{jv}, where $\lambda_{jv} = \lambda$
s_{jv}	:	total service time spent by vehicle v for servicing until node j
u'_j	:	upper bound of the time window at node j shifted by s_{jv}, where $u'_j = u_j - s_{jv}$
l'_j	:	lower bound of the time window at node j shifted by s_{jv}, where $l'_j = l_j - s_{jv}$
$E[T_{ij}]$	:	mean of travel time on arc (i,j), where $E[T_{ij}] = \alpha\lambda d_{ij}$
$E[Y_{jv}]$	:	mean of Y_{jv}, where node j is visited immediately after node i and $E[Y_{jv}] = E[Y_{iv}] + E[T_{ij}]$

The expected delay and expected earliness at node j when it is served by vehicle v are then calculated as follows where the allocated vehicle v departs from the depot at time 0:

$$D_{jv}(\mathbf{x}) = \begin{cases} \alpha_{jv}\lambda_{jv}(1 - \Gamma_{\alpha_{jv}+1,\lambda_{jv}}(u'_j)) - u'_j(1 - \Gamma_{\alpha_{jv},\lambda_{jv}}(u'_j)), & \text{if } u_j > s_{jv} \\ E[Y_{jv}] + s_{jv} - u_j, & \text{otherwise} \end{cases}$$

$$E_{jv}(\mathbf{x}) = \begin{cases} l'_j \Gamma_{\alpha_{jv},\lambda_{jv}}(l'_j) - \alpha_{jv}\lambda_{jv}\Gamma_{\alpha_{jv}+1,\lambda_{jv}}(l'_j), & \text{if } l_j > s_{jv} \\ 0, & \text{otherwise} \end{cases}$$

As we already mentioned in Section 2, the total service cost of a path is calculated with respect to the optimal starting time of that path from the

12

depot. We calculate the optimal departure time of each allocated vehicle from the depot with the Golden Section Search method. The Golden Section Search technique can be applied to find the minimum (or the maximum) value of a continuous and unimodal function. We know from the above calculations of the expected delay and the expected earliness that the total service cost is a continuous function. In order to be able to use the Golden Section Search method, we need to prove that the total service cost of a path is a unimodular function of its corresponding vehicle's departure time from the depot. Since a convex function is also unimodular, we introduce the convexity property of the total service cost component in the following proposition.

Proposition 4.1. *For all routes, the total service cost is a convex function of the corresponding vehicle's departure time from the depot.*

Proof of Proposition 4.1. For a given vehicle v, the total service cost of its route is equal to $\left(c_d \sum_{j \in N} D_{jv}(\mathbf{x}) + c_e \sum_{j \in N} E_{jv}(\mathbf{x}) \right)$ where $c_d \geq 0$ and $c_e \geq 0$. For any node $j \in N$ which is visited by vehicle v, the service cost Z_{jv} is calculated by:

$$Z_{jv} = c_d D_{jv}(\mathbf{x}) + c_e E_{jv}(\mathbf{x}). \tag{15}$$

Let y denote the departure time of vehicle v from the depot. By means of the fact that the sum of convex functions is again convex, we need to show that $\frac{\partial^2 Z_{jv}}{\partial y^2} \geq 0$ to prove that the total service cost of the route of vehicle v $\sum_{j \in N} Z_{jv}$ is a convex function of y. We distinguish between three cases:

Case 1. $l'_j - y \geq 0$.

Z_{jv} is then calculated as follows:

$$
\begin{aligned}
Z_{jv} =& c_d \alpha_{jv} \lambda_{jv} (1 - \Gamma_{\alpha_{jv}+1, \lambda_{jv}}(u'_j - y)) \\
&- c_d(u'_j - y)(1 - \Gamma_{\alpha_{jv}, \lambda_{jv}}(u'_j - y)) \\
&+ c_e(l'_j - y)\Gamma_{\alpha_{jv}, \lambda_{jv}}(l'_j - y) - c_e \alpha_{jv} \lambda_{jv} \Gamma_{\alpha_{jv}+1, \lambda_{jv}}(l'_j - y).
\end{aligned}
$$

We know that,

$$\Gamma_{\alpha,\lambda}(q) = \frac{1}{\Gamma(\alpha)} \int_0^q \frac{z^{\alpha-1} e^{-z/\lambda}}{\lambda^\alpha} dz$$
$$= \frac{1}{\Gamma(\alpha)} \int_0^{\frac{q}{\lambda}} t^{\alpha-1} e^{-t} dt$$
$$= \frac{1}{\Gamma(\alpha)} \gamma(\alpha, \frac{q}{\lambda}),$$

where $\gamma(\alpha, \frac{q}{\lambda})$ represents the lower incomplete gamma function with parameters α and $\frac{q}{\lambda}$ ($\alpha \geq 0$, $q \geq 0$ and $\lambda > 0$). The first and second derivatives of this function with respect to q are given as follows:

$$\frac{\partial \gamma(\alpha, \frac{q}{\lambda})}{\partial q} = \frac{1}{\lambda} \left(\frac{q}{\lambda}\right)^{\alpha-1} e^{-\frac{q}{\lambda}} \quad \text{and,} \tag{16}$$

$$\frac{\partial^2 \gamma(\alpha, \frac{q}{\lambda})}{\partial q^2} = \frac{1}{\lambda^2} \left(\frac{q}{\lambda}\right)^{\alpha-2} e^{-\frac{q}{\lambda}} \left(\alpha - 1 - \frac{q}{\lambda}\right). \tag{17}$$

Then, $\frac{\partial^2 Z_{jv}}{\partial y^2}$ is calculated as follows:

$$\frac{\partial^2 Z_{jv}}{\partial y^2} = - c_d \alpha_{jv} \lambda_{jv} \frac{1}{\Gamma(\alpha_{jv}+1)} \frac{\partial^2 \gamma(\alpha_{jv}+1, \frac{u'_j-y}{\lambda_{jv}})}{\partial y^2}$$
$$+ c_d(u'_j - y) \frac{1}{\Gamma(\alpha_{jv})} \frac{\partial^2 \gamma(\alpha_{jv}, \frac{u'_j-y}{\lambda_{jv}})}{\partial y^2}$$
$$- 2c_d \frac{1}{\Gamma(\alpha_{jv})} \frac{\partial \gamma(\alpha_{jv}, \frac{u'_j-y}{\lambda_{jv}})}{\partial y} + c_e(l'_j - y) \frac{1}{\Gamma(\alpha_{jv})} \frac{\partial^2 \gamma(\alpha_{jv}, \frac{l'_j-y}{\lambda_{jv}})}{\partial y^2}$$
$$- 2c_e \frac{1}{\Gamma(\alpha_{jv})} \frac{\partial \gamma(\alpha_{jv}, \frac{l'_j-y}{\lambda_{jv}})}{\partial y} - c_e \alpha_{jv} \lambda_{jv} \frac{1}{\Gamma(\alpha_{jv}+1)} \frac{\partial^2 \gamma(\alpha_{jv}+1, \frac{l'_j-y}{\lambda_{jv}})}{\partial y^2}. \tag{18}$$

14

By using Equations (16) and (17), Equation (18) can be written as follows:

$$
\begin{aligned}
\frac{\partial^2 Z_{jv}}{\partial y^2} =& - c_d \alpha_{jv} \lambda_{jv} \frac{1}{\Gamma(\alpha_{jv}+1)} \frac{1}{\lambda_{jv}^2} \left(\frac{u'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-1} e^{-\left(\frac{u'_j - y}{\lambda_{jv}}\right)} \left(\alpha_{jv} - \left(\frac{u'_j - y}{\lambda_{jv}}\right)\right) \\
&+ c_d (u'_j - y) \frac{1}{\Gamma(\alpha_{jv})} \frac{1}{\lambda_{jv}^2} \left(\frac{u'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-2} e^{-\left(\frac{u'_j - y}{\lambda_{jv}}\right)} \left(\alpha_{jv} - 1 - \left(\frac{u'_j - y}{\lambda_{jv}}\right)\right) \\
&+ 2c_d \frac{1}{\Gamma(\alpha_{jv})} \frac{1}{\lambda_{jv}} \left(\frac{u'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-1} e^{-\left(\frac{u'_j - y}{\lambda_{jv}}\right)} \\
&+ c_e (l'_j - y) \frac{1}{\Gamma(\alpha_{jv})} \frac{1}{\lambda_{jv}^2} \left(\frac{l'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-2} e^{-\left(\frac{l'_j - y}{\lambda_{jv}}\right)} \left(\alpha_{jv} - 1 - \left(\frac{l'_j - y}{\lambda_{jv}}\right)\right) \\
&+ 2c_e \frac{1}{\Gamma(\alpha_{jv})} \frac{1}{\lambda_{jv}} \left(\frac{l'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-1} e^{-\left(\frac{l'_j - y}{\lambda_{jv}}\right)} \\
&- c_e \alpha_{jv} \lambda_{jv} \frac{1}{\Gamma(\alpha_{jv}+1)} \frac{1}{\lambda_{jv}^2} \left(\frac{l'_j - y}{\lambda_{jv}}\right)^{\alpha_{jv}-1} e^{-\left(\frac{l'_j - y}{\lambda_{jv}}\right)} \left(\alpha_{jv} - \left(\frac{l'_j - y}{\lambda_{jv}}\right)\right),
\end{aligned}
$$

where $\Gamma(\alpha_{jv}+1) = \alpha_{jv}\Gamma(\alpha_{jv})$. The above equation leads to:

$$
\frac{\partial^2 Z_{jv}}{\partial y^2} = c_d \frac{1}{\Gamma(\alpha_{jv})} \frac{(u'_j - y)^{\alpha_{jv}-1}}{\lambda_{jv}^{\alpha_{jv}}} e^{-\left(\frac{u'_j - y}{\lambda_{jv}}\right)} + c_e \frac{1}{\Gamma(\alpha_{jv})} \frac{(l'_j - y)^{\alpha_{jv}-1}}{\lambda_{jv}^{\alpha_{jv}}} e^{-\left(\frac{l'_j - y}{\lambda_{jv}}\right)}.
$$

So, $\frac{\partial^2 Z_{jv}}{\partial y^2} \geq 0$.

Case 2. $u'_j - y \geq 0$ and $l'_j - y \leq 0$.

Since $l_j \leq s_{jv} + y$, $E_{jv}(\mathbf{x})$ is equal to 0. Z_{jv} and its second derivative are then calculated as follows:

$$
Z_{jv} = c_d \alpha_{jv} \lambda_{jv} (1 - \Gamma_{\alpha_{jv}+1, \lambda_{jv}}(u'_j - y)) - c_d (u'_j - y)(1 - \Gamma_{\alpha_{jv}, \lambda_{jv}}(u'_j - y)) \quad \text{and,}
$$

$$
\frac{\partial^2 Z_{jv}}{\partial y^2} = c_d \frac{1}{\Gamma(\alpha_{jv})} \frac{(u'_j - y)^{\alpha_{jv}-1}}{\lambda_{jv}^{\alpha_{jv}}} e^{-\left(\frac{u'_j - y}{\lambda_{jv}}\right)}.
$$

So, $\frac{\partial^2 Z_{jv}}{\partial y^2} \geq 0$.

Case 3. $u'_j - y \leq 0$.

15

In this case, $E_{jv}(\mathbf{x})$ is again equal to 0 since $l_j \leq s_{jv} + y$. We then calculate Z_{jv} and its second derivative as follows:

$$Z_{jv} = c_d(E[Y_{jv}] + s_{jv} + y - u_j) \quad \text{and,}$$

$$\frac{\partial^2 Z_{jv}}{\partial y^2} = 0.$$

These three cases yield that for any node $j \in N$ which is visited by vehicle v, Z_{jv} is a convex function of the departure time of v from the depot. Therefore, we can conclude that for any vehicle v, the total service cost of its route is a convex function of the departure time of v from the depot. $\quad \square$

5. Branch-and-Price Method

To generate an integer solution, we embed our column generation procedure within a branch-and-price method, in which the applied strategy is branching on arcs (see Feillet et al. [13] and Tagmouti et al. [30] for the details about this method). Since we have a homogeneous fleet of vehicles of equal capacity, we calculate sum of flows (f_{ij}) on each arc (i,j) as follows:

$$f_{ij} = \sum_{v \in V} x_{ijv}. \tag{19}$$

We force arc (i,j) into the solution when $\sum_{v \in V} x_{ijv} = 1$, and we exclude arc (i,j) from the solution when $\sum_{v \in V} x_{ijv} = 0$. If we have several fractional flow variables, we choose the arc (i,j) on which the value of f_{ij} is the closest one to the midpoint (0.5). If there are several closest variables, the first one found is chosen.

To make the branching process computationally efficient, all distinctive columns obtained at the root node are stored in a separate pool. Feasible columns with respect to branching rules at a child node are then taken from that pool and used by the column generation algorithm. At each child node, we keep an extra column which serves all customers with a very high total weighted cost. By this way, it is ensured that we have an initial feasible solution at all nodes in the branch-and-price tree.

6. Numerical Results

We use Solomon's problem instances [28] for testing our exact solution approach based on column generation and branch-and-price algorithms. We

focus both on problem instances with tight time windows (type 1) and on problem instances with wide time windows (type 2). Recall that soft time windows in our problem setting, which allow early and late servicing at customers with some penalty costs, do not provide any resource constraint to the ESPPRC. This structure makes our problem similar to the capacitated VRP, and thus very sensitive to the capacity of vehicles, which is the only resource in the pricing subproblem. We first run a number of preliminary tests to determine the most appropriate value of the vehicle capacity, which enables us to obtain results in a reasonable amount of time. Two sets of preliminary tests are then conducted to determine the most appropriate values of parameters used by accelerating methods.

According to preliminary results, we set the capacity of all vehicles to 50. Moreover, we clean the ICP in case the number of columns in that pool is larger than 150. In this situation, all columns that have been kept for more than 15 iterations are removed from the ICP. We stop the ESPPRC if the number of efficient elementary paths with negative reduced costs on the ending depot is larger than 10.

To obtain our computational results, we set $\rho = 0.50$, $C_1 = 1.00$, $C_2 = 1.00$, and $(c_d, c_e, c_t, c_f, c_o)$ are equal to (1.00, 0.10, 1.00, 400, 5/6), respectively. Coefficient of Variation (CV) of travel time spent for traversing one unit distance is equal to 1.00, where $\alpha = 1.00$ and $\lambda = 1.00$. We have two stopping criteria for our solution procedure. The procedure terminates in case the gap between the best Lower Bound (LB) and the best Upper Bound (UB) is smaller than 0.005 (0.5%). In addition, we set a limit for the total CPU time which is equal to 3 hours.

We solve each problem instance by applying Depth-First (DF) and Breadth-First (BF) methods. In DF method, the UB of the root node corresponds to the initial feasible solution generated at that node, leading to a starting value for the UB. To proceed into the next level, we select the child node which has the minimum LB value. In BF method, we solve an integer programming over all columns obtained by column generation algorithm at the root node. This solution is then assigned as the starting value of the UB.

In following tables, "RootLB" and "RootUB" represent the values of the LB and the UB found at the root node of the tree, respectively. "BestLB" and "BestUB" indicate the best LB and the best UB values obtained over the tree, respectively. The percentage of the gap between the best LB and the best UB, and the size of the tree in terms of the highest level reached are also reported. We present the CPU times in seconds for only the problem

instances in RC set with 20 and 25 customers since the algorithm stops by means of the limit given for the gap before it reaches the time limit for some of the instances. For all problem instances in C and R sets, the algorithm stops due to the time limit. The results of the problem instances are not reported in case the column generation algorithm cannot obtain the optimal solution for the root node within the CPU limit. Additionally, we have no result for the problem instances if the integer programming cannot be solved at the root node within the CPU limit when we apply BF method.

Our algorithms are implemented in Visual C++, and linear programming models in our solution approach are solved by IBM ILOG CPLEX 12.2 [18]. We run all experiments on an Intel Core Duo with 2.93 GHz and 4 GB of RAM.

Table 3: Results of problem instances in RC set with 20 customers

Ins.	RootLB	Breadth-First Method						Depth-First Method					
		RootUB	BestLB	BestUB	CPU	Gap%	Tree	RootUB	BestLB	BestUB	CPU	Gap%	Tree
RC101	2184.44	2196.64	2185.71	2196.64	50.1	0.50	3	2422.11	2186.59	2196.93	725.8	0.47	12
RC102	2178.20	2191.19	2179.78	2190.65	100.7	0.50	5	2412.26	2178.61	2189.38	289.6	0.49	10
RC103	2174.45	2188.98	2175.57	2186.42	68.1	0.50	4	2223.97	2175.57	2186.42	941.1	0.50	15
RC104	2173.48	2188.90	2180.47	2186.26	619.6	0.27	8	2220.13	2175.60	2186.26	2004.7	0.49	19
RC105	2179.18	2191.56	2179.87	2190.59	71.1	0.49	4	2231.68	2180.55	2191.32	1035.9	0.49	15
RC106	2175.17	2189.84	2177.42	2187.59	78.1	0.47	4	2225.12	2178.97	2187.59	1971.5	0.40	16
RC107	2172.33	2187.41	2174.97	2185.29	135.2	0.47	5	2220.75	2174.47	2184.52	1508.4	0.46	16
RC108	2172.11	2188.83	2173.95	2184.04	115.2	0.46	5	2215.73	2174.30	2184.04	818.5	0.45	18
RC201	2216.92	2218.81	2216.92	2218.81	11.0	0.09	0	2315.64	2217.82	2219.63	43.1	0.08	4
RC202	2188.57	2199.02	2188.57	2199.02	6.4	0.48	0	2274.88	2188.57	2199.02	42.8	0.48	5
RC203	2176.18	2190.30	2177.09	2187.52	143.1	0.48	6	2264.96	2177.85	2187.52	493.9	0.44	15
RC204	2175.16	2189.77	2176.06	2186.32	153.0	0.47	6	2242.22	2176.82	2186.32	426.2	0.44	15
RC205	2193.60	2201.14	2193.60	2201.14	10.5	0.34	0	2278.38	2193.86	2202.13	44.3	0.38	6
RC206	2191.51	2200.85	2191.51	2200.85	7.6	0.43	0	2271.94	2192.26	2202.62	37.9	0.47	5
RC207	2182.12	2196.91	2183.35	2193.96	47.2	0.49	3	2236.98	2183.35	2193.96	753.2	0.49	17
RC208	2171.98	2186.15	2173.82	2183.96	161.1	0.47	5	2215.05	2174.09	2183.96	1564.3	0.45	17

Table 4: Results of problem instances in C and R sets with 20 customers

Inst.		Breadth-First Method						Depth-First Method			
	RootLB	RootUB	BestLB	BestUB	Gap%	Tree	RootUB	BestLB	BestUB	Gap%	Tree
C101	1681.27	1852.10	1696.46	1852.10	9.17	11	1925.96	1688.76	1886.41	11.70	48
C102	1665.08	1828.85	1666.06	1828.85	9.77	11	1885.19	1665.18	1843.84	10.73	63
C103	1664.75	1828.85	1666.40	1828.85	9.75	10	1870.66	1665.05	1836.54	10.30	58
C104	1664.71	1828.84	1665.74	1828.84	9.79	10	1846.33	1664.81	1832.36	10.06	67
C105	1675.17	1847.31	1682.74	1847.31	9.78	10	1902.05	1677.12	1881.33	12.18	53
C106	1679.18	1849.95	1689.69	1849.95	9.48	11	1916.44	1686.15	1878.39	11.40	45
C107	1672.46	1835.47	1676.08	1835.47	9.51	9	1882.23	1673.58	1862.11	11.26	61
C108	1666.24	1829.23	1669.32	1829.23	9.58	9	1836.62	1667.14	1836.62	10.17	53
C109	1665.51	1828.67	1667.52	1828.67	9.66	10	1832.96	1665.93	1832.96	10.03	73
C201	1781.77	1891.94	1799.48	1891.94	5.14	10	2234.47	1784.12	1922.31	7.75	41
C202	1720.95	1868.89	1727.78	1868.89	8.17	9	2151.94	1723.36	1890.14	9.68	49
C203	1701.86	1862.61	1703.45	1862.61	9.34	9	2059.85	1702.24	1885.02	10.74	61
C204	1693.88	1863.32	1694.91	1863.32	9.94	9	1865.58	1694.02	1865.58	10.13	59
C205	1753.96	1877.77	1763.46	1877.77	6.48	9	2149.62	1757.24	1966.10	11.89	51
C206	1734.51	1862.37	1741.68	1862.37	6.93	9	2086.92	1737.90	1955.14	12.50	52
C207	1717.01	1873.65	1721.28	1873.65	8.85	9	2064.22	1718.32	1884.82	9.69	58
C208	1714.84	1861.50	1717.08	1861.50	8.41	9	2029.32	1714.98	1907.96	11.25	49
R101	1319.43	1437.37	1324.36	1437.37	8.53	7	1474.19	1320.64	1474.19	11.63	65
R102	1300.74	1418.95	1305.57	1418.95	8.68	7	1459.39	1303.30	1452.40	11.44	52
R103	1292.38	1409.51	1294.59	1409.51	8.88	7	1434.82	1293.50	1434.82	10.93	55
R104	1286.63	1403.57	1291.04	1403.57	8.72	7	1429.55	1287.06	1429.55	11.07	68
R105	1303.97	1419.12	1309.47	1419.12	8.37	7	1452.84	1305.13	1452.84	11.32	56
R106	1289.52	1406.26	1295.97	1406.26	8.51	7	1437.22	1289.98	1437.22	11.41	56
R107	1287.86	1405.54	1292.52	1405.54	8.74	7	1430.65	1289.09	1408.12	9.23	53
R108	1284.77	1402.63	1288.48	1402.63	8.86	7	1428.26	1285.08	1428.26	11.14	68
R109	1291.19	1410.28	1295.96	1410.28	8.82	7	1443.29	1293.11	1443.29	11.61	54
R110	1285.22	1404.25	1288.36	1404.25	9.00	7	1440.40	1286.82	1440.40	11.93	62
R111	1286.47	1403.88	1292.00	1403.88	8.66	7	1433.58	1287.53	1425.84	10.74	55
R112	1283.48	1402.61	1285.11	1402.61	9.14	7	1427.53	1284.16	1427.53	11.16	67
R201	1367.54	1485.57	1373.25	1485.57	8.18	6	1522.29	1370.73	1522.29	11.06	71
R202	1316.23	1435.18	1318.74	1435.18	8.83	6	1470.97	1316.79	1470.97	11.71	69
R203	1304.43	1426.40	1307.88	1426.40	9.06	6	1457.82	1304.43	1457.82	11.76	62
R204	1284.61	1402.61	1287.01	1402.61	8.98	6	1425.55	1287.01	1417.64	10.15	49
R205	1334.60	1444.55	1339.86	1444.55	7.81	6	1471.69	1336.32	1471.69	10.13	70
R206	1297.19	1414.87	1301.41	1414.87	8.72	6	1445.43	1298.36	1445.43	11.33	69
R207	1293.59	1412.32	1296.14	1412.32	8.96	6	1441.51	1293.59	1441.51	11.43	74
R208	1284.40	1402.61	1288.52	1402.61	8.85	7	1425.55	1284.84	1425.55	10.95	66
R209	1289.54	1411.14	1293.54	1411.14	9.09	7	1445.48	1290.32	1445.48	12.02	65
R210	1301.79	1420.15	1305.29	1420.15	8.80	6	1451.18	1302.64	1451.18	11.40	65
R211	1283.46	1402.62	1285.26	1402.62	9.13	7	1425.57	1284.53	1425.57	10.98	73

Table 5: Results of problem instances in RC set with 25 customers

Ins.	RootLB	Breadth-First Method						Depth-First Method					
		RootUB	BestLB	BestUB	CPU	Gap%	Tree	RootUB	BestLB	BestUB	CPU	Gap%	Tree
RC101	2663.82	2688.85	2669.81	2682.70	1524.3	0.48	8	2895.13	2664.57	2687.43	10800.0	0.86	34
RC102	2656.18	2678.14	2663.84	2676.58	1399.7	0.48	8	2885.91	2656.18	2678.37	10800.0	0.84	25
RC103	2652.06	2675.63	2658.32	2670.58	1464.8	0.46	8	2903.37	2652.06	2677.50	10800.1	0.96	29
RC104	2651.93	2675.57	2658.23	2670.48	4321.4	0.46	10	2902.25	2651.93	2670.48	10800.6	0.70	31
RC105	2657.93	2684.02	2664.00	2677.30	1594.5	0.50	8	2886.28	2657.93	2681.08	10800.0	0.87	25
RC106	2651.72	2674.03	2659.46	2673.26	10800.0	0.52	12	2913.33	2651.84	2679.39	10800.3	1.04	29
RC107	2648.38	2672.19	2655.92	2669.05	2113.2	0.49	8	2908.49	2648.42	2674.13	10800.2	0.97	36
RC108	2648.18	2670.92	2654.35	2667.37	1956.2	0.49	8	2901.36	2648.18	2677.44	10800.0	1.10	37
RC201	2708.91	2716.07	2708.91	2716.07	18.5	0.26	0	3050.71	2709.69	2721.21	142.3	0.43	8
RC202	2683.89	2689.12	2683.89	2689.12	17.0	0.19	0	2994.67	2683.89	2689.62	105.7	0.21	5
RC203	2662.52	2674.81	2662.52	2674.81	16.9	0.46	0	2929.48	2662.52	2674.46	3019.5	0.45	19
RC204	2660.69	2674.31	2660.69	2673.94	126.8	0.50	4	2929.48	2660.69	2673.72	549.1	0.49	16
RC205	2686.93	2697.12	2686.93	2697.12	22.3	0.38	0	2983.00	2686.93	2699.54	379.7	0.47	18
RC206	2684.94	2696.37	2684.94	2696.37	15.5	0.43	0	2989.09	2686.64	2693.69	121.4	0.26	8
RC207	2657.66	2683.02	2664.92	2680.39	10800.0	0.58	11	2937.62	2657.66	2683.87	10800.0	0.99	28
RC208	2648.18	2669.69	2654.06	2667.19	2047.8	0.49	8	2902.54	2648.18	2677.90	10800.2	1.12	40

Table 6: Results of problem instances in C and R sets with 25 customers

Inst.	Breadth-First Method						Depth-First Method				
	RootLB	RootUB	BestLB	BestUB	Gap%	Tree	RootUB	BestLB	BestUB	Gap%	Tree
C101	2109.32	2287.61	2118.78	2287.61	7.97	9	2379.21	2110.64	2321.27	9.98	61
C102	2095.25	2259.02	2096.23	2259.02	7.77	9	2322.13	2095.47	2279.79	8.80	60
C103	2092.48	2259.02	2094.84	2259.02	7.84	7	2307.60	2092.62	2289.67	9.42	68
C104	2092.12	2259.00	2092.68	2259.00	7.95	8	2276.49	2092.23	2267.49	8.38	75
C105	2104.20	2277.47	2111.60	2277.47	7.86	9	2349.89	2107.12	2336.25	10.87	72
C106	2107.61	2284.53	2118.98	2284.53	7.81	9	2370.29	2111.96	2307.44	9.26	59
C107	2102.62	2265.64	2106.28	2265.64	7.57	9	2322.78	2103.11	2308.22	9.75	74
C108	2096.40	2259.40	2097.92	2259.40	7.70	8	2275.96	2097.30	2275.96	8.52	62
C109	2095.67	2258.83	2097.00	2258.83	7.72	8	2264.00	2096.10	2264.00	8.01	79
C201	2188.29	2345.52	2216.25	2345.52	5.83	8	2630.49	2198.92	2530.86	15.10	54
C202	2148.20	2313.69	2154.18	2313.69	7.40	8	2358.14	2151.05	2341.68	8.86	60
C203	2141.20	2306.83	2143.04	2306.83	7.64	8	2335.17	2141.20	2331.80	8.90	72
C204	2133.85	2302.21	2134.93	2302.21	7.84	8	2304.44	2134.53	2304.44	7.96	82
C205	2166.23	2325.72	2172.42	2325.72	7.06	7	2496.39	2168.52	2446.89	12.84	60
C206	2166.06	2325.61	2168.12	2325.61	7.26	7	2461.36	2166.58	2359.47	8.90	60
C207	2147.02	2313.40	2147.51	2313.40	7.72	7	2534.05	2147.02	2331.65	8.60	52
C208	2146.34	2311.06	2146.63	2311.06	7.66	7	2400.45	2146.41	2361.13	10.00	70
R101	1651.01	1701.87	1655.94	1701.87	2.77	5	1757.57	1653.74	1717.35	3.85	40
R102	1635.08	1692.01	1639.37	1692.01	3.21	6	1730.67	1637.71	1730.67	5.68	52
R103	1622.67	1681.17	1627.55	1681.17	3.29	5	1704.21	1623.44	1700.49	4.75	49
R104	1615.72	1679.70	1619.41	1679.70	3.72	5	1692.17	1618.09	1692.17	4.58	52
R105	1632.57	1689.16	1637.55	1689.16	3.15	6	1726.75	1634.68	1726.75	5.63	45
R106	1619.93	1679.19	1625.56	1679.19	3.30	6	1704.77	1620.22	1704.77	5.22	24
R107	1616.84	1669.92	1619.88	1669.92	3.09	5	1700.07	1617.57	1700.07	5.10	55
R108	1613.40	1672.33	1617.59	1672.33	3.38	5	1690.85	1614.43	1690.85	4.73	23
R109	1618.68	1680.79	1623.98	1680.79	3.50	5	1711.86	1620.22	1710.18	5.55	51
R110	1612.59	1671.69	1616.29	1671.69	3.43	6	1709.76	1613.74	1709.76	5.95	67
R111	1615.66	1673.12	1619.30	1673.12	3.32	5	1697.77	1617.13	1697.77	4.99	54
R112	1610.63	1667.80	1612.15	1667.80	3.45	5	1691.30	1612.14	1675.71	3.94	51
R201	1707.42	1752.16	1710.42	1752.16	2.44	4	1859.22	1709.06	1859.22	8.79	11
R202	1659.86	1710.55	1662.99	1710.55	2.86	4	1784.37	1661.25	1784.37	7.41	12
R203	1645.89	1688.08	1646.99	1688.08	2.49	4	1767.73	1646.23	1767.73	7.38	9
R204	1616.52	1668.16	1616.52	1668.16	3.19	4	1690.64	1616.74	1690.64	4.57	23
R205	1664.37	1723.91	1667.73	1723.91	3.37	5	1787.39	1665.56	1780.33	6.89	66
R206	1635.33	1682.97	1638.80	1682.97	2.70	4	1737.39	1635.33	1737.39	6.24	16
R207	1627.99	1674.31	1630.43	1674.31	2.69	5	1740.86	1629.74	1740.86	6.82	31
R208	1615.72	1669.98	1616.40	1669.98	3.31	5	1689.94	1615.72	1689.94	4.59	10
R209	1621.93	1679.28	1626.96	1679.28	3.22	5	1724.92	1624.22	1724.92	6.20	34
R210	1644.59	1692.09	1646.87	1692.09	2.75	5	1746.88	1645.32	1746.88	6.17	23
R211	1610.58	1670.37	1612.17	1670.37	3.61	5	1690.22	1611.30	1690.22	4.90	36

The results given in Tables 3, 4, 5, 6, 7 and 8 indicate that our solution approach with BF method provides better results than those obtained by DF method in terms of the gap between the best LB and the best UB. However, solutions of six instances with 50 customers and one instance with 100 customers, which can be provided by DF method, cannot be obtained by applying BF method due to the huge amount of CPU time or memory that it requires to solve the integer programming at the root node. Since we do not

22

Table 7: Results of problem instances in C, R and RC sets with 50 customers

Inst.	RootLB		Breadth-First Method					Depth-First Method			
		RootUB	BestLB	BestUB	Gap%	Tree	RootUB	BestLB	BestUB	Gap%	Tree
C101	3981.25	4162.32	3986.67	4162.32	4.41	4	4455.02	3988.35	4455.02	11.70	17
C102	3959.11	4130.22	3959.14	4130.22	4.32	4	4330.52	3959.28	4330.52	9.38	10
C103	3952.71	-	-	-	-	-	4184.55	3952.71	4184.55	5.87	7
C104	3946.34	-	-	-	-	-	4158.11	3946.55	4158.11	5.36	5
C105	3964.07	4140.22	3967.76	4140.22	4.35	5	4398.39	3965.38	4398.39	10.92	22
C106	3968.83	4150.76	3968.83	4150.76	4.58	0	4454.17	3970.02	4454.17	12.20	16
C107	3961.48	4138.15	3961.60	4138.15	4.46	4	4341.27	3961.60	4341.27	9.58	11
C108	3951.19	4125.79	3952.27	4125.79	4.39	3	4292.48	3952.08	4292.48	8.61	12
C109	3948.95	-	-	-	-	-	4266.58	3949.60	4266.58	8.03	4
C201	4066.61	4243.63	4068.51	4243.63	4.30	2	5646.81	4068.90	5646.81	38.78	7
C202	4042.69	4216.51	4042.69	4216.51	4.30	0	4926.45	4043.22	4926.45	21.84	4
C203	4033.78	4199.93	4034.06	4199.93	4.11	2	4450.83	4034.06	4450.83	10.33	1
C204	4020.06	4189.15	4020.20	4189.15	4.20	3	4205.53	4020.25	4205.53	4.61	4
C205	4051.12	4224.86	4051.58	4224.86	4.28	1	5418.65	4051.58	5418.65	33.74	1
C206	4047.85	4218.18	4047.85	4218.18	4.21	0	5147.35	4047.85	5147.35	27.16	1
C207	4044.95	4218.08	4044.95	4218.08	4.28	1	4966.97	4044.95	4966.97	22.79	2
C208	4043.60	4215.95	4043.60	4215.95	4.26	0	4971.79	4043.60	4971.79	22.95	2
R101	3537.25	3631.72	3538.19	3631.72	2.64	2	3684.90	3538.19	3684.90	4.15	5
R102	3503.53	3598.90	3503.53	3598.90	2.72	0	3645.16	3503.88	3645.16	4.03	2
R103	3483.89	3582.74	3483.89	3582.74	2.84	0	3621.07	3484.99	3621.07	3.90	4
R104	3457.50	3554.97	3457.50	3554.97	2.82	0	3634.73	3457.66	3634.73	5.12	2
R105	3504.74	3602.90	3506.25	3602.90	2.76	2	3681.83	3506.25	3681.83	5.01	4
R106	3482.40	3582.84	3482.40	3582.84	2.88	0	3633.09	3484.02	3633.09	4.28	3
R107	3468.94	-	-	-	-	-	3630.53	3470.19	3630.53	4.62	2
R108	3454.54	3552.22	3454.54	3552.22	2.83	0	3606.13	3454.63	3606.13	4.39	1
R109	3470.55	3570.60	3470.55	3570.60	2.88	0	3617.90	3471.18	3617.90	4.23	4
R110	3459.96	3557.78	3460.44	3557.78	2.81	1	3582.07	3460.94	3582.07	3.50	2
R111	3460.12	3558.61	3460.12	3558.61	2.85	0	3620.53	3460.74	3620.53	4.62	2
R112	3452.71	3549.79	3452.71	3549.79	2.81	1	3568.58	3452.74	3568.58	3.36	2
R201	3603.79	3677.91	3603.79	3677.91	2.06	1	3929.10	3603.79	3929.10	9.03	1
R202	3531.58	3621.23	3531.58	3621.23	2.54	0	3815.31	3531.58	3815.31	8.03	1
R204	3453.92	3561.69	3453.92	3561.69	3.12	0	3597.85	3453.92	3597.85	4.17	1
R205	3540.22	3624.99	3540.22	3624.99	2.39	0	3768.11	3540.22	3768.11	6.44	1
R206	3498.43	3597.08	3498.43	3597.08	2.82	0	3710.59	3498.43	3710.59	6.06	1
R207	3470.66	3577.27	3470.66	3577.27	3.07	0	3661.64	3470.70	3661.64	5.50	1
R208	3453.88	3553.29	3453.88	3553.29	2.88	0	3586.21	3453.88	3586.21	3.83	1
R209	3495.34	-	-	-	-	-	3632.68	3495.34	3632.68	3.93	2
R210	3489.15	-	-	-	-	-	3690.16	3489.15	3690.16	5.76	1
R211	3452.69	3548.81	3452.69	3548.81	2.78	0	3574.57	3452.69	3574.57	3.53	1
RC101	4747.43	4878.24	4749.49	4878.24	2.71	6	4952.00	4748.08	4952.00	4.29	112
RC102	4734.82	4868.81	4735.32	4868.81	2.82	6	4936.10	4734.86	4936.10	4.25	57
RC103	4727.63	4858.33	4728.60	4858.33	2.74	5	4911.10	4727.67	4911.10	3.88	39
RC104	4724.19	4855.98	4725.11	4855.98	2.77	5	4941.93	4724.19	4941.93	4.61	26
RC105	4735.41	4867.22	4736.33	4867.22	2.76	5	4956.55	4736.06	4956.55	4.66	47
RC106	4727.51	4858.99	4728.59	4858.99	2.76	5	4954.15	4728.16	4954.15	4.78	32
RC107	4721.59	4853.52	4723.14	4853.52	2.76	5	4948.95	4723.13	4948.95	4.78	21
RC108	4720.13	4852.08	4721.21	4852.08	2.77	5	4908.66	4721.06	4908.66	3.97	19
RC201	4842.82	4964.44	4846.72	4964.44	2.43	4	5271.42	4843.23	5271.42	8.84	16
RC202	4763.64	4894.51	4763.92	4894.51	2.74	4	5071.99	4763.64	5071.99	6.47	16
RC203	4737.49	4870.78	4738.49	4870.78	2.79	5	4951.02	4737.49	4951.02	4.51	24
RC204	4724.75	4857.19	4725.37	4857.19	2.79	5	4965.04	4725.16	4965.04	5.08	22
RC205	4788.26	4913.72	4789.21	4913.72	2.60	4	5094.98	4789.30	5094.98	6.38	13
RC206	4770.63	4893.91	4771.75	4893.91	2.56	5	5099.22	4771.34	5099.22	6.87	24
RC207	4741.09	4867.96	4742.00	4867.96	2.66	5	5017.94	4741.09	5017.94	5.84	17
RC208	4720.13	4852.08	4721.21	4852.08	2.77	5	4935.06	4721.06	4935.06	4.53	24

Table 8: Results of problem instances in C sets with 100 customers

Inst.	RootLB	Breadth-First Method					Depth-First Method				
		RootUB	BestLB	BestUB	Gap%	Tree	RootUB	BestLB	BestUB	Gap%	Tree
C101	8549.79	8747.36	8549.79	8747.36	2.31	0	9403.58	8549.79	9403.58	9.99	0
C105	8512.86	-	-	-	-	-	9263.38	8512.86	9263.38	8.82	1

have such an obstacle in DF method, we solve the problem instances with 100 customers where the limit for the total CPU time is set to 8 hours. The related results that show how far we can go with regard to the level of the branch-and-price tree are provided in Table 9. In these results, the highest level reached in the tree is relatively small due to the size of the problem instances. The effect of this situation is seen in the UB which cannot be improved within the time limit.

Table 9: Results of problem instances in C, R and RC sets with 100 customers obtained by DF method with 8 hour CPU limit

Inst.	RootLB	RootUB	BestLB	BestUB	Gap%	Tree
C101	8549.79	9403.58	8550.86	9403.58	9.97	7
C102	8494.70	9104.29	8494.70	9104.29	7.18	2
C103	8472.20	8878.68	8472.20	8878.68	4.80	2
C105	8512.86	9263.38	8514.10	9263.38	8.80	8
C106	8498.13	9217.98	8501.87	9217.98	8.42	2
C107	8497.82	9140.80	8499.49	9140.80	7.55	2
C108	8481.48	9036.47	8481.48	9036.47	6.54	1
C109	8472.73	8883.03	8472.73	8883.03	4.84	1
C201	8559.23	10480.70	8559.23	10480.70	22.45	1
R109	6830.88	7104.53	6830.88	7104.53	4.01	0
RC101	8380.00	8790.81	8380.00	8790.81	4.90	0
RC102	8346.21	8545.24	8346.21	8545.24	2.38	0
RC103	8325.13	8503.30	8325.13	8503.30	2.14	0
RC106	8329.07	8536.55	8329.07	8536.55	2.49	0
RC107	8311.85	8491.95	8311.85	8491.95	2.17	0

The average gap values between the best LB and the best UB for the sets with 20, 25, 50 and 100 customers provided by BF and DF methods, where the CPU limit is equal to 3 hours, are presented in Table 10. Table 11 shows the results of the sets with 100 customers where the applied strategy is DF

method and the CPU limit is equal to 8 hours.

Table 10: Average results of problem instances in C, R and RC sets with 20, 25, 50 and 100 customers obtained by BF and DF methods with 3 hour CPU limit

Set	Method	Ave. Gap%	Set	Method	Ave. Gap%
C1-20	BF	9.61	C1-20	DF	10.87
C2-20	BF	7.91	C2-20	DF	10.45
R1-20	BF	8.74	R1-20	DF	11.14
R2-20	BF	8.77	R2-20	DF	11.18
RC1-20	BF	0.46	RC1-20	DF	0.47
RC2-20	BF	0.40	RC2-20	DF	0.40
C1-25	BF	7.80	C1-25	DF	9.22
C2-25	BF	7.30	C2-25	DF	10.15
R1-25	BF	3.30	R1-25	DF	5.00
R2-25	BF	2.97	R2-25	DF	6.36
RC1-25	BF	0.49	RC1-25	DF	0.92
RC2-25	BF	0.41	RC2-25	DF	0.55
C1-50	BF	4.42	C1-50	DF	9.07
C2-50	BF	4.24	C2-50	DF	22.78
R1-50	BF	2.80	R1-50	DF	4.27
R2-50	BF	2.71	R2-50	DF	5.63
RC1-50	BF	2.76	RC1-50	DF	4.40
RC2-50	BF	2.67	RC2-50	DF	6.07
C1-100	BF	2.31	C1-100	DF	9.40

Table 11: Average results of problem instances in C, R and RC sets with 100 customers obtained by DF method with 8 hour CPU limit

Set	Method	Ave. Gap%	Set	Method	Ave. Gap%
C1-100	DF	7.26	C2-100	DF	22.45
R1-100	DF	4.01	RC1-100	DF	2.82

7. Conclusion

In this paper, we consider a vehicle routing problem with soft time windows and stochastic travel times. For this problem, we propose an exact solution approach based on column generation algorithm and branch-and-price method. To solve the pricing subproblem in column generation procedure, we extend an existing elementary shortest path algorithm with resource constraints by introducing a new dominance relation and by applying a state space augmentation technique. Moreover, two separate methods are implemented for searching in the branch-and-price tree. Our numerical study is performed on well-known problem instances. The results indicate that our solution approach can effectively be used to solve the model, in which the aim is to construct both reliable and efficient routes, for medium- and large-sized problem instances. Even though our pricing subproblem is really complex due to the stochasticity, we have an effective column generation algorithm. Finally, future research will focus on time-dependent and stochastic formulations, that we have not studied in this paper.

References

[1] N. Ando, E. Taniguchi, Travel time reliability in vehicle routing and scheduling with time windows, Networks and Spatial Economics 6 (2006) 293–311.

[2] J. Beasley, N. Christofides, An algorithm for the resource constrained shortest path problem, Networks 19 (1989) 379–394.

[3] N. Boland, J. Dethridge, I. Dumitrescu, Accelerated label setting algorithms for the elementary resource constrained shortest path problem, Operations Research Letters 34 (2006) 58–68.

[4] O. Bräysy, M. Gendreau, Vehicle routing problem with time windows, part I: route construction and local search algorithms, Transportation Science 39 (2005) 104–118.

[5] O. Bräysy, M. Gendreau, Vehicle routing problem with time windows, part II: metaheuristics, Transportation Science 39 (2005) 119–139.

[6] A. Chabrier, Vehicle routing problem with elementary shortest path based column generation, Computers & Operations Research 33 (2006) 2972–2990.

[7] C.H. Christiansen, J. Lysgaard, A branch-and-price algorithm for the capacitated vehicle routing problem with stochastic demands, Operations Research Letters 35 (2007) 773–781.

[8] N. Dellaert, A. de Kok, W. Wei, Push and pull strategies in multi-stage assembly systems, Statistica Neerlandica 54 (2000) 175–189.

[9] G. Desaulniers, J. Desrosiers, M. Solomon, Column generation, Springer, New York, 2005.

[10] M. Desrochers, An algorithm for the shortest path problem with resource constraints, Technical Report G-88-27, GERAD, 1988.

[11] M. Desrochers, J. Desrosiers, M. Solomon, A new optimization algorithm for the vehicle routing problem with time windows, Operations Research 40 (1992) 342–354.

[12] J. Desrosiers, Y. Dumas, M.M. Solomon, F. Soumis, Time constrained routing and scheduling, in: M. Ball, T. Magnanti, C. Monma, G. Nemhauser (Eds.), Network Routing, Handbooks in Operations Research and Management Science, Volume 8, North-Holland, Amsterdam, 1995, pp. 35–139.

[13] D. Feillet, P. Dejax, M. Gendreau, C. Gueguen, An exact algorithm for the elementary shortest path problem with resource constraints: application to some vehicle routing problems, Networks 44 (2004) 216–229.

[14] D. Feillet, M. Gendreau, L.M. Rousseau, New refinements for the solution of vehicle routing problems with branch and price, INFOR 45 (2007) 239–256.

[15] M. Fischetti, P. Toth, D. Vigo, A branch-and-bound algorithm for the capacitated vehicle routing problem on directed graphs, Operations Research 42 (1994) 846–859.

[16] M. Gendreau, G. Laporte, R. Séguin, Stochastic vehicle routing, European Journal of Operational Research 88 (1996) 3–12.

[17] M. Gendreau, C. Tarantilis, Solving large-scale vehicle routing problems with time windows: the state of art, Technical Report 2010-04, CIRRELT, 2010.

[18] IBM, ILOG CPLEX Optimizer 12.2, `http://www-01.ibm.com/software/integration/optimization/cplex-optimizer`, 2012.

[19] S. Irnich, D. Villeneuve, The shortest-path problem with resource constraints and k-cycle elimination for $k \geq 3$, INFORMS Journal on Computing 18 (2006) 391–406.

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

[21] G. Laporte, The vehicle routing problem: an overview of exact and approximate algorithms, European Journal of Operational Research 59 (1992) 345–358.

[22] G. Laporte, What you should know about the vehicle routing problem, Navals Research Logistics 54 (2007) 811–819.

[23] X. Li, P. Tian, S. Leung, Vehicle routing problems with time windows and stochastic travel and service times: models and algorithm, International Journal of Production Economics 125 (2010) 137–145.

[24] M. Lübbecke, J. Desrosiers, Selected topics in column generation, Operations Research 53 (2002) 1007–1023.

[25] G. Righini, M. Salani, New dynamic programming algorithms for the resouce constrained elementary shortest path problem, Networks 51 (2008) 155–170.

[26] R. Russell, T. Urban, Vehicle routing with soft time windows and erlang travel times, Journal of the Operational Research Society 59 (2008) 1220–1228.

[27] M. Savelsbergh, M. Sol, Drive: dynamic routing of independent vehicles, Operations Research 46 (1998) 474–490.

[28] M.M. Solomon, Algorithms for the vehicle routing and scheduling problems with time window constraints, Operations Research 35 (1987) 254–265.

[29] D. Taş, N. Dellaert, T. van Woensel, T. de Kok, Vehicle routing problem with stochastic travel times including soft time windows and service costs, Computers & Operations Research 40 (2013) 214–224.

[30] M. Tagmouti, M. Gendreau, J.Y. Potvin, Arc routing problems with time-dependent service costs, European Journal of Operational Research 181 (2007) 30–39.

[31] P. Toth, D. Vigo, The vehicle routing problem, vol. 9, SIAM Monographs on discrete mathematics and applications, Philadelphia, 2002.

Working Papers Beta 2009 - 2012

nr.	Year	Title	Author(s)
392	2012	Vehicle Routing with Soft Time Windows and Stochastic Travel Times: A Column Generation And Branch-and-Price Solution Approach	D. Tas, M. Gendreau, N. Dellaert, T. van Woensel, A.G. de Kok
391	2012	Improve OR-Schedule to Reduce Number of Required Beds	J.T. v. Essen, J.M. Bosch, E.W. Hans, M. v. Houdenhoven, J.L. Hurink
390	2012	How does development lead time affect performance over the ramp-up lifecycle? Evidence from the consumer electronics industry	Andres Pufall, Jan C. Fransoo, Ad de Jong
389	2012	The Impact of Product Complexity on Ramp-Up Performance	Andreas Pufall, Jan C. Fransoo, Ad de Jong, Ton de Kok
388	2012	Co-location synergies: specialized versus diverse logistics concentration areas	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v. Donselaar, Jan C. Fransoo
387	2012	Proximity matters: Synergies through co-location of logistics establishments	Frank P.v.d. Heuvel, Peter W.de Langen, Karel H. v.Donselaar, Jan C. Fransoo
386	2012	Spatial concentration and location dynamics in logistics:the case of a Dutch province	Frank P. v.d.Heuvel, Peter W.de Langen, Karel H.v. Donselaar, Jan C. Fransoo
385	2012	FNet: An Index for Advanced Business Process Querying	Zhiqiang Yan, Remco Dijkman, Paul Grefen
384	2012	Defining Various Pathway Terms	W.R. Dalinghaus, P.M.E. Van Gorp
383	2012	The Service Dominant Strategy Canvas: Defining and Visualizing a Service Dominant Strategy through the Traditional Strategic Lens	Egon Lüftenegger, Paul Grefen, Caren Weisleder
382	2012	A Stochastic Variable Size Bin Packing Problem With Time Constraints	Stefano Fazi, Tom van Woensel, Jan C. Fransoo
381	2012	Coordination and Analysis of Barge Container Hinterland Networks	K. Sharypova, T. van Woensel, J.C. Fransoo
380	2012	Proximity matters: Synergies through co-location of logistics establishments	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo

364	2011	[Vehicle Routing Problem with Stochastic Travel Times Including Soft Time Windows and Service Costs](#)	Duygu Tas, Nico Dellaert, Tom van Woensel, Ton de Kok
363	2011	[A New Approximate Evaluation Method for Two-Echelon Inventory Systems with Emergency Shipments](#)	Erhun Özkan, Geert-Jan van Houtum, Yasemin Serin
362	2011	[Approximating Multi-Objective Time-Dependent Optimization Problems](#)	Said Dabia, El-Ghazali Talbi, Tom Van Woensel, Ton de Kok
361	2011	[Branch and Cut and Price for the Time Dependent Vehicle Routing Problem with Time Window](#)	Said Dabia, Stefan Röpke, Tom Van Woensel, Ton de Kok
360	2011	[Analysis of an Assemble-to-Order System with Different Review Periods](#)	A.G. Karaarslan, G.P. Kiesmüller, A.G. de Kok
359	2011	[Interval Availability Analysis of a Two-Echelon, Multi-Item System](#)	Ahmad Al Hanbali, Matthieu van der Heijden
358	2011	[Carbon-Optimal and Carbon-Neutral Supply Chains](#)	Felipe Caro, Charles J. Corbett, Tarkan Tan, Rob Zuidwijk
357	2011	[Generic Planning and Control of Automated Material Handling Systems: Practical Requirements Versus Existing Theory](#)	Sameh Haneyah, Henk Zijm, Marco Schutten, Peter Schuur
356	2011	[Last time buy decisions for products sold under warranty](#)	M. van der Heijden, B. Iskandar
355	2011	[Spatial concentration and location dynamics in logistics: the case of a Dutch provence](#)	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
354	2011	[Identification of Employment Concentration Areas](#)	Frank P. van den Heuvel, Peter W. de Langen, Karel H. van Donselaar, Jan C. Fransoo
353	2011	[BOMN 2.0 Execution Semantics Formalized as Graph Rewrite Rules: extended version](#)	Pieter van Gorp, Remco Dijkman
352	2011	[Resource pooling and cost allocation among independent service providers](#)	Frank Karsten, Marco Slikker, Geert-Jan van Houtum
351	2011	[A Framework for Business Innovation Directions](#)	E. Lüftenegger, S. Angelov, P. Grefen

319	2010	resources utilization under uncertainty	R. Seguel, R. Eshuis, P. Grefen.
318	2010	Minimal Protocol Adaptors for Interacting Services	Tom Van Woensel, Marshall L. Fisher, Jan C. Fransoo.
317	2010	Teaching Retail Operations in Business and Engineering Schools	Lydie P.M. Smets, Geert-Jan van Houtum, Fred Langerak.
316	2010	Design for Availability: Creating Value for Manufacturers and Customers	Pieter van Gorp, Rik Eshuis.
315	2010	Transforming Process Models: executable rewrite rules versus a formalized Java program	Bob Walrave, Kim E. van Oorschot, A. Georges L. Romme
314	2010	Getting trapped in the suppression of exploration: A simulation model	S. Dabia, T. van Woensel, A.G. de Kok
313	2010	A Dynamic Programming Approach to Multi-Objective Time-Dependent Capacitated Single Vehicle Routing Problems with Time Windows	
	2010		
312	2010	Tales of a So(u)rcerer: Optimal Sourcing Decisions Under Alternative Capacitated Suppliers and General Cost Structures	Osman Alp, Tarkan Tan
311	2010	In-store replenishment procedures for perishable inventory in a retail environment with handling costs and storage constraints	R.A.C.M. Broekmeulen, C.H.M. Bakx
310	2010	The state of the art of innovation-driven business models in the financial services industry	E. Lüftenegger, S. Angelov, E. van der Linden, P. Grefen
309	2010	Design of Complex Architectures Using a Three Dimension Approach: the CrossWork Case	R. Seguel, P. Grefen, R. Eshuis
308	2010	Effect of carbon emission regulations on transport mode selection in supply chains	K.M.R. Hoen, T. Tan, J.C. Fransoo, G.J. van Houtum
307	2010	Interaction between intelligent agent strategies for real-time transportation planning	Martijn Mes, Matthieu van der Heijden, Peter Schuur
306	2010	Internal Slackening Scoring Methods	Marco Slikker, Peter Borm, René van den Brink
305	2010	Vehicle Routing with Traffic Congestion and Drivers' Driving and Working Rules	A.L. Kok, E.W. Hans, J.M.J. Schutten, W.H.M. Zijm
304	2010	Practical extensions to the level of repair analysis	R.J.I. Basten, M.C. van der Heijden, J.M.J. Schutten
303	2010	Ocean Container Transport: An Underestimated and Critical Link in Global Supply Chain Performance	Jan C. Fransoo, Chung-Yee Lee
302	2010	Capacity reservation and utilization for a manufacturer with uncertain capacity and demand	Y. Boulaksil; J.C. Fransoo; T. Tan

	delivery time requirements	
279 2009	Co-Evolution of Demand and Supply under Competition	B. Vermeulen; A.G. de Kok
278 2010	Toward Meso-level Product-Market Network Indices for Strategic Product Selection and (Re)Design Guidelines over the Product Life-Cycle	B. Vermeulen, A.G. de Kok
277 2009	An Efficient Method to Construct Minimal Protocol Adaptors	R. Seguel, R. Eshuis, P. Grefen
276 2009	Coordinating Supply Chains: a Bilevel Programming Approach	Ton G. de Kok, Gabriella Muratore
275 2009	Inventory redistribution for fashion products under demand parameter update	G.P. Kiesmuller, S. Minner
274 2009	Comparing Markov chains: Combining aggregation and precedence relations applied to sets of states	A. Busic, I.M.H. Vliegen, A. Scheller-Wolf
273 2009	Separate tools or tool kits: an exploratory study of engineers' preferences	I.M.H. Vliegen, P.A.M. Kleingeld, G.J. van Houtum
272 2009	An Exact Solution Procedure for Multi-Item Two-Echelon Spare Parts Inventory Control Problem with Batch Ordering	Engin Topan, Z. Pelin Bayindir, Tarkan Tan
271 2009	Distributed Decision Making in Combined Vehicle Routing and Break Scheduling	C.M. Meyer, H. Kopfer, A.L. Kok, M. Schutten
270 2009	Dynamic Programming Algorithm for the Vehicle Routing Problem with Time Windows and EC Social Legislation	A.L. Kok, C.M. Meyer, H. Kopfer, J.M.J. Schutten
269 2009	Similarity of Business Process Models: Metics and Evaluation	Remco Dijkman, Marlon Dumas, Boudewijn van Dongen, Reina Kaarik, Jan Mendling
267 2009	Vehicle routing under time-dependent travel times: the impact of congestion avoidance	A.L. Kok, E.W. Hans, J.M.J. Schutten
266 2009	Restricted dynamic programming: a flexible framework for solving realistic VRPs	J. Gromicho; J.J. van Hoorn; A.L. Kok; J.M.J. Schutten;

Working Papers published before 2009 see: http://beta.ieis.tue.nl