Accepted in *Transportation Science*
manuscript (Please, provide the mansucript number!)

Models and Algorithms for Stochastic and Robust Vehicle Routing with Deadlines

Yossiri Adulyasak
Singapore-MIT Alliance for Research and Technology (SMART), Singapore, 138602, yossiri@smart.mit.edu

Patrick Jaillet
Department of Electrical Engineering and Computer Science, Operations Research Center, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, jaillet@mit.edu

We consider the vehicle routing problem with deadlines under travel time uncertainty in the contexts of stochastic and robust optimization. The problem is defined on a directed graph where a fleet of vehicles is required to visit a given set of nodes and deadlines are imposed at a subset of nodes. In the stochastic vehicle routing problem with deadlines (SVRP-D), the probability distribution of the travel times is assumed to be known and the problem is solved to minimize the sum of probability of deadline violations. In the robust vehicle routing problem with deadlines (RVRP-D), however, the exact probability distribution is unknown but it belongs to a certain family of distributions. The objective of the problem is to optimize a performance measure, called lateness index, which represents the risk of violating the deadlines. Although novel mathematical frameworks have been proposed to solve these problems, the size of problem that those approaches can handle is relatively small. Our focus in this paper is the computational aspects of the two solution schemes. We introduce formulations that can be applied for the problems with multiple capacitated vehicles and discuss the extensions to the cases of incorporating service times and soft time windows. Furthermore, we develop an algorithm based on a branch-and-cut framework to solve the problems. The experiments show that these approaches provide substantial improvements in computational efficiency compared to the approaches in the literature. Finally, we provide a computational comparison to evaluate the solution quality of the SVRP-D and the RVRP-D. The results show that the RVRP-D produces solutions that are very competitive to those obtained by the SVRP-D with a large number of scenarios, while much less sensitive to the distributional uncertainty.

Key words: vehicle routing, travel time uncertainty, stochastic programming, robust optimization, branch-and-cut

1. Introduction

The classical vehicle routing problem (VRP) has been a subject of countless studies in the operations research literature. In this problem, one wishes to minimize the total cost of routing a fleet of capacitated vehicles to serve a set of customers where the associated parameters, i.e., customer locations, demands and transportation costs, are assumed to be perfectly known. A number of heuristics and exact algorithms have been proposed to solve this problem over several past decades. Most notably, recent exact algorithms of Fukasawa et al. (2006) and Baldacci et al. (2008) could solve instances with more than a hundred customers to optimality.

As some parameters such as customer demands, travel times and exact customer locations can be uncertain in practice, many studies have addressed these issues and proposed mathematical frameworks for solving the vehicle routing problem under different types of uncertainties (e.g., see (Gendreau et al. 1996)). There are two major different solution schemes for dealing with uncertainty. One approach that has been widely used is stochastic programming (see Birge and Louveaux (2011)). This approach is typically applied to the case where uncertainty can be described by known distributions. On the other hand, robust optimization has been proposed to handle cases where such probability distributions are hard to justify or estimate. This issue was first addressed in Scarf et al. (1958) in the context of robust inventory optimization. An early development of this robust solution framework is to find an optimal solution that is immune to any realization of uncertainty. This solution, however, is typically very conservative (Bertsimas and Sim 2004). Recent studies in robust optimization (see Bertsimas et al. (2011)) offer solution frameworks that can incorporate some statistical information into the models in order to find a solution that is not overly conservative while maintaining a high level of robustness. Recently, Gounaris et al. (2013) discussed the robust VRP with demand uncertainty and proposed several formulations to deal with such problem. Carlsson and Delage (2013) proposed partitioning approach to determine service regions for a fleet of vehicle when the location of demand points and their distribution are not precisely known. The focus of this study is to address the uncertainty in travel times in a vehicle routing network where one seeks to determine an *a priori* solution and the ultimate goal is to satisfy the deadline requirements. The application of this problem is of interest in several applications such as express courier service, food delivery or personal scheduling. We address the problems under the stochastic programming and robust optimization frameworks. Note that the stochastic and the robust vehicle routing problems with deadlines are referred to as SVRP-D and RVRP-D, respectively. We also use the terms SVRP and RVRP to describe other variants of the stochastic and the robust vehicle problems, respectively. Table 1 provides a summary of relevant literature concerning the SVRP-D and RVRP-D.

The SVRP-D where the underlying network is the m-traveling salesman problem (m-TSP) was first introduced in Laporte et al. (1992). They considered two different problems with a deadline imposed at the destination. The first one is a model with chance constraints where routes are not permitted if their probability of total duration exceeding the deadline is higher than a threshold. The second case is a model with recourse where these illegal routes are allowed but a penalty cost must be paid. Both problems were solved so as to minimize the total vehicle dispatching and routing costs as well as the penalty cost in the latter case. As noted in Laporte et al. (1992), the chance constraint model is in fact very similar to the determistic VRP in which additional restrictions such as maximum distance or maximum duration are imposed. The authors focused on the computational aspect of the latter case using two different formulations and performed experiments on instances with up to 20 customers. Lambert et al. (1993) proposed a heuristic to solve the model with recourse to solve instances derived from a real world situation. These studies, however, consider the problem with very few number of scenarios. The solution framework to deal with problems with large number of scenarios was addressed by Verweij et al. (2003) with a sample average approximation (SAA) technique, as applied to the shortest path problem (SPP) and the traveling salesman problem (TSP) with penalty recourse. TSP instances with 22 nodes and 1000 scenarios could be solved. Kenyon and Morton (2003) extended the problem as in Laporte et al. (1992) to the case where the number of scenarios is large and propose a branch-and-cut together with the SAA approach to solve the problem. For the problem with recourse, the algorithm could solve an instance with 28 nodes, 276 arcs and 30 scenarios with an estimated gap within 1% of optimal. Kenyon and Morton also considered the case where one wishes to maximize the probability of meeting the deadline. This problem, however, is much more difficult to solve as one must incorporate the deadline violation for each scenario using a binary variable. Therefore, only instances with 9 nodes and 2 scenarios were tested for this case. Campbell and Thomas (2008, 2009) considered the routing problem with deadlines for the case of the probabilistic traveling salesman problem (PTSP) (Jaillet 1988) where customers' presences are uncertain. They discussed several models with different recourses and proposed a heuristic to solve them. A more general case when time windows are imposed were considered in some studies (e.g., Russell and Urban (2008), Li et al. (2010), Sungur et al. (2010) and Taş et al. (2013)). Since these problems are highly complicated, these studies limit themselves to the development of heuristics.

Unlike the SVRP-D, the robust VRP, even without deadlines, has received little attention in the literature. Montemanni et al. (2007) addressed the TSP with uncertain travel time as an interval. The objective is to minimize a regret function, which is defined as the difference between the routing cost of the solution and the shortest route. The authors proposed exact algorithms based on the branch-and-cut and Benders decomposition (Benders 1962) framework to solve the problem.

Lee et al. (2012) and Agra et al. (2013) considered the VRP with time windows where one seeks to find a feasible solution to any realization where the travel time uncertainty is defined using the notion of budget of uncertainty as introduced in Bertsimas and Sim (2004). Jaillet et al. (2014) proposed a mathematical framework for a routing problem with soft time windows when exact probability distributions of travel times are not known for a single uncapacitated vehicle. They also discussed the routing problem with deadlines as a special case. The objective is to optimize a performance index which represents the risk of violating the time window restrictions. They showed that the special case with one deadline where the underlying network is the SPP can be solved in polynomial time. They also provided computational results for this case and showed that the proposed performance index could also produce solutions that are generally superior to other approaches including stochastic programming solved by sampling techniques.

In this paper, we consider the routing problem with deadlines under travel time uncertainties where the underlying network is the capacitated VRP (CVRP). In the SVRP-D, the travel times along the arcs are characterized by probability distributions and the objective is to minimize the sum of probability of deadline violations. Note that a similar problem for this case is considered by Kenyon and Morton (2003). In the RVRP, the problem definition is generally similar to the paper of Jaillet et al. (2014). In this case, the exact distribution of travel time is unknown and some information such as minimum, maximum and mean values is available. The objective is to minimize a performance index, hereafter called the lateness index. The main contributions of the paper are fourfold. First, we introduce formulations with a generalized routing set for the SVRP-D and the RVRP-D with multiple capacitated vehicles. Note that for the RVRP-D, we adopt the same notion of performance index as in Jaillet et al. (2014) but we extend the scope of the problem to the case of CVRP. We further discuss the extensions of the frameworks for the cases where service times and soft time windows are incorporated. Second, we propose efficient algorithms based on a branch-and-cut framework to solve the problems. This solution framework also offers the capability to solve the SVRP-D exactly if the probability for the tail of the convolution of the travel times can be computed. Third, we perform extensive computational experiments to evaluate the proposed

Table 1 Summary of the related literature

Author(s)	Problem	Uncertainty description	Recourse function	Deadline restriction	Network	Approach
Laporte et al. (1992)	stochastic	scenarios	cost	one deadline	m-TSP	branch-and-cut
Lambert et al. (1993)	stochastic	scenarios (2)	cost	one deadline	m-TSP	heuristic
Kenyon and Morton (2003)	stochastic	scenarios (30)	cost/prob	one deadline	m-TSP	SAA/branch-and-cut
Verweij et al. (2003)	stochastic	scenarios (1000)	cost	one deadline	SPP/TSP	SAA/branch-and-cut
Campbell and Thomas (2008, 2009)	stochastic	scenarios (2)	cost	multiple deadlines	PTSP	heuristic
Montemanni et al. (2007)	robust	interval	regret	N/A	TSP	branch-and-cut/Benders
Sungur et al. (2010)	stochastic	scenarios (n/a)	cost	time windows	VRP	heuristic
Lee et al. (2012)	robust	budget of uncertainty	n/a	time windows	VRP	column generation
Agra et al. (2013)	robust	budget of uncertainty	n/a	time windows	VRP	column generation
Jaillet et al. (2014)	robust	unsatisfactory index	n/a	time windows	SPP/TSP	iterative procedure

formulations and algorithms compared to the algorithms in literature. The results indicate that the proposed algorithms also provide superior performance to the algorithms presented in Kenyon and Morton (2003) and Jaillet et al. (2014) for the SVRP-D and RVRP-D, respectively. Fourth, we provide computational comparisons of the SVRP-D and the RVRP-D on similar instances and discuss the results obtained by the two solution schemes including the results for the case of soft time windows. The paper is organized as follows. Section 2 presents the proposed routing set which is used in both the SVRP-D and RVRP-D model. Section 3 discusses the SVRP and RVRP models and their reformulation schemes. The algorithm for these problems are presented in Section 4 and the computational experiments are shown in Section 5. This is followed by a conclusion.

2. Description of the Routing Set

The problem is defined on a directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{A})$, where $\mathcal{N} = \{1, ..., n\}$ is the set of nodes and $\mathcal{A}$ is the set of arcs. Nodes 1 and n represent the origin and the destination, respectively. Let $\mathcal{N}_{\mathcal{R}} \subseteq \mathcal{N}$ be a subset of the nodes that are required to be visited (including the origin and the destination). Let also $\mathcal{N}_{\mathcal{D}} \subseteq \mathcal{N}$ be a subset of the nodes where deadlines are imposed. Note that it is not necessary that $\mathcal{N}_{\mathcal{D}} \subseteq \mathcal{N}_{\mathcal{R}}$. For a given set of nodes $\mathcal{H}$, we define $\delta^-(\mathcal{H})$ as a set of arcs $(i,j) \in \mathcal{A}$ such that $i \in \mathcal{N} \backslash \mathcal{H}, j \in \mathcal{H}$; we also define $\delta^+(\mathcal{H})$ as a set of arcs $(i,j) \in \mathcal{A}$ such that $i \in \mathcal{H}, j \in \mathcal{N} \backslash \mathcal{H}$. For simplicity, we also write a to represent the arcs in the set $\mathcal{A}$. For the case of multiple vehicles, the set of vehicles is represented by $\mathcal{K} = \{1, ..., m\}$.

We assume that the random travel times are independent random variables. (This assumption, however, is not necessary for the SVRP-D with samples since we can still generate samples for the case of dependent travel times. For the case of RVRP, the solution framework to deal with correlated random travel times is presented in Jaillet et al. (2014).) The objective of the problem is to find a set of routes that satisfies the deadline requirements. The following decision variables are used to formulate the problems. The variable s_a^i is equal to one if arc a is part of the route to node i, 0 otherwise. The variable x_a is equal to one if a vehicle traverses arc a. Finally, the variable z_i is equal to the number of vehicles visiting node i. Note that these variables are equal to the number of dispatched vehicles for nodes $i = 1$ and n. As in the classical VRP, we assume that a customer can be visited at most once and thus the variable z_i is equal to one for each visited customer node. An example of the network is shown in Figure 1. In this network, the set of nodes that must be visited is $\mathcal{N}_{\mathcal{R}} = \{1, 2, 3, n\}$ and the set of nodes with deadlines is $\mathcal{N}_{\mathcal{D}} = \{3, 5, n\}$.

We introduce a set of constraints that are used in the routing part of both the stochastic and robust vehicle routing problems. The routing set, denoted by $\mathcal{S}^{VRP}$, is defined as follows:

$$\mathcal{S}^{VRP} = \{\boldsymbol{s} \,|\, (1) - (5)\}$$

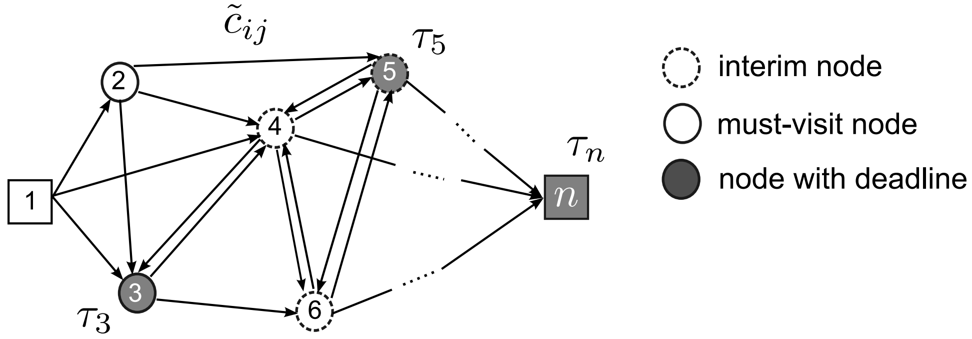

Figure 1 Network representation of the problem

where

$$\sum_{a\in\delta^+(1)} s^i_a = z_i, \qquad \forall i\in\mathcal{N}_\mathcal{D} \tag{1}$$

$$\sum_{a\in\delta^-(u)} s^i_a - \sum_{a\in\delta^+(u)} s^i_a = 0, \qquad \forall i\in\mathcal{N}_\mathcal{D}, u\in\mathcal{N}_\mathcal{D}\setminus\{1,n,i\} \tag{2}$$

$$\sum_{a\in\delta^-(i)} s^i_a - \sum_{a\in\delta^+(i)} s^i_a = z_i, \qquad \forall i\in\mathcal{N}_\mathcal{D}\cup\{n\} \tag{3}$$

$$0\le s^i_a\le x_a, \qquad \forall i\in\mathcal{N}_\mathcal{D}, \forall a\in\mathcal{A} \tag{4}$$

$$(\boldsymbol{x},\boldsymbol{z})\in\mathcal{X}^{VRP}. \tag{5}$$

These constraints control the flow of one unit of commodity associated with $i\in\mathcal{N}_\mathcal{D}$ from the origin to node i. Constraints (1)-(3) ensure flow balance at the nodes through the origin (1), intermediate nodes (2) and the destination (3) of commodity i. Constraints (4) allow a flow through arc a only if a vehicle traverses that arc. Note that it is not necessary to impose the integrality constraints on vector $\boldsymbol{s}$ as well as $\boldsymbol{z}$ since this property holds when $\boldsymbol{x}\in\{0,1\}$ as shown in Jaillet et al. (2014). This relaxation, however, did not appear to improve the computational performance in our experiments with CPLEX and thus we use the original definitions of the vectors $\boldsymbol{s}$ and $\boldsymbol{z}$ with the integrality restrictions. The set $\mathcal{X}^{VRP}$ is the set of constraints that enforce the vehicle and routing restrictions. Since a different set, i.e., multi-commodity flow based or vehicle index formulations, can also be used to describe the routing set, we use the notation $\mathcal{S}$ to refer the set of constraints (1)-(4) with a set of vehicle and routing restrictions in a general form where $\mathcal{S}^{VRP}$ is the set $\mathcal{S}$ where the vehicle and routing constraints are specifically described by our routing set $\mathcal{X}^{VRP}$ which we present next.

Denote by $\sigma(\mathcal{H})$ a minimum number of required vehicles to visit all the nodes in the set $\mathcal{H}\subseteq\mathcal{N}\setminus\{1,n\}$. The set $\mathcal{X}^{VRP}$, which controls the routes for the various vehicles is defined as follows:

$$\mathcal{X}^{VRP} = \left\{ \boldsymbol{x}\in\{0,1\}^{|\mathcal{A}|}, \boldsymbol{z}\in\mathbb{Z}_+^{|\mathcal{N}|} \,\middle|\, (6)-(11) \right\}$$

where

$$1 \le z_i \le m \qquad \forall i \in \{1,n\} \tag{6}$$

$$z_i = 1 \qquad \forall i \in \mathcal{N}_{\mathcal{R}} \setminus \{1,n\} \tag{7}$$

$$z_i \le 1 \qquad \forall i \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{R}} \tag{8}$$

$$\sum_{a \in \delta^+(i)} x_a = z_i \qquad \forall i \in \mathcal{N} \setminus \{n\} \tag{9}$$

$$\sum_{a \in \delta^-(i)} x_a = z_i \qquad \forall i \in \mathcal{N} \setminus \{1\} \tag{10}$$

$$\sum_{a \in \delta^+(\mathcal{H})} x_a \ge \sigma(\mathcal{H}) z_j \quad \forall \mathcal{H} \subseteq \mathcal{N} \setminus \{1,n\} : |\mathcal{H}| \ge 2, \forall j \in \mathcal{H}. \tag{11}$$

Constraints (6) limit the number of vehicles leaving the origin and arriving at the destination as to not exceed the number of available vehicles. Constraints (7) enforce that all the nodes in set $\mathcal{N}_{\mathcal{R}}$ must be visited, while one can choose whether or not to visit the nodes in set $\mathcal{N} \setminus \mathcal{N}_{\mathcal{R}}$. Constraints (9)-(10) are arc flow conservation. Constraints (11) are subtour elimination and vehicle capacity constraints. To impose the minimum number of customers to be served as a service level requirement, one can also add the following constraint

$$\sum_{i \in \mathcal{N}} z_i \ge \kappa. \tag{12}$$

Next, we discuss how this routing set can be used to deal with the problem with a single uncapacitated vehicle and with multiple capacitated vehicles, respectively.

2.1. Single Uncapacitated Vehicle

In the problem with a single uncapacitated vehicle, one can set $\sigma(\mathcal{H}) = 1$ in constraints (11). Therefore, constraints (11) reduce to:

$$\sum_{a \in \delta^+(\mathcal{H})} x_a \ge z_j \quad \forall \mathcal{H} \subseteq \mathcal{N} \setminus \mathcal{N}_{\mathcal{D}} \cup \{1,n\} : |\mathcal{H}| \ge 2, \forall j \in \mathcal{H}. \tag{13}$$

For that case, this formulation is not stronger than the multi-commodity flow based formulation (MCF) of Jaillet et al. (2014) since constraints (13) can be obtained by projecting out the variables $s^i_a, \forall i \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{D}}$ used in the MCF (Padberg and Sung 1991). These models, however, appear to be different in terms of computational aspect since the routing set $\mathcal{S}^{VRP}$ is typically handled by a branch-and-cut, while the formulation MCF is polynomial in size and is solved by a branch-and-bound. Roberti and Toth (2012) show that, for the asymmetric TSP, the computational performance of an efficient branch-and-cut approach is typically superior to the polynomial size models including a multi-commodity flow based formulation solved by a branch-and-bound algorithm. We perform experiments to evaluate these formulations and report the results in Section 5.2.

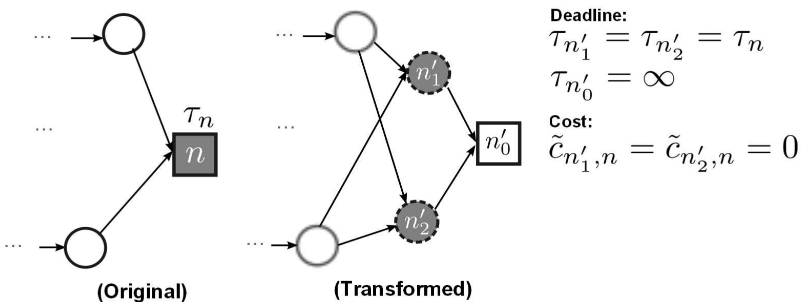

Figure 2 **Network transformation for the problem with multiple capacitated vehicles**

2.2. Multiple Capacitated Vehicles

For the problem with multiple capacitated vehicles, one could simply modify the set $\mathcal{S}^{VRP}$ by incorporating the index $k \in \mathcal{K}$ into the variables s and x, and impose vehicle capacity constraints on variables x for each vehicle separately. The size of the model would however be significantly increased. Therefore, we propose another approach to transform the original network to a new one without increasing the size of the model too much. Since the network here is modified from the original network, for notational convenience, we assume $\hat{\mathcal{N}}, \hat{\mathcal{N}_\mathcal{D}}, \hat{\mathcal{N}_\mathcal{R}}$ and $\hat{\mathcal{A}}$ are the sets $\mathcal{N}, \mathcal{N}_\mathcal{D}, \mathcal{N}_\mathcal{R}$ and $\mathcal{A}$, respectively, of the original network before the modification. The modified network can be obtained by the following steps:

1. Duplicate the destination node n into $|\mathcal{K}|$ dummy nodes, i.e., $n'_1, ..., n'_{|\mathcal{K}|}$;
2. The deadlines of the dummy nodes $\tau_{n'_k}, \forall k \in \mathcal{K}$, and parameters associated with arcs $a \in \delta^-(n'_k), \forall k \in \mathcal{K}$, are equal to those of the original node n. The original destination node n is replaced by n'_1;
3. Add a new destination node n'_0 with no deadline restriction and add arcs from each of the dummy nodes $n'_k, \forall k \in \mathcal{K}$ to this node. The travel times of these arcs are set to zero, i.e., $\tilde{c}_{n'_k,n} = 0, \forall k \in \mathcal{K}$ and set $n = n'_0$.

Figure 2 illustrates the transformation of an original network to a modified network.

We further let $\mathcal{N}' = \left\{n'_1, ..., n'_{|\mathcal{K}|}\right\}$ and $\mathcal{A}'$ be the set of all the arcs added to the network. The new sets corresponding to the transformed network are $\mathcal{N} = \hat{\mathcal{N}} \cup \mathcal{N}'$, $\mathcal{N}_\mathcal{D} = \hat{\mathcal{N}_\mathcal{D}} \backslash \{n\} \cup \mathcal{N}'$, $\mathcal{N}_\mathcal{R} = \hat{\mathcal{N}_\mathcal{R}}$and $\mathcal{A} = \hat{\mathcal{A}} \cup \mathcal{A}'$. The set $\sigma(\mathcal{H})$ must be the number of required vehicles as in the VRP (e.g., see Toth and Vigo (2001)). Denote by d_i the demand of customer i and by Q the vehicle capacity. For a given set of node $\mathcal{H}$, one can compute $\sigma(\mathcal{H}) = \lceil \sum_{i \in \mathcal{H}} d_i / Q \rceil$.

Since the dummy nodes $n'_k, \forall k \in \mathcal{K}$ are identical, one can swap the paths from the origin to any $n'_{k_1}, \forall k_1 \in \mathcal{K}$ with that of another node $n'_{k_2}, \forall k_2 \in \mathcal{K} \backslash \{k_1\}$ and still obtain the same solution. To resolve this symmetry issue, we can add symmetry breaking constraints by computing a unique number for a set of arcs that belong to each route and imposing constraints to rank them. This approach has been successfully applied to several applications (e.g., see Sherali and Smith (2001), Jans (2009), Adulyasak et al. (2013)). We use a form of the symmetry breaking constraints which can be easily obtained from the description of the travel time uncertainty. For example, if a mean value of arc (μ_a) is available, one can add the following constraints to break the symmetry:

$$\sum_{a\in\mathcal{A}} \mu_a s_a^{n'_k} \geq \sum_{a\in\mathcal{A}} \mu_a s_a^{n'_{k+1}} \quad \forall 1 \leq k \leq |\mathcal{K}|-1. \tag{14}$$

Otherwise, one can also use other parameters such as the lower bound or upper bound of the arc travel time to replace the parameter μ_a in the constraints above.

3. Stochastic and Robust Routing Formulations

In this section, we describe part of the formulations that deals with the uncertainty aspect of the problem based on the stochastic and robust optimization frameworks. In the stochastic approach, the travel time uncertainty is described by a probability distribution and the problem is solved so as to minimize the sum of probability of deadline violations. Note that this objective function is equivalent to finding the minimum expected number of deadline violations when applying the sampling approach as presented in Section 3.1. In the robust approach, however, the exact distribution is not known and one wishes to find a solution that minimizes the risk of violating the deadlines as quantified by a given performance measure, called lateness index. This objective function can be seen as a special case of the framework introduced in Lam et al. (2013).

We use tilde ($\tilde{}$) to represent uncertain quantities and denote by $\tilde{c}_a$ the uncertain travel time associated with arc a and by $\tilde{t}_i$ the arrival time at node i and hence $\tilde{t}_i = \sum_{a\in\mathcal{A}} \tilde{c}_a s_a^i$. The deadline at each node $i \in \mathcal{N}_\mathcal{D}$ is represented by τ_i.

3.1. Stochastic Routing Problem with Deadlines

In this problem, one wishes to minimize the sum of probability of deadline violations under a given set of possible realizations of the travel time uncertainty as derived from a known probability distribution $\mathbb{P}$. The minimum sum of probability of deadline violations is computed by solving the following model:

$$\min \sum_{i\in\mathcal{N}_\mathcal{D}} \mathbb{P}(\tilde{t}_i > \tau_i) \tag{15}$$

$$\text{s.t.} \quad \tilde{t}_i = \sum_{a\in\mathcal{A}} \tilde{c}_a s_a^i \quad \forall i \in \mathcal{N}_\mathcal{D} \tag{16}$$

$$(\boldsymbol{s}) \in \mathcal{S}. \tag{17}$$

Note that the objective described above also has the following property:

$$\mathbb{P}(\exists i \in \mathcal{N}_\mathcal{D} : \tilde{t}_i > \tau_i) = \mathbb{P}(\cup_{i\in\mathcal{N}_\mathcal{D}}(\tilde{t}_i > \tau_i)) \leq \sum_{i\in N_D} \mathbb{P}(\tilde{t}_i > \tau_i).$$

Recall that, for the case of multiple vehicles, $|\mathcal{K}|$ dummy nodes are created for the destination node. Then, in that case, the deadline violation is computed separately for each vehicle.

The formulation (15)-(17) is nonlinear due to the objective function (15) and constraints (16). Since the objective function is separable, however, we can exploit a decomposition technique to solve the problem. For a given solution vector $\bar{\boldsymbol{s}}$, denote by $\mathcal{A}^i_{\bar{\boldsymbol{s}}} = \{a \in \mathcal{A} \,|\, \bar{s}^i_a = 1\}$ and by $\beta^i_{\bar{\boldsymbol{s}}}$ the probability that the deadline at node i is violated with the solution $\bar{\boldsymbol{s}}$, i.e., $\beta^i_{\bar{\boldsymbol{s}}} = \mathbb{P}(t^i_{\bar{\boldsymbol{s}}} > \tau_i)$ where $t^i_{\bar{\boldsymbol{s}}} = \sum_{a\in\mathcal{A}} \tilde{c}_a \bar{s}^i_a$. The model (15)-(17) can be reformulated as:

$$\min \sum_{i\in\mathcal{N}_\mathcal{D}} \rho_i \tag{18}$$

$$\text{s.t.} \sum_{a\in\mathcal{A}^i_{\boldsymbol{p}}} \beta^i_{\boldsymbol{p}}(s^i_a - 1) + \beta^i_{\boldsymbol{p}} \le \rho_i \quad \forall i \in \mathcal{N}_\mathcal{D}, \forall (\boldsymbol{p}) \in \mathcal{S} \tag{19}$$

$$\rho_i \ge 0 \quad \forall i \in \mathcal{N}_\mathcal{D} \tag{20}$$

$$(\boldsymbol{s}) \in \mathcal{S}. \tag{21}$$

The left hand side of constraints (19) take the value $\beta^i_{\boldsymbol{p}}$ if the solution vector $\boldsymbol{s}$ is equal to or contains $\boldsymbol{p}$ as part of the solution, while it takes a negative value otherwise, i.e., let $\theta^i(\boldsymbol{s},\boldsymbol{p}) = \sum_{a\in\mathcal{A}^i_{\boldsymbol{p}}} \beta^i_{\boldsymbol{p}}(s^i_a - 1) + \beta^i_{\boldsymbol{p}}$ for a given solution vector $\boldsymbol{s}$ and $\boldsymbol{p}$, we obtain:

$$\theta^i(\boldsymbol{s},\boldsymbol{p}) = \begin{cases} \beta^i_{\boldsymbol{p}} & \text{if } \mathcal{A}^i_{\boldsymbol{p}} \subseteq \mathcal{A}^i_{\boldsymbol{s}} \\ \beta^i_{\boldsymbol{p}}\left(1 - \left|\mathcal{A}^i_{\boldsymbol{p}} \setminus \mathcal{A}^i_{\boldsymbol{s}}\right|\right) & \text{otherwise.} \end{cases}$$

One can see that the model (18)-(21) is equivalent to the original model (15)-(17) when the deadline violation $\mathbb{P}(t^i_{\bar{\boldsymbol{s}}} > \tau_i)$ is computed exactly. Instead of dealing with the nonlinear model (15)-(17) which can be very difficult to solve, the model (18)-(21) allows us to apply an iterative decomposition procedure to solve the problem by computing the values $\beta^i_{\boldsymbol{p}}, \forall i \in \mathcal{N}_\mathcal{D}$ associated with a given solution vector $\boldsymbol{p}$, and adding the cuts (19) during the solution process. This is very useful for the cases where the cumulative distribution function of $t^i_{\bar{\boldsymbol{s}}}$, i.e., the convolution of $\sum_{a\in\mathcal{A}} \tilde{c}_{a|a\in\mathcal{A}^i_{\bar{\boldsymbol{s}}}}$ can be easily derived; for example, independent random travel times with normal distribution where the derived distribution is normal with $t^i_{\bar{\boldsymbol{s}}} \sim N(\mu_{\mathcal{A}^i_{\bar{\boldsymbol{s}}}}, \sigma^2_{\mathcal{A}^i_{\bar{\boldsymbol{s}}}})$ where $\mu_{\mathcal{A}^i_{\bar{\boldsymbol{s}}}} = \sum_{a\in\mathcal{A}} \mu_{a|a\in\mathcal{A}^i_{\bar{\boldsymbol{s}}}}$ and $\sigma^2_{\mathcal{A}^i_{\bar{\boldsymbol{s}}}} = \sum_{a\in\mathcal{A}} \sigma^2_{a|a\in\mathcal{A}^i_{\bar{\boldsymbol{s}}}}$. Moreover, one can apply efficient techniques to compute the tail bound for sums of random parameters such as in Tropp (2012). Since the method to compute or approximate $\mathbb{P}(t^i_{\bar{\boldsymbol{s}}} > \tau_i)$ is distribution dependent, and our main focus is the computational aspect of the formulation when the number of possible realizations is large, we only examine this approach in a limited manner by considering the case of the normal distribution and the results are shown in Section 5.1.

To overcome the computational difficulties in computing exact distributions, a common approach in stochastic programming is to employ a sampling technique to generate a set of scenarios of the travel time vector $\tilde{\boldsymbol{c}}$ as a discrete set. Denote by $\omega \in \Omega$ the set of scenarios of the travel time vector $\tilde{\boldsymbol{c}}$. This set can be enumerated if all possible realizations of $\tilde{\boldsymbol{c}}$ is finite and not too large. In case where the number of realizations is very large, or when the distribution of the travel times are continuous, the set Ω can be generated by a Monte-Carlo sampling-based approach. Let the variable $\phi_{i\omega}$ equal to one if the travel time associated with scenario ω exceeds the deadline at node i and the parameter $M_{i\omega}$ is a sufficiently large constant associated with node i and scenario ω. The model (15)-(17) under a discrete set of scenarios can then be written as follows:

$$\min \frac{1}{|\Omega|} \sum_{i \in \mathcal{N}_{\mathcal{D}}} \sum_{\omega \in \Omega} \phi_{i\omega} \tag{22}$$

$$\text{s.t.} \quad \sum_{a \in \mathcal{A}} c_{a\omega} s_a^i \leq \tau_i + M_{i\omega}\phi_{i\omega} \quad \forall i \in \mathcal{N}_{\mathcal{D}}, \forall \omega \in \Omega \tag{23}$$

$$\phi_{i\omega} \in \{0,1\} \quad \forall i \in \mathcal{N}_{\mathcal{D}}, \forall \omega \in \Omega \tag{24}$$

$$(\boldsymbol{s}) \in \mathcal{S}. \tag{25}$$

Constraints (23) enforce the variable $\phi_{i\omega}$ to be one if the deadline at node i is violated with respect to the travel time under scenario ω. We also note that, with this sampling-based approach, the objective function also corresponds to the expected number of deadline violations, i.e., $\frac{1}{|\Omega|}\sum_{\omega\in\Omega}\left(\sum_{i\in\mathcal{N}_{\mathcal{D}}}\phi_{i\omega}\right) = \sum_{i\in\mathcal{N}_{\mathcal{D}}}\left(\frac{1}{|\Omega|}\sum_{\omega\in\Omega}\phi_{i\omega}\right)$. This problem description is similar to the one in the paper by Kenyon and Morton (2003) except that, in their paper, the deadline is only imposed at the destination, and the deadline violation of the route with the longest travel time is taken into account. The model (22)-(25) can be easily modified to solve such a problem by replacing the variable $\phi_{i\omega}$ with a single variable ϕ_ω in constraints (23), while the modification for the reformulation (22)-(25) is shown in Appendix. We further note that the model of Kenyon and Morton (2003) (also shown in Appendix) cannot handle the problem with multiple deadlines.

The challenge of solving this problem lies in the fact that the number of binary variables $\phi_{i\omega}$ and constraints (23) grow as the number of scenarios increases. Nevertheless, one can observe that, if a solution vector $\bar{\boldsymbol{s}}$ is given, the remaining problem is to obtain the values of the variables $\phi_{i\omega}$, and this can be solved by inspection. Therefore, one can compute $\beta^i_{\bar{\boldsymbol{s}}} = \frac{1}{|\Omega|}\sum_{\omega\in\Omega}\phi_{i\omega}$ and solve the problem using the reformulation (18)-(21) proposed earlier. For ease of presentation in Section 4, the step used to compute the value of the variables $\phi_{i\omega}$ for a given solution vector $(\bar{\boldsymbol{s}}) \in \mathcal{S}$ is referred to as $\mathcal{SP}^s(\bar{\boldsymbol{s}})$.

The set $\mathcal{S}$, however, is exponential in size so that it is not practical to enumerate all $(\boldsymbol{p}) \in \mathcal{S}$. To solve the model (18)-(21), one can replace constraints (19) with

$$\sum_{a \in \mathcal{A}^i_{\boldsymbol{p}}} \beta^i_{\boldsymbol{p}}(s^i_a - 1) + \beta^i_{\boldsymbol{p}} \leq \rho_i \quad \forall i \in \mathcal{N}_{\mathcal{D}}, \forall (\boldsymbol{p}) \in \mathcal{U}, \tag{26}$$

where $\mathcal{U} \subseteq \mathcal{S}$ and these constraints can be added in a branch-and-cut fashion. The model (18) and (20)-(26) with a given set $\mathcal{U} \subseteq \mathcal{S}^{VRP}$ and set $\mathcal{S} = \mathcal{S}^{VRP}$ is hereafter referred to as $\mathcal{P}^s(\mathcal{U})$. We describe the approach to solve this problem in Section 4.

3.2. Robust Routing Problem with Deadlines

In the robust optimization framework, the exact distribution $\mathbb{P}$ is not known but belongs to a certain family of distribution $\mathbb{F}$. We assume that the uncertain travel time $\tilde{c}$ in the set $\mathbb{F}$ is described by an interval and a mean value which is formally stated as:

$$\mathbb{F} = \{\mathbb{P} | \mathbb{E}_{\mathbb{P}}(\tilde{c}) = \mu, \mathbb{P}(\tilde{c} \in [\underline{c}, \overline{c}]) = 1\}.$$

To obtain a robust solution, we have built upon the general mathematical framework of Jaillet et al. (2014) which was introduced to solve robust single uncapacitated routing problems under uncertainty. Note that the details of the decomposition technique used to deal with the robust counterpart presented in this section are in-line with those of Jaillet et al. (2014) but we focus on the special case of routing problems with deadlines and devote the section here to make the paper self-explanatory. The idea is to minimize a performance index, called for this special case, a lateness index, whose definition is based on the idea of certainty equivalent (as commonly used in economic literature). Let $\mathbb{E}_{\mathbb{P}}(\tilde{x})$ denote the expected value of the random variable $\tilde{x}$ under $\mathbb{P}$. The certainty equivalent of random travel time $\tilde{t}_i$ at node i is a deterministic quantity defined as:

$$C_{\alpha_i}(\tilde{t}_i) = \alpha_i \ln \mathbb{E}_{\mathbb{P}} \left(\exp \left(\frac{\tilde{t}_i}{\alpha_i} \right) \right)$$

where $\alpha_i > 0$ is a risk tolerance parameter at node i associated with being late. Under distributional ambiguity about $\mathcal{P}$, the worst-case certainty equivalent of travel time at node i is then defined as:

$$C_{\alpha_i,\mathbb{F}}(\tilde{t}_i) = \sup_{\mathbb{P} \in \mathbb{F}} \alpha_i \ln \mathbb{E}_{\mathbb{P}} \left(\exp \left(\frac{\tilde{t}_i}{\alpha_i} \right) \right).$$

For a given deadline τ_i at node i, the quality of the random travel time $\tilde{t}_i$ for that node will then be defined as the smallest risk tolerance α_i allowable so that the certainty equivalent of travel time at node i does not exceed the deadline τ_i, i.e.,

$$\inf \left\{ \alpha_i | C_{\alpha_i,\mathbb{F}}(\tilde{t}_i) \leq \tau_i, \alpha_i \geq 0 \right\}.$$

This quality measure is a special case of the satisficing measure proposed by Brown and Sim (2009).

One can show that this measure has some nice properties with respect to satisfying a deadline requirement. In particular, it simply takes the value 0 when the travel time is guaranteed to meet the deadline. Here, we exploit this characteristic by reducing the number of variables $\boldsymbol{s}$ in one of

Jaillet et al. (2014)'s formulations. Our routing set $\mathcal{S}^{VRP}$ in Section 2 requires $s_a^i, \forall i \in \mathcal{N_D}, a \in \mathcal{A}$, since for $\forall i \in \mathcal{N} \setminus \mathcal{N_D}, \tau_i = \infty$, and thus $\alpha_i = 0, \forall i \in \mathcal{N} \setminus \mathcal{N_D}$.

More generally, in a network with multiple deadlines, let $\boldsymbol{\tau} = (\tau_i)_{i \in \mathcal{N_D}}$, $\tilde{\boldsymbol{t}} = (\tilde{t}_i)_{i \in \mathcal{N_D}}$ and $\boldsymbol{\alpha} = (\alpha_i)_{i \in \mathcal{N_D}}$. The **lateness index** for the network $\mathcal{G}$ with multiple deadlines is formally defined in this paper as:

$$\rho_{\boldsymbol{\tau}}(\tilde{\boldsymbol{t}}) = \inf \left\{ \sum_{i \in \mathcal{N_D}} \alpha_i | C_{\alpha_i, \mathbb{F}}(\tilde{t}_i) \leq \tau_i, \alpha_i \geq 0, \forall i \in \mathcal{N_D} \right\}.$$

Finding a routing policy (a set of paths from 1 to n) for which the arrival time vector $\tilde{t}$ gives the smallest lateness index is our ultimate objective. An optimal policy can thus be obtained by solving the following optimization problem:

$$\inf \sum_{i \in \mathcal{N_D}} \alpha_i \tag{27}$$

$$\text{s.t.} \quad h(\alpha_i, \boldsymbol{s}^i) \leq \tau_i \quad \forall i \in \mathcal{N_D} \tag{28}$$

$$\alpha_i \geq 0 \quad \forall i \in \mathcal{N_D} \tag{29}$$

$$(\boldsymbol{s}) \in \mathcal{S} \quad \forall i \in \mathcal{N_D} \tag{30}$$

where

$$h(\alpha_i, \boldsymbol{s}^i) = \sup_{\mathbb{P} \in \mathbb{F}} \alpha_i \ln \mathbb{E}_{\mathbb{P}} \left(\exp \left(\frac{\tilde{\boldsymbol{c}} \boldsymbol{s}^i}{\alpha_i} \right) \right)$$

Since the function $h(\alpha_i, \boldsymbol{s}^i)$ is non-linear in α_i, solving this problem is challenging. However, if the vector $\bar{\boldsymbol{s}} \in \mathcal{S}$ is known, the corresponding objective function value (27), denoted by $f^r(\bar{\boldsymbol{s}})$ can be computed as

$$f^r(\bar{\boldsymbol{s}}) = \inf \sum_{i \in \mathcal{N_D}} \alpha_i \tag{31}$$

$$\text{s.t.} \quad h(\alpha_i, \bar{\boldsymbol{s}}^i) \leq \tau_i \quad \forall i \in \mathcal{N_D} \tag{32}$$

$$\alpha_i \geq 0 \quad \forall i \in \mathcal{N_D}. \tag{33}$$

Once can observe that the problem (31)-(33) can be decomposed into $|\mathcal{N_D}|$ convex problems, each with a single variable α_i, which can be solved efficiently. For ease of presentation, the problem (31)-(33) is referred to as $\mathcal{SP}^r(\boldsymbol{s})$. To solve the original model (27)-(30), after obtaining the vector $\boldsymbol{\alpha}$, Jaillet et al. (2014) proposed a solution framework using the subgradient of the Lagrangian function

$$L(\boldsymbol{s}, \boldsymbol{\alpha}, \boldsymbol{\lambda}) = \sum_{i \in \mathcal{N_D}} \alpha_i + \sum_{i \in \mathcal{N_D}} \lambda_i \left(h(\alpha_i, \boldsymbol{s}^i) - \tau_i \right) \tag{34}$$

which is obtained by dualizing constraints (28). Note that under the assumption that travelers are not risk seeking, the worst-case expected travel time along each path to a node with deadline cannot exceed the deadline value, i.e., the following constraints must be satisfied

$$\sum_{a\in\mathcal{A}} \sup_{\mathbb{P}\in\mathbb{F}} \mathbb{E}_{\mathbb{P}}(\tilde{c}_a) s_a^i \le \tau_i \quad \forall i\in\mathcal{N}_{\mathcal{D}}. \tag{35}$$

These constraints are in fact the Slater's conditions which ensure that strong duality of the model (27)-(30) holds. Denote by $\mathcal{S}'$ the routing set $\mathcal{S}$ that also satisfies constraints (35). From strong duality, we then have, for any $(\boldsymbol{s})\in\mathcal{S}'$,

$$f^r(\boldsymbol{s}) = \sup_{\boldsymbol{\lambda}\ge 0}\left(\inf_{\alpha\ge 0} L(\boldsymbol{s},\boldsymbol{\alpha},\boldsymbol{\lambda})\right). \tag{36}$$

Jaillet et al. (2014) showed that the subgradient of the Lagrangian function is also the subgradient of $f^r(\boldsymbol{s})$, i.e., for any $(\boldsymbol{s}),(\boldsymbol{p})\in\mathcal{S}'$, we have

$$\begin{aligned} f^r(\boldsymbol{s}) - f^r(\boldsymbol{p}) &= \sup_{\boldsymbol{\lambda}\ge 0}\left(\inf_{\boldsymbol{\alpha}\ge 0} L(\boldsymbol{s},\boldsymbol{\alpha},\boldsymbol{\lambda})\right) - \sup_{\boldsymbol{\lambda}\ge 0}\left(\inf_{\boldsymbol{\alpha}\ge 0} L(\boldsymbol{p},\boldsymbol{\alpha},\boldsymbol{\lambda})\right) \\ &\ge \inf_{\boldsymbol{\alpha}\ge 0} L(\boldsymbol{s},\boldsymbol{\alpha},\boldsymbol{\lambda}^*) - \inf_{\boldsymbol{\alpha}\ge 0} L(\boldsymbol{p},\boldsymbol{\alpha},\boldsymbol{\lambda}^*) \\ &\ge d^L_{\boldsymbol{p}}(\boldsymbol{p},\boldsymbol{\alpha}^*,\boldsymbol{\lambda}^*)(\boldsymbol{s}-\boldsymbol{p}) \end{aligned}$$

where $(\boldsymbol{\alpha}^*,\boldsymbol{\lambda}^*) = \left\{(\bar{\boldsymbol{\alpha}},\bar{\boldsymbol{\lambda}})\,\middle|\, L(\boldsymbol{s},\bar{\boldsymbol{\alpha}},\bar{\boldsymbol{\lambda}}) = \sup_{\boldsymbol{\lambda}\ge 0}\left(\inf_{\boldsymbol{\alpha}\ge 0} L(\boldsymbol{s},\boldsymbol{\alpha},\boldsymbol{\lambda})\right)\right\}$ and $d^L_{\boldsymbol{p}}(\boldsymbol{p},\boldsymbol{\alpha}^*,\boldsymbol{\lambda}^*) = d^f_{\boldsymbol{p}}(\boldsymbol{p})$ is the vector of subgradient of $L(\boldsymbol{p},\boldsymbol{\alpha}^*,\boldsymbol{\lambda}^*)$ with respect to $(\boldsymbol{p})\in\mathcal{S}'$. Note that the details of the calculation of the vector $d^f_{\boldsymbol{p}}(\boldsymbol{p})$ is provided in Appendix.

Consequently, one can obtain

$$f^r(\boldsymbol{s}) \ge f^r(\boldsymbol{p}) + d^f_{\boldsymbol{p}}(\boldsymbol{p})(\boldsymbol{s}-\boldsymbol{p}), \forall(\boldsymbol{p})\in\mathcal{S}' \tag{37}$$

and the model (27)-(30) and (35) can now be reformulated as:

$$\inf\ w \tag{38}$$

$$\text{s.t.}\quad f^r(\boldsymbol{p}) + d^f_{\boldsymbol{p}}(\boldsymbol{p})(\boldsymbol{s}-\boldsymbol{p}) \le w \quad \forall(\boldsymbol{p})\in\mathcal{S}' \tag{39}$$

$$\sum_{a\in\mathcal{A}} \sup_{\mathbb{P}\in\mathbb{F}} \mathbb{E}_{\mathbb{P}}(\tilde{c}_a) s_a^i \le \tau_i \quad \forall i\in\mathcal{N}_{\mathcal{D}} \tag{40}$$

$$\alpha_i \ge 0 \quad \forall i\in\mathcal{N}_{\mathcal{D}} \tag{41}$$

$$(\boldsymbol{s})\in\mathcal{S} \quad \forall i\in\mathcal{N}_{\mathcal{D}}. \tag{42}$$

As in the SVRP-D model in the previous section, since the set $\mathcal{S}'$ is exponential in size, one can replace constraints (39) with

$$f^r(\boldsymbol{p}) + d^f_{\boldsymbol{p}}(\boldsymbol{p})(\boldsymbol{s}-\boldsymbol{p}) \le w \quad \forall(\boldsymbol{p})\in\mathcal{U}' \tag{43}$$

where $\mathcal{U}'\subseteq\mathcal{S}'$. The model (38) and (40)-(43) with a given set $\mathcal{U}$ and set $\mathcal{S}=\mathcal{S}^{VRP}$ is hereafter referred to as $\mathcal{P}^r(\mathcal{U})$. We describe the approach to solve this problem in Section 4.

3.3. Extensions of the SVRP-D and RVRP-D

3.3.1. Service Times Service times can be readily incorporated into our SVRP-D and RVRP-D since deadlines are soft constraints (i.e., customers who are visited are always served). Denote by ζ_i the service time associated with customer i. This feature can be incorporated as follows:

- *Fixed service time.* The fixed amount service time ζ_i at node i can be added to the random variables, i.e., $\tilde{c}_a + \zeta_i, \forall a \in \delta(i)^+$;
- *Random service time.* In this case, one can incorporate the service times by adding a dummy node for each node with the deadline, i.e., for a given node i, a dummy node i' is created and arc (i, i') with travel time $\tilde{\zeta}_i$ is added to the network as shown in Figure 3.

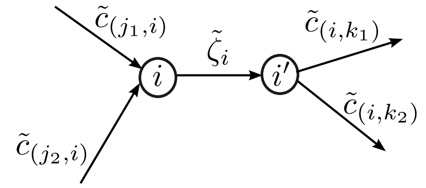

Figure 3 The modified network for the SVRP-D and RVRP-D with random service times

3.3.2. Soft Time Windows. Instead of simply a deadline, the restriction for visiting a given node i may include a time window $[\sigma_i, \tau_i]$ where $\sigma_i \le \tau_i$. In that case, if we consider that this time window is *soft*, then we can extend our frameworks by treating σ_i as the earliest arrival time for which we would not incur a violation at an early arrival, and τ_i as the latest arrival time for which we would not incur a violation of a late arrival. We also assume that vehicles are not allowed to wait during the delivery. This is typically the case for urban transportation in which parking is not available or the cost of parking can be relatively high. If one wishes to incorporate the waiting decision, it can be done by adding dummy nodes that are linked to customers and the waiting time is set as a fixed amount of travel time on the arc that links to the dummy node. We remark that the case where the waiting time is variable is in fact a model with recourse function which would change the structures of the problems substantially. Thus, the frameworks we described earlier could not be applied directly to the case, and this is left as a potential future research opportunity.

The frameworks for the SVRP-D and RVRP-D models presented earlier can be adapted to solve the SVRP and RVRP with time windows as follows.

SVRP with soft time windows. Denote by $\gamma^i_{\bar{s}} = \mathbb{P}(t^i_{\bar{s}} < \sigma_i)$ the probability that node i is visited before the start of the time window σ_i with the solution $\bar{s}$. This can be computed using a similar method as for $\beta^i_{\boldsymbol{s}}$. The SVRP formulation with soft time windows, as an extention of the model (18)-(21), can be stated as follows:

$$\min \sum_{i\in\mathcal{N}_{\mathcal{D}}} (\rho_i + \rho'_i) \tag{44}$$

$$\text{s.t.} \sum_{a\in\mathcal{A}^i_{\boldsymbol{p}}} \beta^i_{\boldsymbol{p}}(s^i_a - 1) + \beta^i_{\boldsymbol{p}} \leq \rho_i \quad \forall i \in \mathcal{N}_{\mathcal{D}}, \forall (\boldsymbol{p}) \in \mathcal{S} \tag{45}$$

$$- \sum_{a\in\mathcal{A}\setminus\mathcal{A}^i_{\boldsymbol{p}}} \gamma^i_{\boldsymbol{p}} s^i_a + \gamma^i_{\boldsymbol{p}} z_i \leq \rho'_i \quad \forall i \in \mathcal{N}_{\mathcal{D}}, \forall (\boldsymbol{p}) \in \mathcal{S} \tag{46}$$

$$\rho_i, \rho'_i \geq 0 \quad \forall i \in \mathcal{N}_{\mathcal{D}} \tag{47}$$

$$(\boldsymbol{s}) \in \mathcal{S}. \tag{48}$$

One can see that, as opposed to constraints (45), the left hand side of constraints (46) take the value $\gamma^i_{\boldsymbol{p}}$ if node i is visited and the solution vector $\boldsymbol{s}$ is a subset of or equal to $\boldsymbol{p}$ (which implies that the travel time from the origin to node i of the route in solution $\boldsymbol{s}$ is equal or shorter than one in solution $\boldsymbol{p}$), while this value takes a negative value otherwise, i.e., for a visited node i ($z_i = 1$) let $\hat{\theta}^i(\boldsymbol{s}, \boldsymbol{p}) = -\sum_{a\in\mathcal{A}\setminus\mathcal{A}^i_{\boldsymbol{p}}} \gamma^i_{\boldsymbol{p}} s^i_a + \gamma^i_{\boldsymbol{p}}$ for a given solution vector $\boldsymbol{s}$ and $\boldsymbol{p}$, we obtain:

$$\hat{\theta}^i(\boldsymbol{s}, \boldsymbol{p}) = \begin{cases} \gamma^i_{\boldsymbol{p}} & \text{if } \mathcal{A}^i_{\boldsymbol{s}} \subseteq \mathcal{A}^i_{\boldsymbol{p}} \\ \gamma^i_{\boldsymbol{p}} \left(1 - \left|\mathcal{A}^i_{\boldsymbol{s}} \setminus \mathcal{A}^i_{\boldsymbol{p}}\right|\right) & \text{otherwise.} \end{cases}$$

Consequently, the sampling-based SVRP with soft time windows can be written as.

$$\min \frac{1}{|\Omega|} \sum_{\omega\in\Omega} \sum_{i\in\mathcal{N}_{\mathcal{D}}} (\phi_{i\omega} + \phi'_{i\omega}) \tag{49}$$

$$\text{s.t.} \sum_{a\in\mathcal{A}} c_{a\omega} s^i_a \leq \tau_i + M_{i\omega}\phi_{i\omega}, \qquad \forall \omega \in \Omega, \forall i \in \mathcal{N}_{\mathcal{D}} \tag{50}$$

$$\sum_{a\in\mathcal{A}} c_{a\omega} s^i_a \geq \sigma_i + M_{i\omega}\phi'_{i\omega} \qquad \forall \omega \in \Omega, \forall i \in \mathcal{N}_{\mathcal{D}} \tag{51}$$

$$\phi_{i\omega}, \phi'_{i\omega} \in \{0, 1\} \qquad \forall \omega \in \Omega, \forall i \in \mathcal{N}_{\mathcal{D}} \tag{52}$$

$$(\boldsymbol{s}) \in \mathcal{S}. \tag{53}$$

The model (49)-(53) can be reformulated into the model (44)-(48) using the reformulation scheme presented in Section 3.

RVRP with soft time windows. In the RVRP with soft time windows, an arrival earlier than the start of a time window can be penalized by the introduction of an earliness index defined in a similar fashion as the lateness index. In that case, given a risk tolerance $\eta_i \geq 0$ for being early, the worst-case certainty equivalent of random travel time $\tilde{t}_i$ can be defined by the following non-positive deterministic quantity.

$$C^e_{\eta_i,\mathbb{F}}(\tilde{t}_i) = \sup_{\mathbb{P}\in\mathbb{F}} \eta \ln \mathbb{E}_{\mathbb{P}} \left(\exp\left(-\frac{\tilde{t}_i}{\eta_i} \right) \right).$$

The earliness index is then defined as follows:

$$\rho_{\boldsymbol{\sigma}}(\tilde{t}_i) = \inf \left\{ \varphi(\eta_i) | C^e_{\eta_i,\mathbb{F}}(\tilde{t}_i) \leq -\sigma_i, \sigma_i \geq 0 \right\}$$

The RVRP with soft time windows can be stated as follows

$$\inf \sum_{i \in \mathcal{N}_{\mathcal{D}}} (\alpha_i + \eta_i) \tag{54}$$

$$\begin{aligned}
\text{s.t.} \qquad & h(\alpha_i, \boldsymbol{s}^i) \leq \tau_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (55)\\
& e(\eta_i, \boldsymbol{s}^i) \leq -\sigma_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (56)\\
& \sum_{a \in \mathcal{A}} \sup_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}}(\tilde{c}_a) s^i_a \leq \tau_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (57)\\
& \sum_{a \in \mathcal{A}} \sup_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}}(-\tilde{c}_a) s^i_a \leq -\sigma_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (58)\\
& \alpha_i, \eta_i \geq 0 && \forall i \in \mathcal{N}_{\mathcal{D}} && (59)\\
& (\boldsymbol{s}) \in \mathcal{S}. && && (60)
\end{aligned}$$

where

$$e(\eta_i, \boldsymbol{s}^i) = \sup_{\mathbb{P} \in \mathbb{F}} \eta_i \ln \mathbb{E}_{\mathbb{P}} \left(\exp \left(-\frac{\tilde{\boldsymbol{c}} \boldsymbol{s}^i}{\eta_i} \right) \right).$$

Denote by $\mathcal{S}''$ the solution vector $\mathcal{S}$ satisfying constraints (57) and (58). Using the reformulation scheme as presented in Jaillet et al. (2014), we obtain

$$\inf \; w \tag{61}$$

$$\begin{aligned}
\text{s.t.} \quad & (f^r(\boldsymbol{p}) + g^r(\boldsymbol{p})) + \left(d^f_{\boldsymbol{p}}(\boldsymbol{p}) + d^g_{\boldsymbol{p}}(\boldsymbol{p}) \right) (\boldsymbol{s} - \boldsymbol{p}) \leq w && \forall (\boldsymbol{p}) \in \mathcal{S}'' && (62)\\
& \sum_{a \in \mathcal{A}} \sup_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}}(\tilde{c}_a) s^i_a \leq \tau_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (63)\\
& \sum_{a \in \mathcal{A}} \sup_{\mathbb{P} \in \mathbb{F}} \mathbb{E}_{\mathbb{P}}(-\tilde{c}_a) s^i_a \leq -\sigma_i && \forall i \in \mathcal{N}_{\mathcal{D}} && (64)\\
& \alpha_i, \eta_i \geq 0 && \forall i \in \mathcal{N}_{\mathcal{D}} && (65)\\
& (\boldsymbol{s}) \in \mathcal{S}. && && (66)
\end{aligned}$$

Since the objective function (54) is separable, for a given solution vector $\boldsymbol{p} \in \mathcal{S}''$, the value $f^r(\boldsymbol{p})$ can be obtained by solving the model (31)-(33), while the value $g^r(\boldsymbol{p})$ is obtained by solving the following model:

$$g^r(\boldsymbol{p}) = \inf \sum_{i \in \mathcal{N}_{\mathcal{D}}} \eta_i \tag{67}$$

$$\text{s.t.} \quad e(\eta_i, \boldsymbol{p}^i) \leq -\sigma_i \quad \forall i \in \mathcal{N}_{\mathcal{D}} \tag{68}$$

$$\eta_i \geq 0 \quad \forall i \in \mathcal{N}_{\mathcal{D}} \tag{69}$$

and $d^g_{\boldsymbol{p}}(\boldsymbol{p})$ is the subgradient of $g^r(\boldsymbol{p})$ with respect to $\boldsymbol{p}$. We provide the details of the subgradient computation of $g^r(\boldsymbol{p})$ in Appendix.

4. Solution Approaches

In this section, we discuss our approach to solve the reformulations of the SVRP-D and RVRP-D (i.e., $\mathcal{P}^s(\mathcal{U})$ and $\mathcal{P}^r(\mathcal{U})$, respectively). First, we describe the separation procedures to detect and add subtour inequalities (11) for the cases of a single uncapacitated vehicle and multiple capacitated vehicles, respectively. Next, we introduce a branch-and-cut based algorithm to solve these problems.

4.1. Separation of Subtour Inequalities

We denote by $\bar{x}$ and $\bar{z}$ the values of the variables x and z retrieved in the branch-and-bound tree. A directed graph $\bar{\mathcal{G}} = (\bar{\mathcal{N}}, \bar{\mathcal{A}})$ consists of a set of nodes $\bar{\mathcal{N}}$ and a set of arcs $\bar{\mathcal{A}}$, constructed from nodes $i \in \mathcal{N}$ with $\bar{z}_i > 0$ and arcs $a \in \mathcal{A}$ with $\bar{x}_a > 0$, respectively.

4.1.1. Single uncapacitated vehicle To detect the cuts (13), at every node of the branch-and-bound tree, we solve a series of min $s-t$ cut on the graph $\bar{\mathcal{G}} = (\bar{\mathcal{N}}, \bar{\mathcal{A}})$ from the origin (node 1) to each destination $i \in \bar{\mathcal{N}} \setminus \{1, n\}$. If the value of the minimum cut is less than $\bar{z}_i$, we generate and add the cut (13) with $j = \arg\max_{i \in \mathcal{H}}\{\bar{z}_i\}$ to the model. We use the min $s-t$ cut algorithm of the Concorde library (Applegate et al. 2011).

4.1.2. Multiple capacitated vehicle For the problem with multiple capacitated vehicle, the cuts (11) are used to eliminate subtours and ensure that vehicle capacity is not violated. These cuts can be detected by calling a separation procedure of the capacitated VRP on the graph $\bar{\mathcal{G}} = (\bar{\mathcal{N}}, \bar{\mathcal{A}})$ with demand $d_i, \forall i \in \mathcal{N}_{\mathcal{D}}$. We use the CVRP separation algorithms of Lysgaard et al. (2004) in our implementation and set the maximum number of generated cuts to $|\mathcal{N}|$.

4.2. Solution Algorithm

In this section, we describe the algorithm that is applied to both the stochastic routing model $\mathcal{P}^s(\mathcal{U})$ and robust routing model $\mathcal{P}^r(\mathcal{U})$. The algorithm is developed based on a branch-and-cut framework where the SECs (11) and the cuts (26) or (43) are added during the branch-and-bound process. In this approach, since the separation procedures for the SECs can detect the cuts for a fractional solution, these procedures are called at any node of the branch-and-bound tree. The cuts (26) and (43), however, are derived from a feasible solution $(\boldsymbol{s}) \in \mathcal{S}$ that is found during the branch-and-bound process. We refer to the proposed branch-and-cut algorithm applied to the SVRP-D and RVRP-D reformulations as the RBC. The details of the RBC algorithm are shown below. Note that the terms $\mathcal{P}(\mathcal{U})$, $\mathcal{SP}(\bar{\boldsymbol{s}})$ and $f(\bar{\boldsymbol{s}})$ are used to represent their corresponding notation in both the stochastic and robust models.

Algorithm RBC

1. Set the upper bound $ub \leftarrow \infty$, lower bound $lb \leftarrow 0$, optimality gap $gap \leftarrow \infty$, the number of cuts $b \leftarrow 0$ and the set $\mathcal{U} \leftarrow \varnothing$
2. Solve the problem $\mathcal{P}(\mathcal{U})$ by a branch-and-bound and apply the following steps at each node of the branch-and-bound tree:
   (a) Set lb to the best overall lower bound of $P(\mathcal{U})$. Solve the separation algorithms as described in Section 4.1 and add the detected SECs (11) to $\mathcal{P}(\mathcal{U})$
   (b) If $gap > \epsilon$ **and** a feasible vector, denote by $\bar{\boldsymbol{s}}$ is found
      i. Solve $\mathcal{SP}(\bar{\boldsymbol{s}})$ to obtain $f(\bar{\boldsymbol{s}})$, generate cuts (26) (for the SVRP) or (43) (for the RVRP) and add to $\mathcal{P}(\mathcal{U})$
      ii. set $\mathcal{U} = \mathcal{U} \cup (\bar{\boldsymbol{s}})$, $ub \leftarrow \max\{f(\bar{\boldsymbol{s}}), ub\}$, $gap \leftarrow (ub - lb)/ub$ and $b \leftarrow b + 1$
   (c) If $gap \leq \epsilon$, stop. Otherwise, the branching continues and the step 2a is repeated at the next node being examined.

In addition to the fact that our routing set $\mathcal{S}^{VRP}$ is more general as compared to the previous models presented in the literature so far, let us highlight here the major differences between this algorithm and the approaches of Kenyon and Morton (2003) for the stochastic routing problem and of Jaillet et al. (2014) for the robust routing problem, in particular on how the inequalities (23) and (39) are handled. In the algorithm of Kenyon and Morton (2003), constraints (23), each with one binary variable, are all added upfront, which makes the problem size depend essentially on the number of scenarios. In the RBC, constraints (23) and their associated binary variables are replaced with the cuts (19) which are added iteratively when a feasible vector $\bar{\boldsymbol{s}}$ is found. For the robust routing problem, Jaillet et al. (2014) developed an iterative algorithm, called RO. The overall process is similar to the classical Benders algorithm where the $P(\mathcal{U})$ is solved from scratch and a single valid inequality (43) generated from an optimal solution is added at each iteration. Thus, it is possible that the same feasible solutions are examined again in the branch-and-bound at each iteration as long as the cut associated with each solution is not generated and added. This process can be very time consuming especially if the model $P(\mathcal{U})$ is difficult to solve. The algorithm RBC, however, compute and add the cuts (39) in a branch-and-cut fashion. The computational experiments in Section 5 show that the computing time of the algorithm can be greatly improved.

5. Computational Results

The problems we consider in this study are defined on a directed graph. We can use a dataset designed for the asymmetric capacitated vehicle routing problem (ACVRP) for the experiments. To the best of our knowledge, the ACVRP dataset that is publicly available is that of Fischetti et al. (1994) which consists of instances with 32 to 71 customers defined on a complete graph. This dataset, however, is designed for the deterministic problem and it is also too large for our experiments. Therefore, we adapted the benchmark of Jaillet et al. (2014) which was originally designed for a robust routing optimization of a single uncapacitated vehicle on an incomplete direct graph. It consists of instances with a number of nodes $|\mathcal{N}|$ varying from 10 to 80 with an increment of 10 and a number of arcs generated so that $|\mathcal{A}| = 3|\mathcal{N}|$. Travel time for each arc c_a is in an interval $c_a \in [\underline{c}_a, \bar{c}_a]$ with a mean value μ_a. To solve the SVRP-D and RVRP-D with capacitated vehicles, we generate customer demands and vehicle capacity for the instances. The details on the dataset used in this study are provided in Appendix.

We use the following notation to represent the approaches used in this section:

BC The branch-and-cut algorithm where only the SECs (13) are added in a branch-and-cut fashion. For the stochastic optimization problem, the inequalities (19) are all added upfront (similar to the approach of Kenyon and Morton (2003)). For the robust optimization problem, the inequalities (43) are added when an optimal solution of the $\mathcal{P}^r(\mathcal{U})$ is found, which is similar to the algorithm RO of Jaillet et al. (2014). This approach is used to show the performance of the routing set $\mathcal{S}^{VRP}$ proposed in this study;

RBC The branch-and-cut algorithm for the SVRP and RVRP reformulations as described in Section 4.2;

KM The approach of Kenyon and Morton (2003) for the stochastic routing problem;

JQS The approach of Jaillet et al. (2014) for the robust routing problem based on the multi-commodity based model (MCF)

The algorithm were coded in C++ and C# on MonoDevelop 3.0 under Windows 8 using CPLEX 12.5.1. The experiments were performed on a workstation with an Intel 2.67GHz processor and 4GB of RAM. Unless stated otherwise, the maximum CPU time is set to two hours.

5.1. Performance of the Algorithms on the Stochastic Routing Problem

In this section, we perform experiments to evaluate the performance of the BC, RBC and KM on the stochastic routing problem. To generate instances for the stochastic routing problem, we create a set of scenarios from the travel times given in the instances. Since the travel time parameters were described by a range and a mean value, we assume that the travel time distribution is triangular with the mean value provided in the dataset. Instances with $10 \leq |\mathcal{N}| \leq 40$ are used.

In Kenyon and Morton (2003), the approach (KM) is designed for the problem with uncapacitated vehicles and with a single deadline at the destination node. The deadline violation is only considered for the vehicle with the latest arrival time at the destination node. The objective function is in fact to minimize the probability that at least one vehicle arrives at the destination after the deadline. To solve this problem, the model (22)-(25) can easily modified by replacing the binary variable $\phi_{i\omega}$ with the binary variable ϕ_{ω}, i.e., one variable per scenario, and adding another set of constraints as shown in Appendix B. This problem is referred to as SVRP-MD. We first evaluate the performance of the algorithms on this case and the results are shown in Table 2. Note that the deadline is set at two different tightness levels and all nodes are required to be visited in this case. We perform the experiments on instances with 100 and 1000 scenarios to evaluate the KM, BC and RBC algorithms. Additionally, as we have mentioned earlier, since the proposed decomposition scheme allows us to solve the problem with a very large number of scenarios without much additional computational effort, we also attempt to solve the instances with 20,000 scenarios with the RBC algorithm. Column ψ_m shows the average number of deadlines that could be met by all vehicles under 20,000 scenarios. Column ϵ_{ψ_m} show the average gap of ψ_m when comparing with the solution obtained by using 20,000 scenarios, computed as $(\psi_m^{|\Omega|} - \psi_m^{20000})/\psi_m^{20000}$ where $\psi_m^{|\Omega|}$ is the value of ψ_m for the instances with $|\Omega|$ scenarios, and column $\epsilon_{\psi_m}^{max}$ shows the maximum value of ϵ_{ψ_m} for the instance of the same size. The average CPU time in seconds are shown in Column *CPU.* The results are shown in Table 2.

As shown in Table 2, the average computing time of the RBC algorithm is significantly lower than the BC and KM approaches, especially for the instances with 1000 scenarios. All the instances were solved to optimality by the RBC in a few seconds. The BC is also more efficient than the KM which demonstrates the efficiency of the proposed routing set $\mathcal{S}$ alone. We further note that the KM algorithm is also far more sensitive to the number of scenarios than the other two where 10 instances with 1000 scenarios could not be solved to optimality by this algorithm. In terms of solution quality, the solutions obtained by solving 1000 scenarios are generally much better than ones obtained by 100 scenarios. The average gap ϵ_{ψ_m} and the average maximum gap $\epsilon_{\psi_m}^{max}$ of the solutions obtained by using 1000 scenarios are 0.01% and 0.22%, respectively, while those of the solutions obtained by using 100 scenarios are 0.11% and 1.22%, respectively.

We next performed the experiments to evaluate the performance of the algorithms on a more general case with capacitated vehicles using the model (22)-(25). In this case, to allow some flexibility, vehicles can also leave a few nodes unserved but the total number of visited nodes cannot be less than the minimum service level (constraint (12)) and all the even-number nodes require a visit. The number of nodes that is allowed to be unserved is shown in Column σ (i.e., $\sigma = |\mathcal{N}| - \delta$). For the KM approach, we use the routing set as described in Kenyon and Morton (2003) and simply

Table 2 Average results on the SVRP-MD instances with uncapacitated vehicles and one deadline

Case	$\lvert\mathcal{V}\rvert$	$\lvert\mathcal{N}\rvert$	$\lvert\mathcal{A}\rvert$	$\lvert\Omega\rvert = 100$						$\lvert\Omega\rvert = 1000$						$\lvert\Omega\rvert = 20000$	
				ψ_m	ϵ_{ψ_m}	$\epsilon^{max}_{\psi_m}$	KM	BC	RBC	ψ_m	ϵ_{ψ_m}	$\epsilon^{max}_{\psi_m}$	KM	BC	RBC	ψ_m	RBC
							CPU	CPU	CPU				CPU	CPU	CPU		CPU
Loose	2	10	30	1.807	0.09	0.80	0.4	0.4	**0.1**	1.809	0.00	0.01	15.6	6.4	**0.1**	1.809	0.8
		20	60	1.812	0.25	1.63	0.6	0.8	**0.2**	1.815	0.06	0.93	43.8	13.5	**0.3**	1.816	2.0
		30	90	1.735	0.07	0.59	1.1	2.0	**0.5**	1.737	0.00	0.00	82.9	41.9	**0.6**	1.737	6.9
		40	120	1.764	0.12	1.87	26.3	23.7	**13.9**	1.765	0.00	0.01	$1663.1^{(3)}$	383.1	**19.2**	1.765	56.7
	Average			1.779	0.13	1.22	7.1	6.7	**3.7**	1.781	0.02	0.24	$451.3^{(3)}$	111.2	**5.0**	1.782	16.6
	3	10	30	2.839	0.07	0.52	0.6	0.5	**0.2**	2.841	0.00	0.02	21.4	10.5	**0.3**	2.841	1.7
		20	60	2.861	0.13	1.09	1.1	1.0	**0.2**	2.864	0.03	0.43	93.5	17.9	**0.4**	2.865	3.0
		30	90	2.736	0.04	0.41	3.5	2.8	**0.6**	2.737	0.00	0.01	150.9	52.0	**0.8**	2.737	4.8
		40	120	2.858	0.03	0.21	27.2	26.2	**12.1**	2.859	0.00	0.03	$814.7^{(2)}$	434.9	**19.5**	2.859	65.4
	Average			2.824	0.07	0.56	8.1	7.6	**3.3**	2.825	0.01	0.12	$270.1^{(2)}$	128.8	**5.2**	2.826	18.7
Tight	2	10	30	1.558	0.16	3.08	0.9	0.5	**0.1**	1.560	0.00	0.00	64.1	11.8	**0.1**	1.560	0.9
		20	60	1.600	0.27	4.50	1.5	1.3	**0.3**	1.604	0.02	0.19	94.6	26.9	**0.4**	1.605	2.6
		30	90	1.674	0.04	0.59	2.1	3.8	**1.8**	1.675	0.02	0.13	183.3	63.4	**2.2**	1.675	11.2
		40	120	1.405	0.04	0.32	41.5	39.8	**13.7**	1.406	0.01	0.13	$1644.7^{(2)}$	859.5	**17.5**	1.406	84.1
	Average			1.559	0.13	2.12	11.5	11.4	**4.0**	1.561	0.01	0.11	$496.7^{(2)}$	240.4	**5.1**	1.561	24.7
	3	10	30	2.639	0.12	1.83	1.1	0.8	**0.2**	2.640	0.09	0.38	71.4	18.2	**0.3**	2.641	1.8
		20	60	2.713	0.04	0.24	3.3	1.8	**0.4**	2.714	0.01	0.14	145.8	35.5	**0.7**	2.714	3.8
		30	90	2.676	0.13	0.93	5.1	3.0	**0.8**	2.679	0.01	0.23	191.9	77.5	**1.3**	2.679	7.2
		40	120	2.676	0.11	0.98	80.7	44.2	**39.8**	2.678	0.05	0.91	$2570.6^{(5)}$	721.4	**53.4**	2.679	151.3
	Average			2.676	0.10	1.00	22.6	12.4	**10.3**	2.678	0.04	0.42	$744.9^{(5)}$	213.1	**13.9**	2.679	41.0

$^{(-)}$ the number of instances (out of 20 per instance size) were not solved to optimality

add the vehicle capacity constraints on each vehicle separately (the full model for this case, denote by $\mathcal{S}^{KM}$, is shown in Appendix B). However, we still consider the case of a single deadline since the model used in the KM approach cannot handle the problem with multiple deadlines. The results are shown in Table 3. In this table, column ψ represent the expected number of deadlines satisfied by a solution under 20,000 scenarios, while columns ϵ_ψ and ϵ_ψ show the corresponding average gap and the maximum gap of ψ, respectively.

The RBC algorithm is still far superior to the other two algorithms especially when the number of scenarios is large, while the performance of the BC model is better than the KM model by a significant margin. The KM approach could not solve all the instances with 100 scenarios and the instances with 20 nodes and 1000 scenarios to optimality. The gap of the solutions obtained by solving the instances with 100 scenarios are relatively large compared to those obtained by solving 1000 scenarios. In this case, the RBC algorithm could still solve the instances with 20,000 scenarios to optimality in a few seconds.

Table 3 Average results on the SVRP-D instances with capacitated vehicles and one deadline

$\|\mathcal{V}\|$	$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	σ	$\|\Omega\|=100$						$\|\Omega\|=1000$						$\|\Omega\|=20000$	
				ψ	ϵ_ψ	ϵ_ψ^{max}	KM	BC	RBC	ψ	ϵ_ψ	ϵ_ψ^{max}	KM	BC	RBC	ψ	RBC
							CPU	CPU	CPU				CPU	CPU	CPU		CPU
2	10	30	1	1.685	0.44	3.23	1.6	0.5	**0.1**	1.690	0.00	0.00	436.6	18.8	**0.1**	1.690	0.6
	20	60	1	1.820	0.39	3.04	9.4	0.8	**0.2**	1.827	0.00	0.02	509.8$^{(1)}$	23.1	**0.3**	1.827	1.8
	30	90	2	1.851	0.12	0.65	73.5	**6.2**	7.1	1.852	0.01	0.07	1168.8$^{(2)}$	173.6	**7.8**	1.853	34.8
	40	120	2	1.940	0.26	1.08	550.8$^{(1)}$	58.0	**52.7**	1.944	0.05	0.40	1479.1$^{(4)}$	656.9	**80.4**	1.945	336.7
Average				1.824	0.30	2.00	158.8$^{(1)}$	16.3	**15.0**	1.828	0.02	0.12	898.6$^{(7)}$	218.1	**22.1**	1.829	93.5
3	10	30	1	2.771	0.14	1.34	7.1	0.7	**0.2**	2.774	0.00	0.00	400.8	14.1	**0.3**	2.774	1.3
	20	60	1	2.894	0.16	1.78	7.9	1.2	**0.3**	2.899	0.00	0.06	431.9$^{(1)}$	20.3	**0.4**	2.899	3.2
	30	90	2	2.862	0.10	0.64	156.0	6.4	**4.8**	2.865	0.00	0.03	1208.8$^{(3)}$	118.0	**5.9**	2.865	21.7
	40	120	2	2.972	0.10	1.06	363.9$^{(1)}$	34.7	**21.9**	2.974	0.01	0.05	421.7$^{(1)}$	496.8	**53.6**	2.975	208.8
Average				2.875	0.12	1.21	133.7$^{(1)}$	10.8	**6.8**	2.878	0.00	0.04	615.8$^{(5)}$	162.3	**15.0**	2.878	58.8

$^{(-)}$ the number of instances (out of 20 per instance size) were not solved to optimality

As mentioned in Section 3.1, the reformulation also provides an exact solution framework for the SPVRP. In Table 4, we show the results of the algorithm RBC applied to solve the case where travel times are independent and the distribution is normal with the mean $\hat{\mu}_a = (\underline{c}_a + \overline{c}_a)/2$ and standard deviation $\sigma_a = (\overline{c}_a - \hat{\mu}_a)/3$. The characteristics of the network are similar to the case in Table 3 except that two deadlines are imposed at nodes $[n/2]$ and n . Column *Obj. Gap* shows the average gap (%) of the objective value of the instances with 20,000 scenarios of the model (18)-(21) and the objective value of the exact solution. Column ϵ^*_ψ shows the average gap of the expected number of deadline violations of the solution with $|\Omega| = 20,000$ computed by the exact cumulative distribution compared to that of the exact solution. Column *BCuts* show the number of cuts (26) generated during the branch-and-cut algorithm. Overall, the solutions obtained by using $|\Omega| = 20,000$ are equivalent to the exact solutions since a large number of scenarios is used. The computing time to solve the problem with the exact normal distribution is generally lower since the value of $\mathbb{P}(t^i_{\bar{s}} > \tau_i)$ can be easily obtained.

5.2. Performance of the Algorithms on the Robust Routing Problem

Since the JQS approach of Jaillet et al. (2014) was designed for the problem with a single uncapacitated vehicle, we perform the experiments on this case to evaluate the performance of the BC and RBC approaches. As in Jaillet et al. (2014), we consider the problem with $|\mathcal{N}_\mathcal{R}| = |\mathcal{N}|$ with two different cases, i.e., two deadlines are imposed at nodes $[n/2]$ and n and deadlines are imposed at all nodes. For the first case, the algorithms JQS, BC and RBC are tested, while the JQS and BC approaches are identical for the latter since no SECs are required as we have explained in Section 2.1. In addition to the instances of Jaillet et al. (2014), we also test the algorithms on larger

Table 4 Average results on the SVRP-D instances with capacitated vehicles and two deadline with underlying normal distribution

$\|\mathcal{V}\|$	$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	σ	Obj. Gap	ϵ^*_ψ	$\|\Omega\| = 20000$			Normal distribution		
				(%)	(%)	CPU	Node	BCuts	CPU	Node	BCuts
2	10	30	1	0.13	0.00	0.7	193	57.1	0.1	190	62.1
	20	60	1	0.16	0.00	2.2	392	79.6	0.5	416	89.7
	30	90	2	0.00	0.00	24.7	2988	490.7	8.2	3246	492.4
	40	120	2	0.59	0.00	354.6	23809	3782.1	188.1	23524	3710.6
Average				0.22	0.00	95.5	6845	1102.4	49.2	6844	1088.7
3	10	30	1	0.13	0.00	1.3	584	67.3	0.3	542	68.1
	20	60	1	0.23	0.00	2.9	496	62.2	0.6	521	74.6
	30	90	2	0.03	0.00	25.5	2846	284.5	7.7	2814	316.3
	40	120	2	0.91	0.00	364.7	24906	2396.0	207.7	24950	2334.0
Average				0.33	0.00	98.6	7208	702.5	54.1	7207	698.2

instances with $|\mathcal{N}| = 70$ and 80 for the case with two deadlines. The results are shown in Tables 5 and 6. Column *BCuts* show the number of cuts (43) generated during the solution process. For the algorithms JQS and BC, which follow from the classical Benders algorithm, the number of cuts (43) also equals to the number of iterations since one cut is added at each iteration.

Table 5 Average results on the robust routing instances with a single uncapacitated vehicle and two deadlines

$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	JQS			BC			RBC		
		CPU	Nodes	BCuts	CPU	Nodes	BCuts	CPU	Nodes	BCuts
10	30	**0.1**	24	3.1	**0.1**	21	3.1	**0.1**	10	5.8
20	60	1.1	139	4.2	0.3	54	4.2	**0.1**	29	8.0
30	90	18.7	861	7.6	3.1	331	7.6	**0.4**	154	12.9
40	120	138.8	2776	11.2	27.2	25	9.4	**1.3**	478	23.3
50	150	$3272.5^{(4)}$	18555	13.2	$468.0^{(1)}$	12655	16.0	**33.2**	5742	48.0
60	180	$4315.9^{(10)}$	19965	10.4	$1774.6^{(3)}$	31616	22.5	**159.4**	15953	61.2
70	210	$5485.9^{(14)}$	9457	7.8	$2363.5^{(5)}$	56876	23.9	**577.4**$^{(1)}$	29730	104.4
80	240	$6764.1^{(18)}$	6291	2.5	$3582.3^{(8)}$	91773	16.9	**1060.3**$^{(1)}$	47239	148.9
Average		$2499.6^{(46)}$	7258	7.5	$1027.4^{(17)}$	24169	12.9	**229.0**$^{(2)}$	12417	51.6

$^{(-)}$ the number of instances (out of 20 per instance size) were not solved to optimality

For both cases, the performance of the algorithm RBC is far superior to the other algorithms. For the case of two deadlines, the algorithms BC and RBC also outperformed the algorithm JQS. Only two instances and 17 instances (out of 160) were not solved by the RBC and BC algorithms, respective, while the algorithm JQS could not solve 46 instances. For the case of deadlines imposed at all nodes, the RBC could not solve only four instances (out of 120) while the JQS could not

Table 6 Average results on the robust routing instances with a single uncapacitated vehicle and deadlines at all nodes

$\|\mathcal{N}\|$	$\|\mathcal{A}\|$	JQS			RBC		
		CPU	Nodes	BCuts	CPU	Nodes	BCuts
10	30	0.3	60	5.4	**0.1**	15	7.6
20	60	7.9	807	11.8	**1.7**	70	17.7
30	90	57.4	2389	17.0	**9.2**	132	26.2
40	120	1325.3$^{(2)}$	12701	36.3	**68.6**	321	54.3
50	150	4882.9$^{(13)}$	40786	29.8	**1297.7**	3128	191.6
60	180	4863.6$^{(13)}$	23233	19.2	**2263.3**$^{(4)}$	2401	198.4
Average		1856.2$^{(28)}$		19.9	**606.8**$^{(4)}$		82.6

$^{(-)}$ the number of instances (out of 20 per instance size) were not solved to optimality

solve 28 instances to optimality. One can also observe that a number of cuts (43) were generated by the algorithm RBC is much higher compared to the other algorithms.

5.3. Results on the Stochastic and Robust Vehicle Routing with Deadlines

We now provide a comparison of the stochastic and robust models on a general problem with multiple capacitated vehicles. hlAlthough the stochastic and robust models are developed with different assumptions and objectives, we would like to evaluate the quality of the solution on different aspects produced by the two frameworks. This set of experiments is by no mean to suppoer a claim regarding the superiority of the performance of one over the other. In the first set of experiments, we examine the solutions of the two solution schemes under a known distribution. Since the instances are described by a range and a mean value, the distribution is assumed to be triangular. For the stochastic approach, we solve the problem with 1,000 and 20,000 scenarios, which are represented by STOC and STOC-L, respectively, while the robust approach is represented by RBST. These solutions are evaluated using the generated 20,000 scenarios, which is solved to optimality by the STOC-L. The tests were performed on instances with 10 to 60 nodes for the case of two deadlines and with 10 to 40 nodes for the case of $|\mathcal{N}|$ deadlines.

Since some instances were infeasible for the RBST due to the uncertainty set and some instances were not solved to optimality by all approaches, we only report the results on the instances that were solved to optimality by all approaches in order to evaluate the quality of the solutions in these two different solution schemes. In Tables 7-10, several performance measurements for the solutions obtained by each approach were computed on the set of generated scenarios as follows:

AvgDR the average value of the travel time per deadline ratio $(\tilde{t}_i/\tau_i)$, computed by

$$AvgDR = \frac{1}{|\mathcal{N}_{\mathcal{D}}|\,|\Omega|}\sum_{i\in\mathcal{N}_{\mathcal{D}}}\sum_{\omega\in\Omega}(t_{i\omega}/\tau_i)\,;$$

VarDR the variance of the travel time per deadline ratio;

MaxT the average maximum travel time among the dispatched vehicles as the ratio of the value obtained by the STOC-L;

%VSS the value of stochastic solution (Birge 1982) as a percentage of the solution obtained by the deterministic (expected value) problem;

N. Violations the average number of scenarios with a specific number of deadline violations.

To evaluate the robustness of the solutions, we perform the experiments using two different types of distributions with the same mean and interval. In this case, the solutions obtained by STOC, STOC-L and RBST are evaluated with 20,000 scenarios under two circumstances, one with the assumed distribution where the STOC-L is basically seen as an optimal one and the other distribution which is different from the assumed distribution. Since the parameters in the data set are not symmetric and thus it is difficult to generate different distributions, we assume that the mean value of the travel time is at the middle of the range. In these experiments, the stochastic approach is solved using the scenarios generated by assuming a triangular and a uniform distribution, respectively, and the results of the three approaches are evaluated by 20,000 scenarios generated by both the triangular and uniform distributions. The results for the cases of two deadlines and deadlines imposed at all nodes are are shown in Tables 7-8 and 9-10, respectively.

Under the same training and evaluation distributions, for the case of two deadlines, the performance of the STOC-L and the RBST are generally comparable in terms of the expected number of deadline violations but the STOC-L provides slightly lower numbers. The STOC is inferior to the other two in terms of solution quality. For the case where deadlines are imposed at all nodes, although the average expected number of deadline violations for the RBST is the worst in general, the RBST could still find comparable solutions to the STOC-L in many instances based on the number of better solutions. When considering other performance measurements, the RBST has slightly better results overall, such as when considering the travel time per deadline ratio, variance of the travel time per deadline ratio and the maximum distance.

When the evaluating distribution is different from the training distribution, the RBST outperforms the other two in all cases by a significant margin in terms of the expected number of deadline violations and other performance matrices. This clearly demonstrates the benefits of the RBST approach which is much less sensitive to the distributional uncertainty compared to the stochastic approach. We further note that the computing time of both the STOC and STOC-L generally increased when solving the instances using a uniform distribution while the computing time of the robust optimization remains stable since the parameters describing the uncertainty do not change. We also remark that both the stochastic and robust solutions provide substantial improvement in terms of solution quality as compared to the solutions obtained by the deterministic (expected

Table 7 Average results on the instances with $10 \leq |\mathcal{N}| \leq 30$ and two deadlines for the stochastic and robust vehicle routing problems under different distributions

Distribution		$\lvert\mathcal{V}\rvert$	Solved	Approach	Better	ψ	CPU	AvgDR	VarDR	MaxT	%VSS	N.Violations		
Training	Evaluation											1	2	≥3
Triangular	Triangular	2	48/60	STOC	42	0.02980	**0.3**	0.600	0.032	0.992	76.97%	593.2	0.9	0.0
				STOC-L	**48**	**0.02971**	2.1	0.597	0.033	1.000	77.04%	592.6	0.9	0.0
				RBST	**48**	**0.02971**	0.7	**0.568**	**0.029**	**0.915**	77.04%	592.5	0.9	0.0
		3	48/60	STOC	41	0.01670	**0.3**	0.527	0.039	1.005	85.31%	332.5	0.8	0.0
				STOC-L	**48**	**0.01660**	1.7	0.538	0.034	1.000	85.40%	330.5	0.8	0.0
				RBST	**48**	**0.01660**	0.8	**0.525**	**0.025**	**0.920**	85.40%	330.5	0.8	0.0
	Uniform	2	48/60	STOC	33	0.04688	**0.3**	0.600	0.034	0.992	74.08%	914.9	11.4	0.0
				STOC-L	36	0.04613	2.1	0.597	0.036	1.000	74.50%	900.0	11.4	0.0
				RBST	**48**	**0.04580**	0.7	**0.568**	**0.032**	**0.915**	74.68%	895.9	10.1	0.0
		3	48/60	STOC	32	0.03289	**0.3**	0.527	0.041	1.005	79.06%	635.8	11.0	0.0
				STOC-L	35	0.03117	1.7	0.538	0.036	1.000	80.15%	601.5	10.9	0.0
				RBST	**48**	**0.03097**	0.8	**0.525**	**0.027**	**0.919**	80.28%	597.8	10.8	0.0
Uniform	Triangular	2	48/60	STOC	**48**	**0.02964**	**0.5**	0.588	0.033	1.023	73.93%	591.4	0.7	0.0
				STOC-L	**48**	**0.02964**	2.4	0.574	0.033	1.000	73.93%	591.4	0.7	0.0
				RBST	**48**	**0.02964**	0.7	**0.568**	**0.029**	**0.952**	73.93%	591.4	0.7	0.0
		3	48/60	STOC	**48**	**0.01816**	**0.4**	0.545	0.032	1.040	84.03%	361.1	1.1	0.0
				STOC-L	**48**	**0.01816**	4.0	0.543	0.029	1.000	84.03%	361.1	1.1	0.0
				RBST	**48**	**0.01816**	0.8	**0.538**	**0.020**	**0.964**	84.03%	361.1	1.1	0.0
	Uniform	2	48/60	STOC	41	0.04653	**0.5**	0.588	0.035	1.023	70.37%	910.5	10.0	0.0
				STOC-L	**48**	**0.04645**	2.4	0.574	0.035	1.000	70.42%	909.1	10.0	0.0
				RBST	**48**	**0.04645**	0.7	**0.568**	**0.032**	**0.952**	70.42%	909.1	10.0	0.0
		3	48/60	STOC	38	0.03259	**0.4**	0.545	0.035	1.040	79.25%	632.5	9.6	0.0
				STOC-L	**48**	**0.03249**	4.0	0.543	0.031	1.000	79.31%	630.5	9.6	0.0
				RBST	**48**	**0.03249**	0.8	**0.538**	**0.022**	**0.964**	79.31%	630.5	9.6	0.0

value) problem. The values of %VSS are generally greater than 70% and 95% for the cases of two and $|\mathcal{N}|$ deadlines, respectively.

5.4. Results on the Stochastic and Robust Vehicle Routing with Soft Time Windows

As described in Section 3.3.2, the approaches in the paper can be extended to handle the SVRP and RVRP with soft time windows. We performed the experiments to compare the performance and solution quality of the two solution frameworks for this case. We use the same dataset as in the previous section and add the start of time window to each node with a deadline. The parameters used for the dataset are described in Appendix.

Table 11 shows the results obtained by the stochastic and robust approaches. Columns *Early*, *Late* show the expected total number of violations in the start and the end of time windows,

Table 8 Average results on the instances with $40 \leq |\mathcal{N}| \leq 60$ and two deadlines for the stochastic and robust vehicle routing problems under different distributions

Distribution		$\lvert\mathcal{V}\rvert$	Solved	Approach	Better	ψ	CPU	AvgDR	VarDR	MaxT	%VSS	N.Violations		
Training	Evaluation											1	2	≥3
Triangular	Triangular	2	46/60	STOC	37	0.03503	173.2	0.680	0.040	1.020	76.60%	566.6	67.0	0.0
				STOC-L	**46**	**0.03496**	272.5	0.684	0.037	1.000	76.64%	565.2	67.0	0.0
				RBST	44	0.03498	**124.9**	**0.634**	**0.036**	**0.911**	76.63%	583.9	66.8	0.0
		3	49/60	STOC	36	0.02041	254.4	0.627	0.051	1.021	87.91%	323.9	42.2	0.0
				STOC-L	**49**	**0.02022**	343.0	0.621	0.046	1.000	88.02%	320.7	41.9	0.0
				RBST	48	0.02025	**83.9**	**0.596**	**0.029**	**0.896**	88.01%	321.2	41.9	0.0
	Uniform	2	46/60	STOC	22	0.05337	173.2	0.680	0.041	1.020	73.78%	837.7	114.9	0.1
				STOC-L	27	0.05205	272.5	0.684	0.038	1.000	74.42%	811.0	115.4	0.1
				RBST	**43**	**0.05160**	**124.9**	**0.634**	**0.037**	**0.911**	74.64%	829.3	110.5	0.0
		3	49/60	STOC	21	0.03000	254.4	0.627	0.052	1.021	85.77%	475.8	63.0	0.0
				STOC-L	34	0.02733	343.0	0.621	0.048	1.000	87.03%	429.1	59.1	0.0
				RBST	**48**	**0.02697**	**83.9**	**0.596**	**0.030**	**0.896**	87.20%	422.6	58.4	0.0
Uniform	Triangular	2	44/60	STOC	40	0.03904	210.8	0.684	0.036	1.032	78.27%	774.5	3.1	0.0
				STOC-L	**42**	0.04034	286.5	0.650	0.041	1.000	77.55%	800.3	3.3	0.0
				RBST	**42**	**0.03763**	**63.7**	**0.635**	**0.033**	**0.928**	79.05%	610.9	70.9	0.0
		3	48/60	STOC	45	0.02204	134.6	0.623	0.041	1.042	86.67%	439.0	1.0	0.0
				STOC-L	**47**	0.02204	137.4	0.596	0.043	1.000	86.67%	438.9	1.0	0.0
				RBST	**47**	**0.02078**	**81.3**	**0.592**	**0.029**	**0.933**	87.43%	329.9	42.8	0.0
	Uniform	2	44/60	STOC	20	0.05119	210.8	0.684	0.037	1.032	77.10%	961.2	29.4	0.2
				STOC-L	**44**	**0.05068**	286.5	0.650	0.043	1.000	77.32%	984.7	30.5	0.2
				RBST	39	0.05558	**63.7**	**0.635**	**0.034**	**0.928**	75.13%	877.7	117.0	0.0
		3	48/60	STOC	30	0.02740	134.6	0.623	0.042	1.042	86.72%	502.5	9.4	0.0
				STOC-L	**48**	**0.02588**	137.4	0.596	0.044	1.000	87.45%	498.8	9.4	0.0
				RBST	47	0.02601	**81.3**	**0.592**	**0.030**	**0.933**	87.39%	433.3	56.8	0.0

respectively, while column *Total* shows the expected total number of violations. Columns *AvgER* and *VarER* are calculated as the *AvgDR* and *VarDR*, respectively, but the deadline value is replaced by the start of the time window. Therefore, $AvgER \geq 1$ indicates that the average arrival time at the node is not less than the start of the time window.

As opposed to the problem with deadlines, the robust approach applied to the problem with time windows is not very competitive as compared to the stochastic solutions. One possible explanation is that the worst-case distributions of the start and the end of time windows are taken into account separately in this robust framework and thus the solution can be overly conservative when evaluated against known and assumed distributions. However, the robust approach can still produce a good solution compared to the deterministic approach and the robust framework can still be very useful when the information of the travel time distributions is not available. Additionally, the presence of

Table 9 Average results on the instances with $10 \leq |\mathcal{N}| \leq 20$ and deadlines imposed at all nodes for the stochastic and robust vehicle routing problems under different distributions

Distribution		$\|\mathcal{V}\|$	Solved	Approach	Better	ψ	CPU	AvgDR	VarDR	MaxT	%VSS	N.Violations		
Training	Evaluation											1	2	≥3
Triangular	Triangular	2	37/40	STOC	34	0.02456	53.0	0.370	0.057	1.011	97.87%	505.8	103.9	24.0
				STOC-L	**37**	**0.02454**	127.1	0.372	0.057	1.000	97.87%	503.9	103.7	23.9
				RBST	36	0.02550	**31.6**	**0.364**	**0.053**	**0.948**	97.79%	471.8	126.4	23.8
		3	37/40	STOC	32	0.01538	20.5	0.334	0.053	1.001	98.67%	476.9	76.8	10.5
				STOC-L	**37**	**0.01535**	67.8	0.331	0.052	1.000	98.67%	476.0	76.5	10.5
				RBST	36	0.01648	**19.6**	**0.327**	**0.050**	**0.984**	98.57%	443.9	100.4	10.4
	Uniform	2	37/40	STOC	31	0.04955	53.0	0.370	0.058	1.010	95.78%	667.5	173.7	94.3
				STOC-L	31	**0.04927**	127.1	0.372	0.058	1.000	95.80%	640.8	170.8	93.2
				RBST	**36**	0.04968	**31.6**	**0.364**	**0.054**	**0.949**	95.77%	594.8	208.5	89.8
		3	37/40	STOC	31	0.03032	20.5	0.334	0.054	1.001	97.42%	584.0	143.1	46.5
				STOC-L	33	**0.02958**	67.8	0.331	0.053	1.000	97.48%	569.6	138.8	46.6
				RBST	**36**	0.03122	**19.6**	**0.327**	**0.051**	**0.982**	97.34%	527.5	176.9	42.2
Uniform	Triangular	2	37/40	STOC	**37**	**0.02400**	60.5	0.367	0.055	1.014	97.92%	1325.1	570.4	77.5
				STOC-L	**37**	**0.02400**	161.7	**0.358**	0.055	1.000	97.92%	1325.1	570.4	77.5
				RBST	36	0.02504	**31.8**	0.364	**0.053**	**0.969**	97.83%	979.8	311.3	130.7
		3	37/40	STOC	**37**	**0.01529**	23.2	0.328	0.051	0.999	98.68%	537.4	67.7	3.2
				STOC-L	**37**	**0.01529**	99.9	0.328	0.051	1.000	98.68%	537.4	67.7	3.2
				RBST	36	0.01635	**19.6**	**0.327**	**0.050**	**0.983**	98.58%	441.0	101.9	10.6
	Uniform	2	37/40	STOC	34	0.04929	60.5	0.367	0.057	1.015	95.80%	676.6	155.7	71.9
				STOC-L	**37**	**0.04927**	161.7	**0.358**	0.056	1.000	95.80%	676.0	155.7	71.9
				RBST	36	0.05112	**31.8**	0.364	**0.054**	**0.971**	95.65%	593.7	209.9	89.7
		3	37/40	STOC	33	0.02978	23.2	0.328	0.053	0.998	97.46%	618.3	124.3	28.1
				STOC-L	**37**	**0.02974**	99.9	0.328	0.052	1.000	97.47%	617.5	124.3	28.1
				RBST	36	0.03144	**19.6**	**0.327**	**0.051**	**0.983**	97.32%	531.6	176.1	43.6

constraints from Slater's conditions (57) and (58) also ensures that the extected arrival times must be within time windows. We further note that for the case where the early start constraints are imposed (which can be done by setting the deadlines to a large value), the computational results are similar to the case where only deadlines are imposed.

6. Conclusion

We examine the computational aspect of the stochastic and robust vehicle routing problem with deadlines under travel time uncertainty. The first approach is applied to the case when the probability distributions of the travel times are known, while the second approach is used to deal with the case where the exact distributions are not known. We introduce a general routing set, formulations and propose solutions approaches based on the branch-and-cut and Benders decomposition frameworks to solve these problems. We also discuss the extensions of the frameworks to the cases

Table 10 Average results on the instances with $30 \leq |\mathcal{N}| \leq 40$ and deadlines imposed at all nodes for the stochastic and robust vehicle routing problems under different distributions

Distribution		$\lvert\mathcal{V}\rvert$	Solved	Approach	Better	ψ	CPU	AvgDR	VarDR	MaxT	%VSS	N.Violations		
Training	Evaluation											1	2	≥3
Triangular	Triangular	2	33/40	STOC	24	0.05427	**223.5**	0.394	0.064	1.022	97.51%	750.0	138.1	19.5
				STOC-L	**33**	**0.05385**	658.9	0.392	0.063	1.000	97.53%	601.7	206.4	20.7
				RBST	32	0.05397	236.1	**0.372**	**0.055**	**0.921**	97.52%	746.1	137.1	19.5
		3	35/40	STOC	27	0.03928	**55.7**	0.349	0.054	0.990	98.43%	497.1	119.6	16.3
				STOC-L	**35**	**0.03919**	154.6	0.349	0.054	1.000	98.44%	496.0	119.2	16.3
				RBST	**35**	**0.03919**	221.0	**0.333**	**0.048**	**0.905**	98.44%	496.0	119.2	16.3
	Uniform	2	33/40	STOC	20	0.08373	**223.5**	0.394	0.065	1.021	96.20%	794.3	286.9	97.1
				STOC-L	20	0.08190	658.9	0.392	0.064	1.000	96.28%	622.4	349.0	101.9
				RBST	**33**	**0.07855**	236.1	**0.372**	**0.056**	**0.923**	96.43%	738.8	267.0	94.4
		3	35/40	STOC	24	0.05648	**55.7**	0.349	0.055	0.989	97.77%	474.6	222.2	68.4
				STOC-L	27	0.05460	154.6	0.349	0.055	1.000	97.84%	450.7	215.1	68.5
				RBST	**35**	**0.05432**	221.0	**0.333**	**0.049**	**0.905**	97.85%	447.5	214.2	68.3
Uniform	Triangular	2	32/40	STOC	30	0.05937	287.1	0.394	0.062	1.025	96.96%	873.3	124.1	21.7
				STOC-L	30	0.05937	804.3	0.387	0.059	1.000	96.96%	873.3	124.1	21.7
				RBST	**32**	**0.05579**	**220.4**	**0.372**	**0.055**	**0.935**	97.15%	765.5	143.1	20.9
		3	35/40	STOC	34	0.04092	**160.3**	0.343	0.053	1.007	97.99%	596.2	104.1	4.7
				STOC-L	34	0.04092	363.2	0.341	0.051	1.000	97.99%	596.2	104.1	4.7
				RBST	**35**	**0.03935**	219.8	**0.333**	**0.048**	**0.928**	98.07%	492.5	122.1	16.7
	Uniform	2	32/40	STOC	20	0.07610	287.1	0.394	0.063	1.025	96.16%	831.1	251.4	59.0
				STOC-L	**32**	**0.07599**	804.3	0.388	0.060	1.000	96.17%	829.2	251.3	59.0
				RBST	30	0.08089	**220.4**	**0.372**	**0.056**	**0.937**	95.92%	757.7	277.9	95.9
		3	32/40	STOC	26	0.05133	**160.3**	0.343	0.053	1.007	97.52%	522.9	191.6	39.9
				STOC-L	**35**	**0.05117**	363.2	0.341	0.052	1.000	97.53%	520.3	191.3	39.9
				RBST	34	0.05465	219.8	**0.333**	**0.049**	**0.929**	97.36%	451.1	212.5	70.1

where service times and soft time windows are incorporated. In our computational experiments, the proposed models and algorithms generally outperformed the approaches presented in the literature. Additionally, the proposed solution framework for the stochastic problem can handle instances with a large number of scenarios. We further examine the two formulation schemes in terms of solution quality based on several performance measurements. The results show that the solutions produced by the stochastic model under a large number of scenarios are slightly better than those obtained by the robust optimization framework when the exact probability distribution is known, while the solutions obtained by the robust approach is superior to the other when the distribution is unknown.

Acknowledgments

Table 11 Average results on the instances with $10 \leq |\mathcal{N}| \leq 40$ and two soft time windows for the stochastic and robust vehicle routing problems under different distributions

Distribution		$\lvert\mathcal{V}\rvert$	Solved	Approach	Better	Early	Late	Total	CPU	AvgER	VarER	AvgDR	VarDR	MaxT	%VSS
Training	Evaluation														
Triangular	Triangular	2	65/80	STOC	51	**0.02872**	0.02796	0.05667	27.7	**0.680**	**0.020**	2.149	8.186	1.009	95.81%
				STOC-L	**65**	0.03200	**0.02423**	**0.05623**	60.7	0.679	0.021	2.077	7.950	1.000	95.84%
				RBST	39	0.03541	0.04884	0.08425	**15.2**	0.665	0.023	**1.844**	**2.471**	0.985	93.77%
		3	65/80	STOC	46	0.02942	0.02388	0.05330	48.6	0.510	**0.098**	1.363	3.051	1.000	96.72%
				STOC-L	**65**	**0.02927**	**0.02368**	**0.05294**	96.6	**0.514**	**0.098**	1.389	3.211	1.000	96.74%
				RBST	38	0.03818	0.04859	0.08677	**16.2**	0.503	0.099	**1.355**	**2.926**	1.010	94.66%
	Uniform	2	65/80	STOC	43	**0.05337**	0.05179	0.10516	27.7	**0.680**	**0.023**	2.149	8.290	1.009	92.30%
				STOC-L	**49**	0.05674	**0.04764**	**0.10438**	60.7	0.679	**0.023**	2.077	8.057	1.000	92.35%
				RBST	39	0.05998	0.08361	0.14360	**15.2**	0.665	0.026	**1.844**	**2.552**	0.985	89.48%
		3	65/80	STOC	40	0.05153	0.04784	0.09938	48.6	0.511	0.101	1.364	3.124	1.005	93.95%
				STOC-L	**49**	**0.04956**	**0.04694**	**0.09650**	96.6	**0.514**	**0.099**	1.387	3.274	1.000	94.13%
				RBST	34	0.06281	0.08509	0.14789	**16.2**	0.503	0.101	**1.355**	**2.995**	1.013	91.00%
	Discrete	2	65/80	STOC	36	0.15079	0.15920	0.30999	27.7	0.672	0.038	1.993	5.923	1.005	
				STOC-L	42	0.15738	**0.14123**	**0.29861**	60.7	**0.675**	0.037	1.938	4.363	1.000	
				RBST	**44**	**0.13138**	0.16745	0.29883	**15.2**	0.665	**0.036**	**1.887**	**4.150**	0.998	
		3	65/80	STOC	37	0.13607	0.15579	0.29187	48.6	0.518	0.107	1.513	6.729	1.000	
				STOC-L	**45**	0.13707	**0.14750**	**0.28458**	96.6	**0.520**	0.108	1.566	8.675	1.000	
				RBST	38	**0.12600**	0.16578	0.29177	**16.2**	0.504	**0.101**	**1.350**	**2.982**	0.990	
Uniform	Triangular	2	65/80	STOC	62	0.02835	0.02806	0.05640	35.2	**0.678**	**0.021**	2.056	7.070	1.017	95.83%
				STOC-L	**63**	**0.02832**	**0.02805**	**0.05637**	76.1	0.668	0.022	1.947	4.248	1.000	95.83%
				RBST	41	0.03536	0.04904	0.08440	**15.1**	0.665	0.023	**1.844**	**2.470**	0.992	93.76%
		3	65/80	STOC	61	0.02571	0.02780	0.05351	65.0	0.509	**0.097**	1.407	4.198	0.995	96.71%
				STOC-L	**62**	**0.02571**	**0.02773**	**0.05344**	133.1	**0.511**	**0.097**	1.433	4.320	1.000	96.71%
				RBST	39	0.03794	0.04850	0.08644	**16.2**	0.503	0.099	**1.355**	**2.927**	1.013	94.68%
	Uniform	2	65/80	STOC	50	0.05145	**0.05066**	0.10211	35.2	**0.678**	**0.024**	2.057	7.169	1.018	92.52%
				STOC-L	**65**	**0.05105**	0.05079	**0.10184**	76.1	0.668	0.025	1.947	4.347	1.000	92.54%
				RBST	33	0.06004	0.08314	0.14318	**15.1**	0.665	0.026	**1.844**	**2.553**	0.992	89.51%
		3	65/80	STOC	47	0.04404	0.05024	0.09428	65.0	0.509	**0.099**	1.407	4.258	0.995	94.26%
				STOC-L	**65**	**0.04398**	**0.04998**	**0.09396**	133.1	**0.511**	**0.099**	1.433	4.387	1.000	94.28%
				RBST	27	0.06284	0.08549	0.14833	**16.2**	0.503	0.101	**1.355**	**2.996**	1.013	90.97%

This research is supported by the Singapore-MIT Alliance for Research and Technology (SMART) Center for Future Mobility (FM). The authors would like to thank Jin Qi and Melvyn Sim from National University of Singapore for their guidance on the implementation of the robust optimization framework. Finally, we are grateful to three anonymous referees for their valuable comments on an earlier version of the paper.

Appendices

This is the online supplement of the paper: Models and Solutions Algorithms of the Stochastic and Robust Vehicle Routing with Deadlines. Section A provides the details of dataset used in our experiments. Section B and C presents the models used in their papers and also how they are adapted to solve the problems in our study. Section D provides the details of subgradients computation. We make the dataset and the detailed results available on the website: https://sites.google.com/site/YossiriAdulyasak/publications.

Appendix A: Details of the Dataset

The details of the dataset of Jaillet et al. (2014) and the parameters used for generating instances with multiple capacitated vehicles are as follows. Note that $[x]$ denotes the nearest integer to x.

1. **Instance size**. The dataset consists of the instances of the size $|\mathcal{N}| = 10$ to 60 with an increment of 10 for both the cases of two deadlines and $|\mathcal{N}|$ deadlines. To test our algorithm, we also obtain the instances of the size $|\mathcal{N}| = 70, 80$ for the case of two deadlines from the authors directly. The number of arcs $|\mathcal{A}| = 3|\mathcal{N}|$ for all instances;

2. **Travel time and deadline**: The arc travel time is described by a minimum, maximum and mean value, i.e., $\tilde{c}_a \in [\underline{c}_a, \overline{c}_a]$ and $\mathbb{E}_{\mathbb{P}}(\tilde{c}_a) = \mu$. Deadlines are imposed at nodes $[n/2]$ and n for the case of two deadlines. The values of the deadlines are provided in the dataset on the website;

3. **Vehicle**. Since the case of a single uncapacitated vehicle is considered in the original dataset, we create a dataset of customer demands for each instance size and the vehicle capacity is computed as $Q = [0.75D]$, where D is the sum of customer demands which are generated for each instance size.

Appendix B: Model of Kenyon and Morton (2003) (KM)

Denote by x_{ak} equal to one if arc a is traversed by vehicle k and z_{ik} equal to one if node i is visited by vehicle k. Let also $\mathcal{A}(\mathcal{H})$ denotes the set of arcs where $\mathcal{A}(\mathcal{H}) = \{\, (i,j) \in \mathcal{A}(\mathcal{H}) |\, i,j \in \mathcal{H}\}$. The model for the SVRP-MD presented in Kenyon and Morton (2003) is as follows:

$$\min \frac{1}{|\Omega|} \sum_{\omega \in \Omega} \phi_\omega \tag{70}$$

$$\text{s.t.} \quad \sum_{a \in \mathcal{A}} c_{a\omega} x_{ak} \le \tau_i + M_{i\omega} \phi_\omega \forall k \in \mathcal{K}, \forall \omega \in \Omega \tag{71}$$

$$\phi_\omega \in \{0,1\} \qquad \forall \omega \in \Omega \tag{72}$$

$$(\boldsymbol{x}) \in \mathcal{S}^{KM}. \tag{73}$$

Note that for the case of minimizing the expected number of deadline violations or the sum of probability of deadline violations can be formulated using the variable $\phi_{i\omega}$ as in (22)-(25) which is relatively straightforward and the details are omitted here. The routing set $\mathcal{S}^{KM}$ is defined as follows:

$$\mathcal{S}^{KM} = \left\{ \boldsymbol{x} \in \{0,1\}^{|\mathcal{A}| \times |\mathcal{K}|}, \boldsymbol{z} \in \mathbb{Z}_+^{|\mathcal{N}| \times |\mathcal{K}|} \,\middle|\, (74) - (78) \right\}$$

where

$$\sum_{a\in\delta^+(i)} x_{ak} = \sum_{a\in\delta^-(i)} x_{ak} \qquad \forall i\in\mathcal{N}, \forall k\in\mathcal{K} \tag{74}$$

$$\sum_{k\in\mathcal{K}} z_{ik} = 1 \qquad \forall i\in\mathcal{N}_{\mathcal{R}} \tag{75}$$

$$z_{ik} \le \sum_{a\in\delta^+(i)} x_{ak} \qquad \forall i\in\mathcal{N}\backslash\{1\}, \forall k\in\mathcal{K} \tag{76}$$

$$z_{1k} \le \sum_{a\in\delta^-(1)} x_{ak} \qquad \forall k\in\mathcal{K} \tag{77}$$

$$\sum_{a\in\delta^+(\mathcal{H})} x_{ak} \ge \frac{1}{|\mathcal{A}(\mathcal{H})|}\sum_{a\in\mathcal{A}(\mathcal{H})} x_{ak} \quad \forall \mathcal{H}\subseteq\mathcal{N}\setminus\{1\} : |\mathcal{H}|\ge 2, \forall j\in\mathcal{H}, \forall k\in\mathcal{K}. \tag{78}$$

For the case of capacitated vehicles, the following constraints are also added to the set $\mathcal{S}^{KM}$:

$$\sum_{i\in\mathcal{N}} d_i z_{ik} \le Q \quad \forall k\in\mathcal{K}. \tag{79}$$

To solve the problem as considered in Kenyon and Morton (2003) where the deadline violation of the vehicle with longest travel time is taken into account, a simple modification can be made to the model (18)-(21). First, all variables $\rho_i, \forall i\in\mathcal{N}_{\mathcal{D}}$ are replaced with a single variable ρ_0 and, second, constraints associated with the probability of deadline violation of the longest route, denote by $\beta^0_{\boldsymbol{p}} = \sum_{\omega\in\Omega}\max_{i\in\mathcal{N}_{\mathcal{D}}}\{\phi_{i\omega}\}$, are added. The reformulation for this problem can be written as follows:

$$\min\ \rho_0 \tag{80}$$

$$\text{s.t.} \qquad \sum_{a\in\mathcal{A}^i_{\boldsymbol{p}}} \beta^i_{\boldsymbol{p}}(s^i_a - 1) + \beta^i_{\boldsymbol{p}} \le \rho_0 \quad \forall i\in\mathcal{N}_{\mathcal{D}}, \forall(\boldsymbol{p})\in\mathcal{S} \tag{81}$$

$$\sum_{i\in\mathcal{N}_{\mathcal{D}}}\sum_{a\in\mathcal{A}^i_{\boldsymbol{p}}} \beta^0_{\boldsymbol{p}}(s^i_a - 1) + \beta^0_{\boldsymbol{p}} \le \rho_0 \quad \forall(\boldsymbol{p})\in\mathcal{S} \tag{82}$$

$$\rho_0 \ge 0 \tag{83}$$

$$(\boldsymbol{s})\in\mathcal{S}. \tag{84}$$

Appendix C: Model of Jaillet et al. (2014) (JQS)

The set $\mathcal{S}$ of the formulation MCF of Jaillet et al. (2014), denote by$\mathcal{S}^{MCF}$, is as follows:

$$\mathcal{S}^{MCF} = \{\boldsymbol{s}\,|(1)-(5)\,\}$$

where

$$\sum_{a\in\delta^-(u)} s^i_a - \sum_{a\in\delta^+(u)} s^i_a = 0 \qquad \forall i\in\mathcal{N}\setminus\{1\}, u\in\mathcal{N}\setminus\{1,n,i\} \tag{85}$$

$$\sum_{a\in\delta^+(1)} s^i_a = \sum_{a\in\delta^-(i)} x_a \quad \forall i\in\mathcal{N}\setminus\{1\} \tag{86}$$

$$\sum_{a\in\delta^-(i)} s^i_a - \sum_{a\in\delta^+(i)} s^i_a = \sum_{a\in\delta^-(i)} x_a \quad \forall i\in\mathcal{N}\setminus\{1\} \tag{87}$$

$$0 \le s^i_a \le x_a \qquad \forall i\in\mathcal{N}\setminus\{1\}, \forall a\in\mathcal{A} \tag{88}$$

$$\sum_{a\in\delta^+(i)} x_a = 1 \qquad \forall i \in \mathcal{N}_{\mathcal{R}} \setminus \{n\} \tag{89}$$

$$\sum_{a\in\delta^-(i)} x_a = 1 \qquad \forall i \in \mathcal{N}_{\mathcal{R}} \setminus \{1\} \tag{90}$$

$$\sum_{a\in\delta^+(i)} x_a \le 1 \qquad \forall i \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{R}} \tag{91}$$

$$\sum_{a\in\delta^-(i)} x_a - \sum_{a\in\delta^+(i)} x_a \le 0 \qquad \forall i \in \mathcal{N} \setminus \mathcal{N}_{\mathcal{R}} \tag{92}$$

$$x_a \in \{0,1\} \qquad \forall a \in \mathcal{A} \tag{93}$$

Appendix D: Subgradient Parameters Computation

Following from the results in Jaillet et al. (2014), we present the details of the subgradient computation for the case where $\mathbb{F} = \{\mathbb{P} | \mathbb{E}_{\mathbb{P}}(\tilde{c}) = \mu, \mathbb{P}(\tilde{c} \in [\underline{c}, \overline{c}]) = 1]$ for both the the lateness and earliness indices in this section. Note that for the case that only deadlines are imposed, one can simply treat $\sigma_i = 0, \forall i \in \mathcal{N}_{\mathcal{D}}$ and thus $\eta_i = 0, \forall i \in \mathcal{N}_{\mathcal{D}}$.

Using the Edmonton-Madansky bound (Madansky 1959), one obtains

$$C_{\alpha_i,\mathbb{F}}(\tilde{c}_a s_a^i) = \sup_{\mathbb{P}\in\mathbb{F}} \alpha_i \ln \mathbb{E}_{\mathbb{P}}\left(\exp\left(\frac{\tilde{c}_a s_a^i}{\alpha_i}\right)\right) = \begin{cases} \alpha_i \ln\left(\frac{(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + (\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}{\overline{c}_a - \underline{c}_a}\right) & \text{if } \alpha_i > 0, \\ \overline{c}_a, & \text{if } \alpha_i = 0 \end{cases}$$

and

$$C_{\eta_i,\mathbb{F}}(-\tilde{c}_a s_a^i) = \sup_{\mathbb{P}\in\mathbb{F}} \eta_i \ln \mathbb{E}_{\mathbb{P}}\left(\exp\left(-\frac{\tilde{c}_a s_a^i}{\eta_i}\right)\right) = \begin{cases} \eta_i \ln\left(\frac{(\overline{c}_a - \mu_a)\exp\left(-\frac{\underline{c}_a s_a^i}{\eta_i}\right) + (\mu_a - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\eta_i}\right)}{\overline{c}_a - \underline{c}_a}\right) & \text{if } \eta_i > 0, \\ -\underline{c}_a, & \text{if } \eta_i = 0. \end{cases}.$$

Note that $\alpha_i^{\boldsymbol{s}}$ and $\eta_i^{\boldsymbol{s}}$ are the solution values of α_i and η_i associated with a given solution $\boldsymbol{s}$ which are obtained after solving the subproblems. The subgradient $d^f_{s_a^i}(\boldsymbol{s})$ and $d^g_{s_a^i}(\boldsymbol{s})$ for a solution $\boldsymbol{s} \in \mathcal{S}$ for the case where $\varphi(\boldsymbol{\alpha}, \boldsymbol{\eta}) = \sum_{i\in\mathcal{N}_{\mathcal{D}}} (\alpha_i + \eta_i)$ can be computed as (Jaillet et al. 2014)

$$d^f_{s_a^i}(\boldsymbol{s}) = \begin{cases} -\frac{\frac{\partial h(\alpha_i, \boldsymbol{s}^i)}{\partial s_a^i}}{\frac{\partial h(\alpha_i, \boldsymbol{s}^i)}{\partial \alpha_i}} & \forall i \in \mathcal{N}_{\mathcal{D}} | \alpha_i^{\boldsymbol{s}} > 0 \\ 0 & \text{otherwise} \end{cases} \text{ and } d^g_{s_a^i}(\boldsymbol{s}) = \begin{cases} -\frac{\frac{\partial e(\eta_i, \boldsymbol{s}^i)}{\partial s_a^i}}{\frac{\partial e(\eta_i, \boldsymbol{s}^i)}{\partial \eta_i}} & \forall i \in \mathcal{N}_{\mathcal{D}} | \eta_i^{\boldsymbol{s}} > 0 \\ 0 & \text{otherwise.} \end{cases}$$

Consequently, the gradient of function $h(\alpha_i, \boldsymbol{s}^i)$ with respect to $\boldsymbol{s}^i$ can be computed as

$$\frac{\partial h(\alpha_i, \boldsymbol{s}^i)}{\partial s_a^i} = \frac{\underline{c}_a(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + \overline{c}_a(\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}{(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + (\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)},$$

and the gradient of function $h(\alpha_i, \boldsymbol{s}^i)$ with respect to α_i can be computed as

$$\begin{aligned} \frac{\partial h(\alpha_i, \boldsymbol{s}^i)}{\partial \alpha_i} =& \sum_{a\in\mathcal{A}} \ln\left(\frac{(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + (\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}{\overline{c}_a - \underline{c}_a}\right) \\ &- \sum_{a\in\mathcal{A}} \left(\frac{s_a^i}{\alpha_i}\right)\left(\frac{\underline{c}_a(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + \overline{c}_a(\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}{(\overline{c}_a - \mu_a)\exp\left(\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + (\mu_a - \underline{c}_a)\exp\left(\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}\right). \end{aligned}$$

Similarly, the gradient of function $e(\eta_i, \boldsymbol{s}^i)$ with respect to $\boldsymbol{s}^i$ and with respect to η_i can be computed as

$$\frac{\partial e(\eta_i, \boldsymbol{s}^i)}{\partial s_a^i} = \frac{(-\underline{c}_a)(\overline{c}_a - \mu)\exp\left(-\frac{\underline{c}_a s_a^i}{\eta_i}\right) + (-\overline{c}_a)(\mu - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\eta_i}\right)}{(\overline{c}_a - \mu_a)\exp\left(-\frac{\underline{c}_a s_a^i}{\eta_i}\right) + (\mu - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\eta_i}\right)},$$

and

$$\begin{aligned}\frac{\partial e(\alpha_i, \boldsymbol{s}^i)}{\partial \alpha_i} =& \sum_{a \in \mathcal{A}} \ln\left(\frac{(\overline{c}_a - \mu)\exp\left(-\frac{\underline{c}_a s_a^i}{\alpha_i}\right) + (\mu - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\alpha_i}\right)}{\overline{c}_a - \underline{c}_a}\right) \\ &+ \sum_{a \in \mathcal{A}} \left(\frac{s_a^i}{\eta_i}\right)\left(\frac{(-\underline{c}_a)(\overline{c}_a - \mu)\exp\left(-\frac{\underline{c}_a s_a^i}{\eta_i}\right) + (-\overline{c}_a)(\mu - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\eta_i}\right)}{(\overline{c}_a - \mu_a)\exp\left(-\frac{\underline{c}_a s_a^i}{\eta_i}\right) + (\mu - \underline{c}_a)\exp\left(-\frac{\overline{c}_a s_a^i}{\eta_i}\right)}\right).\end{aligned}$$

References

Adulyasak, Y., J.-F. Cordeau, R. Jans. 2013. Formulations and branch-and-cut algorithms for multi-vehicle production and inventory routing problems. *INFORMS J. Comput.* **Forthcoming**.

Agra, A., M. Christiansen, R. Figueiredo, L. M. Hvattum, M. Poss, C. Requejo. 2013. The robust vehicle routing problem with time windows. *Comput. Oper. Res.* **40**(3) 856–866.

Applegate, D., R. Bixby, V. Chvátal, W. Cook. 2011. Concorde TSP solver Http://www.tsp.gatech.edu/concorde.html.

Baldacci, R., N. Christofides, A. Mingozzi. 2008. An exact algorithm for the vehicle routing problem based on the set partitioning formulation with additional cuts. *Math. Programming* **115**(2) 351–385.

Benders, J. F. 1962. Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik* **4** 238–252.

Bertsimas, D. J., D. Brown, C. Caramanis. 2011. Theory and applications of robust optimization. *SIAM Review* **53**(3) 464–501.

Bertsimas, D. J., M. Sim. 2004. The price of robustness. *Oper. Res.* **52**(1) 35–53.

Birge, J. R. 1982. The value of the stochastic solution in stochastic linear programs with fixed recourse. *Math. Programming* **24**(1) 314–325.

Birge, J. R., F. V. Louveaux. 2011. Two-stage recourse problems. *Introduction to Stochastic Programming*. Springer Series in Operations Research and Financial Engineering, Springer New York, 181–263.

Brown, D. B., M. Sim. 2009. Satisficing measures for analysis of risky positions. *Management Sci.* **55**(1) 71–84.

Campbell, A. M., B. W. Thomas. 2008. Probabilistic traveling salesman problem with deadlines. *Transportation Sci.* **42**(1) 1–21.

Campbell, A. M., B. W. Thomas. 2009. Runtime reduction techniques for the probabilistic traveling salesman problem with deadlines. *Comput. Oper. Res.* **36**(4) 1231–1248.

Carlsson, J. G., E. Delage. 2013. Robust partitioning for stochastic multivehicle routing. *Oper. Res.* **61**(3) 727–744.

Fischetti, M., P. Toth, D. Vigo. 1994. A branch-and-bound algorithm for the capacitated vehicle routing problem on directed graphs. *Oper. Res.* **42**(5) 846–859.

Fukasawa, R., H. Longo, J. Lysgaard, M. P. de Aragão, M. Reis, E. Uchoa, R. F. Werneck. 2006. Robust branch-and-cut-and-price for the capacitated vehicle routing problem. *Math. Programming* **106**(3) 491–511.

Gendreau, M., G. Laporte, R. Séguin. 1996. Stochastic vehicle routing. *Eur. J. Oper. Res.* **88**(1) 3–12.

Gounaris, C. E., W. Wiesemann, C. A. Floudas. 2013. The robust capacitated vehicle routing problem under demand uncertainty. *Oper. Res.* **61**(3) 677–693.

Jaillet, P. 1988. A priori solution of a traveling salesman problem in which a random subset of the customers are visited. *Oper. Res.* **36**(6) 929–936.

Jaillet, P., J. Qi, M. Sim. 2014. Routing optimization under uncertainty. Working paper, Massachusetts Institute of Technology.

Jans, R. 2009. Solving lot-sizing problems on parallel identical machines using symmetry-breaking constraints. *INFORMS J. Comput.* **21**(1) 123–136.

Kenyon, S. A., D. P. Morton. 2003. Stochastic vehicle routing with random travel times. *Transportation Sci.* **37**(1) 69–82.

Lam, S-W., T. S. Ng, M. Sim, J-H Song. 2013. Multiple objectives satisficing under uncertainty. *Oper. Res.* **61**(1) 214–227.

Lambert, V., G. Laporte, F. Louveaux. 1993. Designing collection routes through bank branches. *Comput. Oper. Res.* **20**(7) 783–791.

Laporte, G., F. Louveaux, H. Mercure. 1992. The vehicle routing problem with stochastic travel times. *Transportation Sci.* **26**(3) 161–170.

Lee, C., K. Lee, S. Park. 2012. Robust vehicle routing problem with deadlines and travel time/demand uncertainty. *J. Oper. Res. Soc.* **63**(9) 1294–1306.

Li, X., P. Tian, S. C.H. Leung. 2010. Vehicle routing problems with time windows and stochastic travel and service times: Models and algorithm. *Int. J. Prod. Econ.* **125**(1) 137–145.

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

Madansky, A. 1959. Bounds on the expectation of a convex function of a multivariate random variable. *Ann. Math. Stat.* **30**(3) 743–746.

Montemanni, R., J. Barta, M. Mastrolilli, L. M. Gambardella. 2007. The robust traveling salesman problem with interval data. *Transportation Sci.* **41**(3) 366–381.

Padberg, M., T-Y. Sung. 1991. An analytical comparison of different formulations of the travelling salesman problem. *Math. Programming* **52** 315–357.

Roberti, R., P. Toth. 2012. Models and algorithms for the asymmetric traveling salesman problem: an experimental comparison. *EURO Journal on Transportation and Logistics* **1** 113–133.

Russell, R. A., T. L. Urban. 2008. Vehicle routing with soft time windows and erlang travel times. *J. Oper. Res. Soc.* **59**(9) 1220–1228.

Scarf, H., K. J. Arrow, S. Karlin. 1958. A min-max solution of an inventory problem. *Studies in the mathematical theory of inventory and production* **10** 201–209.

Sherali, H. D., J. C. Smith. 2001. Improving discrete model representations via symmetry considerations. *Management Sci.* **47**(10) 1396–1407.

Sungur, I., Y. Ren, F. Ordó nez, M. Dessouky, H. Zhong. 2010. A model and algorithm for the courier delivery problem with uncertainty. *Transportation Sci.* **44**(2) 193–205.

Taş, D., N. Dellaert, T. van Woensel, T. de Kok. 2013. Vehicle routing problem with stochastic travel times including soft time windows and service costs. *Comput. Oper. Res.* **40**(1) 214–224.

Toth, P., D. Vigo. 2001. An overview of vehicle routing problems. P. Toth, D. Vigo, eds., *The Vehicle Routing Problem*. Society for Industrial and Applied Mathematics, 1–26.

Tropp, J. A. 2012. User-friendly tail bounds for sums of random matrices. *Found. Comput. Math.* **12**(4) 389–434.

Verweij, B., S. Ahmed, A. J. Kleywegt, G. Nemhauser, A. Shapiro. 2003. The sample average approximation method applied to stochastic routing problems: A computational study. *Comput. Optim. Appl.* **24**(2-3) 289–333.