informs
http://pubsonline.informs.org/journal/ijds

ANY INFORMS JOURNAL
Vol. 00, No. 0, Xxxxx 0000, pp. 000–000
ISSN 2694-4022, EISSN 2694-4030

Submitted to *Any INFORMS Journal*

The Service-Centric Vehicle Routing Problem with Crowdshipping

Bingjie Zhou, Yu Zhang*
Department of Supply Chain Management, School of Business Administration, Southwestern University of Finance and Economics, Chengdu 611130, China, bj.zhou@foxmail.com, y.zhang@swufe.edu.cn

Roberto Baldacci
Division of Engineering Management and Decision Sciences, College of Science and Engineering, Hamad Bin Khalifa University, Qatar Foundation, Doha, Qatar, rbaldacci@hbku.edu.qa

Jiafu Tang
College of Management Science and Engineering, Dongbei University of Finance and Economics, Shahekou, Dalian, 116026, P.R. China, tangjiafu@mail.neu.edu.cn

Abstract. Last-mile delivery services worldwide have embraced crowdshipping, which involves both regular and occasional drivers to reduce transportation costs and potentially ensure timely deliveries. However, real-world uncertainty in travel times leads to delays in deliveries. Motivated by empirical studies on customer impatience with late deliveries, this paper focuses on a service-centric Vehicle Routing Problem with Crowdshipping (VRPC) under uncertain travel times. Apart from traditional lateness measures, such as on-time arrival probability and expected lateness, we also consider scenarios where customers show exponential impatience towards lateness. We introduce a novel approach to calibrate the disutility, leading to salient managerial implications and probabilistic insights. We develop an exact branch-price-and-cut algorithm for the deterministic VRPC and a route enumeration-based exact algorithm for the non-convex and non-smooth service-centric VRPC. Numerical studies based on existing instances validate the computational efficiency of the developed algorithms and the efficacy of the newly proposed lateness measures in mitigating the risk of late deliveries.

Key words: Vehicle routing problem, crowdshipping, optimization under uncertainty, exact methods

1. Introduction

The surge in last-mile and same-day delivery services has presented significant challenges to retailers striving to offer cost-effective and customer-centric delivery options. The need to reduce transportation costs, a major component of logistics delivery, has driven many companies to explore innovative solutions. One such

* Corresponding author. The first three authors have contributed equally to this work.

solution, embraced by retail giants like Walmart and Amazon, is crowdshipping. This approach involves maintaining a fleet of fully equipped regular vehicles and drivers (RDs) for deliveries and harnessing in-store customers' assistance, referred to as "occasional drivers" or ODs. These ODs are willing to make a single delivery using their vehicles in exchange for modest compensation, provided that the delivery location is not too distant from their intended destinations after completing their shopping. Crowdshipping maximizes the utilization of underutilized private vehicles for deliveries, potentially reducing transportation costs while ensuring timely deliveries.

In real-life scenarios, travel times within the transportation network can be influenced by various factors, such as weather conditions (Hao et al. 2020) and traffic congestion, leading to increased delays for same-day deliveries. Additionally, studies have shown that customers are impatient when deliveries are delayed (Daley 1965, Garnett et al. 2002). Therefore, a solution that enhances service quality with only a minor increase in the travel cost budget can potentially mitigate the uncertainties associated with travel times.

This paper examines a same-day crowdshipping platform with uncertain travel times. We assume that customer requests come with specific service time windows, and we have prior knowledge of the request details and the availability of ODs. RDs must return to the depot (i.e., the store), and ODs must complete their deliveries as soon as they drop off packages to customers and then return to their destinations within specified time windows. Our goal is to minimize the risk of late deliveries. We call the problem the service-centric Vehicle Routing Problem with Crowdshipping (VRPC). Below, we review the literature closely related to the VRPC, followed by an outline of the contributions and organization of the paper.

1.1. Literature Review

In essence, crowdshipping is a unique shared transportation mode originating from the "sharing economy", a topic of considerable interest in various fields. This includes ridesharing (Agatz et al. 2011, 2012, Furuhata et al. 2013), item-sharing (Behrend and Meisel 2018, Behrend et al. 2021), shared delivery locations (Mancini and Gansterer 2021, Kafle et al. 2017), and more. Crowdshipping has experienced explosive growth recently and represents an extraordinarily innovative approach to last-mile, same-day, and urban delivery systems. Empirical analyses have also been conducted in the context of crowdshipping (see Allahviranloo and Baghestani 2019, Punel and Stathopoulos 2017). For a recent overview of future directions and literature in this field, refer to Alnaggar et al. (2021).

Archetti et al. (2016) were the first to introduce the VRP with ODs (VRPOD) as a variant of VRP, where a company has its drivers and utilizes ODs. They proposed a Mixed-Integer Programming (MIP) mathematical model and described variable neighborhood search and tabu search solution algorithms. Subsequently, Macrina et al. 2020, Mancini and Gansterer 2021 further considered time window constraints and introduced transshipment points (also known as transfers and relay points, as detailed in Sampaio et al. 2020, Kafle et al. 2017, Kızıl and Yıldız 2022), which added complexity to the model. Notably, all the aforementioned

models are static. In contrast, Arslan et al. (2019) proposed a dynamic crowdshipping platform in which customer requests and ODs arrive dynamically. They designed an exact rolling horizon approach to solve the matching and routing problems. This dynamic approach generates the corresponding feasible matching and routing sets and adds to the model once a new order or an OD arrives. Dayarian and Savelsbergh (2020) proposed a dynamic same-day crowdshipping model that incorporates the stochastic information regarding the arrival of customer requests and ODs while imposing limitations on the number of RDs to mitigate lateness of requests. They designed static models solved using a neighborhood search method and dynamic models, including myopic and stochastic approaches. In particular, the myopic model assumes no future information can be observed, while the stochastic model assumes a known distribution for the arrival of customer requests and ODs. Furthermore, Archetti et al. (2021) extended this work by considering that customer requests arrive dynamically, and the availability of ODs is known. They also introduced time window violations as a penalty in the objective function and obtained solutions using neighborhood search.

Regarding uncertainty in crowdshipping, recent studies, as described in Arslan et al. 2019, Ulmer 2020, Yıldız 2021, Archetti et al. 2021, Fatehi and Wagner 2022, Torres et al. 2022b, have paid increased attention to the distribution of customer requests and ODs. Additionally, the uncertain availability of ODs has gained considerable attention. Some studies, such as Dahle et al. 2017 and Mousavi et al. 2022, model OD availability as a 0-1 discrete case, where 1 denotes an OD accepting the delivery and 0 denoting rejection. This leads to the proposal of two-stage stochastic models, which can pose computational challenges. The computational study conducted by Dahle et al. (2017) showed that only small instances could be solved using this approach, while Mousavi et al. (2022) used Benders' decomposition and sample average approximation to handle up to 40 nodes. Additionally, Gdowska et al. 2018 and Torres et al. 2022a proposed two-stage stochastic models based on the probability of ODs accepting or rejecting deliveries. In Gdowska et al. (2018), the probability of accepting a delivery is assumed to follow a specific distribution, while Torres et al. (2022a) derived the probability of acceptance based on preferences in economics. Concerning the availability of ODs, there are other simpler settings, such as wage threshold functions (ODs accept deliveries when the compensation exceeds their expected wage level), flexibility restrictions (ODs only accept when the detour for the request is not too large), or scenarios with no restrictions.

Recently, Pugliese et al. (2022b) explored the uncertainty of travel times and introduced a chance-constraint stochastic model that imposes constraints on the probability of exceeding a maximum number of missing deliveries. They presented a solution approach based on Benders' decomposition and constraint-column-generation techniques. In contrast, our problem setting is service-centric. We aim to minimize the risk of late deliveries while adhering to a cost budget constraint.

For VRPs involving ODs and time windows, heuristic algorithms have commonly been employed to achieve rapid computational results. These heuristics include variable neighborhood search and tabu search, as described in Archetti et al. 2016, Kafle et al. 2017, Archetti et al. 2021. Additionally, Dahle et al. (2019)

Table 1 Summary on the closely related literature on the VRPC

Reference	Objective	Lateness measure	Uncertainty	Solution method	Instances solved	Setting
Archetti et al. (2016)	cost	-	-	VNS	100 nodes, 50 ODs	deterministic
Pugliese et al. (2022a)	cost	-	-	GRASP	400 nodes, 60 ODs	deterministic
Macrina et al. (2020)	cost	-	-	VNS	50 nodes	deterministic
Dayarian and Savelsbergh (2020)	lateness of delivery and cost	-	customer requests and supply of ODs	TS	50 nodes, 100 ODs	dynamic, stochastic
Arslan et al. (2019)	cost	-	customer requests and supply of ODs	ERA	100 nodes, 100 ODs	dynamic
Pugliese et al. (2022b)	cost	lateness probability	travel time	BD, CCG	50 nodes, 8 ODs	stochastic
Torres et al. (2022a)	cost	-	availability of ODs	BP	50 nodes, 100 ODs	stochastic
Mousavi et al. (2022)	cost	-	availability of ODs	BC	40 nodes	stochastic
Torres et al. (2022b)	cost	-	destinations of ODs	BP	50 nodes, 100ODs	stochastic
Our work	cost	-	-	BPC, PH	100 nodes, 30 ODs	deterministic
	lateness measure	lateness probability, expected lateness duration, expected exponential lateness disutility, calibrated exponential disutility	travel time	RE, PH	50 nodes, 15 ODs	stochastic

Terminology: VNS: Variable Neighborhood Search; GRASP: Greedy Randomized Adaptive Search Procedure; TS: Tabu Search;
ERA: a dedicated Exact Recursive Algorithm; BD: Benders' Decomposition; CCG: Constraint-Column Generation; BP: Branch and Price;
BC: Branch and Cut; BPC: Branch and Price and Cut; PH: Primal Heuristic; RE: Route Enumeration.

reformulated the traditional MIP model as a network flow problem, which can be efficiently solved using various pruning acceleration strategies. Furthermore, some authors have focused on exact algorithms in recent years. These approaches include Benders' decomposition, as referred to in Mousavi et al. 2022, Pugliese et al. 2022b, Boysen et al. 2022, and branch and price, which combines branch and bound with column generation techniques. For instance, Behrend et al. (2019) proposed an exact solution method based on a set packing model and utilized a labeling algorithm to generate ODs columns. Additionally, Torres et al. (2022b,a) introduced an exact branch and price algorithm for solving this problem.

We summarize the representative literature and position our paper in Table 1. The table includes information on the objective functions, measures to weight the lateness time, factors of uncertainty considered, the solution methods, the maximum size of instances solved by the corresponding solution method, and the problem setting.

1.2. Contributions and Paper Organization

The contributions of our paper can be summarized as follows:

- While most studies primarily focus on cost optimization, specifically minimizing delivery costs with service level constraints, our unique approach sets us apart. We are the first to approach the VRPC from a service-centric perspective, emphasizing the mitigation of late deliveries while adhering to a cost budget constraint.
- We introduce four distinct lateness measures. Initially, we study traditional metrics—on-time arrival probability and expected lateness. Subsequently, we explore scenarios where customers exhibit exponential impatience toward lateness (Daley 1965, Garnett et al. 2002), proposing a truncated exponential disutility criterion. Additionally, we provide the flexibility to adjust the disutility and introduce a novel approach to calibrate the disutility carefully. These last two lateness measures represent new contributions to the literature, and we delve into their managerial implications and probabilistic insights.

- To address the challenge of minimizing delivery costs in the deterministic VRPC, we employ an exact branch-price-and-cut (BPC) algorithm. For the service-centric VRPC, we propose an exact method combining a route enumeration strategy and a column-and-row generation algorithm. Our method applies to various lateness measures. Furthermore, we demonstrate the effectiveness of this method even when dealing with the non-convex and non-smooth nature of a lateness measure.
- For the deterministic VRPC, we report optimal solutions for instances from the literature involving up to 100 customers for the first time. In the service-centric VRPC, we demonstrate that our approach can more effectively mitigate lateness compared to existing measures.

The remainder of this paper is organized as follows: §2 provides a detailed description of the VRPC, along with deterministic and service-centric mathematical formulations. In §3, we introduce alternative lateness measures. The exact methods for solving the deterministic and service-centric VRPCs are presented in §4. §5 showcases the numerical results and conducts a comprehensive analysis. Finally, we conclude in §6.

1.3. Notation

We use $|\mathcal{A}|$ to denote the cardinality of set $\mathcal{A}$. Boldface lowercase letters represent vectors (e.g., $\boldsymbol{x}$). We use tilde signs to represent uncertain parameters (e.g., $\tilde{\xi}$) and $\mathbb{P}$ to denote the distribution. Let $\mathbb{P}[\cdot]$ denote the probability of some event, and $\mathbb{E}_{\mathbb{P}}[\tilde{\xi}]$ represent the expectation of the uncertain parameter $\tilde{\xi}$ under probability distribution $\mathbb{P}$. The set of all one-dimensional random variables is denoted by $\mathbb{V}$. Additionally, we let the indicator function $\mathbb{I}[\cdot] = 1$ if an event happens and $= 0$ otherwise; let $(x)^+ = \max\{x, 0\}$.

2. Problem Description and Mathematical Formulations

In this section, we formally described the VRPC. The section also presents deterministic (§2.1) and service-centric formulations for the problem (§2.2). The problems are formulated based on set-partitioning models widely used to model VRPs. A significant advantage of formulating VRPs using these models is that certain constraints, such as time windows, do not need to be explicitly expressed in the formulation; rather, they are implicit in defining the set of decision variables. Consequently, set partitioning formulations typically provide better lower bounds than those obtained with compact formulations. Nevertheless, they involve many variables, making it impossible to handle them all simultaneously. Fortunately, existing state-of-the-art techniques for VRPs allow us to overcome this drawback (see, for example, Costa et al. 2019).

The VRPC can be described as follows. A digraph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ is given, where $\mathcal{V} = \{0\} \cup \mathcal{N} \cup \mathcal{D}$ is the set of vertices and $\mathcal{A}$ is the set of arcs. Vertex 0 represents the depot, and vertex set $\mathcal{D}$ represents the destinations of a vehicle fleet composed of a set $\mathcal{K}$ of vehicle types. More specifically, each vehicle type $k \in \mathcal{K}$ is associated with a destination depot $d(k) \in \mathcal{D}$. Vertex set $\mathcal{N}$ corresponds to the set of customers. Each customer $i \in \mathcal{N}$ requires a supply of q_i units from the depot 0. It has associated a time window $[e_i, l_i]$, where e_i and l_i represent the earliest and latest time to visit customer i, respectively, and a service time w_i. For each vehicle type $k \in \mathcal{K}$, m_k vehicles are available at the depot, each having a capacity equal to

Q_k and a time window $[e_{d(k)}, l_{d(k)}]$, where $e_{d(k)}$ and $l_{d(k)}$ represent the time when the vehicle associated with destination $d(k)$ becomes available at the main depot (0) and the latest time for the vehicle to reach destination $d(k)$, respectively. For each arc $(i,j) \in \mathcal{A}$ and for each vehicle type $k \in \mathcal{K}$, a routing cost b_{ij}^k and a traveling time t_{ij}^k are given.

A route $R = (0, i_1, \ldots, i_\nu, d(k))$ performed by a vehicle of type $k \in \mathcal{K}$ is a simple path in $\mathcal{G}$ starting from depot 0, visiting customers $\mathcal{N}(R) = \{i_1, \ldots, i_\nu\} \subseteq \mathcal{N}$, with $\nu \geq 1$, and ending at the destination vertex $d(k)$. Route R is feasible if (i) the total demand of visited customers does not exceed the vehicle capacity Q_k; (ii) the vehicle leaves the depot 0 at time $e_{d(k)}$, visits each customer in $\mathcal{N}(R)$ within its time window, and returns to the depot before $l_{d(k)}$; (iii) if the vehicle arrives at $i \in \mathcal{N}(R)$ before e_i, the service is delayed to time e_i. The cost of route R equals the sum of the travel costs of the arc set, $\mathcal{A}(R)$, traversed by route R.

The problem consists of designing a set of feasible routes of minimum total cost such that each customer is visited by exactly one route and the number of routes performed by the vehicles of type k is not greater than m_k, $k \in \mathcal{K}$.

2.1. A Mathematical Formulation for the Deterministic VRPC

Let $\mathcal{R}^k$ be the index set of all feasible routes of vehicle type $k \in \mathcal{K}$ and let $\mathcal{R} = \bigcup_{k \in \mathcal{K}} \mathcal{R}^k$. Each route $r \in \mathcal{R}^k$ is associated with the cost c_r^k computed as $\sum_{(i,j) \in \mathcal{A}(R_r)} b_{ij}^k$. Let a_{ir}^k be a 0-1 coefficient equal to 1 if and only if route $r \in \mathcal{R}^k$ covers or services customer $i \in \mathcal{N}$. Let x_r^k be a binary variable equal to 1 if and only if route $r \in \mathcal{R}^k$ is chosen in the solution. The deterministic VRPC can be formulated as the following deterministic model (DM):

$$\text{(DM)} \qquad \min \sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} c_r^k x_r^k \tag{1}$$

$$\text{s.t.} \sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} a_{ir}^k x_r^k = 1, \quad \forall i \in \mathcal{N}, \tag{2}$$

$$\sum_{r \in \mathcal{R}^k} x_r^k \leq m_k, \quad \forall k \in \mathcal{K}, \tag{3}$$

$$x_r^k \in \{0,1\}, \quad \forall k \in \mathcal{K}, r \in \mathcal{R}^k. \tag{4}$$

Constraints (2) specify that each customer $i \in \mathcal{N}$ must be covered exactly by one route. Constraints (3) impose the upper bound on the number of vehicles of each type that can be used.

Problem VRPC and formulation DM encompass various important VRP variants studied in the literature. However, when modeling crowdshipping deliveries in real-world applications, the set of vehicle types can include RDs, who are company-owned and stationed at the main depot, as well as ODs. More specifically, vehicle type $k = 1$ consists of m_1 vehicles (RDs) with a capacity of Q_1. They terminate their routes at destination $d(1)$, corresponding to the same location as depot 0. Vehicle types from 2 to $|\mathcal{K}|$ represent different vehicles associated with the ODs. In this case, each vehicle type $k \in \mathcal{K} \setminus \{1\}$ has a specific vehicle

capacity Q_k (which can be identical for all ODs), a destination $d(k)$ (usually the home associated with the corresponding OD), and $m_k = 1$. Specifically, the VRPOD studied by Archetti et al. (2016) can be modeled as a special case of the VRPC. Indeed, in addition to the use of RDs and ODs, the set of routes $\mathcal{R}^k$ with $k \in \mathcal{K} \setminus \{1\}$ are simply defined as the set of feasible single-customer route for OD k. Hence, a customer is either served by an RD or by an OD in a single-customer route.

2.2. A Service-Centric Formulation for the VRPC Under Travel Time Uncertainty

We address the VRPC that involves uncertain travel times. The objective is to plan routes for the vehicles so that they arrive at customer locations within their specified time windows as much as possible while ensuring that the total travel cost does not exceed a predetermined budget.

We allow the possibility of a route arriving at a vertex $i \in \mathcal{N} \cup \mathcal{D}$ later than l_i, which means that we replace each set $\mathcal{R}^k$, where $k \in \mathcal{K}$, with a superset containing routes for vehicle k that can violate the deadline constraints. We use $\mathcal{R}^k$ to denote the resulting set of feasible routes for convenience. We assume that $\tilde{t}_{ij}^k$—the consolidated nonnegative random variable associated with the random travel time for traversing arc (i, j) and the random service time at node i—follow the same distribution for $k \in \mathcal{K}$, represented by $\tilde{t}_{ij}$.

Given a route $r \in \mathcal{R}^k$ performed by vehicle of type $k \in \mathcal{K}$ represented by a path $R_r = (i_0 = 0, i_1, \ldots, i_\nu, i_{\nu+1} = d(k))$ and a realization $\boldsymbol{t}$ of travel times $\tilde{\boldsymbol{t}}$, we can determine the service start times $s_i(R_r, \boldsymbol{t})$ of the vertices visited by the route using the recursive expression

$$s_{i_k}(R_r, \boldsymbol{t}) = \max\{s_{i_{k-1}}(R_r, \boldsymbol{t}) + t_{i_{k-1} i_k}, e_{i_k}\}, \quad k = 1, \ldots, \nu + 1, \tag{5}$$

where $s_0(R_r, \boldsymbol{t}) = e_{d(k)}$.

Based on the service start time function, we define the *delay function* at vertex $i \in \mathcal{V}(R_r)$, where $\mathcal{V}(R_r)$ denotes the set of vertices visited by route $r \in \mathcal{R}^k$, as follows

$$\xi_i(R_r, \boldsymbol{t}) = s_i(R_r, \boldsymbol{t}) - l_i. \tag{6}$$

Hence, a late service at vertex i occurs if and only if $\xi_i(R_r, \boldsymbol{t}) > 0$. Because travel times $\tilde{\boldsymbol{t}}$ are uncertain, the delay function $\xi_i(R_r, \tilde{\boldsymbol{t}})$ is also uncertain.

In the deterministic model, the deadlines cannot be violated, i.e., $\xi_i(R_r, \boldsymbol{t}) \leq 0$. To avoid poor service under uncertainty, we pose the following assumption.

Assumption 1 *As a necessary condition for a route $r \in \mathcal{R}$ to be feasible, we require that the deadline constraints be met in expectation, i.e., $\mathbb{E}_{\mathbb{P}}[\xi_i(R_r, \tilde{\boldsymbol{t}})] \leq 0, \ \ \forall i \in \mathcal{V}(R_r)$.*

To further quantify and mitigate the risk associated with the violation of deadlines, we consider a lateness measure computed by a function $\rho(\cdot) : \mathbb{V} \mapsto [0, +\infty]$ that evaluates the risk of lateness at vertex $i \in \mathcal{V}$. Given

a set of routes $\{r_1, r_2, \ldots, r_h\} \subseteq \mathcal{R}$, $h \geq 1$, the sum of the lateness measures over all vertices with deadlines associated with the set of routes can be computed as:

$$\sum_{s=1}^{h} \sum_{i \in \mathcal{V}(R_{r_s})} \rho(\xi_i(R_{r_s}, \tilde{\boldsymbol{t}})) = \sum_{s=1}^{h} \sum_{i \in \mathcal{V}(R_{r_s})} \rho(s_i(R_{r_s}, \tilde{\boldsymbol{t}}) - l_i). \tag{7}$$

Let B be the prescribed budget for the total solution cost. The service-centric VRPC model (SM) is:

$$\text{(SM)} \qquad \min \sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} \left(\sum_{i \in \mathcal{V}(R_r)} \rho(\xi_i(R_r, \tilde{\boldsymbol{t}})) \right) x_r^k \tag{8}$$

$$\text{s.t. } (2), (3), (4), \tag{9}$$

$$\sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} c_r^k x_r^k \leq B. \tag{10}$$

In the above model, the objective function states to minimize the total lateness measures and constraint (10) imposes that the total cost is within the budget B.

As an extension, we note that formulation SM can also solve the service-centric VRPC under demand uncertainty and without time window constraints. In this scenario, let $\tilde{q}_i$ be the nonnegative random variable associated with the random demand of customer $i \in \mathcal{N}$. We define an instance of the service-centric problem where $q_i = 0$ for all $i \in \mathcal{N}$, $\tilde{t}_{ij} = \tilde{q}_j$ for all $(i,j) \in \mathcal{A}$, $e_i = 0$, $l_i = +\infty$ for all $i \in \mathcal{N}$, $e_{d(k)} = 0$, and $l_{d(k)} = Q_k$ for all $k \in \mathcal{K}$. Given a route $r \in \mathcal{R}^k$ performed by a vehicle of type $k \in \mathcal{K}$ represented by a path $R_r = (i_0 = 0, i_1, \ldots, i_\nu, i_{\nu+1} = d(k))$ and a realization $\boldsymbol{t}$ of travel times $\tilde{\boldsymbol{t}}$ (i.e., demands $\tilde{\boldsymbol{q}}$), we can determine the cumulative load along the route by functions $s_{i_k}(R_r, \boldsymbol{t})$. Hence, the *violation capacity function* at vertex $d(k)$, computed as $\xi_{d(k)}(R_r, \boldsymbol{t}) = s_{d(k)}(R_r, \boldsymbol{t}) - l_{d(k)}$, where $l_{d(k)} = Q_k$, represents a violated vehicle capacity if and only if $\xi_{i_k}(R_r, \boldsymbol{t}) > 0$.

3. Measuring the Risk of Lateness

This section discusses alternative lateness measures $\rho(\cdot)$. For notational convenience, we use shorthand $\tilde{\xi}$ interchangeably to represent the uncertain delay $\xi_i(R_r, \tilde{\boldsymbol{t}})$ at some node $i \in \mathcal{V}(R_r)$ for some given route r.

3.1. Common Lateness Measures

We start by considering classical measures from the literature:

- *The lateness probability*: $\rho_P(\tilde{\xi}) = \mathbb{P}\left[\tilde{\xi} > 0\right]$. In the broad literature in management science, the probability measure dates back to Charnes and Cooper (1959). In the VRP literature, one may limit the lateness probability within some tolerable bound as constraints (see, e.g., Laporte et al. 1992, Errico et al. 2018) or minimize the lateness probability from a service-centric perspective (see, e.g., Kenyon and Morton 2003, Adulyasak and Jaillet 2015). In VRPOD, Pugliese et al. (2022b) developed a chance-constrained model to minimize operational costs while bounding the probability of on-time delivery. Despite its popularity, this criterion fails to account for the duration of lateness, which is an important aspect of the service quality.

For instance, a delivery route with a probability 5% of being late for 30 minutes is deemed indifferent to another with the same probability but being late for 1 minute. In reality, the latter certainly offers a better service.

- *The expected lateness*: $\rho_T(\tilde{\xi}) = \mathbb{E}\big[(\tilde{\xi})^+\big]$. This criterion penalizes the duration of lateness. In the VRP literature, the expected lateness is usually considered as a penalty cost, which, together with other operational costs, are minimized (see, e.g., Laporte et al. 1992, Taş et al. 2014). In VRPOD, one may simply regard the violation of lateness under deterministic travel times as a large penalty in the objective function (Archetti et al. 2021, Dahle et al. 2017, Kafle et al. 2017, Zhen et al. 2021). To our knowledge, this paper is the first to consider expected lateness as a lateness measure under uncertain travel times. Despite its ability to account for the duration of lateness, it, in turn, offers no probability insight to the logistics operator.

3.2. Exponential Disutility

In service-centric operations, customer dissatisfaction is the ultimate concern. In this context, the expected lateness criterion implicitly assumes that the customer *dissatisfaction* (or *disutility*) concerning the lateness grows linearly with the increase in its duration. However, in reality, it may not be so. Inspired by empirical studies (see, e.g., Daley 1965, Garnett et al. 2002) showing that customers exhibit exponential impatience towards lateness, we next consider a truncated exponential disutility function

$$\phi_E(\xi) = 0\mathbb{I}[\xi \leq 0] + \exp(\xi)\mathbb{I}[\xi > 0]. \tag{11}$$

For on-time or early delivery cases ($\xi \leq 0$), the disutility is 0, and for late delivery cases, the disutility is exponential in the delay, $\xi > 0$. Thereby, we propose the lateness measure termed *expected exponential lateness*:

$$\rho_E(\tilde{\xi}) = \mathbb{E}\big[\phi_E(\tilde{\xi})\big].$$

Apart from the argued rationality of considering the exponential disutility, this criterion offers the following probability insights.

Proposition 1 *For any uncertain delay $\tilde{\xi} \in \mathbb{V}$ and any duration $\theta > 0$ of lateness, the expected exponential lateness criterion offers the following probability envelope bound:*

$$\mathbb{P}\big[\tilde{\xi} > \theta\big] \leq \rho_E(\tilde{\xi}) \exp(-\theta). \tag{12}$$

Proof: For all $\theta > 0$, we have

$$\begin{aligned}\mathbb{P}\big[\tilde{\xi} > \theta\big] &\leq \mathbb{P}\big[\phi_E(\tilde{\xi}) \geq \phi_E(\theta)\big] \\ &\leq \frac{\mathbb{E}[\phi_E(\tilde{\xi})]}{\phi_E(\theta)} \\ &= \rho_E(\tilde{\xi}) \exp(-\theta).\end{aligned}$$

The first inequality holds because $\phi_E(\cdot)$ is non-decreasing. The second inequality follows from Markov's inequality and the fact that $\phi_E(\theta) > 0$. □

This probability envelope (12) is essentially an *a posteriori* bound because it depends on the optimized value of $\rho_E(\tilde{\xi})$; recall that $\tilde{\xi}$ is a function of routing decision R_r, where $r \in \mathcal{R}^k, k \in \mathcal{K}$. It seems to provide promising managerial justification for the expected exponential criterion. However, we observe that when the value of $\rho_E(\tilde{\xi})$ is large and for some small duration θ, the probability upper bound, $\rho_E(\tilde{\xi})\exp(-\theta)$, would easily be greater than one. That is, the probability envelope may offer a trivial bound because the probability of an event is always upper bounded by one.

3.3. Calibrated Expected Disutility

To circumvent the above issue, we propose calibrating the exponential disutility, and we show that this approach has salient managerial implications. Specifically, we introduce a parameter to scale the exponential disutility and calibrate it by ensuring a nontrivial probability envelope *a priori*. We implement the idea by considering the following constrained optimization problem (13).

Definition 1 *(Optimal Calibration Parameter, OCP) Given a random delay $\tilde{\xi} \in \mathbb{V}$ and a probability bound $\gamma \in [0,1]$, the OCP is defined as:*

$$\rho_{OCP}\big(\tilde{\xi}\big) = \inf\left\{\alpha > 0 \,\middle|\, \mathbb{E}\big[\phi_E(\tilde{\xi}/\alpha)\big] \le \gamma\right\}. \tag{13}$$

According to the definition, we guarantee the following nontrivial probability envelope.

Theorem 1 *For uncertain delay $\tilde{\xi} \in \mathbb{V}$ and any duration $\theta > 0$ of lateness, the OCP approach offers the following probability envelope bound:*

$$\mathbb{P}\big[\tilde{\xi} > \theta\big] \le \gamma \exp\big(-\theta/\rho_{OCP}\big(\tilde{\xi}\big)\big). \tag{14}$$

Proof: Putting $\alpha^\star \triangleq \rho_{OCP}(\tilde{\xi})$, we have for all $\theta > 0$,

$$\begin{aligned}
\mathbb{P}\big[\tilde{\xi} > \theta\big] &= \mathbb{P}\big[\tilde{\xi}/\alpha^\star > \theta/\alpha^\star\big] \\
&\le \mathbb{P}\big[\phi_E(\tilde{\xi}/\alpha^\star) \ge \phi_E(\theta/\alpha^\star)\big] \\
&\le \frac{\mathbb{E}[\phi_E(\tilde{\xi}/\alpha^\star)]}{\phi_E(\theta/\alpha^\star)} \\
&\le \frac{\gamma}{\phi_E(\theta/\alpha^\star)} \\
&= \gamma \exp\big(-\theta/\alpha^\star\big).
\end{aligned}$$

The first inequality holds because $\phi_E(\cdot)$ is non-decreasing, the second one owes to Markov's inequality and the fact that $\phi_E(\cdot)$ is strictly positive in $\mathbb{R}_+$. The last inequality follows from the definition (13). □

From this theorem, we show that the probability envelope provided by the OCP is now nontrivial because the probability bound is always less than one, implied by the fact that $\gamma \in [0,1]$ and $\exp\big(-\theta/\rho_{OCP}\big(\tilde{\xi}\big)\big) \in (0,1)$. In addition to the probability envelope, our proposed lateness measure, OCP, guarantees the following desired properties.

Proposition 2 *For any $\tilde{\xi}, \tilde{\xi}_1$ and $\tilde{\xi}_2 \in \mathbb{V}$, we have:*

1. *Monotonicity: If $\mathbb{P}[\tilde{\xi}_1 \geq \tilde{\xi}_2] = 1$, then $\rho_{OCP}(\tilde{\xi}_1) \geq \rho_{OCP}(\tilde{\xi}_2)$.*
2. *Satisficing: If $\mathbb{P}[\tilde{\xi} < 0] = 1$, then $\rho_{OCP}(\tilde{\xi}) = 0$.*
3. *Positive homogeneity: $\rho_{OCP}(k\tilde{\xi}) = k\rho_{OCP}(\tilde{\xi})$ for all $k > 0$.*

Proof:

1. Monotonicity: If $\mathbb{P}[\tilde{\xi}_1 \geq \tilde{\xi}_2] = 1$, then $\mathbb{P}[\tilde{\xi}_1/\alpha \geq \tilde{\xi}_2/\alpha] = 1, \alpha > 0$. Because $\phi_E(\cdot)$ is non-decreasing, we have $\mathbb{E}[\phi_E(\tilde{\xi}_1/\alpha)] \geq \mathbb{E}[\phi_E(\tilde{\xi}_2/\alpha)]$. Hence, $\mathcal{Y}_1 \subseteq \mathcal{Y}_2$ where $\mathcal{Y}_i \triangleq \{\alpha > 0 | \mathbb{E}[\phi_E(\tilde{\xi}_i/\alpha)] \leq \gamma\}, i = 1, 2$. Finally, we conclude that $\rho_{OCP}(\tilde{\xi}_1) = \inf \mathcal{Y}_1 \geq \inf \mathcal{Y}_2 = \rho_{OCP}(\tilde{\xi}_2)$.
2. Satisficing: If $\mathbb{P}[\tilde{\xi} < 0] = 1$, then as $\alpha \to 0^+$, we have $\mathbb{P}[\tilde{\xi}/\alpha \to -\infty] = 1$, and $\mathbb{P}[\phi_E(\tilde{\xi}/\alpha) = 0] = 1$, which implies that $\mathbb{E}[\phi_E(\tilde{\xi}/\alpha)] \leq \gamma$, and that $\alpha = 0^+$ satisfies the constraint in (13). In addition, $\alpha > 0$; taking the infimum, we obtain $\rho_{OCP}(\tilde{\xi}) = 0$.
3. Positive homogeneity: For all $k > 0$, we have

$$\begin{aligned}
\rho_{OCP}(k\tilde{\xi}) &= \inf\left\{\alpha > 0 \;\middle|\; \mathbb{E}\left[\phi_E(k\tilde{\xi}/\alpha)\right] \leq \gamma\right\} \\
&= \inf\left\{(k\alpha) > 0 \;\middle|\; \mathbb{E}\left[\phi_E(k\tilde{\xi}/(k\alpha))\right] \leq \gamma\right\} \\
&= k \inf\left\{\alpha > 0 \;\middle|\; \mathbb{E}\left[\phi_E(\tilde{\xi}/\alpha)\right] \leq \gamma\right\} \\
&= k\rho_{OCP}(\tilde{\xi}),
\end{aligned}$$

where the second equality holds by substituting α by $k\alpha$ without loss of generality. □

Monotonicity indicates that a route with a lesser delay is always preferred. Satisficing says that if a route never violates the deadline, the OCP equals zero, its best possible value. Positive homogeneity implies that if a delay is scaled by k times, the value of OCP is also scaled by k times.

3.3.1. Comparing with the Riskiness Index Our OCP is not unlike the Riskiness Index (RI) proposed by Aumann and Serrano (2008), widely used to measure the risk of violating a prescribed target. Its successful applications range from behavior economics (Aumann and Serrano 2008) to project selection (Hall et al. 2015), to routing optimization (Jaillet et al. 2016), and healthcare bed management (Xie et al. 2023).

Definition 2 *(Riskiness Index, RI) Given a random delay $\tilde{\xi} \in \mathbb{V}$, the RI is defined as*

$$\rho_{RI}(\tilde{\xi}) = \inf\left\{\alpha > 0 \;\middle|\; \mathbb{E}\left[\exp(\tilde{\xi}/\alpha)\right] \leq 1\right\}. \tag{15}$$

Comparing the definition of our OCP (13) with RI (15), we observe two differences. First, the OCP assumes a zero disutility for on-time or early delivery ($\xi < 0$), whereas the RI still poses an exponential disutility even if no lateness occurs; we opine that our OCP is more reasonable in the VRPC context. Second, our OCP allows the expected disutility to be upper bounded by $\gamma \in [0, 1]$, providing more flexibility than imposing it to be one, as in the RI.

In the literature, to justify the RI as a plausible decision criterion, it is shown that the RI offers the following probability envelope concerning the lateness risk.

Proposition 3 (see, e.g., Hall et al. (2015) and Jaillet et al. (2016)) *For uncertain delay $\tilde{\xi} \in \mathbb{V}$ and any duration $\theta > 0$, the RI offers the following probability envelope bound:*

$$\mathbb{P}\big[\tilde{\xi} > \theta\big] \leq \exp\big(-\theta/\rho_{RI}(\tilde{\xi})\big). \tag{16}$$

Observing that the two probability bounds of OCP and RI (see (14) and (16) respectively) are similar in forms, it is natural to question which is better. To be fair, we fix $\gamma = 1$ for the OCP. We next show that our probability envelope is indeed tighter.

Theorem 2 *Given a random delay $\tilde{\xi} \in \mathbb{V}$ and any duration $\theta > 0$, the probability envelope of OCP is tighter than that of the RI, i.e.,* $\exp(-\theta/\rho_{OCP}(\tilde{\xi})) \leq \exp(-\theta/\rho_{RI}(\tilde{\xi}))$.

Proof: Putting $\phi_{RI}(\xi/\alpha) = \exp(\xi/\alpha)$, when $\xi/\alpha \in [-\infty, 0]$, $\phi_{RI}(\xi/\alpha) \in (0, 1]$, while $\phi_E(\xi/\alpha) = 0$. When $\tilde{\xi}/\alpha \in (0, \infty)$, $\phi_{RI}(\xi/\alpha) = \phi_E(\xi/\alpha) = \exp(\xi/\alpha) \in [1, \infty)$. Hence we have $\phi_{RI}(\xi/\alpha) \geq \phi_E(\xi/\alpha)$; then $\mathcal{Y}_{RI} \subseteq \mathcal{Y}_{OCP}$ where $\mathcal{Y}_{RI} \triangleq \{\alpha > 0 | \mathbb{E}[\phi_{RI}(\tilde{\xi}/\alpha)] \leq 1\}$ and $\mathcal{Y}_{OCP} \triangleq \{\alpha > 0 | \mathbb{E}[\phi_E(\tilde{\xi}/\alpha)] \leq 1\}$. Finally, we conclude that $\rho_{RI}(\xi) = \inf \mathcal{Y}_{RI} \geq \inf \mathcal{Y}_{OCP} = \rho_{OCP}(\xi)$. According to the probability envelope of Theorem 1, putting $\alpha^*_{RI} = \rho_{RI}(\tilde{\xi})$ and $\alpha^*_{OCP} = \rho_{OCP}(\tilde{\xi})$, we have

$$\begin{aligned}
\mathbb{P}\big[\tilde{\xi} > \theta\big] &\leq \frac{1}{\exp\big(\theta/\alpha^*_{OCP}\big)} \\
&\leq \frac{1}{\exp\big(\theta/\alpha^*_{RI}\big)} \\
&= \frac{1}{\exp\big(\theta/\alpha^*_{RI}\big)}.
\end{aligned}$$

This completes the proof. □

In the following, we present a concrete example.

Example 1 *Let us consider a random delay $\tilde{\xi}$ following a two-point distribution: $\mathbb{P}[\tilde{\xi} = -10] = 50\%$ and $\mathbb{P}[\tilde{\xi} = 5] = 50\%$. Using Algorithm 1 to be presented later, we calculate that $\rho_{OCP}\big(\tilde{\xi}\big) \approx 7.22$ (where $\gamma = 1$ in the OCP) and $\rho_{RI}\big(\tilde{\xi}\big) \approx 10.39$. The induced probability envelopes, calculated using (14) and (16), respectively, are shown in Figure 1.*

Through Theorem 2 and Example 1, we demonstrate the advantage of our OCP compared to the well-established RI. As discussed earlier, this is a benefit of setting the disutility to zero in on-time or early delivery cases. However, it causes an unwanted issue—the disutility function (11) is now non-convex and non-smooth, which may plague the computational aspect of our optimization model. Fortunately, our route enumeration algorithm is compatible with this format, as shown in §4.2.

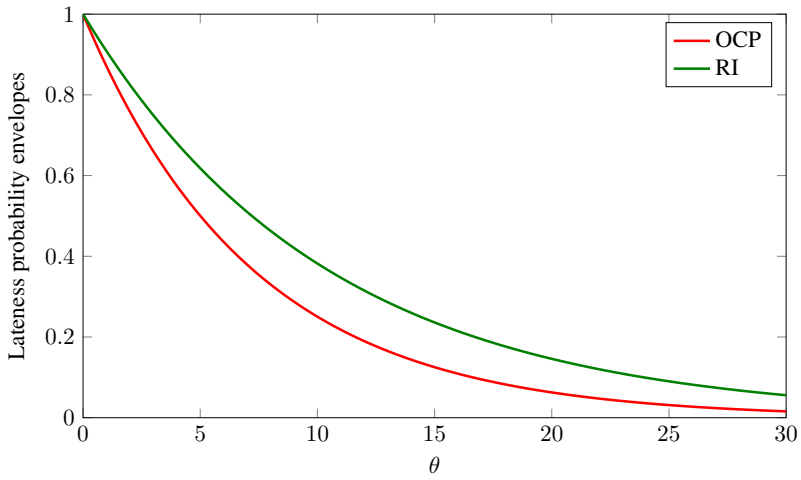


Figure 1 Lateness probability envelopes

3.3.2. Computing the Optimal Calibration Parameter We aim to develop an efficient algorithm to compute the OCP, $\rho_{OCP}(\xi_i(R_r, \tilde{\boldsymbol{t}}))$, at some node $i \in \mathcal{V}(R_r)$ along some route r, given some distribution $\mathbb{P}$ governing the uncertain travel times $\tilde{\boldsymbol{t}}$.

In the real world, we cannot observe the true distribution $\mathbb{P}$ of travel times $\tilde{\boldsymbol{t}}$. Instead, we may have access to a set of historical data, $\{\boldsymbol{t}^1, \boldsymbol{t}^2, \cdots, \boldsymbol{t}^N\}$. It is common to use the data-driven empirical distribution as a proxy to the true but unknown distribution $\mathbb{P}$. In this paper, we also adopt this assumption.

Assumption 2 *The distribution $\mathbb{P}$ of travel times $\tilde{\boldsymbol{t}}$ is given as the empirical distribution defined by $\mathbb{P}[\tilde{\boldsymbol{t}} = \boldsymbol{t}^\omega] = 1/N$ for all $\omega \in \Omega$, where $\Omega = \{1, 2, \cdots, N\}$.*

As such, the OCP (13) can be rewritten as

$$\rho_{OCP}(\tilde{\xi}) = \inf\{\alpha > 0 \mid \psi(\alpha) \le \gamma\}, \tag{17}$$

where $\psi(\alpha) = N^{-1} \sum_{\omega \in \Omega} \phi_E(\xi^\omega/\alpha)$, and $\xi^\omega = \xi(R_r, \boldsymbol{t}^\omega)$ is the delay function (6) under sample $\omega \in \Omega$. To compute the OCP, we first need a convexity result.

Proposition 4 *The function $\psi(\alpha)$ is convex in $\alpha > 0$.*

Proof: For all $\alpha > 0$, by definition (11) of $\phi_E(\cdot)$, we have $\psi(\alpha) = \begin{cases} 0, & \xi \le 0 \\ N^{-1} \sum_{\omega \in \Omega} \exp(\xi^\omega/\alpha), & \xi > 0 \end{cases}$. Taking the first derivative, we have $\mathrm{d}\psi(\alpha)/\mathrm{d}\alpha = \begin{cases} 0, & \xi \le 0 \\ N^{-1} \sum_{\omega \in \Omega} -\frac{\xi^\omega}{\alpha^2} \exp(\xi^\omega/\alpha), & \xi > 0 \end{cases}$; here, when $\xi = 0$, we have $\lim_{\xi \to 0^-} \mathrm{d}\psi(\alpha)/\mathrm{d}\alpha = 0$ and $\lim_{\xi \to 0^+} \mathrm{d}\psi(\alpha)/\mathrm{d}\alpha = 0$, i.e., the first derivative at $\xi = 0$ is continuous. Then, we take the second derivative and get $\mathrm{d}^2\psi(\alpha)/\mathrm{d}\alpha^2 = \begin{cases} 0, & \xi \le 0 \\ N^{-1}\big(\sum_{\omega \in \Omega} \frac{(\xi^\omega)^2}{\alpha^4} \exp(\xi^\omega/\alpha) + \frac{2\xi^\omega}{\alpha^3} \exp(\xi^\omega/\alpha)\big), & \xi > 0 \end{cases}$; here, when $\xi = 0$, $\lim_{\xi \to 0^-} \mathrm{d}^2\psi(\alpha)/\mathrm{d}\alpha^2 = 0$

Algorithm 1 Bisection for computing the OCP

Input: Function $\psi(\alpha)$, parameter γ

Output: Optimal OCP value $\alpha^\star$

Let $\overline{\alpha} \leftarrow M$ and $\underline{\alpha} \leftarrow \varepsilon$ ▷ M is big number and ε is small positive number
if $\psi(\underline{\alpha}) \leq \gamma$ **then**
  return 0
if $\psi(\overline{\alpha}) > \gamma$ **then** ▷ This implies infeasibility
  return M
Initialize $\alpha^\star \leftarrow M$ ▷ We use $\alpha^\star$ to restore OCP
while $\overline{\alpha} - \underline{\alpha} > \epsilon$ **do** ▷ If yes, we find some feasible $\alpha \in [\underline{\alpha}, \overline{\alpha}]$; ϵ is tolerable error bound
  Let $\alpha \leftarrow (\underline{\alpha} + \overline{\alpha})/2$
  if $\psi(\alpha) \leq \gamma$ **then** ▷ Such α is feasible
    Let $\overline{\alpha} \leftarrow \alpha$ and $\alpha^\star \leftarrow \alpha$
  else
    Let $\underline{\alpha} \leftarrow \alpha$
return $\alpha^\star$

and $\lim_{\xi \to 0^+} \mathrm{d}^2\psi(\alpha)/\mathrm{d}\alpha^2 = 0$. Consequently, we observe that $\mathrm{d}^2\psi(\alpha)/\mathrm{d}\alpha^2 \geq 0$ and function $\psi(\alpha)$ is convex in $\alpha > 0$. □

By the convexity property, we can compute the OCP by a bisection search; see Algorithm 1. We must find an OCP α that satisfies $\psi(\alpha) \leq \gamma$. Specifically, because each iteration cuts the search space, $[\varepsilon, M]$, by half, it takes at most $\lceil \log_2((M-\varepsilon)/\epsilon) \rceil$ iterations to terminate. Note that the exponential function can cause numerical overflow (Mihatsch and Neuneier 2002, Gosavi et al. 2014); we use a preprocessing technique presented in §EC.1 to address this issue.

4. Solving formulations DM and SM

In this section, we present exact methods for solving formulations DM (§4.1) and SM (§4.2). The first exact method employs a BPC approach, while the second exact method, used to solve formulation SM, relies on a route enumeration-based approach. Additionally, we design primal heuristics combined with a variable neighborhood search (VNS) algorithm to solve formulations DM and SM (§4.3). Notably, the lower bounding procedure employed in solving formulation DM is effectively utilized in solving formulation SM.

4.1. Solving Formulation DM

To solve formulation DM, we employ a BPC algorithm based on state-of-the-art techniques for solving VRPs (see, for example, Costa et al. 2019).

The linear programming (LP) relaxation of formulation DM can be solved through column generation. This involves iteratively solving a restricted master problem (RMP) with a subset of variables from a large set of explanatory variables associated with $\mathcal{R}$. A pricing problem is also solved iteratively, determining whether a variable associated with route $r \in \mathcal{R}$ should be added to the RMP to enhance its current solution. The column generation technique and corresponding solution approaches have been thoroughly described by Lübbecke and Desrosiers (2005) and Desaulniers et al. (2006).

The pricing problem involves determining the most negative reduced cost route, corresponding to an elementary shortest path problem with resource constraints (ESPPRC) (Irnich and Desaulniers 2005). State-of-the-art techniques in this context rely on relaxations of the ESPPRC, such as the shortest path problem with resource constraints (SPPRC) (Irnich and Desaulniers 2005) and the ng-path (ng-route) relaxation (Baldacci et al. 2011b), the latter of which has been proven to be highly effective in practice. We tackled the pricing problem in our implementation using the ng-path relaxation. For each customer $i \in \mathcal{N}$, let $\mathcal{N}_i$ denote the *neighborhoods* of i. An ng-path is not necessarily elementary; it can contain a cycle starting and ending at vertex j if and only if a vertex i exists in this cycle such that $j \notin \mathcal{N}_i$. Further details can be found in Baldacci et al. (2011b). Under the ng-path relaxation, coefficients a_{ir}^k of formulation DM are nonnegative integer coefficients.

To enhance the LP relaxation of formulation DM, we utilized a variation of the Subset Rows Cuts (SRCs), which are non-robust cuts initially introduced by Jepsen et al. (2008). To mitigate the impact of SRCs on labeling algorithms, Pecin et al. (2017a) introduced a milder version known as limited-memory SRCs (lm-SRCs). An lm-SRC is defined by a set $\mathcal{C}$, a multiplier β, and a memory set $\mathcal{M}$ ($\mathcal{C} \subseteq \mathcal{M} \subseteq \mathcal{N}$), and it can be expressed as follows:

$$\sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} \alpha(\mathcal{C}, \mathcal{M}, \beta, r) x_r^k \leq \lfloor \beta \left| \mathcal{C} \right| \rfloor , \tag{18}$$

where the function $\alpha(.)$ calculates the route coefficients (for separation, refer to Pecin et al. 2017a). In our approach, the multiplier β was set to 0.5, and $|\mathcal{C}| = 3$.

At various nodes of the BPC enumeration tree, the lower bound is calculated using a column-and-row generation approach, strengthened by lm-SRCs. Two branching rules are employed as the branching strategy. The first rule branches on cutsets, while the second one branches on the arc variables associated with the graph $\mathcal{G}$ using a classical branching-on-arc strategy commonly used for VRPs. Importantly, both branching rules maintain the pricing problem structure. The lowest-first strategy is adopted for node selection, where the node with the minimum lower bound is initially selected and explored. For further details on the branching strategy, readers can refer to Costa et al. (2019).

4.2. Solving Formulation SM

We start by presenting the following proposition.

Proposition 5 *Consider a route $r \in \mathcal{R}$, for each $i \in \mathcal{V}(R_r)$, we have:*

$$s_i(R_r, \mathbb{E}_{\mathbb{P}}[\tilde{\boldsymbol{t}}]) \leq \mathbb{E}_{\mathbb{P}}[s_i(R_r, \tilde{\boldsymbol{t}})] \leq l_i. \tag{19}$$

Proof. We have:

$$\begin{aligned} & \mathbb{E}_{\mathbb{P}}[\xi_i(R_r, \tilde{\boldsymbol{t}})] \leq 0 \\ \Rightarrow \quad & \mathbb{E}_{\mathbb{P}}[s_i(R_r, \tilde{\boldsymbol{t}}) - l_i] \leq 0 \\ \Rightarrow \quad & \mathbb{E}_{\mathbb{P}}[s_i(R_r, \tilde{\boldsymbol{t}})] \leq l_i \\ \Rightarrow \quad & s_i(R_r, \mathbb{E}_{\mathbb{P}}[\tilde{\boldsymbol{t}}]) \leq \mathbb{E}_{\mathbb{P}}[s_i(R_r, \tilde{\boldsymbol{t}})] \leq l_i. \end{aligned} \tag{20}$$

The four implications result from the (i) Assumption 1, (ii) definition of the delay function (6), (iii) translation invariance of $\mathbb{E}_{\mathbb{P}}$, and (iv) property that the service start time function $s_i(R_r, \tilde{\boldsymbol{t}})$ is convex piecewise affine in $\tilde{\boldsymbol{t}}$ (see Zhang et al. (2021)) and Jensen's inequality. □

The above proposition allows to reduce the cardinality of the set of routes $\mathcal{R}$. Specifically, let $\boldsymbol{\mu} = \mathbb{E}_{\mathbb{P}}(\tilde{\boldsymbol{t}})$ be the expectation of $\tilde{\boldsymbol{t}}$ under probability distribution $\mathbb{P}$ representing the mean travel times. Given a route $r \in \mathcal{R}^k$ and travel times $\boldsymbol{\mu}$, the service start times of path $R_r = (i_0 = 0, i_1, i_2, \ldots, i_{\nu-1}, i_\nu, i_{\nu+1} = d(k))$ under mean travel times $\boldsymbol{\mu}$ can be computed as $s_{i_k}(R_r, \boldsymbol{\mu}) = \max\{s_{i_{k-1}}(R_r, \boldsymbol{\mu}) + \mu_{i_{k-1}i_k}, e_{i_k}\}, k = 1, \ldots, \nu+1$. Let $\overline{\mathcal{R}}^k \subseteq \mathcal{R}^k$ be the set of routes in $\mathcal{R}^k$ satisfying the time window constraints based on the mean travel times $\boldsymbol{\mu}$, i.e., $\overline{\mathcal{R}}^k = \{r \in \mathcal{R}^k : e_{\pi(i)} \leq s_i(R_r, \boldsymbol{\mu}) \leq l_i, \forall i \in \mathcal{V}(R_r)\}$, where $\pi(i) = i$ if $i \in \mathcal{N} \cup \mathcal{D}$, otherwise $(i = 0)$ $\pi(i) = d(k)$. Let $\overline{\mathcal{R}} = \bigcup_{k \in \mathcal{K}} \overline{\mathcal{R}}^k$.

The route set $\mathcal{R}$ satisfies the capacity and hard earliest service time window constraints. However, a route $r \in \mathcal{R}$ might not satisfy the deadlines, l_i, of the time windows constraints under the mean travel times, $\boldsymbol{\mu}$. Hence, the following proposition holds.

Proposition 6 $\overline{\mathcal{R}} \subseteq \mathcal{R}$.

Let $\overline{\boldsymbol{u}} \in \mathbb{R}^{|\mathcal{N}|}$ and $\overline{\boldsymbol{v}} \in \mathbb{R}_-^{|\mathcal{K}|}$ be the vectors of the dual variables associated with the LP relaxation of model DM, where variables $\overline{u}_i \in \mathbb{R}$, $i \in \mathcal{N}$, are associated with constraints (2) whereas variables $\overline{v}_k \leq 0$, $k \in \mathcal{K}$, are associated with constraint (3). The following theorem shows that the set of routes $\overline{\mathcal{R}}$ can be further reduced.

Theorem 3 *Let $(\overline{\boldsymbol{u}}, \overline{\boldsymbol{v}})$ be a feasible solution of the dual of the LP relaxation of model DM of cost $\overline{z}$. Define the reduced cost of a route $r \in \overline{\mathcal{R}}^k$, $k \in \mathcal{K}$, with respect to the dual solution $(\overline{\boldsymbol{u}}, \overline{\boldsymbol{v}})$ as $\overline{c}_r = c_r^k - \sum_{i \in \mathcal{N}} a_{ir}^k \overline{u}_i - \overline{v}_k$. Any optimal solution $\overline{\boldsymbol{x}}$ of SM, due to the budget constraint (10), cannot contain any route $r \in \overline{\mathcal{R}}^k$ such that $\overline{c}_r > B - \overline{z}$.*

Proof. Let $\mathcal{B} = \{r \in \overline{\mathcal{R}} : \overline{x}_r^k = 1, k \in \mathcal{K}\}$ be the index set of the routes of solution $\overline{\boldsymbol{x}}$. We have $\sum_{r \in \mathcal{B}} c_r^{\pi(r)} \overline{x}_r^{\pi(r)} \leq B$, where $\pi(r)$ is the vehicle associated with route r and, by contradiction, suppose that there exists $r' \in \mathcal{B}$ such that $\overline{c}_{r'} > B - \overline{z}$. We have

$$\sum_{r \in \mathcal{B}} \overline{c}_r = \sum_{r \in \mathcal{B}} c_r^{\pi(r)} - \sum_{r \in \mathcal{B}} \sum_{i \in \mathcal{N}} a_{ir}^{\pi(r)} \overline{u}_i - \sum_{r \in \mathcal{B}} \overline{v}_{\pi(r)}. \tag{21}$$

Because $\mathcal{B}$ represents a feasible solution of model SM, we have:

$$\sum_{r\in\mathcal{B}}\sum_{i\in\mathcal{N}} a_{ir}^{\pi(r)}\overline{u}_i + \sum_{r\in\mathcal{B}}\overline{v}_{\pi(r)} = \sum_{i\in\mathcal{N}}\overline{u}_i + \sum_{r\in\mathcal{B}}\overline{v}_{\pi(r)} \geq \sum_{i\in\mathcal{N}}\overline{u}_i + \sum_{r\in\mathcal{K}} m_k\overline{v}_k, \tag{22}$$

where the last inequality holds since $|r\in\mathcal{B}:\pi(r)=k|\leq m_k$, $k\in\mathcal{K}$, and $\overline{v}_k\leq 0$, $k\in\mathcal{K}$. The last term of the above inequality equals the cost of the dual solution, $\overline{z}$, and since $\sum_{k\in\mathcal{K}}\sum_{r\in\mathcal{R}_k} c_r^k\overline{x}_r^k = \sum_{r\in\mathcal{B}} c_r^{\pi(r)}$ we obtain

$$\sum_{k\in\mathcal{K}}\sum_{r\in\mathcal{R}_k} c_r^k\overline{x}_r^k \geq \overline{z} + \sum_{r\in\mathcal{B}}\overline{c}_r. \tag{23}$$

Because solution $(\overline{\boldsymbol{u}},\overline{\boldsymbol{v}})$ is a feasible dual solution, we have $\overline{c}_r\geq 0$, $\forall r\in\mathcal{B}$, and from inequality (23), we derive

$$\begin{aligned}\sum_{k\in\mathcal{K}}\sum_{r\in\mathcal{R}_k} c_r^k\overline{x}_r^k \geq \overline{z} + \sum_{r\in\mathcal{B}}\overline{c}_r \geq \overline{z} + \overline{c}_{r'} > \\ \overline{z} + B - \overline{z} = B,\end{aligned} \tag{24}$$

which contradicts the fact that $\overline{\boldsymbol{x}}$ satisfies the budget constraint (10). □

Based on the above theorem, given a feasible dual solution $(\overline{\boldsymbol{u}},\overline{\boldsymbol{v}})$ of the LP relaxation of model DM of cost $\overline{z}$, the set of routes $\overline{\mathcal{R}}$ can be replaced by the reduced set of routes $\{r\in\overline{\mathcal{R}}:\overline{c}_r\leq B-\overline{z}\}\subseteq\overline{\mathcal{R}}$. Hereafter, for the sake of the notation, we denote with $\overline{\mathcal{R}}$ the final reduced set of routes.

4.2.1. Route-Enumeration Based Method To derive the reduced set of routes $\overline{\mathcal{R}}$, we employ the dual solution computed by the column-and-row generation procedure at the root node of the BPC algorithm used to solve formulation DM. Specifically, let $\mathcal{S}$ be the set of lm-SRCs (18) associated with the root node lower bound, and let variables $\eta_c\leq 0$, $\forall c\in\mathcal{S}$, represent the corresponding dual variables and set $\mathcal{C}_c$ denotes the corresponding customer base set. Here, $\overline{z}$ signifies the computed lower bound, and $(\overline{\boldsymbol{u}},\overline{\boldsymbol{v}},\overline{\boldsymbol{\eta}})$ represents the associated dual solution.

The reduced cost of a route $r\in\overline{\mathcal{R}}^k$, $k\in\mathcal{K}$, with respect to the dual solution $(\overline{\boldsymbol{u}},\overline{\boldsymbol{v}},\overline{\boldsymbol{\eta}})$ can be computed as

$$\overline{c}_r = c_r^k - \sum_{i\in\mathcal{N}} a_{ir}^k\overline{u}_i - \overline{v}_k - \sum_{c\in\mathcal{S}}\alpha(\mathcal{C}_c,\mathcal{M},\beta,r)\eta_c. \tag{25}$$

It can be shown that Theorem 3 applies also for dual solution $(\overline{\boldsymbol{u}},\overline{\boldsymbol{v}},\overline{\boldsymbol{\eta}})$.

The exact route-enumeration-based method performs the following main steps:

(i) Generate the largest route set, $\overline{\mathcal{R}} = \{r\in\mathcal{R}: \overline{c}_r\leq B-\overline{z}$ and r satisfies the time window constraints based on the mean travel times $\boldsymbol{\mu}\}$ such that $|\overline{\mathcal{R}}|\leq\Delta_{\max}$, where $\Delta_{\max}$ is a user-defined parameter.

(ii) Solve formulation SM, replacing the set of routes $\mathcal{R}$ with $\overline{\mathcal{R}}$, using a general-purpose MIP solver. Let z^* be the resulting optimal solution cost.

(iii) If $|\overline{\mathcal{R}}|\leq\Delta_{\max}$, z^* is proved to be an optimal SM solution. Otherwise, z^* is a valid upper bound on the optimal solution cost of SM.

The following section describes how the route set $\overline{\mathcal{R}}$ is generated.

4.2.2. Generating the Reduced Route Set $\overline{\mathcal{R}}$ The generation of the set of routes $\overline{\mathcal{R}}$ involves computing sets $\overline{\mathcal{R}}^k$ for each $k \in \mathcal{K}$. In the following description, we outline the generation process for a generic vehicle, omitting the index k for clarity.

We developed a forward dynamic programming-based algorithm in which a label L represents a partial path P starting at depot 0 and ending at vertex $i \in \mathcal{V}$. The label L contains the following information: i: the last vertex in P, $\mathcal{X}$: the set of vertices visited by P (also denoting the sequence visited vertices), c: the travel cost of P, $\bar{c}$: the reduced cost of P, w: the cumulative load along P, $s(\boldsymbol{\mu})$: the earliest time at which a service can start at vertex i under the mean travel time $\boldsymbol{\mu}$, $\mathcal{H}$: the set of resources associated with the lm-SRCs in set $\mathcal{S}$. Let $\mathcal{C}(s)$ denote the base set, $\mathcal{M}(s)$ the memory set, $\beta(s)$ the multiplier, and $\eta(s)$ the dual variable of lm-SRCs of index $s \in \mathcal{H}$. Additionally, two components are specifically introduced to handle the lateness measure $\rho(\cdot)$:

(i) ζ: the cumulative lateness measure along P,

(ii) $\{s(\omega)\}_{\omega\in\Omega}$: the earliest time at which service can start at vertex i under different samples in set Ω.

The label extension rule for a label $L = (i, \mathcal{X}, c, \bar{c}, w, s(\boldsymbol{\mu}), \mathcal{H}, \zeta, \{s(\omega)\}_{\omega\in\Omega})$ is as follows. Let j be a vertex in set $\mathcal{N} \cup \{d(k)\} \setminus \mathcal{X}$ such that $w + q_j \le Q_k$ and $s(\boldsymbol{\mu}) + \mu_{ij} \le l_j$. A new label L' is created to append vertex j to path P to generate path $P' = (P, j)$ as follows: $i(L') = j$, $\mathcal{X}(L') = \mathcal{X}(L) \cup \{j\}$, $c(L') = c(L) + b^k_{ij}$, $w(L') = w(L) + q_j$, $s(\boldsymbol{\mu})(L') = \max\{e_j, s(\boldsymbol{\mu})(L) + \mu_{i(L)j}\}$, $\zeta(L') = \zeta(L) + \rho(\xi_j(P', \tilde{\boldsymbol{t}}))$, $s(\omega)(L') = \max\{e_j, s(\omega)(L) + t^{\omega}_{ij}\}, \forall \omega \in \Omega$, $\bar{c}(L') = \bar{c}(L) - \overline{u}_j$, and the calculation of vector $\mathcal{H}(L')$ and the update of the reduced cost $\bar{c}(L')$ based on the lm-SRCs are given by Algorithm 2. If $j = d(k)$, then path P' corresponds to a complete route which is added to the final set of routes $\overline{\mathcal{R}}^k$ if $\bar{c}(L') \le B - \overline{z}$. Function $\rho(\xi_j(P', \tilde{\boldsymbol{t}}))$ computes the lateness measures in §3. In particular, the OCP is computed by Algorithm 1.

The algorithm starts with an initial label $L = (0, \emptyset, 0, -\overline{v}_k, 0, 0, \{0\}_{\kappa\in\mathcal{H}}, 0, \{0\}_{\omega\in\Omega})$. The dominance rules below are proposed to identify the labels that can be safely discarded to reduce the number of labels.

Dominance 1 *Let L_1 and L_2 be two labels with $i(L_1) = i(L_2)$. Label L_1 dominates label L_2 if*

(i) $X(L_1) = X(L_2)$,

(ii) $s(\boldsymbol{\mu})(L_1) \le s(\boldsymbol{\mu})(L_2)$,

(iii) $s(\omega)(L_1) \le s(\omega)(L_2), \forall \omega \in \Omega$,

(iv) $\bar{c}(L_1) - \sum_{s\in\mathcal{H}:\mathcal{H}_s(L_1)>\mathcal{H}_s(L_2)} \eta(s) \le \bar{c}(L_2)$,

(v) $c(L_1) \le c(L_2)$,

(vi) $\zeta(L_1) \le \zeta(L_2)$.

Finally, two different routes visiting the same customers may have the same lateness measure. To further reduce the number of routes, the following dominance rule is used:

Dominance 2 *A route $r_1 \in \overline{\mathcal{R}}^k$ dominates a route $r_2 \in \overline{\mathcal{R}}^k$ if the following conditions are satisfied:*

Algorithm 2 Calculation of vector $\mathcal{H}(L')$

$\mathcal{H}(L') \leftarrow \mathcal{H}(L)$
for $s = 0$ to $|\mathcal{H}|$ **do**
  if $i \notin \mathcal{M}(s)$ **then**
    $\mathcal{H}(L')[s] \leftarrow 0$
  else if $i \in \mathcal{C}(s)$ **then**
    $\mathcal{H}(L')[s] \leftarrow \mathcal{H}(L')[s] + \beta(s)$
    if $\mathcal{H}(L')[s] \geq 1$ **then**
      $\bar{c}(L') \leftarrow \bar{c}(L') - \eta(s)$
      $\mathcal{H}(L')[s] \leftarrow \mathcal{H}(L')[s] - 1$
Return $\mathcal{H}(L')$

(i) $\mathcal{N}(R_{r_1}) = \mathcal{N}(R_{r_2})$,

(ii) $c^k_{r_1} \leq c^k_{r_2}$,

(iii) $\sum_{i \in \mathcal{V}(R_{r_1})} \rho(\xi_i(R_{r_1}, \tilde{\boldsymbol{t}}) \leq \sum_{i \in \mathcal{V}(R_{r_2})} \rho(\xi_i(R_{r_2}, \tilde{\boldsymbol{t}})$.

4.3. Computing Primal Solutions

We designed a primal heuristic combined with a VNS algorithm to compute upper bounds on the optimal solution costs of formulations DM and SM. The resulting heuristic algorithm is applied at the root node of the BPC algorithm for DM and after the computation of the dual solution used by the route enumeration method used to solve SM.

Specifically, for formulation DM, we consider the set of columns contained in the final RMP, denoted as $\mathcal{R}'$, to define a reduced DM formulation obtained by replacing the route set $\mathcal{R}$ with $\mathcal{R}'$. Similarly, for formulation SM, we consider the route set $\mathcal{R}''$ that composes the final RMP used to compute the dual solution (see §4.2.1). We also include in the set $\mathcal{R}''$ the routes in $\mathcal{R}'$ not included in $\mathcal{R}''$, for which we recompute the corresponding route costs based on the lateness measure at hand. The resulting reduced DM and SM formulations are then solved to optimality using a general-purpose MIP solver.

To further improve the solutions, we also run a VNS algorithm based on similar algorithms described in the literature for VRPs (see, among others, Bräysy 2003, Kytöjoki et al. 2007, Wei et al. 2015, Lim et al. 2017, Zhang et al. 2021). The VNS is applied, receiving the solution computed for DM and SM as input. For formulation DM, the original objective function (1) is considered, whereas for SM, the budget constraint is not considered, and its violation is penalized based on the following objective function.

$$\sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} \left(\sum_{i \in \mathcal{V}(R_r)} \rho(\xi_i(R_r, \tilde{\boldsymbol{t}})) \right) x^k_r + M \left(\sum_{k \in \mathcal{K}} \sum_{r \in \mathcal{R}^k} c^k_r x^k_r - B \right)^+, \tag{26}$$

where M is a large constant.

Our VNS is based on the scheme described in Hansen et al. (2019). The algorithm starts from a primal solution $\boldsymbol{x}$. A neighbor solution $\boldsymbol{x}'$ is first generated using a defined neighborhood structure and is then improved by the local search procedure. If the new solution $\boldsymbol{x}''$ is better than the incumbent $\boldsymbol{x}$, then the search moves to $\boldsymbol{x}''$ and reverts to the first neighborhood structure; otherwise, it continues with the next structure. After all neighborhood structures are attempted, the starting solution is diversified (as described next), and the next loop is repeated until the stopping criterion is met. The local search operators include *relocate*, which relocates one customer to another position; *swap*, which exchanges the positions of two customers; *2-opt*, where a selected sequence of customers is reversed; and *2-opt**, which interchanges the end parts of two different routes. Neighborhood structures are built on the block exchange, involving the exchange of two blocks of consecutive customers. A series of H neighborhood structures are designed, and in the hth structure, the block exchange is performed h times, where $h = 1, 2, \ldots, H$. Diversification follows the *ruin–reconstruct* approach, where several customers are randomly removed, and a full solution is then rebuilt using the insertion method.

5. Computational Study

This section presents extensive experimental analysis conducted on instances from the literature (see §5.1). First, we describe our experimental setup (see §5.2). Second, we report the detailed results of solving the deterministic model DM using the exact method and the primal heuristic (see §5.3). Third, we present the results of the model SM under different lateness measures (see §5.4). Fourth, we visualize, analyze, and compare the solutions of DM and SM, to understand their difference (see §5.5). Additionally, we conduct sensitivity analysis on the budget B, and number N of in-samples (see §5.6). Finally, we analyze the computational effectiveness of the BPC components (see §5.7).

We implemented the algorithms using Java on a Dell Vostro 3910, equipped with an Intel (R) Core (TM) i7-12700F CPU running at 2.10 GHz and 64 GB RAM. We used the IBM CPLEX 22.1.0.0 LP and MIP solvers (IBM CPLEX 2023). Moreover, we imposed a time limit of 2 hours for executing the algorithms.

5.1. Benchmark Instances

The benchmark instances in our experiments correspond to the set used by Pugliese et al. (2022a), derived from Solomon's deterministic VRPTW instances (Solomon 1987). These instances include destination, capacity, and available time windows of ODs. They are divided into six main classes: C1, C2, R1, R2, RC1, and RC2. C1 and C2 represent clustered instances where customer locations are grouped, while R1 and R2 indicate randomly distributed instances. Sets RC1 and RC2 combine randomly and clustered customers. Type 1 sets (C1, R1, RC1) have narrow time windows, while type 2 sets (C2, R2, RC2) have larger ones. Pugliese et al. (2022a) derived 36 small instances with 5, 10, and 15 customers, covering all six main classes. Additionally, they derived 60 medium instances with 25 and 50 customers and 30 large instances with 100 customers. They also considered 30 large instances with 200 customers and 36 very large instances with

400 customers. Our computational study for the DM model used 36 small, 60 medium, and 30 large-size instances, involving up to 100 customers. Furthermore, we used only instances with tight time windows (C1, R1, and RC1) to better evaluate on-time arrival performances. Additionally, our computational study for the SM model used 18 small and 30 medium-sized instances involving up to 50 customers.

We further categorize instances into Sets A and B based on the types of OD routes. Suppose ODs can serve at most one customer in a delivery (Archetti et al. 2016), corresponding to single-customer routes. In that case, we designate these instances as Set A. Conversely, instances where an OD can serve more than one customer (Pugliese et al. 2022a), referred to as general routes, are labeled as Set B. While Set A represents a simpler case compared to Set B, exact solutions for Set A were not reported in the literature. For Set B, Pugliese et al. (2022a) reported optimal solutions obtained by directly solving an MIP model using the Cplex solver. Their results showed that some instances with up to 50 customers could be solved optimally and that many instances involving 25 and 50 customers remain unsolved within an imposed time limit of 1,800 seconds.

We use the original distance value computed based on the Euclidean distance as the corresponding travel cost and time. We assume that the unit of travel time is in minutes. As commonly done in the literature, the distance and travel time between two nodes are rounded down to the first decimal place (see, for example, Jepsen et al. 2008, Baldacci et al. 2011a, Pecin et al. 2017c).

5.2. Setup

In preliminary experiments, we identified the following parameter settings as computationally beneficial. To mitigate the impact of lm-SRCs on the labeling algorithm, we allow a maximum of 50 cuts to be generated, with the most violated ones selected. Other parameters include $\delta^{perc}\%$ and δ^{num}, used to stop cut generation by tailing off (Sadykov et al. 2021). If the number of times the dual-primal gap decreases by less than $\delta^{perc}\% = 1.5\%$ and reaches $\delta^{num} = 3$, branching is performed.

Zhang et al. (2021) obtained uncertain travel times based on an asymmetric two-point distribution. Specifically, the travel time, $\tilde{t}_{ij}$, for each arc $(i,j) \in \mathcal{A}$, is assumed to follow an asymmetric two-point distribution supported on $t_{ij} - \sigma_{ij}/\sqrt{3}$ and $t_{ij} + \sqrt{3}\sigma_{ij}$ with respective probabilities of 0.75 and 0.25. Here, the deterministic travel time of arc (i,j) is used as the mean t_{ij}, and the standard deviation, $\sigma_{ij} = \lambda_{ij} t_{ij}$, with λ_{ij} randomly chosen from the interval $[0.1, 0.5]$. In the experiments, unless otherwise stated, the number N of in-samples is set as 100, the service level γ is set as 1 for all nodes, and the budget B is set as $1.04 \times opt$, where opt is the best solution cost of the deterministic VRPC solution. For the route R delivered by the OD $k \in \mathcal{K} \setminus \{1\}$, the cost can be computed as the compensation coefficient $\vartheta = 1.2$ times the detour distance, i.e., $\vartheta \times (\sum_{(i,j)\in\mathcal{A}(R)} b_{ij} - b_{0,d(k)})$.

5.3. Results on the Deterministic VRPC: Solving Model DM

We present summarized results for the deterministic VRPC on Set A and Set B in Tables 2 and 3. Detailed solutions are reported in the e-companion to this paper (see §EC.2). In these tables, instances are grouped by customers' number $|\mathcal{N}|$, and the values reported are averaged over the corresponding groups of instances. We report the number $\#opt$ of instances solved to optimality, the total number $\#tot$ of instances, the solution cost obj, the gap of the primal heuristic $\%UB_P$ from the optimal value, the gap of the lower bound at the root node $\%LB_r$ relative to the optimal value, the number $\#RD$ of utilized RDs in the optimal solution, the number $\#OD$ of utilized ODs, the number $\#ACRD$ of customers served by an RD, the number $\#ACOD$ of customers served by an OD, the number $\#nodes$ of nodes in the enumeration tree, number $\#cols$ of columns (routes) generated by the pricing algorithm, and number $\#cuts$ of valid inequalities separated. Columns t_r, t_{PH}, t_{SEP}, t_{PP}, and t_{tot} represent the computing time in seconds spent in computing the lower bound at the root node, the primal heuristic, separating procedures, pricing, and the total computing time, respectively.

The tables show that BPC solved instances with up to 100 nodes on Set A and Set B to optimality. We observe that the average computing time in Set B is less than in Set A, given that the number of nodes in the enumeration tree in Set A is larger than in Set B. However, when considering the computing time for the same instances, Set B is, on average, more challenging than Set A (see the details reported in §EC.2). The columns $\%UB_P$ and $\%UB_r$ indicate that our primal heuristic performs well, and the lower bound of the VRPC at the root node is, on average, tight.

Additionally, columns $\#RD$ and $\#OD$ indicate similar utilization of RDs and ODs in sets A and B in the solutions. However, the solution costs of instances from Set B are generally significantly lower than those from Set A for the same dimension. This suggests that allowing an OD to serve multiple customers in these instances leads to more cost-effective solutions. Indeed, columns $\#ACOD$ show that on Set B instances, the ODs routes, on average, serve more than one customer.

The computing times reveal that the pricing algorithm consumes most of the time, underscoring its significance in our BPC.

Table 2 Results of the VRPC on Set A

$\|\mathcal{N}\|$	$\#opt/\#tot$	obj	$\%UB_P$	$\%LB_r$	$\#RD$	$\#OD$	$\#ACRD$	$\#ACOD$	$\#nodes$	$\#cols$	$\#cuts$	t_r	t_{PH}	t_{SEP}	t_{PP}	t_{tot}
5	12/12	146.0	0.0	1.6	1.3	1.9	2.5	1.0	2.0	20.8	0.2	0.0	0.0	0.0	0.0	0.0
10	12/12	234.7	0.0	1.4	2.1	2.3	3.9	1.0	3.2	90.1	2.3	0.0	0.0	0.0	0.0	0.0
15	12/12	292.2	0.1	1.7	2.8	4.0	4.1	0.9	16.2	1746.3	95.4	0.0	0.7	0.0	4.2	5.4
25	30/30	313.1	0.1	1.6	3.1	6.0	8.2	1.0	106.7	15467.8	552.8	0.1	4.1	0.3	11.0	31.3
50	24/30	488.7	0.5	2.2	4.1	11.5	10.9	1.0	146.5	105899.3	2139.6	0.4	7.8	4.0	233.6	445.9
100	9/30	835.0	0.2	1.2	6.8	28.3	12.4	1.0	1361.2	444127.3	16312.4	0.3	37.4	233.5	626.7	1828.9
all	99/126	384.9	0.2	1.6	3.4	9.0	7.0	1.0	272.6	94558.6	3183.8	0.1	8.4	39.6	145.9	385.3

Table 3 Results of the VRPC on Set B

$\|\mathcal{N}\|$	$\#opt/\#tot$	obj	$\%UB_P$	$\%LB_r$	$\#RD$	$\#OD$	$\#ACRD$	$\#ACOD$	$\#nodes$	$\#cols$	$\#cuts$	t_r	t_{PH}	t_{SEP}	t_{PP}	t_{tot}
5	12/12	142.3	0.0	1.2	1.3	1.8	2.4	1.1	2.2	23.8	0.4	0.0	0.0	0.0	0.0	0.0
10	12/12	227.3	0.0	2.1	2.0	2.3	3.6	1.3	11.2	292.5	4.8	0.0	0.1	0.0	0.0	0.0
15	12/12	275.1	0.0	0.7	2.5	4.2	4.1	1.3	5.2	409.1	24.2	0.0	0.3	0.0	0.2	0.3
25	30/30	295.7	0.0	1.3	2.7	5.8	8.5	1.4	77.2	19151.4	483.0	0.0	7.2	0.2	19.9	36.0
50	24/30	418.2	0.3	1.8	3.0	13.0	11.7	1.7	830.2	294621.1	2526.1	0.2	42.0	9.4	134.1	278.5
100	4/30	727.0	0.3	0.9	6.0	28.3	8.9	1.8	550.0	812953.5	7605.3	0.5	25.4	165.9	181.7	1027.0
all	94/126	347.6	0.1	1.4	2.9	9.2	6.5	1.4	246.0	187908.6	1774.0	0.1	12.5	29.2	56.0	223.7

5.4. Results on the Uncertain VRPC: Solving Model SM

We considered SM with four specific lateness measures: the lateness probability and expected lateness (described in §3.1), expected truncated exponential disutility (§3.2), and the Optimal Calibration Parameter (§3.3), denoted by SM-P, SM-T, SM-E and SM-OCP, respectively. We solved the service-centric VRPC with these lateness measures using the route enumeration-based method described in §4.2. Table 4 summarizes the results obtained on solving SM-OCP. In addition to the notations already introduced, the table displays the budget B, the gap between the lower bound $\bar{z}$ at the root node and the budget, and the route $cost$. The columns $|\overline{\mathcal{R}}|$, $|\overline{\mathcal{R}}_{RD}|$, and $|\overline{\mathcal{R}}_{OD}|$ report the cardinality of the set of all routes, RDs' routes, and ODs' routes generated by the route enumeration procedure. The table also lists t_{obj}, t_{RE}, and t_{IP}, which denote the times spent in computing the corresponding lateness measure, in generating the route set $\overline{\mathcal{R}}$, and in solving the IP model.

From the column t_{obj}, we observe that our bisection Algorithm 1 is very efficient. Most of the solution time is spent on route enumeration. Moreover, as the cardinality of the enumerated routes increases, the time spent solving the IP model also increases significantly. Here, we omit the column $\%UB_P$ and only present the computing time of the primal heuristic t_{PH} because the special lateness measure equals 0 in some cases. We solved 38 out of 48 instances optimally involving up to 50 nodes. The results of SM-P, SM-T, and SM-E, as reported in the e-companion to this paper (see §EC.3), demonstrate that the exact methods SM-P, SM-T, and SM-E exhibit performance similar to that of SM-OCP.

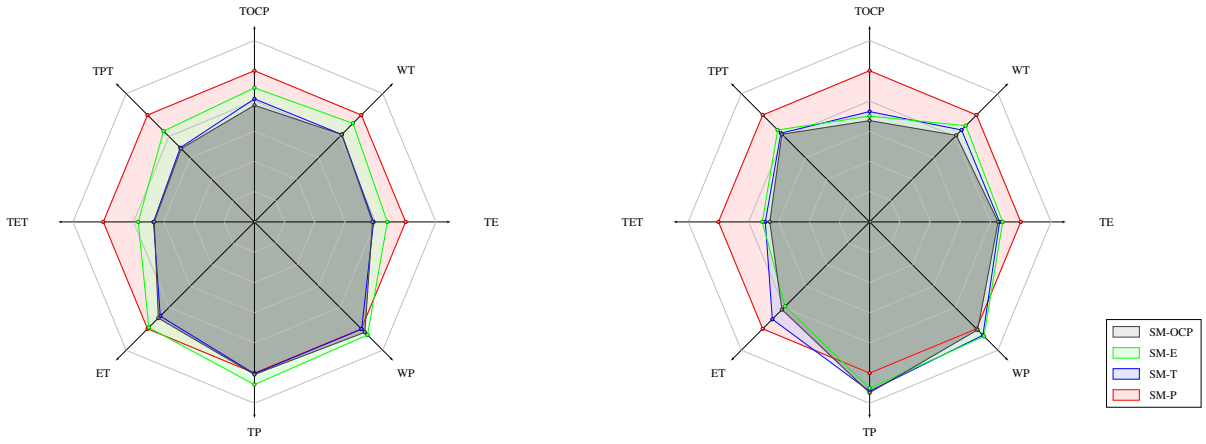
Table 4 Results of the exact method on solving SM-OCP

$\|\mathcal{N}\|$	$\#opt/\#tot$	B	gap	obj	$cost$	t_{PH}	$\|\overline{\mathcal{R}}\|$	$\|\overline{\mathcal{R}}_{RD}\|$	$\|\overline{\mathcal{R}}_{OD}\|$	t_{obj}	t_{IP}	t_{RE}	t_{tot}
5	6/6	151.6	6.8	0.1	145.8	0.0	9.5	5.8	3.7	0.0	0.0	0.0	0.5
10	6/6	272.7	17.0	2.3	267.3	0.0	143.2	122.0	21.2	0.0	0.0	0.1	0.2
15	6/6	249.3	11.9	1.1	246.9	1.4	135.0	107.0	28.0	0.0	0.0	0.2	0.4
25	13/15	359.7	18.6	1.3	356.0	7.9	3559.2	3397.7	161.5	0.0	0.3	6.0	6.4
50	7/15	517.8	32.4	9.5	516.7	77.7	41452.0	38162.1	3289.9	0.0	63.2	516.8	580.6
all	38/48	310.2	17.3	2.9	306.5	17.4	9059.8	8358.9	700.8	0.0	12.7	104.6	117.6

Below, we assess the service performance of the SM-OCP, SM-P, SM-T, and SM-E; we use the deterministic model DM as the benchmark to investigate the value of considering uncertainty. We compare the results based on instances with 25 nodes because they were all solved to optimality; we summarize them

Table 5 Comparison of different decision criterion

Model	TP	TE	WP	WT	ET	TOCP	TPT	TET	Avg
DM	3.660	3.835	2.374	2.354	2.571	1.557	3.191	4.528	2.177
SM-P	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000	1.000
SM-T	1.004	0.788	1.003	0.818	0.881	0.814	0.694	0.668	0.848
SM-E	1.076	0.879	1.058	0.922	0.988	0.887	0.850	0.769	0.940
SM-OCP	1.009	0.781	1.033	0.817	0.899	0.772	0.686	0.663	0.840



(a) Performance under 25-node instances

(b) Performance under 50-node instances

Figure 2 Normalized values of the indicators for the SM-OCP, SM-P, SM-T, and SM-E

in Table 5. We conduct an out-of-sample evaluation by generating a new set $\bar{\Omega}$ of $\bar{N} = 10,000$ independent samples of $\tilde{\boldsymbol{t}}$. We consider various indicators that decision-makers may be concerned with, as follows:

- TP: Total lateness probability: $\frac{\sum_{i\in\mathcal{V}}\sum_{\omega\in\bar{\Omega}}\mathbb{I}[\xi_i^\omega>0]}{\bar{N}}$.
- WP: Worst lateness probability among all the nodes: $\max_{i\in\mathcal{V}}\frac{\sum_{\omega\in\bar{\Omega}}\mathbb{I}[\xi_i^\omega>0]}{\bar{N}}$.
- TE: Total expected lateness duration: $\frac{\sum_{i\in\mathcal{V}}\sum_{\omega\in\bar{\Omega}}(\xi_i^\omega)^+}{\bar{N}}$.
- WT: Worst-case expected lateness duration: $\max_{i\in\mathcal{V}}\frac{\sum_{\omega\in\bar{\Omega}}(\xi_i^\omega)^+}{\bar{N}}$.
- ET: Total expected truncated exponential lateness duration: $\frac{\sum_{i\in\mathcal{V}}\sum_{\omega\in\bar{\Omega}}\exp(\xi_i^\omega)\mathbb{I}[\xi_i^\omega>0]}{\bar{N}}$.
- TOCP: Total OCP, each computed by algorithm 1 using the set $\bar{\Omega}$ of samples in place of Ω. Note that a smaller OCP implies a lower probability bound (see Theorem 1).
- TPT: Total lateness probability that exceeds 5 minutes: $\frac{\sum_{i\in\mathcal{V}}\sum_{\omega\in\bar{\Omega}}\mathbb{I}[\xi_i^\omega-5>0]}{\bar{N}}$.
- TET: Total expected lateness time that exceeds 5 minutes: $\frac{\sum_{i\in\mathcal{V}}\sum_{\omega\in\bar{\Omega}}(\xi_i^\omega-5)^+}{\bar{N}}$.

In Table 5, we normalize the values of the different methods. The DM performs the worst under uncertain travel times. It is obvious that the service-centric model with different lateness measures, SM-P, SM-T, SM-E and SM-OCP, can greatly mitigate uncertainty relative to DM, but at the expense of an additional 4% cost budget (see §5.2). Comparing the methods under uncertainty, SM-OCP outperforms SM-P, SM-T, and SM-E on average in mitigating lateness, followed by SM-T. The SM-OCP achieves the smallest TE, WT,

TOCP, TPT, and TET, but has a slightly higher ET, TP, and WP than SM-T or SM-P. The computation of the indicator ET is the same as the objective function of SM-E, but, unexpectedly, SM-E does not perform the best in this indicator, perhaps due to i) the *overfitting effect* widely recognized in machine learning and ii) the fact that optimizing over exponential functions are computationally challenging (Mihatsch and Neuneier 2002, Gosavi et al. 2014). Commonly, SM-P performs best on the probability indicators TP and WP since its objective is to minimize the total lateness probability. Overall, SM-OCP and SM-T show a good balanced performance on all on-time service indicators. We also visualize the results using Figure 2(a) to be intuitive.

To further elucidate the benefits of SM-OCP, we also test the performance based on instances with 50 nodes using, in this case, the solutions from the primal heuristic. The results are illustrated in Figure 2(b). According to the results, the SM-OCP generally outperforms SM-P, SM-T, and SM-E. Compared to SM-T and SM-E, although the SM-OCP performs worse on TP, the differences are very small. Additionally, the SM-E achieves the smallest ET since its objective function is ET.

5.5. Comparison of Solutions Derived from DM and SM Models

This section compares and analyzes solutions for a specific instance under the VRPTW model (a special case of DM when no ODs exist), DM, and SM-OCP. We arbitrarily chose instance 050_R101C50 with 50 customers, 8 RDs, and 15 ODs. We set the compensation coefficient $\vartheta = 1$ to ensure that the objective values (i.e., travel costs) of VRPTW and DM are comparable. Consequently, their travel costs are proportional to the distances, encompassing the travel distances of RDs and detour distances of ODs. Figure 3(a) displays the optimal routes obtained by solving the VRPTW. To ensure a feasible solution, we set the available number m_1 of RDs equal to the number $|\mathcal{N}|$ of customers. To investigate the effect of introducing ODs, we illustrate the routes of RDs and ODs for the DM in Figure 3(b). As shown, 30 out of 50 customers are served by ODs with destinations nearby, significantly reducing travel costs from 1017.0 to 568.8.

In Figure 3(c), we present the routes of SM-OCP. To explore the structure of the routes while considering travel time uncertainty, we highlight customer nodes in red in Figures 3(b) and 3(c) where service start times differ between DM and SM-OCP. Additionally, we annotate the corresponding time window and service start time for each node i in Figure 3(d) using the format $[e_i, l_i] : s_i, s'_i$, where s_i is the service start time for DM, and s'_i is for SM-OCP. As observed, some service start times in DM are close to the deadlines; for example, the service start time of the bottom-right customer is 59.8, slightly less than the deadline of 60. While this is not problematic in a deterministic environment, such a solution carries a risk of lateness when implemented in the real world of uncertainty. Our SM-OCP mitigates this risk, where the (deterministic/planned) service start times tend to align closely with the required earliest arrival times, reserving time buffers to hedge against uncertainty in travel times.

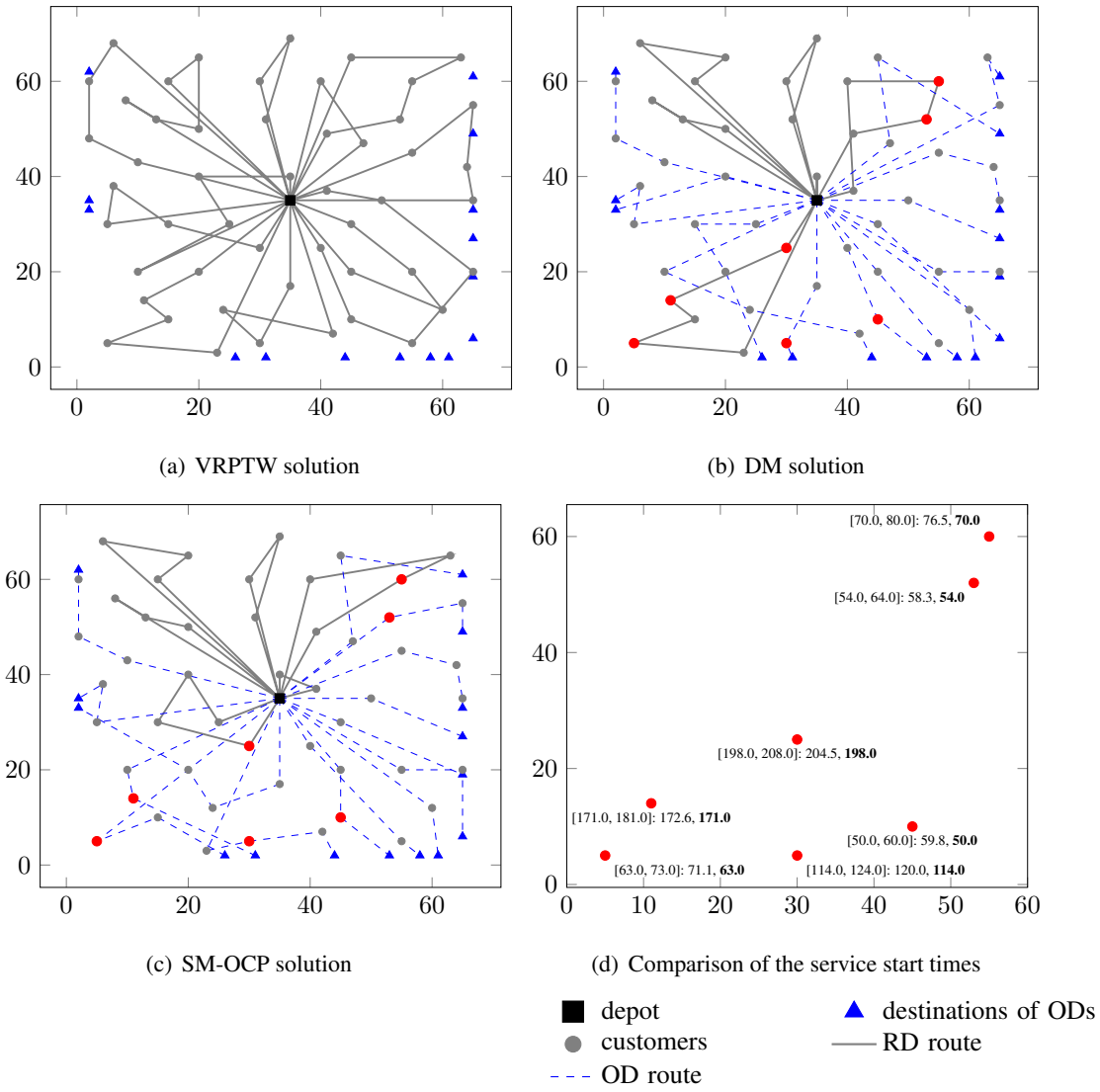


Figure 3 Routes of the solutions under VRPTW, DM, and SM

5.6. Sensitivity Analysis

In this section, we offer insights into the sensitivity of problem characteristics, specifically the cost budget B and the number N of in-samples. Our analysis focuses on the 25 node instances solved using the exact method, the lateness measure SM-OCP, and some key indicators.

5.6.1. Budget One of the primary concerns for companies is determining the value of the budget, specifically the maximum cost they can afford. The cost budget cannot be overlooked in maintaining a high level of service. A larger cost budget can accommodate fluctuations in travel time, ensuring prompt service. However, it also leads to higher costs. Figure 4 compares the results obtained for different budget values $B = 1.01, 1.02$, and $1.03 \times opt$ with our default setting. The shaded area represents the 95% confidence interval over all the instances tested. It is interesting to study the results when $B = opt$, but in this case, the SM-

OCP is infeasible in most cases. Here, we normalized the results of the SM-OCP based on the performance of DM; for instance, the y-axis value TP is computed as TP(SM-OCP)/TP(DM), where TP(SM-OCP) and TP(DM) denote the total lateness probability of the methods SM-OCP and DM in the out-of-sample test, respectively. We observe that both TP and TE are less than 1, usually less than 0.5, demonstrating the efficacy of our service-centric model in mitigating uncertainty. In contrast, the total computing time t_{tot} is much larger than 1, indicating longer computing time than DM. With the increased budget, the service level improves while higher costs and longer computing times are incurred. Based on these results, a decision-maker can extract valuable insights to facilitate a balanced trade-off between operational costs and service levels.

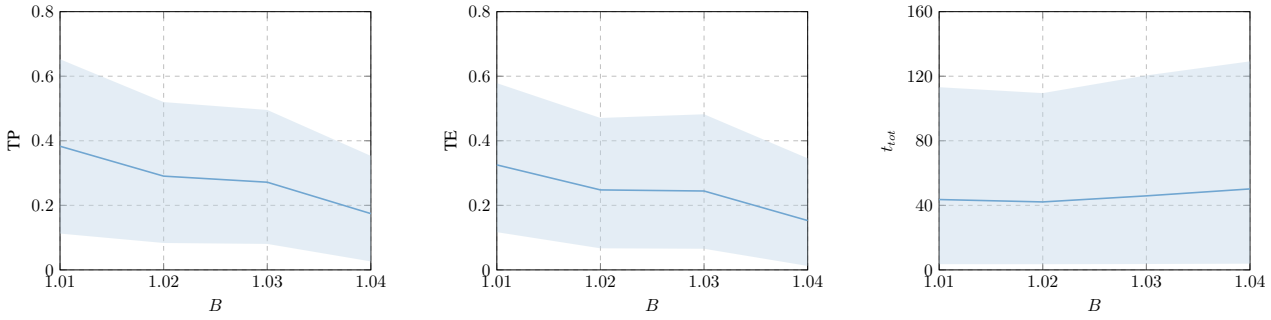


Figure 4 Normalized performance on different budget values

5.6.2. Number of Samples The number of samples directly affects the problem's computational difficulty. Figure 5 presents results for cases with the number of in-samples set as $N \in \{10, 100, 1000, 3000\}$. Here, we normalized the results based on DM. We observe that the values of TP and TE are less than 1, underscoring the importance of considering uncertainty. As expected, the service level improves with an increased sample size but incurs greater computational complexity. In particular, the curves of TP and TE appear to be convex, demonstrating the diminishing value of samples (or information); in this case, 1000 samples seem sufficient to guarantee a good performance. From the t_{tot} curve, we conclude that the t_{tot} time increases linearly as the sample size increases.

5.7. Effectiveness of the BPC Components

To comprehensively evaluate the effectiveness of various components of our algorithms, we employ the concept of a *performance profile* introduced by Dolan and Moré (2002) in the analysis. This approach allows us to evaluate and compare the performance of optimization algorithms, utilizing cumulative distribution functions for a given performance metric.

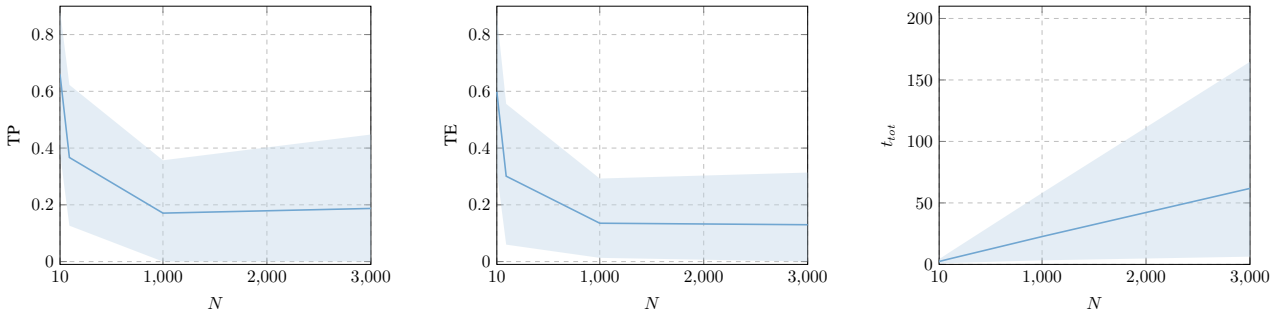


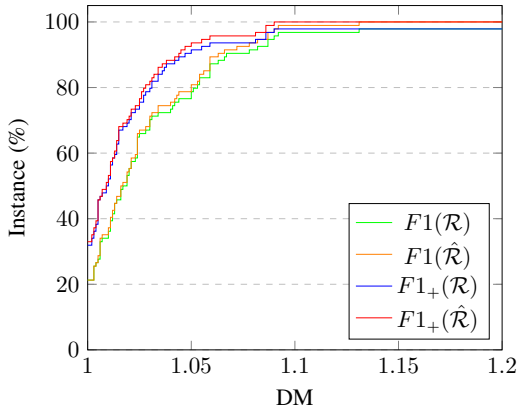
Figure 5 Normalized performance on different samples

5.7.1. Analysis of the DM Model We consider four formulations for the deterministic VRPC. (i) $F1(\mathcal{R})$: formulation DM based on elementary routes; (ii) $F1(\hat{\mathcal{R}})$: DM based on *ng*-path relaxation; (iii) $F1_+(\mathcal{R})$: DM based on elementary routes and lm-SRCs; (iv) $F1_+(\hat{\mathcal{R}})$: DM based on *ng*-path relaxation and lm-SRCs.

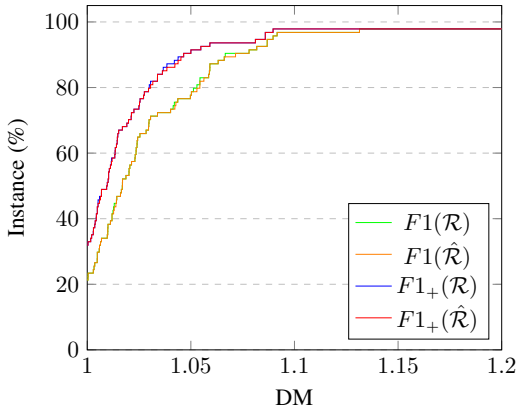
We first compare the lower bounds obtained by solving the LP relaxations of these formulations at the root nodes of the enumeration trees. We report the results in the first three figures in Figure 6. Figure 6(a) reports the results for all the instances described in §5.1. The horizontal axis represents the ratio of the best-known upper bound versus the lower bound; the vertical axis reports the percentage of instances achieving the ratios. Formulation $F1_+(\hat{\mathcal{R}})$ outperforms the others, providing optimal solutions for 33% of all instances. Here, the horizontal value of 1 implies optimality because the lower and upper bounds meet. We expect $F1_+(\mathcal{R})$ to dominate $F1_+(\hat{\mathcal{R}})$ because the latter is a (*ng*-path) relaxation of the former. However, this is not the case in Figure 6(a), because some instances of $F1_+(\mathcal{R})$ cannot be solved optimally within the imposed time limit. Hence, we report in Figure 6(b) a counterpart of Figure 6(a) but using only the instances that can be solved optimally by all the four formulations' LP relaxations. Here, the lower bounds of $F1(\mathcal{R})$ and $F1_+(\mathcal{R})$ are tighter than those of $F1(\hat{\mathcal{R}})$ and $F1_+(\hat{\mathcal{R}})$, respectively, as expected. This highlights the effectiveness of the *ng*-path relaxation—although being a relaxation, it is relatively tight and, importantly, reduces the computing time.

In Figure 6(c), we report the computing time for solving the LP relaxations. The horizontal axis represents the normalized computing time. For each instance and each formulation, its normalized computing time is calculated as its computing time divided by the minimum computing time among the four formulations for solving this instance. The comparison between $F1(\mathcal{R})$ and $F1_+(\mathcal{R})$, as well as $F1(\hat{\mathcal{R}})$ and $F1_+(\hat{\mathcal{R}})$, in Figure 6(a) clearly illustrate the benefits of considering the lm-SRCs (rendering a tighter lower bound), whereas Figure 6(c) shows that the benefits come at the expense of longer computing times. To investigate the overall computational performance, we solve the above four formulations using our BPC framework in §5.7 but may remove some corresponding algorithmic components, resulting in algorithms BP($\mathcal{R}$), BP($\hat{\mathcal{R}}$),

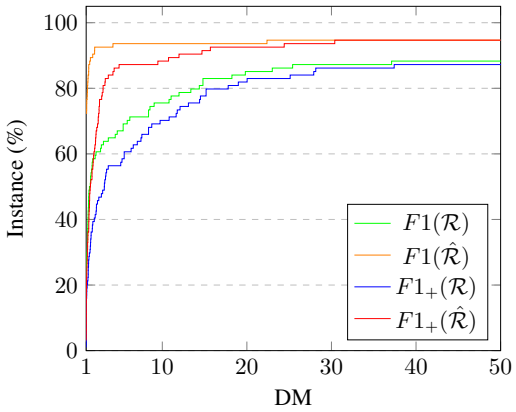
BPC($\mathcal{R}$), and BPC($\hat{\mathcal{R}}$), respectively. We report the results in Figure 6(d); it is evident that BPC($\hat{\mathcal{R}}$) outperforms the other three variants. This further underscores the effectiveness of considering *ng*-path relaxation and lm-SRCs.



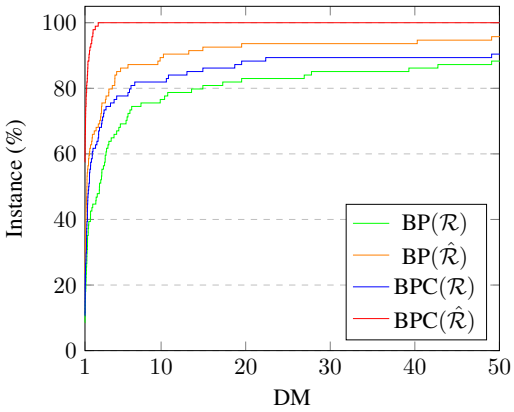
(a) Quality of the lower bounds for all the instances



(b) Quality of lower bounds for the instances that are solved optimally at root nodes in enumeration trees



(c) Normalized computing time for the lower bounds



(d) Normalized computing time for solving DM

Figure 6 Effectiveness of the components of BPC

5.7.2. Analysis of the SM Model Based on the OCP In the route enumeration process of solving SM-OCP, we introduced the dominance condition (i) $\mathcal{X}(L_1) = \mathcal{X}(L_2)$, meaning that the visited customers of two different labels must be the same. Inspired by Sadykov et al. (2021) and Pecin et al. (2017b), we divide $|q_i|$ buckets according to the supply of customer $i \in \mathcal{N}$ to reduce the enumeration times. For details, refer to Sadykov et al. (2021). To assess the effectiveness of this method, we consider the following two approaches. (i) M1: the route enumeration-based algorithm in §4.2 based on buckets; (ii) M2: the algorithm in §4.2 without buckets. Figure 7 displays the performance profiles of these different exact methods concerning computing times. In this case, for each instance and approach, its normalized computing time is computed

as its computing time divided by the maximum computing time between the two approaches for solving the instance. M1 successfully solves 38 instances optimally in our experiments, whereas M2 can only solve 32 instances, with significantly longer computing times. The figure illustrates that M1 outperforms M2 regarding the number of instances solved optimally and the corresponding computing times, demonstrating the benefit of the bucket-based approach.

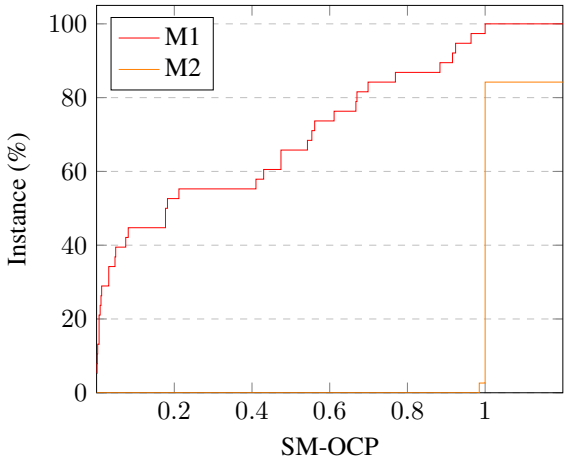


Figure 7 Effectiveness of the bucket

6. Conclusions

In recent years, vehicle routing problems with crowdshipping under uncertainty have garnered significant interest from industry and academia. Notably, the recent literature has primarily focused on the uncertainty in the availability of occasional drivers and requests from customers, with limited attention given to uncertain travel times.

We investigated a service-centric vehicle routing problem with crowdshipping (VRPC) under uncertain travel times. Our objective is to mitigate the lateness risk and improve service quality. Recognizing that customers' impatience exponentially increases with lateness duration (Daley 1965, Garnett et al. 2002), we introduce a truncated exponential disutility and a novel lateness measure, called Optimal Calibration Parameter (OCP), to calibrate the parameter in the disutility. As the main results of our study, we showcase its notable managerial and computational properties.

We proposed solution approaches for both the deterministic and service-centric VRPC. Concerning the deterministic variant, we developed a state-of-the-art branch-price-and-cut algorithm. We extensively tested it on instances proposed in the literature, solving to optimality several open VRPC instances. Specifically, 99 out of 126 instances with up to 100 nodes are solved to optimality. Exploiting the structure of the service-centric variants, we formulate a set partitioning-based model and develop an exact solution framework combining route enumeration and state-of-the-art column-and-row generation algorithms. We solved 38

out of 48 instances optimally with up to 50 nodes. We also designed an efficient primal heuristic for the deterministic and service-centric VRPC.

Our computational experiments also demonstrate the efficacy of our service-centric approach in mitigating lateness, examining the influence of essential parameters, and assessing the computational efficiency of the proposed algorithms. Notably, our newly proposed OCP proves to be more effective in mitigating lateness than existing approaches. The sensitivity analyses on the budget and the number of travel time samples provide insights into their impacts on the service level and computing time. Specifically, the experiments reveal a trade-off between the service level and the imposed budget. Additionally, a number of samples equal to 1,000 represents a good compromise between the performance indicators and the computing time of the proposed solution approach.

Some interesting avenues for future research stemming from this study can be explored. Firstly, in reality, occasional drivers and customer requests become available over time. While we assumed that the information about occasional drivers and orders is known in advance, an interesting extension would be to investigate a dynamic service-centric problem where information is revealed gradually. Secondly, the compensation coefficient influences operating costs and service levels without considering its impact on occasional drivers' availability or willingness to participate. Further research could involve treating the compensation coefficient as a decision variable, enabling decision-makers to better balance operating costs and service levels through joint pricing and routing decisions.

Acknowledgments

We thank Prof. Claudia Archetti and Dr. Giusy Macrina for providing the instances used for our experiments.

This research was supported by the National Natural Science Foundation of China (72371204, 72293563, and 71901180). The first three authors are supported by award ARG01-0430-230029 from the Qatar National Research Fund ("QNRF") under the umbrella of Qatar Research Development and Innovation Council ("QRDI"). The content is solely the responsibility of the authors and does not necessarily represent the official views of QNRF and QRDI.

References

Adulyasak Y, Jaillet P (2015) Models and algorithms for stochastic and robust vehicle routing with deadlines. *Transportation Science* 50(2):608–626.

Agatz N, Erera A, Savelsbergh M, Wang X (2012) Optimization for dynamic ride-sharing: A review. *European Journal of Operational Research* 223(2):295–303.

Agatz N, Erera AL, Savelsbergh MW, Wang X (2011) Dynamic ride-sharing: A simulation study in metro atlanta. *Procedia-Social and Behavioral Sciences* 17:532–550.

Allahviranloo M, Baghestani A (2019) A dynamic crowdshipping model and daily travel behavior. *Transportation Research Part E: Logistics and Transportation Review* 128:175–190.

Alnaggar A, Gzara F, Bookbinder JH (2021) Crowdsourced delivery: A review of platforms and academic literature. *Omega* 98:102139.

Archetti C, Guerriero F, Macrina G (2021) The online vehicle routing problem with occasional drivers. *Computers & Operations Research* 127:105144.

Archetti C, Savelsbergh M, Speranza MG (2016) The vehicle routing problem with occasional drivers. *European Journal of Operational Research* 254(2):472–480.

Arslan AM, Agatz N, Kroon L, Zuidwijk R (2019) Crowdsourced delivery—a dynamic pickup and delivery problem with ad hoc drivers. *Transportation Science* 53(1):222–235.

Aumann RJ, Serrano R (2008) An economic index of riskiness. *Journal of Political Economy* 116(5):810–836.

Baldacci R, Bartolini E, Mingozzi A (2011a) An exact algorithm for the pickup and delivery problem with time windows. *Operations research* 59(2):414–426.

Baldacci R, Mingozzi A, Roberti R (2011b) New route relaxation and pricing strategies for the vehicle routing problem. *Operations Research* 59(5):1269–1283.

Behrend M, Meisel F (2018) The integration of item-sharing and crowdshipping: Can collaborative consumption be pushed by delivering through the crowd? *Transportation Research Part B: Methodological* 111:227–243.

Behrend M, Meisel F, Fagerholt K, Andersson H (2019) An exact solution method for the capacitated item-sharing and crowdshipping problem. *European Journal of Operational Research* 279(2):589–604.

Behrend M, Meisel F, Fagerholt K, Andersson H (2021) A multi-period analysis of the integrated item-sharing and crowdshipping problem. *European Journal of Operational Research* 292(2):483–499.

Boysen N, Emde S, Schwerdfeger S (2022) Crowdshipping by employees of distribution centers: Optimization approaches for matching supply and demand. *European Journal of Operational Research* 296(2):539–556.

Bräysy O (2003) A reactive variable neighborhood search for the vehicle-routing problem with time windows. *INFORMS Journal on Computing* 15(4):347–368.

Charnes A, Cooper WW (1959) Chance-constrained programming. *Management science* 6(1):73–79.

Costa L, Contardo C, Desaulniers G (2019) Exact branch-price-and-cut algorithms for vehicle routing. *Transportation Science* 53(4):946–985.

Dahle L, Andersson H, Christiansen M (2017) The vehicle routing problem with dynamic occasional drivers. *International conference on computational logistics*, 49–63 (Springer).

Dahle L, Andersson H, Christiansen M, Speranza MG (2019) The pickup and delivery problem with time windows and occasional drivers. *Computers & Operations Research* 109:122–133.

Daley D (1965) General customer impatience in the queue gi/g/1. *Journal of Applied probability* 2(1):186–205.

Dayarian I, Savelsbergh M (2020) Crowdshipping and same-day delivery: Employing in-store customers to deliver online orders. *Production and Operations Management* 29(9):2153–2174.

Desaulniers G, Desrosiers J, Solomon MM (2006) *Column generation*, volume 5 (Springer Science & Business Media).

Dolan ED, Moré JJ (2002) Benchmarking optimization software with performance profiles. *Mathematical programming* 91:201–213.

Errico F, Desaulniers G, Gendreau M, Rei W, Rousseau LM (2018) The vehicle routing problem with hard time windows and stochastic service times. *EURO Journal on Transportation and Logistics* 7(3):223–251.

Fatehi S, Wagner MR (2022) Crowdsourcing last-mile deliveries. *Manufacturing & Service Operations Management* 24(2):791–809.

Furuhata M, Dessouky M, Ordóñez F, Brunet ME, Wang X, Koenig S (2013) Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological* 57:28–46.

Garnett O, Mandelbaum A, Reiman M (2002) Designing a call center with impatient customers. *Manufacturing & Service Operations Management* 4(3):208–227.

Gdowska K, Viana A, Pedroso JP (2018) Stochastic last-mile delivery with crowdshipping. *Transportation research procedia* 30:90–100.

Gosavi AA, Das SK, Murray SL (2014) Beyond exponential utility functions: A variance-adjusted approach for risk-averse reinforcement learning. *2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL)*, 1–8 (IEEE).

Hall NG, Long DZ, Qi J, Sim M (2015) Managing underperformance risk in project portfolio selection. *Operations Research* 63(3):660–675.

Hansen P, Mladenović N, Brimberg J, Pérez JAM (2019) *Variable Neighborhood Search*, 57–97 (Cham: Springer International Publishing), ISBN 978-3-319-91086-4.

Hao Z, He L, Hu Z, Jiang J (2020) Robust vehicle pre-allocation with uncertain covariates. *Production and Operations Management* 29(4):955–972.

IBM CPLEX (2023) *IBM ILOG CPLEX 22.1.0.0 callable library*.

Irnich S, Desaulniers G (2005) Shortest path problems with resource constraints. *Column generation*, 33–65 (Springer).

Jaillet P, Qi J, Sim M (2016) Routing optimization under uncertainty. *Operations research* 64(1):186–200.

Jepsen M, Petersen B, Spoorendonk S, Pisinger D (2008) Subset-row inequalities applied to the vehicle-routing problem with time windows. *Operations Research* 56(2):497–511, ISSN 0030-364X.

Kafle N, Zou B, Lin J (2017) Design and modeling of a crowdsource-enabled system for urban parcel relay and delivery. *Transportation research part B: methodological* 99:62–82.

Kenyon AS, Morton DP (2003) Stochastic vehicle routing with random travel times. *Transportation Science* 37(1):69–82.

Kızıl KU, Yıldız B (2022) Public transport-based crowd-shipping with backup transfers. *Transportation Science* .

Kytöjoki J, Nuortio T, Bräysy O, Gendreau M (2007) An efficient variable neighborhood search heuristic for very large scale vehicle routing problems. *Computers & Operations Research* 34(9):2743–2757.

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

Lim A, Zhang Z, Qin H (2017) Pickup and delivery service with manpower planning in Hong Kong public hospitals. *Transportation Science* 51(2):688–705.

Lübbecke ME, Desrosiers J (2005) Selected topics in column generation. *Operations research* 53(6):1007–1023.

Macrina G, Pugliese LDP, Guerriero F, Laporte G (2020) Crowd-shipping with time windows and transshipment nodes. *Computers & Operations Research* 113:104806.

Mancini S, Gansterer M (2021) Vehicle routing with private and shared delivery locations. *Computers & Operations Research* 133:105361.

Mihatsch O, Neuneier R (2002) Risk-sensitive reinforcement learning. *Machine learning* 49:267–290.

Mousavi K, Bodur M, Roorda MJ (2022) Stochastic last-mile delivery with crowd-shipping and mobile depots. *Transportation Science* 56(3):612–630.

Pecin D, Contardo C, Desaulniers G, Uchoa E (2017a) New enhancements for the exact solution of the vehicle routing problem with time windows. *INFORMS Journal on Computing* 29(3):489–502.

Pecin D, Pessoa A, Poggi M, Uchoa E (2017b) Improved branch-cut-and-price for capacitated vehicle routing. *Mathematical Programming Computation* 9(1):61–100.

Pecin D, Pessoa A, Poggi M, Uchoa E, Santos H (2017c) Limited memory rank-1 cuts for vehicle routing problems. *Operations Research Letters* 45(3):206–209.

Pugliese LDP, Ferone D, Festa P, Guerriero F, Macrina G (2022a) Solution approaches for the vehicle routing problem with occasional drivers and time windows. *Optimization Methods and Software* 37(4):1384–1414.

Pugliese LDP, Ferone D, Macrina G, Festa P, Guerriero F (2022b) The crowd-shipping with penalty cost function and uncertain travel times. *Omega* 102776.

Punel A, Stathopoulos A (2017) Modeling the acceptability of crowdsourced goods deliveries: Role of context and experience effects. *Transportation Research Part E: Logistics and Transportation Review* 105:18–38.

Sadykov R, Uchoa E, Pessoa A (2021) A bucket graph–based labeling algorithm with application to vehicle routing. *Transportation Science* 55(1):4–28.

Sampaio A, Savelsbergh M, Veelenturf LP, Van Woensel T (2020) Delivery systems with crowd-sourced drivers: A pickup and delivery problem with transfers. *Networks* 76(2):232–255.

Solomon MM (1987) Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research* 35(2):254–265.

Taş D, Gendreau M, Dellaert N, van Woensel T, de Kok A (2014) Vehicle routing with soft time windows and stochastic travel times: A column generation and branch-and-price solution approach. *European Journal of Operational Research* 236(3):789–799.

Torres F, Gendreau M, Rei W (2022a) Crowdshipping: An open vrp variant with stochastic destinations. *Transportation Research Part C: Emerging Technologies* 140:103677.

Torres F, Gendreau M, Rei W (2022b) Vehicle routing with stochastic supply of crowd vehicles and time windows. *Transportation Science* 56(3):631–653.

Ulmer MW (2020) Dynamic pricing and routing for same-day delivery. *Transportation Science* 54(4):1016–1033.

Wei L, Zhang Z, Zhang D, Lim A (2015) A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints. *European Journal of Operational Research* 243(3):798–814.

Xie J, Loke GG, Sim M, Lam SW (2023) The analytics of bed shortages: Coherent metric, prediction, and optimization. *Operations Research* 71(1):23–46.

Yıldız B (2021) Express package routing problem with occasional couriers. *Transportation Research Part C: Emerging Technologies* 123:102994.

Zhang Y, Zhang Z, Lim A, Sim M (2021) Robust data-driven vehicle routing with time windows. *Operations Research* 69(2):469–485.

Zhen L, Wu Y, Wang S, Yi W (2021) Crowdsourcing mode evaluation for parcel delivery service platforms. *International Journal of Production Economics* 235:108067.

This page is intentionally blank. Proper e-companion title page, with INFORMS branding and exact metadata of the main paper, will be produced by the INFORMS office when the issue is being assembled.

Additional Details About Algorithm 1 and the Computational Experiments

EC.1. Preprocessing on Algorithm 1

This section reports our approach to address the numerical overflow issue caused by the exponential function. We first consider the preprocessing on the computation of OCP in the following.

$$\begin{aligned} & \psi(\alpha) = \frac{1}{N}\sum_{\omega\in\Omega}\exp\left(\xi^{\omega}\right)\mathbb{I}[\xi^{\omega}>0] \leq \gamma \\ \Longleftrightarrow \quad & \log\left(\sum_{\omega\in\Omega}\exp(\xi^{\omega})\mathbb{I}[\xi^{\omega}>0]\right) \leq \log(\gamma N) \\ \Longleftrightarrow \quad & \log\left(\exp(\xi^{*})\sum_{\omega\in\Omega}\exp(\xi^{\omega}-\xi^{*})\mathbb{I}[\xi^{\omega}>0]\right) \leq \log(\gamma N) \\ \Longleftrightarrow \quad & \xi^{*}+\log\left(\sum_{\omega\in\Omega}\exp(\xi^{\omega}-\xi^{*})\mathbb{I}[\xi^{\omega}>0]\right) \leq \log(\gamma N). \end{aligned} \tag{EC.1}$$

where $\xi^{*} = \max_{\omega\in\Omega}\{\xi^{\omega}\}$. A similar preprocessing can be applied to the expected exponential lateness $\rho_{E}(\cdot)$.

$$\begin{aligned} & \frac{1}{N}\sum_{\omega\in\Omega}\exp(\xi^{\omega})\mathbb{I}[\xi_{j}^{\omega}>0] \\ \Longleftrightarrow \quad & \frac{1}{N}\log\left(\sum_{\omega\in\Omega}\exp(\xi^{\omega})\mathbb{I}[\xi^{\omega}>0]\right) \\ \Longleftrightarrow \quad & \frac{1}{N}\log\left(\exp(\xi^{*})\sum_{\omega\in\Omega}\exp(\xi^{\omega}-\xi^{*})\mathbb{I}[\xi^{\omega}>0]\right) \\ \Longleftrightarrow \quad & \frac{1}{N}\left(\xi^{*}+\log\left(\sum_{\omega\in\Omega}\exp(\xi^{\omega}-\xi^{*})\mathbb{I}[\xi^{\omega}>0]\right)\right). \end{aligned} \tag{EC.2}$$

EC.2. Additional Details about the Computational Results on solving DM

This section gives detailed results on Set A and B of Instances in solving formulation DM. Tables EC.1-EC.6 show the following columns. Specifically, columns instance, obj, $\#RD$, $\#OD$, $\#CRD$, $\#COD$, $\#ACRD$, $\#ACOD$, and t_{tot} represent the instance name, optimal objective value, number of utilized RDs, number of utilized ODs, number of customers served by RDs, number of customers served by ODs, number of customers served by an RD, number of customers served by an OD, and total computing time, respectively.

EC.3. Additional Details about the Computational Results on Solving SM

This section gives detailed results on solving formulation SM-P, SM-T, and SM-E. In addition to the notations already introduced in Table 4, column "Model" shows the lateness measure variant in Table EC.7.

Table EC.1 Detailed results on small-size Set A instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
005_C101C5	183.4	2	2	3	2	1.5	1.0	0.0
005_C103C5	111.4	1	3	2	3	2.0	1.0	0.0
005_C206C5	174.0	1	2	3	2	3.0	1.0	0.0
005_C208C5	130.0	1	2	3	2	3.0	1.0	0.0
005_R104C5	110.0	1	2	3	2	3.0	1.0	0.0
005_R105C5	147.1	2	1	4	1	2.0	1.0	0.0
005_R202C5	131.9	2	1	4	1	2.0	1.0	0.0
005_R203C5	172.5	1	2	3	2	3.0	1.0	0.0
005_RC105C5	158.9	2	2	3	2	1.5	1.0	0.0
005_RC108C5	208.8	1	2	3	2	3.0	1.0	0.0
005_RC204C5	103.5	1	2	3	2	3.0	1.0	0.0
005_RC208C5	121.0	1	2	3	2	3.0	1.0	0.0
Avg	146.0	1.3	1.9	3.1	1.9	2.5	1.0	0.0

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
010_C101C10	307.9	3	3	7	3	2.3	1.0	0.0
010_C104C10	243.3	2	3	7	3	3.5	1.0	0.1
010_C202C10	174.1	2	3	7	3	3.5	1.0	0.0
010_C205C10	182.7	2	2	8	2	4.0	1.0	0.0
010_R102C10	208.4	2	2	8	2	4.0	1.0	0.0
010_R103C10	164.4	2	2	8	2	4.0	1.0	0.0
010_R201C10	186.3	2	2	8	2	4.0	1.0	0.0
010_R203C10	143.9	1	3	7	3	7.0	1.0	0.0
010_RC102C10	358.2	3	2	8	2	2.7	1.0	0.0
010_RC108C10	336.1	2	2	8	2	4.0	1.0	0.0
010_RC201C10	239.6	2	2	8	2	4.0	1.0	0.0
010_RC205C10	271.5	2	2	8	2	4.0	1.0	0.0
Avg	234.7	2.1	2.3	7.7	2.3	3.9	1.0	0.0

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
015_C103C15	268.9	2	5	10	5	5.0	1.0	0.2
015_C106C15	169.4	2	5	10	5	5.0	1.0	0.0
015_C202C15	344.0	3	4	11	4	3.7	1.0	0.1
015_C208C15	294.2	3	3	12	3	4.0	1.0	0.0
015_R102C15	316.6	3	5	10	5	3.3	1.0	0.0
015_R105C15	226.7	2	5	10	5	5.0	1.0	0.0
015_R202C15	322.2	3	4	11	4	3.7	1.0	0.0
015_R209C15	255.7	3	4	11	4	3.7	1.0	0.4
015_RC103C15	336.7	3	3	12	3	4.0	1.0	0.0
015_RC108C15	274.7	3	5	10	5	3.3	1.0	0.0
015_RC202C15	346.3	3	5	10	5	3.3	1.0	0.0
015_RC204C15	350.6	3	0	15	0	5.0	0.0	64.3
Avg	292.2	2.8	4.0	11.0	4.0	4.1	0.9	5.4

Table EC.2 Detailed results on medium-size Set A instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
025_C101C25	254.6	4	6	19	6	4.8	1.0	0.1
025_C102C25	263.4	4	7	18	7	4.5	1.0	0.1
025_C103C25	269.7	4	4	21	4	5.3	1.0	10.2
025_C104C25	261.9	4	3	22	3	5.5	1.0	370.8
025_C105C25	296.7	5	2	23	2	4.6	1.0	1.0
025_C201C25	211.8	1	4	21	4	21.0	1.0	0.2
025_C202C25	210.0	2	2	23	2	11.5	1.0	6.9
025_C203C25	192.5	1	6	19	6	19.0	1.0	2.4
025_C204C25	184.6	1	6	19	6	19.0	1.0	5.4
025_C205C25	212.0	2	4	21	4	10.5	1.0	1.1
025_R101C25	378.4	4	10	15	10	3.8	1.0	0.1
025_R102C25	312.3	3	10	15	10	5.0	1.0	0.7
025_R103C25	316.4	3	9	16	9	5.3	1.0	0.2
025_R104C25	308.1	3	10	15	10	5.0	1.0	9.2
025_R105C25	316.9	3	9	16	9	5.3	1.0	0.1
025_R201C25	335.0	3	9	16	9	5.3	1.0	0.1
025_R202C25	306.8	3	10	15	10	5.0	1.0	0.6
025_R203C25	283.8	1	10	15	10	15.0	1.0	0.4
025_R204C25	271.2	2	10	15	10	7.5	1.0	0.2
025_R205C25	276.8	2	9	16	9	8.0	1.0	0.1
025_RC101C25	492.0	5	4	21	4	4.2	1.0	0.1
025_RC102C25	506.6	5	3	22	3	4.4	1.0	407.4
025_RC103C25	440.9	5	6	19	6	3.8	1.0	2.1
025_RC104C25	438.4	5	6	19	6	3.8	1.0	23.8
025_RC105C25	529.5	5	3	22	3	4.4	1.0	3.5
025_RC201C25	339.2	3	2	23	2	7.7	1.0	0.1
025_RC202C25	309.8	3	5	20	5	6.7	1.0	1.3
025_RC203C25	311.2	2	2	23	2	11.5	1.0	0.7
025_RC204C25	239.8	1	4	21	4	21.0	1.0	89.9
025_RC205C25	322.5	3	4	21	4	7.0	1.0	0.3
Avg	313.1	3.1	6.0	19.0	6.0	8.2	1.0	31.3

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
050_C101C50	372.8	4	12	38	12	9.5	1.0	20.1
050_C103C50	318.6	4	13	37	13	9.3	1.0	201.4
050_C105C50	359.7	4	13	37	13	9.3	1.0	77.4
050_C201C50	312.5	2	9	41	9	20.5	1.0	12.8
050_C202C50	285.3	2	11	39	11	19.5	1.0	11.7
050_C203C50	261.7	2	15	35	15	17.5	1.0	50.6
050_C205C50	306.8	2	12	38	12	19.0	1.0	19.3
050_R101C50	704.7	8	15	35	15	4.4	1.0	0.6
050_R102C50	611.5	7	15	35	15	5.0	1.0	4.1
050_R103C50	477.6	5	15	35	15	7.0	1.0	25.2
050_R105C50	607.5	6	15	35	15	5.8	1.0	2.9
050_R201C50	565.9	4	15	35	15	8.8	1.0	3.4
050_R202C50	500.5	2	15	35	15	17.5	1.0	5818.7
050_R203C50	442.8	3	15	35	15	11.7	1.0	2288.6
050_R205C50	490.3	3	15	35	15	11.7	1.0	139.8
050_RC101C50	625.6	5	12	38	12	7.6	1.0	1.7
050_RC102C50	614.9	5	10	40	10	8.0	1.0	5.2
050_RC103C50	539.4	5	6	44	6	8.8	1.0	413.6
050_RC104C50	526.4	5	5	45	5	9.0	1.0	356.3
050_RC105C50	583.2	5	6	44	6	8.8	1.0	1.1
050_RC201C50	637.6	5	8	42	8	8.4	1.0	4.5
050_RC202C50	551.3	4	7	43	7	10.8	1.0	79.1
050_RC203C50	489.4	3	8	42	8	14.0	1.0	1158.3
050_RC205C50	542.4	4	9	41	9	10.3	1.0	5.9
Avg	488.7	4.1	11.5	38.5	11.5	10.9	1.0	445.9

Table EC.3 Detailed results in large-size Set A instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
100_C101	723.0	8	29	71	29	8.9	1.0	2481.3
100_C201	512.5	3	21	79	21	26.3	1.0	21.5
100_R101	1162.8	10	27	73	27	7.3	1.0	21.8
100_R201	772.1	5	30	70	30	14.0	1.0	161.7
100_RC101	1005.7	9	30	70	30	7.8	1.0	4367.0
100_RC102	925.1	8	30	70	30	8.8	1.0	2789.3
100_RC105	845.0	8	30	70	30	8.8	1.0	6116.0
100_RC201	884.3	6	28	72	28	12.0	1.0	36.5
100_RC202	684.3	4	30	70	30	17.5	1.0	465.1
Avg	835.0	6.8	28.3	71.7	28.3	12.4	1.0	1828.9

Table EC.4 Detailed results on small-size Set B instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
005_C101C5	183.4	2	2	3	2	1.5	1.0	0.0
005_C103C5	111.4	1	2	2	3	2.0	1.5	0.0
005_C206C5	174.0	1	2	3	2	3.0	1.0	0.0
005_C208C5	130.0	1	2	3	2	3.0	1.0	0.0
005_R104C5	110.0	1	2	3	2	3.0	1.0	0.0
005_R105C5	147.1	2	1	4	1	2.0	1.0	0.0
005_R202C5	131.9	2	1	4	1	2.0	1.0	0.0
005_R203C5	172.5	1	2	3	2	3.0	1.0	0.0
005_RC105C5	158.9	2	2	3	2	1.5	1.0	0.0
005_RC108C5	164.0	1	2	2	3	2.0	1.5	0.0
005_RC204C5	103.5	1	2	3	2	3.0	1.0	0.0
005_RC208C5	121.0	1	2	3	2	3.0	1.0	0.0
Avg	142.3	1.3	1.8	3.0	2.0	2.4	1.1	0.0

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
010_C101C10	302.9	3	3	6	4	2.0	1.3	0.1
010_C104C10	237.4	2	3	6	4	3.0	1.3	0.0
010_C202C10	169.2	2	3	6	4	3.0	1.3	0.0
010_C205C10	182.7	2	2	8	2	4.0	1.0	0.0
010_R102C10	189.0	2	2	7	3	3.5	1.5	0.0
010_R103C10	160.5	2	2	7	3	3.5	1.5	0.2
010_R201C10	186.3	2	2	8	2	4.0	1.0	0.0
010_R203C10	126.4	1	3	6	4	6.0	1.3	0.0
010_RC102C10	347.6	2	2	7	3	3.5	1.5	0.0
010_RC108C10	336.1	2	2	8	2	4.0	1.0	0.1
010_RC201C10	227.2	2	2	7	3	3.5	1.5	0.0
010_RC205C10	262.2	2	2	7	3	3.5	1.5	0.0
Avg	227.3	2.0	2.3	6.9	3.1	3.6	1.3	0.0

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
015_C103C15	202.2	1	5	7	8	7.0	1.6	0.0
015_C106C15	149.4	2	5	8	7	4.0	1.4	0.0
015_C202C15	327.8	3	4	9	6	3.0	1.5	0.1
015_C208C15	294.2	3	3	12	3	4.0	1.0	0.0
015_R102C15	304.0	3	5	9	6	3.0	1.2	0.0
015_R105C15	209.7	2	5	9	6	4.5	1.2	0.0
015_R202C15	322.2	3	4	11	4	3.7	1.0	0.0
015_R209C15	236.9	3	3	11	4	3.7	1.3	0.1
015_RC103C15	336.7	3	3	12	3	4.0	1.0	0.0
015_RC108C15	236.4	2	5	8	7	4.0	1.4	0.1
015_RC202C15	343.2	3	5	9	6	3.0	1.2	0.0
015_RC204C15	338.1	2	3	11	4	5.5	1.3	3.7
Avg	275.1	2.5	4.2	9.7	5.3	4.1	1.3	0.3

Table EC.5 Detailed results on medium-size Set B instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
025_C101C25	252.7	4	5	19	6	4.8	1.2	1.5
025_C102C25	260.6	4	7	17	8	4.3	1.1	0.1
025_C103C25	256.9	4	4	19	6	4.8	1.5	0.1
025_C104C25	252.6	4	3	20	5	5.0	1.7	935.5
025_C105C25	291.2	4	4	20	5	5.0	1.3	0.2
025_C201C25	210.8	1	3	21	4	21.0	1.3	0.2
025_C202C25	195.6	1	3	21	4	21.0	1.3	2.2
025_C203C25	185.8	1	6	18	7	18.0	1.2	1.8
025_C204C25	181.2	1	6	18	7	18.0	1.2	13.9
025_C205C25	212.0	2	4	21	4	10.5	1.0	0.4
025_R101C25	297.1	3	10	9	16	3.0	1.6	0.0
025_R102C25	269.7	2	10	7	18	3.5	1.8	2.8
025_R103C25	292.6	2	10	11	14	5.5	1.4	0.2
025_R104C25	272.9	2	9	13	12	6.5	1.3	0.1
025_R105C25	277.1	3	10	11	14	3.7	1.4	0.1
025_R201C25	319.6	2	9	10	15	5.0	1.7	3.3
025_R202C25	285.4	2	8	13	12	6.5	1.5	0.1
025_R203C25	238.9	1	9	10	15	10.0	1.7	1.4
025_R204C25	262.4	2	8	13	12	6.5	1.5	1.5
025_R205C25	252.7	1	9	14	11	14.0	1.2	0.2
025_RC101C25	455.9	5	3	21	4	4.2	1.3	0.0
025_RC102C25	460.7	5	3	20	5	4.0	1.7	0.5
025_RC103C25	425.5	5	5	18	7	3.6	1.4	6.4
025_RC104C25	427.2	4	7	16	9	4.0	1.3	60.1
025_RC105C25	529.5	5	3	22	3	4.4	1.0	7.3
025_RC201C25	337.4	3	2	22	3	7.3	1.5	0.1
025_RC202C25	299.5	2	5	18	7	9.0	1.4	0.9
025_RC203C25	310.5	2	1	23	2	11.5	2.0	0.7
025_RC204C25	239.6	1	3	21	4	21.0	1.3	38.6
025_RC205C25	318.2	2	5	19	6	9.5	1.2	0.6
Avg	295.7	2.7	5.8	16.8	8.2	8.5	1.4	36.0

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
050_C101C50	312.8	3	12	27	23	9.0	1.9	3.3
050_C103C50	292.1	3	12	29	21	9.7	1.8	383.0
050_C105C50	306.5	3	13	24	26	8.0	2.0	86.6
050_C201C50	292.6	1	13	29	21	29.0	1.6	47.4
050_C202C50	254.9	1	11	29	21	29.0	1.9	6.3
050_C203C50	250.8	2	14	31	19	15.5	1.4	28.9
050_C204C50	218.7	1	13	27	23	27.0	1.8	272.7
050_C205C50	299.1	2	12	35	15	17.5	1.3	22.4
050_R101C50	581.0	6	15	21	29	3.5	1.9	1.0
050_R102C50	503.0	4	15	20	30	5.0	2.0	144.9
050_R103C50	387.5	3	15	19	31	6.3	2.1	253.6
050_R105C50	503.8	4	15	22	28	5.5	1.9	0.5
050_R201C50	482.4	4	14	27	23	6.8	1.6	12.9
050_R202C50	410.4	2	15	24	26	12.0	1.7	26.3
050_R203C50	389.6	2	14	27	23	13.5	1.6	368.8
050_R205C50	397.4	2	15	24	26	12.0	1.7	31.8
050_RC101C50	511.9	3	15	24	26	8.0	1.7	217.8
050_RC102C50	532.2	3	15	28	22	9.3	1.5	285.8
050_RC103C50	472.6	4	11	32	18	8.0	1.6	200.4
050_RC104C50	448.7	3	14	26	24	8.7	1.7	1784.1
050_RC105C50	540.6	4	10	34	16	8.5	1.6	2262.0
050_RC201C50	606.1	4	12	33	17	8.3	1.4	118.4
050_RC202C50	524.8	4	6	40	10	10.0	1.7	35.6
050_RC205C50	518.1	3	11	34	16	11.3	1.5	89.6
Avg	418.2	3.0	13.0	27.8	22.3	11.7	1.7	278.5

Table EC.6 Detailed results on large-size Set B instances

instance	obj	$\#RD$	$\#OD$	$\#CRD$	$\#COD$	$\#ACRD$	$\#ACOD$	t_{tot}
100_C105	580.4	5	29	47	53	9.4	1.8	482.0
100_R101	873.6	9	27	51	49	5.7	1.8	4.0
100_R105	693.6	6	28	44	56	7.3	2.0	2438.9
100_RC201	760.4	4	29	52	48	13.0	1.7	1183.1
Avg	727.0	6	28.25	48.5	51.5	8.9	1.8	1027.0

Table EC.7 Details results on soving SM

$\|\mathcal{N}\|$	Model	$\#opt/\#tot$	B	gap	obj	$cost$	t_{PH}	$\|\overline{\mathcal{R}}\|$	$\|\overline{\mathcal{R}}_{RD}\|$	$\|\overline{\mathcal{R}}_{OD}\|$	t_{obj}	t_{IP}	t_{RE}	t_{tot}
5	SM-E	6/6	151.6	6.8	0.0	145.8	0.0	9.5	5.8	3.7	0.0	0.0	0.0	0.5
	SM-P	6/6	151.6	6.8	0.0	145.8	0.0	9.5	5.8	3.7	0.0	0.0	0.0	0.6
	SM-T	6/6	151.6	6.8	0.0	145.8	0.0	9.5	5.8	3.7	0.0	0.0	0.0	0.5
10	SM-E	6/6	272.7	17.0	0.1	267.3	0.1	143.2	122.0	21.2	0.0	0.0	0.1	0.2
	SM-P	6/6	272.7	17.0	0.0	267.7	0.1	143.0	121.8	21.2	0.0	0.0	0.1	0.2
	SM-T	6/6	272.7	17.0	0.3	267.7	0.1	143.2	122.0	21.2	0.0	0.0	0.0	0.2
15	SM-E	6/6	249.3	11.9	0.0	246.9	0.8	134.8	106.8	28.0	0.0	0.0	0.1	0.3
	SM-P	6/6	249.3	11.9	0.1	247.0	0.9	134.5	106.5	28.0	0.0	0.0	0.1	0.3
	SM-T	6/6	249.3	11.9	0.4	247.3	0.8	135.0	107.0	28.0	0.0	0.0	0.1	0.3
25	SM-E	13/15	359.7	18.6	0.0	357.2	6.1	3558.5	3397.0	161.5	0.0	2.6	5.8	8.5
	SM-P	13/15	359.7	18.6	0.1	357.6	5.5	3558.2	3396.7	161.5	0.0	4.5	5.8	10.5
	SM-T	13/15	359.7	18.6	0.4	356.6	5.4	3559.6	3398.2	161.5	0.0	0.3	5.5	6.0
50	SM-E	7/15	517.8	32.4	0.3	517.0	67.7	41403.4	38113.6	3289.9	0.0	18.8	482.7	502.1
	SM-P	7/15	517.8	32.4	0.6	517.4	60.2	41566.6	38276.7	3289.9	0.0	43.0	413.2	456.8
	SM-T	7/15	517.8	32.4	2.4	516.7	65.8	41508.9	38219.0	3289.9	0.0	36.5	551.6	588.7