Stochastic Programming Models for a Fleet Sizing and Appointment Scheduling Problem with Random Service and Travel Times

Shutian Li[1], Karmel S. Shehadeh[1,*], Man Yiu Tsang[1]

[a] *Department of Industrial and Systems Engineering, Lehigh University, Bethlehem, PA, USA*

Abstract

We propose a new stochastic mixed-integer linear programming model for a *home service fleet sizing and appointment scheduling problem* (HFASP) with random service and travel times. Specifically, given a set of providers and a set of geographically distributed customers within a service region, our model solves the following problems simultaneously: (i) a *fleet sizing problem* that determines the number of providers required to serve customers; (ii) an *assignment problem* that assigns customers to providers; and (iii) a *sequencing and scheduling problem* that decides the sequence of appointment times of customers assigned to each provider. The objective is to minimize the fixed cost of hiring providers plus the expectation of a weighted sum of customers' waiting time and providers' travel time, overtime, and idle time. We compare our proposed model with an extension of an existing model for a closely related problem in the literature, theoretically and empirically. Specifically, we show that our newly proposed model is more compact (i.e., has fewer variables and constraints) and provides a tighter linear programming relaxation. Furthermore, to handle large instances observed in other application domains, we propose two optimization-based heuristics that decompose the HFASP decision process into two steps. The first step involves determining fleet sizing and assignment decisions, and the second constructs a routing plan and a schedule for each provider. We present extensive computational results to show the size and characteristics of HFASP instances that can be solved with our proposed model, demonstrating its computational efficiency over the extension. Results also show that the proposed heuristics can quickly produce high-quality solutions to large instances with an optimality gap not exceeding 5% on tested instances. Finally, we use a case study based on a service region in Lehigh County to derive insights into the HFASP.

Keywords: Fleet sizing; scheduling; routing; home services; stochastic programming

[*] Corresponding author.

Email addresses: `shl919@lehigh.edu` (Shutian Li), `kas720@lehigh.edu` (Karmel S. Shehadeh), `mat420@lehigh.edu` (Man Yiu Tsang)

Preprint submitted to Optimization Online *December 21, 2023*

1. Introduction

Home service agencies provide a wide range of services to customers at their homes, including health care, beauty treatments, fitness training, housekeeping, appliance repair service, and babysitting. The increasingly aging populations, busy lifestyles, extended work hours, and the outspread of infectious and chronic diseases have led to a substantial increase in demand for home services, especially home health care (Fikar and Hirsch, 2017; Alkaabneh et al., 2023). The cost of various home services has also been increasing in recent years. For example, 88% of home service companies raised their service prices in the past two years. The United States home service market is projected to reach $1219 Billion by 2026. With competitive pressure from the market, home service providers face the challenge of improving service quality and reducing operational costs. This motivates the need for computationally efficient optimization models for home service planning and scheduling.

Like traditional delivery services, home services require a professional provider (or service team) to travel and deliver services to geographically distributed customers. In contrast to most delivery services, however, customers must present to receive their service (Zhan and Wan, 2018). In addition, home service operators assign a provider and quote an appointment time (i.e., planned service start time) to each customer in advance to avoid delivery failure. Then, on the day of service, each provider visits customers assigned to them one by one.

In this paper, we address a *home service fleet sizing and appointment scheduling problem* (HFASP) under stochastic service and travel times. Specifically, given a set of providers and a set of customers within a service region, the HFASP consists of solving the following decision-making problems simultaneously: (i) a *fleet sizing problem* that determines the number of providers required to serve customers; (ii) an *assignment problem* that assigns customers to providers; and (iii) a *sequencing and scheduling problem* that decides the sequence of appointment times of customers assigned to each provider. Here, a sequence of customers assigned to a provider is equivalent to the provider's *route*. Each provider must start from the main office (i.e., depot) and visit each customer in their schedule one by one before returning to the main office. If a provider arrives at a customer's location before the scheduled service start time, the provider must wait (i.e., remains idle) until the scheduled service start time. Conversely, the customer must wait if the provider arrives after the scheduled appointment time. Moreover, each provider has a fixed service hour beyond which s/he experiences overtime. Given the uncertainty of service time and travel time between customers, the goal is to minimize the fixed cost of establishing the providers' fleet (i.e., provider hiring or labor cost) and the expectation of a weighted sum of customers' waiting time and providers' idle time, overtime, and travel time.

The HFASP is a challenging stochastic optimization problem for various reasons, foremost of which are the following. First, suppose we fix the number of providers and customers' appointment times. In this case, the HFASP becomes similar to the multiple vehicle routing problem with time constraints and stochastic travel time, which is a challenging optimization problem (Cook, 2011; Toth and Vigo, 2014). Second, suppose we fix the number of providers and customer assignments. In this case, the HFASP reduces to a multi-server sequencing and scheduling problem with stochastic service time, which is another well-known complex optimization problem (Denton et al., 2007).

To address service time uncertainty, Zhan and Wan (2018) proposed the first two-stage stochastic mixed-integer program (SMIP) for a closely related problem to the HFASP. In the first stage, the model decides the number of providers and their routing and scheduling decisions using traditional routing variables and constraints. Then, in the second stage, the model computes providers' overtime and customers' waiting time. The objective of this model is to minimize the fixed cost of hiring providers and their total travel time plus a weighted sum of providers' overtime and customers' waiting time. Note that Zhan and Wan (2018) assumed that the travel time is deterministic and ignored providers' idle time. Ignoring random travel time may lead to sub-optimal solutions with excessive customers' waiting time and providers' overtime, consequently impacting service quality. Ignoring idle time may yield the underutilization of providers' time. Finally, Zhan and Wan (2018)'s results indicate that their model is challenging to solve.

In this paper, we propose a new two-stage SMIP for the HFASP, denoted as model (S). In contrast to Zhan and Wan (2018), our model incorporates both random travel and service times. In addition, our second stage includes variables and constraints to compute providers' idle time. Furthermore, instead of using traditional routing variables and constraints, we use sequencing variables and constraints to determine the order of customers assigned to each provider (equivalently, the provider's route). We also derive an extension of Zhan and Wan (2018)'s model (denoted as model (Z)) that incorporates both random travel and service times and providers' idle time. We rigorously analyze the relative strengths and weaknesses of the two proposed models, theoretically and empirically. Specifically, we show that our newly proposed model (S) is more compact (i.e., has fewer variables and constraints), provides a tighter linear programming (LP) relaxation, and is more computationally efficient. Furthermore, to handle large instances observed in other application domains, we propose two optimization-based heuristics that decompose the HFASP decision process into two steps. The first involves determining fleet sizing and assignment decisions, and the second constructs a routing plan and a schedule for each provider.

Finally, we conduct extensive computational experiments to show the size and characteristics of problem instances that can be solved with our proposed model (S), demonstrating the significant computational performance improvements that can be gained with model (S). Specifically, our results show that model (S) can solve larger instances faster and within a reasonable time than model (Z). In addition, our results show that the proposed heuristics can quickly produce high-quality solutions to large instances with an optimality gap not exceeding 5% on tested instances. Finally, we use a case study based on a service region in Lehigh County, Pennsylvania, to derive insights into the HFASP.

1.1. Structure of the paper

The remainder of this paper is organized as follows. In Section 2, we review the relevant literature. In Section 3, we describe our problem setting. In Section 4, we introduce our proposed model (S) and model (Z) for the HFASP. In Section 5, we provide theoretical analyses of the proposed models. In Section 6, we present our two optimization-based heuristics. In Section 7, we present computational results. Finally, we draw conclusions in Section 8.

2. Relevant Literature

Planning and scheduling problems in home services have received significant attention. We refer to Fikar and Hirsch (2017); Cissé et al. (2017); Grieco et al. (2021); Di Mascolo et al. (2021) for comprehensive surveys on home service problems and applications. Next, we discuss relevant studies to our work.

Determining the number of providers (fleet size) is critical because it is a major fixed investment for home service companies. An inappropriate fleet size may lead to poor operational performance (e.g., long travel time and excessive delays and overtime) and, consequently, poor service quality (Shehadeh, 2023). Note that the fleet sizing problem depends on the service team's operational performance, and the latter depends on the routing and scheduling decisions. Thus, it is important to integrate fleet sizing, assignment, scheduling, and routing problems. However, most of the existing approaches for home service planning focus on a subset of these problems. For example, Restrepo et al. (2020) focus on staff dimensioning and scheduling in home care services. They take into account the uncertainty of customers' demands. Other studies proposed models for integrated routing and scheduling (see, e.g., Han et al., 2017; Zhan et al., 2021; Cinar et al., 2021; Liu et al., 2019). Next, we limit the scope of this review to recent literature that is most relevant to our work. Namely, we focus on studies that propose stochastic programming (SP) models for fleet sizing, assignment, routing, and scheduling problems arising from the home service practice.

Zhan and Wan (2018) propose the first two-stage SMIP for vehicle routing and appointment scheduling with team assignment for home services. In the first stage, the model determines the number of providers to use/hire and the route and schedule for each. The model computes teams' overtime and customers' waiting time in the second stage. Zhan and Wan (2018) incorporate service time uncertainty in the model, but they assume that the travel time is deterministic. The objective is to minimize the fixed hiring cost and the total deterministic travel time plus the expected overtime and waiting time. Observing the challenges of solving small instances, they propose a modified parallel saving algorithm to obtain feasible routes and use a tabu-search method to obtain near-optimal solutions. Later, Zhan et al. (2021) propose a two-stage SMIP for a single provider home service routing and appointment scheduling with stochastic service time. The objective is to minimize the providers' travel costs and the expected second-stage cost, including providers' idle time and customers' waiting time. They exploit the structural properties of the proposed model and develop an L-shaped method to solve problem instances with six, eight, and ten customers.

Note that ignoring travel time uncertainty and idle cost as in Zhan and Wan (2018) and Zhan et al. (2021) may lead to sub-optimal solutions with, for example, excessive idle time, overtime, and travel time. Recent studies that incorporate stochastic travel time include Shi et al. (2018) and Hashemi Doulabi et al. (2020). Shi et al. (2018) propose an SP model for a home healthcare routing and scheduling problem with random service and travel times. The objective is to minimize the hiring cost plus the expected weighted sum of transportation costs, providers' overtime, and customers' waiting costs. They integrate simulation and the simulated annealing algorithm to solve the model. Hashemi Doulabi et al. (2020) study a vehicle routing problem (VRP) with synchronized visits and propose a two-stage SMIP. They discuss the application of the model in home healthcare scheduling. Given the challenges of solving the proposed model with commercial solvers, they employ the L-shaped algorithm and implement a branch-and-cut method to solve the problem. Moreover, they propose valid inequalities to speed up the convergence. Yu et al. (2021) model a combination of vehicle routing with pick-up and delivery and appointment scheduling as a scenario-based mixed-integer program. They apply the model in the context of medical service routing and scheduling. Yu et al. (2021)'s model aims to minimize the operational cost plus the expected penalty cost of the early/late arrival and extra working duration of vehicles. To solve large instances, they use K-means algorithms to cluster customers into k groups and then make routing and scheduling decisions for each identified group (cluster) of customers.

As mentioned earlier, fixing customers' assignments to providers reduces our problem to a sequencing and scheduling (SAS) problem with stochastic service time. Berg et al. (2014), Mancilla and Storer (2012), and Shehadeh et al. (2019) propose SMIP models for the single-server SAS. The SAS problem with multiple servers has been extensively studied in the healthcare scheduling literature (Gupta and Denton, 2008; Ahmadi-Javid et al., 2017). The HFASP is also related to VRP. We refer to Laporte et al. (1992); Kenyon and Morton (2003) for detailed discussions on formulations and methodologies for various VRP problems under random service and travel times.

Our HFASP has similar characteristics to those addressed in Zhan and Wan (2018) and Zhan et al. (2021). Nevertheless, our HFASP model is different from Zhan and Wan (2018) and Zhan et al. (2021) in the following aspects. First, we consider multiple service teams while Zhan et al. (2021)'s model focuses on the routing and scheduling of one team. Second, we consider both random travel and service times and aim to minimize expected total travel time, while Zhan and Wan (2018) and Zhan et al. (2021) assume that travel time is deterministic. Various vehicle routing studies have motivated the need for hedging against travel time uncertainty to obtain high-quality routing decisions (Anderluh et al., 2020; Lecluyse et al., 2009). Third, Zhan and Wan (2018)'s second stage does not include idle time variables or objectives, and Zhan et al. (2021)'s second stage does not include overtime variables or objectives. Our second stage objective includes overtime, idle time, waiting time, and travel time. Fourth, recognizing that the sequence of customers is equivalent to the provider's route, we use sequencing variables and constraints instead of routing variables and constraints to determine routing decisions. In Section 5, we show that our sequencing-based SMIP is more compact and provides a tighter LP relaxation than an extension of Zhan and Wan (2018) for the HFASP. Moreover, our model can efficiently solve realistic (previously unsolved) HFASP instances (see Section 7.2). Finally, we propose two efficient heuristics that leverage variants of our proposed model and show that they could quickly obtain near-optimal solutions to large instances. Table A1 in Appendix A summarizes the differences between our proposed model and those of Zhan and Wan (2018) and Zhan et al. (2021).

3. Problem Setting

We start by introducing our problem settings. We consider a set of customers P and a set of providers K. Each customer $p \in P$ must be served by exactly one provider. On the other hand, each provider has fixed service hours $[0, L]$, which are long enough to serve multiple customers. The cost of hiring one provider is λ^f. Each hired provider must start from an origin (e.g., the provider's office) and visit each customer on his/her schedule exactly once before returning to the origin. The service time d_p of each customer $p \in P$ and the travel time $t_{p,p'}$ between customers $(p, p') \in P \times P$ are random with known probability distributions. Given sets P and K, we aim to solve the following decision problems simultaneously: (1) a fleet sizing problem that determines the number of providers required to serve customers; (2) an assignment problem that determines the assignment of customers to providers; and (3) a sequencing (routing) and scheduling problem that determines the order and appointment times of customers assigned to each provider. The objective is to minimize the fixed hiring cost plus the expected operational costs associated with customers' waiting time and providers' idle time, overtime, and travel time.

This problem can be formulated as a two-stage SMIP. The first stage contains binary (for assignment and sequencing) and continuous (for scheduling) decision variables. Given the sequence of appointment times decided in the first stage, the second stage problem contains continuous decision variables representing what happens for each realization of service and travel times (i.e., compute waiting time, idle time, travel time, and overtime). To incorporate service and travel time uncertainty into the model, we use a Sample Average Approximation (SAA) approach. That is, we generate a sample of N scenarios (each scenario consists of a vector of realizations of service and travel times which are drawn independently from the distributions corresponding to each customer and pair of customers, respectively), and then optimize the sample average of the objective. We refer to Kim et al. (2015); Kleywegt et al. (2002); Homem-de Mello and Bayraksan (2014) for the technical details and discussions on SAA.

Some companies may require each provider to serve a specific number of customers to utilize their time better and balance their load in terms of the number of customers assigned to each. Other companies do not have such requirements. Accordingly, we consider two types of service providers, namely *fully used* and *partially used* providers. Fully used providers must serve a particular number of customers, which effectively means that each provider's sequence consists of $|I|$ customers [1] (e.g., $|I| = 6$ customers). In this case, the number of providers needed to serve all customers equals $|P|/|I|$, and the problem is reduced to an assignment, sequencing, and scheduling problem with multiple providers.

On the other hand, partially used providers can serve any number of customers but at least one customer (or any other threshold specified by the decision maker). In this case, the model will determine the number of providers to hire and the assignment, sequencing, and scheduling decisions. In Section 4, we present SMIP models for each provider type.

4. SMIP Models for the HFASP

In this section, we propose two SMIP formulations for the HFASP. In Section 4.1, we present our new SMIP model denoted as model (S). In Section 4.2, we derive an extension of Zhan and Wan (2018)'s model denoted as model (Z).

4.1. Model (S)

In this section, we present our proposed model, denoted as model (S). Let us first introduce the variables and parameters defining this model. For all $i \in I, p \in P$, and $k \in K$, we define binary decision variables $x_{i,p,k}$ that equal one if customer p is assigned to the ith position in the sequence of customers assigned to provider k. For all $i \in I$ and $k \in K$, we define non-negative continuous decision variables $a_{i,k}$ to represent the scheduled appointment time of the ith customer in the schedule of provider k.

We define the following scenario-based decision variables to compute waiting time, idle time, and travel time in each scenario $n \in [N]$ of service and travel times. We define non-negative continuous decision variables $s^n_{i,k}$ to represent the actual service start time of the ith customer in provider k's schedule. We define non-negative continuous decision variables $g^n_{i,k}$ to represent provider k's idle time before the ith customer/appointment. We define non-negative continuous decision variables o^n_k to represent provider k's overtime. Finally, we let non-negative parameters λ^w, λ^o, λ^g, and λ^t represent unit penalty cost of waiting, idle time, overtime, and travel time, respectively. A complete list of the parameters and decision variables can be found in Table 1. Using this notation, our SAA model can be stated as follows (see Appendix B for details on the derivation of this model):

[1] Note that serving the same number of customers does not necessarily prevent variation in the actual working hours among providers. Our proposed models do not explicitly address this issue.

Table 1: Notation

Index sets	
P	the set of customers
K	the set of providers
I	the set of positions in the serving sequence
Parameters	
N	the number of scenarios
λ^f	fixed cost of hiring one provider
$\lambda^o/\lambda^g/\lambda^w$	unit overtime/idle time/ waiting time cost
λ^t	unit travel time cost
L	working hours
$t^n_{i,j}$	travel time from node i to node j under scenario n
d^n_i	customer i's service time under scenario n
Deterministic decision variables	
$x_{i,p,k}$	equals one if customer p is assigned to the ith position of provider k's sequence
$a_{i,k}$	scheduled time of ith appointment served by provider k
$z_{i,p,p',k}$	equals one if the customer p follows the customer p' on provider k's serving sequence
Random (scenario-based) variables	
$s^n_{i,k}$	actual start time of ith appointment served by provider k under scenario n
$g^n_{i,k}$	idle time before ith appointment served by provider k under scenario n
o^n_k	overtime of provider k under scenario n

$$\min_{\substack{x,z,a,\\ s,g,o}} \sum_{p\in P}\sum_{k\in K}\lambda^f x_{1,p,k} + \sum_{n\in[N]}\frac{1}{N}\Bigg\{\lambda^t\Bigg[\sum_{k\in K}\Bigg(\sum_{\substack{(p,p')\in P\times P\\ p\neq p'}}\sum_{i\in I} t^n_{p,p'} z_{i,p,p',k} + \sum_{p\in P} t^n_{0,p}x_{1,p,k} + \sum_{p\in P} t^n_{p,0}x_{0,p,k}\Bigg)\Bigg]$$

$$+ \sum_{k\in K}\sum_{i\in I}\Big[\lambda^w(s^n_{i,k}-a_{i,k}) + \lambda^g g^n_{i,k}\Big] + \sum_{k\in K}\lambda^o o^n_k\Bigg\} \tag{1a}$$

$$\text{s.t.} \quad \sum_{i\in I}\sum_{k\in K} x_{i,p,k} = 1, \qquad \forall p\in P, \tag{1b}$$

$$\sum_{p\in P} x_{i,p,k} \leq 1, \qquad \forall i\in I,\, k\in K, \tag{1c}$$

$$x_{0,p,k} \geq x_{i,p,k} - \sum_{p'\in P: p'\neq p} x_{i+1,p',k}, \qquad \forall i\in I, p\in P, k\in K, \tag{1d}$$

$$\sum_{p\in P} x_{i,p,k} \geq \sum_{p\in P} x_{i+1,p,k}, \quad \forall i\in[1,|I|-1]_{\mathbb{Z}}, k\in K, \tag{1e}$$

$$z_{i,p,p',k} \leq x_{i-1,p,k}, \qquad \forall i\in[2,|I|]_{\mathbb{Z}}, (p,p')\in P\times P: p\neq p', k\in K, \tag{1f}$$

$$z_{i,p,p',k} \leq x_{i,p',k}, \qquad \forall i\in[2,|I|]_{\mathbb{Z}}, (p,p')\in P\times P: p\neq p', k\in K, \tag{1g}$$

$$z_{i,p,p',k} \geq x_{i-1,p,k} + x_{i,p',k} - 1, \qquad \forall i\in[2,|I|]_{\mathbb{Z}}, (p,p')\in P\times P: p\neq p', k\in K, \tag{1h}$$

$$a_{i,k} \leq L\sum_{p\in P} x_{i,p,k}, \qquad \forall k\in K, i\in I, \tag{1i}$$

$$s^n_{i,k} \geq a_{i,k}, \qquad \forall i\in I, k\in K, n\in[N], \tag{1j}$$

$$s^n_{1,k} \geq \sum_{p\in P} t^n_{0,p}x_{1,p,k}, \qquad \forall k\in K, n\in[N], \tag{1k}$$

$$s^n_{i,k} \geq s^n_{i-1,k} + \sum_{p \in P} d^n_p x_{i-1,p,k} + \sum_{\substack{(p,p') \in P \\ p \neq p'}} t^n_{p,p'} z_{i,p,p',k} - M_i \left(1 - \sum_{p \in P} x_{i,p,k} \right), \forall i \in [2, |I|]_{\mathbb{Z}}, k, n, \tag{1l}$$

$$g^n_{1,k} \geq s^n_{1,k} - \left(\sum_{p \in P} t^n_{0,p} x_{1,p,k} \right), \qquad \forall k \in K, n \in [N], \tag{1m}$$

$$g^n_{i,k} \geq s^n_{i,k} - \left(s^n_{i-1,k} + \sum_{\substack{(p,p') \in P \times P \\ p \neq p'}} t^n_{p,p'} z_{i,p,p',k} + \sum_{p \in P} d^n_p x_{i-1,p,k} \right), \forall i \in [2, |I|]_{\mathbb{Z}}, k, n, \tag{1n}$$

$$o^n_k \geq \left[s^n_{i,k} + \sum_{p \in P} (d^n_p + t^n_{p,0}) x_{0,p,k} \right] - L, \qquad \forall i \in I, k \in K, n \in [N], \tag{1o}$$

$$(a, s, g, o, z) \geq 0, \qquad x \in \{0, 1\}^{(|I|+2) \times |P| \times |K|}. \tag{1p}$$

Formulation (1) finds optimal sizing, routing, and scheduling decisions that minimize the fixed cost related to establishing the providers fleet or hiring costs (first term), and the sample average of the random operational costs consisting of total waiting time, and providers' idle time, overtime, and travel time. Constraints (1b) ensure that each customer is assigned to exactly one position in the schedule of one provider. Constraints (1c) ensure that at most one customer is assigned to each position in provider k's sequence. Constraints (1d) define the variable $x_{0,p,k}$, which is equal to one if customer p is the last customer in the schedule of provider k, and is zero otherwise. Constraints (1e) prohibit assigning customers to position $(i+1)$ when position i is vacant. Constraints (1f)-(1h) ensure that if customer p' is assigned to the ith position of provider k's sequence and customer p is assigned to the $(i-1)$th position of provider k's sequence, then $z_{i,p,p',k} = 1$, and $z_{i,p,p',k} = 0$ otherwise. It is easy to verify that variables z equal one or zero in any feasible solution satisfying constraints (1f)–(1h). Constraints (1i) ensure that all appointments are scheduled within the provider's service hours. These constraints also ensure that $a_{i,k} = 0$ whenever $x_{i,p,k} = 0, \forall p \in P$, i.e., when position i is empty or provider k is not hired.

For each scenario $n \in [N]$, constraints (1j)-(1l) require that the actual start time of the ith appointment to be no smaller than the scheduled start time and the service completion time of the preceding appointment plus the travel time between the $(i-1)$th and ith customer. Note that, for a sufficiently large M constant, constraints (1l) are relaxed and thus $s^n_{i,k} = 0$ if the ith position is empty. Constraints (1m) compute the idle time before the actual start time of the first customer. Constraints (1n) compute the idle time before the ith customer as the non-negative difference between the actual start time of the ith customer and the completion time of the $(i-1)$th customer plus the travel time from the $i-1$th customer to the ith customer. Constraints (1o) compute the overtime of each provider (if any).

Note that formulation (1) does not require hired providers to serve a particular number of customers; thus, some providers' capacity may not be fully utilized (i.e., partially used providers).

However, as mentioned earlier, some companies may require each provider to serve a particular number I_k of customers. Formulation (1) for the fully used provider case reduces to:

$$\min_{x,z,a,s,g,o} \sum_{n\in[N]} \frac{1}{N} \Bigg\{ \lambda^t \Bigg[\sum_{k\in K} \Bigg(\sum_{i\in I_k} \sum_{\substack{(p,p')\in P\times P\\ p\neq p'}} t^n_{p,p'} z_{i,p,p',k} + \sum_{p\in P} t^n_{0,p} x_{1,p,k} + \sum_{k\in K} \sum_{p\in P} t^n_{p,0} x_{|I_k|,p,k} \Bigg) \Bigg]$$

$$+ \sum_{k\in K} \sum_{i\in I_k} \Big[\lambda^w (s^n_{i,k} - a_{i,k}) + \lambda^g g^n_{i,k} \Big] + \sum_{k\in K} \lambda^o o^n_k \Bigg\} \tag{2a}$$

$$\text{s.t. } (1b), (1f) - (1h), \tag{2b}$$

$$\sum_{p\in P} x_{i,p,k} = 1, \quad \forall i \in I_k, k \in K, \tag{2c}$$

$$a_{i,k} \leq L, \quad \forall i \in I_k, k \in K, \tag{2d}$$

$$(1j), (1k), (1m), (1n), (1p), \tag{2e}$$

$$s^n_{i,k} \geq s^n_{i-1,k} + \sum_{p\in P} d^n_p x_{i-1,p,k} + \sum_{\substack{(p,p')\in P\\ p\neq p'}} t^n_{p,p'} z_{i,p,p',k}, \quad \forall i \in [2, |I_k|]_{\mathbb{Z}}, k \in K, n \in [N], \tag{2f}$$

$$o^n_k \geq \left(s^n_{|I_k|,k} + \sum_{p\in P} (d^n_p + t^n_{p,0}) x_{|I_k|,p,k} \right) - L, \qquad \forall k \in K, n \in [N]. \tag{2g}$$

In Proposition 1, we derive a tight lower bound estimation of the big-M coefficients involved in constraints (1l) of the partially used provider model (S) in (1); see Appendix C for a proof.

Proposition 1. *Let* $\bar{d} = \max_{n\in[N], p\in P} \{d^n_p\}$*,* $\bar{t} = \max_{n\in[N], p\in P, p'\in P} \{t_{p,p'}\}$ *and* $t^{max}_1 = \max_{n\in[N], p\in P} \{t^n_{0,p}\}$*. Suppose* $(\lambda^f, \lambda^t, \lambda^w, \lambda^g, \lambda^o) > 0$*. Then,* $M_i \geq L + t^{max}_1 + (i-1)(\bar{d} + \bar{t})$*, for* $i \in [2, |I_k|]$ *are valid lower bound values for the* M_i *constants in* (1l)*.*

4.2. Model (Z)

Let us now introduce our extension of Zhan and Wan (2018)'s formulation for the HFASP, denoted as model (Z). First, we note that Zhan and Wan (2018) treat customers as nodes with customer 0 representing the depot (i.e., the set of customers is $P \cup \{0\}$) and employs routing variables and constraints to find providers' routes. Thus, as in Zhan and Wan (2018), we define binary decision variables $z_{p,q,k}$ that equal one if provider k travels from customer p to q, and zero otherwise, for all $p \in P \cup \{0\}$, $q \in P \cup \{0\}$, and $k \in K$. For all $p \in P$, we define non-negative continuous decision variables A_p to represent the scheduled appointment time of customer p. For each $p \in P \cup \{0\}$ and $n \in [N]$, we define non-negative continuous variables S^n_p and W^n_p to respectively represent the actual start time and waiting time of customer p under scenario n. Finally, we define non-negative continuous decision variables O^n_k to represent provider k's overtime, and non-negative continuous decision variables G^n_p to represent the provider's idle time after serving customer p under scenario n. Using this notation, model (Z) can be stated as follows:

$$\min_{z,A,W,O,G} \sum_{k\in K} \sum_{p\in P} \lambda^f z_{0,p,k} + \sum_{n\in[N]} \frac{1}{N} \Bigg[\lambda^t \sum_{k\in K} \sum_{p\in P\cup\{0\}} \sum_{q\in P\cup\{0\}} t^n_{p,q} z_{p,q,k}$$

$$+ \lambda^w \sum_{p\in P} W_p^n + \lambda^o \sum_{k\in K} O_k^n + \lambda^g \sum_{p\in P} G_p^n \Bigg], \tag{3a}$$

$$\text{s.t.} \sum_{k\in K} \sum_{q\in P\cup\{0\}} z_{p,q,k} = 1, \quad \forall p \in P, \tag{3b}$$

$$\sum_{p\in P\cup\{0\}} z_{p,q,k} - \sum_{p\in P\cup\{0\}} z_{q,p,k} = 0, \quad \forall q \in P, k \in K, \tag{3c}$$

$$\sum_{p\in P\cup\{0\}} z_{p,0,k} = 1, \quad \forall k \in K, \tag{3d}$$

$$\sum_{p\in P\cup\{0\}} z_{0,p,k} = 1, \quad \forall k \in K, \tag{3e}$$

$$\sum_{p\in P\cup\{0\}} \sum_{q\in P\cup\{0\}} z_{p,q,k} \le |I| + 1 \quad \forall k \in K, \tag{3f}$$

$$A_p \le L \quad \forall p \in P, \tag{3g}$$

$$\sum_{p\in P} \sum_{\substack{q\in P\\ q\neq p}} z_{p,q,k} \le |P'| - 1, \quad \forall P' \subset P, k \in K, \tag{3h}$$

$$S_0^n = 0, \quad \forall n \in [N], \tag{3i}$$

$$S_p^n \ge A_p, \quad \forall p \in P, n \in [N], \tag{3j}$$

$$S_q^n \ge S_p^n + d_p^n + t_{p,q}^n - M\Bigg(1 - \sum_{k\in K} z_{p,q,k}\Bigg), \quad p \in P \cup \{0\}, q \in P, n \in [N], \tag{3k}$$

$$W_p^n = S_p^n - A_p, \quad \forall p \in P, n \in [N], \tag{3l}$$

$$O_k^n \ge S_p^n + d_p^n + t_{p,0}^n - L - M(1 - z_{p,0,k}), \quad \forall p \in P, k \in K, n \in [N], \tag{3m}$$

$$G_q^n \ge S_q^n - S_p^n - d_p^n - t_{p,q}^n - M\Bigg(1 - \sum_{k\in K} z_{p,q,k}\Bigg), \quad \forall p \in P \cup \{0\}, q \in P, n \in [N], \tag{3n}$$

$$(A, S, W, O, G) \ge 0, \quad z \in \{0,1\}^{(|P|+1)\times(|P|+1)\times|K|}. \tag{3o}$$

Formulation (3) finds optimal fleet sizing, routing, and scheduling decisions that minimize the fixed cost and the sample average of the random operational cost. Constraints (3b) ensure that every customer must be visited exactly once. Constraints (3c) ensure the conservation of flow for each provider k at each customer and the depot (i.e., node 0). Constraints (3d)–(3e) ensure that every provider k needs to start from and end at the depot (i.e., node 0). Constraints (3f) ensure that the number of customers assigned to each provider is at most $|I|$. Constraints (3g) ensure that all appointments are scheduled within service hours. Constraints (3h) are subtour elimination constraints. Note that although these constraints are not necessary for finding the optimal tour, as reported by Zhan and Wan (2018), these constraints could improve the model's computational performance. Constraints (3i) set the actual service start time of the service team's office (or the service start time of the workday) to zero.

For each scenario n, constraints (3j)–(3k) require that the actual start time, S_q^n, of each customer

q to be no smaller than the scheduled start time A_q, and the completion time of the preceding customer p plus the travel time from customer p to q. Note that, for a sufficiently large M constant, constraints (3k) are relaxed if customer p is not followed by customer q in provider k's schedule. Constraints (3l) compute the waiting time of each customer in each scenario as the difference between the actual start and scheduled times. Constraints (3m) and (3n) compute the overtime and idle time of each provider in each scenario. Formulation (3) extends that of Zhan and Wan (2018) in the following aspects. First, it incorporates uncertainty of travel time, which is ignored in Zhan and Wan (2018). Second, we modify Zhan and Wan (2018)'s first stage by requiring all appointments to be scheduled within the service hours. Third, we generalize Zhan and Wan (2018)'s second stage by (a) including variables and constraints to compute the random idle time; (b) including random idle time in the objective; and (c) including the random total travel time in the objective of the second stage (recall that Zhan and Wan (2018) assume that the travel time is deterministic).

Note that formulation (3) is for partially used providers (i.e., it does not require hired providers to serve a particular number of customers). However, recall that for the case of fully used providers, we require each provider to serve $|I|$ customers, and accordingly, we know that we need $|P|/|I|$ providers to serve all customers. Therefore, to model the fully used provider case, we remove the fixed cost (first term) from model (3)'s objective and replace constraints (3f) with

$$\sum_{p \in P \cup \{0\}} \sum_{q \in P \cup \{0\}} z_{p,q,k} = |I| + 1, \quad \forall k \in K. \tag{4a}$$

In Proposition 2, we derive a tight lower bound estimation of the big-M coefficients involved in constraints (3k), (3m)–(3n) of model (Z) in (3); see Appendix D for a proof.

Proposition 2. *Let* $\bar{d} = \max_{n \in [N], p \in P} \{d_p^n\}$, $\bar{t} = \max_{n \in [N], p \in P, p' \in P} \{t_{p,p'}\}$, $t_1^{max} = \max_{n \in [N], p \in P} \{t_{0,p}^n\}$, *and* $t_2^{max} = \max_{n \in [N], p \in P} \{t_{p,0}^n\}$. *Suppose* $(\lambda^f, \lambda^t, \lambda^w, \lambda^g, \lambda^o) > 0$. *Then,* $M \geq L + |I|\bar{d} + (|I| - 1)\bar{t} + t_1^{max} + t_2^{max}$ *is a valid lower bound value for the big-M constant in constraints* (3k) *and* (3m)–(3n).

5. Theoretical Analysis of Model (S) and Model (Z)

For a fixed sample of service duration and travel time scenarios, model (S) and model (Z) reduce to large-scale mixed-integer linear programs (MILP). It is well-known that the computational performance of an MILP is mainly influenced by its size (number of decision variables and constraints) and the tightness of its linear programming relaxation (LPR). Therefore, in this section, we analyze the sizes and LPRs of the proposed models. First, in Table 2, we compare the size of the proposed formulations in terms of the number of decision variables and constraints. Note that the number of variables and constraints used in model (Z) under the fully and partially used cases are equal. Hence, we only present the size for the fully used case. In contrast, model (S) for the fully used provider case has fewer binary variables and second-stage constraints than model (S) for the

Table 2: Size of formulations of the HFASP with $|I|$ positions, $|P|$ customers, $|K|$ providers, and N scenarios

	Model (Z)	**Model (S)**	
		fully used	partially used
Binary variables	$\lvert K\rvert(\lvert P\rvert+1)^2$	$\lvert I\rvert\lvert P\rvert\lvert K\rvert$	$(\lvert I\rvert+1)\lvert P\rvert\lvert K\rvert$
Continuous variables	$(3\lvert P\rvert+\lvert K\rvert+1)N+\lvert P\rvert$	$(\lvert I\rvert-1)\lvert P\rvert^2\lvert K\rvert+\lvert I\rvert\lvert K\rvert+(2\lvert I\rvert\lvert K\rvert+\lvert K\rvert)N$	$(\lvert I\rvert-1)\lvert P\rvert^2\lvert K\rvert+\lvert I\rvert\lvert K\rvert+(2\lvert I\rvert\lvert K\rvert+\lvert K\rvert)N$
First-stage constraints	$2\lvert P\rvert+(2^{\lvert P\rvert}+\lvert P\rvert+1)\lvert K\rvert$	$3\lvert I\rvert\lvert K\rvert+\lvert P\rvert+3(\lvert I\rvert-1)\lvert K\rvert\lvert P\rvert^2-\lvert K\rvert$	$3\lvert I\rvert\lvert K\rvert+\lvert I\rvert\lvert P\rvert\lvert K\rvert+\lvert P\rvert+3(\lvert I\rvert-1)\lvert K\rvert\lvert P\rvert^2-\lvert K\rvert$
Second-stage constraints	$(2\lvert P\rvert^2+4\lvert P\rvert+\lvert P\rvert\lvert K\rvert+1)N$	$3\lvert I\rvert\lvert K\rvert N+\lvert K\rvert N$	$4\lvert I\rvert\lvert K\rvert N$

partially used provider case. In particular, for the partially used provider case, we need $|P||K|$ additional variables to identify the last customer on each provider's sequence (i.e., $x_{0,p,k}$, for all $p \in P$ and $k \in K$) and $|I||P||K|$ additional constraints (1d) on $x_{0,p,k}$ in the first stage. In addition, we need more constraints to compute partially used providers' overtime. It is clear from Table 2 that model (Z) has more binary variables and second-stage constraints than model (S). The number of continuous scenario-based variables is also smaller in model (S) under the assumption that $|P| < |I||K|$, which always holds because all customers must be served in HFASP formulations.

Next, in Theorem 1, we show that models (S) and (Z) for fully used providers are equivalent (see Appendix E for a proof). Similar analysis techniques can be used to show the equivalence of partially used models.

Theorem 1. *Suppose* $(\lambda^f, \lambda^t, \lambda^w, \lambda^g, \lambda^o) > 0$. *Model (S) and model (Z) for fully used providers are equivalent. In particular, given an optimal solution to the model (S), we can construct a feasible solution to the model (Z) with the same objective function value and vice versa.*

Finally, in Theorem 2, we show that the LPR of model (S) provides a tighter linear relaxation than the LPR of model (Z); see Appendix F for a proof. The theoretical analyses in this section suggest that the smaller and tighter model (S) has a better computational performance than model (Z). Indeed, our computational results in Section 7.2 support this conclusion.

Theorem 2. *Suppose* $(\lambda^f, \lambda^t, \lambda^w, \lambda^g, \lambda^o) > 0$. *The optimal objective value of the LPR of model (S) is greater than or equal to the optimal objective value of the LPR of model (Z).*

6. Heuristics for the HFASP

In Section 7, we show that our proposed model (S) can efficiently solve realistic (and previously unsolved) HFASP instances to optimality. However, solution times increase as the instance size increases. Therefore, in this section, we propose two heuristics (denoted as `FAS-RS` and `FAS-R-S`) that allow for obtaining near-optimal solutions of larger instances that may be observed in applications other than the HFASP within an acceptable time. These heuristics decompose the decision process into two parts. The first part involves deciding the number of providers to hire (fleet sizing) and customer assignments to hired providers. The second part involves constructing a routing plan and a schedule for each provider. Both heuristics implement an integer program denoted as `FAS` to determine the number of providers to hire and customer-to-provider assignments. Then, the FAS-RS heuristic employs a single-provider variant of model (S) to obtain an optimal routing plan and a schedule for each provider. In contrast, the FAS-R-S heuristic employs a modified insertion heuristic to determine a routing plan for each provider and then an LP to determine the appointment time for each customer. We discuss the details of these heuristics in the next subsections.

6.1. Two-phase heuristic: FAS-RS

Algorithm 1 summarizes the steps of our two-phase FAS-RS heuristic. In phase 1, we solve a fleet sizing and assignment (FAS) problem that determines the number of providers to hire and customer-to-provider assignments. In the second phase, we implement a stochastic single-provider routing and scheduling (RS) model to determine an optimal routing plan and a schedule for each provider. Next, we discuss the details of the FAS model employed in phase 1. We define a binary decision variable u_k, which equals one if provider $k \in K$ is hired, and a binary decision variable $y_{p,k}$, which equals one if customer $p \in P$ is assigned to provider k. For each $p \in P$, $k \in K$ and scenario $n \in [N]$, we define a parameter $\lambda^n_{p,k} = t^n_{0,p} + t^n_{p^*_k,p} - t^n_{0,p^*_k}$, where $t^n_{i,j}$ is the travel time between customers $(i,j) \in (P \cup \{0\}) \times (P \cup \{0\})$ under scenario $n \in [N]$, and p^*_k is the customer with the kth smallest expected travel time to the depot. For example, p^*_1 is the customer with the shortest expected travel time to the depot. Intuitively, $\lambda^n_{p,k}$ evaluates the additional travel time if provider k visits customer p under scenario n. Using this notation, we formulate the FAS problem as follows

$$\begin{aligned}
\underset{\boldsymbol{u},\boldsymbol{y}}{\text{minimize}} \quad & \sum_{k \in K} \lambda^f u_k + \sum_{p \in P} \sum_{k \in K} \sum_{n \in [N]} \frac{1}{N} \lambda^n_{p,k} y_{p,k} && \text{(5a)}\\
\text{subject to:} \quad & \sum_{p \in P} y_{p,k} \leq |I_k| u_k, \quad \forall k \in K, && \text{(5b)}\\
& \sum_{k \in K} y_{p,k} = 1, \quad \forall p \in P, && \text{(5c)}\\
& u_k, y_{p,k} \in \{0,1\} \quad \forall p \in P, k \in K. && \text{(5d)}
\end{aligned}$$

Formulation (5) determines the optimal number of providers and customer assignments that minimize the total hiring cost and the additional travel time of adding customers into a provider's route. Constraints (5b) ensure that each provider serves at most $|I_k|$ customers. Constraints (5c) ensure that each customer is assigned to exactly one provider. For the fully used provider case, we replace constraints (5b) with $\sum_{p \in P} y_{p,k} = |I_k| u_k$ to ensure that each hired provider serves exactly $|I_k|$ customers. Let $P_k := \{p \in P : y^*_{p,k} = 1\}$ represent the set of customers assigned to provider k, where $\boldsymbol{y}^*$ is an optimal solution to (5). In phase 2, we solve a single-provider variant of model (S) in (2) for each k with P fixed to P_k and $|K| = 1$ to obtain an optimal schedule and a routing plan. Our results in Section 7.3 show that our FAS-RS heuristic could quickly obtain near-optimal solutions for large HFASP instances.

```
Algorithm 1: The FAS-RS Heuristic
  Input   : Sets of customers $P$, providers $K$, and scenarios $[N]$
  Output: Subset $\bar{K} \subseteq K$ of hired providers, a route ($\boldsymbol{x}$) and a schedule ($\boldsymbol{a}$) for each $k \in \bar{K}$
  Phase 1. Solve the FAS problem in (5), record the optimal solution $(\boldsymbol{u^*}, \boldsymbol{y^*})$, and set $\bar{K} \leftarrow \{k \in K : u^*_k = 1\}$
           and $P_k \leftarrow \{p \in P : y^*_{p,k} = 1\}$, for all $k \in \bar{K}$;
  Phase 2. For each $k \in \bar{K}$, solve formulation (2) and return optimal $(\boldsymbol{x^*}, \boldsymbol{a^*})$
```

```
Algorithm 2: The FAS-R-S Heuristic
  Input   : Sets of customers $P$, providers $K$, and scenarios $[N]$
  Output: Subset $\bar{K} \subseteq K$ of hired providers, a route ($R_k$) and a schedule ($\boldsymbol{a}$) for each $k \in \bar{K}$
  Phase 1. Solve the FAS problem in (5), record the optimal solution $(\boldsymbol{u^*}, \boldsymbol{y^*})$, and set $\bar{K} \leftarrow \{k \in K : u^*_k = 1\}$;
           and $P_k \leftarrow \{p \in P : y^*_{p,k} = 1\}$, for all $k \in \bar{K}$;
  for $k \in \bar{K}$ do
      Phase 2. Implement Algorithm 3 to obtain a route $R_k$;
      Phase 3. Solve model (2) with ($\boldsymbol{x}$, $\boldsymbol{z}$) fixed according to $R_k$ and return $\boldsymbol{a^*}$;
  end
```

6.2. Three-phase heuristic: `FAS-R-S`

In this section, we present our three-phase heuristic, donated as `FAS-R-S`. Algorithm 2 summarizes the steps of this heuristic. Phase 1 is the same as that of `FAS-RS` and involves solving the `FAS` model in (5) to obtain a set of hired providers and customer-to-provider assignments. Then, in phase 2, we employ a modified insertion heuristic (MIH) to determine a routing plan for each provider. Finally, given the routing plan from phase 2, we solve an LP model to determine the appointment time for each customer (phase 3).

Next, we discuss the details of the MIH employed in phase 2. Let P_k represent the set of customers assigned to provider k obtained from phase 1. We define $R_k := \{(1, q), (2, –), \dots, (|I|_k, –)\}$ as the partial route of provider k, where (i, q) indicates that customer $q \in P_k$ is in position $i \in I_k$ and $(i, –)$ indicates that position i is empty. Finally, we let $R_k(i, p)$ represent the new route resulting from inserting customer $p \in P_k$ into position i. For example, suppose that the current route is $R_k := \{(1, q), (2, q^{''}), (3, q^{'''}), \dots\}$ and we want to insert customer p between the second and third customer in R_k, i.e., assign p to position $i = 3$. Then, the new route will be $R_k(3, p) := \{(1, q), (2, q^{''}), (3, p), (4, q^{'''}), \dots\}$. Note that inserting a customer in the route may increase the provider's travel time, change customers' start times, and potentially increase waiting time and overtime. Thus, the MIH finds a position for each customer with the lowest insertion cost; see Figure 1. Algorithm 3 summarizes the steps of our MIH heuristic, which extends the insertion heuristic for VRP (see, e.g., Campbell and Savelsbergh, 2004) to fit the HFASP. Starting with an empty route, the MIH heuristic iteratively inserts each as-of-yet unassigned customer in P_k into a position in the partially constructed route R_k that leads to the lowest insertion cost (described next). Since the number of customers and positions are finite, the algorithm terminates in a finite

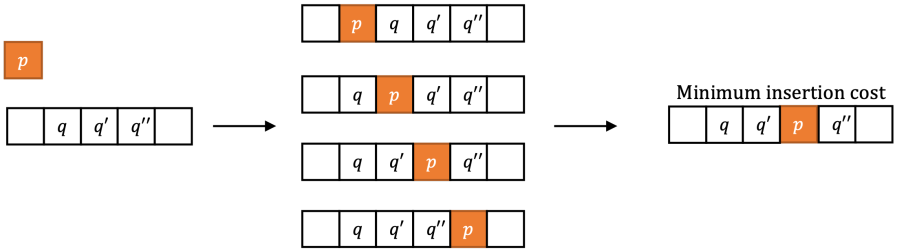

Figure 1: An illustration of inserting a customer into a partially constructed route.

number of iterations. To compute the cost on route R_k, we first compute the actual start time for each customer in route R_k as follows.

$$\texttt{start(1,k,n)} = t^n_{0,r_1}, \tag{6a}$$

$$\texttt{start(i,k,n)} = \texttt{start(i-1,k,n)} + d^n_{r_{i-1}} + t^n_{r_{i-1},r_i}, \quad \forall i \in [2, |R_k|]. \tag{6b}$$

Recall that the start time should be greater than or equal to the scheduled appointment time, and the latter should not exceed L. Accordingly, we compute the appointment time appoint(i,k) as follows (we later refine the appointment time in phase 3).

$$\texttt{appoint(i,k)} = \min\left\{\min_{n\in[N]}\{\texttt{start(i,k,n)}\}, L\right\}. \tag{7}$$

Given customers' actual service start times and appointment times, we can compute the total waiting time $\texttt{EW}(R_k)$ and overtime $\texttt{EO}(R_k)$ on route R_k as follows.

$$\texttt{EW}(R_k) = \sum_{i=1}^{|R_k|}\Big[\max\{\texttt{start(i,k,n)} - \texttt{appoint(i,k)}, 0\}\Big],$$

$$\texttt{EO}(R_k) = \max\left\{\texttt{start(}|R_k|\texttt{,k,n)} + d^n_{|R_k|} + t^n_{|R_k|,0} - L, 0\right\}.$$

Finally, we define the cost of inserting customer $p \in P_k$ into the ith position of provider k's route Insert(i,p,k) as the weighted sum of extra travel cost ($\lambda^t \cdot \texttt{ET(i,p,k)}$), extra waiting cost ($\lambda^w \cdot \texttt{EW(i,p,k)}$) and extra overtime cost ($\lambda^o \cdot \texttt{EO(i,p,k)}$):

$$\texttt{Insert(i,p,k)} = \lambda^t \cdot \texttt{ET(i,p,k)} + \lambda^w \cdot \texttt{EW(i,p,k)} + \lambda^o \cdot \texttt{EO(i,p,k)}, \tag{8}$$

where

$$\texttt{ET(i,p,k)} = \sum_{n\in[N]} \frac{1}{N}\left[t^n_{r^k_{i-1},p} + t^n_{p,r^k_i} - t^n_{r^k_{i-1},r^k_i}\right], \tag{9a}$$

$$\texttt{EW(i,p,k)} = \texttt{EW}(R_k(i,p)) - \texttt{EW}(R_k), \quad \texttt{EO(i,p,k)} = \texttt{EO}(R_k(i,p)) - \texttt{EO}(R_k). \tag{9b}$$

Once we obtain a routing plan R_k from phase 2, in phase 3, we determine the optimal appointment time for each customer on the route by solving a single-provider LP variant of model (S) in (2) with $|K| = 1$ and sequencing variables ($\boldsymbol{x}$, $\boldsymbol{z}$) fixed according to R_k.

7. Computational Experiments

In this section, we present computational experiments that explore the size and characteristics of the HFASP instances that can be solved using the proposed models for the HFASP. In addition,

Algorithm 3: The Modified Insertion Heuristic (MIH)

```
Input  : A set of hired providers $K$ and the set of assigned customers $P_k$ to each
Output: A routing plan $R_k$ for each provider $k \in K$

for $k \in K$ do
    Initialize an empty route $R_k = \varnothing$ and the set of unassigned customers $\bar{P}_k \leftarrow P_k$;
    while $\bar{P}_k \neq \varnothing$ do
        for $p \in \bar{P}_k$ do
            for $i \in [1, |R_k| + 1]$ do
                // Insertion step
                Construct a new route $R_k(i, p)$ by inserting $p$ into the $i$th position in $R_k$;
                for $j \in [i, |R_k(i, p)|]$ do
                    Compute start(j,k,n) and appoint(j,k) using (6a)–(6b) and (7)
                end
                Compute the insertion cost Insert(i,p,k) using equation (8)
            end
            Assign $p$ to position $i^* \in \arg\min_{i \in [1,|R_k|+1]}$ Insert(i,p,k) and Update $R_k \leftarrow R_k(i^*, p)$;
            Update start(i,k,n) and appoint(i,k) based on $R_k$ using (6a)–(6b) and (7);
        end
        $\bar{P}_k \leftarrow \bar{P}_k \setminus \{p\}$;
    end
end
```

we derive insights into the HFASP. In Section 7.1, we describe the set of HFASP instances that we constructed and discuss other experimental setups. In Sections 7.2.1 and 7.2.2, we analyze solution times of the proposed models for fully used and partially used providers, respectively. In Section 7.3, we analyze solution times and solution quality of the proposed heuristics. Finally, in Section 7.4, we construct HFASP instances based on Lehigh County of PA to drive managerial insights and provide sensitivity analyses for input parameter values.

7.1. Description of experiments

We construct HFASP instances in part based on benchmarks and parameters settings made in recent related literature (e.g., Zhan and Wan, 2018; Hashemi Doulabi et al., 2020; Yu et al., 2021). Each instance is characterized by the number of customers $|P|$ and providers $|K|$. We consider instances with $|P| \in \{6, \ldots, 72\}$ customers and $\lceil |P|/6 \rceil + 1$ service providers (Zhan and Wan, 2018). We generate most instances using the approach proposed in Zhan and Wan (2018). Specifically, we assume that customers are located uniformly and randomly on a 50×50 square, and the service provider's office is at the point $[0, 0]$. We set the daily working hour of each provider L to 8 hours.

We set the cost parameters in the objective function in part based on Zhan and Wan (2018). Specifically, we set the unit overtime cost λ^o to 1, unit idle cost λ^g to 0. We generate the unit waiting cost λ^w from $U[0, 1.5]$ ($U[a, b]$ is a uniform distribution over the interval $[a, b]$), and the unit travel cost λ^t from $U[0.1, 0.5]$. We set the unit fixed cost λ^f to 1000. As in prior literature, we use the lognormal distribution for the service time d_i^n truncated on the interval $[10, 50]$ with mean μ and $\sigma = 0.5\mu$, where μ is generated from $U[25, 35]$. We assume that customers in the same neighborhood spend an average of T minutes (e.g., $T = 20$ minutes) traveling from one place to another. The assumption is realistic and widely adopted in recent literature (see, e.g., Nikzad et al., 2021; Tsang and Shehadeh, 2023). Accordingly, we consider two types of travel times: travel time from the provider's office to the customer and travel time between customers. The travel times between the office and customer p, $t_{0,p}^n$ and $t_{p,0}^n$, follow a normal distribution $N(t_p, \frac{t_p}{6})$, where t_p is the Euclidean distance between the customer p and the office (Hashemi Doulabi et al., 2020). Consistent with the literature, we generate the travel time between customers, $t_{p,p'}^n$, from $U[15, 25]$, for all $(p, p') \in P \times P, n \in [N]$. We set $d_0^n = 0$ and $t_{0,0}^n = 0$ in model (Z) for all scenarios $n \in [N]$ ($p = 0$ is the provider's office (or depot)). Note that the average number of customers that an provider may visit per day is often less than 6 in home health care and banking, and less than 10 in repair services (NAHC, 2010; Yuan et al., 2015; Zhan et al., 2021). Accordingly, we consider cases where a provider can visit 6 or 8 customers.

To decide an appropriate sample size for the proposed SAA models, we employed the Monte Carlo optimization (MCO) procedure, which provides statistical lower and upper bounds on the optimal value of the HFASP based on the optimal solution to its SAA with N scenarios. This, in turn, provides a statistical estimate of the approximated relative gap between the lower and upper bounds. Applying the MCO procedure with $N = 50$, the relative approximation gaps for the HFASP instances were around 1%, whereas larger N resulted in longer solution times without consistent and significant improvements in the gap. Therefore, we selected $N = 50$ for our computational experiments. In Appendix G, we provide a description of the MCO procedure and a numerical example. We refer readers to Kenyon and Morton (2003), Kleywegt et al. (2002), Shapiro et al. (2021) and references therein for further details on MCO and related technical results.

In our implementations, we add symmetry-breaking constraints (H.1) to model (S) and constraints (H.2) to model (Z) to enforce that provider k is hired before provider $k + 1$. We set the values of the big-M coefficients in constraints (1l) according to Proposition 1 and those in (3k),(3m) and (3n) according to Proposition 2. We implemented our proposed models in AMPL modeling language and used CPLEX (version 20.1.0.0) as the solver with default settings. We imposed a solver time limit of 3600 seconds (1 hour) for each SAA instance. We conducted all the experiments on a computer with an Intel Xeon Silver processor with 2.10 GHz CPU and 128 Gb memory.

7.2. CPU time

In this section, we compare solution times of the proposed models for the fully used (Section 7.2.1) and partially used (Section 7.2.2) provider cases.

Table 3: Solution time (in seconds) of model (S) for fully used providers

model (S)	$\lvert I\rvert = 6$			$\lvert I\rvert = 8$			
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max
24	3.2	3.3	3.5	24	3.9	9.1	19.4
30	7.6	9.1	13.1	32	12.3	13.3	15.0
36	18.6	20.6	22.3	40	31.6	46.4	60.3
42	99.5	140.7	172.5	48	462.2	751.0	1288.8
48	169.5	397.1	854.5	56	1484.9	1848.6	3165.7
54	68.1	850.9	1821.7	64	-	-	-
60	532.3	1458.2	2435.1	72	-	-	-

When $|I| = 8, |P| \geq 32$, we set the relative MIP gap to 0.04

7.2.1. Fully used provider case

Recall that when the provider must serve $|I|$ customers, we know the number of providers needed to serve all customers (i.e., $|K| = |P|/|I|$). Hence, the problem reduces to an assignment, sequencing, and scheduling problem. Therefore, we solve problem instances with $|I| = 6$ ($|I| = 8$), $|P| \in \{6, 12, \ldots, 60\}$ ($|P| \in \{8, 16, \ldots, 56\}$), and $|K| \in \{1, 2, \cdots, 10\}$. For each combination of $|I|$, $|P|$, and $|K|$, we generated and solved five random SAAs as described in Section 7.1.

Table 3 presents the minimum (Min), average (Avg), and maximum (Max) solution time (in seconds) solved instances using model (S) within the imposed one-hour time limit. We observe the following from this table. First, using model (S), we were able to solve all the SAAs that correspond to problem instances with ($|I| = 6,\ |P| \leq 36$) and ($|I| = 8,\ |P| \leq 32$) within 30 seconds. Second, the average solution time of the larger instances with $|I| = 6$ ranges from 2.3 minutes ($|P| = 42$) to 24 minutes ($|P| = 60$), and with $|I| = 8$ ranges from 46 seconds ($|P| = 40$) to 30 minutes ($|P| = 56$). Finally, we were not able to solve instances with $|I| = 8$ and $|P| = 64$ and 72 customers within the imposed time limit of one hour. However, the model terminated with a relative MIP (relMIP) gap around 0.1 (relMIP=$\frac{UB-LB}{UB}$, where UB is the best upper bound and LB is the LP relaxation-based lower bound obtained at termination).

In contrast, using model (Z), we were only able to solve small instances, specifically, all the SAAs that correspond to instances with $|I| = 6, |P| \leq 18$ and $|I| = 8, |P| \leq 16$. We present a comparison of solution times of these instances by model (S) and (Z) in Table 4. Clearly, model (Z) takes from 0.3 to 528 times longer than model (S) to solve these instances. Moreover, for those instances that model (Z) failed to solve within the time limit, it terminated with either a relMIP gap around 50% (when $|I| = 6$) or without any feasible MIP solutions (and thus no upper bound).

We attribute the difference in solution times to the following. First, as discussed in Section 5, model (Z) has significantly more binary first-stage variables than model (S). Moreover, model (Z) has a significantly larger number of first-stage constraints and a larger number of scenario-based constraints and variables. As argued in Klotz and Newman (2013), this increase in model size

Table 4: Ratios of solution times of models (Z) and (S) on the SAAs solved by both (fully used models).

$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$		$\lvert I\rvert = 6$				$\lvert I\rvert = 8$	
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max
6	1.2	1.7	3.2	8	0.3	0.6	1.5
12	2.3	3.9	5.6	16	7.8	39.1	80.6
18	235.8	299.6	528.6	24	-	-	-

Table 5: Ratios of optimal objective values of LP relaxations of models (S) and (Z) (fully used models).

$\frac{\text{(S) relaxation.obj}}{\text{(Z) relaxation.obj}}$		$\lvert I\rvert = 6$				$\lvert I\rvert = 8$	
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max
6	1744.5	1750.8	1758.4	8	1503.8	1520.9	1542.1
12	1761.3	1764.4	1769.3	16	1514.1	1521.3	1534.5
18	1761.4	1774.9	1789.6	24	1523.2	1531.0	1545.1

often suggests an increase in solution time for the LP relaxations. Second, as shown in Table 5, the LP relaxations obtained using model (S) are strictly tighter than using model (Z) by a factor of 1503 to 1789, which is consistent with the theoretical results in Theorem 2. Finally, model (S) for the fully used provider does not have constraints involving big-M coefficients, but model (Z) has such constraints. It is well-known that the big-M coefficients and constraints could undermine computational efficiency, enlarge the feasible region of the LP relaxation of the model, and cause numerical errors (Klotz and Newman, 2013; Camm et al., 1990). In Appendix I, we provide additional solution time results under a larger sample size and a different cost structure in the objective function. These results emphasize that model (S) is more computationally efficient than model (Z) under these settings.

It is worth noting that using off-the-shelf optimization software such as CPLEX to solve model (S) directly is more computationally efficient than using Benders decomposition (BD). Specifically, using BD, we were only able to solve small instances of HFASP with a maximum size of ($|I|$, $|P|$)=(6, <24) within the given time limit. In contrast, model (S) can solve larger instances much faster, as shown in Table 3. Furthermore, when BD fails to solve instances within the given time limit, it terminates either with a large relMIP or without feasible MIP solutions. In such cases, the solution obtained by BD at the termination has poor quality. For instance, consider the instance with $(|I|, |P|) = (6, 30)$. As shown in Table 6, BD fails to solve this instance within the given time limit and terminates with a 7% gap. Moreover, the solution obtained from BD at termination has longer travel time, waiting time, and overtime.

7.2.2. Partially used provider case

Let us now analyze solution times of the proposed models for the partially used provider case. We present results for problem instances with $|I| = 6$ ($|I| = 8$) and $|P| = \{6, 8, \ldots, 62\}$ ($|P| =$

Table 6: Comparison of solutions using Benders decomposition (BD) and CPLEX (fully used models).

$(\lvert I\rvert, \lvert P\rvert)$	Method	Sol.Time/Gap	Obj	Travel Time	Waiting Time	Overtime
(6,30)	BD	7.4%	5454	821	2171	60
	Model (S)	10.3	5083	819	4.6	0.1

$\{6, 8, \dots, 42\}$). We set the number of service providers $|K|$ to $\lceil |P|/6 \rceil + 1$. We generated and solved five instances for each combination of $|P|$ and $|K|$. Table 7 presents the Min, Avg, and Max solution time of the instances solved by model (S). we observe that using model (S), we were able to solve all the instances with $|I| = 6, |P| \leq 42$ and $|I| = 8, |P| \leq 30$ within one minute. The average solution times using model (S) with $|I| = 6$ ranges from 6.8 seconds ($|P| = 24$) to 6 minutes ($|P| = 62$), and with $|I| = 8$ ranges from 11.8 seconds ($|P| = 24$) to 18 minutes ($|P| = 40$). Additionally, we could not solve problem instances with $|I| = 8, |P| = 48$ within one hour. In contrast, using model (Z), we were only able to solve small instances, namely all the SAAs that correspond to instances with $|I| = 6$ and $|P| \leq 10$ and with $|I| = 8$ and $|P| \leq 8$. We present a comparison of solution times of these instances by models (S) and (Z) in Table 8. Observe that the solution times of model (Z) is 2.4 to 6235.2 times longer than the solution times of model (S). Moreover, for those instances that were not solved by model (Z) within the imposed time limit, the solver terminated with a relMIP of around 100% ($|I| = 6$) and 70% ($|I| = 8$) or without any feasible MIP solutions.

We attribute the differences in solution time to the following. First, model (Z) has more binary variables and constraints than model (S) (see Section 5), indicating that model (Z) is potentially more challenging to solve than model (S). Second, as shown in Table 9, the LP relaxations of model (S) are strictly tighter than that of model (Z), by a factor of 1500.6 to 1771 (proved theoretically in Theorem 2). We also compare the computational performance of models (S) and (Z) using another cost structures and a larger number of scenarios (see Appendix J for details). We observe that model (S) is always better than model (Z) in the sense that model (S) can solve larger instances and is more computationally efficient.

Finally, it is worth noting that using BD, we cannot solve even small instances with 12 customers and 3 providers within the imposed time limit. For example, after two hours, the average BD gap for instances with $(|I|, |P|, |K|) = (6, 12, 3)$ is approximately 6%. In contrast, model (S) can solve this instance within a few seconds and quickly solve other larger instances (see Table 7). This again emphasizes that solving model (S) directly is more computationally efficient.

7.3. Analysis of the `FAS-RS` and `FAS-R-S` heuristics

In this section, we investigate solutions quality and computational performance of the `FAS-RS` and `FAS-R-S` heuristics. First, in Table 10, we present the relative gap $\frac{\nu - \nu^*}{\nu^*} \times 100\%$ between the optimal objective value ν^* of model (S) and the objective value ν computed using solutions obtained via the proposed heuristics for all instances that model (S) can solve to optimality. The small gap

Table 7: Solution time (in seconds) of model (S) for partially used providers.

model (S)	$\|I\| = 6$			$\|I\| = 8$		
$\|P\|$	Min	Avg	Max	Min	Avg	Max
24	6.1	6.8	7.4	9.2	11.8	15.5
30	15.1	16.1	18.6	43.6	50.5	55.1
36	17.0	18.8	20	168.6	191.5	216.3
40	27.4	29.6	32.2	644.5	1084.4	1819.4
42	28.8	31.6	34.0	484.7	738.6	967.6
48	51.8	61.8	72.6	-	-	-
50	67.5	109.8	148.2	-	-	-
54	93.0	102.0	120.2	-	-	-
58	122.4	199.7	306.7	-	-	-
60	148.8	175.3	253.4	-	-	-
62	154.2	361.5	1067.9	-	-	-

Table 8: Ratios of solution times of models (Z) and (S) on the SAAs solved by both (partially used provider models).

$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$	$\|I\| = 6$			$\|I\| = 8$		
$\|P\|$	Min	Avg	Max	Min	Avg	Max
6	4.2	4.5	4.9	2.4	2.7	3.1
8	81.5	88.2	102.8	6.1	20.6	30.8
10	3405.1	4286.9	6235.2	-	-	-

Table 9: Ratios of optimal objective values of LP relaxations of model (S) and (Z) (partially used provider models).

$\frac{\text{(S) relaxation.obj}}{\text{(Z) relaxation.obj}}$	$\|I\| = 6$			$\|I\| = 8$		
$\|P\|$	Min	Avg	Max	Min	Avg	Max
6	1744.5	1750.8	1758.4	1500.6	1511.0	1520.9
8	1745.0	1755.1	1771.0	1504.4	1516.5	1533.2
10	1760.1	1764.6	1767.1	1509.1	1513.1	1518.1

values in Table 10 indicate that `FAS-RS` and `FAS-R-S` could produce near-optimal solutions to the HFASP with gap values ranging from 0.86% to 5.10%. However, `FAS-RS` solutions generally yield lower gap values than `FAS-R-S`, especially when $|I| = 8$. This makes sense because `FAS-RS` employs model (S) in phase 2 to optimize routing and scheduling decisions for each provider. In contrast, `FAS-R-S` uses the MIH heuristic to obtain routing decisions in phase 2 and then an LP to find the appointment times in phase 3.

Next, we investigate the computational performance of `FAS-RS` and `FAS-R-S` on solving large instances that might be observed in other application domains. Table 11 presents the total time required by each heuristic to solve large instances with customers ranging from 102 to 504 (fully used) and 100 to 500 (partially used). We observe the following from this table. First, both heuristics can efficiently solve large instances of the HFASP within a reasonable time. Specifically,

Table 10: The relative gap between ν^* and ν.

Fully used provider case					
$\lvert I\rvert = 6$			$\lvert I\rvert = 8$		
$\lvert P\rvert$	FAS-RS	FAS-R-S	$\lvert P\rvert$	FAS-RS	FAS-R-S
24	0.89%	1.21%	24	1.62%	3.43%
30	0.96%	1.36%	32	2.03%	3.58%
36	0.99%	1.31%	40	1.74%	5.10%
42	1.01%	1.16%	48	1.90%	4.34%
48	1.01%	1.14%	56	2.11%	3.50%
54	1.01%	1.12%	64	2.00%	2.85%
Partially used provider case					
$\lvert I\rvert = 6$			$\lvert I\rvert = 8$		
$\lvert P\rvert$	FAS-RS	FAS-R-S	$\lvert P\rvert$	FAS-RS	FAS-R-S
24	1.05%	1.16%	24	2.08%	3.52%
36	1.20%	1.29%	36	2.02%	5.07%
42	1.15%	1.07%	42	1.93%	4.39%
48	1.15%	1.21%	48	1.15%	3.92%
54	1.15%	1.12%	54	1.15%	3.09%
60	1.15%	1.07%	60	1.15%	4.23%

solution times range from 2.8 seconds (using FAS-RS for $|P| = 104$ and $|I| = 8$ fully used provider case) to 16 minutes (using FAS-RS for $|P| = 504$ and $|I| = 8$ fully used provider case). Second, the solution times of FAS-R-S are shorter than FAS-RS, and the difference is significant for larger instances with $|I| = 8$. For example, consider the fully used provider case with $|I| = 8$ and $|P| = 504$. FAS-R-S takes about 34 seconds to solve this instance, while FAS-RS takes 16 minutes. Recall that FAS-RS involves solving a SMIP in phase 2 to obtain routing and scheduling decisions. In contrast, FAS-R-S obtains routes using the MIH heuristic and then solves an LP to obtain appointment times. Thus, it makes sense that FAS-RS takes a longer time to solve each instance.

Note that using model (S) in phase 2 of FAS-RS is more computationally efficient than using model (Z). For example, consider the partially used provider case. For example, solution times of instances with $|P| = (100, 200, 300, 400)$ and $|I| = 8$ using FAS-RS with model (Z) implemented in phase 2 are (470, 734, 1480, 1683) seconds and with model (S) implemented in phase 2 are (116, 197, 453, 596). These results demonstrate that model (S) also enables computationally efficient heuristics for the HFASP.

7.4. Case study

In this section, we consider a service region based on twenty-five cities in Lehigh County of Pennsylvania (see Figure 2). Then, we construct two HFASP instances based on this region as follows. First, we used the population estimate for each city based on the most updated information posted in 2013–2017 US Census Bureau to construct two instances denoted as L-50 and L-100, where 50 and 100 are the total number of customers. In L-50 and L-100, we used the population percentage (weight) in each city to calculate the number of customers as population% × 50 and population% × 100, respectively (see Table K1 in Appendix K). To a certain extent, these

Table 11: Solution times (in seconds) using FAS-RS and FAS-R-S .

Fully used provider case					
$\|I\| = 6$			$\|I\| = 8$		
$\|P\|$	FAS-RS	FAS-R-S	$\|P\|$	FAS-RS	FAS-R-S
102	9.2	3.2	104	110	2.8
204	22.8	8.4	200	249	7.4
300	31.8	15.8	304	587	14.6
402	46.4	25.8	400	700	23
504	64.4	38.6	504	960	34.6
Partially used provider case					
$\|I\| = 6$			$\|I\| = 8$		
$\|P\|$	FAS-RS	FAS-R-S	$\|P\|$	FAS-RS	FAS-R-S
100	9.4	2.8	100	116	3.2
200	22	9.4	200	197	7.6
300	34.2	13.4	300	453	17.6
400	72.8	21.4	400	596	23.2
500	79.4	43.2	500	687	44.6

instances reflect what may be observed in real life, i.e., locations with larger populations may potentially create greater demand. We chose Lehigh Valley Hospital Home Care (in Allentown), which primarily serves Lehigh County, as the depot (provider office). For each instance, we first randomly located customers (nodes) in each city at some residential area within the city, such as apartments or gated communities (see Figure K1 in Appendix K). Second, we obtained the travel time $\bar{t}_{p,q}$ between each pair of customers (p, q) and used it as the average travel time. Third, we generated the non-negative travel time between each pair of customers (p, q) from a normal distribution $N(\bar{t}_{p,q}, \frac{\bar{t}_{p,q}}{6})$. Finally, we follow the same procedure described in Section 7.1 to generate the service time and other parameters. For the L-100 instance, we use the K-means clustering method to group customers into three clusters based on the distance between them.

Next, we analyze the impact of key input parameters on the optimal solutions and the associated operational performance. In Section 7.4.1, we investigate the impact of service time variability. In Section 7.4.2, we study the impact of the per-unit waiting and overtime costs. Finally, we analyze the impact of ignoring uncertainty in Section 7.4.3.

7.4.1. Impact of variability in service time

In this section, we analyze the impact of service time variability on the number of hired providers and the associated operational (second-stage) cost. In addition to the base range of average service time ($\mu \sim U[25, 35]$), we consider the following three ranges: (a) $\mu \sim U[25, 50]$; (b) $\mu \sim U[50, 60]$; and (c) $\mu \sim U[50, 90]$. In ranges (a) and (c), we increase the variability of service time by extending the range (difference between the lower and upper bounds) of μ from 10 to 25 and 40, respectively. In range (b), we keep the difference between the upper and lower bounds of μ to 10 and increase

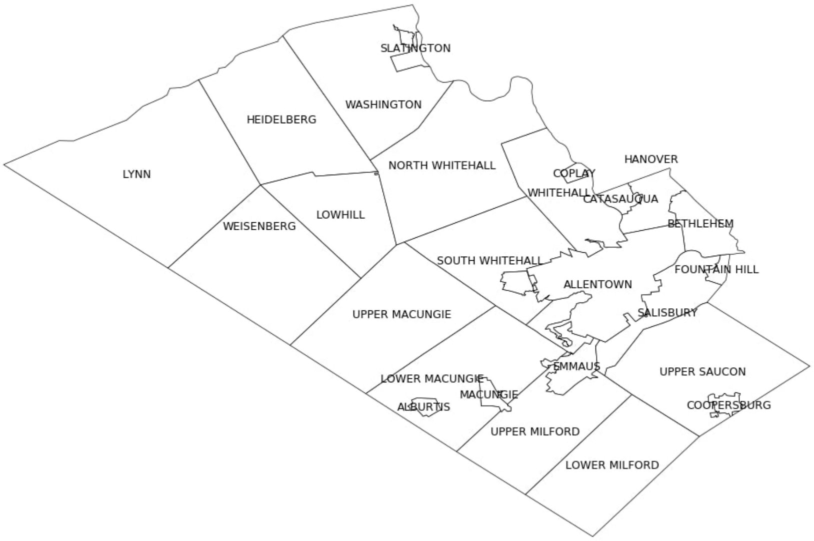

Figure 2: Map of 25 cities in Lehigh County (PA).

the average length of service time. Note that ranges (b) and (c) correspond to applications with longer service time than the base case and range (a). In addition, we consider fixed cost $\lambda^f \in \{50, 100, 1000\}$, $|I| = 6$, and $|K| = 24$ and 45 for L-50 and L-100 instances, respectively. All other parameter settings are the same as described in Section 7.1.

Figure 3 shows the number of hired providers and the second-stage operational cost under each combination of μ and λ^f. We observe the following from this figure. First, we need more providers to serve all customers in the L-100 instance under most ranges of μ and values of λ^f, which makes sense as we have a larger number of customers in this instance. Second, the optimal number of hired providers under the base range and range (a) equals the minimum number of providers required to serve the customers (9 and 18 providers for L-50 and L-100, respectively). This makes sense because under these ranges, the length and variability of service time are lower than the remaining ranges, and both the waiting time and overtime values are low (see Figures 3b and 3d). Thus, hiring additional providers will not improve the second-stage objective but will increase the fixed cost. Third, the optimal number of hired providers under ranges (b) and (c) (i.e., longer service time) with $\lambda^f \in \{50, 100\}$ (i.e., lower fixed cost) is larger than the base range and range (a). Moreover, the optimal number of hired providers under range (c) with $\lambda^f \in \{50, 100\}$ is larger than under range (b). These results make sense because the service time is longer under (b) and (c), and thus, we need more providers to serve customers and mitigate overtime and waiting time. Finally, the optimal number of hired providers under a large value of $\lambda^f = 1000$ equals the minimum number of the required providers to serve the 50/100 customers under all service time ranges. This is because when $\lambda^f = 1000$, the fixed cost is much higher than the operational cost. However, by hiring fewer providers under $\lambda^f = 1000$, the overtime and waiting time worsen, especially under ranges (b) and

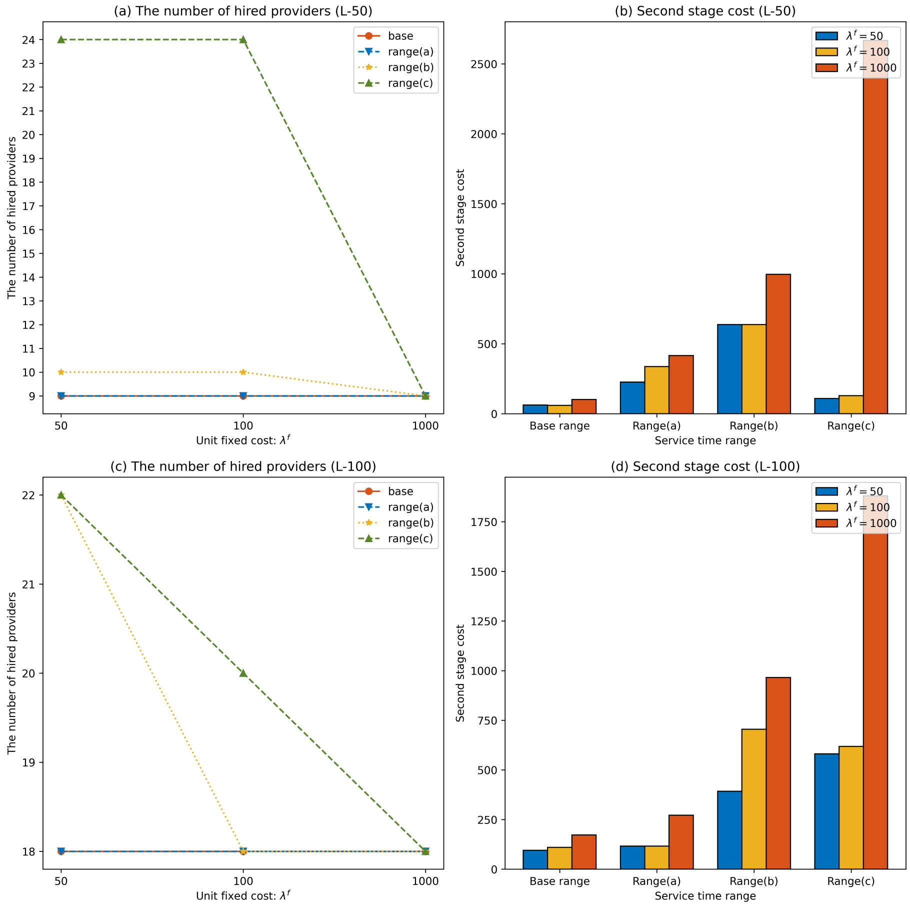

Figure 3: The number of hired providers and associated second stage cost under different λ^f and range of μ.

(c) (see Figures 3b and 3d).

7.4.2. Impact of variability in the unit waiting and overtime costs

In this section, we analyze the optimal number of hired providers as a function of unit waiting time and overtime costs. First, we fix $(\lambda^f, \lambda^t, \lambda^o) = (100, 0.1, 1)$, and vary $\lambda^w \in \{1, 5, 10, 15, 20, 25\}$. For brevity, we discuss results for L-100, and the observations for L-50 are similar. Figure 4 shows the optimal number of hired providers and the associated total waiting time for each combination of λ^w and μ. We observe the following from this figure. First, the optimal number of hired providers under the base range is 18 (i.e., the minimum number of providers required to serve all the 100 customers) irrespective λ^w. This is reasonable because the service time under the base range is lower than the other ranges. Moreover, as shown in Figure 4b, the associated total waiting time under this range is very low. Second, the optimal number of hired providers increases with λ^w under ranges (a)–(c), consequently leading to lower waiting times. These results make sense because the length and variability of service time under ranges (a)–(c) is greater than the base range. Thus, by hiring more providers, we could mitigate excessive waiting time.

Next, we analyze the impact of the unit overtime cost. Figure 5 shows the optimal number of

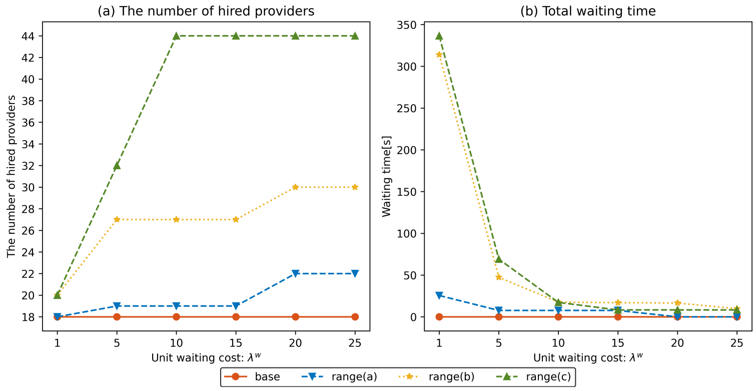

Figure 4: The number of hired providers (a) and total waiting time (b) under different λ^w and range μ.

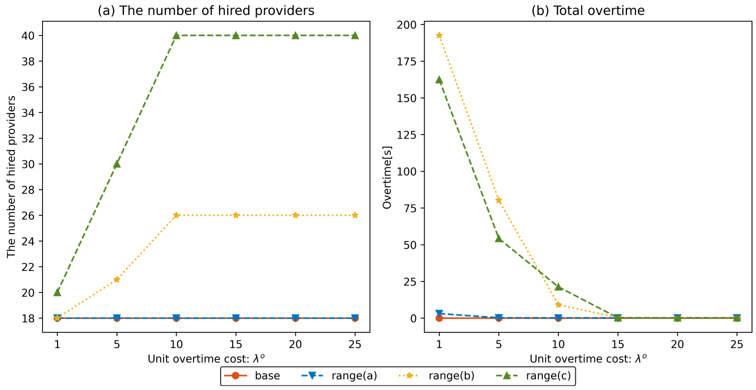

Figure 5: The number of hired providers (a) and total overtime (b) under different λ^w and range μ.

hired providers and total overtime for each combination of λ^o and μ. We observe that the optimal number of hired providers under the base range and range (a) is 18 (i.e., the minimum number of providers required to serve all the 100 customers) irrespective of λ^o. This is because the total overtime under the base range and range (a) is small (see Figure 5b). On the other hand, the optimal number of hired providers under range (c) is greater than the remaining ranges. This is reasonable because the length and variability of service time under range (c) are larger than the remaining ranges, and additional providers are needed to mitigate providers' overtime.

Finally, it is worth mentioning that we observe variability in the actual working time among providers under different ranges of service time and unit overtime cost (see results in Appendix L). Mitigating such variability is out of the scope of this paper but is worth future investigation.

7.4.3. Impact of ignoring uncertainty

Let us now compare the performance of solutions obtained using the proposed SAA model for the HFASP and those of its deterministic counterpart. In the deterministic model, we replace the random travel and service times with their mean values. We compute the total cost of adopting optimal solutions to the deterministic models, denoted as TC $^{\text{DET}}$, as follows. First, we solved the deterministic model and recorded optimal solutions $(\bar{z}, \bar{x}, \bar{a})$. Second, we generated a sample of

1000 scenarios, where each scenario consists of a vector of realizations of service and travel times drawn independently from the distributions corresponding to each customer and pair of customers, respectively. Finally, we used $(\bar{z}, \bar{x}, \bar{a})$ and the generated sample to compute operational metrics (providers' travel time and overtime and customers' waiting time) and costs in each scenario. Then, we computed TC $^{\text{DET}}$ as the fixed cost plus the average operational cost. For illustrative purposes, we use the L-50 instance with $\lambda^f = \{50, 100, 1000\}$ under the base range and range (c) of service time.

Table 12 presents the total cost obtained from the SAA (TC$^{\text{SAA}}$) and deterministic models (TC $^{\text{DET}}$) and the relative difference between them. It is clear that solutions to the deterministic model result in significantly higher total costs than solutions to the proposed stochastic models. Moreover, the difference between TC $^{\text{DET}}$ and TC$^{\text{SAA}}$ is larger under higher variability. For example, the relative difference under range (c) of service time (higher variability) ranges from 21% to 223%, while under the base range (lower variability), it ranges from 4% to 146%. We also observe that the deterministic model often hires fewer providers, resulting in significantly longer providers'overtime and customers'waiting time. For example, when $\lambda^f = 100$, the deterministic model hires nine providers, while the SAA model hires 24 providers. The total waiting time (over the 50 customers) associated with solutions to (deterministic, SAA) models ranges from (367, 0) to (676,0) minutes under the base range and from (2426, 9) to (2489, 740) under range (c). These results demonstrate the importance of incorporating uncertainty in HFASP models and emphasize the negative consequences of ignoring it.

Table 12: Comparison of total cost of the SAA and deterministic models.

	Base range			Range (c)		
λ^f	TC$^{\text{SAA}}$	TC $^{\text{DET}}$	$\frac{\text{TC }^{\text{DET}} - \text{TC}^{\text{SAA}}}{\text{TC }^{\text{DET}}}$	TC$^{\text{SAA}}$	TC $^{\text{DET}}$	$\frac{\text{TC }^{\text{DET}} - \text{TC}^{\text{SAA}}}{\text{TC }^{\text{DET}}}$
50	516.0	1271.3	146%	1340.8	4334.6	223%
100	966.6	1636.9	69%	2552.4	4754.1	86%
1000	9105.1	9468.2	4%	10679.1	12878.4	21%

8. Conclusion

In this paper, we propose and analyze two new SMIPs, denoted as model (S) and model (Z), for the HFASP with random service and travel times. Specifically, given sets of providers and customers, these models aim to find the number of providers to hire, the order of customers assigned to each provider, and an appointment time for each customer. Given the uncertainty in service and travel times, the goal is to minimize the sum of the fixed cost of hiring providers plus the expected cost associated with customers' waiting time and providers' travel time, overtime, and idle time.

The HFASP is an important multiple-vehicle fleet sizing, routing, and scheduling problem that has been studied in closely related contexts with random service time and deterministic travel time. Therefore, in model (Z), we extend an existing model that employs traditional routing variables and constraints for a closely related problem by incorporating the co-existing uncertainty of random travel and service times and providers' idle time. In model (S), we propose a new sequencing-based formulation of the problem. Our theoretical analyses show that model (S) is more compact and provides a tighter LP relaxation, suggesting a better computational performance. Indeed, the computational results demonstrate that significant improvements in computational performance could be gained with model (S) over model (Z). We also propose two computationally efficient heuristics and show that they could quickly obtain near-optimal solutions to large instances of the problem. Finally, we use instances based on Lehigh County of PA to derive insights into the HFASP.

We suggest the following areas for future research. First, our proposed models can be considered the first step towards building comprehensive stochastic optimization approaches for home service staffing, capacity planning, and routing and scheduling, considering all relevant organizational and technical constraints. Second, we have assumed that the planner knows the customers at the time of the planning. However, customer requests for service may arrive randomly in some applications, especially in home healthcare. Moreover, some customers may need multiple visits (e.g., twice a month) instead of one. In this case, one needs a multi-period planning approach. Extending the proposed model to account for these two important aspects, currently simplified in the proposed models, represents an important and interesting future direction. Third, in these extensions, one could consider various sources of uncertainty, such as random demand, capacities, and cancellations. Finally, designing user-friendly decision support tools that implement the models will enable practitioners to adopt them in practice.

Acknowledgement.
We thank all colleagues and practitioners who have contributed significantly to the related literature. We are also grateful to the review team for the thoughtful and constructive feedback that helped improve the exposition of the paper. Dr. Karmel S. Shehadeh dedicates her effort in this paper to every little dreamer in the whole world who has a dream so big and so exciting. Believe in your dreams and do whatever it takes to achieve them—the best is yet to come for you.

References

Ahmadi-Javid, A., Jalali, Z., Klassen, K.J., 2017. Outpatient appointment systems in healthcare: A review of optimization studies. European Journal of Operational Research 258, 3–34.

Alkaabneh, F., Shehadeh, K.S., Diabat, A., 2023. Routing and resource allocation in non-profit settings with equity and efficiency measures under demand uncertainty. Transportation Research Part C: Emerging Technologies 149, 104023.

Anderluh, A., Larsen, R., Hemmelmayr, V.C., Nolz, P.C., 2020. Impact of travel time uncertainties on the solution cost of a two-echelon vehicle routing problem with synchronization. Flexible Services and Manufacturing Journal 32, 806–828.

Berg, B.P., Denton, B.T., Erdogan, S.A., Rohleder, T., Huschka, T., 2014. Optimal booking and scheduling in outpatient procedure centers. Computers & Operations Research 50, 24–37.

Camm, J.D., Raturi, A.S., Tsubakitani, S., 1990. Cutting big M down to size. Interfaces 20, 61–66.

Campbell, A.M., Savelsbergh, M., 2004. Efficient insertion heuristics for vehicle routing and scheduling problems. Transportation Science 38, 369–378.

Cinar, A., Salman, F.S., Bozkaya, B., 2021. Prioritized single nurse routing and scheduling for home healthcare services. European Journal of Operational Research 289, 867–878.

Cissé, M., Yalçındağ, S., Kergosien, Y., Şahin, E., Lenté, C., Matta, A., 2017. OR problems related to home health care: A review of relevant routing and scheduling problems. Operations Research for Health Care 13, 1–22.

Cook, W.J., 2011. In Pursuit of the Traveling Salesman. Princeton University Press.

Denton, B., Viapiano, J., Vogl, A., 2007. Optimization of surgery sequencing and scheduling decisions under uncertainty. Health Care Management Science 10, 13–24.

Di Mascolo, M., Martinez, C., Espinouse, M.L., 2021. Routing and scheduling in home health care: A literature survey and bibliometric analysis. Computers & Industrial Engineering 158, 107255.

Fikar, C., Hirsch, P., 2017. Home health care routing and scheduling: A review. Computers & Operations Research 77, 86–95.

Grieco, L., Utley, M., Crowe, S., 2021. Operational research applied to decisions in home health care: A systematic literature review. Journal of the Operational Research Society 72, 1960–1991.

Gupta, D., Denton, B., 2008. Appointment scheduling in health care: Challenges and opportunities. IIE Transactions 40, 800–819.

Han, S., Zhao, L., Chen, K., Luo, Z.w., Mishra, D., 2017. Appointment scheduling and routing optimization of attended home delivery system with random customer behavior. European Journal of Operational Research 262, 966–980.

Hashemi Doulabi, H., Pesant, G., Rousseau, L.M., 2020. Vehicle routing problems with synchronized visits and stochastic travel and service times: Applications in healthcare. Transportation Science 54, 1053–1072.

Kenyon, A.S., Morton, D.P., 2003. Stochastic vehicle routing with random travel times. Transportation Science 37, 69–82.

Kim, S., Pasupathy, R., Henderson, S.G., 2015. A guide to sample average approximation, in: Handbook of Simulation Optimization. Springer, pp. 207–243.

Kleywegt, A.J., Shapiro, A., Homem-de Mello, T., 2002. The sample average approximation method for stochastic discrete optimization. SIAM Journal on Optimization 12, 479–502.

Klotz, E., Newman, A.M., 2013. Practical guidelines for solving difficult mixed integer linear programs. Surveys in Operations Research and Management Science 18, 18–32.

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

Lecluyse, C., Van Woensel, T., Peremans, H., 2009. Vehicle routing with stochastic time-dependent travel times. 4OR 7, 363–377.

Liu, R., Tao, Y., Xie, X., 2019. An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and synchronized visits. Computers & Operations Research 101, 250–262.

Mancilla, C., Storer, R., 2012. A sample average approximation approach to stochastic appointment sequencing and scheduling. IIE Transactions 44, 655–670.

Homem-de Mello, T., Bayraksan, G., 2014. Monte Carlo sampling-based methods for stochastic optimization. Surveys in Operations Research and Management Science 19, 56–85.

NAHC, 2010. Basic statistics about home care. `http://www.nahc.org/assets/1/7/10hc_stats.pdf`. Accessed: 2022-06-12.

Nikzad, E., Bashiri, M., Abbasi, B., 2021. A matheuristic algorithm for stochastic home health care planning. European Journal of Operational Research 288, 753–774.

Restrepo, M.I., Rousseau, L.M., Vallée, J., 2020. Home healthcare integrated staffing and scheduling. Omega 95, 102057.

Shapiro, A., Dentcheva, D., Ruszczynski, A., 2021. Lectures on Stochastic Programming: Modeling and Theory. SIAM.

Shehadeh, K.S., 2023. Distributionally robust optimization approaches for a stochastic mobile facility fleet sizing, routing, and scheduling problem. Transportation Science 57, 197–229.

Shehadeh, K.S., Cohn, A.E., Epelman, M.A., 2019. Analysis of models for the stochastic outpatient procedure scheduling problem. European Journal of Operational Research 279, 721–731.

Shi, Y., Boudouh, T., Grunder, O., Wang, D., 2018. Modeling and solving simultaneous delivery and pick-up problem with stochastic travel and service times in home health care. Expert Systems with Applications 102, 218–233.

Toth, P., Vigo, D., 2014. Vehicle Routing: Problems, Methods, and Applications. SIAM.

Tsang, M.Y., Shehadeh, K.S., 2023. Stochastic optimization models for a home service routing and appointment scheduling problem with random travel and service times. European Journal of Operational Research 307, 48–63.

Yu, X., Shen, S., Wang, H., 2021. Integrated vehicle routing and service scheduling under time and cancellation uncertainties with application in nonemergency medical transportation. Service Science 13, 172–191.

Yuan, B., Liu, R., Jiang, Z., 2015. A branch-and-price algorithm for the home health care scheduling and routing problem with stochastic service times and skill requirements. International Journal of Production Research 53, 7450–7464.

Zhan, Y., Wan, G., 2018. Vehicle routing and appointment scheduling with team assignment for home services. Computers & Operations Research 100, 1–11.

Zhan, Y., Wang, Z., Wan, G., 2021. Home service routing and appointment scheduling with stochastic service times. European Journal of Operational Research 288, 98–110.

Stochastic Programming Models for a Fleet Sizing and Appointment Scheduling Problem with Random Service and Travel Times (Appendices)

Shutian Li, Karmel S. Shehadeh, Man Yiu Tsang

Appendix A. Comparison between model (S) and those of Zhan and Wan (2018) and Zhan et al. (2021)

Table A1: Summary of the key differences between our proposed model (S) and those of Zhan and Wan (2018) and Zhan et al. (2021).

Models	**Providers**	**Random factors**		**First stage objectives**	**Second stage objectives**			
		TT	ST		PT	WT	OT	IT
Zhan and Wan (2018)	M		✓	Fixed +Det PT		✓	✓	
Zhan et al. (2021)	S		✓	Det PT		✓		✓
Model (S)	M	✓	✓	Fixed	✓	✓	✓	✓

Notation: S is Single provider/vehicle; M is Multiple provider/vehicle; TT is Travel Time; SS is Service Time; PT is Providers' Travel Time; WT is Waiting Time; OT is Overtime; IT is Idle Time; and Det is Deterministic

Appendix B. Model (S) derivation

We show the detailed derivation of model (S). We start with the second-stage formulation without the travel cost. For notational convenience, we suppress the scenario index $n \in [N]$ from the scenario-dependent variables and parameters. For all $i \in I$, we define the actual arrival time of the ith appointment by R_i. It is clear that R_i should satisfy

$$R_1 = \sum_{p \in P} t_{0,p} x_{1,p}$$

$$R_i = \begin{cases} S_{i-1} + \sum\limits_{p \in P} d_p x_{i-1,p} + \sum\limits_{(p_1,p_2) \in P \times P} t_{p_1,p_2} z_{i,p_1,p_2}, & \text{if } \sum\limits_{p \in P} x_{i,p} = 1, \\ 0, & \text{if } \sum\limits_{p \in P} x_{i,p} = 0, \end{cases} \qquad \forall i \in [2, |I|].$$

Next, we define the actual start time of the ith appointment by S_i. Note the ith appointment cannot start before the scheduled appointment time, a_i, nor the provider's actual arrival time R_i. Mathematically, the actual start time of the ith appointment S_i should satisfy.

$$S_i = \max\{R_i, a_i\}, \quad \forall i \in I.$$

If the completion time of all appointments assigned to a provider exceeds the working hours, s/he experiences overtime. We let O represent the provider's overtime, and compute it as follows.

$$O = \left(\max_{i \in I} \left\{ S_i + \sum_{p \in P} (d_p + t_{p,0}) x_{0,p} - L \right\} \right)^+,$$

where, $(b)^+ = \max\{b, 0\}$. If the provider arrives before the appointment time, he/she must wait (i.e., remain idle) until the scheduled appointment time to start the service. Mathematically, we can compute the provider idle time cost before the ith appointment as $\lambda^g(a_i - R_i)^+$. If the provider arrives after the scheduled appointment time, the customer experience waiting. Mathematically, we can compute the waiting cost as $\lambda^w(R_i - a_i)^+$, for all $i \in I$. Accordingly, the second-stage formulation of the HFASP without the travel cost is as follows.

$$\textbf{(P0)} \qquad \min_{S,R,O} \quad \lambda^w \sum_i (R_i - a_i)^+ + \lambda^g \sum_i (a_i - R_i)^+ + \lambda^o O \tag{B.1a}$$

$$\text{s.t.} \quad S_i = \max\{R_i, a_i\}, \quad \forall i \in I, \tag{B.1b}$$

$$R_1 = \sum_{p \in P} t_{0,p} x_{1,p}, \tag{B.1c}$$

$$R_i = \left[S_{i-1} + \sum_{p \in P} d_p x_{i-1,p} + \sum_{(p_1,p_2) \in P \times P} t_{p_1,p_2} z_{i,p_1,p_2} - M\left(1 - \sum_{p \in P} x_{i,p}\right)\right]^+, \quad \forall i \in [2, |I|], \tag{B.1d}$$

$$O = \left(\max_{i \in I} \left\{ S_i + \sum_{p \in P} (d_p + t_{p,0}) x_{0,p} - L \right\} \right)^+. \tag{B.1e}$$

Note that formulation (B.1) is not straightforward to solve in its presented form. Next, in Theorem 3, we derive an equivalent mixed integer program reformulation that is solvable.

Theorem 3. *Problem **(P0)** is equivalent to the following mixed integer programming model **(P2)**.*

$$\textbf{(P2)} \qquad \min_{s,g,o} \quad \lambda^w \sum_{i \in I} (s_i - a_i) + \lambda^g \sum_{i \in I} g_i + \lambda^o o \tag{B.2a}$$

$$\text{s.t.} \quad s_i \geq a_i, \quad \forall i \in I, \tag{B.2b}$$

$$s_1 \geq \sum_{p \in P} t_{0,p} x_{1,p}, \tag{B.2c}$$

$$s_i \geq s_{i-1} + \sum_{p \in P} d_p x_{i-1,p} + \sum_{(p_1,p_2) \in P \times P} t_{p_1,p_2} z_{i,p_1,p_2} - M\left(1 - \sum_{p \in P} x_{i,p}\right), \forall i \in [2, |I|], \tag{B.2d}$$

$$g_1 \geq s_1 - \sum_{p \in P} t_{0,p} x_{1,p}, \tag{B.2e}$$

$$g_i \geq s_i - \left(s_{i-1} + \sum_{p \in P} d_p x_{i-1,p} + \sum_{(p_1,p_2) \in P \times P} t_{p_1,p_2} z_{i,p_1,p_2} \right), \quad \forall i \in [2, |I|], \tag{B.2f}$$

$$o \geq s_i + \sum_{p \in P} (d_p + t_{p,0}) x_{i,p} - L, \quad \forall i \in I, \tag{B.2g}$$

$$(s, g, o) \geq 0. \tag{B.2h}$$

Proof of Theorem 3. Observe that, for all $i \in I$, we have $S_i = \max\{R_i, a_i\} = (R_i - a_i)^+ + a_i = (a_i - R_i)^+ + R_i$. Thus, for a feasible first-stage solution (x, a), the objective function of **(P0)** equals

$$\lambda^w \sum_{i \in I} (S_i - a_i) + \lambda^g \sum_{i \in I} (S_i - R_i) + \lambda^o \left(\max_{i \in I} \left\{ S_i + \sum_{p \in P} (d_p + t_{p,0}) x_{0,p} - L \right\} \right)^+$$

$$
\begin{aligned}
&= \lambda^w \sum_{i\in I}(S_i - a_i) + \lambda^g \left(S_1 - \left[\sum_{p\in P} t_{0,p} x_{1,p} \right] \right) \\
&\quad + \lambda^g \sum_{i=2}^{|I|} \left(S_i - \left[S_{i-1} + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M \left(1 - \sum_{p\in P} x_{i,p} \right) \right]^+ \right) \\
&\quad + \lambda^o \left(\max_{i\in I} \left\{ S_i + \sum_{p\in P} (d_p + t_{p,0}) x_{0,p} - L \right\} \right)^+ .
\end{aligned}
$$

Hence, **(P0)** is equivalent to

$$
\begin{aligned}
\textbf{(P0')} \quad \min_{s} \quad & \lambda^w \sum_{i\in I}(s_i - a_i) + \lambda^g \left(s_1 - \left[\sum_{p\in P} t_{0,p} x_{1,p} \right] \right) \\
& + \lambda^g \sum_{i=2}^{|I|} \left(s_i - \left[s_{i-1} + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M(1 - \sum_{p\in P} x_{i,p}) \right]^+ \right) \\
& + \lambda^o \left(\max_{i\in I} \left\{ s_i + \sum_{p\in P} (d_p + t_{p,0}) x_{0,p} - L \right\} \right)^+ \qquad \text{(B.3a)} \\
\text{s.t.} \quad & s_1 = \max \left\{ \sum_{p\in P} t_{0,p} x_{1,p},\ a_1 \right\}, \qquad \text{(B.3b)} \\
& s_i = \max \left\{ \left[s_{i-1} + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M \left(1 - \sum_{p\in P} x_{i,p} \right) \right]^+,\ a_i \right\}. \qquad \text{(B.3c)}
\end{aligned}
$$

Since the objective function (B.3a) is increasing in $(s_1, s_2, \ldots, s_{|I|})$, we can relax equality constraints (B.3b)–(B.3c) to the following inequalities:

$$s_i \geq a_i, \quad \forall i \in I, \qquad \text{(B.4a)}$$

$$s_1 \geq \sum_{p\in P} t_{0,p} x_{1,p}, \qquad \text{(B.4b)}$$

$$s_i \geq s_{i-1} + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M \left(1 - \sum_{p\in P} x_{i,p} \right), \quad \forall i \in [2, |I|], \qquad \text{(B.4c)}$$

$$s_i \geq 0. \qquad \text{(B.4d)}$$

Next, we can introduce non-negative decision variables o and g satisfying

$$o \geq s_i + \sum_{p\in P} (d_p + t_{p,0}) x_{0,p} - L, \quad \forall i \in I, \qquad \text{(B.5a)}$$

$$g_1 \geq s_1 - \sum_{p\in P} t_{0,p} x_{1,p}, \qquad \text{(B.5b)}$$

$$g_i \geq s_i - s_{i-1} + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M \left(1 - \sum_{p\in P} x_{i,p} \right), \quad \forall i \in [2, |I|], \qquad \text{(B.5c)}$$

which represent the overtime and idle time, respectively. Therefore, **(P0)** is equivalent to the following mixed integer programming model **(P1)**.

$$\textbf{(P1)} \qquad \min_{s,g,o} \quad \lambda^w \sum_{i\in I}(s_i - a_i) + \lambda^g \sum_{i\in I} g_i + \lambda^o o \tag{B.6a}$$

$$\text{s.t.} \quad \text{(B.4a)} - \text{(B.4d)}, \text{(B.5a)} - \text{(B.5c)}, \tag{B.6b}$$

$$(s, g, o) \geq 0. \tag{B.6c}$$

Finally, we show that **(P1)** is equivalent to **(P2)**, where the only difference between **(P1)** and **(P2)** is the big-M constant in constraints (B.5c). To show the equivalence, we let v_1 and v_2 represent the optimal values of **(P1)** and **(P2)**, respectively. We claim that $v_1 = v_2$. From our derivation of **(P1)**, we know that an optimal solution of **(P1)** is given by

$$g_1^* = s_1^* - \sum_{p\in P} t_{0,p} x_{1,p},$$

$$g_i^* = s_i^* - \left[s_{i-1}^* + \sum_{p\in P} d_p x_{i-1,p} + \sum_{(p_1,p_2)\in P\times P} t_{p_1,p_2} z_{i,p_1,p_2} - M\left(1 - \sum_{p\in P} x_{i,p}\right)\right]^+, \quad \forall i \in [2, |I|],$$

$$o^* = \left(\max_{i\in i} \left\{ s_i^* + \sum_{p\in P}(d_p + t_{p,0}) x_{i,p} - L \right\}\right)^+,$$

where s_i^* is the actual appointment start time of ith appointment if $\sum_{p\in P} x_{i,p} = 1$, and is zero otherwise, for all $i \in I$. It is easy to verify that g^* satisfies (B.2e)–(B.2f) and thus, this optimal solution to **(P1)** is feasible to **(P2)**. For a minimization problem, a feasible solution is always an upper bound for the optimal solution. Since the objective functions of **(P1)** and **(P2)** are the same, we must have $v_2 \leq v_1$. On the other hand, constraints (B.5c) is a relaxation of (B.2f) because of the existence of the big-M constant, i.e., the feasible region of **(P1)** contains that of **(P2)**. Hence, we also have $v_1 \leq v_2$. Therefore, we conclude that $v_1 = v_2$, implying that **(P1)** is equivalent to **(P2)**. This completes the proof. □

Appendix C. Proof of Proposition 1

Proof. Consider any provider $k \in K$ and position $i \in [2, |I|]$ in his/her sequence of customers. From constraints (1l), we have

$$s_{i,k}^n \geq s_{i-1,k}^n + \sum_{p\in P} d_p^n x_{i-1,p,k} + \sum_{\substack{(p,q)\in P\times P: \\ p\neq q}} t_{p,q}^n z_{i,p,q,k} - M_i\Big(1 - \sum_{p\in P} x_{i,p,k}\Big).$$

It follows that to preserve optimality when $\sum_{p\in P} x_{i,p,k} = 0$, M_i should be

$$\begin{aligned} M_i &\geq s_{i-1,k}^n + \sum_{p\in P} d_p^n x_{i-1,p,k} + \sum_{\substack{(p,q)\in P\times P: \\ p\neq q}} t_{p,q}^n z_{i,p,q,k} - s_{i,k}^n, \\ &\geq s_{i-1,k}^n + \sum_{p\in P} d_p^n x_{i-1,p,k} + \sum_{\substack{(p,q)\in P\times P: \\ p\neq q}} t_{p,q}^n z_{i,p,q,k} \qquad (\text{since } s_{i,k}^n \geq 0). \end{aligned}$$

By constraints (1k) and (1l), we can recursively find that

$$M_i \geq \max_{k\in K, n\in[N], i\in[2,|I|]} \left\{ s_{1,k}^n + \sum_{j=1}^{i-1} \sum_{p\in P} d_p^n x_{j,p,k} + \sum_{j=2}^{i} \sum_{\substack{(p,q)\in P\times P: \\ p\neq q}} t_{p,q}^n z_{j,p,q,k} \right\}. \tag{C.1}$$

Since $(\lambda^w, \lambda^g, \lambda^o) > 0$, the actual start time of the 1st customer $s^n_{1,k}$ is at most $L+t^{\max}_1$ by constraints (1i) and (1k), for any scenario $n \in [N]$ and provider $k \in K$. It follows from (C.1) that

$$M_i \geq L + t^{\max}_1 + (i-1)\bar{d} + (i-1)\bar{t}, \quad \forall i \in [2, |I|]. \tag{C.2}$$

□

Appendix D. Proof of Proposition 2

Proof. Let us first prove the validity of M in constraints (3k),(3m), and (3n) of model (Z). For constraints (3k) and (3n), we consider the following two cases:

- Case 1: $p = 0, q \in P$. By constraints (3k), we have
$$S^n_q \geq S^n_0 + d^n_0 + t^n_{0,q} - M\bigg(1 - \sum_{k\in K} z_{0,q,k}\bigg), \qquad \forall n \in [N] \tag{D.1a}$$
$$S^n_q \geq t^n_{0,q} - M\bigg(1 - \sum_{k\in K} z_{0,q,k}\bigg), \qquad \forall n \in [N]. \tag{D.1b}$$
Inequalities (D.1b) hold because (1) $S^n_0 = 0$ by constraints (3f), and (2) $d^n_0 = 0$. When $\sum_{k\in K} z_{0,q,k} = 0$, to ensure feasibility, we should satisfy
$$M \geq t^n_{0,q} \qquad \forall n \in [N]. \tag{D.2}$$
Similarly, by constraints (3n), we have
$$M \geq S^n_q - S^n_0 - d^n_0 - t^n_{0,q} - G^n_q \geq S^n_q - t^n_{0,q}, \quad \forall n \in [N]. \tag{D.3}$$

- Case 2: $p \in P, q \in P$. If $\sum_{k\in K} z_{p,q,k} = 0$. By constraints (3k), we have
$$M \geq S^n_p + d^n_p + t^n_{p,q} - S^n_q \geq S^n_p + d^n_p + t^n_{p,q}, \quad \forall n \in [N]. \tag{D.4}$$
By constraints (3n), we have
$$M \geq S^n_q - S^n_p - d^n_p - t^n_{p,q} - G^n_q \geq S^n_q - (S^n_p + d^n_p + t^n_{p,q}), \quad \forall n \in [N]. \tag{D.5}$$

It is easy to verify from constraints (3m) that M should satisfies:

$$M \geq S^n_p + d^n_p + t^n_{p,0} - L - O^n_k \geq S^n_p + d^n_p + t^n_{p,0}, \quad \forall n \in [N]. \tag{D.6}$$

Combining inequalities (D.2)–(D.6), we conclude that M should satisfies:

$$M \geq \max_{p\in P, n\in[N]} \{S^n_p\} + \bar{d} + t^{\max}_2. \tag{D.7}$$

Next, we derive an upper bound on the actual start time. From constraints (3f), we know that each provider will serve at most $|I|$ customers. Moreover, the scheduled appointment time in any feasible solution should be less than or equal to L by constraints (3g). Thus, when $\lambda^o > 0$ and $\lambda^w > 0$, the actual start time of the first customer $p' \in P$ in the sequence of customers assigned to any provider $k \in K$ (i.e., $z_{0,p',k} = 1$) should satisfies: $S^n_{p'} \leq L + t^{\max}_1$. Moreover, it is easy to verify that the actual start time of any customer $p \in P$ satisfies:

$$S^n_p \leq L + t^{\max}_1 + (|I|-1)(\bar{d} + \bar{t}), \quad \forall n \in [N]. \tag{D.8}$$

It follows from inequalities (D.7) and (D.8) that to preserve optimality, it is sufficient to choose the big-M constant as $M \geq L + t^{\max}_1 + |I|\bar{d} + (|I|-1)\bar{t} + t^{\max}_2$. This completes the proof. □

Appendix E. Proof of Theorem 1

Proof. Suppose $(\bar{x}, \bar{z}, \bar{a}, \bar{s}, \bar{g}, \bar{o})$ is an optimal solution to model (S). Below, we construct a feasible solution to model (Z) with the same objective function value.

A) For all $p \in P, k \in K$, let $z_{0,p,k} = \bar{x}_{1,p,k}$ and $z_{p,0,k} = \bar{x}_{|I|,p,k}$. In addition, let $z_{0,0,k} = 1 - \sum_{p\in P} \bar{x}_{1,p,k}$. For a fixed $k \in K$, note that if $\sum_{p\in P} \bar{x}_{1,p,k} = 0$ (i.e., provider k is not hired in the optimal solution to model (S)), we have $\bar{x}_{i,p,k} = 0$ for all $i \in I$ and $p \in P$, by constraints (1e). On the other hand, if $\sum_{p\in P} \bar{x}_{1,p,k} = 1$, then by constraints (2c), there must exist $p' \in P$ such that $\bar{x}_{|I|,p',k} = 1$. Therefore, we conclude that
$$\sum_{p\in P\cup\{0\}} z_{0,p,k} = \sum_{p\in P} z_{0,p,k} + z_{0,0,k} = \sum_{p\in P} \bar{x}_{1,p,k} + \left(1 - \sum_{p\in P} \bar{x}_{1,p,k}\right) = 1,$$
$$\sum_{p\in P\cup\{0\}} z_{p,0,k} = \sum_{p\in P} z_{p,0,k} + z_{0,0,k} = \sum_{p\in P} \bar{x}_{|I|,p,k} + \left(1 - \sum_{p\in P} \bar{x}_{1,p,k}\right) = 1.$$
Thus $z_{0,p,k}$ and $z_{0,p,k}$ satisfy constraints (3e) and (3d), respectively.

B) Let $z_{p,q,k} = \sum_{i=2}^{|I|} (\bar{x}_{i-1,p,k} \cdot \bar{x}_{i,q,k})$ for all $p \in P, q \in P$ and $k \in K$. We first show that $z_{p,q,k} \leq 1$ and hence, binary. Since $\bar{x}$ is binary and satisfies constraints (1b) (i.e., for each $p \in P$, there exists a unique pair $(i,k) \in I \times K$ such that $\bar{x}_{i,p,k} = 1$), we have $\bar{x}_{i_1,p,k} \in \{0,1\}, \bar{x}_{i_2,q,k} \in \{0,1\}$, where $i_1, i_2 \subset I$ with $i_1 \neq i_2$. It follows that $z_{p,q,k} = \sum_{i=2}^{|I|} (\bar{x}_{i-1,p,k} \cdot \bar{x}_{i,q,k})$ is binary, for all $p \in P, q \in P$. Next, we show that $z_{p,q,k}$ satisfies constraints (3b). Since $\bar{x}$ satisfies constraint (2c), we know that, for each $(i,k) \in I \times K$, there exists a $q \in P$ such that $\bar{x}_{i,q,k} = 1$. Thus
$$\begin{aligned}
\sum_{k\in K} \sum_{q\in P\cup\{0\}} z_{p,q,k} &= \sum_{k\in K} (\sum_{q\in P} z_{p,q,k} + z_{p,0,k}) \\
&= \sum_{k\in K} (\sum_{i=2}^{|I|} \sum_{q\in P} \bar{x}_{i-1,p,k} \bar{x}_{i,q,k} + \bar{x}_{|I|,p,k}) \\
&= \sum_{k\in K} \left(\sum_{i=2}^{|I|} \bar{x}_{i-1,p,k} + \bar{x}_{|I|,p,k} \right) && \text{(since } \bar{x} \text{ satisfies constraints (3b))} \\
&= \sum_{k\in K} \sum_{i\in I} \bar{x}_{i,p,k} = 1, && \text{(since } \bar{x} \text{ satisfies constraints (1b))}
\end{aligned}$$
which implies that $z_{p,q,k}$ satisfies constraints (3b).

C) Next, we show that z satisfies constraints (3c). Since $\bar{x}$ satisfies constraints (2c), for each $(i,k) \in I \times K$, there must exists p such that $\bar{x}_{i,p,k} = 1$. Thus, for a fixed (q,k), we have
$$\begin{aligned}
\sum_{p\in P\cup\{0\}} z_{p,q,k} - \sum_{p\in P\cup\{0\}} z_{q,p,k} &= \left[\sum_{p\in P} z_{p,q,k} + z_{0,q,k} \right] - \left[\sum_{p\in P} z_{q,p,k} + z_{q,0,k} \right] \\
&= \left[\sum_{p\in P} \sum_{i=2}^{|I|} (\bar{x}_{i-1,p,k} \bar{x}_{i,q,k}) + \bar{x}_{1,q,k} \right] - \left[\sum_{p\in P} \sum_{i=2}^{|I|} (\bar{x}_{i-1,q,k} \bar{x}_{i,p,k}) + \bar{x}_{|I|,q,k} \right]
\end{aligned}$$

$$= \left[\sum_{i=2}^{|I|} \bar{x}_{i,q,k} + \bar{x}_{1,q,k}\right] - \left[\sum_{i=2}^{|I|} \bar{x}_{i-1,q,k} + \bar{x}_{|I|,q,k}\right] = 1 - 1 = 0.$$

Thus $z_{p,q,k}$ satisfies constraints (3c).

D) Now it remains to show that $z_{p,q,\hat{k}}$ satisfies constraints (4a) for all $p \in P\bigcup\{0\}$, $q \in P\bigcup\{0\}$ and $\hat{k} \in K$. For a fixed $k \in K$, by definition of $z_{p,q,k}$ in points (A) and (B), we have

$$\sum_{p \in P\cup\{0\}} \sum_{q \in P\cup\{0\}} z_{p,q,k} = \sum_{p \in P} \sum_{q \in P} z_{p,q,k} + \sum_{p \in P} z_{p,0,k} + \sum_{q \in P} z_{0,q,k}$$

$$= \sum_{p \in P} \sum_{q \in P} \left(\sum_{i=2}^{|I|} \bar{x}_{i-1,p,k} \bar{x}_{i,q,k} \right) + \sum_{q \in P} \bar{x}_{1,q,k} + \sum_{p \in P} \bar{x}_{|I|,p,k}.$$

By constraints (1e) and (2c), there exist distinctive indices $\{p_1, \cdots, p_{|I|}\}$ such that $\bar{x}_{i,p_i,k} = 1$ for all $i \in \{1, \ldots, |I|\}$. Hence

$$\sum_{p \in P\cup\{0\}} \sum_{q \in P\cup\{0\}} z_{p,q,k} = \sum_{i=2}^{|I|} \bar{x}_{i-1,p_{i-1},k} \bar{x}_{i,p_i,k} + \sum_{q \in P} \bar{x}_{1,q,k} + \sum_{p \in P} \bar{x}_{|I|,p,k}$$

$$= \bar{x}_{1,p_1,k}\bar{x}_{2,p_2,k} + \bar{x}_{2,p_2,k}\bar{x}_{3,p_3,k} + \cdots + \bar{x}_{|I|-1,p_{|I|-1},k}\bar{x}_{|I|,p_{|I|},k} + 1 + 1$$

$$= (|I| - 1) + 1 + 1 = |I| + 1,$$

which implies that $z_{p,q,k}$ satisfies constraints (4a).

E) Let $A_p = \sum_{i \in I} \sum_{k \in K} \bar{a}_{i,k} \bar{x}_{i,p,k}$, for all $p \in P$. Since $\bar{a}_{i,k}$ satisfies constraints (1b), by constraints (2d), we have

$$A_p = \sum_{i \in I} \sum_{k \in K} \bar{a}_{i,k} \bar{x}_{i,p,k} \leq L \sum_{i \in I} \sum_{k \in K} \bar{x}_{i,p,k} = L, \quad \forall p \in P.$$

Thus, A_p satisfies constraints (3g).

F) Let $S^n_p = \sum_{i \in I} \sum_{k \in K} \bar{s}^n_{i,k} \bar{x}_{i,p,k}$ for all $p \in P$ and $n \in [N]$. Since $\bar{s}^n_{i,k}$ satisfies constraints (1j), by constraints (1b) and point (E), we have

$$S^n_p = \sum_{i \in I} \sum_{k \in K} \bar{s}^n_{i,k} \bar{x}_{i,p,k} \geq \sum_{i \in I} \sum_{k \in K} \bar{a}_{i,k} \bar{x}_{i,p,k} = A_p.$$

Thus, S^n_p satisfies constraints (3j).

G) To verify that S^n_p satisfies constraints (3k), for each $n \in [N]$, we consider the following cases.

- Case 1: $p = 0, q \in P$. In this case, let $S^n_0 = d^n_0 = 0$, for a fixed $q \in P$, constraints (3k) are equivalent to

  $$S^n_q \geq S^n_0 + d^n_0 + t^n_{0,q} - M(1 - \sum_{k \in K} z_{0,q,k}) = t^n_{0,q} - M(1 - \sum_{k \in K} z_{0,q,k}).$$

  If $\bar{x}_{1,q,k} = 0$ for all $k \in K$, then by point (A), we have $z_{0,q,k} = 0$ for all $k \in K$. Thus, constraints (3k) are relaxed with a sufficiently large number M because $\sum_{k \in K} z_{0,q,k} = 0$. One the other hand, if $\bar{x}_{1,q,\hat{k}} = 1$ for some $\hat{k} \in K$ then $z_{0,q,\hat{k}} = \sum_{k \in K} z_{0,q,k} = 1$ (by point (A)). From the definition of S^n_q in point (F), we have $S^n_q = \bar{s}^n_{1,\hat{k}}$. Then we have,

  $$S^n_q - S^n_p = S^n_q - S^n_0 = \bar{s}^n_{1,\hat{k}} - 0$$

$$\begin{aligned} &\geq \sum_{p' \in P} t^n_{0,p'} \bar{x}_{1,p',\hat{k}} && \text{(since } \bar{s}^n_{1,\hat{k}} \text{ satisfies constraints (1k))} \\ &= t^n_{0,q}. && \text{(since } \sum_{p' \in P} \bar{x}_{1,p',\hat{k}} = \bar{x}_{1,q,\hat{k}} = 1) \end{aligned}$$

- Case 2: Consider the case when $(p, q) \in P \times P$. Without loss of generality, we assume that $p \neq q$. By constraints (1b), there must exist $\bar{x}_{i_1,p,k_1} = \bar{x}_{i_2,q,k_2} = 1$, for some $\{i_1, i_2\} \in I$ and $\{k_1, k_2\} \in K$. By definition of $z_{p,q,k}$ in point (B), it is trivial that if $k_1 \neq k_2$, then $\sum_{k \in K} z_{p,q,k} = 0$, and constraints (3k) are relaxed. Now, consider the first case when $\bar{x}_{\hat{i}-1,p,\hat{k}} = \bar{x}_{\hat{i},q,\hat{k}} = 1$ for some $\hat{i} \in [2, |I|]$ and $\hat{k} \in K$. By constraints (1f)-(1h), we have $\bar{z}_{\hat{i},p,q,\hat{k}} = \bar{x}_{\hat{i}-1,p,\hat{k}} \bar{x}_{\hat{i},q,\hat{k}} = 1$. Accordingly, by definition of S in point (F), we have $S^n_q = \bar{s}_{\hat{i},\hat{k}}$ and $S^n_p = \bar{s}^n_{\hat{i}-1,\hat{k}}$. Hence,

$$\begin{aligned} S^n_q - S^n_p &= \bar{s}^n_{\hat{i},\hat{k}} - \bar{s}^n_{\hat{i}-1,\hat{k}} \\ &\geq \sum_{p' \in P} d^n_{p'} \bar{x}_{\hat{i}-1,p',k} + \sum_{(p',q') \in P \times P} t^n_{p',q'} \bar{z}_{i,p',q',k} && \text{(since } \bar{s}_{\hat{i},\hat{k}} \text{ and } \bar{s}^n_{\hat{i}-1,\hat{k}} \text{ satisfy (2f))} \\ &= d^n_p + t^n_{p,q}. && \text{(since } \bar{x}_{\hat{i},q,\hat{k}} = \bar{x}_{\hat{i}-1,p,\hat{k}} = \bar{z}_{\hat{i},p,q,\hat{k}} = 1) \end{aligned}$$

  Consider the second case when $\bar{x}_{i_1,p,k_1} = \bar{x}_{i_2,q,k_2} = 1$ with $i_2 > i_1 + 1$. In this case, $\bar{x}_{i-1,p,k} \cdot \bar{x}_{i,q,k} = 0$ for all $i \in [2, |I|]$ and $k \in K$ (i.e., customer p and q are not visited consecutively by the same provider). By the definition of z in point (B), we have $\sum_{k \in k} z_{p,q,k} = 0$, then constraints (3k) are relaxed to $S^n_p \geq 0$.

Accordingly, we conclude that S^n_p satisfies constraints (3k).

H) Let $W^n_p = \sum_{i \in I} \sum_{k \in K} (\bar{s}^n_{i,k} - \bar{a}_{i,k}) \bar{x}_{i,p,k}$. By definitions of S and A in points (E) and (F) respectively, for a fixed $p \in P$ and for each $n \in [N]$, we have

$$W^n_p = \sum_{i \in I} \sum_{k \in K} \bar{s}^n_{i,k} \bar{x}_{i,p,k} - \sum_{i \in I} \sum_{k \in K} \bar{a}_{i,k} \bar{x}_{i,p,k} = S^n_p - A_p.$$

Thus, W^n_p satisfies constraints (3l).

I) Let $O^n_k = \bar{o}^n_k$ for all $k \in K$ and $n \in [N]$. For a fixed $k \in K$, if $\bar{x}_{|I|,p',k} = z_{p',0,k} = 1$ for some $p' \in P$. Then, by constraints (1o), for each $n \in [N]$ and for a fixed $i \in I$, we have

$$\begin{aligned} O^n_k = \bar{o}^n_k &\geq \bar{s}^n_{i,k} + \sum_{p \in P} (d^n_p + t^n_{p,0}) \bar{x}_{|I|,p,k} - L \\ &\geq \bar{s}^n_{i,k} + d^n_{p'} + t^n_{p',0} - L. && \text{(since } \sum_{p' \in P} \bar{x}_{|I|,p',k} = \bar{x}_{|I|,p,k} = 1) \end{aligned} \tag{E.1}$$

Consider the inequality (E.1) when $i = |I|$. By definitions of S in point (F), we have $\bar{s}^n_{|I|,k} = S^n_{p'}$. Therefore, inequality (E.1) is equivalent to

$$O^n_k = \bar{o}^n_k \geq S^n_{p'} + d^n_{p'} + t^n_{p',0} - L.$$

On the other hand, if $\bar{x}_{|I|,p,k} = z_{p,0,k} = 0$ for all $p \in P$, constraints (1o) are relaxed to $\bar{o}^n_k \geq 0$ with the non-negativity of the decision variable $\bar{o}$. Since $z_{p,0,k} = 0$ for all $p \in P$, constraints

(3m) are also relaxed to $O_k^n \geq 0$ with a sufficiently large number M. Consequently, we have

$$O_k^n \geq S_p^n + d_p^n + t_{p,0}^n - L - M(1 - z_{p,0,k}), \quad \forall p \in P,$$

which implies that O_k^n satisfies constraints (3m).

J) Let $G_p^n = \sum_{i\in I}\sum_{k\in K} \bar{x}_{i,p,k}\bar{g}_{i,k}^n$ for all $p \in P$ and $n \in [N]$. To verify that G_p^n satisfies constraints (3n), for each $n \in [N]$, we consider the following two cases.

- Case 1: $p = 0, q \in P$. Let $S_0^n = d_0^n = 0$. In this case, for a fixed $q \in P$, constraints (3n) are equivalent to

$$\begin{aligned} G_q^n &\geq S_q^n - S_p^n - d_p^n - t_{p,q}^n - M\left(1 - \sum_{k\in K} z_{p,q,k}\right) \\ &\geq S_q^n - S_0^n - d_0^n - t_{0,q}^n - M\left(1 - \sum_{k\in K} z_{0,q,k}\right) = S_q^n - t_{0,q}^n - M\left(1 - \sum_{k\in K} z_{0,q,k}\right). \end{aligned}$$

  If $\bar{x}_{1,q,k} = z_{0,q,k} = 0$ for all $k \in K$, constraints (3n) are relaxed. Otherwise, if $\bar{x}_{1,q,k'} = z_{0,q,k'} = 1$ for some $k' \in K$, we have $G_q^n = \bar{g}_{1,k'}^n$. It follows that

$$\begin{aligned} G_q^n = \bar{g}_{1,k'}^n &\geq \bar{s}_{1,k'}^n - \sum_{\bar{p}\in P} t_{0,\bar{p}}^n \bar{x}_{1,\bar{p},k'} && \text{(since } \bar{g}_{1,k'}^n \text{ satisfies constraints (1m))} \\ &\geq \bar{s}_{1,k'}^n - t_{0,q}^n && \text{(since } \sum_{\bar{p}\in P} \bar{x}_{1,\bar{p},k'} = \bar{x}_{1,q,k'} = 1) \\ &\geq S_q^n - t_{0,q}^n. && \text{(since } S_q^n = \bar{s}_{1,k'}^n \text{ by point (F))} \end{aligned}$$

- Case 2: $(p,q) \in P \times P, p \neq q$. If $\bar{x}_{i'-1,p,k'} = \bar{x}_{i',q,k'} = 1$ for some $i' \in [2,|I|]$ and $k' \in K$, by definition of S in point (F), we have $S_p^n = \bar{s}_{i'-1,k'}^n$ and $S_q^n = \bar{s}_{i',k'}^n$, and accordingly $G_p^n = \bar{g}_{i'-1,k'}^n$ and $G_q^n = \bar{g}_{i',k'}^n$. Hence by constraints (1n), we have

$$G_q^n = \bar{g}_{i',k'}^n \geq \bar{s}_{i',k'}^n - \bar{s}_{i'-1,k'}^n - \sum_{\bar{p}\in P} d_{\bar{p}}^n \bar{x}_{i'-1,\bar{p},k'} - \sum_{(\bar{p},\bar{q})\in P\times P} t_{\bar{p},\bar{q}}\bar{z}_{i',\bar{p},\bar{q},k'} = S_q^n - S_p^n - d_p^n - t_{p,q}^n.$$

  Note that if $\bar{x}_{i_1,p,k_1} = \bar{x}_{i_2,q,k_2} = 1$ for some $\{i_1, i_2\} \in I$ such that $i_2 > i_1 + 1$ and $\{k_1, k_2\} \in K$. In this case, $\bar{x}_{i-1,p,k} \cdot \bar{x}_{i,q,k} = 0$ for all $i \in [2, |I|]$ and $k \in K$ (i.e., customers p and q are not visited consecutively by the same provider). Then we have $\sum_{k\in K} z_{p,q,k} = 0$ by the conclusion of point (B), and constraints (3n) are relaxed to $G_p^n \geq 0$.

Accordingly, we conclude that G_p^n satisfies constraints (3n).

Therefore, we conclude that (z, A, W, S, G, O) defined above is a feasible solution of model (Z). The objective value of this solution equals to

$$\sum_{k\in K}\sum_{p\in P} \lambda^f z_{0,p,k} + \sum_{n\in[N]} \frac{1}{N}\left[\lambda^t \sum_{k\in K}\sum_{p\in P\cup\{0\}}\sum_{q\in P\cup\{0\}} t_{p,q}^n z_{p,q,k} + \lambda^w \sum_{p\in P} W_p^n + \lambda^o \sum_{k\in K} O_k^n + \lambda^g \sum_{p\in P} G_p^n\right].$$

Note that by construction (see point (A)), we have

$$\sum_{k\in K}\sum_{p\in P} \lambda^f z_{0,p,k} = \sum_{k\in K}\sum_{p\in P} \lambda^f \bar{x}_{1,p,k}, \tag{E.2}$$

and by the construction in point (B), we have

$$\lambda^t \sum_{k\in K} \sum_{p\in P\cup\{0\}} \sum_{q\in P\cup\{0\}} t^n_{p,q} z_{p,q,k} = \lambda^t \sum_{k\in K} \left(t^n_{p,q} \sum_{(p,q)\in P\times P} z_{p,q,k} + \sum_{q\in P} t^n_{0,q} z_{0,q,k} + \sum_{p\in P} t^n_{p,0} z_{p,0,k} \right)$$

$$= \lambda^t \sum_{k\in K} \left(\sum_{i=2}^{|I|} \sum_{(p,q)\in P\times P} t^n_{p,q,} \bar{x}_{i-1,p,k} \bar{x}_{i,q,k} + \sum_{p\in P} t^n_{p,0} \bar{x}_{1,p,k} + \sum_{q\in P} t^n_{0,q} \bar{x}_{|I|,p,k} \right). \tag{E.3}$$

We define $\bar{z}_{i,p,q,k} = \bar{x}_{i-1,p,k}\bar{x}_{i,q,k}$ for all $i \in [2,|I|], p \in P, q \in P$ and $k \in K$. By McCormick inequalities (constraints (1f)-(1h)), for each $n \in [N]$, equation (E.3) is equivalent to

$$\lambda^t \sum_{k\in K} \sum_{p\in P\cup\{0\}} \sum_{q\in P\cup\{0\}} t^n_{p,q} z_{p,q,k} = \lambda^t \sum_{k\in K} \left(\sum_{i=2}^{|I|} \sum_{(p,q)\in P\times P} t^n_{p,q,} \bar{z}_{i,p,q,k} + \sum_{p\in P} t^n_{p,0} \bar{x}_{1,p,k} + \sum_{q\in P} t^n_{0,q} \bar{x}_{|I|,p,k} \right). \tag{E.4}$$

By the construction in point (H), we have for each $n \in [N]$

$$\lambda^w \sum_{p\in P} W^n_p = \lambda^w \sum_{i\in I}\sum_{k\in K} (\bar{s}^n_{i,k} - \bar{a}_{i,k}) x_{i,p,k} = \lambda^w \sum_{i\in I}\sum_{k\in K} (\bar{s}^n_{i,k} - \bar{a}_{i,k}). \tag{E.5}$$

By the construction in point (I), we have for each $n \in [N]$

$$\lambda^o \sum_{k\in K} O^n_k = \lambda^o \sum_{k\in K} \bar{o}^n_k. \tag{E.6}$$

Finally, by the construction in point (J), we have for each $n \in [N]$

$$\lambda^g \sum_{p\in P} G^n_p = \lambda^g \sum_{i\in I}\sum_{k\in K} \bar{g}^n_{i,k} x_{i,p,k} = \lambda^g \sum_{i\in I}\sum_{k\in K} \bar{g}^n_{i,k}. \tag{E.7}$$

Combining equations (E.2)–(E.7), we show that the objective function value of model (Z) is equal to the objective function value of model (S).

Conversely, suppose that (z, A, S, W, G, O) is an optimal solution to model (Z), we will construct a feasible solution $(\bar{x}, \bar{z}, \bar{a}, \bar{s}, \bar{g}, \bar{o})$ to model (S) with the same objective value.

A) By constraints (3b)–(3e) and (4a), for a fixed $k \in K$, there exists a unique sequence $\{p^k_1, \ldots, p^k_{|I|}\}$ such that $z_{0,p^k_1,k} = 1$, $z_{p^k_i,p^k_{i+1},k} = 1$ for all $i \in [1, |I|-1]$ and $z_{p^k_{|I|},0,k} = 1$. Then, for each $p^k_i \in P$ and $i \in I$, we define $\bar{x}_{i,p^k_i,k} = 1$. Consequently, $\bigcup_{\bar{k}\in K}\{p^{\bar{k}}_1, p^{\bar{k}}_2, \ldots, p^{\bar{k}}_{|I|}\}$ forms a partition of P, implying that $\bar{x}$ satisfies constraints (1b) and (2c).

B) Using the construction in point (A), we let $\bar{z}_{i,p,q,k} = \bar{x}_{i-1,p,k}\bar{x}_{i,q,k}$, for all $i \in [2, |I|], p \in P, q \in P$ and $k \in K$, for a given $\bar{x}_{i-1,p,k}$ and $\bar{x}_{i,q,k}$. One can easily verify that $\bar{z}$ satisfies constraints (1f)–(1h) by McCormick inequalities.

C) Let $\bar{a}_{i,k} = \sum_{p\in P} A_p \bar{x}_{i,p,k}$ for all $i \in I$ and $k \in K$. For a fixed $(i', k') \in I \times K$, we have $\bar{x}_{i',p_{i'},k'} = 1$ (from the construction in point (A)). Then, by constraints (3g), we have

$$\bar{a}_{i',k'} = \sum_{p\in P} A_p \bar{x}_{i',p,k'} = A_{p_{i'}} \le L. \text{ (Thus } \bar{a}_{i,k} \text{ satisfies constraints (2d)).}$$

D) Let $\bar{s}^n_{i,k} = \sum_{p\in P} S^n_p \bar{x}_{i,p,k}$ for all $i \in I, k \in K$ and $n \in [N]$. By the result of point (A), for a fixed $(i', k') \in I \times K$, there exists $p_{i'} \in P$ such that $\bar{x}_{i',p_{i'},k'} = 1$. Since S satisfies constraints

(3g) and (3j) and $\bar{a} \leq L$ (as shown in point (C)), for each $n \in [N]$, we have

$$\bar{s}^n_{i',k'} = \sum_{p \in P} S^n_p \bar{x}_{i',p,k'} = S^n_{p_{i'}} \geq A_{p_{i'}} = \sum_{p \in P} A_p \bar{x}_{i',p,k'} = \bar{a}_{i',k'},$$

which satisfies constraints (1j). It remains to verify that $\bar{s}^n_{i,k} = \sum_{p \in P} S^n_p \bar{x}_{i,p,k}$ satisfies constraints (1k) and (2f). Consider the following two cases for a fixed $k \in K$ and each $n \in [N]$:

- $i = 1$. By construction of $\bar{x}$ and conclusion of point (A), there must exist $p^k_1 \in P$ such that $z_{0,p^k_1,k} = \bar{x}_{1,p^k_1,k} = 1$. Since $S^n_0 = 0$ (by constraints (3i)) and $d^n_0 = 0$, we have

$$\begin{aligned} \bar{s}^n_{1,k} &= \sum_{p \in P} S^n_p \bar{x}_{1,p,k} = S^n_{p^k_1} \\ &\geq S^n_0 + d^n_0 + t^n_{0,p^k_1} - M(1 - \sum_{k' \in K} z_{0,p^{k'}_1,k'}) \quad \text{(since } S^n_{p^k_1} \text{ satisfies constraints (3k))} \\ &= t^n_{0,p^k_1} = \sum_{p \in P} t_{0,p} \bar{x}_{1,p,k}. \quad \text{(since } \bar{x} \text{ satisfies constraints (2c))}. \end{aligned}$$

- $i \in [2, |I|]$. Given $i' \in [2, |I|]$, by definitions of $\bar{x}$ and $\bar{z}$ in points (A) and (B), there must exist $(p^k_{i'-1}, p^k_{i'}) \in P \times P$ such that $\bar{x}_{i'-1,p^k_{i'-1},k} = \bar{x}_{i',p^k_{i'},k} = \bar{z}_{i',p^k_{i'-1},p^k_{i'},k} = 1$. By constraints (3k), we have

$$\begin{aligned} \bar{s}^n_{i',k} = \sum_{\bar{p} \in P} S^n_{\bar{p}} \bar{x}_{i',\bar{p},k} = S^n_{p^k_{i'}} &\geq S^n_{p^k_{i'-1}} + d^n_{p^k_{i'-1}} + t^n_{p^k_{i'-1},p^k_{i'}} - M\left(1 - \sum_{k' \in K} z_{p^{k'}_{i'-1},p^{k'}_{i'},k'}\right) \\ &= \bar{s}^n_{i'-1,k} + \sum_{\bar{p} \in P} d^n_{\bar{p}} \bar{x}_{i'-1,\bar{p},k} + \sum_{(\bar{p},\bar{q}) \in P \times P} t_{\bar{p},\bar{q}} \bar{z}_{i',\bar{p},\bar{q},k}. \end{aligned}$$

Hence, we conclude that $\bar{s}$ satisfies constraints (1k) and (2f).

E) Let $\bar{g}^n_{i,k} = \sum_{p \in P} G^n_p \bar{x}_{i,p,k}$ for all $i \in I, k \in K$ and $n \in [N]$. To show that $\bar{g}$ satisfies constraints (1m) and (1n), we consider the following two cases for a fixed $k \in K$ and each $n \in [N]$.

- $i = 1$. By construction of $\bar{x}$ in point (A), there must exist $p^k_1 \in P$ such that $z_{0,p^k_1,k} = \bar{x}_{1,p^k_1,k} = 1$. By definition of $\bar{s}$ in point (D), we know that $\bar{s}^n_{1,k} = S^n_{p^k_1}$. Using the fact that $S^n_0 = 0$, we have

$$\begin{aligned} \bar{g}^n_{1,k} &= \sum_{p \in P} G^n_p \bar{x}_{1,p,k} = G^n_{p^k_1} \\ &\geq S^n_{p^k_1} - t^n_{0,p^k_1} - M(1 - \sum_{k' \in K} z_{0,p^{k'}_1,k'}) \quad \text{(since } G^n_{p^k_1} \text{ satisfies constraints (3n))} \\ &= \bar{s}^n_{1,k} - t^n_{0,p^k_1} = \bar{s}^n_{1,k} - \sum_{p \in P} t^n_{0,p} \bar{x}_{1,p,k}. \quad \text{(since } \sum_{p \in P} \bar{x}_{1,p,k} = \bar{x}_{1,p^k_1,k} = 1) \end{aligned}$$

- $i \in [2, |I|]$. Given $i' \in [2, |I|]$, by definitions of $\bar{x}$ and $\bar{z}$ in points (A) and (B), there must exist $(p^k_{i'-1}, p^k_{i'}) \in P \times P$ such that $\bar{x}_{i'-1,p^k_{i'-1},k} = \bar{x}_{i',p^k_{i'},k} = \bar{z}_{i',p^k_{i'-1},p^k_{i'},k} = 1$. Using the definition of $\bar{s}$ in point (D), we know that $\bar{s}^n_{i',k} = S^n_{p^k_{i'}}$ and $\bar{s}^n_{i'-1,k} = S^n_{p^k_{i'-1}}$. Then we have

$$\bar{g}^n_{i',k} = \sum_{\bar{p} \in P} G^n_{\bar{p}} \bar{x}_{i',\bar{p},k} = G^n_{p^k_{i'}}$$

$$\geq S^n_{p^k_{i'}} - S^n_{p^k_{i'-1}} - d^n_{p^k_{i'-1}} - t^n_{p^k_{i'-1},p^k_{i'}} - M(1 - \sum_{k' \in K} z_{p^{k'}_{i'-1},p^{k'}_{i'},k'}) \quad \text{(by constraints (3n))}$$
$$= \bar{s}^n_{i',k} - \bar{s}^n_{i'-1,k} - \sum_{\bar{p} \in P} d^n_{\bar{p}} \bar{x}_{i'-1,\bar{p},k} - \sum_{(\bar{p},\bar{q}) \in P \times P} t_{\bar{p},\bar{q}} \bar{z}_{i',\bar{p},\bar{q},k}.$$

Consequently, we conclude that $\bar{g}$ satisfies constraints (1m) and (1n).

F) Let $\bar{o}^n_k = O^n_k, \forall k \in K, n \in [N]$. If $z_{p,0,k} = 0, \forall p \in P \cup \{0\}$ for a fixed $k \in K$, then constraints (3m) are relaxed to $O^n_k \geq 0$, for each $n \in [N]$. Now, given $\bar{x}$ from point (A), we know that there must exist $p^k_{|I|} \in P$ such that $z_{p^k_{|I|},0,k} = \bar{x}_{|I|,p^k_{|I|},k} = 1$. Using $\bar{s}$ as defined in point (D), we have

$$\bar{o}^n_k = O^n_k \geq S^n_{p'} + d^n_{p'} + t^n_{p',0} - L - M(1 - \sum_{k' \in K} z_{p',0,k'}) \quad \text{(since } O^n_k \text{ satisfies constraints (3m))}$$
$$\geq \bar{s}^n_{|I|,k} + \sum_{p \in P} d^n_p \bar{x}_{|I|,p,k} + \sum_{p \in P} t^n_{p,0} \bar{x}_{|I|,p,k} - L. \quad \text{(since } \sum_{p \in P} \bar{x}_{|I|,p,k} = \bar{x}_{|I|,p^k_{|z|},k} = 1)$$

Thus, $\bar{o}$ satisfies constraints (2g).

From points (A)–(F), we conclude that $(\bar{x}, \bar{z}, \bar{a}, \bar{s}, \bar{g}, \bar{o})$ is a feasible solution of model (S). The objective value of this solution equals

$$\sum_{k \in K} \sum_{j \in P} \lambda^f \bar{x}_{1,p,k} + \sum_{n \in [N]} \frac{1}{N} \Bigg[\lambda^t \sum_{k \in K} \Bigg(\sum_{i=2}^{|I|} \sum_{(p,q) \in P \times P} t^n_{p,q,} \bar{z}_{i,p,q,k} + \sum_{p \in P} t^n_{p,0} \bar{x}_{1,p,k} + \sum_{q \in P} t^n_{0,q} \bar{x}_{|I|,p,k} \Bigg)$$
$$+ \lambda^w \sum_{i \in I} \sum_{k \in K} (\bar{s}^n_{i,k} - \bar{a}_{i,k}) + \lambda^o \sum_{k \in K} \bar{o}^n_k + \lambda^g \sum_{i \in I} \sum_{k \in K} \bar{g}^n_{i,k} \Bigg] \tag{E.8}$$

Using the logic similar to equations (E.2)–(E.7), we conclude that (E.8) is equivalent to

$$\sum_{k \in K} \sum_{p \in P} \lambda^f z_{0,p,k} + \sum_{n \in [N]} \frac{1}{N} \Bigg[\lambda^t \sum_{k \in K} \sum_{p \in P \cup \{0\}} \sum_{q \in P \cup \{0\}} t^n_{p,q} z_{p,q,k} + \lambda^w \sum_{p \in P} W^n_p + \lambda^o \sum_{k \in K} O^n_k + \lambda^g \sum_{p \in P} G^n_p \Bigg]. \tag{E.9}$$

i.e., the optimal value of model (S) equals the optimal value of model (Z). □

Appendix F. Proof of Theorem 2

We provide proof of Theorem 2, showing that the linear programming relaxation (LPR) of model (S) for the partially used provider case provides a tighter lower bound than the LPR of model (Z). We let LP(S) and LP(Z) respectively represent the optimal objective values of LPR(S) and LPR(Z). Our proof has the following steps. First, we derive a valid upper bound on LP(Z). Second, we derive a lower bound on LP(S). Then, we compare the difference between these bounds.

STEP 1. In Theorem 4, we construct a feasible solution for LPR(Z) and use it to establish an upper bound on the optimal objective value of LPR(Z), denoted as LP(Z).

Theorem 4. *The optimal objective value of LPR(Z), denoted as LP(Z), satisfies*

$$LP(Z) \leq UB_Z = \Bigg[\lambda^f |P| + \sum_{n \in [N]} \sum_{p \in P} \frac{\lambda^t}{N} (t^n_{0,p} + t^n_{p,0}) \Bigg] \frac{\bar{d}}{V}, \tag{F.1}$$

for some large positive constant $V > 0$ (e.g., $V = M$).

Proof of the Theorem 4. We construct the following feasible solution to LPR(Z). In this solution we let $V = M = L + (\bar{d}|I| + 1)(t_1^{\max} + t_2^{\max}) + (|I| - 1)\bar{t}$. Given the number of positions on the serving sequence $|I|$ and the set of providers K, we consider a partition of the set of customers $P = \bigcup_{k\in K}\{P_1, P_2, \ldots, P_k\}$ such that $P_k \cap P_{k'} = \varnothing$ and $0 \leq |P_k| \leq |I|$, for all $k \in K, k' \in K : k \neq k'$. Then, we use this partition to construct the following feasible solution (z, A, S, W, G, O) to LPR(Z).

- For any provider $k \in K$ and $(p, q) \in (P \cup \{0\}) \times (P \cup \{0\})$,
$$z_{p,q,k} = \begin{cases} 1 - |P_k|\frac{\bar{d}}{M}, & \text{if } p = q = 0 \\ \frac{\bar{d}}{M}, & \text{if } p = 0, q \in P_k \text{ or } q = 0, p \in P_k \\ 1 - \frac{\bar{d}}{M}, & \text{if } p, q \in P_k \times P_k : p = q \\ 0 & \text{otherwise} \end{cases} \tag{F.2}$$
- $A_p = 0, \quad \forall p \in P$; $S_p^n = 0, \quad \forall p \in P, n \in [N]$; $G_p^n = 0, \quad \forall p \in P, n \in [N]$; $W_p^n = 0, \quad \forall p \in P, n \in [N]$; and $O_k^n = 0, \quad \forall k \in K, n \in [N]$.

Let us first show that this solution is a feasible solution for LPR(Z). First, we show that $z_{p,q,k} \in [0, 1]$. Since $M \geq |I|\bar{d}$ and $|P_k| \leq |I|$ by construction, then $0 \leq \frac{\bar{d}}{M} \leq 1$, $0 \leq 1 - \frac{\bar{d}}{M} \leq 1$, and $0 \leq 1 - |P_k|\frac{\bar{d}}{M} \leq 1$. It follows that $z_{p,q,k} \in [0, 1]$, for all $p \in P \cup \{0\}, q \in P \cup \{0\}$ and $k \in K$.

Next, we show that solution (z, A, S, W, G, O) as defined above is a feasible solution to LPR(Z) by verifying that it satisfies all constraints.

A) By construction, for any $p \in P$, there must exist a unique $\bar{k} \in K$ such that $p \in P_{\bar{k}}$, i.e., $z_{p,q,\bar{k}}$ assumes value as defined in (F.2) and $z_{p,q,k} = 0$, $\forall k \neq \bar{k}$, $q \notin P_{\bar{k}}$. Accordingly, we have
$$\sum_{k\in K}\sum_{q\in P\cup\{0\}} z_{p,q,k} = \sum_{q\in P_{\bar{k}}\cup\{0\}} z_{p,q,\bar{k}} = \sum_{q\in P_{\bar{k}}} z_{p,q,\bar{k}} + z_{p,0,\bar{k}} = z_{p,p,\bar{k}} + z_{p,0,\bar{k}} = 1 - \frac{\bar{d}}{M} + \frac{\bar{d}}{M} = 1.$$
Thus, $\boldsymbol{z}$ defined in (F.2) satisfies constraints (3b).

B) We show that $\boldsymbol{z}$ defined in (F.2) satisfies constraints (3c). Similar to the argument in point (A), for any $q \in P$, we can always find a unique $\bar{k} \in K$ such that $q \in P_{\bar{k}}$. Thus, if $k \in K \setminus \{\bar{k}\}$, we have $z_{p,q,k} = 0$ for all $p \in P \setminus \{q\}$, which satisfies constraints (3c). When $k = \bar{k}$, we have
$$\begin{aligned} \sum_{k\in K}\sum_{p\in P\cup\{0\}} z_{p,q,k} - \sum_{k\in K}\sum_{p\in P\cup\{0\}} z_{q,p,k} &= \sum_{p\in P\cup\{0\}} z_{p,q,\bar{k}} - \sum_{p\in P\cup\{0\}} z_{q,p,\bar{k}} \\ &= \left(\sum_{p\in P} z_{p,q,\bar{k}} + z_{0,q,\bar{k}}\right) - \left(\sum_{p\in P} z_{q,p,\bar{k}} + z_{q,0,\bar{k}}\right) \\ &= (z_{q,q,\bar{k}} + z_{0,p,\bar{k}}) - (z_{q,q,\bar{k}} + z_{q,0,\bar{k}}) \\ &= (1 - \frac{\bar{d}}{M} + \frac{\bar{d}}{M}) - (1 - \frac{\bar{d}}{M} + \frac{\bar{d}}{M}) = 0. \end{aligned}$$
Thus, $\boldsymbol{z}$ defined in (F.2) satisfies constraints (3c).

C) For any provider $k \in K$, there is a set of customers $P_k \subset P$ by the partition of P defined in our construction. If $P_k = \varnothing$, by construction, we have

$$\sum_{p\in P\cup\{0\}} z_{p,0,k} = z_{0,0,k} + \sum_{p\in P_k} z_{p,0,k} = z_{0,0,k} = 1 - 0\cdot\frac{\bar{d}}{M} = 1.$$

On the other hand, when $P_k \neq \varnothing$, we have $z_{p,q,k} = 0$ for all $(p,q) \in (P \setminus \{P_k\}) \times (P \setminus \{P_k\})$. Then

$$\sum_{p\in P\cup\{0\}} z_{p,0,k} = z_{0,0,k} + \sum_{p\in P_k} z_{p,0,k} = \left(1 - |P_k|\frac{\bar{d}}{M}\right) + \sum_{p\in P_k}\frac{\bar{d}}{M} = \left(1 - |P_k|\frac{\bar{d}}{M}\right) + |P_k|\frac{\bar{d}}{M} = 1.$$

Similarly, we have $\sum_{p\in P\cup\{0\}} z_{0,p,k} = z_{0,0,k} + \sum_{p\in P_k} z_{0,p,k} = 1$. Thus, $\boldsymbol{z}$ defined in (F.2) satisfies constraints (3d) and (3e).

D) For a provider $k \in K$ and corresponding customer set P_k, If $P_k = \varnothing$, then following the same argument in point (C), we have $z_{0,0,k} = 1$ and it is trivial that constraints (3f) hold. Now, consider the case when $P_k \neq \varnothing$, we have

$$\begin{aligned}\sum_{p\in P\cup\{0\}}\sum_{q\in P\cup\{0\}} z_{p,q,k} &= \sum_{p\in P_k}\sum_{q\in P_k} z_{p,q,k} + \sum_{q\in P_k} z_{0,q,k} + \sum_{p\in P_k} z_{p,0,k} + z_{0,0,k}\\ &= \sum_{p\in P_k} z_{p,p,k} + \sum_{q\in P_k} z_{0,q,k} + \sum_{p\in P_k} z_{p,0,k} + z_{0,0,k}\\ &= |P_k|\left(1 - \frac{\bar{d}}{M}\right) + 2|P_k|\frac{\bar{d}}{M} + \left(1 - |P_k|\frac{\bar{d}}{M}\right)\\ &= |P_k| + 1 \leq |I| + 1 \qquad \text{(since } |P_k| \leq |I| \text{ by construction)}\end{aligned}$$

Thus, $\boldsymbol{z}$ defined in (F.2) satisfies constraints (3f).

E) Since $A_p^n = S_p^n = 0$ for all $p \in P, n \in [N]$, they satisfy constraints (3g), (3i), (3j), and (3l).

F) To verify that S_p^n and $z_{p,q,k}$ satisfy constraints (3k), we need to check if they satisfy

$$S_p^n + d_p^n + t_{p,q}^n - M(1 - \sum_{k\in K} z_{p,q,k}) \leq 0, \quad \forall p \in P\cup\{0\}, q \in P.$$

To this end, we consider the following three cases:

- Case 1: $p = 0$ and $q \in P$. In this case, following the conclusion in point (A), we have $\sum_{k\in K} z_{0,q,k} = \frac{\bar{d}}{M}$ by construction. Then, for any $q \in P$, we have

$$\begin{aligned}S_0^n + d_0^n + t_{0,q}^n - M(1 - \frac{\bar{d}}{M}) &= t_{0,q}^n - M(1 - \frac{\bar{d}}{M}) \qquad \text{(since } S_0^n = d_0^n = 0 \text{ by construction)}\\ &\leq t_1^{\max} - M + \bar{d} \leq 0 \qquad \text{(since } M > t_1^{\max} + \bar{d}\text{).}\end{aligned}$$

- Case 2: $p \in P, q \in P$ and $p = q$. By construction, we have $\sum_{k\in K} z_{p,q,k} = \sum_{k\in K} z_{p,p,k} = 1 - \frac{\bar{d}}{M}$. Thus, for any $p \in P$, we have

$$S_p^n + d_p^n + t_{p,p}^n - M\left(1 - \sum_{k\in K} z_{p,p,k}\right) = d_p^n - M\frac{\bar{d}}{M} \leq 0 \;\; \text{(since } \bar{d} = \max_{n\in[N],p\in P}\{d_p^n\}\text{).}$$

- Case 3: $p \in P, q \in P$ and $p \neq q$. In this case $\sum_{k\in K} z_{p,q,k} = 0$ by construction. Then, for any $p \in P$ and $q \in P$, we have

$$S_p^n + d_p^n + t_{p,q}^n - M = d_p^n + t_{p,q}^n - M \leq 0 \text{ (since } M > \max_{n\in[N],(p,q)}\{t_{p,q}^n\} + \max_{n\in[N],p\in P}\{d_p^n\}\text{).}$$

Thus, the constructed solution satisfies constraints (3k).

G) We verify that the constructed solution satisfies constraints (3m). First, if $P_k = \varnothing$ for any $k \in K$, then $z_{p,0,k} = 0$ for all $p \in P$ by construction. In this case, constraints (3m) trivially hold. Second, consider the case when P_k is not empty. In this case, we have $z_{p,0,k} = \bar{d}/M$ for all $p \in P_k$ by construction. Next, we check if the following inequalities hold.

$$O_k^n - S_p^n - d_p^n - t_{p,0}^n + L + M(1 - z_{p,0,k}) \geq 0, \quad \forall p \in P \tag{F.3}$$

Since $O_k^n = S_p^n = 0, \forall k \in K, p \in P, n \in [N]$, inequalities (F.3) reduce to

$$L + M - \bar{d} - d_p^n - t_{p,0}^n \geq 0, \quad \forall p \in P. \tag{F.4}$$

Inequalities (F.4) hold valid since $M > |I|\bar{d} + t_2^{\max} + L = |I| \max_{n \in [N], p \in P}\{d_p^n\} + \max_{n \in [N], p \in P}\{t_{p,0}^n\} + L > \bar{d} + d_p^n + t_{p,0}^n$. Thus, constraints (3m) are satisfied.

H) To check the feasibility of constraints (3n), we need to show that, for any $p \in P \cup \{0\}, q \in P$ and $n \in [N]$, $G_q^n - S_p^n + S_p^n + d_p^n + t_{p,q}^n + M(1 - \sum_{k \in K} z_{p,q,k}) \geq 0$. Similar to point (G), we consider the following three cases:

- Case 1: $p = 0, q \in P$. In this case, $\sum\limits_{k \in K} z_{0,q,k} = \frac{\bar{d}}{M}$. For any $n \in [N]$ and $q \in P$, we have
$$G_p^n - S_p^n + S_p^n + d_p^n + t_{0,p}^n + M(1 - \frac{\bar{d}}{M}) = d_p^n + t_{0,p}^n + M - M\frac{\bar{d}}{M} \geq 0.$$
The result follows because $G_p^n = S_p^n = 0, \forall p \in P, n \in [N]$ by construction and $M > \bar{d} + t_1^{\max}$.
- Case 2: $p \in P, q \in P$ and $p = q$. In this case, $\sum_{k \in K} z_{p,p,k} = 1 - \frac{\bar{d}}{M}$. Thus, for any $p \in P, n \in [N]$, we have
$$G_p^n - S_p^n + S_p^n + d_p^n + t_{p,p}^n + M\frac{\bar{d}}{M} = d_p^n + M\frac{\bar{d}}{M} \geq 0 \ \ \text{(since } (d_p^n, \bar{d}) \geq 0).$$
- Case 3: $p \in P, q \in P$ and $p \neq q$. In this case, $\sum_{k \in K} z_{p,q,k} = 0$. By constraints (3n), for all $p \in P, q \in P$ and $n \in [N]$, we have
$$G_q^n - S_q^n + S_p^n + d_p^n + t_{p,q}^n + M = d_p^n + t_{p,q}^n + M \geq 0 \ \ \text{(since } (d_p^n, t_{p,q}^n, M) \geq 0).$$
Thus, the constructed solution satisfies constraints (3n).

From points (A)–(H), we conclude that the constructed solution (z, A, S, W, G, O) is a feasible solution for LPR(Z). The objective function value UB_Z of this feasible solution is as follows:

$$UB_Z = \sum_{k \in K} \lambda^f |P_k| \frac{\bar{d}}{V} + \sum_{n \in [N]} \sum_{p \in P} \frac{\lambda^t}{N}[t_{0,p}^n + t_{p,0}^n]\frac{\bar{d}}{V}.$$

Since this is a feasible solution to LPR(Z), the objective value of this solution denoted as UB_Z provides an upper bound on the optimal objective value, $LP(Z)$, of LPR(Z), i.e., $LP(Z) \leq UB_Z$. Moreover, since $\sum_{k \in K} |P_k| = |P|$, we have

$$LP(Z) \leq UB_Z = \left[\lambda^f |P| + \sum_{n \in [N]} \sum_{p \in P} \frac{\lambda^t}{N}(t_{0,p}^n + t_{p,0}^n)\right]\frac{\bar{d}}{V}.$$

This completes the proof of Theorem 4. □

STEP 2. In Theorem 5, we derive a lower bound on the optimal objective value LP(S) of LPR(S)

Theorem 5. *The optimal objective value of the LPR(S), denoted as LP(S), satisfies*

$$LP(S) \geq \frac{|P|}{|I|}(\lambda^f + \lambda^t \cdot t_1^{min}) = LB_S.$$

Proof of the Theorem 5. Let (x, a, z, s, g, o) be an optimal solution to LPR(S). The objective value of this solutions is

$$LP(S) = \sum_{p\in P}\sum_{k\in K} \lambda^f x_{1,p,k} + \sum_{n\in[N]} \frac{1}{N}\Bigg\{\lambda^t \Bigg[\sum_{\substack{(p,p')\in P\times P \\ p\neq p'}} \sum_{i\in I}\sum_{k\in K} t^n_{p,p'} z_{i,p,p',k} + \sum_{p\in P} t^n_{0,p} x_{1,p,k}$$

$$+ \sum_{p\in P} t^n_{p,0} x_{0,p,k}\Bigg] + \sum_{k\in K}\sum_{i\in I}\Bigg[\lambda^w (s^n_{i,k} - a_{i,k}) + \lambda^g g^n_{i,k}\Bigg] + \sum_{k\in K} \lambda^o o^n_k \Bigg\}$$

Since decision variables (x, z, s, a, g, o) are non-negative, we have

$$\begin{aligned} LP(S) &\geq \sum_{p\in P}\sum_{k\in K} \lambda^f x_{1,p,k} + \sum_{n\in[N]} \frac{\lambda^t}{N}\Bigg[\sum_{p\in P}\sum_{k\in K} t^n_{0,p} x_{1,p,k}\Bigg] \\ &\geq \sum_{p\in P}\sum_{k\in K} \lambda^f x_{1,p,k} + \lambda^t \Bigg[\sum_{p\in P}\sum_{k\in K} t_1^{\min} x_{1,p,k}\Bigg] \\ &= (\lambda^f + \lambda^t t_1^{\min}) \sum_{p\in P}\sum_{k\in K} x_{1,p,k}. \end{aligned} \tag{F.5}$$

Next, we claim that for any $k \in K$, there exists some $p' \in P$ such that

$$x_{1,p',k} \geq \frac{1}{|I||K|} \tag{F.6}$$

We prove this claim by contradiction. Suppose, on the contrary, that $x_{1,p,k} < 1/|I||K|$ for all $p \in P$ and $k \in K$. Summing over the customers set P we have

$$\sum_{p\in P} x_{1,p,k} < \sum_{p\in P} 1/|I||K = |P|/|I||K|, \quad \forall k \in K. \tag{F.7}$$

Form constraints (1e), we know that any feasible solution to LPR(S) should satisfies $\sum_{p\in P} x_{i,p,k} \leq \sum_{p\in P} x_{1,p,k}$, for all $i \in [2, |I|]$ and $k \in K$. Thus, from (F.7), we have $\sum_{p\in P} x_{i,p,k} < |P|/|I||K|$, for all $i \in [2, |I|]$ and $k \in K$. Summing over the position set I and provider set K, we have

$$\sum_{i\in I}\sum_{p\in P}\sum_{k\in K} x_{i,p,k} < \frac{|P|}{|I||K|} \cdot |I||K| = |P|. \tag{F.8}$$

Note that any feasible solution should also satisfy constraints (1b), i.e., $\sum_{i\in I}\sum_{k\in K} x_{i,p,k} = 1$, $\forall p \in P$, which implies that $\sum_{i\in I}\sum_{p\in P}\sum_{k\in K} x_{i,p,k} = |P|$. However, from (F.8), we have $\sum\limits_{i\in I}\sum\limits_{p\in P}\sum\limits_{k\in K} x_{i,p,k} < |P|$. Hence, we have a contradiction. This complete the proof of our claim. Summing inequality (F.6) over the customer set P and provider set K, we obtain

$$\sum_{p\in P}\sum_{k\in K} x_{1,p,k} \geq \frac{|P|}{|I||K|} \cdot |K| = \frac{|P|}{|I|} \tag{F.9}$$

Combining (F.9) with (F.5), we conclude

$$LP(S) \geq \frac{|P|}{|I|}\big(\lambda^f + \lambda^t t_1^{\min}\big).$$

This completes the proof of Theorem 5. □

STEP 3. From Theorem 4, we know that the optimal objective value of LPR(Z) is not greater than the constructed upper bound UB_Z in (F.1), i.e.,

$$LP(Z) \leq UB_Z = \left[\lambda^f|P| + \sum_{n\in[N]}\sum_{p\in P}\frac{\lambda^t}{N}(t^n_{0,p} + t^n_{p,0})\right]\frac{\bar{d}}{V} \leq \left[\lambda^f + \lambda^t(t^{\max}_1 + t^{\max}_2)\right]\frac{\bar{d}|P|}{V}, \qquad \text{(F.10)}$$

where the last inequality follows from $t^{\max}_1 = \max\limits_{p\in P, n\in[N]}\{t^n_{0,p}\}$ and $t^{\max}_2 = \max p \in P, n \in [N]\{t^n_{p,0}\}$. On the other hand, from Theorem 5, we know that the optimal objective value of LPR(S) satisfies $LP(S) \geq LB_S = \frac{|P|}{|I|}\big(\lambda^f + \lambda^t t^{\min}_1\big)$. The difference between the lower bound on LP(S) (LB_S) and the upper bound of LPR(Z) defined in (F.10) is as follows

$$\frac{V - |I|\bar{d}}{V|I|}|P|\lambda^f + \left[\frac{Vt^{\min}_1 - |I|\bar{d}(t^{\max}_1 + t^{\max}_2)}{|I|V}\right]|P|\lambda^t \qquad \text{(F.11)}$$

Since $V > (\bar{d}|I|+1)(t^{\max}_1 + t^{\max}_2)$, the first and second terms in (F.11) are greater than zero, i.e., i.e., the difference between the lower bound on LP(S) and the upper bound on LP(Z) is positive. This indicates that LPR(S) is tighter than LPR(Z). Indeed, the numerical results in Sections 7.2.1 and 7.2.2 show that LPR(S) is strictly tighter than LPR(Z).

Appendix G. Sample average approximation and sample size

We use the SAA method with Monte Carlo Optimization (MCO) procedure to decide the sample size for the SAA model. We refer to Kenyon and Morton (2003) and Kleywegt et al. (2002) for details of the algorithm. We initialize the MCO procedure with sample size N, simulation sample size N', and the number of replicates M. In each replicate $m \in [M]$, we first solve the SAA problem with sample size N, obtain the optimal solution $\hat{x}^m_N$, and the optimal objective value v^m_N. Second, we solve the second-stage problem with $\hat{x}^m_N$ and N' scenarios to compute $v^m_{N'}$. We repeat these steps M times, each time with new N and N' scenarios of service and travel time sampled from their distributions. Finally, we compute the average of the SAA objective value $\bar{v}_N = \frac{1}{M}\sum_{k\in K} v^m_N$ and the simulated objective values $\bar{v}_{N'} = \frac{1}{M}\sum_{k\in K} v^m_{N'}$. As detailed in Kenyon and Morton (2003) and Kleywegt et al. (2002), $\bar{v}_N$ and $\bar{v}_{N'}$ respectively represents statistical lower and upper bounds on the optimal value of the HFASP. Thus, we estimate the Approximate Optimality Index $AOI = \frac{v_{N'} - v_N}{v_{N'}}$.

We implement the algorithm using model (S) for partially used providers with the problem instance $|I| = 6, |P| = 24, |K| = 5$ under the cost structure defined in Section 7.1. We run the experiment with the sample size N ranging from 1 to 100. For each value of N, we repeat the algorithm ten times ($M = 10$) and choose the Monte Carlo simulation sample size $N' = 10000$. We present SAA objective value v_N, objective function value of simulation $v_{N'}$, AOI, and their 95% Confidence Interval (95% CI) in Table G1. This table shows that AOI with $N = 50$ equals 0.01%. In addition, the 95%CI of $\bar{v}_{N=50}$ and $\bar{v}_{N'}$ are very tight. These results qualify $v_{N=50}$ as a tight estimator of the optimal value.

Table G1: v_N, $v_{N'}$, and AOI of the partially used model with $|I| = 6$, $|P| = 24$ and $|K| = 5$.

N	$\bar{v}_N$	$\bar{v}_{N'}$	$\lvert AOI(\%)\rvert$	95% CI $\bar{v}_N$	95%CI $\bar{v}_{N'}$
1	4052.98	4186.69	3.19	[4050,4056]	[4171,4203]
5	4053.23	4061.81	0.21	[4052,4054]	[4059,4065]
10	4054.45	4057.24	0.07	[4054,4055]	[4056,4058]
20	4054.57	4056.65	0.05	[4054,4055]	[4056,4057]
30	4054.49	4058.98	0.11	[4054,4055]	[4058,4060]
40	4054.50	4055.81	0.03	[4054,4055]	[4056,4056]
50	4055.06	4055.42	0.01	[4055,4056]	[4055,4056]
60	4054.73	4055.58	0.02	[4054,4055]	[4055,4056]
70	4055.03	4056.34	0.03	[4055,4055]	[4056,4057]
80	4054.70	4054.47	0.01	[4054,4055]	[4054,4055]
90	4054.59	4055.20	0.02	[4054,4055]	[4055,4055]
100	4054.82	4054.74	0.01	[4055,4055]	[4055,4055]

Appendix H. Symmetry breaking constraints

Suppose there are three homogeneous providers $K = \{1, 2, 3\}$, i.e., they share the same hiring cost λ^f and have same service time distribution. Then, solutions $(\sum_{p\in P} x_{1,p,1} = 1, \sum_{p\in P} x_{1,p,2})$, $(\sum_{p\in P} x_{1,p,1} = 1, \sum_{p\in P} x_{1,p,3} = 1)$, and $(\sum_{p\in P} x_{1,p,2} = 1, \sum_{p\in P} x_{1,p,3} = 1)$ are equivalent (i.e., yield the same objective) in the sense that they all permit hiring 2 out of 3 providers. To prevent wasting time exploring such equivalent solutions, we assume that providers are numbered sequentially and add constraints (H.1) to model (S). Similarly, we add constraints (H.2) to model (Z) to enforce that provider k is hired before provider $k + 1$.

$$\sum_{p\in P} x_{1,p,k} \geq \sum_{p\in P} x_{1,p,k+1}, \quad \forall k \in [1, |K| - 1]. \tag{H.1}$$

$$\sum_{p\in P} z_{0,p,k} \geq \sum_{p\in P} z_{0,p,k+1}, \quad \forall k \in [1, |K| - 1]. \tag{H.2}$$

Appendix I. Additional computational results for model (S) for fully used provider

We provide additional computational time results of model (S) for fully used providers. We present the results in two parts. In the first part, we analyze the solution times of model (S) and compare them with that of model (Z) under 100 scenarios. In the second part, we present solution times of model (S) and the comparison results of solution times of models (S) and (Z) under another cost structure in the objective. All the remaining experiment settings are the same as Section 7.1.

We start with reporting solution times of model (S) using N =100 scenarios in Table I1. First, we observe that solution time increases when N increases from 50 to 100 scenarios. In fact, the average solution time of model (S) with $|I| = 6$ ranges from 5.4 seconds ($|P| = 24$) to 35 minutes ($|P| = 54$), with $|I| = 8$ ranges from 7.1 seconds ($|P| = 24$) to 47 minutes ($|P| = 56$). In contrast, using model (Z), we were only able to solve instances with $|I| = 6, |P| \leq 18$ or $|I| = 8, |P| \leq 16$. We compare the ratios of solution times of model (Z) and (S) in Table I2. Clearly, solution times of model (Z) are longer than model (S). Next, we present numerical results under a different cost structure (hereafter denoted as cost structure 2). Specifically, as in Yu et al. (2021), we set the unit

Table I1: Solution time (in seconds) of model (S) for fully used providers with 100 scenarios.

model (S)	$\lvert I\rvert = 6$			$\lvert I\rvert = 8$			
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max
24	4.8	5.4	5.9	24	5.5	7.1	12.4
30	11.6	13.9	19.5	32	20.3	22.9	25.4
36	28.7	31.9	39.7	40	47.3	67.0	114.2
42	88.6	390.7	682.6	48	793.0	1142.3	1656.3
48	248.9	658.4	1240.4	56	2388.3	2860.7	3500.3
54	1339.3	2116.3	2817.1	64	-	-	-

When $|I| = 8, |P| \geq 32$, we set the relative MIP gap to 0.04.

Table I2: Ratios of solution times of models (Z) and (S) on the SAAs solved by both with $N = 100$ (fully used)

$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$	$\lvert I\rvert = 6$			$\lvert I\rvert = 8$			
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max
6	1.6	2.4	3.9	8	0.1	0.7	2.0
12	2.0	3.2	5.5	16	16.1	25.1	31.8
18	148.7	187.7	305.0	24	-	-	-

Table I3: Solution time (in seconds) of model (S) for fully used providers with cost structure 2.

$\lvert I\rvert = 6$	$(\lambda_w, \lambda_o, \lambda_g) = (2, 10, 0)$		
$\lvert P\rvert$	Min	Avg	Max
24	3.2	3.4	3.7
30	7.9	9.0	11.6
36	18.0	21.5	28.3
42	26.5	142.2	192.4
48	155.7	261.6	392.9
54	245.0	861.4	1993.4
60	110.5	777.6	2221.9

overtime cost $\lambda^o = 10$, and unit waiting cost $\lambda^w = 2$. The other elements in the cost structure are the same. We generate unit travel cost λ^t from $U[0.1, 0.5]$ (Zhan and Wan, 2018) and set the fixed cost of hiring one provider λ^f to 1000 based on real-world applications. In Table I3, we present the solution time of model (S) under cost structure 2 and 50 scenarios. We observe that using model (S), the average solution time ranges from 3.4 seconds ($|P| = 24$) to 13 minutes ($|P| = 60$).

Finally, we compare the solution time of two models with cost structure 2 and present the ratio of solution times of models (S) and (Z) in Table I4. Observe that, model (Z) takes a longer time to solve all instances than model (S). For those instances that model (Z) failed to solve, it terminated with the average relMIP around 6%.

Table I4: Ratios of solution time (in seconds) of models (Z) and (S) on the SAAs solved by both with cost structure 2. Results are for fully used models.

$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$	$(\lambda_w, \lambda_o, \lambda_g) = (2, 10, 0)$							
	$\lvert I\rvert = 6$				$\lvert I\rvert = 8$			
$\lvert P\rvert$	Min	Avg	Max	$\lvert P\rvert$	Min	Avg	Max	
6	2.0	2.5	3.1		8	2.0	2.5	3.1
12	1.1	2.0	2.2		16	-	-	-
18	250.6	335.0	447.8		24	-	-	-

Table J1: Solution time (in seconds) of model (S) for partially used providers with 100 scenarios.

model (S)	$\lvert I\rvert = 6$			$\lvert I\rvert = 8$		
$\lvert P\rvert$	Min	Avg	Max	Min	Avg	Max
24	7.8	9.6	10.8	16.6	17.3	22.3
30	27.6	31.1	34.0	51.8	65.8	80.9
36	60.9	64.7	69.8	331.3	443.1	525.0
40	106.7	121.6	134.3	1127.3	2419.8	3089.4
42	121.5	146.0	178.5	914.2	1218.5	1532.8
48	965.3	1361.3	1590.5	-	-	-
50	991.3	1661.9	2334.3	-	-	-
54	1166.0	2520.7	3170.7	-	-	-

Appendix J. Additional computational results for model (S) for partially used provider

In this section, we provide additional computational time results of model (S) for partially used providers with 100 scenarios and cost structure 2. The experiment settings are same as what we discussed in Section 7.1, and details about cost structure 2 are described in Appendix I.

First, we report the Min, Avg and Max solution time of generated instances with $N = 100$ in Table J1. We observe that the solution time of model (S) with $|I| = 6$ varies from 7.8 seconds ($|P| = 24$) to 53 minutes ($|P| = 54$), and with $|I| = 8$ varies from 17.3 seconds ($|P| = 24$) to 20 minutes ($|P| = 42$). In contrast, model (Z) was able to solve only instances with $|I| = \{6, 8\}, |P| \leq 8$. We compare the ratios of solution times of model (Z) and (S) in Table J2. It is clear that solution time of model (Z) is longer than that of model (S). For those instances that model (Z) failed to solve, it terminated with the average relMIP gap around 74% ($|I| = 6$) and 100% ($|I| = 8$).

Finally, we present results with cost structure 2 in Table J3. We observe that the average solution time of model (S) with $|I| = 6$ ranges from 7.1 ($|P| = 24$) seconds to 9.5 minutes ($|P| = 62$). In contrast, model (Z) can only solve instances with $|I| \subset \{6, 8\}, |P| \leq 8$. The average relMIP at termination is 100%. We present the comparison of solution times between models (S) and (Z) in Table J4. It is clear that model (Z) takes a longer time to solve all instances than model (S).

Table J2: Ratios of solution time of models (Z) and (S) on the SAAs solved by both with $N = 100$ (partially used).

$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$	$\lvert I \rvert = 6$			$\lvert I \rvert = 8$		
$\lvert P \rvert$	Min	Avg	Max	Min	Avg	Max
6	2.2	5.1	7.8	1.1	2.9	3.9
8	86.2	95.8	100.1	9.2	21.0	54.0

Table J3: Solution time (in seconds) of model (S) for partially used providers with cost structure 2.

$\lvert I \rvert = 6$	$(\lambda_w, \lambda_o, \lambda_g) = (2, 10, 0)$		
$\lvert P \rvert$	Min	Avg	Max
24	6.1	7.1	9.0
30	13.5	14.5	15.4
36	16.2	16.8	18.3
40	23.7	25.5	27.7
42	28.3	29.0	30.1
48	43.4	46.6	52.3
50	60.7	132.4	340.1
54	81.4	85.6	96.0
58	115.7	197.2	311.2
60	119.7	136.4	150.0
62	163.6	574.9	1599.9

Table J4: Ratios of solution time of models (S) and (Z) on the SAAs solved by both with cost structure 2. Results are for partially used models.

	$(\lambda_w, \lambda_o, \lambda_g) = (2, 10, 0)$					
$\frac{\text{(Z) sol.time}}{\text{(S) sol.time}}$	$\lvert I \rvert = 6$			$\lvert I \rvert = 8$		
$\lvert P \rvert$	Min	Avg	Max	Min	Avg	Max
6	1.5	1.7	1.8	0.8	1.0	1.3
8	69.8	73.6	80.5	3.3	3.5	4.0

Appendix K. Details of Lehigh County instances

Table K1: The number of customers in each city/township of Lehigh Valley Instance.

City	**Pop**	**Pop%**	**L-50**	**L-100**	**City**	**Pop**	**Pop%**	**L-50**	**L-100**
Alburtis	2596	0.7	1	1	Bethlehem	25868	6.6	3	7
Coplay	3348	0.9	1	1	Hanover	11783	3	1	3
Heidelberg	3324	0.9	1	1	Lower Macungie	32426	8.3	4	8
Lower Milford	3861	1	1	1	Lynn	4232	1.1	1	1
Lowhill	2292	0.6	1	1	Salisbury	13621	3.5	1	3
North Whitehall	15655	4	2	4	South Whitehall	21080	5.4	2	5
Weisenberg	4976	1.3	1	1	Upper Macungie	26377	6.8	3	7
Catasauqua	6518	1.7	1	2	Upper Milford	7777	2	1	2
Coopersburg	2447	0.6	1	1	Upper Saucon	16973	4.3	2	4
Emmaus	11652	3	1	3	Washington	6551	1.7	1	2
Fountain Hill	4832	1.2	1	1	Whitehall	29173	7.5	3	7
Macungie	3257	0.8	1	1	Allentown	125845	32.2	14	32
Slatington	4283	1.1	1	1					

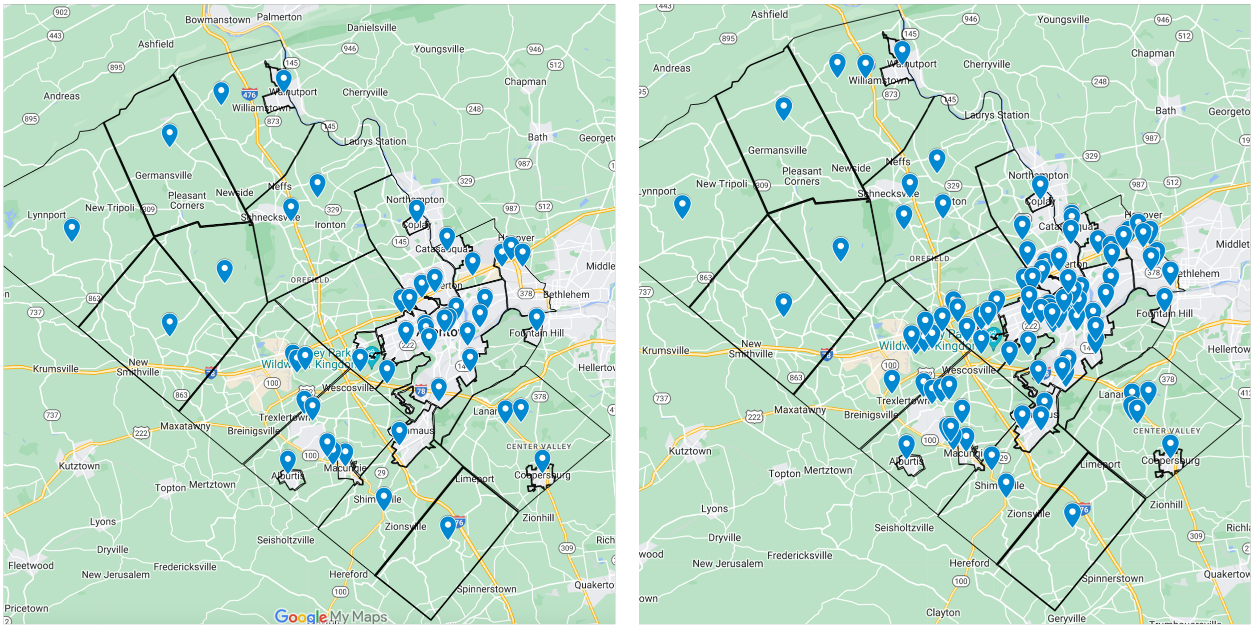

Figure K.1: Location of customers in L-50 (left) and L-100 (right)

Appendix L. Analysis of providers' actual working time

In this section, we analyze the variability in providers' actual working time. For illustrative purposes, we use the L-50 instance from Section 7.4. Table L1 presents the number of hired providers, the average and standard deviation of providers' working time (the number of hired providers, avg, stdv) under different unit overtime cost $\lambda^o \in \{1, 5, 10\}$ and service time ranges. First, we observe that the average actual working time varies under different service time ranges and λ^o. This makes sense because, as we discussed in Section 7.4.2 and shown in Table L1, different numbers of providers are hired under each combination of these parameter settings. For example, the average working time under range (c) is lower because we hire more providers under this range. Second, it is clear that there is a slight variability in working time among providers under each setting. This also makes sense, as the HFASP does not have a criterion to control such variability. Mitigating such variability is worth future investigation.

Table L1: The (number of hired providers, average actual working time in minutes, the standard deviation of actual working time) under different unit overtime cost λ^o and service time ranges for L-50.

	Unit overtime cost λ^o		
Service time range	**1**	**5**	**10**
Base range	$(9, 352, 50)$	$(9, 395, 47)$	$(9, 408, 25)$
Range (a)	$(9, 422, 17)$	$(9, 380, 11)$	$(9, 397, 20)$
Range (b)	$(9, 452, 31)$	$(24, 346, 8)$	$(24, 349, 9)$
Range (c)	$(24, 263, 44)$	$(24, 242, 46)$	$(24, 280, 42)$