An optimization and simulation method for the home service assignment, routing, and scheduling problem with stochastic travel time, service time, and cancellation

Team *The Optimistics*
Daniel Yamín, Daniel Barahona
Advisor: Alfaima L. Solano-Blanco

Centro para la Optimización y Probabilidad Aplicada (COPA), Departamento de Ingeniería
Industrial, Universidad de los Andes, Bogotá, Colombia
E-mail: {`d.yamin`, `ds.barahona`, `al.solano`}@uniandes.edu.co

Abstract

In the Home Service Assignment, Routing, and Appointment scheduling (H-SARA) problem, a set of homogeneous service teams must visit a set of customers. The home service provider needs to decide how many teams to hire (i.e., sizing problem), how to assign service teams to customers (i.e., assignment problem), how to route service teams (i.e., vehicle routing problem), and how to schedule the appointment times for the customers (i.e., appointment scheduling problem) such that the total cost is minimized. To tackle the H-SARA problem, we propose an efficient solution method that comprises two stages. In the first stage, we present a column generation algorithm to solve the sizing, assignment, and routing problem. The algorithm is enhanced by a high-quality initial solution which is found using the route-first cluster-second principle and a polynomial-time 2-approximation algorithm. In the second stage, due to the stochastic nature of travel time, service time, and cancellation, we propose a simulation-driven approach to decide the appointment times such that a desired on-time arrival probability is achieved. To ensure the suitability of the simulation model, we discuss the characterization of the stochastic parameters. The proposed ideas can be embedded in different solution schemes, including a fast heuristic method that finds good solutions within seconds or a more elaborate algorithm to find near-optimal solutions at the expense of longer computational time. At last, we provide a high-level flexible decision support tool implemented in AIMMS.

Keywords: home service, column generation, routing and scheduling problem, stochastic travel and service times, AIMMS.

1   Introduction

Home services provide essential needs such as health care, beauty care, and banking services at the customers' homes. Due to several factors, including population aging, work obligations, and the outspread of chronic and infectious diseases, the demand for home services is expected to increase rapidly in the near future. In 2016, there were about 65,600 regulated, long-term care services providers in the United States which served more than 8.3 million people (Harris-Kojetin et al., 2016). In most European countries, between 1% and 5% of the public budget is assigned to home health care services (Genet et al., 2012). In a broader perspective, the worth of the global home service industry was estimated at $282 billion and is expected to reach $1,133.4 billion by 2026 (Verified Market Research, 2019). In a competitive market, lowering public expenditures, increasing service quality, and decreasing operational costs becomes a critical activity for home services providers (Fikar and Hirsch, 2017). In light of the discussion above, the development of highly efficient computational models to support decision-making in the home service industry has become increasingly important.

Home services require professional service teams to travel between geographically distributed customers. Further, each served customer has an appointment time, referred as to a planned service start time. Therefore, in their day-to-day planning, home service providers need to address the following critical operational decisions: how many teams to hire (i.e., sizing problem), how to assign service teams to customers (i.e., assignment problem), how to route service teams (i.e., vehicle routing problem), and how to schedule the appointment times for the customers (i.e., appointment scheduling problem) such that their total operational cost is minimized. The above problem can be framed as the *Home Service Assignment, Routing, and Appointment scheduling* (H-SARA) problem. Three random factors are considered to capture the stochastic nature of real world: travel time, service duration, and customer cancellation. Regardless of the high complexity that arises from various challenging optimization problems, the planning of home services is often performed manually (Eveborn et al., 2006).

The H-SARA problem –and in a more general sense, the routing and scheduling of home care services– is a major stream of research with different objectives and particular considerations (Fikar and Hirsch, 2017). An overview of logistics management problems in the field of home services can be found in Gutiérrez and Vidal (2013). From the operations research (OR) perspective, Milburn (2012) describes tactical and operational planning problems arising in home health care. Also, and considering that the H-SARA problem is closely related to the vehicle routing problem (VRP), Toth and Vigo (2014) present many variants of the VRP and the most efficient solution methods to tackle them. Concretely in the routing and scheduling of home services, most solution methods rely on metaheuristics (Akjiratikarl et al., 2007; Braekers et al., 2016; Mankowska et al., 2014), heuristics and approximation algorithms (Eveborn et al., 2006; Hindle et al., 2009), and few exact optimization techniques (Rasmussen et al., 2012). Despite that real-world operations are subject to uncertainty, most articles consider static information.

More recently, the home service problem with underlying uncertainty has gained the attention of researchers. For instance, Chen et al. (2017) tackle the problem by formulating an integer program with chance constraints to cope with uncertainty in durations. Also, Cappanera and Scutellà (2021) address the consistency and demand uncertainty in the home care planning problem. We refer to Zhan et al. (2021) for a complete literature review on the home service routing and appointment scheduling with stochastic times. We highlight that most solution methods are either very complex or unable to find rapidly high-quality solutions for large-scale, real-world applications.

In this study, we tackle the H-SARA problem by proposing an end-to-end methodology that solves the problem in a reasonable time and considers the stochastic nature of travel time, service

time, and cancellation. The solution scheme comprises a column generation-based heuristic for the sizing, assignment, and routing (SAR) problem and a simulation model for the scheduling problem. To find an initial solution in the column generation scheme, we extend the route-first cluster-second split mechanism proposed by Prins (2004). To better control the simulation, we provide a framework to characterize stochastic travel time, service time, and cancellation. Finally, we embed the solution method into an AIMMS-based user-friendly application that serves as a high-level decision-support tool for home service providers.

2 Problem definition

Mathematically, we can formulate the H-SARA problem as follows. Let $\mathcal{G} = (\mathcal{V}, \mathcal{A})$ be a directed graph in which $\mathcal{V} = \{0, 1, \ldots, n, n+1\}$ is the set of nodes, $\mathcal{N} = \{1, \ldots, n\} \subseteq \mathcal{V}$ is the set customers that needs to be serviced, and $\mathcal{A} = \{(i,j) \mid (i = 0 \wedge j \in \mathcal{N}) \vee (i \in \mathcal{N} \wedge j \in \mathcal{N} \wedge i \neq j) \vee (i \in \mathcal{N} \wedge j = n+1)\}$ is the set of arcs. The depot is represented by two nodes, namely, 0 and $n+1$. Each customer $i \in \mathcal{N}$ has a random service time s_i and may cancel his/her appointment on the day of service. We denote by p_i the probability associated with customer $i \in \mathcal{N}$ canceling its appointment. Also, each arc $(i,j) \in \mathcal{A}$ has a random travel time t_{ij}.

To service customers there is a set $\mathcal{M}$ of homogeneous service teams. We assume that $\mathcal{M}$ is a sufficiently large set since home service providers usually can hire third-party services. Each hired team $k \in \mathcal{M}$ departs from the depot (node 0), services some customers, and returns to the depot (node $n+1$). Moreover, each customer is scheduled a service start time and the hired teams leave the depot at time 0 and must return before time L (*a priori*). Given the stochastic nature of travel time, service time, and cancellations, we consider the following scenarios. First, the service team arrives at the customer's location before the scheduled service time and must wait (i.e., they remain idle until the scheduled start time). Second, the service team arrives at the customer's location after the scheduled start time and therefore the customer waits. Third, a service team needs additional time to finish serving all scheduled customers beyond L and thus incurs in overtime. For the purpose of the present study, we assume that travel times satisfy the triangle inequality. Also, the probability distributions of travel time, service time, and cancellations are known.

There are several costs associated with the day-to-day activities of home service providers. First, a fixed cost c_f is charged for each hired team. Second, c_t and c_o are the costs that the home service provider pays for one unit of travel time and for one unit of overtime, respectively. Finally, c_e and c_d can be seen as the cost that the company pays for being one time unit early and for being one time unit late while visiting a customer, respectively. These costs penalize early and late services.

We define a mathematical formulation for the H-SARA along the lines of Taş et al. (2013) and Toth and Vigo (2014). Since travel and service times are random variables, we use their expected value. Henceforth, t_{ij} denotes the expected travel time along arc $(i,j) \in \mathcal{A}$, whereas s_i denotes the expected service time when visiting customer $i \in \mathcal{N}$. Note that using expected values is convenient by the linearity of expectation. Note too that the expected utility is the most straightforward metric to evaluate decision-making under uncertainty (Li et al., 2016). Alternatively, to add robustness to the optimization model, one could calculate a high percentile, say 95, of the distribution of travel and service times, and use those values rather than the expected value. Regarding variables, for each arc $(i,j) \in \mathcal{A}$ and each team $k \in \mathcal{M}$, let x_{ij}^k be the binary flow variable indicating whether service team k travels directly from node i to node j or not. To define the scheduling of the appointments, for each node $i \in \mathcal{V}$ and each team $k \in \mathcal{M}$, let w_i^k be the expected time at which team k begins service at node i. To incorporate the overtime cost, we define an auxiliary variable Δ^k for each team $k \in \mathcal{M}$. These variables capture the working time beyond time horizon L. For a

vehicle $k \in \mathcal{M}$, given the values of $\boldsymbol{x}^k = \{x_{ij}^k \mid (i,j) \in \mathcal{A}\}$ and $\boldsymbol{w}^k = \{w_i^k \mid i \in \mathcal{N}\}$, we can compute the total expected earliness $E(\boldsymbol{x}^k, \boldsymbol{w}^k)$ and the total expected delay $D(\boldsymbol{x}^k, \boldsymbol{w}^k)$. To capture the expected travel time cost and the fixed hiring cost, for each arc $(i,j) \in \mathcal{A}$ we define

$$c_{ij} = \begin{cases} c_t \, t_{ij} + c_f, & i = 0; \\ c_t \, t_{ij}, & i \in \mathcal{N}. \end{cases}$$

Finally, to model when a service team is idle (not being used), we add a fictitious arc between node 0 and node $n+1$ with a cost of zero. The mathematical program (2.1)–(2.10) describes the H-SARA problem.

$$\min \quad \sum_{k \in \mathcal{M}} \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij}^k + \sum_{k \in \mathcal{M}} c_o \Delta^k + \sum_{k \in \mathcal{M}} c_e E(\boldsymbol{x}^k, \boldsymbol{w}^k) + \sum_{k \in \mathcal{M}} c_d D(\boldsymbol{x}^k, \boldsymbol{w}^k) \tag{2.1}$$

s.t.,

$$\sum_{k \in \mathcal{M}} \sum_{j \in \mathcal{N} \cup \{n+1\}} x_{ij}^k = 1, \qquad \forall i \in \mathcal{N}; \tag{2.2}$$

$$\sum_{j \in \mathcal{N} \cup \{n+1\}} x_{0,j}^k = 1, \qquad \forall k \in \mathcal{M}; \tag{2.3}$$

$$\sum_{i \in \mathcal{N} \cup \{0\}} x_{ij}^k - \sum_{i \in \mathcal{N} \cup \{n+1\}} x_{ji}^k = 0, \qquad \forall k \in \mathcal{M}, j \in \mathcal{N}; \tag{2.4}$$

$$\sum_{i \in \mathcal{N} \cup \{0\}} x_{i,n+1}^k = 1, \qquad \forall k \in \mathcal{M}; \tag{2.5}$$

$$w_i^k + (1 - p_i) \times (s_i + t_{ij}) - w_j^k \le (1 - x_{ij}^k)|\mathcal{M}|, \qquad \forall k \in \mathcal{M}, (i,j) \in \mathcal{A}; \tag{2.6}$$

$$w_{n+1}^k \le L + \Delta^k, \qquad \forall k \in \mathcal{M}; \tag{2.7}$$

$$w_i^k \ge 0, \qquad \forall k \in \mathcal{M}, i \in \mathcal{V}; \tag{2.8}$$

$$\Delta^k \ge 0, \qquad \forall k \in \mathcal{M}; \tag{2.9}$$

$$x_{ij}^k \in \{0, 1\}, \qquad \forall k \in \mathcal{M}, (i,j) \in \mathcal{A}. \tag{2.10}$$

The objective function (2.1) minimizes the total cost, including the hiring cost and the expected travel time, overtime, earliness, and delay costs. Constraints (2.2) impose that each customer is served exactly once. Constraints (2.3)–(2.5) ensure that the route of each service team starts from the origin and ends at the destination (i.e., characterize a multi-commodity flow structure). Constraints (2.6) guarantee the consistency of the time variables. Note that constraints (2.6) consider customers cancellations and appointments are scheduled accordingly (see §5.2 for details). Soft constraints (2.7) ensure that the service teams return to the depot before L, or else they incur in overtime. Finally, constraints (2.8), (2.9), and (2.10) impose the non-negativity and binary requirements.

3 Solution method

The mathematical formulation (2.1)–(2.10) captures the key elements of the H-SARA problem. Given that the problem combines several difficult problems –and the routing problem alone is NP-Hard (Garey and Johnson, 1979)–, we establish that the H-SARA is also a NP-Hard problem,

making it difficult to solve exactly. In this light, we decompose the problem to develop a two-stage solution method. Figure 1 presents an overview of our solution scheme. In the first stage, we use a column generation algorithm for the SAR problem. The algorithm builds from a set covering model and a high-quality initial solution which is found by extending the route-first cluster-second principle to the H-SARA problem. Subsequently, we iteratively solve a subproblem to find new promising routes for the service teams until the optimality conditions are met. In the second stage, we solve the appointment scheduling problem using a simulation-driven approach which is able to ensure the reliability of the solution. To guarantee the suitability of the simulation approach, we discuss the characterization of the stochastic parameters.
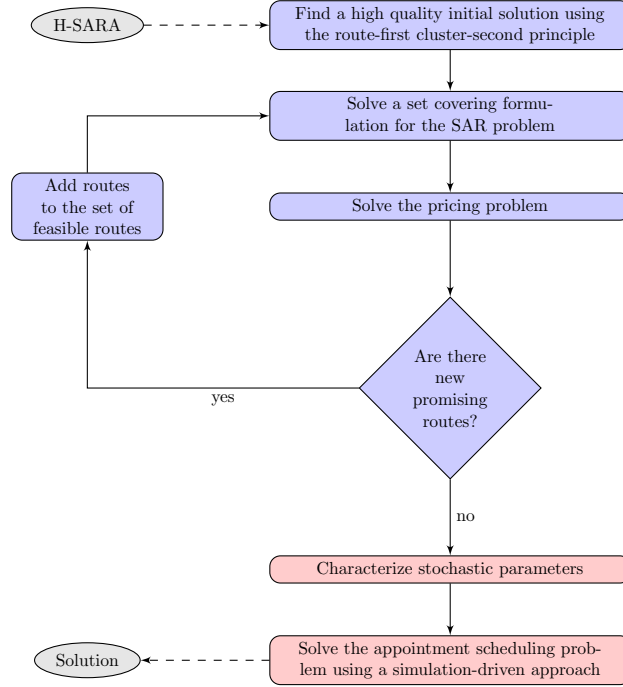
Figure 1: Solution method for the H-SARA problem with the first stage highlighted in blue and the second stage highlighted in red

The remainder of this paper is organized as follows. Section 4 explains the column generation algorithm for the SAR problem. Section 5 presents the simulation-based method for the appointment scheduling problem. Section 6 shows the computational experiments. Finally, Section 7 concludes the paper and outlines future work. For the sake of the readability of the paper, we summarize the notation of the problem statement and solution method in Table 2.

4 Column generation algorithm for the sizing, assignment, and routing problem

In this section, we tackle the sizing, assignment, and routing problem using a column generation algorithm. In particular, with a column generation-based heuristic, we find the number of teams to hire, the assignment of customers to teams, and the routes that each team follows such that each customer is served and the fixed hiring cost, the expected travel time cost, and the expected overtime cost are minimized. We leave the appointment scheduling problem –and its associated costs of earliness and delay– for Section 5.

To build the column generation scheme, we first define a set covering formulation to build a column generation algorithm (§4.1). Subsequently, we propose an efficient method to rapidly find a high-quality initial feasible solution for the problem mentioned above by applying the route-first cluster-second principle (§4.2). Next, we present a mathematical programming-based solution method as well as an heuristic approach to solve the corresponding *pricing problem* (§4.3). Finally, we embed all these ideas into a column generation-based heuristic that solves the SAR problem in practical time for real-world applications (§4.4).

4.1   Set covering formulation and column generation

By inspecting the mathematical formulation of the H-SARA problem, we see that the set of constraints (2.3)–(2.10) as well as the objective function (2.1) are separable for each service team $k \in \mathcal{M}$. By applying Dantzig-Wolfe decomposition, we can derive an equivalent set covering formulation of the problem (Feillet, 2010).

Let Ω be the set of feasible routes satisfying constraints (2.3)–(2.5). Note that each route represents a corresponding service team; that is, routes and service teams are equivalent. For each route $r \in \Omega$, let c_r be the cost of the route (including the hiring, travel time, and overtime costs), a_{ir} be a binary parameter indicating whether the route visits node $i \in \mathcal{N}$ or not, and b_{ijr} be a binary parameter indicating whether the route uses arc $(i,j) \in \mathcal{A}$ or not. Finally, let y_r be a binary variable which indicates if route $r \in \Omega$ is used in the solution or not. The SAR problem can be described by the integer program (4.1)–(4.3).

$$\min \quad \sum_{r \in \Omega} c_r y_r \tag{4.1}$$

$$\text{s.t.,}$$

$$\sum_{r \in \Omega} a_{ir} y_r \geq 1, \qquad \forall i \in \mathcal{N}; \tag{4.2}$$

$$y_r \in \mathbb{Z}_+^1, \qquad \forall r \in \Omega. \tag{4.3}$$

The objective (4.1) minimizes the sum of the hiring cost, the expected travel time cost, and the expected overtime cost. Constraints (4.2) correspond to constraints (2.2) and guarantee that each customer is visited by a route. Finally, constraints (4.3) impose the integer nature of the variables. Unfortunately, the size of the set Ω grows exponentially with the number of nodes, making the set covering model intractable with a standard branch-and-bound algorithm. Thus, we tackle the problem with a column generation scheme.

A lower (i.e., dual) bound on the optimal objective of the set covering model (4.1)–(4.3) can be obtained by dropping the integrality requirements (4.3) and replacing them with non-negative constraints. Then, the resulting linear programming (LP) relaxation is iteratively solved considering only a subset $\Omega' \subseteq \Omega$ of routes. For each covering constraint in (4.2) we have a corresponding dual variable $\pi_i \geq 0$ associated with customer $i \in \mathcal{N}$. Keeping a set partitioning formulation for constraints (4.2) –rather than the set covering formulation we picked– would lead to free dual variables π_i, which typically slows down the convergence of the column generation algorithm (Feillet, 2010). At every iteration, a promising route –that is, one with negative *reduced cost*– can be found by solving a pricing problem as established by Proposition 1.

Proposition 1. For a given route $r \in \Omega$, the pricing problem is defined by the following integer program:

$$\min \quad \sum_{(i,j)\in\mathcal{A}} r_{ij}\, b_{ijr} + c_o \Delta^r \tag{4.4}$$

s.t.,

$$\sum_{j|(i,j)\in\mathcal{A}} b_{ijr} - \sum_{j|(j,i)\in\mathcal{A}} b_{jir} = \begin{cases} 1, & i = 0 \\ -1, & i = n+1 \\ 0, & \text{else} \end{cases}, \qquad \forall i \in \mathcal{V}; \tag{4.5}$$

$$\sum_{j|(i,j)\in\mathcal{A}} b_{ijr} \le 1, \qquad \forall i \in \mathcal{V}; \tag{4.6}$$

$$\sum_{(i,j)\in\mathcal{A}} \tilde{t}_{ij}\, b_{ijr} \le L + \Delta^r; \tag{4.7}$$

$$\Delta^r \ge 0; \tag{4.8}$$

$$b_{ijr} \in \{0,1\}, \qquad \forall (i,j) \in \mathcal{A}, \tag{4.9}$$

where r_{ij} is defined as

$$r_{ij} = \begin{cases} c_f + c_t\, t_{ij}, & i = 0; \\ c_t\, t_{ij} - \pi_i, & i \in \mathcal{N}, \end{cases} \tag{4.10}$$

and $\tilde{t}_{ij}$ is given by

$$\tilde{t}_{ij} = \begin{cases} t_{ij}, & i = 0; \\ t_{ij} + s_i, & i \in \mathcal{N}, \end{cases} \tag{4.11}$$

for ease of notation. Objective function (4.4) minimizes the reduced cost. The *mass balance constraints* (4.5) guarantee that the outflow minus the inflow equals the supply/demand of the node. Constraints (4.6) ensure that the outgoing degree of each node is at most one. Constraints (4.7) impose that the total expected time of the route (including travel times and service times) do not exceed the time limit L, or if it does, then Δ^r captures the overtime. Finally, constraints (4.8) and (4.9) define the nature of the decision variables.

Proof. Each feasible route must satisfy constraints (2.3)–(2.10). Further, the reduced cost of a route r, denoted by $\bar{c}_r$, can be calculated as

$$\bar{c}_r = c_r - \sum_{i\in\mathcal{N}} \pi_i\, a_{ir}$$

$$= c_f + \sum_{(i,j)\in\mathcal{A}} c_t\, t_{ij}\, b_{ijr} + c_o \Delta^r - \sum_{i\in\mathcal{N}} \pi_i\, a_{ir} \tag{4.12}$$

Equality (4.12) holds since we are only considering the hiring cost, the expected travel time cost,

and the expected overtime cost. Since any feasible route starts at 0, we can rewrite $\bar{c}_r$ as

$$
\begin{aligned}
\bar{c}_r &= \sum_{j|(0,j)\in\mathcal{A}} c_f\, b_{ijr} + \sum_{(i,j)\in\mathcal{A}} c_t\, t_{ij}\, b_{ijr} + c_o\Delta^r - \sum_{i\in\mathcal{N}} \pi_i\, a_{ir} \\
&= \sum_{(i,j)\in\mathcal{A}|i=0} (c_f + c_t\, t_{ij})b_{ijr} + \sum_{(i,j)\in\mathcal{A}|i\in\mathcal{N}} (c_t\, t_{ij} - \pi_i)b_{ijr} + c_o\Delta^r
\end{aligned}
$$

The above equation can be summarized by equation (4.4), where the weight for each arc is given by r_{ij} as defined by equation (4.10).

$\square$

The problem defined by objective (4.4) and constraints (4.5)–(4.9) is an *Elementary Shortest Path Problem* with an additional soft time limit constraint.

4.2   High-quality initial solution

To start the column generation algorithm, we need an initial set of routes to derive an initial set of columns. On the one hand, having a high-quality initial solution usually accelerates the convergence of the column generation algorithm. On the other hand, the procedure to find the initial solution should be computationally fast. Therefore, in this study, we adapt the route-first cluster-second principle proposed by Prins (2004) to the H-SARA problem. The route-first cluster-second split procedure for routing problems comprises two steps: (i) constructs a *giant tour* –that is, solve a Traveling Salesman Problem (TSP)–; and (ii) split the giant tour into feasible trips.

We first need to find a Hamiltonian cycle, referring to a tour that visits all nodes in $\mathcal{N}$ exactly once. Unfortunately, finding such cycle is a NP-Complete problem and therefore difficult to solve exactly (Garey and Johnson, 1979). Consequently, we use a polynomial-time 2-approximation algorithm to find such a cycle. Algorithm 1 presents an overview of the approximation algorithm for the TSP (Cormen et al., 2009). Given an undirected graph $\mathcal{G}^t$, the algorithm computes the minimum spanning tree $\mathcal{T}$ from root node r using Prim's polynomial-time algorithm. Then, the resulting tree $\mathcal{T}$ is traversed in preorder and the nodes are listed in $\mathcal{H}$ according to when they are first visited. Finally, the algorithm returns $\mathcal{H}$, which is a Hamiltonian cycle. Proposition 2 establishes that Algorithm 1 has a polynomial running time in the input size and also has an approximation ratio of 2 meaning that the cost of the solution returned is within a factor of 2 of the cost of an optimal tour.

Algorithm 1: Approximation algorithm for the TSP

Input: $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{A}^t)$, undirected graph; r, root node
Output: $\mathcal{H}$, Hamiltonian cycle
1   compute a minimum spanning tree $\mathcal{T}$ for $\mathcal{G}^t$ from root node r using Prim's algorithm
2   define $\mathcal{H}$ as the list of vertices ordered according to when they are first visited in a preorder tree walk of $\mathcal{T}$
    return $\mathcal{H}$

Proposition 2. Algorithm 1 is a polynomial-time 2-approximation algorithm for the TSP with the triangle inequality.
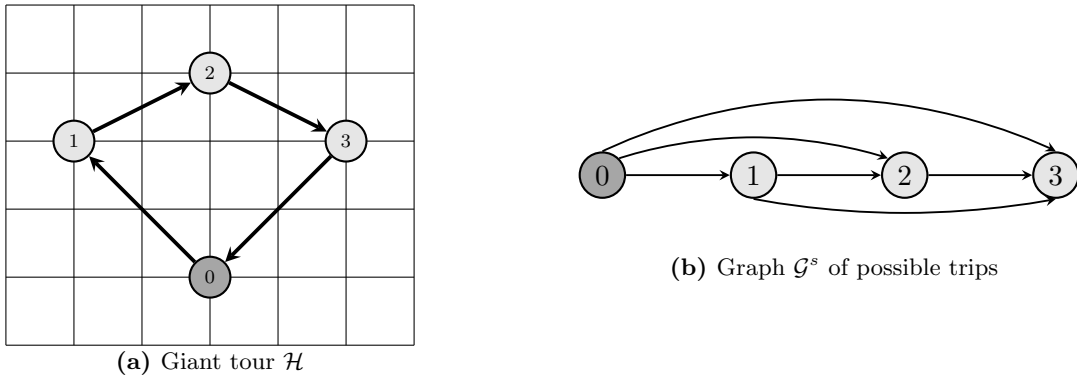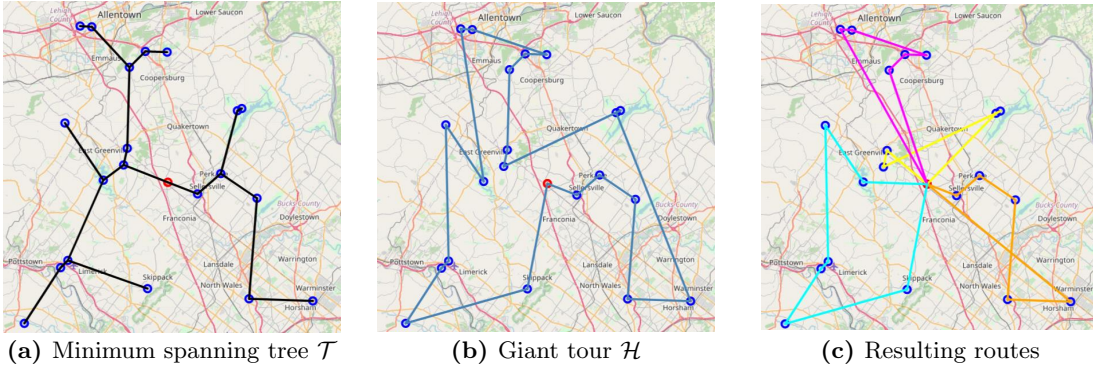
Proof. See Theorem 35.2 in Cormen et al. (2009). $\square$

We make the following two observations. First, the approximation algorithm's input is an *undirected* graph $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{A}^t)$ obtained from the original directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{A})$. Specifically, we define $\mathcal{V}^t = \{0, \dots, n\}$ and $\mathcal{A}^t = \{(i,j) \mid (i,j) \in \mathcal{A} \vee (j,i) \in \mathcal{A}\}$. We also define the cost function

$c' : \mathcal{A}^t \mapsto \mathbb{R}$ satisfying the triangle inequality as $c'_{ij} = c_t \, t_{ij}$ for each edge $(i, j) \in \mathcal{A}^t$. This means that to construct the giant tour, we only consider the expected travel time cost. Second, by using a polynomial-time 2-approximation algorithm, we are rapidly finding a Hamiltonian cycle whilst ensuring the quality of the solution found.

Given a giant tour $\mathcal{H}$, we split it to find a set of routes that compose an initial feasible solution for the SAR problem. This procedure is referred to as split and uses an auxiliary directed acyclic graph of possible trips, denoted as $\mathcal{G}^s = (\mathcal{V}^s, \mathcal{A}^s)$. To explain this modeling technique and the construction of $\mathcal{G}^s$, Figure 3 presents an example considering an illustrative giant tour $\mathcal{H} = \langle 0, 1, 2, 3 \rangle$. Each arc $(i, j) \in \mathcal{A}^s$ represents the route that departs from the depot 0, visits all vertices in the sequence $\mathcal{H}$ which are after node i and until node j, and returns to the depot $n+1$. Therefore, the cost of each arc is the sum of the hiring cost, the expected travel time cost, and the expected overtime cost of the route it represents. Since the graph considers all possible trips that can be obtained following the sequence of the giant tour, by finding the shortest path between 0 and 3 in $\mathcal{G}^s$ –which can be performed efficiently using Bellman-Ford's algorithm– we find a set of optimal routes (subject to the order defined by $\mathcal{H}$). Interestingly, the time complexity of this procedure is polynomial in the input size. Figure 2 shows the approximation algorithm applied to a test instance in which the depot is represented by the red node whereas the dark blue nodes represent customers. Figure 2(a) exhibits the minimum spanning tree (where the black edges constitute the tree $\mathcal{T}$), Figure 2(b) depicts the giant tour $\mathcal{H}$, and Figure 2(c) shows the routes after the split procedure. By extending the route-first cluster-second principle to the H-SARA, we rapidly find a high-quality feasible solution for the SAR problem.

(a) Minimum spanning tree $\mathcal{T}$ (b) Giant tour $\mathcal{H}$ (c) Resulting routes

Figure 2: An illustrative example of the approximation algorithm and Spit procedure

(a) Giant tour $\mathcal{H}$

(b) Graph $\mathcal{G}^s$ of possible trips

Figure 3: Illustrative example of the split procedure

4.3 Solving the pricing problem

We use an integer programming formulation to solve the Elementary Shortest Path Problem. Since the costs $r_{ij} \in \mathbb{R}$ might induce negative cycles, the system of inequalities (4.5)–(4.9) becomes insufficient to guarantee the *elementarity* of the path (Taccari, 2016). Thus, we must incorporate additional constraints to the integer program (4.4)–(4.9) to prevent *subtours*. In particular, we adopt a *cutting-plane* like approach in which we dynamically add *generalized cutset inequalities* (GCS) to the formulation (Taccari, 2016). The GCS are defined as

$$\sum_{(i,j)\in\mathcal{A}(\mathcal{S})} b_{ijr} \leq \sum_{i\in\mathcal{S}\setminus\{u\}} \sum_{j|(i,j)\in\mathcal{A}} b_{ijr}, \qquad \forall u \in \mathcal{S}, \mathcal{S} \subseteq \mathcal{V} \qquad (4.13)$$

where $\mathcal{S}$ denotes a subtour and $\mathcal{A}(\mathcal{S})$ is the set of arcs with both ends in $\mathcal{S}$. Algorithm 2 presents the integer programming-based solution approach to solve the subproblem. We iteratively solve the integer program (4.4)–(4.9) and add the GCS when necessary. Once we find a solution without subtours, we return the corresponding elementary path. We make the following three observations. First, we tested other families of cuts (e.g., the *Dantzig-Fulkerson-Johnson*) and the *Miller-Tucker-Zemlin* model, but the GCS proved to be the most effective. Second, in an intermediate iteration of the **while** loop of Algorithm 2, the solution to the integer program might have an underlying elementary path. If that path has a negative reduced cost, we stop the execution and return it. Third, an integer-programming based solution method probably is less effective than a specialized algorithm, but it can be easily implemented in mathematical optimization software.

Algorithm 2: Integer programming-based method for the pricing problem

---

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, directed graph; r_{ij}, costs.
Output: $\mathcal{P}$, elementary path

1 **while** true **do**
2   solve integer program (4.4)–(4.9)
3   let $\mathcal{F}$ be the set of subtours in the solution found
4   **if** $|\mathcal{F}| = 0$ **then**
5   let $\mathcal{P}$ be the elementary path found by the integer program
6   **break**
7   **else**
         `/* we add the GCS to break subtours                    */`
8   **for** $\mathcal{S} \in \mathcal{F}$ **do**
9   add to the formulation constraints (4.13) for subtour $\mathcal{S}$

10 **return** $\mathcal{P}$

To accelerate the column generation algorithm, we also propose a heuristic method to solve the pricing problem. According to Desaulniers et al. (2002), two of the most prominent acceleration strategies for column generation-based algorithms are: (i) adding a *pool* of columns each iteration; and (ii) solving the pricing problem heuristically (i.e., not to optimality). To that end, we use the route-first cluster-second principle –as in Section 4.2– to solve the Elementary Shortest Path Problem. Specifically, we construct a giant tour $\mathcal{H}$ and a graph of possible trips $\mathcal{G}^s$ as before, except that using the reduced cost r_{ij} to compute the weight of the arcs and considering only the nodes $i \in \mathcal{N}$ such that $\pi_i > 0$. Thus, each time we execute the split, we obtain a set of feasible routes, and we keep those with a negative reduced cost. We remark the following. First, the costs r_{ij} do not necessarily satisfy the triangle inequality, and therefore, the approximation ratio of Algorithm

1 is no longer guaranteed. Second, in the graph of possible trips, the arcs' weight may be negative, but there cannot be negative cycles meaning the shortest path problem can be efficiently solved.

4.4 Column generation-based heuristic

The column generation algorithm described could be embedded in a branch-and-price scheme to find an optimal integer solution for the SAR problem. However, in order to find solutions in a reasonable time for on-time real-world applications, we propose a column generation-based heuristic described in Algorithm 3. The heuristic method finds new, promising routes solving the pricing problem until either the running time exceeds a predefined maximum running time $t_{\max}$ or no paths with negative reduced cost are found. Then, to find an integer solution, we solve the set covering model imposing the integrality conditions with the generated routes. Finally, the heuristic returns Ω^*, the set of routes used in the solution for the SAR problem.

The proposed ideas can be embedded as a building block in different algorithms. For the purpose of the present study, we propose two algorithm configurations: (i) the Exact Method (EM), in which the pricing problem is solved exactly using the mathematical programming-based method; and (ii) the Heuristic Method (HM), in which we use the route-first cluster-second split mechanism to solve the subproblem. Since the EM solves the pricing problem exactly, its solution to the LP relaxation of the set covering model (4.1)–(4.3) is a dual (i.e., lower) bound on the optimal objective and can be used to compute an optimality gap. Regarding the parameter $t_{\max}$, note that it allows decision-makers to balance between the computational time and the quality of the solution. Note too that $t_{\max} = 0$ implies solving the SAR problem with the initial solution.

Algorithm 3: Column generation-based heuristic

Input: $\mathcal{G} = (\mathcal{V}, \mathcal{A})$, directed graph; $t_{\max}$, maximum running time.
Output: Ω^*, solution for the SAR problem.

1 find an initial feasible solution   /* see §4.2 */
2 $\Omega' \leftarrow$ routes corresponding to the initial solution
3 **while** running time $\leq t_{\max}$ **do**
4   solve the LP relaxation of the set covering model (4.1)–(4.3)   /* see §4.1 */
5      $\mathcal{P} \leftarrow$ path found by solving the pricing problem   /* see §4.3 */
6   **if** $\mathcal{P}$ has a negative reduced cost **then**
7          $\Omega' \leftarrow \Omega' \cup \mathcal{P}$
8   **else**
9   **break**

10 let Ω^* be the optimal solution of the set covering model with integrality conditions
11 **return** Ω^*

---

5 Simulation-based method for the appointment scheduling problem

In this section, we tackle the appointment scheduling problem using a simulation-based method that captures the risk profile of decision-makers (§5.2). Before presenting the simulation model, we discuss the characterization of stochastic travel time, service time, and cancellation (§5.1).

5.1 Stochastic travel time, service time, and cancellation

Choosing the distribution of stochastic parameters correctly becomes critical since the results of the simulation model can be heavily determined by the assumptions made about those distributions. Therefore, we discuss the characterization of stochastic travel time, service time, and cancellation. For that purpose, we assume there is historical data from which we can compute the sample mean travel time $\hat{t}_{ij}$ between each pair of locations i and j, the sample mean service time $\hat{s}$, and an estimated probability of cancellation $\hat{p}_i$ for each customer $i \in \mathcal{N}$.

First, we address stochastic travel times. For each arc $(i,j) \in \mathcal{A}$, we have a random variable t_{ij} representing the traveling time from node i to node j. Characterizing travel times on real-world transportation networks has been a major stream of research (e.g., Gómez et al., 2016; Mahmassani et al., 2012; Prakash and Srinivasan, 2017; Sun et al., 2011; Taş et al., 2013). For the purpose of this study, we use a simple yet robust method to obtain the distribution of travel times along the lines of Prakash and Srinivasan (2017). To estimate travel time variance $\hat{v}_{ij}$, we use a linear relation between the sampled mean $\hat{t}_{ij}$ and the standard deviation $\sqrt{\hat{v}_{ij}}$ of travel time. This relation measured in minutes per mile is given by the following equation (Mahmassani et al., 2012)

$$\sqrt{\hat{v}_{ij}} = -0.4736 + 0.9936\,\hat{t}_{ij}$$

Moreover, Prakash and Srinivasan (2017) claim that real-world travel times follow a lognormal distribution. In this light, for each arc $(i,j) \in \mathcal{A}$ we define

$$t_{ij} \sim \text{Lognormal}(\mu_{ij}, \sigma_{ij})$$

Using the method of moments, the relation between the mean $\hat{t}_{ij}$ and variance $\hat{v}_{ij}$ of the travel time and the parameters of the lognormal distribution follows (Prakash, 2018; Yamín et al., 2021)

$$\hat{\mu}_{ij} = \ln\left(\frac{\hat{t}_{ij}^2}{\sqrt{\hat{v}_{ij} + \hat{t}_{ij}^2}}\right) \quad \text{and} \quad \hat{\sigma}_{ij} = \sqrt{\ln\left(1 + \frac{\hat{v}_{ij}}{\hat{t}_{ij}^2}\right)}$$

Second, we consider stochastic service times. Since the exponential distribution is often used to represent service times (Gómez et al., 2016), for each $i \in \mathcal{N}$, we have that

$$s_i \sim \text{Exponential}(\lambda)$$

Again, we use the method of moments to establish the relationship between the sample mean $\hat{s}$ and the parameter of the exponential distribution λ. We obtain the following equation

$$\hat{\lambda} = \frac{1}{\hat{s}}$$

Lastly, to capture the customers' cancellations during the day of service, we use a Bernoulli(γ_i) distribution. Given an estimated (or predicted) probability of cancellation $\hat{p}_i$ of customer i, by the method of moments, we have that $\hat{\gamma}_i = \hat{p}_i$.

Using distributions that correctly model the stochastic parameters enhances our solution scheme as it becomes more robust and reliable. Nevertheless, the proposed ideas are not limited to these distributions and can be flexibly adapted to other families. To that end, Appendix B presents an overview of the method of moments in order to estimate the parameters of a given probability

(a) Maximizing on-time arrival   **(b)** Minimizing teams' idle time
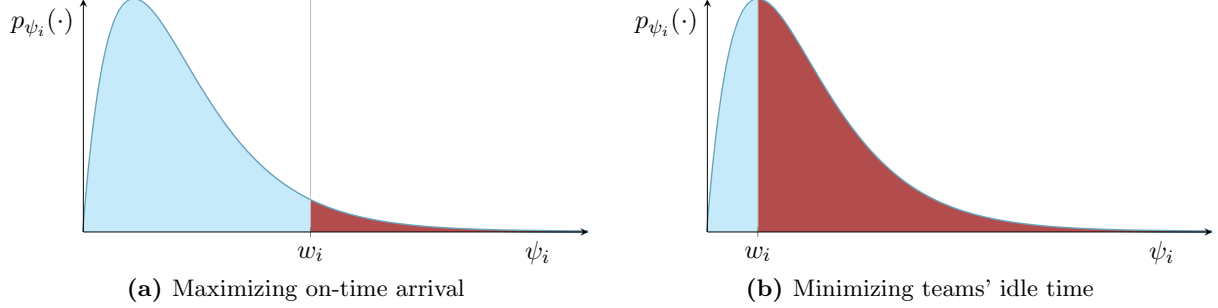
Figure 4: Empirical arrival time distribution $p_{\psi_i}(\cdot)$. The blue area is the probability of on-time arrival. The red area is the probability of late-arrival.

distribution. Also, we remark that one can find a well-suited distribution to a parameter by performing a goodness-of-fit test on the historical data set.

5.2 Simulation model

To solve the H-SARA problem, we still need to schedule a visiting time for each customer; that is, we need to solve an appointment scheduling problem. Since we have previously solved the SAR problem with the column generation-based heuristic (see §4.4), we already have the set of optimal routes Ω^*. For each route $r \in \Omega^*$, we must establish the visiting time w_i for each customer i visited by that route. Therefore, we need to solve the appointment scheduling problem for each route independently. For the sake of clarity, we explain the simulation model for a given route r.

Let the random variable ψ_i denote the arrival time to customer i when visited by route r and let $p_{\psi_i}(\cdot)$ be its probability distribution. Since $p_{\psi_i}(\cdot)$ can be difficult to find theoretically, we use a simulation-driven approach. More precisely, we conduct a Monte-Carlo simulation experiment, compute the arrival time to each node in the route, and obtain the *empirical* distribution of ψ_i. We remark that the simulation model does account for the customers' cancellations. In particular, if the result of the Bernoulli trial is a success for a given customer –meaning that the customer cancels its appointment– the service teams go directly to the location of the next customer on the route. Note that, in this way, appointments are scheduled early when the previous customer on the route has a high chance of canceling. Thus, when a cancellation occurs, the vehicles' idle time is minimized.

We use $p_{\psi_i}(\cdot)$ to decide the appointment time for customer $i \in \mathcal{N}$. When doing so, we need to consider the costs for one time unit of earliness c_e and one time unit of delay c_d. However, c_e and c_d can be difficult to estimate in real-world applications. Also, due to the high variance in urban transportation systems, it might be more relevant to consider a reliability measure rather than a cost. In this light, we define $\alpha \in [0, 1]$ as the desired probability of on-time arrival (i.e., service team arriving before the scheduled time) to the customers' appointments. Correspondingly, $(1 - \alpha)$ is the late-arrival probability referring to the probability of the service team arriving after the scheduled time. Given a value of α –which is specified by the decision-maker– we simply set w_i to the α-percentile of the arrival time distribution $p_{\psi_i}(\cdot)$. On the one hand, if α is close to one –meaning the decision-maker is interested in maximizing the on-time arrival to the appointments– the value of w_i would lie on the *right tail* of ψ_i, as Figure 4(a) shows. On the other hand, if α is close to zero –meaning the decision-maker is interested in minimizing teams' idle time– the value of w_i would be lie on the *left tail* of ψ_i, as Figure 4(b) exhibits. Using a simulation-driven approach, our solution method captures decision-makers' risk profiles and ensures a reliable solution.

12

6 Computational experiments

In this section, we execute computational experiments to evaluate the performance of the proposed method for the H-SARA problem. In Section 6.1, we investigate the trade-off between computational time and quality of the solution. In Section 6.2, we evaluate the impact of the cost parameters and the decision-makers' risk profiles on the solution. Finally, Section 6.3 presents a high-level decision support tool implemented in AIMMS. We coded all programs in Python and compiled them on a computer with an Intel Core i7-8665U CPU @ 2.11GHz with 16GB of memory. Also, we used Gurobi as mathematical optimization engine and *NetworkX* for the graph algorithms.

6.1 Performance of the proposed method

In this set of experiments, we investigate the trade-off between the computational time and the quality of the solution provided by the column generation algorithm for the SAR problem. To do so, we compare two configurations of the algorithm: the Exact Method (EM) and the Heuristic Method (HM). To compare the gains of solving the pricing problem exactly and heuristically, we use performance metrics such as the speedup and the optimality gap. We also evaluate the quality of the Initial Solution (IS).

We generate the test instances as follows. We assume that the depot is at the origin node, and the customers are distributed uniformly in a square of edge 50 km with the origin at the center. The mean value of travel time between two locations is equal to the Euclidean distance since the mean travel speed is assumed to be 1 km/min. The mean customer service times are randomly generated from a uniform distribution in the interval $[30, 60]$. Furthermore, we set $L = 250$, $c_f = 100$, $c_t = 1$, and $c_o = 2$. For the simulation model, we consider 100 replicas and a probability of cancellation equal to 0.1 for each customer, and we set the value of α to 0.5. We execute each configuration of the proposed method in 10 randomly generated instances varying the number of customers $n = 10, 20, 30, 40, 50, 200, 500$, leading to a total of 100 runs (we did not execute the EM in test cases with more than 30 customers since the computational time exceeded three hours). Lastly, we set $t_{\max} = \infty$ which means that the route-finding procedure stops once no paths with negative reduced cost are found. To show the scalability of the proposed ideas, we execute the method with 1,000 and 3,000 customers, and solved the SAR problem with the IS. For this last experiment, we consider the costumers to be in a square of edge 400 km with the depot at the center.

Table 1 summarizes the main results. Column 1 shows the number of customers. Columns 2 through 7 present the average objective (Avg. `Obj`) and running time in seconds (Avg. `CPU`) of the IS, the HM, and the EM, respectively. Column 8 reports the average `speedup` achieved by the HM against the EM. Finally, columns 9 and 10 exhibit the average `gap` –calculated as in equation (6.1)– between the objective found by the IS and the HM with respect to the objective found by the EM. For the EM, we report the Avg. `Obj` of the LP relaxation of the set covering model (since it is a lower bound on the optimal objective) whereas, for the HM, we report the Avg. `Obj` of the integer set covering problem solved with the generated routes.

$$\texttt{gap}(z, z') = \frac{(z - z')}{z'} \times 100 \tag{6.1}$$

From Table 1 we highlight the following takeaways. First, the initial solution is found in less than a second (except for the instances with 500 customers) and is always within 12% of the optimal objective. Second, we observe that the HM can be 33,000 times faster than the EM. Interestingly, the running time of the HM seems to increase linearly with the number of customers, indicating

| n	IS		HM		EM		Avg. speedup	Avg. gap (%)	
Customers	Avg. Obj	Avg. CPU	Avg. Obj	Avg. CPU	Avg. Obj	Avg. CPU	EM/HM	(IS,EM)	(HM,EM)
10	511.32	0.00	505.32	0.03	467.93	4.91	176.03	9.27	8.05
20	908.94	0.00	867.86	0.08	811.67	307.67	4,282.48	11.92	6.81
30	1,306.51	0.01	1,283.05	0.29	1,174.36	7,441.67	33,304.38	11.25	9.26
40	1,647.48	0.01	1,633.26	0.33	-	-	-	-	-
50	1,999.88	0.04	1,982.81	0.66	-	-	-	-	-
200	7,234.94	0.66	7,209.98	35.23	-	-	-	-	-
500	17,456.96	9.18	17,377.76	1,709.62	-	-	-	-	-
1,000	123,303.80	111.34	-	-	-	-	-	-	-
3,000	324,215.30	3,001.84	-	-	-	-	-	-	-

Table 1: Performance of the proposed method

its capacity to scale and solve large-scale instances efficiently. Moreover, the optimality gap of the solution found by the HM ranges from 8.05% to 9.26%, which is likely to be sufficient for real-world applications. Third, we see that the column generation procedure reduces the optimality gap with respect to the initial solution by approximately 1% to 5%. We note that the running time of the simulation model was, on average, just 10% of the total running time, meaning it is an inexpensive procedure. We conclude that the HM finds good solutions in a reasonable time, even for large-scale, real-world instances. However, the EM provides better solutions for long-term planning applications at the expense of a heavy computational burden. Finally, regarding the instances with 1,000 and 3,000 customers, we see that the method is able to solve very large test cases using the IS.

6.2 Sensitivity analysis on the costs and the decision-makers profiles

In this set of experiments, we perform a sensitivity analysis on the costs and the decision-makers profiles. In particular, we analyze the changes in the solution and the performance of the algorithm when varying the hiring cost, the unit travel time cost, the unit overtime cost, and the desired probability of on-time arrival. We generate the instances as described in Section 6.1 but considering only 20, 30, and 40 customers and executing only the HM.

First, we investigate the impact of changing the cost-related parameters. To do so, we use three configurations: A (high hiring cost), B (high overtime cost), and C (high travel time cost). We execute the HM on each configuration 15 times, leading to a total of 135 runs. Figure 5(a) and Figure 5(b) present, respectively, the box-plots with the number of routes used and the running time for each configuration varying the number of customers.

(a) Number of routes

(b) Running time
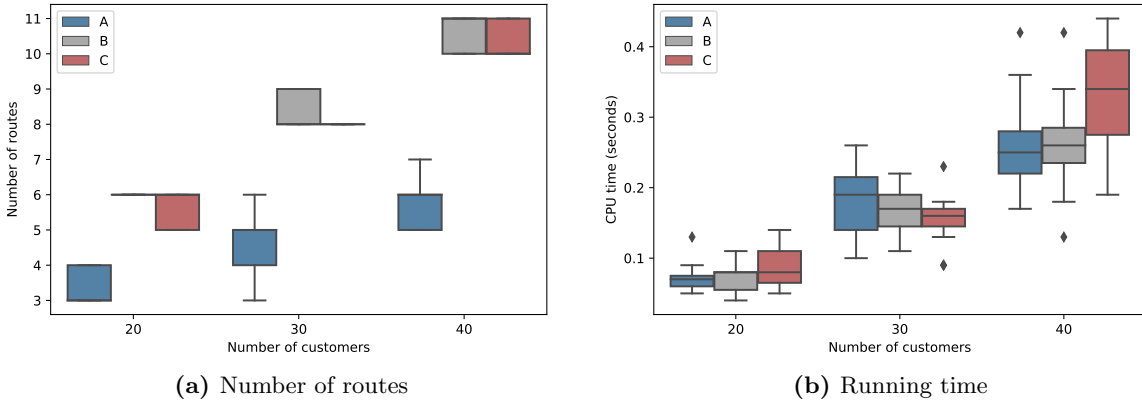
Figure 5: Sensitivity analysis on the costs

14

From Figure 5(a), we observe that, as expected, configuration A uses fewer routes than configurations B and C because of the high fixed cost of hiring a team. We also see that the number of routes does not vary much for a fixed configuration and number of customers. Regarding Figure 5(b), we note that there is not a common trend to conclude that the changes in the cost-related parameters influence the running time of the method. Furthermore, these results confirm those discussed in Section 6.1 since the running time of the HM seems to scale linearly with the number of customers (regardless of the configuration).

Second, we assess the effect of varying the desired probability of on-time arrival. For that purpose, we use three configurations: X (maximizing on-time arrival), Y (neutral), and Z (minimizing teams' idle time). Again, we execute the HM on each configuration 15 times, leading to a total of 135 runs. Figure 6(a) shows the average difference in the scheduled return time to the depot among the three configurations. We refer to the scheduled return time to the depot as the sum of the latest appointment time in the route, the mean service time (of attending that last customer), and the expected travel time from that last customer to the depot. Figure 6(b) presents the running time for each configuration varying the number of customers.

(a) Difference in the return time to the depot   **(b)** Running time
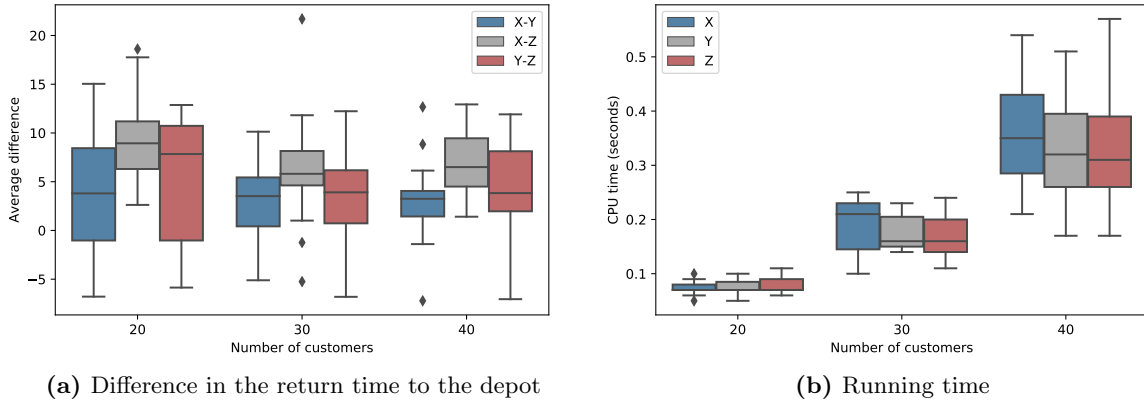
Figure 6: Sensitivity analysis on the decision-makers profiles

From Figure 6(a), we see that the average differences in the scheduled return time to the depot between each pair of configurations is frequently greater than zero. This is an expected result considering that configuration Z schedules appointments early whereas configuration X schedules appointments late; therefore, Z's scheduled return should be prior to that of X (with that of Y in between). Nevertheless, for each configuration, we conducted an independent Monte-Carlo simulation. For this reason, the difference between configurations is less than zero in some cases. As was foreseeable, the largest average difference is between configurations X and Z. We remark that the magnitude of the differences between configurations depends on the shape of the empirical distribution that ultimately depends on the assumptions made. Regarding Figure 6(b), we note that there is no clear pattern, indicating that the running time of the method is not affected by the parameter α, that is, by the decision-maker risk profiles.

6.3 High-level decision support system in AIMMS

We embedded our solution method into a user-friendly high-level decision support system (DSS) developed in AIMMS. This computerized program can be used in decision-making in the home service industry, allowing decision-makers to solve problems timely, improve their efficiency, and perform other critical tasks related to operations, planning, and management. Conveniently, the DSS is flexible and adaptable to accommodate changes in the environment and the user's decision-making.

15

The application can be used by upper- and mid-level managers to analyze multiple outcomes based on optimization models and analytical techniques.

We discuss some of the essential features of the AIMMS-based application. First, as Figure 7 shows, the user interface of the DSS is intuitive, easy to use, and visually attractive to help managers interact with the tool. Second, the DSS provides a real-world geographical visualization of the solution, facilitating planning and operations-related tasks. Third, the information system presents a cost breakdown so that users gain insight into their operational costs. This feature allows decision-makers to decrease costs, a critical activity in a competitive market. Fourth, the DSS details information regarding the cost, travel time, and visiting customers of each route to help automate the managerial processes. Fifth, the application allows users to perform a what-if analysis on the costs, the desired on-time probability, and the number of customers serviced. This analysis reveals new approaches for the company to increase its competitiveness. Sixth, the DSS outlines a schedule for each customer, including its appointment time and route, to increase the control of the operation. Seventh, the users can specify a maximum running time to balance between the computational time and the quality of the solution. We highlight that this feature allows managers to find solutions within seconds for on-time applications or near-optimal solutions (at the expense of longer computational time) for long-term planning applications. Finally, the information can be uploaded to the application from an external file to facilitate the handling of large data sets. In short, the AIMMS-based DSS helps managers make critical decisions based on analytical methods improving the efficiency and competitiveness of their companies.
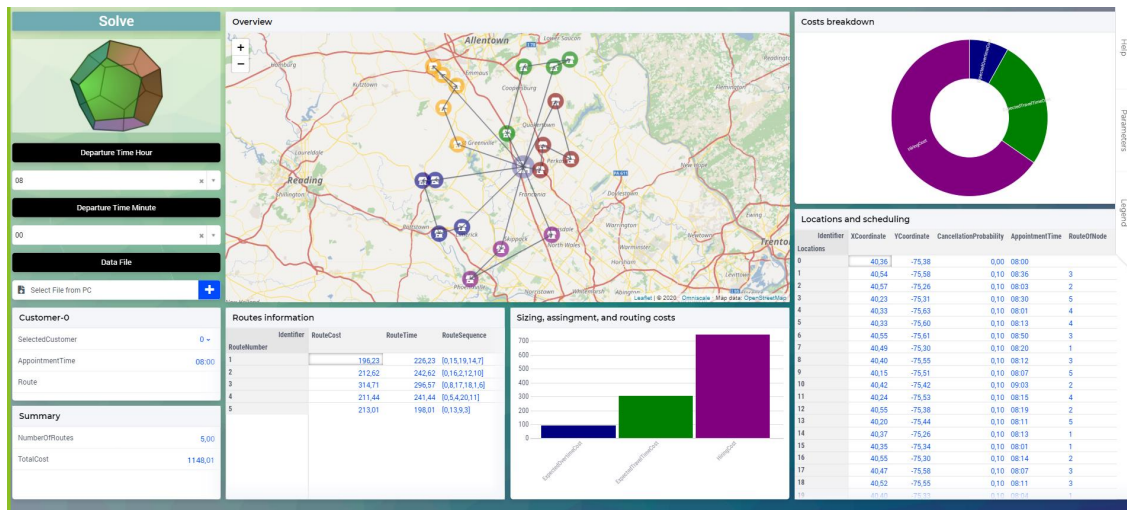

Figure 7: Dashboard of the AIMMS-based decision support system

7 Concluding remarks

In this study, we tackled the *Home Service Assignment, Routing, and Appointment scheduling* (H-SARA) problem problem by proposing a two-stage solution scheme. In the first stage, we used a column generation-based heuristic to solve the sizing, assignment, and routing (SAR) problem. The column generation algorithm is enhanced by a high-quality initial solution found using the route-first cluster-second principle and a polynomial-time 2-approximation algorithm. In the second stage, we proposed a simulation-driven approach to solve the appointment scheduling problem that guarantees reliability in the solution found according to the decision-makers' profile. To ensure

the suitability of the simulation model, we discussed the characterization of the stochastic parameters. The computational study concluded that the Heuristic Method (HM) finds good solutions in a reasonable time for real-world applications whereas, for long-term planning applications, the Exact Method (EM) can provide a better solution at the expense of longer computational running time. Furthermore, a sensitivity analysis showed that changing the parameters does not influence the running time of the method and that the solution scheme can adapt to several settings and profiles. Finally, the AIMMS-based decision support system (DSS) helps managers make critical decisions based on analytical methods improving the efficiency and competitiveness of their companies. Future research could embed the proposed ideas into a branch-and-price algorithm, test the method on real-world instances, or implement a specialized exact algorithm for the pricing problem.

Acknowledgements

References

Akjiratikarl, C., Yenradee, P., & Drake, P. R. (2007). Pso-based algorithm for home care worker scheduling in the uk. *Computers & Industrial Engineering*, *53*(4), 559–583.

Braekers, K., Hartl, R. F., Parragh, S. N., & Tricoire, F. (2016). A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *European Journal of Operational Research*, *248*(2), 428–443.

Cappanera, P., & Scutellà, M. G. (2021). Addressing consistency and demand uncertainty in the home care planning problem. *Flexible Services and Manufacturing Journal*, 1–39.

Chen, C., Rubinstein, Z., Smith, S., & Lau, H. C. (2017). Tackling large-scale home health care delivery problem with uncertainty. *Proceedings of the International Conference on Automated Planning and Scheduling*, *27*(1).

Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to algorithms*. MIT press.

Desaulniers, G., Desrosiers, J., & Solomon, M. M. (2002). *Accelerating strategies in column generation methods for vehicle routing and crew scheduling problems*. Springer.

Eveborn, P., Flisberg, P., & Ronnqvist, M. (2006). Laps care: An operational system for staff planning of home care. *European Journal of Operational Research*, *171*(3), 962–976.

Feillet, D. (2010). A tutorial on column generation and branch-and-price for vehicle routing problems. *4or*, *8*(4), 407–424.

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

Garey, M. R., & Johnson, D. S. (1979). Computers and intractability. *A guide to the theory of NP-completeness*.

Genet, N., Boerma, W., Kroneman, M., Hutchinson, A., Saltman, R. B., Organization, W. H., et al. (2012). *Home care across europe: Current structure and future challenges*. World Health Organization. Regional Office for Europe.

Gómez, A., Mariño, R., Akhavan-Tabatabaei, R., Medaglia, A. L., & Mendoza, J. E. (2016). On modeling stochastic travel and service times in vehicle routing. *Transportation Science*, *50*(2), 627–641.

Gutiérrez, E. V., & Vidal, C. J. (2013). Home health care logistics management: Framework and research perspectives. *International Journal of Industrial Engineering and Management*, *4*(3), 173–182.

Harris-Kojetin, L., Sengupta, M., Park-Lee, E., Valverde, R., Caffrey, C., Rome, V., & Lendon, J. (2016). Long-term care providers and services users in the united states: Data from the national study of long-term care providers, 2013-2014. *Vital & Health Statistics*, (38), x–xii.

Hindle, T., Hindle, G., & Spollen, M. (2009). Travel-related costs of population dispersion in the provision of domiciliary care to the elderly: A case study in english local authorities. *Health Services Management Research*, *22*(1), 27–32.

Li, D., Weng, P., & Karabasoglu, O. (2016). Finding risk-averse shortest path with time-dependent stochastic costs. *International Workshop on Multi-disciplinary Trends in Artificial Intelligence*, 99–111.

Mahmassani, H. S., Hou, T., & Dong, J. (2012). Characterizing travel time variability in vehicular traffic networks: deriving a robust relation for reliability analysis. *Transportation Research Record*, *2315*(1), 141–152.

Mankowska, D. S., Meisel, F., & Bierwirth, C. (2014). The home health care routing and scheduling problem with interdependent services. *Health Care Management Science*, *17*(1), 15–30.

Milburn, A. B. (2012). Operations research applications in home healthcare. *Handbook of healthcare system scheduling* (pp. 281–302). Springer.

Prakash, A. A. (2018). Pruning algorithm for the least expected travel time path on stochastic and time-dependent networks. *Transportation Research Part B: Methodological*, *108*, 127–147.

Prakash, A. A., & Srinivasan, K. K. (2017). Finding the most reliable strategy on stochastic and time-dependent transportation networks: A hypergraph based formulation. *Networks and Spatial Economics*, *17*(3), 809–840.

Prins, C. (2004). A simple and effective evolutionary algorithm for the vehicle routing problem. *Computers & Operations Research*, *31*(12), 1985–2002.

Rasmussen, M. S., Justesen, T., Dohn, A., & Larsen, J. (2012). The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies. *European Journal of Operational Research*, *219*(3), 598–610.

Sun, L., Gu, W., & Mahmassani, H. (2011). Estimation of expected travel time using the method of moment. *Canadian Journal of Civil Engineering*, *38*(2), 154–165.

Taccari, L. (2016). Integer programming formulations for the elementary shortest path problem. *European Journal of Operational Research*, *252*(1), 122–130.

Taş, D., Dellaert, N., Van Woensel, T., & De Kok, T. (2013). Vehicle routing problem with stochastic travel times including soft time windows and service costs. *Computers & Operations Research*, *40*(1), 214–224.

Toth, P., & Vigo, D. (2014). *Vehicle routing: Problems, methods, and applications*. SIAM.

Verified Market Research. (2019). *Global home services market by deployment, by geographic scope and forecast to 2026*. https://www.verifiedmarketresearch.com/product/global-home-service-market-size-and-forecast-to-2025/

Yamín, D., Medaglia, A. L., & Prakash, A. A. (2021). Exact bidirectional algorithm for the least expected travel-time path problem on stochastic and time-dependent networks. *Under review: Computers & Operations Research*.

Zhan, Y., Wang, Z., & Wan, G. (2021). Home service routing and appointment scheduling with stochastic service times. *European Journal of Operational Research*, *288*(1), 98–110.

A   Notation table

Notation	Description
	Problem definition
$\mathcal{G} = (\mathcal{V}, \mathcal{A})$	directed graph in which $\mathcal{V}$ is the set of nodes, $\mathcal{N} \subseteq \mathcal{V}$ is the set customers, and $\mathcal{A}$ is the set of arcs.
s_i	service time of customer $i \in \mathcal{N}$.
p_i	probability associated with customer $i \in \mathcal{N}$ canceling its appointment.
t_{ij}	travel time between $i \in \mathcal{V}$ and $j \in \mathcal{V}$.
$\mathcal{M}$	set of homogeneous service teams.
L	time limit.
c_f	fixed hiring cost.
c_t	cost for one unit of travel time.
c_o	cost for unit of overtime.
c_e	cost for one unit of earliness.
c_d	cost for one unit of delay.
x_{ij}^k	binary flow variable of vehicle $k \in \mathcal{M}$ along arc $(i,j) \in \mathcal{A}$.
w_i^k	expected start time of service of vehicle $k \in \mathcal{M}$ at node $i \in \mathcal{V}$.
Δ^k	auxiliary overtime variable of vehicle $k \in \mathcal{M}$.
	Solution method
Ω	set of feasible routes satisfying constraints (2.3)–(2.5).
c_r	cost of route $r \in \Omega$ including the hiring, travel time, and overtime costs.
a_{ir}	binary parameter indicating whether the route visits node $i \in \mathcal{N}$ or not.
b_{ir}	binary parameter indicating whether the route uses arc $(i,j) \in \mathcal{A}$ or not.
y_r	binary variable indicating if route $r \in \Omega$ is used in the solution or not.
π_i	dual variable of the covering constraint (4.2).
r_{ij}	reduced cost. Refer to equation (4.10).
$\tilde{t}_{ij}$	modified travel time. Refer to equation (4.11).
$\mathcal{G}^t = (\mathcal{V}^t, \mathcal{A}^t)$	undirected graph to compute the minimum spanning tree.
$\mathcal{H}$	Hamiltonian cycle (giant tour).
$\mathcal{G}^s = (\mathcal{V}^s, \mathcal{A}^s)$	directed acyclic graph of possible trips.
$t_{\max}$	maximum running time.
$\hat{t}_{ij}, \hat{v}_{ij}$	sampled mean and variance of travel time.
μ_{ij}, σ_{ij}	parameters of the lognormal distribution.
$\hat{s}$	sampled mean service time.
λ	parameter of the exponential distribution.
$\hat{p}_i$	estimated probability of cancellation of customer $i \in \mathcal{N}$.
γ_i	parameter of the Bernoulli distribution.
ψ_i	random arrival time to customer $i \in \mathcal{N}$ when visited by route $r \in \Omega$.
$p_{\psi_i}(\cdot)$	probability distribution of ψ_i.
α	desired probability of on-time arrival.

Table 2: Notation table

B   Method of moments

In this section, we explain the procedure to use the method of moments in order to estimate the unknown parameters $\theta_1, \theta_2, \ldots, \theta_k$ that characterize the probability distribution $f_X(x; \theta)$ of random variable X. Suppose that the first k moments of the distribution can be expressed as

$$\mu_1 \equiv E[X] = g_1(\theta_1, \theta_2, \ldots, \theta_k)$$
$$\mu_2 \equiv E[X^2] = g_2(\theta_1, \theta_2, \ldots, \theta_k)$$
$$\vdots$$
$$\mu_k \equiv E[X^k] = g_k(\theta_1, \theta_2, \ldots, \theta_k)$$

Then, using a sample of size n, we obtain the values $x_1, \ldots, x_n$. For $j = 1, \ldots, k$, we estimate μ_j using the j-th sample moment given by

$$\hat{\mu}_j = \frac{1}{n} \sum_{i=1}^{n} x_i^j$$

Therefore, the estimators found by the method of moments –denoted as $\hat{\theta}_1, \ldots, \hat{\theta}_k$– are defined by the solution (provided it exists) of the following system of equations

$$\mu_1 = g_1(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k)$$
$$\mu_2 = g_2(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k)$$
$$\vdots$$
$$\mu_k = g_k(\hat{\theta}_1, \hat{\theta}_2, \ldots, \hat{\theta}_k).$$