A Matheuristic Approach for the Home Care Scheduling Problem With Chargeable Overtime and Preference Matching

Gong, Xuran; Geng, Na; Zhu, Yanran; Matta, Andrea; Lanzarone, Ettore

This is a post-peer-review, pre-copyedit version of an article published in IEEE TRANSACTIONS ON AUTOMATION SCIENCE AND ENGINEERING. The final authenticated version is available online at: http://dx.doi.org/10.1109/TASE.2020.3026484

A Matheuristic Approach for the Home Care Scheduling Problem with Chargeable Overtime and Preference Matching

Xuran Gong, Na Geng, *Member, IEEE*, Yanran Zhu, Andrea Matta, *Member, IEEE*, Ettore Lanzarone

***Abstract*—Home Care (HC) services represent an effective solution to face the health issues related to population aging. However, several scheduling problems arise in HC and the providers must make several scheduling and routing decisions, e.g. the assignment of caregivers to clients, in order to balance operating costs and client satisfaction. Starting from the analysis of a real HC provider operating in New York City, NY, USA, this paper addresses a scheduling problem with chargeable overtime and preference matching, and formulates it as an integer programming model. The objective is to minimize a cost function that includes travelling costs, the overtime cost paid by the provider, the preference mismatch and a penalty related to the continuity of care violation. To solve this problem, we design a matheuristic algorithm which integrates a specific variable neighborhood search with a set covering model. The results demonstrate the applicability and efficiency of our approach in solving real-size instances. Sensitivity analyses are also performed to discuss practical insights.**

***Note to Practitioners*—This paper provides a decision support tool to HC managers, which appropriately assigns caregivers to clients and makes routing decision over a long horizon. Chargeable overtimes and preference matching enclosed in this tool are rarely considered in the literature, despite matching is relevant in HC caregiver-to-client assignments and chargeable overtime has a potential in tailoring the service level based on the specific client. We formulate the scheduling problem as a mathematical model. Then, we propose a matheuristic algorithm to efficiently solve the problem in real-size instances. The results show the applicability and efficiency of our method. Thus, HC managers can exploit it to efficiently make assignment and routing decisions, and to analyze the impact on other operating costs when adjusting any of them.**

***Index Terms*—Home Care Assignment and Scheduling, Vehicle Routing Problem, Variable Neighborhood Search, Set Covering Model.**

I. Introduction

WITH the development of society and the rising of life expectancy, aging has become one of the most significant population trends all over the world. Currently, over 11% of the world population are aged 60 and over, and the number is estimated to rise to 22% by 2050 [1]. Along this trend, Home Care (HC) has become an increasingly important part of health care systems in many countries. In the US, about 3.3 million people received HC services from more than 11400 agencies in 2009, and Medicare expenditure on HC is more than 19 billion dollars [2]. In this light, the HC Scheduling Problem (HCSP) attracts more and more attention of researchers and practitioners. Generally, HCSP addresses the optimization of assignment of caregivers to customers and construction of their routes among customers in different locations over a day or a longer horizon. The definition of HCSP resembles vehicle routing problem (VRP) while some specific medical requirements make HCSP different. For example, continuity of care is required to assign the same caregiver to the same customer in HCSP, and preference of care should be respected to improve the quality of care. These requirements make HCSP more complex. It is challenging for the HC service providers to solve the HCSP to reduce the cost and improve the patients' satisfaction.

This paper is motivated by a real HC provider operating in New York City, NY, USA, which provides HC services to customers in different locations. Based on this specific case, in this paper we address a particular HCSP setting with several specific features. The main feature consists of the so-called chargeable overtime, i.e., a part of the overtime costs is sustained by the clients who are willing to pay a higher fee to cover the additional overtime costs incurred because extra time worded with the same client to preserve his/her continuity of care. The feature correlates two significant operational objectives in HCSP, i.e. overtime cost and continuity of care. On the one hand, these clients are beneficial for the provider to reduce its overtime cost; on the other hand, the feature puts even more pressure on the provider to meet clients' requirements for continuity of care. Moreover, the preference of clients for being served by a special type of caregiver or a specific caregiver is considered. The objective of the HCSP is to minimize a cost function that includes travelling cost, the overtime cost paid by the provider, the preference mismatch and a penalty related to the continuity of care violation. The first two components aim at reducing the costs sustained by the provider, while the others aim at improving the quality of care. Finally, some strict constraints need to be satisfied, which include available working time periods, rest times and the maximum working

X. Gong and Y. Zhu are with the Department of Industrial Engineering & Management, Shanghai Jiao Tong University, Shanghai, China (e-mail: zhuyanran_95@163.com; gongxuran01@sjtu.edu.cn).

N. Geng is with the Sino-US Global Logistics Institute, Shanghai Jiao Tong University, Shanghai, China (e-mail: gengna@sjtu.edu.cn).

A. Matta is with the Department of Mechanical Engineering, Politecnico di Milano, Milan, Italy (e-mail: andrea.matta@polimi.it). A. Matta is the corresponding author.

E. Lanzarone is with the Institute for Applied Mathematics and Information Technologies (IMATI), National Research Council of Italy (CNR), Milan, Italy (e-mail: ettore.lanzarone@cnr.it).

times of caregivers. These specific and strict constraints make it more difficult to model the problem and narrow the feasible region, making it even difficult to find a feasible solution.

This problem was first formulated in [3], in which it has been formulated as a linear programming model and the problem has been solved exploiting a cluster-based decomposition. Then, in our preliminary conference paper [4], we proposed a simple Variable Neighborhood Search (VNS) algorithm to solve it. In this paper, we extend and consolidate the conference paper. First, we design a matheuristic algorithm that combines VNS with a Set Covering Model (SCM). Both components can perform the optimization in a collaborative manner and provide beneficial information for each other. Furthermore, we revise the previous VNS and put forward some adaptive mechanisms to provide more and better routing input to the SCM. To assess the effectiveness and efficiency of our algorithm, we test it with a real-life case and compare the results with the optimal solution. We also perform sensitivity analyses and provide managerial insights for the practitioners.

The rest of this paper is organized as follows. Section II presents a review on the related literature. Section III describes the problem and formulates the HCSP model, while the solution approach is proposed in Section IV. Section V describes the case study and sensitivity analyses. Finally, Section VI draws the conclusions of the paper.

II. Literature Review

Despite the importance of HC services, there is still not enough work to tackle HC scheduling problems. The difficulty of HCSP is that not only the revenue of caregiver and the regular constraints in VRP, but also the medical constraints and some matching and preference requirements, need to be considered when making scheduling decisions. In these years, more and more researchers and practitioners make efforts to narrow the gap between research and practice and solve real-life HCSP. Eveborn et al [5] consider different types of tasks in HC and for each client, he/she should be visited by the same caregiver if possible. They develop a HC planning system to aid the planners that includes several modules about data, optimization and report. Carello et al [6] consider nurse-to-patient assignment under the constraint of continuity of care and uncertainty of demand. A cardinality-constrained robust assignment model that does not rely on generation of scenarios is proposed. Liu et al [7] consider three types of demand of patients at home: transportation of medical devices between HC company and clients, delivery of drugs from care deliver to patients and delivery of blood samples from clients to the hospital. They intend to determine the visit pattern of each patient in a period and vehicle routing for each day. A Tabu search scheme, in which several neighborhood structures are proposed and infeasible solutions are allowed, is designed to solve the problem. In the following, they also consider stochastic travelling and service time in HC and home health care problem [8] and synchronized visits [9]. Duque et al [10] consider training and skills of caregivers in HC service to meet the preference of both patients and caregivers and minimize the total travelling distance. The problem is formulated as a bi-objective mathematical program, based on a set-partitioning problem formulation. A flexible two-stage solution strategy is designed to efficiently tackle the problem. Qin et al [11] study a multi-period inspector scheduling problem that is a new variant of multi-trip VRP with time windows. The inspectors are permitted not to return the depot in the problem. The objective is to construct routes for inspectors to complete as many workloads as possible. Several local operators are proposed and integrated into a Tabu search framework. An upper bound is also established by constructing a constrained knapsack. Yalçındağ et al [12] propose a data-driven method to estimate the travel times of care givers in the assignment problem when their routes are not available. The method, based on the Kernel regression technique, uses the travel times observed from previous periods to estimate the time necessary for visiting a set of patients located in specific geographical locations. In another paper, Yalçındağ et al [13] jointly consider three characteristics in HC services, namely the assignment of patients to caregivers, the pattern applied by caregivers to visit patients and the routing problem. A two-stage approach framework is proposed to incrementally incorporate decisions into the first stage. Riazi et al [14] address a home health care routing and scheduling problem, in which time windows of caregivers and clients are restricted and the caregivers for each client must be qualified. A hybrid algorithm is proposed to solve the problem, which integrates a heuristic distributed gossip algorithm with a local solver based on column generation. Zhan et al [15] simultaneously consider team assignment, routing and appointment scheduling in HC problem. By considering the random service time, a scenario-based mixed integer program is proposed and a heuristic algorithm is developed based on Tabu search. Gomes et al [16] present a HC problem, in which each patient must be served by exactly one caregiver within a week and between weeks, the caregivers must rotate among patients. In addition, the updating of the plan is triggered by the events of patients' arrival or departure. The problem is formulated as mixed-integer linear program which is solved by decomposing the whole problem by patients' typology and day. Chaieb et al [17] consider the skills of caregivers and time windows and preference of clients. The problem is decomposed into three subproblems which is solved by k-means, Hungarian algorithm, and Tabu search respectively. Grenouilleau et al [18] study a weekly HCSP considering patients' requirements, caregivers' skills, time-dependent travel times and contracted working hours. More recently, Liu et al [19] study a periodic home health care server assignment problem to properly assign servers to customers along the planning horizon; the continuity of care is regarded as a hard constraint. Grenouilleau et al [20] maximize the acceptance of new patients considering visit patterns in HC problem (each patient is assigned to a single provider and visit times must be the same throughout the week) and time windows of patients and caregivers and maximum time for caregivers. Three mathematical formulations are proposed and a hybrid algorithm are developed that combines logic-based Benders decomposition and large neighborhood search. The algorithm iteratively solves the problem using large neighborhood search and then solves Dantzig-Wolfe formulation using the information found by large neighborhood search. Zabinsky et al [21] study how to transport medical specimens from clinics, physician's offices and hospitals to a central laboratory. A vehicle routing and scheduling algorithm is proposed, in which some propositions are proved to efficiently traverse a branch-and-bound tree.

According to Fikar et al [22] and Cissé et al [23], the objectives in multi-period HCSP includes travel cost, wait time, overtime, preferences, number of caregivers, fairness, number of tasks, number of caregivers and continuity of care. The constraints include time windows, skill requirements, regulations, breaks, workload balance and uncertainty, etc. Compared with the existing literature, this paper considers HCSP with multi-depot and long planning horizon characteristic and the caregivers are allowed not to go home at the end of a period. The payment mechanism that clients can choose to pay the overtime to get continuity of care has been rarely considered in the literature (only in our previous works [3] and [4]). Otherwise, we also consider matching requirements that have received little attention in previous HC problems, which are represented as the caregivers' and clients' preference, "Not send" and "Must go" lists by thorough investigation. In Table I, we list the objectives and constraints of the papers regarding HCSP after 2015 and compare them with our problem setting. Note that in our setting clients propose their requirements and usually they need several visits over a week and each requirement must be strictly respected, so multiple pattern is out of our consideration. In addition, considering the synchronization with other types of assistance or some other conditions, the time window constraint is not considered. However, our model and approach can be easily extended to enclose time window constraint. Our problem considers a richer version of HCSP than most of the existing papers and there is no literature tackling with chargeable overtime and caregivers' strict preference of working period.

TABLE I

OBJECTIVE AND CONSTRAINTS OF AVAILABLE HCSP FORMULATIONS AND OUR FORMULATION

Article	Horizon	Objectives	Constraints							
			TW	MM	CO	CC	BK	CP	WC	MP
Riazi et al [14]	Daily	↓ Travel distance	√	√						
Cappanera et al [24]	Weekly	↑ Minimum workload		√		√				√
Grenouilleau et al [20]	Weekly	↑ # patients visited	√	√		√			√	√
Gomes et al [16]	Weekly	↓ Travel time ↓ Change of visiting time ↓ Maximum workload	√			√	√		√	
Yalçındağ et al [13]	Weekly	↓ Maximum workload ↓ Travel distance		√		√			√	
Duque et al [10]	Weekly	↑ Preference match ↓ Travel distance							√	√
Grenouilleau et al [18]	Weekly	↓ Uncovered jobs ↓ Travel time ↓ Continuity of care violation ↓ Working hours penalty ↓ Soft requirement mismatch	√	√					√	
Liu et al [8]	Daily	↓ Travel time ↓ Uncovered jobs	√	√						
Liu et al [19]	Weekly	↓ Maximum workload				√				√
Chaieb et al [17]	Daily	↓ Uncovered jobs ↓ Waiting time ↓ Travel time ↓ Preference mismatch	√	√						
This paper	Weekly	↓ Preference mismatch ↓ Travel time ↓ Overtime cost ↓ Continuity of care violation		√	√	√	√	√	√	

Note: ↓, Minimize; ↑, Maximize; #, Number of; TW, Time Window; MM, Mandatory Match (Skill requirement or clients' strict preference etc.); CO, Chargeable Overtime; CC, Continuity of Care; BK, Break; CP, Caregivers' Strict Working Preference (Work on weekend or at night etc.); WC, Work Time Contracts; MP, Multiple Service Pattern

Two classes of methodology are applied to tackle HCSP or multi-depot periodic VRP (MPVRP), namely exact approaches and approximate methods [23]. Exact approaches for HCSP or MPVRP usually include integer programming [16], [24], [25], branch-and-bound [21], branch-and-price [8], [26], [27], or branch-and-cut [28], [29], [30], [31], [32]. Although these approaches can theoretically obtain the optimal solution, they usually require the models to have specific structures and the running times are rather long. Approximate methods can be divided into two categories: metaheuristic and hybrid algorithms. In metaheuristic algorithms, the frameworks of different algorithms are inherited and specific operators or rules are defined based on problem characteristics. Tabu search [7], [11], [15], [33], adaptive large neighborhood search [34], [35], genetic algorithm [36], modular heuristic algorithm [37], path relinking algorithm [38], constraint programming [39] are frequently-used metaheuristic algorithms. In hybrid algorithms, two or more strategies are integrated; for example, matheuristic algorithms combine exact solution procedures for subproblems of the overall problem together with metaheuristics [22]. Therefore, they inherit the advantages of both strategies, which is also the reason why we use this type of algorithm in our problem. Column generation [14], Benders decomposition [20], set covering-type model [40] are usually integrated in the metaheuristic approach. Metaheuristic algorithms are known for their efficiency; however, there is no guarantee of convergence.

III. Problem Description And Formulation

The complexity of HCSP considered in this paper lies in some real-life constraints that must be respected when making the scheduling plan. Different from traditional VRP, HC providers balance their own revenue and the satisfaction of clients. In this section, we first describe the specific characteristics existing in HCSP, then the integer linear programming (ILP) model formulation is presented for solving the problem.

A. Problem description

The provider has a set of clients to take care of at home and a pool of caregivers to assign jobs. Clients propose a weekly list of requests for assistance with specific starting times and durations. Caregivers can be assigned to clients based on a set of constraints including clients' needs and preferences. In this problem, all the requirements from clients are collected before the scheduling work. Therefore, the demand information is known in advance. A fixed weekly schedule determines which jobs are assigned to which caregiver, as well as each caregiver's travelling route. No temporal dependency exists in this problem, i.e., no team work exists thus every caregiver does the jobs on his/her own. Once the caregivers communicate their availabilities, they are not allowed to reject the requests in the schedule. Note that in our case, where shifts are related to the assistance to the client, the provider has to guarantee that the client is assisted over a very specific time period considering the synchronization with other types of assistance (e.g., another member of the family is taking care of the patient in specific hours or a shift has to start when the caregiver of the previous shift stops working), the starting and finishing times of each job need to be strictly respected. From an optimization respective, a flexible starting time results in better solution in terms of costs and workload balancing. In fact, our model and approach can be easily extended to enclose time window constraints, as shown in Appendix B.

Several relevant characteristics define the specific problem faced by provider:

1) *Travel time*. It is not considered as work time and, thus, it is not paid. We assume the caregivers' home as the initial and final location. The travel time to and from a client is calculated as the driving time in the two directions respectively.

TABLE II
PREFERENCE CRITERIA

Index	Criteria
1	Prefer female caregiver
2	Prefer male caregiver
3	Native language
4	Car holder
5	Driver's license required
6	CNA license required
7	CPR/First Aid certification required
8	HHA certification required

2) *Chargeable overtime*. A reference number of weekly working hours is defined according to regulation, which is equal to 40 hours. Moreover, each caregiver will define a threshold for the maximum working hours per week, which can be lower than 40 hours, or equal to 40 hours plus the extra hours he/she is willing to work. The total number of working hours cannot be greater than each caregiver's threshold. The caregivers are paid for their actual working hours and if caregivers work overtime, i.e., more than 40 hours one week with the US regulation, the hourly payment is higher. However, since some clients value continuity of care, the clients can decide to pay such extra overtime cost by themselves. In reality, the company provides all the information about such additional fee, including the hourly overtime cost, the range of variability for the cost and so on. Then the clients decide whether to pay the overtime cost or not. More specifically, if one caregiver works for more than 40 hours per week with one client and the client is willing to pay for the overtime, the company will have no extra cost for this caregiver. If the client decides to pay for the overtime cost, then the care of continuity can be better maintained; otherwise, the company may assign another caregiver to this client to reduce the total cost. This payment mechanism that clients pay more for better service can also be found in some other services. For example, the price of a flight ticket is higher if a traveler asks for a better place in the aircraft or for more flexible cancellation policies, or the shipping cost of goods purchased online is higher if special delivery conditions are required.

3) *Preference matching*. A questionnaire is designed for clients and caregivers for satisfying clients' personal demands, in which some preference criteria are collected and presented in Table II. All the clients and caregivers need to answer to the survey containing these criteria with answer "YES" or "NO". It is important to notice that the criteria do not give precise indications. For example, if a client answers "NO" to the statement "Prefer female caregiver", this may not mean he/she prefers a male caregiver; if a client answers "NO" to the statement "Require driver's license", this does not mean he/she does not want a caregiver with driver's license. Therefore, all clients need to answer to another list of questions that whether they are interested in each criterion in the questionnaire. So the answers form above two questionnaires are both considered when calculating preference matching and it is calculated using equation (1), where the notations are explained in Table III.

$$\gamma_c^k = \sum_{q \in M} ((\pi_{cq} - \omega_{kq})^2 \lambda_{cq}), \ \ \forall c \in C, k \in K \tag{1}$$

4) *Night shift*. In the real case problem, night shift is not a real working shift as is traditionally considered. In reality, some clients may need assistance when sleeping during the night and some may not, so the provider defines an 8-hour night shift attached to a regular job for the latter. If a job including a night shift is assigned to a caregiver, he/she will spend an 8-hour break during the night at the client's home. Therefore, the 8-hour night shift is considered as rest time instead of working time, and thus the caregiver is not paid. Note that, night shifts do not contribute to the reaching of maximum working hours for caregivers. Caregivers may have different preferences for working at night and, specifically, some part-time caregivers only accept jobs that start after certain time (9 p.m.) at night.

5) *Continuity of care*. Clients are allowed to choose the maximum number of caregivers that is sent to him/her each week. Although full continuity of care with one caregiver per client [23] [41] is widely pursued, it is not usually possible while respecting other constraints. Thus, the clients choose the maximum number of different caregivers that they are willing

to accept per week and the provider tries to keep the number of caregivers lower than or equal to this maximum number.

TABLE III
SETS, PARAMETERS, AND VARIABLES FOR HCSP

Notations for HCSP	
Sets	**Description**
I	Set of jobs
C	Set of clients
K	Set of caregivers
M	Set of preferences
WE	Set of weekend hours in a week
NS	Set of night hours in a week
ON	Set of hours after 9 p.m. each day in a week
Parameters	**Description**
S_k	Maximum working hours of caregiver k
t_i	Starting time of job i
β	Max duration of a break for caregivers not going home
θ_i^c	Binary, 1 if job i is of client c
mg_c^k	Binary, 1 if caregiver k must do a job of client c
ns_i^k	Binary, 1 if job i cannot be allotted to caregiver k
τ	Maximum working hours according to regulation
d_i	Duration of job i
re	The duration of a rest at home for caregivers
δ_{ij}	Travel time from job i to job j
$\widetilde{\delta}_i^k$	Travel time between caregiver k and job i
ε_c	Binary, 1 if client c is willing to pay for overtime
ξ_k	Binary, 1 if k is willing to work on weekends
ν_k	Binary, 1 if k is willing to work at night
φ_k	Binary, 1 if k only accepts job after 9 p.m.
γ_c^k	Preference mismatch of client c and caregiver k
η_c	Max number of caregivers that can be sent to client c
π_c	Vector of preferences for client c
ω_k	Vector of characteristic for caregiver k
λ_c	Vector of interests of client c
Variables	**Description**
x_{ij}^k	Binary, 1 if job j is done after i by caregiver k without going home
y_{ij}^k	Binary, 1 if job j is done after i by caregiver k when going back home between two jobs
z_i^k	Binary, 1 if job i is done by caregiver k
p_c^k	Binary, 1 if caregiver k does at least one job of client c
f_i^k	Binary, 1 if job i is the first job of caregiver k
l_i^k	Binary, 1 if job i is the last job of caregiver k
O_c^k	Overtime of caregiver k worked on client c
σ_k	Total overtime not paid by clients
E_c	Exceeded number of caregivers sent to client c
u, g_c^k	Auxiliary binary variables

6) *"Not send" and "Must go" lists*. It is natural that some clients have personal preference for certain caregivers. Trying to match the preferred caregivers with clients as well as avoid assigning the unwished caregivers will increase clients' satisfaction. The provider creates a "Must go" list and a "Not send" list for each client. If a caregiver appears in the "Not send" list, he/she will not be assigned the job. On the other hand, if a caregiver appears in the "Must go" list, he/she will be assigned at least one job requested by the client per week. It is important for the provider to increase clients' satisfaction.

Note that the requirement of "Not send" and "Must go" is rather stricter than the general preference. The former can seriously affect the quality of HC and therefore must be respected.

The objective of the HCSP is to provide a service that minimize:

1) Preference mismatches between clients and caregivers
2) Travelling distance of caregivers
3) Overtime paid by the provider
4) Penalty for violating continuity of care

B. Problem formulation

We list all the notations in Table III that can be referred to. Multiple objectives are transformed into single objective by applying weight α_i associated to each component. As different components in the objective function may have different magnitude, a normalization is required [42]. The weights are computed as $\alpha_i = a_i\theta_i$, where a_i are the weights assigned by the decision maker and θ_i are the normalization factors calculated as $\theta_i = 1/(e_i^U - e_i^L)$, where e_i^U and e_i^L are the upper and lower bound of each component respectively.

The objective can be formulated as follows:

$$\begin{aligned} \min\ & \alpha_1 \sum_{i,c,k} z_i^k \gamma_c^k d_i \theta_i^c + \\ & \alpha_2 [\sum_{k,i} \widetilde{\delta}_i^k f_i^k + \sum_{k,i} \widetilde{\delta}_i^k l_i^k + \sum_{i,j,k} \delta_{ij} x_{ij}^k + \sum_{i,j,k} (\widetilde{\delta}_i^k + \widetilde{\delta}_j^k) y_{ij}^k] + \\ & \alpha_3 \sum_k \sigma_k + \alpha_4 \sum_c E_c \end{aligned} \tag{2}$$

The first component represents the cost of preference mismatch. The second part is the travelling cost and the total travelling time is obtained by summing up the travelling time between caregivers and clients, travelling time between different clients as well. The third and fourth parts represent overtime cost and penalty for violating continuity of care respectively. In (2), γ_c^k, σ_k and E_c need to be calculated respectively. Parameter γ_c^k is calculated as equation (1).

The other two variables, σ_k and E_c are dependent on decision variables z_i^k; σ_k can be calculated as follows:

$$\begin{aligned} & O_c^k = \max\{0, \sum_{i\in I} z_i^k \cdot d_i \cdot \theta_i^c - \tau\}, \quad \forall c \in C, k \in K \\ & \sigma_k \ge \sum_{i\in I} z_i^k d_i - \sum_{c\in C} O_c^k \varepsilon_c - \tau, \ \forall k \in K \\ & \sigma_k \ge 0, \quad \forall k \in K \end{aligned} \tag{3}$$

The overtime of caregiver k served for c is first calculated and then the willingness of client c to pay for overtime is considered when summing up the overtime of caregiver k. The first equation in (3) can be linearized by:

$$\begin{aligned} & O_c^k \ge 0, \quad \forall c \in C, k \in K \\ & O_c^k \ge \sum_{i\in I} z_i^k \cdot d_i \cdot \theta_i^c - \tau, \quad \forall c \in C, k \in K \\ & O_c^k \le (1 - g_c^k) M, \quad \forall c \in C, k \in K \\ & O_c^k \le \sum_{i\in I} z_i^k \cdot d_i \cdot \theta_i^c - \tau + g_c^k M, \quad \forall c \in C, k \in K \end{aligned}$$

E_c represents the exceeded number of caregivers for client c and can be calculated as shown in (4). Note that in all formulas M represents a sufficient large number.

$$\begin{aligned} & M \cdot p_c^k \geq \sum_{i\in I} \theta_i^c z_i^k, \quad \forall c \in C, k \in K \\ & p_c^k \leq M \cdot \sum_{i\in I} \theta_i^c z_i^k, \quad \forall c \in C, k \in K \\ & \sum_{k\in K} p_c^k \leq \eta_c + E_c, \quad \forall c \in C \\ & E_c \geq 0, \quad \forall c \in C \end{aligned} \tag{4}$$

Other constraints are listed as follows:

$$\sum_{i\in I} d_i \cdot z_i^k \leq S_k, \quad \forall k \in K \tag{5}$$

$$\sum_{k\in K} z_i^k = 1, \quad \forall i \in I \tag{6}$$

$$\begin{aligned} & \sum_{j\in I} (x_{ij}^k + y_{ij}^k) + l_i^k - z_i^k = 0, \quad \forall i \in I, k \in K \\ & \sum_{i\in I} (x_{ij}^k + y_{ij}^k) + f_j^k - z_j^k = 0, \quad \forall j \in I, k \in K \end{aligned} \tag{7}$$

$$\begin{aligned} & -M(1 - x_{ij}^k) \leq t_j - (t_i + d_i + \delta_{ij}), \\ & \qquad \qquad \forall i, j \in I, \forall k \in K \\ & -M(1 - y_{ij}^k) \leq t_j - (t_i + d_i + re + \tilde{\delta}_i^k + \tilde{\delta}_j^k), \\ & \qquad \qquad \forall i, j \in I, \forall k \in K \end{aligned} \tag{8}$$

$$\begin{aligned} & (t_j - (t_i + d_i + \delta_{ij})) - \beta \leq Mu, \quad \forall i, j \in I \\ & \sum_{k\in K} x_{ij}^k \leq M(1-u), \quad \forall i, j \in I \end{aligned} \tag{9}$$

$$\begin{aligned} & \sum_{i\in I} z_i^k \leq M \sum_{i\in I} f_i^k, \quad \forall k \in K \\ & \sum_{i\in I} z_i^k \geq \sum_{i\in I} f_i^k, \quad \forall k \in K \\ & \sum_{i\in I} z_i^k \leq M \sum_{i\in I} l_i^k, \quad \forall k \in K \\ & \sum_{i\in I} z_i^k \geq \sum_{i\in I} l_i^k, \quad \forall k \in K \\ & \sum_{i\in I} f_i^k \leq 1, \quad \forall k \in K \\ & \sum_{i\in I} l_i^k \leq 1, \quad \forall k \in K \end{aligned} \tag{10}$$

$$x_{ij}^k + y_{ij}^k \leq 1, \quad \forall i, j \in I, \forall k \in K \tag{11}$$

$$z_i^k \leq 1 - ns_i^k, \quad \forall i \in I, k \in K \tag{12}$$

$$p_c^k \geq mg_c^k, \quad \forall c \in C, k \in K \tag{13}$$

$$\sum_{i\in I; t_i \in WE} z_i^k \leq M\xi_k, \quad \forall k \in K \tag{14}$$

$$\sum_{i\in I; t_i \in NS} z_i^k \leq M\nu_k, \quad \forall k \in K \tag{15}$$

$$\sum_{i\in I; t_i \notin ON} z_i^k \leq M(1-\varphi_k), \quad \forall k \in K \tag{16}$$

$$z_i^k, x_{ij}^k, y_{ij}^k, p_c^k, f_i^k, l_i^k \in \{0,1\} \tag{17}$$

Constraints (5) ensure that the total working time of caregivers cannot exceed the maximum hours each caregiver has declared. Constraints (6) guarantee that each job is done exactly once. Constraints (7) impose that if a caregiver does a job, this job can either be the last/first one, or the predecessor/successor of another job. In the latter case, the two jobs can be done either one after the other or by going home in between. Constraints (8) guarantee that the starting times of the jobs are set correctly in the cases when two jobs are done one after the other or by going home in between. Constraints (9) guarantee that if two jobs have more than β hours in between, then the caregivers must go home. Constraints (10) ensure that each caregiver starts and ends work at home. Constraints (11) state that either a job is done after another or the caregiver goes home in between. Constraints (12) and (13) deal with "Not send" and "Must go" lists respectively. Constraints (14)-(16) guarantee that, if a caregiver is not available at night or on the weekend of before 9 p.m., he/she is not assigned jobs in this period.

The model is from our previous conference paper [4]. In this paper all the preferences are treated as soft requirements and share identical weights. Constraints (4) are augmented to present the relation of decision variables z_i^k and E_c. Note that there are different ways to formulate the problem, we construct several formulations and choose the best one. Two other alternatives are presented in Appendix A.

IV. Solution Approach

In this section, we propose a matheuristic approach that embeds an integer programming model in the VNS process, which is named *IP-Insert VNS* algorithm and resembles the idea in [40]. The framework of the IP-Insert VNS is presented in Algorithm 1. The solution is in terms of a set of routes, where each route contains all jobs assigned to the corresponding caregiver in the planning horizon.

In the initialization phase, a heuristic mechanism generates a single solution that is saved as the current optimal solution. At each iteration, the current best solution is first diversified by removal and insertion operations, namely r caregivers are randomly selected and the jobs assigned to them are removed and reinserted. The number r of selected caregivers is adaptively adjusted according to the status of the current best solution. Then, the solution is improved in the intensification phase by means of a neighborhood search. During the above two procedures, some routes (also called *columns*) can be generated and recorded in the *column pool*. When the trigger condition of SCM is met, it is solved with CPLEX, otherwise it is not solved at that iteration.

Algorithm 1 IP-Insert VNS Algorithm

Input: Data set of caregivers and clients; Weights for objectives; Parameters in algorithm
Output: Schedule for caregivers; Cost of each objective
1: *Initialization*
2: **While** stopping criteria is not satisfied
3: *Diversification*(r)
4: *Intensification*
5: *SCM solving*
6: Update the current optimal and the number of caregivers adaptively in the process of diversification r
7: **End While**

The details of each step are illustrated below. We describe in Sections IV.A-IV.D the framework and each component of the IP-Insert VNS; then, we describe in Section IV.E a variant of the IP-Insert VNS, which is named *IP-Post VNS*.

A. Generation of initial solution

Since the “Must go” and “Not send” constraints are relatively hard to satisfy, they are considered first for the generation of the initial solution. The clients who require “Must go” caregivers are sorted by descending order of number of “Must go” caregivers. Jobs randomly chosen from these clients are assigned to the required caregiver to firstly satisfy “Must go” constraints. For the remaining jobs, in order to give priority to the jobs that are more difficult to assign, the total number of “Not send” caregivers for each job is calculated. Then, the jobs are sorted by descending order of this number. For each job, all available caregivers are sorted by ascending order of preference mismatches between the client and caregiver. If there are caregivers that have the same preference mismatch, they are then sorted according to the travelling time from the job. Jobs are assigned according to the priority of minimum preference violation and then minimum travelling time. The pseudocode of initialization process is presented in Algorithm 2.

After assigning all jobs to caregivers, an initial solution can be generated. It is important to mention that a feasible solution might not be easy to find for the real case problem. Therefore, infeasible initial solutions are allowed by relaxing “Not send” constraint in line 11 of Algorithm 2, and the infeasible solutions will be improved by diversification process presented in Algorithm 3 and neighborhood 1 in intensification process presented in Algorithm 4, until a feasible solution is found.

Algorithm 2 Initialization

Input: Data set of caregivers and clients.
Output: Initial solution
1: Sort clients by descending order of number of “Must go” caregivers requested
2: **For** $c = 1$ to $|C|$
3: **Repeat**
4: Randomly choose a job from client c and assign to unassigned “Must go” caregiver who is feasible
5: **Until** All “Must go” caregivers are assigned one job from client c
6: **End For**
7: Sort unassigned jobs by descending order of number of “Not send” caregivers
8: **For** $i = 1$ to $|I|$
9: **Repeat**
10: Rank available caregivers by ascending order of preference mismatches and travelling time
11: Assign job i to first feasible caregiver
12: **Until** all jobs are assigned
13: **End For**

B. Diversification process

Diversification is the most important part of the algorithm to avoid falling into local optima. This section proposes a strong shaking procedure [43] to diversify the solution. As is shown in Algorithm 3, firstly r caregivers are randomly selected and all jobs assigned to them are removed. Then these unassigned jobs are randomly selected and assigned to the first available caregiver. Note that if removing a job can incur violation of “must go” constraint, then this removal operation is revoked. Similar to the initialization process, there may exist jobs that cannot be inserted in all routes, in this case the “Not send” constraint is relaxed in line 5 of Algorithm 3.

The effectiveness of the algorithm depends on r: it is harder to get the local optima if r is too large, while a small r is detrimental to the exploration of the feasible region. For SCM, a larger r is helpful for generating diverse routes and brings more opportunities to get better solution, however, a large column pool can decelerate the speed for solving SCM. To balance the effectiveness and efficiency, r is adjusted adaptively in each iteration. The percentage of caregivers participating in the diversification takes values in [*minS*, *maxS*]. In order to generate more diverse routes, SCM is not triggered and r is set to *maxS* in the first *kFC* iterations. After that, r starts from *minS* and increases by 10% if there is no improvement in *iFS* iterations.

Algorithm 3 Diversification(r)

Input: Caregiver and client dataset; current optimal solution
Output: Diversified solution
1: Randomly select r caregivers and remove all jobs within their routes
2: **Repeat**
3: Randomly choose an unassigned job i
4: **For** $k \in K$
5: **If** job i can be assigned to caregiver k
6: Job i is assigned to caregiver k
7: **Break**
8: **End If**
9: **End For**
10: **Until** all jobs are assigned

C. Intensification process

In this subsection, four different neighborhood structures are designed for exploiting the neighborhood of a solution. Based on the related literature, there are multiple intra-route and inter-route neighborhood operators that are commonly used [44] [45]. In our problem, where time windows are fixed, only inter-route neighborhood operators are implemented.

1) Neighborhood 1. Shift one job from a route to another.

2) Neighborhood 2. Swap two jobs in different routes.

3) Neighborhood 3. Swap a segment that contains two or more jobs from a route with another segment from another route.

4) Neighborhood 4. Choose three routes and swap two jobs in different routes.

Algorithm 4 Intensification

Input: Data set of caregivers and clients; A solution after diversification
Output: A local optimal solution
1: **While** True
2: Search in Neighborhood 1
3: **If** “Not send” pairs diminish or the total cost is reduced
4: Update the solution and **Continue**
5: **Else**
6: Search in Neighborhood 2 (Neighborhood 3 and Neighborhood 4 successively)
7: **If** The total cost is reduced
8: Update the solution and **Continue**
9: **Else If** No improvement achieved in all Neighborhoods
10: **Break**
11: **End If**
12: **End If**
13: **End While**

The neighborhood structure from 1 to 4 is increasingly more complicated. The intensification process with the four structures is presented in Algorithm 4. During the process of intensification, all possible shifts or swaps are screened to ensure the local optima can be obtained. Lines 6-10 in Algorithm 4 express the sequence of neighborhoods to use. Only when there is no improvement after Neighborhood 1, Neighborhood 2 is called. If a better solution is obtained after

Neighborhood 2 then it returns to Line 2, otherwise Neighborhood 3 is used. The trigger mechanism of Neighborhood 4 is in the same way. To accelerate the intensification process, only the changes of the objective caused by applying neighborhood search rather than the entire new solution are evaluated. Note that there may be infeasible solutions after initialization or diversification, so Neighborhood 1 is not only designed to search for the local optima but also responsible for repairing these infeasible solutions. The repair procedure is applied in lots of literature works [7] [36] and its effectiveness has been verified. In this work, a sufficiently large reward is given to the shift that can diminish the "Not send" pair. The reward aims at redirecting the search toward feasible solutions.

D. Set covering model

SCM is the most important part of IP-Insert VNS. On the one hand, SCM can screen the current column pool and exploit the information that is missed by VNS. On the other hand, SCM can guide the searching direction of VNS by offering a new current optimal solution. In terms of functionality of the model, there are three key components significantly affecting the effectiveness of SCM, namely the *construction of SCM*, *the generation mechanism of routes* and *the rule to trigger SCM*.

1) Construction of SCM. To construct the SCM, more notations are needed and presented in Table IV, other notations used in SCM is the same as those in HCSP and readers can refer to Table III.

TABLE IV
NOTATIONS FOR SET COVERING MODEL

Sets	**Description**
R_k	Set of routs in a week period for caregiver k
Parameters	**Description**
δ_r	Travel time of route r
f_r	Preference mismatch of route r
σ_r	Overtime paid by company of route r
ρ_k	Binary, 1 if caregiver k has must-go clients
ς_{ir}	Binary, 1 if job i is in route r
d_r	Total working time of route r
Variables	**Description**
z_r	Binary, 1 if route r is selected in the solution
o_c^k	Auxiliary variable

The model is expressed as follows:

$$\text{(SCM)}\ \min \sum_{k\in K}\sum_{r\in R_k}(\alpha_1\delta_r+\alpha_2 f_r+\alpha_3\sigma_r)\cdot z_r + +\alpha_4\sum_{c\in C}E_c \tag{18}$$

Subject to

$$\sum_{k\in K}\sum_{r\in R_k}\varsigma_{ir}\cdot z_r = 1,\quad \forall i\in I \tag{19}$$

$$\rho_k \le \sum_{r\in R_k} z_r \le 1,\quad \forall k\in K \tag{20}$$

$$M\cdot p_c^k \ge \sum_{i\in I}\sum_{r\in R_k}\theta_i^c\cdot\varsigma_{ir}\cdot z_r,\quad \forall c\in C, k\in K \tag{21}$$

$$\sum_{k\in K} p_c^k \le \eta_c + E_c,\quad \forall c\in C \tag{22}$$

$$E_c \ge 0,\quad \forall c\in C \tag{23}$$

$$z_r, p_c^k \in\{0,1\},\quad \forall r\in R_k, k\in K, i\in I, c\in C \tag{24}$$

In the model, objective (18) includes the cost of travelling, preference mismatch, overtime and penalty for exceeding the maximum number of caregivers for clients. Constraints (19) guarantee that every job is done. Constraints (20) ensure that at most one route can be selected for a caregiver k and the caregivers with must-go clients must be selected to satisfy "Must go" constraint. Constraints (21)-(23) calculate the exceeded number of caregivers for each client.

If a feasible schedule (also called a route) for a caregiver is obtained, then the traveling time, service time and preference cost can be obtained correspondingly. For a route $r\in R_k$, the overtime that the company need to pay is calculated as follows and the method of calculating σ_r resembles that in (3).

$$o_c^r = \max\{\sum_{i\in I}\varsigma_{ir}\cdot\theta_i^c\cdot d_i-\tau, 0\},\ \forall c\in C$$

$$\sigma_r \ge d_r - \tau - \sum_{c\in C}\varepsilon_c\cdot o_c^r$$

$$\sigma_r \ge 0$$

2) Generation mechanism of routes. It is important to notice that all the routes in the pool are feasible, namely for a caregiver all "Must go" and "Not send" constraints are satisfied. In addition, all the routes in the pool are different. As mentioned in the diversification section, in the first *kFC* (the number of iterations before triggering SCM) iterations a large percentage of caregivers participates in the process of diversification. Although too much diversification may lead to infeasibility in the early stage of the algorithm, it can generate more diverse routes and is beneficial for the SCM to execute optimization. The computational results in Section V support the statement. On the other hand, the routes can be generated in the process of intensification if a better solution is found with any neighborhood structure. The routes generated in the process of intensification are generally superior to those by diversification, which can provide for SCM more local information. Appropriately arranging the intensification and diversification can balance the global information and local information of routes for SCM.

Algorithm 5 SCM solving

Input: Data set of caregivers and clients; Column pool; The number of iteration k; The number of iterations from last trigger n; The number of iterations in which the optimal is not changed h
Output: The solution of SCM
If No feasible solution is found or $k > kFC$
  If $n = nFC$ or $h = nFC$
    Solving SCM and update the optimal
    $n = 0$;
  End If
End If

3) The rule to trigger SCM. It is inefficient to solve SCM in each iteration because the routes generated in each iteration have some similarities. However, when the column pool is large, it takes long time to solve SCM. Therefore, a parameter *nFC*, which denotes the number of iterations between two SCM process, is proposed to balance the solution efficiency and computational time. In Algorithm 5, if no feasible solution can be found by VNS or the current optimal is not changed for *nFC* iterations, then SCM is triggered to help find a feasible solution. After *kFC* iterations, SCM is triggered when the number of iterations from the last run of SCM is equal to *nFC* or the current optimal is not changed for *nFC* iterations. Note that whenever a better solution is found in the whole IP-Insert VNS process, *h* is set to zero.

E. A post-optimization VNS algorithm

In this subsection, we describe a variant of IP-Insert VNS, i.e., IP-Post VNS, in which the definition of initialization, diversification and intensification process is the same as in the IP-Insert VNS while SCM only triggered at the end of VNS procedure. This framework is firstly proposed by [46] to solve VRP.

V. COMPUTATIONAL EXPERIMENTS

In this section, we test the performance of our algorithms by conducting experiments with the data collected from a HC service provider operating in New York City, USA. We implement three algorithms in C++ and run on a computer equipped with Intel(R) Core(TM) i7@ 2.60GHz and 8GB RAM. The ILP model proposed in Section III and SCM are both exactly solved with CPLEX (Version 12.9) and the number of parallel threads is 12.

A. Experimental setting

The base case data is collected from a HC service provider operating in New York City, USA. When focusing on the operations of the mentioned provider of one certain week in 2019, there are 57 caregivers and 21 clients. The requirements from clients range from 4 hours per week to a 24/7 live in service. The location of all clients and caregivers are shown in Fig.1. Although the number of caregivers is much larger than that of clients, the total number of working hours requested by clients is 1807 hours during that week, while the total number of available working hours of caregivers is only 2063 hours. This is because the maximum working hours requested by many caregivers is much less than the regulated maximum working time. Moreover, some caregivers are part time employees who only accepts jobs which start at night. Therefore, even though the number of caregivers is larger than that of clients, it is not easy to have all jobs from clients properly assigned.

Fig.1. Location of clients and caregivers

Based on the base case data, we first tune the parameters in IP-Insert algorithm. There are five key parameters in IP-Insert algorithm, namely the minimum (*minS*) and maximum (*maxS*) percentage of caregivers to execute diversification process, the number of iteration to start solving SCM (*kFC*), the number of iterations between two SCM processes (*nFC*) and the number of iterations for increasing caregivers in diversification process (*iFS*). We conduct some trial experiments and finally set *minS* as 0.3, *maxS* as 0.8, *kFC* as 15, *nFC* as 5 and iFS as 3. In the following, we first test the functionality of the adaptive mechanism used in IP-Insert. Then we use the basecase data and compare the IP-Insert VNS algorithm, ILP solver in CPLEX (ILP_C), IP-Post VNS algorithm, VNS algorithm. Finally, sensitivity analyses are performed.

B. Effectiveness of adaptive mechanism

For the IP-Insert VNS, the adaptive mechanism, i.e., the adaptive choice for the number *r* of caregivers in Section IV.B and the rule to trigger SCM in Section IV.D, plays an important role cooperated with SCM. When trapped in the local optima, more feasible routes should be generated and provided for the model by shaking more caregivers. Therefore, the probability of jumping out of the trap increases. When the current best solution is frequently improved, fewer caregivers are selected for shaking so as to enhance the effect of local search.

To verify the effectiveness of the adaptive mechanism, we compare IP-Insert VNS with the algorithm without adaptive mechanism which is denoted by IP-Insert-N. Note that for IP-Insert-N, we use an increasing number of caregivers with the iteration for diversification instead of the adaptive mechanism. The two algorithms are run for 10 times and the stopping criterion is set to 90 seconds. Table V shows the average cost and Standard Deviation (StD) over 10 replications of two algorithms. In conclusion, IP-Insert can reduce the total cost by 5.3% averagely compared with IP-Insert-N. In addition, the StD of IP-Insert is much smaller than that of IP-Insert-N. Note that IP-Insert can provide 5654 different routes while only 1092 different routes are generated in the process of IP-Insert-N. Fig.2 shows the downtrend of two algorithms, which indicates the efficiency of IP-Insert VNS.

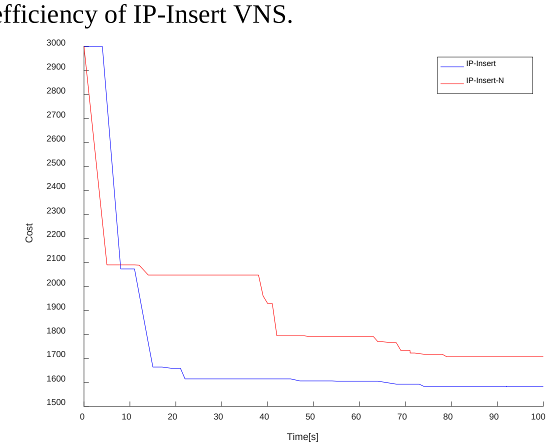

Fig.2. Comparison of IP-Insert and IP-Insert-N algorithms

TABLE V
VERIFICATION OF ADAPTIVE MECHANISM

Algorithm	Optimal		Percentage Gap (%)			StD
	Cost	Time (s)	Average	Maximum	Minimum	
IP-Insert	1563.1	402.5	1.5	1.9	1.1	4.5
IP-Insert-N	1563.1	402.5	6.8	11.6	3.3	50.9

C. Real-life case study

According to our consultant with managers of the company, the weights of four objectives are estimated. Using the normalization method proposed in section III, the weights is estimated as [7.194, 4.107, 8.591, 25.51] respectively for preference mismatch, travel distance, unpaid overtime and exceeded number of maximum number of caregivers sent to clients.

The optimal solution obtained with CPLEX is 1563.1, which is obtained in 402.5 seconds. For all the four algorithms, 10 independent repetitions are executed. For IP-Insert VNS, we set the maximum number of iterations as 100. To compare with the other three algorithms, we get the running time of IP-Insert VNS which is used as the maximum running time of ILP_C, IP-Post VNS and VNS algorithms. Table VI shows the performance of four algorithms, where the average value (Ave) and the StD are over 10 replications. The percentage gap can be calculated as:

$$Percentage\ Gap = (\frac{solution\ of\ algorithm}{optimal\ solution} - 1) \times 100\%$$

The average and maximum percentage gap and the average running time is shown in the table. For IP-Insert VNS, the average gap is 1.5% and the maximum gap is 2.6%, which indicates that IP-Insert can get near-optimal solutions stably. The StD over 10 replications of two algorithms also illustrates the stability of IP-Insert. The running time is 54 seconds, which is rather shorter than the CPLEX. For ILP_C, IP-Post and VNS, the average gaps are 7.1%, 5.1% and 10.7% respectively. Although sometimes they can also get near-optimal solutions, the fluctuation of them is much larger than IP-Insert. Furthermore, the converge process of IP-Insert and IP-Post is also depicted in Fig.3, in which we prolong the running time to 90s in order to depict the trend. As the IP-Post is based on VNS and SCM only functions at the end of the VNS process, the convergence process of VNS is included in that of IP-Post. From Fig.3, IP-Insert can rapidly decrease to a lower level of total cost and can get satisfactory solutions within 25 seconds, while VNS converges slowly before the post optimization. The box plot in Fig.4 shows that the results of IP-Insert is more superior to the other two algorithms and CPLEX. In addition, there is no feasible solution in the early process of IP-Insert, which is due to the large percentage of caregivers for diversification. Although diversification process is harmful for convergence, it can generate diverse routes quickly and is beneficial for the SCM.

TABLE VI
COMPARISON OF THREE ALGORITHMS

Algorithm	Ave	StD	Percentage Gap (%)		Time (s)
			Average	Maximum	
IP-Insert	1586.7	6.6	1.5	2.6	54
ILP_C	1674.5	16.4	7.1	8.0	54
IP-Post	1643.0	26.1	5.1	8.4	55.2
VNS	1730.0	36.9	10.7	14.7	54

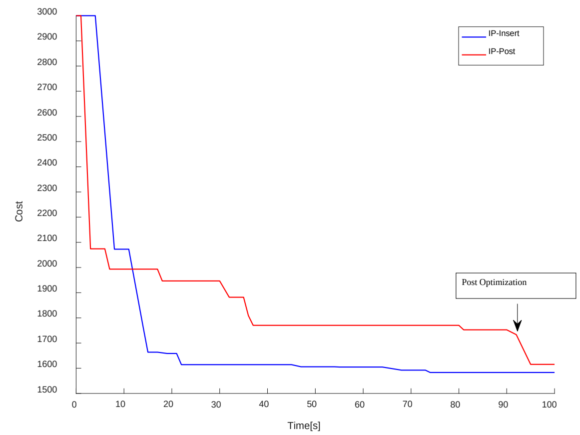

Fig.3. Downtrend of three algorithms

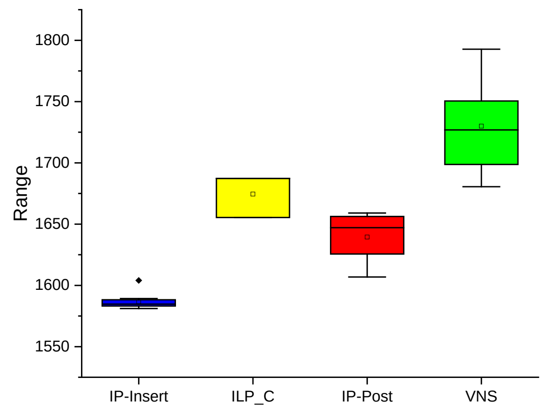

Fig.4. Box plot of IP-Insert, ILP_C, IP-Post, and VNS

D. Sensitivity analyses

In this subsection, we present some sensitivity analyses of the weights of four objectives using IP-Insert VNS algorithm. Then we test IP-Insert VNS with a larger instance. In each real-life case, the cost of corresponding objective is changed by multiplying a factor. Each case is repeated 10 times and the maximum number of iterations is 100. Other parameters setting is the same as that in the base case. The optimal cost and running time of CPLEX, the average and maximum percentage gap, StD over 10 replications and average running time of IP-Insert VNS are presented in Table VII. In sensitivity analyses, the performance of IP-Insert VNS is similar to that of the base case. The cost decreases rapidly in the early process, then the algorithm focuses on exploitation and attempts to improve the current best solution. Therefore, we only focus on IP-Insert VNS in the sensitivity analysis part. For all cases, it is enough to stabilize the outcome after 100 iterations, so it is reasonable to analyze the variation of cost by using the outcome. For the larger instance, CPLEX is unable to get the optimal solution; so we compare IP-Insert VNS algorithm with IP-Post VNS and VNS.

Impact of preference mismatch penalty: Preference part of Table VII and Fig.5a show the results of sensitivity analysis by varying the preference penalty. Note that the average exceeded number of caregivers for each client (continuity indicator) in the figure is 10 times larger than the original value because the exceeded number is too small. Generally, the total cost increases with the multiplier of preference mismatch penalty and the StD under each multiplier is below 8. From Fig.5a, the overall distance basically remain unchanged and continuity indicator has a slightly downward trend. In contrast, the preference mismatch decreases significantly and overtime increases dramatically. The results show that overtime and preference are sensitive to the change of preference mismatch penalty.

Impact of travelling cost: Travelling part of Table VII and Fig.5b show the results of sensitivity analysis by varying the travelling cost. Generally, the total cost increases with the multiplier and StD values are all under 10. From Fig.5b, the four objective indicators have little variation, which indicates that the change of total cost in Travelling part of Table VII are almost caused by the variation of travelling cost and the four objectives are not so sensitive to the travelling cost.

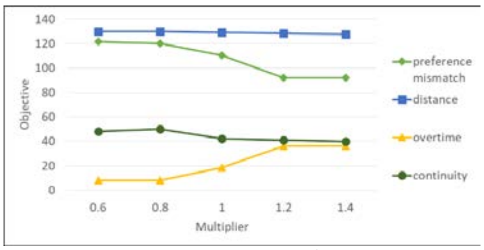

Fig.5a. Sensitivity analysis on preference mismatch

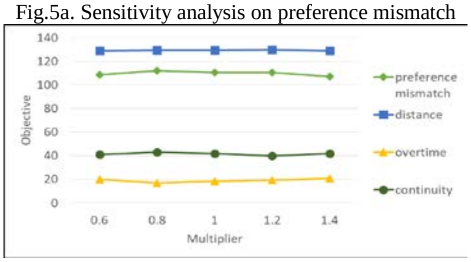

Fig.5b. Sensitivity analysis on traveling cost

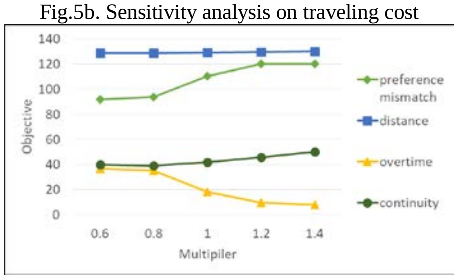

Fig.5c. Sensitivity analysis on overtime cost

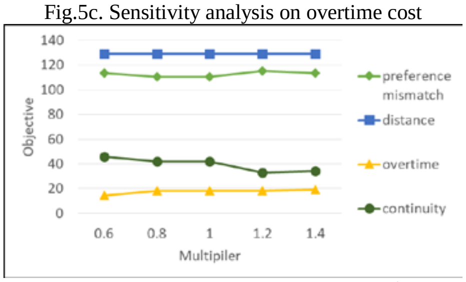

Fig.5d. Sensitivity analysis on continuity of care
Fig.5 Sensitivity analysis

Impact of overtime cost: Overtime part of Table VII and Fig.5c show the results of sensitivity analysis by varying overtime cost. From Overtime part of Table VII, the total cost increases with the multiplier and StD values are nearly all larger those in the other sensitivity analyses, which indicates that the overtime cost matters a lot for the total cost. From Fig.5c, continuity indicator increases slightly, while overtime decreases sharply and preference mismatch increases significantly. The results show that preference mismatch and overtime are both sensitive to the variation of overtime cost.

Impact of penalty for exceeded number of caregivers: Continuity part of Table VII and Fig.5d show the results of sensitivity analysis of varying penalty for exceeded number of caregivers. Similarly, the total cost increases with the multiplier. The overtime decreases slightly and overtime increases slightly with the multiplier respectively. However, generally four objectives are not very sensitive to the variation of penalty.

Impact of problem size: To test the ability of IP-Insert VNS algorithm for tackling larger problems, we construct a large-scale instance based on the real-life case. There are 100 caregivers, 40 clients with 225 requests in the instance. CPLEX cannot solve this large case. For IP-Insert VNS, the number of iterations is set to 100. The average running time is obtained and set as the maximum running time for IP-Post VNS. We run the three algorithms 5 replications. The results show that the average gap of IP-Post from IP-Insert VNS is 8%, and that of VNS is 14.5%. The running time of IP-Insert VNS is less than 1000 second. Fig. 6 shows the results of three replications, where the three blue and red lines separately denote three replications for IP-Insert VNS and IP-Post VNS. Fig.6 shows that IP-Insert VNS can get better solutions compared with VNS and IP-Post VNS. Although IP-Post VNS may bring sharp reduction of cost in the early stage, it leads to local convergence ultimately. Thanks to the SCM, IP-Post VNS can largely reduce the cost compared to VNS.

To sum up, although CPLEX can obtain the optimal with an acceptable time for the small-scale problem, it is unable to solve the problem when its scale gets a little bit larger. Considering the serious problem of population aging and the consequent growth of home services, it is reasonable to foresee a larger volume of clients in the near future. As our approach is intended as a decision support tool for the considered provider and other similar providers, a key requirement is the capability to solve even larger instances. Thus, heuristic approaches as the one we propose turn out to be necessary.

TABLE VII
RESULTS OF SENSITIVITY ANALYSIS

Objective & Multiplier		Optimal		Percentage Gap (%)		StD	Time (s)
		Cost	Time (s)	Average	Maximum		
Preference	0.6	1241.1	401.2	1.4	2.3	8.3	42
	0.8	1413.5	360.1	1.4	2.1	7.1	46
	1.0	1563.1	402.5	1.5	2.6	6.6	56
	1.2	1705.1	353.3	1.6	2.7	8.3	64
	1.4	1837.5	369.0	1.3	1.7	4.5	57
Travelling	0.6	1346.7	351.4	2.2	3.5	4.3	72
	0.8	1454.9	463.3	1.8	2.5	4.3	72
	1.0	1563.1	402.5	1.5	2.6	6.6	56
	1.2	1671.4	393.9	1.4	2.1	5.5	56
	1.4	1779.6	424.1	1.2	1.8	8.1	75
Overtime	0.6	1462.8	457.2	0.9	1.3	3.5	63
	0.8	1517.7	477.7	1.2	1.4	2.8	58
	1.0	1563.1	402.5	1.5	2.6	6.6	56
	1.2	1597.5	464.2	0.7	1.6	6.2	58
	1.4	1613.9	351.8	0.3	0.7	3.1	73
Continuity	0.6	1522.3	342.9	1.6	3.2	9.6	64
	0.8	1542.7	461.3	1.7	3.1	8.7	68
	1.0	1563.1	402.5	1.5	2.6	6.6	56
	1.2	1583.5	362.1	2.1	3.4	10.6	56

	1.4	1603.9	443.4	1.7	2.3	5.9	58

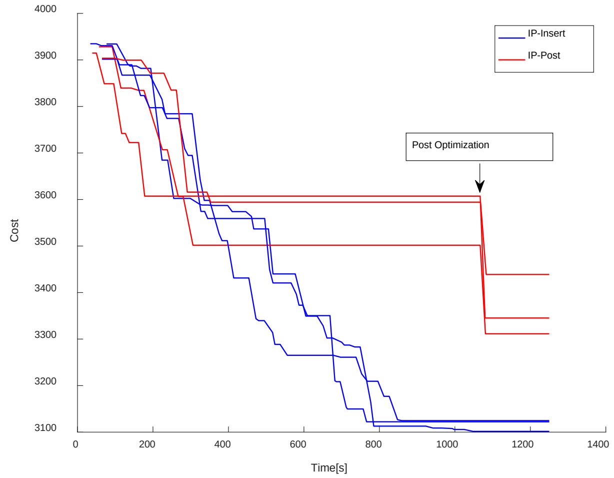

Fig.6. Results of large-scale instance

VI. Conclusion And Future Work

In this paper, we study a particular HC scheduling problem according to the request of a HC provider in New York City. We consider chargeable overtime, which is rarely concerned in the HC literature, preference matching, and continuity of care. To deal with this problem, an integer programming model is proposed to minimize the total cost, including travelling cost, overtime cost, preference mismatch and penalty of violating continuity of care. A matheuristic algorithm is developed, which combines VNS with a SCM. The SCM can be used to generate the optimal solution with route pool and guide the searching direction of VNS. The components, including generation of initial solution, diversification process and intensification process are designed in VNS. To realize the cooperation with the SCM, some adaptive mechanisms are designed to provide superior routes for the SCM.

The computational results demonstrate the applicability and efficiency of IP-Insert VNS algorithm. Compared with the method in [4], IP-Insert VNS can obtain better results more efficiently and stably. Sensitivity analyses are conducted and the results show how the objectives react to the change of the parameters. Managerial insights can be obtained from our approach and analyses. The managers should keep an eye on the variation of other operating costs when adjusting specific operating costs. For the real case in this work, the preference satisfaction and overtime have mutually exclusive relationships, which can help managers to set a suitable salary standard.

This paper can be extended in several aspects. First, the objective of minimizing the additional overtime cost paid by clients can be integrated in the objective function (e.g., with the formulation of minimizing the maximum overtime cost paid by clients). Another extension is to consider the dynamic conditions like the cancellation of service.

Acknowledgement

This research is sponsored by the National Natural Science Foundation of China under Grants [71972129, 71671111 and 72031007].

References

[1] J. Guzman, A. Pawliczko, S. Beales, C. Till, and I. Voelcker, "Ageing in the twenty-first century: A celebration and a challenge. New York, NY: United Nations Population Fund," ed, 2012.

[2] E. Lanzarone and A. Matta, "Robust nurse-to-patient assignment in home care services to minimize overtimes under continuity of care," *Operations Research for Health Care,* vol. 3, pp. 48-58, 2014.

[3] L. Malagodi, E. Lanzarone, and A. Matta, "Home Care Vehicle Routing Problem with Chargeable Overtime and Strict and Soft Preference Matching," *Health care management science,* 2020.

[4] Y. Zhu, A. Matta, E. Lanzarone, and N. Geng, "A Variable Neighborhood Search for Home Care Scheduling Under Chargeable Overtime and Preference Matching," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 281-286.

[5] P. Eveborn, P. Flisberg, and M. Rönnqvist, "Laps Care—an operational system for staff planning of home care," *European journal of operational research,* vol. 171, pp. 962-976, 2006.

[6] G. Carello and E. Lanzarone, "A cardinality-constrained robust model for the assignment problem in Home Care services," *European Journal of Operational Research,* vol. 236, pp. 748–762, 2014.

[7] R. Liu, X. Xie, and T. Garaix, "Hybridization of tabu search with feasible and infeasible local searches for periodic home health care logistics," *Omega,* vol. 47, pp. 17-32, 2014.

[8] R. Liu, B. Yuan, and Z. Jiang, "A branch-and-price algorithm for the home-caregiver scheduling and routing problem with stochastic travel and service times," *Flexible Services and Manufacturing Journal,* vol. 31, pp. 989-1011, 2019.

[9] R. Liu, Y. Tao, and X. Xie, "An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and synchronized visits," *Computers & Operations Research,* vol. 101, pp. 250-262, 2019.

[10] P. M. Duque, M. Castro, K. Sörensen, and P. Goos, "Home care service planning. The case of Landelijke Thuiszorg," *European Journal of Operational Research,* vol. 243, pp. 292-301, 2015.

[11] H. Qin, W. Ming, Z. Zhang, Y. Xie, and A. Lim, "A tabu search algorithm for the multi-period inspector scheduling problem," *Computers & Operations Research,* vol. 59, pp. 78-93, 2015.

[12] S. Yalçındağ, A. Matta, E. Şahin, and J. G. Shanthikumar, "The patient assignment problem in home health care: using a data-driven method to estimate the travel times of care givers," *Flexible Services and Manufacturing Journal,* vol. 28, pp. 304-335, 2016.

[13] S. Yalçındağ, P. Cappanera, M. G. Scutellà, E. Şahin, and A. Matta, "Pattern-based decompositions for human resource planning in home health care services," *Computers & Operations Research,* vol. 73, pp. 12-26, 2016.

[14] S. Riazi, O. Wigström, K. Bengtsson, and B. Lennartson, "A Column Generation-Based Gossip Algorithm for Home Healthcare Routing and Scheduling Problems," *IEEE Transactions on Automation Science and Engineering,* vol. 16, pp. 127-137, 2018.

[15] Y. Zhan and G. Wan, "Vehicle routing and appointment scheduling with team assignment for home services," *Computers & Operations Research,* vol. 100, pp. 1-11, 2018.

[16] M. I. Gomes and T. R. P. Ramos, "Modelling and (re-) planning periodic home social care services with loyalty and non-loyalty features," *European Journal of Operational Research,* vol. 277, pp. 284-299, 2019.

[17] M. Chaieb, J. Jemai, and K. Mellouli, "A decomposition-construction approach for solving the home health care scheduling problem," *Health care management science,* pp. 1-23, 2019.

[18] F. Grenouilleau, A. Legrain, N. Lahrichi, and L. Rousseau, "A set partitioning heuristic for the home health care routing and scheduling problem," *European Journal of Operational Research,* vol. 275, pp. 295-303, 2019.

[19] R. Liu, B. Yuan, and Z. Jiang, "The Large-Scale Periodic Home Health Care Server Assignment Problem: A Region-Partition-Based Algorithm," *IEEE Transactions on Automation Science and Engineering,* pp. 1-12, 2020.

[20] F. Grenouilleau, N. Lahrichi, and L.-M. Rousseau, "New decomposition methods for home care scheduling with predefined visits," *Computers & Operations Research,* vol. 115, p. 104855, 2020.

[21] Z. B. Zabinsky, P. Dulyakupt, S. Zangeneh-Khamooshi, C. Xiao, P. Zhang, S. Kiatsupaibul, *et al.*, "Optimal collection of medical specimens and delivery to central laboratory," *Annals of Operations Research,* vol. 287, pp. 537-564, 2020.

[22] C. Fikar and P. Hirsch, "Home health care routing and scheduling: A review," *Computers & Operations Research,* vol. 77, pp. 86-95, 2017.

[23] M. Cissé, S. Yalçındağ, Y. Kergosien, E. Şahin, C. Lenté, and A. Matta, "OR problems related to Home Health Care: A review of relevant routing and scheduling problems," *Operations Research for Health Care,* vol. 13, pp. 1-22, 2017.
[24] P. Cappanera and M. G. Scutelli, "Joint Assignment, Scheduling, and Routing Models to Home Care Optimization: A Pattern-Based Approach," *Transportation Science,* vol. 49, pp. 830-852, 2015.
[25] J. Wirnitzer, I. Heckmann, A. Meyer, and S. Nickel, "Patient-based nurse rostering in home care," *Operations Research for Health Care,* vol. 8, pp. 91-102, 2016.
[26] I. Dayarian, T. G. Crainic, M. Gendreau, and W. Rei, "A branch-and-price approach for a multi-period vehicle routing problem," *Computers & Operations Research,* vol. 55, pp. 167-184, 2015.
[27] N. Azi, M. Gendreau, and J.-Y. Potvin, "An exact algorithm for a vehicle routing problem with time windows and multiple use of vehicles," *European Journal of Operational Research,* vol. 202, pp. 756-763, 2010.
[28] J. F. Bard, Y. Shao, X. Qi, and A. I. Jarrah, "The traveling therapist scheduling problem," *IIE Transactions,* vol. 46, pp. 683-706, 2014.
[29] C. Tilk, M. Drexl, and S. Irnich, "Nested branch-and-price-and-cut for vehicle routing problems with multiple resource interdependencies," *European Journal of Operational Research,* vol. 276, pp. 549-565, 2019.
[30] C. Archetti, O. Jabali, and M. G. Speranza, "Multi-period vehicle routing problem with due dates," *Computers & Operations Research,* vol. 61, pp. 122-134, 2015.
[31] A. Trautsamwieser and P. Hirsch, "A Branch‐Price‐and‐Cut approach for solving the medium‐term home health care planning problem," *Networks,* vol. 64, pp. 143-159, 2014.
[32] I. Rodríguez-Martín, J.-J. Salazar-González, and H. Yaman, "The periodic vehicle routing problem with driver consistency," *European Journal of Operational Research,* vol. 273, pp. 575-584, 2019.
[33] J. W. Escobar, R. Linfati, P. Toth, and M. G. Baldoquin, "A hybrid granular tabu search algorithm for the multi-depot vehicle routing problem," *Journal of heuristics,* vol. 20, pp. 483-509, 2014.
[34] I. Dayarian, T. G. Crainic, M. Gendreau, and W. Rei, "An adaptive large-neighborhood search heuristic for a multi-period vehicle routing problem," *Transportation Research Part E: Logistics and Transportation Review,* vol. 95, pp. 95-123, 2016.
[35] P. Grangier, M. Gendreau, F. Lehuédé, and L.-M. Rousseau, "An adaptive large neighborhood search for the two-echelon multiple-trip vehicle routing problem with satellite synchronization," *European Journal of Operational Research,* vol. 254, pp. 80-91, 2016.
[36] T. Vidal, T. G. Crainic, M. Gendreau, N. Lahrichi, and W. Rei, "A hybrid genetic algorithm for multidepot and periodic vehicle routing problems," *Operations Research,* vol. 60, pp. 611-624, 2012.
[37] A. Rahimi-Vahed, T. G. Crainic, M. Gendreau, and W. Rei, "Fleet-sizing for multi-depot and periodic vehicle routing problems using a modular heuristic algorithm," *Computers & Operations Research,* vol. 53, pp. 9-23, 2015.
[38] A. Rahimi-Vahed, T. G. Crainic, M. Gendreau, and W. Rei, "A path relinking algorithm for a multi-depot periodic vehicle routing problem," *Journal of heuristics,* vol. 19, pp. 497-524, 2013.
[39] S. Nickel, M. Schröder, and J. Steeg, "Mid-term and short-term planning support for home health care services," *European Journal of Operational Research,* vol. 219, pp. 574-587, 2012.
[40] İ. Muter, Ş. İ. Birbil, and G. Şahin, "Combination of metaheuristic and exact algorithms for solving set covering-type optimization problems," *INFORMS Journal on Computing,* vol. 22, pp. 603-619, 2010.
[41] M. Hewitt, M. Nowak, and N. Nataraj, "Planning strategies for home health care delivery," *Asia-Pacific Journal of Operational Research,* vol. 33, p. 1650041, 2016.
[42] O. Grodzevich and O. Romanko, "Normalization and other topics in multi-objective optimization," 2006.
[43] S. Huber and M. J. Geiger, "Order matters–A Variable Neighborhood Search for the Swap-Body Vehicle Routing Problem," *European Journal of Operational Research,* vol. 263, pp. 419-445, 2017.
[44] M. R. Khouadjia, B. Sarasola, E. Alba, L. Jourdan, and E.-G. Talbi, "A comparative study between dynamic adapted PSO and VNS for the vehicle routing problem with dynamic requests," *Applied Soft Computing,* vol. 12, pp. 1426-1439, 2012.
[45] Y. Kuo and C.-C. Wang, "A variable neighborhood search for the multi-depot vehicle routing problem with loading cost," *Expert Systems with Applications,* vol. 39, pp. 6949-6954, 2012.
[46] Y. Rochat and É. D. Taillard, "Probabilistic diversification and intensification in local search for vehicle routing," *Journal of heuristics,* vol. 1, pp. 147-167, 1995.
[47] M. S. Rasmussen, T. Justesen, A. Dohn, and J. Larsen, "The home care crew scheduling problem: Preference-based visit clustering and temporal dependencies," European Journal of Operational Research, vol. 219, pp. 598-610, 2012.

Xuran Gong received the B.S. and the M.S. degree in industrial engineering from the Hunan University, Changsha, China, in 2015 and 2018, respectively. He is currently Ph.D. student at the Department of Industrial engineering and Management, Shanghai Jiao Tong University, Shanghai, China. His research interests include health care operations management.

Na Geng (M'12) received the Ph.D. degree in industrial engineering from the Ecole Nationale Superieure des Mines de Saint-Etienne, Saint-Étienne, France, and the Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2010. She is currently a Professor with the Sino-US Global Logistics Institute, SJTU. She is the Principle Investigator of three NSF of China funded projects. Her research interests include production and service operational management. Dr. Geng has been an Associate Editor of the *Flexible Service and Manufacturing* journal since 2017.

Yanran Zhu received the B.S. in industrial engineering from the Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2013. She is currently pursuing the double M.S. with the Department of Industrial Engineering and Management, Shanghai Jiao Tong University, Shanghai, China and the Department of Mechanical Engineering, Politecnico di Milano, Milan, Italy. Her research interests include health care operations management.

Andrea Matta is Professor of Manufacturing at Politecnico di Milano. He was Distinguished Professor at the School of Mechanical Engineering of Shanghai Jiao Tong University from 2014 to 2016. His research area includes analysis, design and management of manufacturing and health care systems. He is Editor in Chief of Flexible Services and Manufacturing Journal since 2017, past editorial board member of OR Spectrum journal and IEEE Robotics and Automation Letters journal. He was awarded with the Shanghai One Thousand Talent and Eastern Scholar in 2013.

Ettore Lanzarone received the Ph.D. and the M.S. cum laude in Bioengineering from Politecnico di Milano, Milan, Italy, in 2008 and 2004, respectively. He is currently researcher at the Institute for Applied Mathematics and Information Technology "E. Magenes" (IMATI) of the National Research Council of Italy (CNR), Milan, Italy, and Adjunct Professor at the Politecnico di Milano, Milan, Italy, and the University of Bergamo, Bergamo, Italy. Moreover, he is member of the CIRRELT (Centre Interuniversitaire de Recherche sur les Reseaux d'Entreprise, la Logistique et le Transport), Montréal and Quebec City, Canada. His current research activities include operations research for health care and industrial problems, statistical modelling with particular reference to Bayesian approaches, and the application of mathematical approaches to bioengineering problems.

APPENDIX A

In this part, we construct two different integer linear programming (ILP) models for our specific home care scheduling problem (HCSP). The difficulty of constructing the model lies in how to calculate the travelling distance when there are two types of visiting modes, namely serve two jobs successively and serve two jobs with going home between them. In the manuscript, we capture the latter visiting mode using a set of variable y_{ij}^k. In this part, we use a set of binary variable u_{ij} to denote whether the caregiver needs to go home between and another set of binary variable w_{ij}^k to calculate the travelling distance when caregivers need to go home.

All the notations in the two new models are presented in Table A.I. The formulation of route constraints in Model 1 is similar to that in [47], $\mathrm{N}^k = I \cup \{0^k, n^k\}$ denotes the set of jobs (including the start job and end job) for caregiver k.

Similar to the original model in our paper, the preference mismatch between each caregiver and each client can be calculated beforehand as follows:

$$\gamma_c^k = \sum_{q\in M}((\pi_{cq}-\omega_{kq})^2\lambda_{cq}),\ \forall c\in C, k\in K$$

Model 1:

The objective is:

$$\min \alpha_1 \sum_{k\in K}\sum_{c\in C}\sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} x_{ij}^k \gamma_c^k d_j \theta_j^c + \alpha_2 T + \alpha_3 \sum_{k\in K}\sigma_k + \alpha_4 \sum_{c\in C} E_c \quad \text{(a1)}$$

Route constraints:

$$\sum_{k\in K}\sum_{i\in \mathrm{N}^k} x_{ij}^k = 1, \quad \forall j\in I \quad \text{(a2)}$$

$$\sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} x_{ij}^k d_j \le S_k, \quad \forall k\in K \quad \text{(a3)}$$

$$\sum_{j\in \mathrm{N}^k} x_{0^k,j}^k = 1, \quad \forall k\in K \quad \text{(a4)}$$

$$\sum_{i\in \mathrm{N}^k} x_{i,n^k}^k = 1, \quad \forall k\in K \quad \text{(a5)}$$

$$\sum_{i\in \mathrm{N}^k} x_{ir}^k - \sum_{j\in \mathrm{N}^k} x_{rj}^k = 0, \quad \forall k\in K, r\in I \quad \text{(a6)}$$

$$t_i + d_i + \delta_{ij} \le t_j + M(1-x_{ij}^k), \quad \forall k\in K, i\in \mathrm{N}^k, j\in \mathrm{N}^k \quad \text{(a7)}$$

$$t_j - (t_i + d_i + \delta_{ij}) - \beta \le M u_{ij}, \quad \forall i\in I, j\in I \quad \text{(a8)}$$

$$T = \sum_{k\in K}\sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} x_{ij}^k \delta_{ij} + \sum_{k\in K}\sum_{i\in I}\sum_{j\in I} w_{ij}^k(\delta_{i,0^k} + \delta_{0^k,j} - \delta_{ij}) \quad \text{(a9)}$$

$$w_{ij}^k = u_{ij} x_{ij}^k, \quad \forall i,j\in I, k\in K \quad \text{(a10)}$$

Equation (a10) can be linearized as follows.

$$w_{ij}^k \le u_{ij}$$

$$w_{ij}^k \le x_{ij}^k$$

$$w_{ij}^k \ge u_{ij} + x_{ij}^k - 1$$

TABLE A.I

SETS, PARAMETERS, AND VARIABLES FOR ILP

Notations for HCSP	
Sets	**Description**
I	Set of jobs
N^k	Set of jobs (including the start job and end job) for caregiver k
C	Set of clients
K	Set of caregivers
M	Set of preferences
WE	Set of weekend hours in a week
NS	Set of night shifts in a week
ON	Set of hours after 9 p.m. each day in a week
Parameters	**Description**
S_k	Maximum working hours of caregiver k
t_i	Starting time of job i
β	Max duration of a break for caregivers not going home
θ_i^c	Binary, 1 if job i is of client c
mg_c^k	Binary, 1 if caregiver k must do a job of client c
ns_i^k	Binary, 1 if job i cannot be allotted to caregiver k
τ	Maximum working hours according to regulation
d_i	Duration of job i
δ_{ij}	Travel time from job i to job j
ε_c	Binary, 1 if client c is willing to pay for overtime
ξ_k	Binary, 1 if k is willing to work on weekends
ν_k	Binary, 1 if k is willing to work on night shifts
φ_k	Binary, 1 if k only accepts job after 9 p.m.
γ_c^k	Preference mismatch of client c and caregiver k
η_c	Max number of caregivers that can be sent to client c
π_c	Vector of preferences for client c
ω_k	Vector of characteristic for caregiver k
λ_c	Vector of interests of client c
Variables	**Description**
x_{ij}^k	Binary, 1 if job j is done after i by caregiver k
z_i^k	Binary, 1 if job i is done by caregiver k
T	Overall travelling time
p_c^k	Binary, 1 if caregiver k does at least one job of client c
O_c^k	Overtime of caregiver k worked on client c
σ_k	Total overtime not paid by clients
E_c	Exceeded number of caregivers sent to client c
u_{ij}, w_{ij}^k, g_c^k	Auxiliary binary variables

The overtime can be calculated as follows

$$O_c^k = \max\{0, \sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} x_{ij}^k \cdot d_j \cdot \theta_j^c - \tau\}, \quad \forall c\in C, k\in K$$

$$\sigma_k \ge \sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} x_{ij}^k \cdot d_j - \tau - \sum_{c\in C} O_c^k \varepsilon_c, \forall k\in K \quad \text{(a11)}$$

$$\sigma_k \ge 0, \quad \forall k\in K$$

E_c represents the exceeded number of caregivers for client c and can be calculated as follows:

$$M\cdot p_c^k \ge \sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} \theta_j^c x_{ij}^k, \quad \forall c\in C, k\in K$$

$$p_c^k \le M\cdot \sum_{i\in \mathrm{N}^k}\sum_{j\in \mathrm{N}^k} \theta_j^c x_{ij}^k, \quad \forall c\in C, k\in K \quad \text{(a12)}$$

$$\sum_{k\in K} p_c^k \le \eta_c + E_c, \quad \forall c\in C$$

$$E_c \ge 0, \quad \forall c\in C$$

Other constraints:

$$\sum_{i\in \mathrm{N}^k} x_{ij}^k \le 1 - ns_j^k, \quad \forall j\in I, k\in K \quad \text{(a13)}$$

$$p_c^k \ge mg_c^k, \quad \forall c\in C, k\in K \quad \text{(a14)}$$

$$\sum_{i\in \mathrm{N}^k}\sum_{j\in I; t_j\in WE} x_{ij}^k \le M\xi_k, \quad \forall k\in K \quad \text{(a15)}$$

$$\sum_{i\in N^k}\sum_{j\in I;t_j\in NS} x_{ij}^k \le M\nu_k, \quad \forall k\in K \tag{a16}$$

$$\sum_{i\in N^k}\sum_{j\in I;t_j\notin ON} x_{ij}^k \le M(1-\varphi_k), \quad \forall k\in K \tag{a17}$$

In equation (a1), the first component represents the cost of preference mismatch. The second part is the travelling cost and the total travelling time is obtained by summing up the travelling time between caregivers and clients, travelling time between different clients as well. The third and fourth parts represent overtime cost and penalty for violating continuity of care respectively.

Constraints (a2) ensure that each job is done exactly once. Constraints (a3) ensure that the total working time of caregivers cannot exceed the maximum hours each caregiver has declared. Constraints (a4)-(a6) guarantee the continuity of each caregiver's route. Constraints (a7) impose the time precedence of jobs on one route. Constraints (a8) denote whether the caregiver needs to go home between job i and job j. Constraint (a9) calculates the overall travelling distance and the relations between w_{ij}^k, u_{ij} and x_{ij}^k are illustrated in constraints (a10). Constraints (a13)-(a14) deal with "Not send" and "Must go" lists respectively. Constraints (a15)-(a17) guarantee that, if a caregiver is not available at night or on the weekend of before 9 p.m., he/she is not assigned jobs in this period.

Model 2:

The objective is:

$$\begin{aligned} \min\ & \alpha_1\sum_{k\in K}\sum_{c\in C}\sum_{i\in I} z_i^k\gamma_c^k d_i\theta_i^c + \alpha_2 T + \\ & \alpha_3\sum_{k\in K}\sigma_k + \alpha_4\sum_{c\in C}E_c \end{aligned} \tag{a18}$$

Route constraint:

$$\sum_{i\in I} d_i\cdot z_i^k \le S_k, \ \forall k\in K \tag{a19}$$

$$\sum_{k\in K} z_i^k = 1, \quad \forall i\in I \tag{a20}$$

$$\begin{aligned} &\sum_{i\in N^k} x_{ir}^k - z_r^k = 0, \quad \forall r\in I, k\in K \\ &\sum_{j\in N^k} x_{rj}^k - z_r^k = 0, \quad \forall r\in I, k\in K \end{aligned} \tag{a21}$$

$$\begin{aligned} &(t_i + d_i + \delta_{ij}) - t_j \le M(1-x_{ij}^k), \\ &\qquad\qquad \forall k\in K,\ i,j\in N^k \end{aligned} \tag{a22}$$

$$\begin{aligned} &t_j - (t_i + d_i + \delta_{ij}) - \beta \le M u_{ij}, \quad \forall i,j\in I \\ &w_{ij}^k = u_{ij} x_{ij}^k \end{aligned} \tag{a23}$$

$$T = \sum_{k\in K}\sum_{i\in N^k}\sum_{j\in N^k} x_{ij}^k\delta_{ij} + \sum_{k\in K}\sum_{i\in I}\sum_{j\in I} w_{ij}^k(\delta_{i,0^k} + \delta_{0^k,j} - \delta_{ij}) \tag{a24}$$

Other constraints:

$$z_i^k \le 1 - ns_i^k, \quad \forall i\in I, k\in K \tag{a25}$$

$$p_c^k \ge mg_c^k, \quad \forall c\in C, k\in K \tag{a26}$$

$$\sum_{i\in I;t_i\in WE} z_i^k \le M\xi_k, \quad \forall k\in K \tag{a27}$$

$$\sum_{i\in I;t_i\in NS} z_i^k \le M\nu_k, \quad \forall k\in K \tag{a28}$$

$$\sum_{i\in I;t_i\notin ON} z_i^k \le M(1-\varphi_k), \quad \forall k\in K \tag{a29}$$

$$z_i^k, x_{ij}^k, p_c^k, w_{ij}^k, u_{ij} \in \{0,1\} \tag{a30}$$

The overtime can be calculated as follows:

$$\begin{aligned} &O_c^k = \max\{0, \sum_{i\in I} z_i^k\cdot d_i\cdot\theta_i^c - \tau\}, \quad \forall c\in C, k\in K \\ &\sigma_k \ge \sum_{i\in I} z_i^k\cdot d_i - \tau - \sum_{c\in C} O_c^k\varepsilon_c, \ \forall k\in K \\ &\sigma_k \ge 0, \quad \forall k\in K \end{aligned} \tag{a31}$$

E_c represents the exceeded number of caregivers for client c and can be calculated as follows:

$$\begin{aligned} &M\cdot p_c^k \ge \sum_{i\in I}\theta_i^c z_i^k, \quad \forall c\in C, k\in K \\ &p_c^k \le M\cdot\sum_{i\in I}\theta_i^c z_i^k, \quad \forall c\in C, k\in K \\ &\sum_{k\in K} p_c^k \le \eta_c + E_c, \quad \forall c\in C \\ &E_c \ge 0, \quad \forall c\in C \end{aligned} \tag{a32}$$

The meaning of constraints can be referred to those in Model 1. However, we use z_i^k to express the relation of caregivers and jobs in Model 2.

Comparison of models:

In this part, we compare the three models (including the original model in our paper) using the real case data. We implement three models in C++ and CPLEX (Version 12.9) and run on a computer equipped with Intel(R) Core(TM) i7@ 2.60GHz and 8GB RAM. The number of parallel threads is 12.

In conclusion, the original model is the best. As reported in the paper, the optimal solution is 1563.1, which is obtained in 402.5 seconds. For Model 1, we set the running time as one hour and the solution is 1598.5, whose percentage gap with the optimal is 2.3%. Model 2 can obtain the optimal solution while the running time is 1554.5 seconds. In addition, we test three models with the running time for IP-Insert in the paper (54 seconds). For each model, 10 independent repetitions are executed and the results are presented in Table A.II, where ILP_C denotes the model used in our paper and it can obtain the best solution among three models. Compared with ILP_C, Model 1 and Model 2 can only obtain the solution with the percentage gap above 30%.

TABLE A.II
COMPARISON OF THREE MODELS

Model	Ave	StD	Percentage Gap (%)		Time (s)
			Average	Maximum	
ILP_C	1674.5	16.4	7.1	8.0	54
Model 1	2134.7	0	36.6	36.6	54
Model 2	2045.9	0	30.9	30.9	54

APPENDIX B

In this Appendix we extend our model to enclose flexible starting times within predefined time windows. We assume that each job i can start within a hard time window [a_i, b_i], and that variable t_i^k denotes the actual starting time when caregiver k serves job i. Accordingly, the extended model can be formulated by replacing constraints (8) and (9) of the original model with the following constraints (b1)-(b5).

$$t_i^k + x_{ij}^k(d_i + \delta_{ij}) + y_{ij}^k(d_i + re + \tilde{\delta}_i^k + \tilde{\delta}_j^k) \\ \leq t_j^k + b_i(1 - x_{ij}^k - y_{ij}^k), \quad \forall i, j \in I; k \in K \tag{b1}$$

$$t_j^k - (t_i^k + d_i + \delta_{ij}) - \beta \leq M u_{ij}^k, \quad \forall i, j \in I; k \in K \tag{b2}$$

$$x_{ij}^k \leq M(1 - u_{ij}^k), \quad \forall i, j \in I; k \in K \tag{b3}$$

$$t_i^k \leq b_i z_i^k \quad \forall i \in I, k \in K \tag{b4}$$

$$t_i^k \geq a_i z_i^k \quad \forall i \in I, k \in K \tag{b5}$$

Constraints (b1) ensure that the starting times of successive jobs are set correctly, while constraints (b2) and (b3) ensure that the caregivers go home if there is enough time. Finally, constraints (b4) and (b5) ensure the starting times to be in the allowed time window.

This extended model has been tested by setting the time window of each job i as $a_i = t_i - \Delta$ and $b_i = t_i + \Delta$, where t_i is the exact starting time in the base case and Δ is set to 1, 2 or 3, respectively (Δ=0 refers to the base case with no flexibility). Results in TABLE B.I show that the total cost is reduced when Δ is larger while the computational time increases dramatically.

TABLE B.I
IMPACT OF TIME WINDOW

Δ	Objective function value	Computational time (s)
0	1563.1	402.5
1	1545.7	1819.5
2	1536.4	2931.0
3	1512.8	7065.1