Research Article
Academic Platform Journal of Engineering and Smart Systems (APJESS) 11(3), 135-150, 2023
Received: 23-Feb-2023 Accepted: 25-Aug-2023
homepage: https://dergipark.org.tr/tr/pub/apjess
https://doi.org/10.21541/apjess.1255522

Tabu Search with Variable Neighborhood Search Algorithm for Home Healthcare Routing Problem for Multiple Hospitals with Balanced Workload

*[1]Gülçin DİNÇ YALÇIN, [2]Tuğçe YAVUZ, [3]Şüheda ALTINTAŞ

[1] Department of Industrial Engineering, Eskisehir Technical University, Eskişehir, Türkiye, gdinc@eskisehir.edu.tr
[2] Department of Industrial Engineering, Eskisehir Technical University, Eskişehir, Türkiye, yavuz_tgc@hotmail.com
[3] Department of Industrial Engineering, Eskisehir Technical University, Eskişehir, Türkiye, suhedaaltintas@gmail.com

Abstract

In this paper, we study home healthcare routing and scheduling problem where multiple hospitals serve patients. In the public hospitals in healthcare system of Türkiye, patients requiring home healthcare are assigned to the hospital that serves their place of residence. This can cause the workload of hospitals to become unbalanced in terms of the time needed for both traveling and operation. The aim of this paper is to generate routes with a balanced workload for hospitals, giving consideration to the time windows of patients and the working hours of health workers. Firstly, we construct a mathematical model which can solve toy and small-scale problems whilst taking into account the importance of a balanced workload. Then, a Tabu Search with a Variable Neighborhood Search (TS-VNS) algorithm is developed to solve large-scale problems. The performance of the TS-VNS algorithm is tested by comparing the results of the mathematical model with the generated test problems at a small scale. Additionally, large-scale test problems from the literature are sourced for the problem and solved by the TS-VNS algorithm. The results demonstrate the efficiency of the TS-VNS algorithm.

Keywords: home healthcare, tabu search, variable neighborhood search, workload balance, OR in healthcare

1. INTRODUCTION

There are many patients who are unable to go to hospitals for various reasons, such as old age, and home healthcare (HHC) has a significant impact on their well-being as it has been shown that life quality increases with HHC [1]. Since the population of people over 60 years is more than 1 billion and the rate of population aging is higher than ever before [2], HHC services will become increasingly important in the future. In fact, the size of the global HHC market was $345.6 billion in 2022, and it is expected to grow at an annual rate of 7.9% from 2022 to 2030 [3].

In HHC services, the scheduling of appointments is complicated by the fact that a suitable time has to be found between two parties: the patients, with their preferred time windows, and the caregivers, such as nurses, doctors, physiotherapists, with their own hours of working and busy schedules. Additionally, all routes start from and end at the related hospital, pharmacy, laboratory, etc., but the order in which patients are visited (the routes) presents an optimization problem. All these operations in HHC service were first modeled as a Vehicle Routing Problem (VRP) with time windows by [4] (see also [5, 6, 7, 8, 9] for detailed survey). Generally, the objective is to minimize the total cost (time) of all the routes.

In the public hospitals HHC system of Türkiye, each hospital that has caregivers for HHC serves a predetermined area, and patients are assigned to whichever hospital covers their place of residence. Given that hospitals have different capacities and that each area hosts a varied number of patients, hospitals are often burdened with unbalanced workloads and this may impact the quality of HHC offered by the hospitals that handle relatively more patients. In an attempt to address this issue, we propose a new problem for HHC with multi hospital and time windows under balanced workload of objectives. This problem is referred to as the multi-hospital home-healthcare routing problem with balanced workload (MH-HHCRP-BW). The basic constraints of MH-HHCRP-BW are no route between hospitals, each route starts from and ends at the hospital, and each patient is served once. Additionally, the working hours and the time schedule for each patient is taken into account with time windows constraints.

Firstly, we develop a mathematical model for MH-HHCRP-BW. Then, since the model can solve the problem with a limited number of patients and hospitals, we propose a hybrid of a Tabu Search and Variable Neighborhood Search (TS-VNS) algorithm since VNS provides the flexibility of designing of neighborhood structures and TS prevents cycles in the search space.

Additionally, we generate some small problems to solve by both the mathematical model and TS-VNS. Then, we compare the performance of the TS-VNS with the solutions obtained from mathematical model. Finally, multi-depot VRP test problems with time windows are modified to generate a large-scale data set, and then, these problems are solved by TS-VNS.

The main contribution of this paper can be summarized as follows: 1) We create a multi-hospital HHC model. 2) We aim to generate solutions with balanced workloads among hospitals. 3) We model the problem as a mixed-integer mathematical model. 4) We propose a TS-VNS algorithm with a new insertion operator to solve the problem within a reasonable time. 5) We generate new small-scale test problems and modify the large-scale from the literature for the problem, which we refer to as MH-HHCRP-BW.

The rest of the paper is organized as follows: A literature review is given in Section 2. The mathematical model of the problem is described in detail in Section 3. The TS-VNS algorithm is explained in Section 4. The computational results are given in Section 5. Finally, some conclusions are drawn in Section 6.

2. LITERATURE REVIEW

Bredstöm and Rönnqvist [10] studied the formulation of a combined VRP with time windows and additional temporal constraints and presented a home healthcare problem as a VRP. Furthermore, they proposed balancing constraints for vehicles; however, they optimized balance service duration or traveling time and model the problem for one hospital. In this paper, we extend the problem for multiple hospitals and we balance the workload in both service duration and traveling time.

Lanzarone and Matta [11] dealt with home healthcare problem by only considering the patient-nurse assignment under the objective function-balancing maximum over workload of nurses. Each nurse has a predetermined workload and if this workload is not exceeded for each nurse, then the value of the objective function is zero whether the workload between nurses is balanced or not. Carello et al. [12] studied a patient-nurse assignment problem, where they addressed the problem from the perspectives of patients, nurses, and service providers. To solve the problem, they employed integer linear programming and utilized different objective functions to optimize the assignment process.

Yuan et al. [13] worked on the HHC problem in the case of patients' stochastic service times and caregivers' skill requirements. A stochastic model was proposed and, while column generation was used to solve the master problem, a label algorithm was developed to solve the pricing sub-problem. Rest and Hirsch [14] considered the problem with time-dependent public transport since most of the caregivers from the Austrian Red Cross in Vienna use a combination of public transport and walking. Then, they proposed a tabu search to solve the problem.

Shi et al. [15] dealt with a HHC problem with fuzzy demands related to the quantity of drugs required for each customer and designed a fuzzy change constraint model. Additionally, a hybrid genetic algorithm integrated with stochastic simulation methods was developed. Masmoudi and Cheikhrouhou [16] considered HHC for one hospital with a heterogeneous fleet and a lunch break under the objective of cost minimization. They introduced a mathematical model and developed the Adaptive Large Neighborhood Search. Liu et al. [17] dealt with stochastic travel and service time and proposed a method that combines a branch-and-price algorithm and a discrete approximation method.

Bahadori-Chinibelagh et al. [18] proposed a multi-depot VRP with time windows model for HHC where depots are pharmacies. Each route starts from a pharmacy and ends at the related laboratory. The objective is to minimize the total cost. Additionally, two constructive heuristics were developed. Our study differs from it in terms of the objective function and the fact that each route starts from and ends at the related hospital.

Tanoumand and Ünlüyurt [19] considered new resource constraints: there are two types of personnel providing the service, but the number of personnel is limited. Then, they proposed an exact algorithm, the branch-and-price algorithm, to solve the problem. Li et al. [20] extended the problem for outpatient services and considered a new objective: minimizing the waiting times for outpatients. Then, they adopted an outer-approximation method and developed a hybrid genetic algorithm.

Besides these studies, in order to solve the HHC routing problem, Allaoua [21] proposed a matheuristic based on the decomposition of mathematical programming. Cappanera et al. [22] conducted a study that involves scheduling, assignment, and routing decisions for random requests. To address this complex problem, they employed a matheuristic approach. Frifita et al. [23] developed a general variable neighborhood search (see also [24] for a detailed survey about variable neighborhood search in healthcare management). Rahimian et al. [25] proposed a hybrid approach that combines integer programming and variable neighborhood search methods. Riazi et al. [26] studied decomposition and distributed algorithms, and Riazi et al. [27] proposed a gossip-column generation algorithm. Moussavi et al. [28] proposed a new mathematical formulation and then developed a matheuristic approach based on the decomposition of the mathematical formulation. Grenouilleau et al. [29] presented a set partitioning heuristic based on a set partitioning formulation and a large neighborhood search framework, and Grenouilleau et al. [30] developed new decomposition methods for predefined visits. Shahnejat-Bushehri et al. [31] conducted a study that involves random travel and transaction times. They employed three different meta-heuristic algorithms (i.ei. simulated anneling, genetic algorithm, and memtic algorithm) to address this problem with the objective of minimizing the total processing time. Dekhici et al. [32] solved the home-care problem by utilizing the firefly algorithm, which is a metaheuristic algorithm. They modeled the problem as a vehicle routing problem with time windows, where the objective function aimed to minimize the total route time. Hassani et al. [33] introduced the differential evaluation algorithm as a solution approach for a nurse scheduling problem. The objective of their proposed algorithm is to minimize the overall cost, which includes various factors such as overtime, undertime, employment, and other related costs.

Furthermore, the HHC problem can also have multiple objectives. Rasmussen et al. [34] considered three objectives: minimizing the total cost, maximizing preference-based visits of caregiving sans patients, and maximizing the number of visited patients. Nickel et al. [35] dealt with the HHC problem by minimizing four objectives: the number of unscheduled tasks, the nurse-patient loyalty and overtime study, and the distance travelled by all nurses. It should be noted that only total overtime study and distance travelled is aimed to be minimized and the workload balanced is not taken into consideration. Mankowska et al. [36] considered minimizing three objectives: the total distance travelled, the total tardiness of services and the maximal tardiness observed overall service operation. The third objective function is similar to our objective function, however the tardiness is determined if a service lasts after the time windows of a patient. Thus, if all services can be scheduled between time windows of patients, then the value of the third objective function is equal to zero whether the workload is balanced or not. Braekers et al. [37] studied the HHC routing problem with two objectives: minimizing the cost and customer inconvenience. They applied an ϵ^{-}constraint method to solve their mathematical model and also developed a metaheuristic based on a multi-directional local search.

Moreover, Hertz and Lahrichi [38] studied the problem for balancing workload of both service time and traveling time. Yalcindag et al. [39] proposed a data-driven method to estimate the travel times and dealt with the HHC problem with balancing the workload. Yalcindag et al. [40] presented a two-stage approach for addressing a home health care problem. This approach gradually combines the stages of assignment, planning, and scheduling decisions. The objective of the proposed model is to balance the workloads of operators, taking into account both travel time and service time. Decerle et al. [41] considered three objective functions: balancing the workload, and minimizing both the total time and patients' dissatisfaction. They formulated the problem and then developed a hybrid memetic-ant colony optimization algorithm. Kandakoglu et al. [42] tackled the scheduling and routing problem of home health nurses by employing a mixed-integer linear programming algorithm. The objective of their study was to balance the workload among the nurses effectively. Gomes et al. [43] focused on addressing a multi-objective problem in a system that involves patient-caregiver loyalty and dynamic patient numbers. Their study aimed to optimize multiple objectives, including workload balancing, minimizing total travel time, and minimizing the variation in visiting hours. Yang et al. [44] dealt with three objectives: minimizing the route cost, improving service consistency and balancing the workload. Additionally, the travel and service time were considered to be uncertain, so a multi-objective artificial bee colony framework was developed to solve the problem. The workload objective of these three studies [34-44] is similar to our work, but those covered the case of only one hospital.

We summarized the literature in Table 1 and it is shown from Table 1, the problem MH-HHCRP-BW defined in this paper cover all the criteria: balance workload for both service duration and traveling time, and multiple hospital. It is seen from Table 1 that there are studies that deal with balancing the workload taking into consideration only service duration or only travelling time or both of them. However, all these problems are for one hospital. A HHC system may have multiple hospitals, e.g., the main motivation of this paper, public hospitals in Türkiye. Addition, there are private healthcare companies having multiple affiliates in an area and providing services for HHC. Thus, balancing the workload for only one hospital of the healthcare systems having multiple hospitals is not sufficient to be fair among healthcare workers of the relevant healthcare system. According to best of our knowledge, our study is the first considering all these criteria, especially for multiple hospitals.

Table 1. Literature Review

Study	**Balance the Workload**		**Multiple Hospital**
	Service Duration	**Travelling Time**	
Bredstöm and Rönnqvist [10]	–	+	–
Lanzarone and Matta [11]	+	–	–
Carello et al. [12]	+	–	–
Yuan et al. [13]	–	–	–
Rest and Hirsch [14]	–	–	–
Shi et al. [15]	–	–	–
Masmoudi and Cheikhrouhou [16]	–	–	–
Liu et al. [17]	–	–	–
Bahadori-Chinibelagh et al. [18]	–	–	+
Tanoumand and Ünlüyurt [19]	–	–	–
Li et al. [20]	–	–	–
Allaoua [21]	–	–	–
Cappanera et al. [22]	+	+	–
Frifita et al. [23]	–	–	–
Rahimian et al. [25]	–	–	–
Riazi et al. [26]	–	–	–
Riazi et al. [27]	–	–	–
Moussavi et al. [28]	–	–	–
Grenouilleau et al. [29]	–	–	–
Grenouilleau et al. [30]	–	–	–
Shahnejat-Bushehri et al. [31]	–	–	–
Dekhici et al. [32]	–	–	–
Hassani et al. [33]	–	–	–
Rasmussen et al. [34]	–	–	–
Nicket et al. [35]	–	–	–
Mankowska et al. [36]	–	–	–
Braekers et al. [37]	–	–	–
Hertz and Lahrichi [38]	+	+	–
Yalcindag et al. [39]	+	+	–
Yalcindag et al. [40]	+	+	–
Decerle et al. [41]	+	+	–
Kandakoglu et al. [42]	+	+	–
Gomes et al. [43]	+	+	–
Yang et al. [44]	+	+	–
This study	+	+	+

3. PROBLEM DESCRIPTION AND MATHEMATICAL MODEL

In the MH-HHCRP-BW, the assumptions are: there is more than one hospital; patients and caregivers have time windows; each patient is visited at one time by one caregiver; a route between the hospitals is not allowed; and each route starts from and ends at a hospital. A demonstration of the problem is given in Figure 1. The pentagons and circles represent hospitals and patients, respectively.

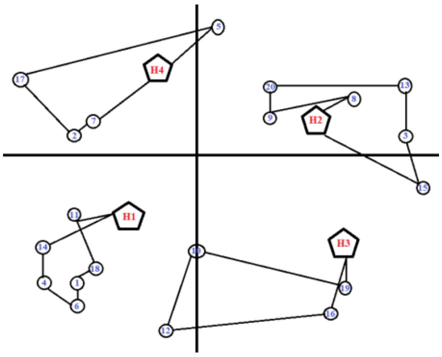


Figure 1. A demonstration of the MH-HHCRP-BW

First, we define sets, parameters and decision variables of the problem as follows.

Sets:

K	Set of routes.
I_0	Set of hospitals.
I_1	Set of patients.
$I_0 \cup I_1 = I$	Set of hospitals and patients.

Parameters:

T_{ij}	Travel time between ith hospital/patient and jth hospital/patient, $\forall i,j \in I$.
D_i	Operation time for the ith patient, $\forall i \in I_1$.
$[a_i, b_i]$	Time windows for the ith patient, $\forall i \in I_1$.
$[e_i, l_i]$	Time windows (working hours) for the ith hospital, $\forall i \in I_0$.

Decision variables:

x_{ijk} $\begin{cases} 1, \text{if route } k \text{ goes from } i \text{ to } j \\ 0, \text{otherwise} \end{cases}$, $\forall i,j \in I, i \neq j, k \in K$.

t_i The start time of the visit to the ith patient, $\forall i \in I_1$.

The MILP is reformulated for the MH-HHCRP-BW as follows (see e.g. [45, 46]).

$$min \max_{\forall k,l \in K, l \neq k} \{\sum_{i\in I}\sum_{j\in I}(T_{ij}+D_i)x_{ijk} - \sum_{i\in I}\sum_{j\in I}(T_{ij}+D_i)x_{ijl}\} \ , \tag{1}$$

subject to

$$\sum_{k\in K}\sum_{i\in I_0} x_{ijk} = 0 \quad \forall j \in I_0 \tag{2}$$

$$\sum_{k\in K}\sum_{i\in I} x_{ijk} = 1 \quad \forall j \in I_1 \tag{3}$$

$$\sum_{k\in K}\sum_{i\in I} x_{ijk} = \sum_{k\in K}\sum_{i\in I} x_{jik} \quad \forall j \in I_1 \tag{4}$$

$$\sum_{j\in I_1} x_{ijk} = \sum_{j\in I_1} x_{jik} \quad \forall i \in I_0, k \in K \tag{5}$$

$$\sum_{k\in K}\sum_{j\in I_1} x_{ijk} = 1 \quad \forall i \in I_0 \tag{6}$$

$$t_{ik} + (T_{ij} + D_i) \leq t_{ik} + b_i(1 - x_{ijk}) \quad \forall i,j \in I, k \in K \tag{7}$$

$$a_i \sum_{j\in I} x_{ijk} \leq t_{ik} \leq b_i \sum_{j\in I} x_{ijk} \quad \forall i \in I_1, k \in K \tag{8}$$

$$e_i \leq t_{ik} \leq l_i \ \forall i \in I_0, k \in K \tag{9}$$

$$x_{ijk} \in \{0,1\} \ \forall i,j \in I, k \in K \tag{10}$$

$$t_{ik} \geq 0 \ \forall i \in I_1 \ . \tag{11}$$

The objective (1) is to balance the workload between hospitals. The workload contains both the traveling time and the operation time for patients. Equation (2) prevents travel between hospitals. Equation (3) ensures that every patient is visited once. Equation (4) says that if a patient is visited by a hospital team, then the team should leave from the patient. Equation (5) declares that the team should leave from a hospital and then return to the hospital. Equation (6) ensures that only one home healthcare vehicle leaves from a hospital. Equation (7) takes into consideration both the traveling time and operation time of the patient to determine the visiting time of the patient. Equation (8) restricts the visiting time of a patient to the suitable time window of the patient. Equation (9) ensures that the vehicles for home healthcare leave and return to hospitals within a work shift. Equations (10) and (11) specify the variable domains. The objective function (1) for multiple hospitals and the equation (7) with operations time of patients and the equation (8) for the work shift of home healthcare team are newly defined in this paper.

The objective function (1) is linearized as following:

$$\min w \tag{12}$$

$$\sum_{i\in I}\sum_{j\in I}(T_{ij}+D_i)x_{ijk} - \sum_{i\in I}\sum_{j\in I}(T_{ij}+D_i)x_{ijl} \leq w \ \ \forall k,l \in K, i \neq j \tag{13}$$

Thus, the final model is to minimize (12) under the constraints (2)-(11) and (13).

Now, the importance of the new objective function is explained using a toy problem. Consider the following objective function that is minimizing the travel time of the routes.

$$\min \sum_k \sum_{i\in I}\sum_{j\in I} T_{ij} x_{ijk} \tag{14}$$

When the objective function about time minimization (14) is taken into consideration, an unbalanced workload may occur. A toy problem is generated (the data for the toy problem is given in the appendix). Then, the toy problem is solved with the model, having objective function (12) under the constraints (2)-(11), (13), and objective function (14) under the constraints (2)-(11), individually. The results are given in detail in Table 2. The time difference between the hospitals having maximum and minimum total time (the sum of total travel time and total operation time) is 15.7 minutes for objective function (12) while it is 135.1 minutes for objective function (14). It is clearly seen that the new objective function ensures a balanced workload among hospitals in terms of both travel time from hospital to homes and operation time at homes.

Table 2. Results of the toy problem for the objective functions (12) and (14)

Objective function	Hospital number	Total travel time	Total operation time	Total time
(12)-balancing the workload	Hospital 1	210	19.4	229.4
	Hospital 2	195	34.8	229.8
	Hospital 3	208	25.1	233.1
	Hospital 4	200	17.4	217.4
			Maximum time difference	15.7
(14)-minimizing total time	Hospital 1	264	29.2	293.2
	Hospital 2	259	22.9	281.9
	Hospital 3	138	20.1	158.1
	Hospital 4	152	24.5	176.5
			Maximum time difference	135.1

4. TABU SEARCH WITH VARIABLE NEIGHBOURHOOD SEARCH (TS-VNS)

At this stage of the study, we use a heuristic and a metaheuristic algorithm together to solve the problem. Heuristic algorithms are preferred because they can find acceptable results in a relatively short time for problems that are difficult to solve or take a long time with classical optimization approaches. While heuristic algorithms offer problem-specific approaches, metaheuristic algorithms provide a guiding framework for solving the problem. In this study we use a heuristic algorithm to generate a feasible initial solution. After that we apply the hybrid algorithm, we propose which combines two different metaheuristic algorithms: Tabu search and Variable neighborhood search.

4.1. The Initial Solution

We developed a heuristic algorithm to find a feasible initial solution. The algorithm that is given in Figure 2 assigns each route in random order by taking into account the time window constraints. The algorithm continues this assignment process until all patients are assigned. We used permutation representation to represent routes since the permutation representation provides an efficient representation of the solution space of our problem and is advantageous for search procedures. And so, the solution is generated by the heuristic algorithm has matrix form in which each row is a route.

One of the parameters used in the pseudocode in Figure 2, **lb**, is a vector which represents the lower bounds of the time window constraints. The variable **nodes** represent the patient group to be assigned to routes, and **routes** represents the number of routes to be created.

```
Function InitialSolution (nodes, routes, lb)
for each x:                    /* x is the notation of routes */
    x[0] = argmin(lb)
    delete lb[argmin(lb)]
    delete nodes [x[0]]
while nodes ≠ ∅
    routes = Permutation (routes)
    for each r:
        Generate a list of nodes eligible to be assigned to the current route.
        if list ≠ ∅
            Choose the node which has a minimum time window lower bound in the list.
        else
            break
        end if
    end for
end while
```

Figure 2. Pseudo code for feasible initial solution

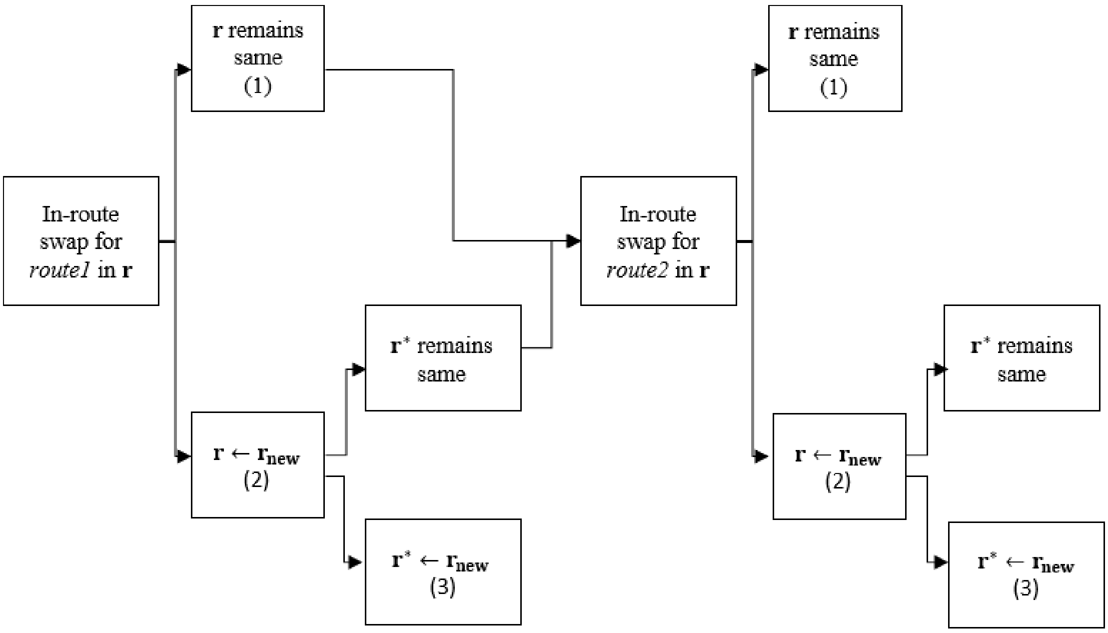
4.2. TS-VNS

The tabu search algorithm is a metaheuristic with memory [47]. The memory is a tabu list that stores previous solutions or moves to prevent cycles in the search space. The algorithm starts with a feasible solution and proceeds by searching for neighboring solutions of this current solution with move operators. The algorithm calculates a fitness value for each feasible neighboring solution it reaches. If this fitness value is better than the fitness value of the best available solution, the current best solution is updated. If there is no better solution than the current best solution among the neighboring solutions, the current solution is updated with a feasible neighbor solution with a worse fitness value. The searching continues from this current solution. Besides, the variable neighborhood search aims to reach the global optimum by changing the neighborhood of a local search stuck in the local optimum, and it also uses neighborhood change in descent to the local optimum [48].

In the TS-VNS approach, we use some procedures of Tabu search and Variable neighborhood search algorithms together. The part that we inherit from the VNS algorithm in the proposed approach is the use of three different move operators which are inter-route swap, in-route swap, and insertion. We use inter-route swap and in-route swap operators nested. Inter-route swap disrupts the initial solution and diversion the local search to another area. Then, in-route swap searches feasible and improvement solutions and thus searching intensifies in this search area. This nested loop iterates over a deterministic number of iterations. And finally, we aim to shake the current solution in a controlled way with the insertion operator. The algorithm uses tabu lists for swap operators and this is the part of taken from the TS algorithm. We use the short-term memory and aim to prevent circles in the search space with using tabu lists as with the TS algorithm. Pseudocode of the TS-VNS algorithm is given in Figure 3.

Function TS-VNS (**r**, *end1*, *end2*)
r* = **r** *\/* the best current solution *\/*
Z^* = ObjValue(**r***)
iter = 0
Create a empty tabu list for inter-route swap operation which called *tabucswap*
while *iter* < *end1:* */*inter-route swap phase*/*
    Choose 2 random routes and 2 random positions $[route1, route2, pos1, pos2]$ for inter-route swap operation and check if $[route1, route2, pos1, pos2]$ in *tabucswap*
    If this move is a tabu then repeat previous step
    Apply inter-route swap operation on current solution and update current solution as $\mathbf{r}[route1][pos1], \mathbf{r}[route2][pos2] \leftarrow \mathbf{r}[route2][pos2], \mathbf{r}[route1][pos1]$
    if r is feasible
        $Z = \mathrm{ObjValue}(\mathbf{r})$
        if $Z \leq Z^*$
            $\mathbf{r}^* \leftarrow \mathbf{r}$
            $Z^* \leftarrow \mathrm{ObjValue}(\mathbf{r})$
        end if
    else */*in-route swap phase*/*
        Apply swap operation for *route1* and update current solution as $\mathbf{r} \leftarrow \mathrm{Swap}(\mathbf{r}, \mathbf{r}[route1], i)$
        if swap operator found a better solution
            $\mathbf{r}^* \leftarrow \mathbf{r}$
            $Z^* \leftarrow \mathrm{ObjValue}(\mathbf{r})$
        else
            Apply swap operation for *route2* and update current solution as $\mathbf{r} \leftarrow \mathrm{Swap}(\mathbf{r}, \mathbf{r}[route2], i)$
            if swap operator found a better solution
                $\mathbf{r}^* \leftarrow \mathbf{r}$
                $Z^* \leftarrow \mathrm{ObjValue}(\mathbf{r})$
            end if
        end if
    end if
    $iter := iter + 1$
    Append $[route1, route2, pos1, pos2]$ to *tabucswap*
end while
$iter = 0$
while *iter* < *end2* */*insertion phase*/*
    $\mathbf{r_{new}} = \mathrm{Insertion}(\mathbf{r})$
    if $\mathbf{r_{new}}$ is feasible:
        $\mathbf{r} \leftarrow \mathbf{r_{new}}$
        $Z = \mathrm{ObjValue}(\mathbf{r})$
        if $Z \leq Z^*$
            $\mathbf{r}^* \leftarrow \mathbf{r}$
            $Z^* \leftarrow \mathrm{ObjValue}(\mathbf{r})$
        end if
    end if
    $iter := iter + 1$
end while

Figure 3. Pseudo code for TS-VNS



The numbers represent the following conditions, respectively:
1) There is no different feasible neighboring solution,
2) A different feasible neighboring solution ($\mathbf{r_{new}}$) is exist,
3) $Z_{new} \leq Z^*$.
Figure 4. In-route swap phase in TS-VNS algorithm.

Function Swap (**r**, **route**, *i*)
$\mathbf{r_{new}} = \mathbf{r}$
$iter = 0$
Create a empty list called *tabuswap*
Calculate *t* which is total number of swap move for **route** which has *n* elements: $t = n(n-1)/2$
While *iter* < t
Choose 2 random position [*pos1*, *pos2*] and check if this moves in *tabuswap*
If this move is a tabu then repeat previous step
Swap elements on these position in **route**: **route** [*pos1*], **route** [*pos2*] = **route** [*pos2*], **route** [*pos1*]
Append [*pos1*, *pos2*] to *tabuswap*
Update *i*. route in current solution $\mathbf{r_{new}}$ with new **route**
if $\mathbf{r_{new}}$ is feasible
$\mathbf{r} \leftarrow \mathbf{r_{new}}$
break
end if
$iter := iter + 1$
end while

Figure 5. Pseudo code for in-route swap operator

The first inputs of the TS-VNS function, **r**, is a feasible initial solution. We find this feasible initial solution by random assignments and for this, we run the function in Figure 2 repeatedly until it finds a feasible solution. The other four inputs are hyperparameters, and decision maker determines their values. The first hyperparameter, *iteration*, is the number of iterations of outside loop which is the repeat number of process, and similarly *inter-route swap* represents the number of the inter-route swaps and *insertion* represents the number of insertions in the Figure 3. The last hyperparameter, *tabu*, is the size of the tabu list for the inter-route swap operator. If the tabu list is complete when the last move is added to the tabu list, the oldest move is removed from the list. The in-route swap operator does not remove any tabu from the list as it searches for all neighboring solutions on a route.

The TS-VNS procedure starts with the inter-route swap, which applies a swap operation between two randomly determined routes. The algorithm finds a new solution with the inter-route swap operator and moves on with this current solution (**r**). If **r** is a feasible solution that provides improvement, the current best solution is updated. If **r** is not feasible, the algorithm switches to another move operator. In this point, searching moves on with in-route swap operator. This operator aims to search the local area to reach any feasible solution.

After the inter-route swap, there is two routes to apply in-route swap process. We follow a sequential procedure to not miss the benefit of the swap made on one of the routes. Firstly, the algorithm applies the in-route swap for *route1* and it continues according to the result obtained. The in-route swap operator can produce two different kinds of solutions: i) there is a different feasible neighboring solution and current solution is updated with this new solution, ii) current solution is not updated. If the in-route swap operator finds a new feasible solution, the objective function value of this new current solution is calculated. If the new current solution improves the current best solution, the current best solution is also updated. After the in-route swap operation if the current solution is the same, or if the new solution is not better than the current best solution then the in-route swap operator is applied on *route2*. The algorithm returns to the inter-route swap phase after the second in-route swap operation and repeats the same steps. We explain this phase of the algorithm with a flowchart in Figure 4. As seen in the figure, if the algorithm reaches a new feasible solution by neighborhood search, the current solution is updated regardless of whether this solution is better than the current best solution. In this way, it is aimed to search for good solutions that can be reached over relatively bad solutions. After this first loop, which nested the diversification and intensification procedures, the algorithm switches the search operator for the last time.

The in-route swap operator gets current solution **r** which is a 2-dimensional vector, **route** which is a one-dimensional vector and *i* which is index of **route** in **r** as inputs. Then, it chooses 2 random position and swaps elements on these position in **route**. Finally, it updates **route** in **r** with new **route**. With the swap operator, it is possible to search for all

Possible neighboring solutions. If neighboring searching find a feasible solution, then search stops, and the current route is updated with the new route. In the situation that is all neighboring solutions are searched, and no feasible solution is found, the current solution remains the same. Pseudo code for in-route swap operator is given in Figure 5.

Function Insertion (**r**)
Calculate workload of every route on current solution **r** and hold with an array; **w**
Detect routes with maximum and minimum workloads: **max_w** and **min_w**
Choose a random element in **max_w** and remove it from **max_w**, then insert this element in a random position in **min_w**
Update **r** with these new two routes

Figure 6. Pseudo code for insertion operator

Third and last move operator is the insertion operator (see Figure 6) that is modified in a way that contributes to the workload balance, which is the objective of the problem. The proposed insertion operator is named as “balanced-insertion” operator. The balanced-insertion operator transfers a patient from the route with the highest workload to the route with the least workload. In this way, we aim to find better neighboring solutions with small steps.

5. COMPUTATIONAL RESULTS

For numerical results, we used two types of test problems, which were classified as small-scale and large-scale. Table 3 summarizes the information about the test problems. Small-scale problems were generated, while large-scale problems were modified from the literature [49]. Small-scale problems were solved by both mathematical model and TS-VNS algorithm. Obtaining solutions of large-scale problems by the mathematical model within reasonable time is difficult due to the size of problems. Thus, these problems were solved by only the TS-VNS algorithm. In this section, we showed the performance of the TS-VNS algorithm against the solution of the mathematical model obtained by General Algebraic Modeling system (GAMS) with CPLEX solver for small-scale problems. In addition, we analyze the efficiency of the TS-VNS algorithm in large-scale problems in terms of time and the improvement it provides in random feasible initial solutions. We solved all runs by a computer with a core i7 processor (2.3 GHz) and 16 GB RAM.

Table 3. Properties of test problems

	Problem no	**Number of patients**	**Number of caregivers team for each hospital**	**Number of hospitals**
Small-scale	pr-s-1	30	1	6
	pr-s-2	35	1	7
	pr-s-3	40	1	8
	pr-s-4	45	1	9
Large-scale	pr-l-1	48	2	4
	pr-l-2	96	3	4
	pr-l-3	144	5	4
	pr-l-4	192	7	4
	pr-l-5	240	8	4
	pr-l-6	288	9	4
	pr-l-7	72	2	6
	pr-l-8	144	3	6
	pr-l-9	216	5	6
	pr-l-10	288	6	6
	pr-l-11	48	2	4
	pr-l-12	96	4	4
	pr-l-13	144	6	4
	pr-l-14	192	7	4
	pr-l-15	240	8	4
	pr-l-16	288	11	4
	pr-l-17	72	2	6
	pr-l-18	144	4	6
	pr-l-19	216	5	6
	pr-l-20	288	7	6

In all of the solutions of the test problems with the TS-VNS algorithm, we performed three repetitions by considering the random initial solution of the algorithm and the random search mechanism in the structure of the algorithm. The value of time in the tables are the total duration of three repetitions, and the “best value” and “average value” are the best and average values found during this period. Since the purpose of the problem is the balanced distribution of workloads in terms of time, the results of the TS-VNS algorithm also express time.

5.1. Hyperparameter Optimization

A hyperparameter tuning is required to determine the hyperparameter values (the number of iterations, inter-route, insertion, and tabu length) of the TS-VNS algorithm. For this purpose, we used a Python module called “hyperopt” [50]. Hyperopt is based on Bayesian optimization and supports automated hyperparameter optimization. Four basic components need to be defined to use hyperopt: an objective function to be minimized, a search space including hyperparameter values, a database to be used to store the points evaluated during the search, and a search algorithm to enable the transition between points in the search space.

Table 4. Search spaces of hyperparameters for hyperopt

Hyperparameter	Parameter of uniform distribution	
	Small-scale	Large-scale
Iteration	(100, 500)	(100, 1000)
Inter-route swap	(100, 500)	(100, 1000)
Insertion	(50, 500)	(100, 1000)
Tabu length	(10, 50)	(10, 100)

The objective function for this problem is to balance the total workload, that is, to minimize the difference between the workloads. Search space can be made up of discrete values as well as probability distributions. We used uniform distribution. Finally, tree of Parzen estimators (TPE) [51] and random search can be used for the search algorithm. We used TPE which is a sequential model-based optimization approach.

We performed hyperparameter optimizations for small-scale and large-scale problems separately with using the problems pr-s-4 and pr-l-10 for small-scale problems and large-scale problems, respectively. We chose these problems because they are relatively large. Table 4 summarizes the distribution parameters we used for the search space. We made different trials before deciding on the distribution parameters. As a result of the guidance of these trials, we used search spaces with as wide ranges as possible.

At the end of the search, which took about 3 hours (170.6 min) for small-scale and about 21 hours (1249.4 min) for large-scale, the hyperparameter values we determined with hyperopt were [iteration:466, inter-route swap: 190, insertion: 429, tabu: 41] for small-scale problems and [iteration: 726, inter-route swap: 887, insertion: 318, tabu: 49] for large-scale problems.

5.2. Small-Scale Test Problems

We solved the small-scale test problems by both the mathematical model and TS-VNS algorithm. We used the General Algebraic Modeling System (GAMS) with CPLEX solver to solve the mathematical model within a 12-hour time limit. We solved the same problems with TS-VNS using the hyperparameter values mentioned above. The best and average values and also, a comparison with the GAMS-CPLEX results are given in Table 5. The gap is calculated as

$$\left(\frac{z^{TS-VNS}-z^{GAMS-CPLEX}}{z^{GAMS-CPLEX}}\right)100 \tag{15}$$

where z^{TS-VNS} and z^{GAMS} show the results obtained by TS-VNS and GAMS-CPLEX, respectively.

It is clear that the TS-VNS approach we proposed achieves successful results, especially as the problem size increases, it finds competitive results compared to GAMS-CPLEX. According to the best results TS-VNS found a better solution than GAMS-CPLEX for all problems of small-scale problems while for the average solution, it found better results for 3 out of the 4 small-scale problems. It is seen that the TS-VNS algorithm is more effective in terms of time. Addition, it should be noted, the best possible lower bound given by GAMS-CPLEX is 0 (zero) for all small-scale problems which means that the workload of all caregivers are equal. However, it is impossible to obtain the results that has the value of 0 (zero) since the problem is mixed-integer programming and the dual gap is highly possible.

Table 5. Comparison of the results obtained by the mathematical model-GAMS and TS-VNS

	pr-s-1	pr-s-2	pr-s-3	pr-s-4
GAMS-CPLEX (within 12 hours)	0.6	1.4	2.6	10.8
TS-VNS (Average)	0.9	0.7	1.3	2.3
Gap (%)	61.2	-49.2	-51.3	-78.8
TS-VNS (Best)	0.3	0.4	0.9	0.9
Gap (%)	-42.3	-69.6	-64.6	-91.3
TS-VNS time (min)	4.2	4.4	4.9	5.1

5.3. Large-Scale Test Problems

We solved the 20 large-scale problems in Table 3 with the TS-VNS algorithm. We summarize the results in Table 6. In addition, we have included the averages of the objective function values of the feasible initial solutions produced in each iteration. The difference between the values of the initial solutions and the final values shows the success of the TS-VNS algorithm in improving the solution.

Table 6. Results of large-scale test problems

Problem no	Initial solution	TS-VNS (Best)	TS-VNS (Average)	Time (min)
pr-l-1	201.1	4.1	5.1	40.3
pr-l-2	222.8	7.9	12.9	123.5
pr-l-3	220.3	8.0	11.9	128.7
pr-l-4	419.3	7.4	9.3	144.5
pr-l-5	257.5	4.4	6.7	199.8
pr-l-6	266.0	6.7	7.9	280.6
pr-l-7	159.2	6.2	7.4	52.7
pr-l-8	266.0	7.7	9.8	161.4
pr-l-9	277.8	5.9	7.0	168.2
pr-l-10	288.6	8.5	10.0	285.5
pr-l-11	232.7	1.1	1.2	48.3
pr-l-12	344.9	1.9	2.1	78.9
pr-l-13	333.3	3.9	4.8	101.9
pr-l-14	301.4	2.8	3.8	161.5
pr-l-15	264.3	2.7	2.9	229.6
pr-l-16	359.2	4.2	5.2	215.1
pr-l-17	212.1	1.6	2.2	61.1
pr-l-18	337.4	3.7	3.9	101.9
pr-l-19	298.3	3.7	4.5	194.4
pr-l-20	303.8	5.2	5.5	228.9

Finally, we made a comparison with large-sized problems to show that the balanced-insertion operator can achieve better results than the insertion moves with random selection. The numerical results are given in appendix Table A3. The graphs in Figure 7 visualize these numerical results. It can be seen from the graphs that the balanced-insertion approach yields better results. We use a t-test to test the significance of the difference between the two approaches. "H_0: The average of results for the two different insertion type is equal", and "H_1: The average of results for two different insertion type is not equal" are the hypothesis of t-test for both best and average results. The value of the test statistic is 2e-06 for best values and 8e-06 for average values. Both values are lower than the critic value at the 0.05 significance level, so the H_0 hypothesis is rejected. As a result, there is a statistically significant difference between the results of balanced-insertion and random insertion.

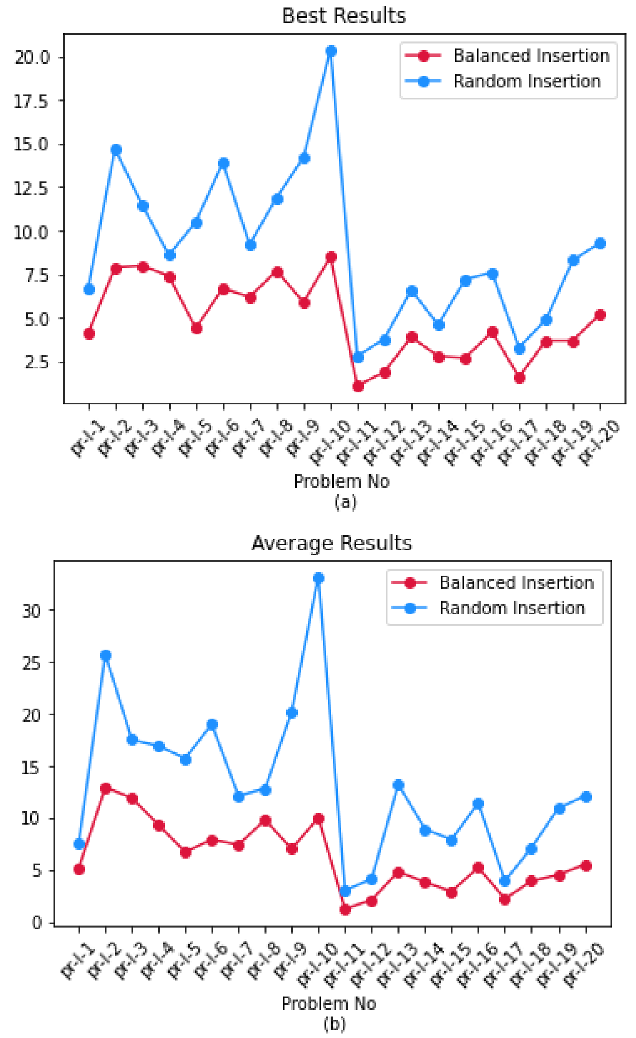


Figure 7. The comparison of balanced-insertion and random insertion: a) Best results of problems, b) Average results of problems

6. CONCLUSION

In this paper, we study the home healthcare routing problem for multiple hospitals with the objective of balancing workloads between hospitals. For this purpose, we propose a mathematical model based on a multi-depot vehicle routing problem with time windows. We show that when the balancing of the workload is not taken into consideration the differences between the workloads of the hospitals may be very large.

We generated small-scale problems and modified test problems from the literature for large-scale problems. Since the mathematical model can only solve the problem for up to 45 patients and 9 hospitals (small-scale problems), we use general-purpose metaheuristic algorithms to find good solutions in large search spaces. In the study, we proposed a hybrid algorithm (TS-VNS) that combines the strengths of two different metaheuristic algorithms. For small-scale test problems, we compared the results obtained by the TS-VNS algorithm and the mathematical model solved by GAMS. Solutions show that the TS-VNS algorithm is capable of finding good solutions. Furthermore, large-scale problems are solved by the TS-VNS algorithm within a reasonable time.

TS-VNS includes arrangements suitable for the representation of the problem, its purpose and time window constraints. In accordance with the objective function of the problem, the proposed "balanced-insertion" operator works better than the "random insertion" operator. In the study, results supporting that the algorithm has a successful search procedure in producing time-efficient and good solutions were obtained. In addition, TS-VNS has the flexibility of a general-purpose algorithm that can be easily applied to problems of similar nature.

The study deals with the home health care problem with the aim of balanced workload in terms of both service time and travel time. Unlike studies in the literature that have both purposes, it seeks a solution for multiple hospitals. It is also innovative in terms of the proposed hybrid metaheuristic approach.

For future research. Patients' hospital preference could be considered. Since every preference may not be acted upon, a second objective function is modeled as maximizing the patients' preference. Thus, a multi-objective mathematical model would be constructed and multi-objective solution methods, such as the weighted-sum method, would be needed to generate solutions.

Author contributions: Gülçin Dinç Yalçın; Literature Review, Mathematical Model, Developing Algorithm, Writing Manuscript- Tuğçe Yavuz; Developing Algorithm, Computational Results, Writing Manuscript-Şüheda Altıntaş; Literature Review, Mathematical Model.
Conflict of Interest: No conflict of interest was declared by the authors.
Financial Disclosure: The authors declared that this study has received no financial support.

REFERENCES

[1] 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, no. 3, pp. 962–976, Jun. 2006, doi: 10.1016/j.ejor.2005.01.011.

[2] "WHO's work on the UN decade of healthy ageing (2021-2030)". Accessed on: Feb. 13, 2023. [Online]. Available: https://www.who.int/initiatives/decade-of-healthy-ageing

[3] "Home healthcare market size, share & trends analysis report, 2022-2030". Accessed on: Feb. 13, 2023. [Online]. Available: https://www.researchandmarkets.com/reports/5450245/home-healthcare-market-size-share-and-trends

[4] E., Cheng, and, J. L. Rich, "A home healthcare routing and scheduling problem", 1998. Accessed on: Feb. 13, 2023. [Online]. Available: https://scholarship.rice.edu/handle/1911/101899

[5] S. Afifi, D.-C. Dang, and A. Moukrim, "Heuristic solutions for the vehicle routing problem with time windows and synchronized visits", Optim Lett, vol. 10, no. 3, pp. 511–525, Mar. 2016, doi: 10.1007/s11590-015-0878-3.

[6] 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. 13–14, pp. 1–22, Jun. 2017, doi: 10.1016/j.orhc.2017.06.001.

[7] M. Di Mascolo, C. Martinez, and M.-L. Espinouse, "Routing and scheduling in home health care: A literature survey and bibliometric analysis", Computers & Industrial Engineering, vol. 158, p. 107255, Aug. 2021, doi: 10.1016/j.cie.2021.107255.

[8] C. Fikar and P. Hirsch, "Home health care routing and scheduling: A review", Computers & Operations Research, vol. 77, pp. 86–95, Jan. 2017, doi: 10.1016/j.cor.2016.07.019.

[9] L. Grieco, M. Utley, and S. Crowe, "Operational research applied to decisions in home health care: A systematic literature review", Journal of the Operational Research Society, vol. 72, no. 9: pp. 1960-1991, 2021. https://doi.org/10.1080/01605682.2020.1750311

[10] D. Bredström and M. Rönnqvist, "Combined vehicle routing and scheduling with temporal precedence and synchronization constraints", European Journal of Operational Research, vol. 191, no. 1, pp. 19–31, Nov. 2008, doi: 10.1016/j.ejor.2007.07.033.

[11] 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, no. 2, pp. 48-58, June 2014, doi: https://doi.org/10.1016/j.orhc.2014.01.003 .

[12] G. Carello, E. Lanzarone, and S. Mattia, "Trade-off between stakeholders' goals in the home care nurse-topatient assignment problem", Operations Research for Health Care vol. 16, pp. 29–40, March 2018, doi: https://doi.org/10.1016/j.orhc.2017.12.002

[13] B. Yuan, R. Liu, and Z. Jiang, "A branch-and-price algorithm for the home health care scheduling and routing problem with stochastic service times and skill requirements", International Journal of Production Research, vol. 53, no. 24, pp. 7450–7464, Dec. 2015, doi: 10.1080/00207543.2015.1082041.

[14] K.-D. Rest and P. Hirsch, "Daily scheduling of home health care services using time-dependent public transport", Flex Serv Manuf J, vol. 28, no. 3, pp. 495–525, Sep. 2016, doi: 10.1007/s10696-015-9227-1.

[15] Y. Shi, T. Boudouh, and O. Grunder, "A hybrid genetic algorithm for a home health care routing problem with time window and fuzzy demand", Expert Systems with Applications, vol. 72, pp. 160–176, Apr. 2017, doi: 10.1016/j.eswa.2016.12.013.

[16] M. Masmoudi and N. Cheikhrouhou, "Heterogeneous vehicle routing problems with synchronization: Application to homecare scheduling routing problem". 9th Conference on Engineering and Management of Healthcare Systems GISEH, Geneva, 2018.

[17] 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", Flex Serv Manuf J, vol. 31, no. 4, pp. 989–1011, Dec. 2019, doi: 10.1007/s10696-018-9328-8.

[18] S. Bahadori-Chinibelagh, A. M. Fathollahi-Fard, and M. Hajiaghaei-Keshteli, "Two constructive algorithms to address a multi-depot home healthcare routing problem", IETE Journal of Research, vol. 68, no. 2, pp. 1108–1114, Mar. 2022, doi: 10.1080/03772063.2019.1642802.

[19] N. Tanoumand and T. Ünlüyurt, "An exact algorithm for the resource constrained home health care vehicle routing problem", Ann Oper Res, vol. 304, no. 1, pp. 397–425, Sep. 2021, doi: 10.1007/s10479-021-04061-9.

[20] Y. Li, T. Xiang, and W. Y. Szeto, "Home health care routing and scheduling problem with the consideration of outpatient services", Transportation Research Part E: Logistics and Transportation Review, vol. 152, p. 102420, Aug. 2021, doi: 10.1016/j.tre.2021.102420.

[21] H. Allaoua, S. Borne, L. Létocart, and R. Wolfler Calvo, "A matheuristic approach for solving a home health care problem", Electronic Notes in Discrete Mathematics, vol. 41, pp. 471–478, Jun. 2013, doi: 10.1016/j.endm.2013.05.127.

[22] P. Cappanera, M. G. Scutellà, F. Nervi, and L. Galli, "Demand uncertainty in robust Home Care optimization". Omega, vol. 80, pp. 95–110, October 2018, doi: https://doi.org/10.1016/j.omega.2017.08.012

[23] S. Frifita, M. Masmoudi, and J. Euchi, "General variable neighborhood search for home healthcare routing and scheduling problem with time windows and synchronized visits", Electronic Notes in Discrete Mathematics, vol. 58, pp. 63–70, Apr. 2017, doi: 10.1016/j.endm.2017.03.009.

[24] S. Lan, W. Fan, S. Yang, P. M. Pardalos and, N. Mladenovic, "A survey on the applications of variable neighborhood search algorithm in healthcare management", Annals of Mathematics and Artificial Intelligence, vol. 89, no. 8, pp. 741-775, 2021, doi: 10.1007/s10472-021-09727-5.

[25] E. Rahimian, K. Akartunalı, J. Levine, “A hybrid integer programming and variable neighbourhood search algorithm to solve nurse rostering problems”, Eur J Oper Res, vol. 258, no. 2, pp. 411–423, April 2017, doi: https://doi.org/10.1016/j.ejor.2016.09.030

[26] S. Riazi, O. Wigström, K. Bengtsson, and B. Lennartson, “Decomposition and distributed algorithms for home healthcare routing and scheduling problem”, in 2017 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), Sep. 2017, pp. 1–7. doi: 10.1109/ETFA.2017.8247622.

[27] 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, no. 1, pp. 127–137, Jan. 2019, doi: 10.1109/TASE.2018.2874392.

[28] S. E. Moussavi, M. Mahdjoub, and O. Grunder, “A matheuristic approach to the integration of worker assignment and vehicle routing problems: Application to home healthcare scheduling”, Expert Systems with Applications, vol. 125, pp. 317–332, Jul. 2019, doi: 10.1016/j.eswa.2019.02.009.

[29] F. Grenouilleau, A. Legrain, N. Lahrichi, and L.-M. Rousseau, “A set partitioning heuristic for the home health care routing and scheduling problem”, European Journal of Operational Research, vol. 275, no. 1, pp. 295–303, May 2019, doi: 10.1016/j.ejor.2018.11.025.

[30] 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, Mar. 2020, doi: 10.1016/j.cor.2019.104855.

[31] S. Shahnejat-Bushehri, R. Tavakkoli-Moghaddam, M. Boronoos, and A. Ghasemkhani, “A robust home health care routing-scheduling problem with temporal dependencies under uncertainty”, Expert Syst Appl, vol. 182, no. 115209, November 2021, doi: https://doi.org/10.1016/j.eswa.2021.115209

[32] L. Dekhici, R. Redjem, K. Belkadi, and A. Mhamedi, “Discretization of the firefly algorithm for home care”, Can J Electr Comput Eng, vol. 42, no. 1, pp. 20–26, 2016.

[33] M. R. Hassani, and J. Behnamian, “A scenario-based robust optimization with a pessimistic approach for nurse rostering problem”, J Comb Optim, vol. 41, no. 1, pp. 143–169, November 2021, doi: https://doi.org/10.1007/s10878-020-00667-0

[34] 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, no. 3, pp. 598–610, Jun. 2012, doi: 10.1016/j.ejor.2011.10.048.

[35] 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, Jun. 2012, doi: https://doi.org/10.1016/j.ejor.2011.10.042 .

[36] D. S. Mankowska, F. Meisel, and C. Bierwirth, “The home health care routing and scheduling problem with interdependent services”, Health Care Management Science, vol. 17, pp. 15-30, 2014, doi: https://doi.org/10.1007/s10729-013-9243-1 .

[37] K. Braekers, R. F. Hartl, S. N. Parragh, and, F. Tricoire, “A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience”, European Journal of Operational Research, vol. 248, no. 2, pp. 428-443, 2016, doi: 10.1016/j.ejor.2015.07.028

[38] A. Hertz and N. Lahrichi, “A patient assignment algorithm for home care services”, Journal of the Operational Research Society, vol. 60, no. 4, pp. 481-495, 2009, doi: https://doi.org/10.1057/palgrave.jors.2602574 .

[39] 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, June 2016, doi: https://doi.org/10.1007/s10696-015-9222-6 .

[40] S. Yalçındağ, P. Cappanera, M. Scutella, E. Şahin, and A. Matta, “Pattern-based decompositions for human resource planning in home health care services”, Comput Oper Res, vol. 73, pp. 12–26, September 2016, doi: https://doi.org/10.1016/j.cor.2016.02.011

[41] J. Decerle, O. Grunder, A. Hajjam El Hassani, and O. Barakat, “A hybrid memetic-ant colony optimization algorithm for the home health care problem with time window, synchronization and working time balancing”, Swarm and Evolutionary Computation, vol. 46, pp. 171–183, May 2019, doi: 10.1016/j.swevo.2019.02.009.

[42] M. I. Gomes, and T.R.P. Ramos, “Modelling and (re-)planning periodic home social care services with loyalty and non-loyalty features”, Eur J Oper Res, vol. 277, no. 1 pp. 284–299, August 2019, doi: https://doi.org/10.1016/j.ejor.2019.01.061

[43] A. Kandakoglu, A. Sauré, W. Michalowski, M. Aquino, J. Graham, and B. McCormick, “A decision support system for home dialysis visit scheduling and nurse routing”, Decis Support Syst, vol. 130, no. 113224, March 2020, doi: https://doi.org/10.1016/j.dss.2019.113224

[44] M. Yang, Y. Ni, and L. Yang, “A multi-objective consistent home healthcare routing and scheduling problem in an uncertain environment”, Computers & Industrial Engineering, vol. 160, no. 107560, Oct. 2021, doi: 10.1016/j.cie.2021.107560.

[45] H. Bae and I. Moon, “Multi-depot vehicle routing problem with time windows considering delivery and installation vehicles”, Applied Mathematical Modelling, vol. 40, no. 13, pp. 6536–6549, Jul. 2016, doi: 10.1016/j.apm.2016.01.059.

[46] E. Jabir, V. V. Panicker, and R. Sridharan, “Design and development of a hybrid ant colony-variable neighbourhood search algorithm for a multi-depot green vehicle routing problem”, Transportation Research Part D: Transport and Environment, vol. 57, pp. 422–457, Dec. 2017, doi: 10.1016/j.trd.2017.09.003.

[47] F. Glover, “Tabu search: Part I”, ORSA Journal on Computing, vol. 1, no. 3, pp. 190–206, Aug. 1989. doi: https://doi.org/10.1287/ijoc.1.3.190

[48] N. Mladenović and P. Hansen, “Variable neighborhood search”, Computers & Operations Research, vol. 24, no. 11, pp. 1097–1100, Nov. 1997, doi: 10.1016/S0305-0548(97)00031-2.

[49] J.-F. Cordeau, G. Laporte, and A. Mercier, “A unified tabu search heuristic for vehicle routing problems with time windows”, J Oper Res Soc, vol. 52, no. 8, pp. 928–936, Aug. 2001, doi: 10.1057/palgrave.jors.2601163.

[50] J. Bergstra, D. Yamins, and D. Cox. “Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures”, International Conference on Machine Learning, pp. 115-123, Feb. 2013, doi: https://doi.org/10.48550/arXiv.1209.5111

[51] J. Bergstra, R. Bardenet, Y. Bengio, and B. K´egl. “Algorithms for hyper-parameter optimization”, Advances in Neural Information Processing Systems 24, pp. 2546-2554, 2011.

Appendix

The data of the toy problem is given in the following tables. Hospitals and patients are represented as “H” and “P”. respectively.

Table A1. The time matrix between hospitals and patients

	H1	H2	H3	H4	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15	P16	P17	P18	P19	P20
H1	0	10.8	10.2	10.2	5.0	5.8	13.9	6.4	13.0	5.8	6.3	13.0	9.2	4.5	3.0	6.3	16.1	5.4	14.1	10.3	10.8	3.6	10.8	10.6
H2	10.8	0	8.1	8.1	15.6	12.0	4.1	17.2	7.2	16.3	11.0	2.2	2.0	9.4	13.4	13.9	5.4	16.1	6.4	11.0	15.3	14.2	10.0	2.8
H3	10.2	8.1	0	14.4	13.2	14.8	7.6	15.1	14.9	13.3	14.4	9.1	8.5	6.0	13.2	8.9	10.8	15.0	5.7	3.2	19.4	12.0	2.0	10.4
H4	10.2	8.1	14.4	0	14.8	7.1	12.1	15.7	3.6	15.8	5.7	9.5	6.4	12.2	11.2	16.0	12.2	13.9	14.4	16.6	8.1	13.6	16.1	5.4
P1	5.0	15.6	13.2	14.9	0	9.0	18.4	2.0	17.9	1.0	10.0	17.8	14.1	7.3	4.0	5.4	20.8	2.8	18.0	12.0	13.3	1.4	13.0	15.6
P2	5.8	12.0	14.8	7.1	9.0	0.0	16.0	9.2	10.6	10.0	1.4	14.1	10.0	9.9	5.0	12.1	17.3	7.3	17.3	15.6	5.0	8.1	15.8	10.4
P3	13.9	4.1	7.6	12.1	18.4	16.0	0.0	20.1	10.6	18.9	15.0	2.8	6.1	11.4	16.8	15.6	3.2	19.3	3.2	10.8	19.4	17.0	9.5	6.7
P4	6.4	17.2	15.1	15.7	2.0	9.2	20.1	0.0	18.9	2.2	10.4	19.4	15.6	9.2	4.5	7.3	22.5	2.0	19.9	14.0	13.0	3.2	15.0	17.0
P5	13.0	7.2	14.9	3.6	17.9	10.6	10.6	18.9	0.0	18.8	9.2	7.8	6.3	14.0	14.4	18.2	9.8	17.2	13.5	17.5	11.4	16.6	16.8	4.5
P6	5.8	16.3	13.3	15.8	1.0	10.0	18.9	2.2	18.8	0.0	11.0	18.4	14.9	7.6	5.0	5.1	21.4	3.6	18.4	12.0	14.3	2.2	13.0	16.4
P7	6.3	11.0	14.4	5.7	10.0	1.4	15.0	10.4	9.2	11.0	0.0	13.0	9.0	10.0	6.1	12.6	16.1	8.5	16.5	15.6	5.0	9.0	15.6	9.2
P8	13.0	2.2	9.1	9.5	17.8	14.1	2.8	19.4	7.8	18.4	13.0	0.0	4.1	11.4	15.7	15.8	3.2	18.4	5.8	12.2	17.1	16.4	11.0	4.1
P9	9.2	2.0	8.5	6.4	14.1	10.0	6.1	15.6	6.3	14.9	9.0	4.1	0.0	8.5	11.7	13.0	7.3	14.4	8.1	11.2	13.3	12.7	10.4	2.0
P10	4.5	9.4	6.0	12.2	7.3	9.9	11.4	9.2	14.0	7.6	10.0	11.4	8.5	0.0	7.3	4.5	14.1	9.0	10.8	5.8	14.9	6.1	6.3	10.4
P11	3.0	13.4	13.2	11.2	4.0	5.0	16.8	4.5	14.4	5.0	6.1	15.7	11.7	7.3	0.0	7.8	18.8	2.8	17.1	13.0	9.5	3.2	13.6	12.8
P12	6.3	13.9	8.9	16.0	5.4	12.1	15.6	7.3	18.2	5.1	12.6	15.8	13.0	4.5	7.8	0.0	18.4	8.1	14.4	7.1	17.0	5.0	8.2	14.9
P13	16.1	5.4	10.8	12.2	20.8	17.3	3.2	22.5	9.8	21.4	16.1	3.2	7.3	14.1	18.8	18.4	0.0	21.5	6.0	13.9	20.0	19.4	12.6	7.0
P14	5.4	16.1	15.0	13.9	2.8	7.3	19.3	2.0	17.2	3.6	8.5	18.4	14.4	9.0	2.8	8.1	21.5	0.0	19.4	14.3	11.0	3.2	15.1	15.6
P15	14.1	6.4	5.7	14.4	18.0	17.3	3.2	19.9	13.5	18.4	16.5	5.8	8.1	10.8	17.1	14.4	6.0	19.4	0.0	8.6	21.2	16.8	7.2	9.2
P16	10.3	11.0	3.2	16.6	12.0	15.6	10.8	14.0	17.5	12.0	15.6	12.2	11.2	5.8	13.0	7.1	13.9	14.3	8.6	0.0	20.5	11.2	1.4	13.2
P17	10.8	15.3	19.4	8.1	13.3	5.0	19.4	13.0	11.4	14.3	5.0	17.1	13.3	14.9	9.5	17.0	20.0	11.0	21.2	20.5	0.0	12.6	20.6	13.0
P18	3.6	14.2	12.0	13.6	1.4	8.1	17.0	3.2	16.6	2.2	9.0	16.4	12.7	6.1	3.2	5.0	19.4	3.2	16.8	11.2	12.6	0.0	12.0	14.2
P19	10.8	10.0	2.0	16.1	13.0	15.8	9.5	15.0	16.8	13.0	15.6	11.0	10.4	6.3	13.6	8.2	12.6	15.1	7.2	1.4	20.6	12.0	0.0	12.4
P20	10.6	2.8	10.4	5.4	15.6	10.4	6.7	17.0	4.5	16.4	9.2	4.1	2.0	10.4	12.8	14.9	7.0	15.6	9.2	13.2	13.0	14.2	12.4	0.0

Table A2. Caregivers and patients" time windows and operation times (in minutes)

No	e_i for "H" a_i for "P"	l_i for "H" b_i for "P"	D_i
H1	480	1080	–
H2	480	1080	–
H3	480	1080	–
H4	480	1080	–
P1	900	960	56
P2	720	1080	45
P3	480	540	28
P4	840	960	47
P5	900	1020	26
P6	540	960	54
P7	600	720	42
P8	840	1020	56
P9	660	720	39
P10	600	960	42
P11	780	1080	28
P12	660	840	21
P13	540	1020	45
P14	540	600	42
P15	480	540	35
P16	780	1080	55
P17	780	960	39
P18	600	1080	37
P19	480	660	20
P20	660	1080	56

Table A3. Comparison of insertion types: i) balanced-insertion, ii) random insertion

Problem no	Insertion type	TS-VNS (Best)	TS-VNS (Average)
pr-l-1	i	4.1	5.1
	ii	6.7	7.5
pr-l-2	i	7.9	12.9
	ii	14.7	25.7
pr-l-3	i	8.0	11.9
	ii	11.5	17.5
pr-l-4	i	7.4	9.3
	ii	8.6	16.9
pr-l-5	i	4.4	6.7
	ii	10	16
pr-l-6	i	6.7	7.9
	ii	13.9	19.0
pr-l-7	i	6.2	7.4
	ii	9.2	12.1
pr-l-8	i	7.7	9.8
	ii	11.9	12.8
pr-l-9	i	5.9	7.0
	ii	14.2	20.2
pr-l-10	i	8.5	10.0
	ii	20.4	33.2
pr-l-11	i	1.1	1.2
	ii	2.8	3.0
pr-l-12	i	1.9	2.1
	ii	3.8	4.1
pr-l-13	i	3.9	4.8
	ii	6.6	13.2
pr-l-14	i	2.8	3.8
	ii	4.6	8.9
pr-l-15	i	2.7	2.9
	ii	7.2	7.9
pr-l-16	i	4.2	5.2
	ii	7.6	11.4
pr-l-17	i	1.6	2.2
	ii	3.3	3.9
pr-l-18	i	3.7	3.9
	ii	4.9	7.1
pr-l-19	i	3.7	4.5
	ii	8.3	10.9
pr-l-20	i	5.2	5.5
	ii	9.3	12.1