



UWS Academic Portal

Homecare staff scheduling problem using a GA based approach with local search techniques

Sinthamrongruk, Thepparit; Dahal, Keshav; Satiya, Oranut ; Ricky, Michael Yosep; Smith, Lorraine

Published in:

9th International Conference on Cloud Computing, Data Science & Engineering (Confluence)

DOI:

[10.1109/CONFLUENCE.2019.8776958](https://doi.org/10.1109/CONFLUENCE.2019.8776958)

Published: 29/07/2019

Document Version

Peer reviewed version

[Link to publication on the UWS Academic Portal](#)

Citation for published version (APA):

Sinthamrongruk, T., Dahal, K., Satiya, O., Ricky, M. Y., & Smith, L. (2019). Homecare staff scheduling problem using a GA based approach with local search techniques. In 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 250-257). IEEE. <https://doi.org/10.1109/CONFLUENCE.2019.8776958>

General rights

Copyright and moral rights for the publications made accessible in the UWS Academic Portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy

If you believe that this document breaches copyright please contact pure@uws.ac.uk providing details, and we will remove access to the work immediately and investigate your claim.

“© © 2019 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

Sinthamrongruk, T., Dahal, K., Satiya, O., Ricky, M. Y., & Smith, L. (2019). Homecare staff scheduling problem using a GA based approach with local search techniques. In 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence) (pp. 250-257). IEEE. <https://doi.org/10.1109/CONFLUENCE.2019.8776958>

Homecare staff scheduling problem using a GA based approach with local search techniques

Thepparit Sinthamrongruk¹, Keshav Dahal¹, Oranut Satiya³, Michael Yosep Ricky¹³, Lorraine Smith¹

¹University of the West of Scotland, UK

²Phramongkutklo Hospital, Thailand

³Bina Nusantara University

ABSTRACT

Home health care service (HHC) provides daily medical services in patients' homes. The service aims to satisfy the patients' requirements, which must be done by one or more qualified staff visiting them within the right time and minimising operational cost. The homecare staff-scheduling problem (HSSP) is an extension of the HHC problem involving planning routes for caregivers visiting n patients. This paper proposes the route scheduling system based on a genetic algorithm approach for solving the HSSP. The focus is to introduce the novel techniques consisting of adaptive-opt (OPT), external k-nearest neighbour swapping (EKN), and internal nearest neighbour search (INN) to incorporate with the GA while also increasing the performance for solving the HSSP. In addition, the trade-off ratio between the percentage of improvement and the execution time is also concerned. The ratio provides the most suitable structure of the proposed GA that can continue to consider other issues for HHC service such as the dynamic scheduling that requires execution speed for re-calculating the service plan when some service activities are cancelled immediately.

Our empirical study reveals that the proposed techniques can produce improved solutions on two simulated datasets and an adapted case study instance compared to the original GA. (200 words)

KEYWORDS

Genetic algorithm, two-part chromosome, local search; homecare service, route scheduling

1 INTRODUCTION

Home healthcare service plays a fundamental and essential role in daily life. The service starts with the patients' requirements, which must be done by qualified caregivers. The staff members should possess the right skills and qualifications for the given task, for example, speaking languages, license to administer medicines, etc. The staff members referred to as caregivers are typically equipped with private cars or bikes, the use public transport or walking to patient' homes from the homecare office under a specific time window, usually between 08:00 to 16:30. This problem is referred to here as "Homecare Staff Scheduling Problem (HSSP)". The HSSP also includes

many factors such as task size, staff members' preferences, cost under specified time and capacity, along with other corresponding constraints. Due to the increase of elderly people in any areas already steering towards an aging society, HHC service providers are facing increasing operational costs for the service. Therefore, effective scheduling of the homecare service is essential to provide healthcare with minimum cost. In general, the experienced senior caregivers manually plan the less optimal schedules. Organisations providing the HHC service are either a charity service funded by the government or a private company having varying procedures for the HHC service in detail. However, the principal of the service procedure can be summarised firstly by the patients register and their requirements sent to the homecare office, then the sequence of service routes planned considering the information received before assigning caregivers to offer medical services at patients' homes in each service route. **Fig.1** illustrates a general overview of HSSP where there are five people, from *Patient-A* to *Patient-E*, who require home homecare service.

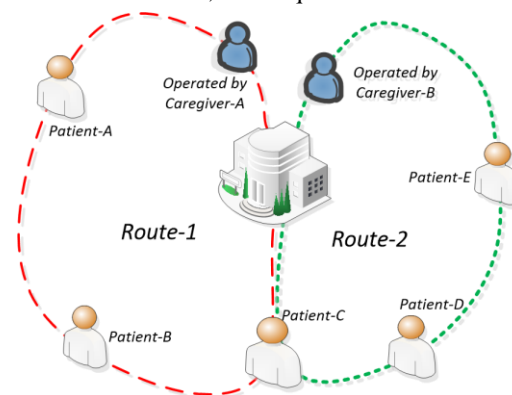


Fig. 1. Overview of homecare staff scheduling problem (HSSP)

In this paper, we focus on how to arrange effective routes for caregivers offering the HHC with minimum distance, which is the majority phase for solving the HSSP. To increase the performance of the method for planning the service solution, there are three local searches especially designed for HSSP, which consist of adaptive-opt(OPT), external k-nearest neighbour swapping (EKN) and internal nearest neighbour search (INN), which are introduced to incorporate into the GA. Also the proposed algorithms will apply to stimulate and adapt real-life situations for measuring its performance.

The rest of this paper is organised in the following sections. A literature review in section 2. Descriptions of the HSSP details, the proposed techniques and parameter settings are explained in sections 3. In section 4, I evaluate the proposed model and methods with simulated, adapt and real-world instances. The summary of results that the proposed method has efficiency for generating service schedule in practical in section 5.

2 RELATED WORK ON HSSP

HSSP has received intensive attention after the pioneering work in the UK from 1974 [1]. Since then, several techniques such as mathematical, heuristics and meta-heuristics have been investigated for solving HSSP in different regions and problem domains. As part of mathematical programming methods, Felici and Gentile [1] presented an integer programming model that maximises the total satisfaction of the nursing staff. Bard and Purnomo [2] adopted the column generation scheme to solve the problem regarding minimising the nursing staff members' violating preferences. The fuzzy theory was applied to a multi-objective integer programming model in order to determine the changeable factors that influence nurse satisfaction [3]. Bredström and Rönnqvist [4] proposed a mathematical programming model combined with a classic OPT to handle vehicle routing and nurse scheduling problem under time windows and additional temporal constraints. Experimental results illustrated a positive effect. Constantino et al. [5] proposed a new deterministic heuristic algorithm to solve a nurse scheduling problem consisting of two phases; a constructive scheduling phase and an improvement phase. A heuristic method to plan home healthcare service has been compared with the mixed integer linear programming based approach for caregivers routing in [6]. The heuristic approach is not able to reach the optimal solution for real problems, but the empirical results showed better computation times.

Genetic algorithm (GA) was applied in [7] and adaptive-penalty GA was employed in [8] to determine the optimal number of doctors. Harmony search, a metaheuristic algorithm, was used for the nurse scheduling problem in a hospital in Malaysia [9]. Artificial bee colony optimisation [10] was implemented for nurse scheduling problems. Particle-swarm-based approach (PSO) had been applied to generate a schedule for caregivers [11],[12]. Hiermann et al. [13] presented a general framework for solving homecare scheduling problems in Austria. This problem was generated by a random technique in the first phase before assigning four metaheuristics: variable neighbourhood search, a memetic algorithm (MA), scatter search and a simulated annealing hyper-heuristic to improve the solution. The results of this research show that the MA gives significantly better objective values overall for all instances compared to other methods.

A recovery scheme for the GA was added to tackle the more complex problem for HSSP. Chang-Chun Tsai et al.[14] proposed a new two-stage model for HSSP to reduce infeasible solutions in GA explicitly designed for our nurse preference scheduling. Braekers et al. [15] introduced a mixed model based on neighbourhood search heuristic. The aim of this research focuses on the multi-objective problem that concerns the trade-

off relationship between costs and client inconvenience regarding multiple travelling salesperson problems (MTSP).

Concerning research topics, the majority of research papers emphasises the single-purpose problem, such as Akjiratikarl et al. [11] developed a model for the vehicle routing problem. Bredström and Rönnqvist[4] focused on settings where the service required multiple-caregivers for operations simultaneously, while Trautsamwieser et al.[16] raised issues of mandatory working and staff's break regulations. Mutingi and Mbohwa[17] applied machine learning techniques to handle with preferences, workload balance of the service.

In this paper, we contribute a novel route scheduling approach based on the GA that exploits local searches designed particularly for HSSP for supporting the multiple caregivers' operation while minimising travel cost.

3 DESCRIPTIONS OF THE HSSP

3.1 Problem details and model

The problem description given below is derived from interviews with a homecare staff member at a hospital and adapted from [11].

- Caregivers are recommended to provide daily service between 4 to 6 patients as the higher number of patients may lead to fatigue and exertion.
- Each patient has the same priority which cannot demand specified service time. However, in some cases, i.e. patients with multiple disabilities can have a higher priority for the arrangement.
- Each caregiver starts and terminates from the homecare office at 08:00 and 16:30.
- The assigned jobs and each task must be completed under time slot windows.
- Geolocation (Latitude and Longitude coordination) defines the location of patient homes.
- Operating time is set at the default operating length of 1 hour.
- Caregivers are social workers focusing on the hygiene of food, medicine, accommodation and psychological counselling
- Lunchbreaks are taken into consideration. Time allocated is half an hour.

Firstly, we give the mathematical notations of the model:

$N = \{0, 1, 2, 3, \dots, n, n + 1\}$: The set of locations on the map, which includes the patients' locations and the homecare office. Usually 0 and $n+1$ are the index of the homecare office

i, j = the index of locations where $i, j \in N$ as well as $j \neq i$ and $j > i$

$S = \{1, 2, 3, \dots, s\}$: The set of caregivers

$R = \{1, 2, 3, \dots, r\}$: The set of service routes

C_{ij} = cost of travel between $location_i$ to $location_j$ depending on modes of transportation

$d_{i,j}^{r,d}$ = distance between $location_i$ to $location_j$ in $route_r$ on day_d

n = The number of tasks

r = The number of routes

D = The number of days for scheduling

$w_k^{r,d}$ = The working time at task k in minutes

$t_{i,j}$ = The travel time between $location_i$ to $location_j$ in minutes

TW = Time window

$$\text{Min}_{\text{travel cost}} \sum_{d \in D} \sum_{r \in R} \sum_{i, j \in N} P_{i,j}^{r,d} \cdot C_{i,j}^{r,d} \cdot d_{i,j}^{r,d} \quad (1)$$

Note that $d_{i,j}^{r,d}$ may be different from $d_{j,i}^{r,d}$

Subject to

$$P_{i,j}^{r,d} = \begin{cases} 1, & \text{if a journey is made from location } i \text{ to location } j \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

$$\sum_{r \in R} \sum_{i, j \in N} P_{i,j}^{r,d} = 1 \quad \forall d \in D \quad (3)$$

$$\sum_{j \in N} P_{0,j}^{r,d} = 1 \quad \forall r \in R, \forall d \in D \quad (4)$$

$$\sum_{j \in N} P_{j,(n+1)}^{r,d} = 1 \quad \forall r \in R, \forall d \in D \quad (5)$$

$$\sum_{i \in N} P_{i,h}^{r,d} - \sum_{j \in N} P_{h,j}^{r,d} = 0 \quad \forall r \in R, \forall d \in D \quad (6)$$

$$s_i^{r,d} + w_i^{r,d} + t_{i,j} \leq TW - 30 \quad \forall r \in R, \forall d \in D \quad (7)$$

$$w_j^{r,d} = (e_j^{r,d} - s_j^{r,d}) + TP K_j^{r,d} \quad \forall r \in R, \forall d \in D \quad (8)$$

The objective function (1) minimises the total of the travel cost, which includes the total of distances travelled by caregivers and the total travel cost varies in different modes of transportation. Constraints (2) are the binary constraints for variable $P_{i,j}^{r,d}$. Constraints (3) ensures that each patient is visited once only. Constraints (4) - (5) show that each route begins and ends at the homecare office. Constraint (6) confirms that each task location is visited and left while (7) guarantees that each job starts and terminates within a time window. Parking time is set as a constant value of 15 minutes while lunchtime is defined at 30 minutes. Constraints (8) estimates the working time of a task $w_i^{r,d}$ calculated by a finishing time $e_i^{r,d}$, starting time $s_i^{r,d}$, and parking time $TP K_i^{r,d}$ of task j . Note that in this study caregivers are homogenous. All caregivers earn daily pay and $w_j^{r,d}$ is set as a constant value at 75 minutes.

Note that the concept of dynamic scheduling and the requirement of multiple caregivers are not included in this paper.

3.2 Proposed route scheduling approach

The goal of the route scheduling approach is to create effective routes for caregivers to offer the homecare service, at different task locations, with the shortest route. In this section, the components of the proposed GA-based approach are described as follows:

3.2.1 Chromosome representation

Based on the work of A. E. Carter and C. T. Ragsdale[18], the planned solution is encrypted in a two-part chromosome representation which signifies possible sequences of service or solution. Fig. 2 shows an example of a chromosome containing three routes ($m=3$) where *Route-1* starts from the homecare office going to T_A, T_I, T_C, T_E, T_D , and then going back to the office. *Route-2* begins from the office going to T_K, T_G, T_L and going back to the office. The last one is *Route-3* starting at the office before offering services at T_B, T_J, T_F, T_H , and then going back to the office respectively while T_M and T_N are two

examples of overload task-locations, which cannot be included daily due to the time-slot constraint.

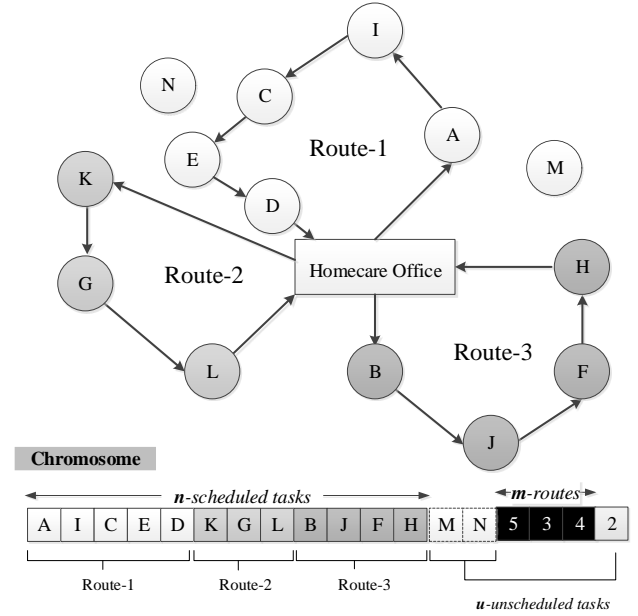


Fig. 2. Design of chromosome representation where $n = 12$, $u = 2$, and $r = 3$

3.2.2 Mating operators

Two-part crossover and modified two-part mutation are used to generate new offsprings. Crossover rate P_c is defined as the ratio of the number of genes that will be crossed over to the population size which ranges between $[0,1]$ while the mutation operator flips or alters one or more bit values randomly in a chromosome. Mutation rate P_m is defined as the ratio of the number of genes that will be mutated to the population size ranging between $[0,1]$. These operators assist the offspring exploration for searching the optimal global solution.

3.2.3 Parent selection

The roulette wheel is a popular parent selection procedure to choose some individuals to become parent chromosome members for producing offsprings. The probability of selection depending upon the total of distance travelled by caregivers.

3.2.4 Adaptive-GA

Two essential parameters: P_c and P_m are controlled in GA. Both parameters affect the quality of reproduction conducting to a variety of computational time significantly. They are usually set as constant values. In this paper, both are allowed to tune values while the GA runs automatically. Feedback from the fitness value is used to control P_c and adaptive P_m as shown in **Algorithm 1**.

Algorithm 1 AdaptiveParameter ($X_{best}(t)$, $X_{best}(t-1)$, $P_{original}$)

Input: Let $X_{best}(t)$ be the best solution of current generation
 Let $X_{best}(t-1)$ be the best solution of previous generation
 Rate $P_{original}$ and $P(t-1)$

Output: P_{new}

1. **if** EvalFitness($X_{best}(t)$) is better than EvalFitness($X_{best}(t-1)$)
2. $P(t) \leftarrow (P(t-1) + P_{original})/2$;
3. **else**
4. $P(t) \leftarrow P(t-1)$;
5. **end if**
6. **return** $P(t)$;

3.2.5 Adaptive-2OPT (OPT)

OPT is defined as the most important searching technique to improve the solution. The OPT starts by a random two pairs of edges, i.e., $\overline{T_A T_B}$ and $\overline{T_C T_D}$. If $\overline{T_A T_B}$ and $\overline{T_C T_D}$ are not adjacent edges, then pairs from $\overline{T_A T_C}$ and $\overline{T_B T_D}$ are removed and reconnected. The new planned service path is then recalculated. If an improvement is found, the solution is updated, otherwise it returns the original solution. The process iterates until the move limit is reached. Fig. 3 is an example of a service route of six patients' locations represented as connecting edges on a graph. The distance before and after improvement are 56 and 41 units respectively.

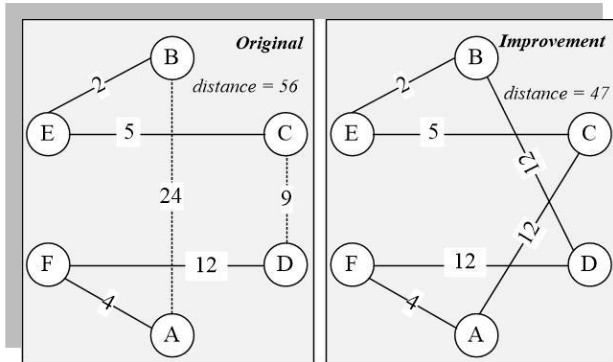


Fig. 3. An example of the OPT method

3.2.6 Internal nearest neighbour search (INN)

The INN is designed for optimising service path inside each route only. The INN is carried out by separating a chromosome into routes. In each route, the greedy search is used to re-arrange the route tasks. A greedy search firstly starts to find the nearest task from the homecare office and update the service sequences inside the route until all tasks are re-arranged. Fig. 4 illustrates an original service sequence of route-2 containing six service activities of cares: T_{14} , T_{12} , T_1 , T_8 , T_5 , and T_3 with an original cost of 140 distance-units and the improvement of service sequence starting at T_1 , T_8 , T_3 , T_5 , T_{14} , and T_{12} with 100 distance-units.

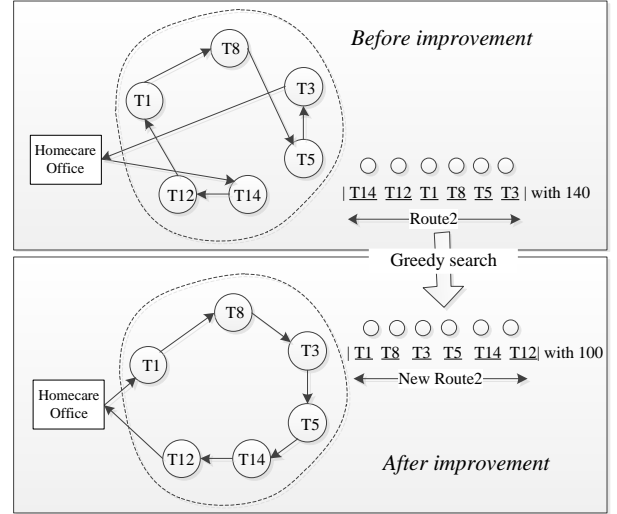


Fig. 4. INN enhances the sequence of solution

3.2.7 External k-nearest neighbour swapping (EKN)

The EKN is introduced to swap between k -task locations in different routes. Before the algorithm, we define two parameters: the rate k and the desired task location. The process begins by creating a set of k -pairs Pop_{cand} to store k -swapped solutions of care task locations between T_{target} of a route R_a and k -nearest locations outside R_a . The fitness of Pop_{cand} is recalculated and arranged in ascending order according to the objective function evaluation. The best fitness value is selected as an improved solution. Fig. 5 presents an example of EKN for swapping between T_{15} in route₃ and T_2 in route₁.

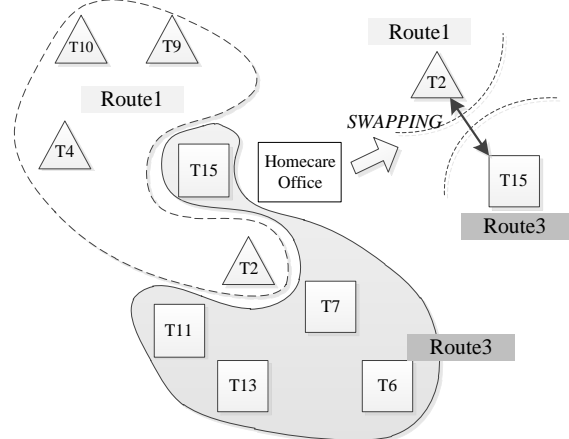


Fig. 5. KNN Swapping

Note that the aim of the proposed local searches contrasts from the mutation operator. The local searches aim to produce an enhancement to overcome the trapped area issue. The local searches are executed many times on the best solution of each generation until the improvement is found. If the enhancement is not met, the original solution is returned to the GA. In contrast, the mutation operator focuses on making the population diversified. This operator is run just once in each iteration and the reproduction of the mutation cannot turn back to the original.

4 EXPERIMENTAL RESULTS

4.1 Design of experiment

The proposed algorithms for solving HSSP was developed with R language on laptop Intel i5 systems with 8 GB of RAM, running at 1.7-2.5 GHz on Windows 10 64-bits OS.

We developed the hybrid GA based on the components provided in previous sections and then implemented the GA for the different combination of local searches to explore the results. **Algorithms 2** presents the general structure of the proposed hybrid GA. With the pseudo code, the GA is initially executed and then OPT, EKN, and INN are sequentially applied to improve the solution based on the designed structures varying the sequences of local searches from **S1** to **S16** before repeating the loop of GA. The designed structures are shown in Table 3.

Algorithm 2 HybridGAforHSSP ($Set_p, Set_s, P_c, P_m, P_{opt}, P_{EKN}$)

Input: Let Set_p be the patients' dataset containing n -task locations
 Let Set_s be the staffs' dataset
 Rate P_c of crossover
 Rate P_m of mutation
 Rate P_{opt} of OPT
 Rate k of EKN

Output: The optimal solution $X_{globalbest}$

1. $t \leftarrow 0$ #Initialisation
2. $Pop(t=0) \leftarrow$ CratePopulationSet(Set_p, Set_s)
3. $Pool(t) \leftarrow Pop(t)$
4. **while** $isNotTerminate$
5. **function** $Pop_{off}(t) \leftarrow$ ApplyGA($Fitness(t), Pool(t)$)
6. $Fitness(t) \leftarrow$ EvalFitness ($Pool(t)$)
7. $X_{best}(t) \leftarrow$ SelectBest($Fitness(t), Pool(t)$)
8. $Parent(t) \leftarrow$ ParentSelection($Fitness(t), Pool(t)$)
9. **while** $PopulationSetIsNotAchieved$ **do**
10. $Offspring(t) \leftarrow$ Crossover($Parent(t), P_c$)
11. $Offspring(t') \leftarrow$ Mutation($Offspring(t), P_m$)
12. $Pool(t+1) \leftarrow Pool(t+1) \cup Offspring(t')$
13. $X_{best}'(t) \leftarrow$ Apply(OPT(P_{opt}), EKN(k), INN) to $X_{best}(t)$
14. $[P_c, P_m, P_{opt}] \leftarrow$ AdaptiveParameter(P_c, P_m, P_{opt})
15. Update X_{global} with $X_{best}'(t)$
16. $t \leftarrow t + 1$;

4.2 Instances and parameters setting

The scheduling environment instances considered in this computational study is extracted from different sources: an adapted operational environment in a hospital, simulated instances and a modified instance from the public instance. The patient's location Set_p is provided in x and y-coordinates. The following list describes further details of all instances.

- **Instance1 (In1)** is simulated instances of 45 task-locations with 8 service routes ($n = 45, r = 8$)
- **Instance2 (In2)** is simulated instances of 65 task-locations with 12 service routes ($n = 65, r = 12$)
- **Instance3 (In3)** is an adapt instance of 104 task-locations with 20 service routes ($n = 104, r = 20$). This instance is adapted from xqf131-public instance[19] illustrating 131 points on the map. We programmed a random function to pick up 104 points from whole points to create the instance.
- **Instance4 (In4)** is modified data of a case study of 44 task-locations with 8 service routes ($n = 44, r = 8$). Patient's information such as name, address and types of disorders was anonymised before use in the experiment.

Table 1 shows the parameter values for all experiments performed in this paper. These parameters conduce to 576 experimental runs for each benchmark problem. The terminated criterion is the number of generations at 300. All caregivers start and end their service tours at the homecare office. Each structure is run with seeding for the benchmark problems.

TABLE 1. PARAMETER SETTING

Parameters	Level of parameters
Crossover rate (P_c)	0.5- 0.7
Mutation rate (P_m)	0.001 - 0.1
P_{OPT}	0 - 0.1
P_{EKN}	$K=0-5$
INN	Y:Enabled N:Disabled
Maximum iteration	300

4.3 Experimental results

According to (1), given any $c_{ij} = 1$. Performance of the proposed methods for solving the HSSP can be measured by using the best value and means of the total of distance travelled by caregivers visiting all n patients' locations for the daily service. Note that one caregiver performs each patient as well as one caregiver can do each service route. Based on the novel local searches mentioned in the previous sections, the experiments were conducted for 16 different combinations between the GA and the local searches as shown in Table3. Table 2 shows experimental results obtained for four HSSP instances run with different compositions.

For **In1** of 45 task-locations, with the goal of minimising total distance, the **S11** structure of the hybrid GA shows the best solution at around 111 km with *the size of population* = 60, $P_c = 0.7, P_m = 0.1, P_{2OPT} = 0.1, P_{EKN} = 5$ as well as INN enabled distance-units for **In1** of 45 task-locations.

For the higher scale of **In2** with 65 locations, the **S13** reveals superior solutions to its rival structures. The best solution gives 181.5 distance-units of the fitness value when *the size of population* = 40, $P_c = 0.5, P_m = 0.1, P_{OPT} = 0.1, P_{EKN} = 5$ with and INN enabled.

The best output of **In3** of 104 task locations is 326.9 km with *the size of population* = 60, $P_c = 0.7, P_m = 0.1, P_{OPT} = 0.1, P_{EKN} = 5$ and INN enabled based on the **S11**.

For a case study, the structure of **S11** gives the best solution quality (minimum travelling distance) for **In4** of the real case study tested when *the size of population* = 20, $P_c = 0.7, P_m = 0.1, P_{OPT} = 0.1, P_{EKN} = 5$ with INN enabled.

Table 3 provides experimental results regarding the average of the total of distances (mean) and the standard deviation (SD). The results of each structure (a group of samples) on each instance obtained from 36 independent runs in varying parameter adjusting. As seen from Table 3, **S11**: GA+OPT+EKN+INN with all location searches are implemented into the GA gives the lowest mean of the fitness value at 117.5, 345.3, and 504.2 for **In1**, **In3**, and **In4** respectively. Besides, **S13**: GA +EKN+OPT+INN shows the best means at 203.62 km. for **In2**.

TABLE 2. BEST RESULTS OF INSTANCES

Instance	Structure	Size of population	P_c	P_m	P_{OPT}	P_{EKN}	INN	Distance
<i>In1</i>	<i>S11</i>	60	0.7	0.1	0.1	5	Y	111
<i>In2</i>	<i>S13</i>	40	0.5	0.1	0.1	5	Y	181.5
<i>In3</i>	<i>S11</i>	60	0.7	0.1	0.1	5	Y	326.9
<i>In4</i>	<i>S11</i>	20	0.7	0.1	0.1	5	Y	454.5

TABLE 3. RESULTS OF THE STRUCTURES OF THE HYBRID GA FOR HSSP

Designed structures of hybrid GA	<i>In1</i>		<i>In2</i>		<i>In3</i>		<i>In4</i>		%Improvement (A)
	Mean Distance	SD.	Mean Distance	SD.	Mean Distance	SD.	Mean. Distance	SD.	
<i>S1</i> : GA	133.5	2.14	241.1	4.75	400.5	5.15	647.7	27.91	
<i>S2</i> : GA +OPT	121.3	4.41	205.2	9.34	358.6	6.15	508.8	30.25	14
<i>S3</i> : GA +EKN	131.3	1.54	237.6	4.54	396.6	5.62	656.4	21.01	0.7
<i>S4</i> : GA +INN	119.9	5.1	211.9	4	347.7	10.31	655.6	18.09	8.6
<i>S5</i> : GA +OPT+EKN	121.5	4.8	203.8	7.91	357.4	11.58	506.8	28.78	14.2
<i>S6</i> : GA +OPT +INN	119.7	4.42	205.9	8.14	356.5	17.73	508.8	33.12	14.3
<i>S7</i> : GA +EKN +INN	120	4.55	210.3	2.89	358.5	11.89	660	14.14	7.9
<i>S8</i> : GA +EKN +OPT	121.31	5.63	204.55	7.26	346.2	14.11	528.37	40.27	14.1
<i>S9</i> : GA +INN+EKN	120.12	4.82	209.65	2.78	350.45	9.13	590.06	14.58	11.1
<i>S10</i> : GA +INN +OPT	120.81	6.42	203.74	6.92	350.62	9.59	514.07	34.27	14.5
<i>S11</i> : GA +OPT+EKN+INN	*117.5	2.56	204	8.33	*345.3	6.07	*504.2	36.07	15.8
<i>S12</i> : GA +OPT+INN +EKN	118.64	6.13	204.68	7.21	346.5	12.99	539.31	33.87	14.1
<i>S13</i> : GA +EKN+OPT+INN	120.21	6.07	*203.62	7.84	346.43	15.99	529.53	49.19	14.3
<i>S14</i> : GA +EKN+INN+OPT	121.02	6.11	204.77	7.63	347.43	15.42	531.9	42.23	13.9
<i>S15</i> : GA+INN+OPT+EKN	120.28	5.45	205.62	7.09	347.59	15.58	539.42	38.59	13.6
<i>S16</i> : GA+INN+EKN+OPT	121.9	5.45	207.24	6.25	359.12	11.75	517.79	33.84	13.3

TABLE 4. RESULTS OF THE CPU TIME FOR ALL INSTANCES

Designed structures of hybrid GA	The means of CPU time				% Increasing (B)
	<i>In1</i>	<i>In2</i>	<i>In3</i>	<i>In4</i>	
<i>S1</i> : GA	45.7	79.3	195	44.1	
<i>S2</i> : GA +OPT	56.8	102	244	51.4	23.6
<i>S3</i> : GA +EKN	47.2	89.5	233	49.9	12.3
<i>S4</i> : GA +INN	69.3	128	319	51.1	48.2
<i>S5</i> : GA +OPT+EKN	65.1	108	240	54.6	31.5
<i>S6</i> : GA +OPT +INN	72.7	129	322	53.9	52.3
<i>S7</i> : GA +EKN +INN	74.6	144	362	54	63.5
<i>S8</i> : GA +EKN +OPT	73.5	145	326	56.5	59.7
<i>S9</i> : GA +INN+EKN	74.6	148	335	57.4	63
<i>S10</i> : GA +INN +OPT	69.1	139	308	52.9	51.1
<i>S11</i> : GA +OPT+EKN+INN	78.2	145	361	58.6	68.1
<i>S12</i> : GA +OPT+INN +EKN	78.4	147	371	57.3	69.5
<i>S13</i> : GA +EKN+OPT+INN	78	146	363	57.2	67.7
<i>S14</i> : GA +EKN+INN+OPT	78.4	147	367	56.8	68.6
<i>S15</i> : GA +INN+OPT+EKN	78.2	146	367	56.8	68.2
<i>S16</i> : GA +INN+EKN+OPT	74	139	346	52.9	58.7

Even though HSSP is not a real-time problem required solving immediately, computational time is limited. The HHC providers want the planned schedule as soon as possible. As mentioned before the CPU time is also extremely important for solving the HSSP. Table 4 illustrates experimental results concerning the means of the CPU time for all instances.

The trade-off ratio between the percentage of improvement (A) in Table 3 and the percentage of the additional CPU time

for execution in Table 4 (B) is summarised and shown on the boxplot in fig. 6 from *S2* to *S16*. The ratio on the graph ranges between [0,1]. The difficulty for analysing the trade-off is the fact that when all local searches are employed, the hybrid GA can produce superior improvement, however they consume a considerable number of the additional CPU time for execution.

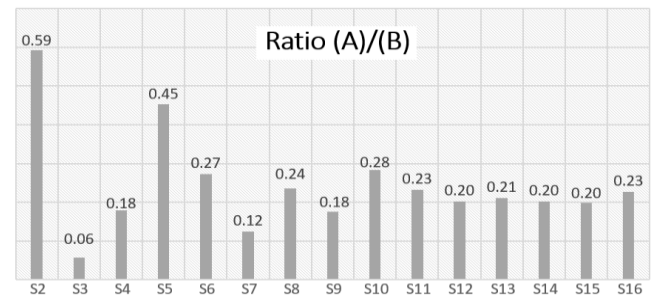


Fig. 6. The trade-off ratio between %improvement (A) and %increasing of CPU time (B) compared to *S1* (Higher is better)

Concerning the ratio, the best structure is *S2*: GA+OPT, which provides the highest ratio approximately at 0.59 obtained from all instances. This structure is recommended to apply in more complicated issues for HHC service such as dynamic scheduling requiring the speed for execution.

Table 5.1 – 5.4 show the optimal results of route scheduling with *S2* structure of the hybrid GA for all instances where *H* is the homecare office location and unsch-tasks is jobs that cannot be included daily due to the time-slot constraint and the minimum cost. The algebraic relationship for solving each instance is shown in fig. 7.

TABLE 5.1 THE OPTIMAL RESULTS OF ROUTE SCHEDULING WITH S2 STRUCTURE FOR INSTANCE 1

<i>In1</i>	
Minimum cost	115.74
CPUtime	6.39 s.
Route 1	H >> T14 >> T35 >> T13 >> T34 >> T11 >> H
Route 2	H >> T43 >> T25 >> T37 >> T19 >> T12 >> H
Route 3	H >> T24 >> T39 >> T7 >> T6 >> T8 >> H
Route 4	H >> T1 >> T22 >> T20 >> T38 >> T21 >> H
Route 5	H >> T28 >> T27 >> T42 >> T4 >> T3 >> H
Route 6	H >> T18 >> T17 >> T16 >> T15 >> T36 >> H
Route 7	H >> T23 >> T30 >> T9 >> T29 >> T10 >> T33 >> H
Route 8	H >> T40 >> T31 >> T2 >> T32 >> H
Unsch-tasks	H >> T5 >> T41 >> T26 >> T45 >> T44 >> H

TABLE 5.2 THE OPTIMAL RESULTS OF ROUTE SCHEDULING WITH S2 STRUCTURE FOR INSTANCE 2

<i>In2</i>	
Minimum cost	208.39
CPUtime	10.46 s.
Route 1	H >> T53 >> T60 >> T26 >> T25 >> T43 >> H
Route 2	H >> T56 >> T28 >> T6 >> T39 >> T24 >> H
Route 3	H >> T10 >> T47 >> T27 >> T9 >> T23 >> H
Route 4	H >> T30 >> T64 >> T15 >> T20 >> T21 >> H
Route 5	H >> T37 >> T44 >> T38 >> T16 >> T36 >> H
Route 6	H >> T4 >> T5 >> T52 >> T7 >> T8 >> H
Route 7	H >> T3 >> T45 >> T51 >> T40 >> T32 >> H
Route 8	H >> T34 >> T35 >> T14 >> T11 >> T13 >> H
Route 9	H >> T17 >> T18 >> T12 >> T63 >> T22 >> H
Route 10	H >> T50 >> T48 >> T29 >> T49 >> T33 >> H
Route 11	H >> T59 >> T46 >> T55 >> T19 >> T62 >> H
Route 12	H >> T1 >> T61 >> T31 >> T65 >> T2 >> H
Unsch-tasks	H >> T54 >> T57 >> T41 >> T42 >> T58 >> H

TABLE 5.3 THE OPTIMAL RESULTS OF ROUTE SCHEDULING WITH S2 STRUCTURE FOR INSTANCE 3

<i>In3</i>	
Minimum cost	350.42
CPUtime	23.11
Route 1	H >> T97 >> T22 >> T1 >> T50 >> T24 >> H
Route 2	H >> T14 >> T34 >> T79 >> T35 >> T87 >> H
Route 3	H >> T23 >> T64 >> T41 >> T51 >> T56 >> H
Route 4	H >> T7 >> T6 >> T8 >> T74 >> H
Route 5	H >> T33 >> T100 >> T57 >> T102 >> T75 >> T92 >> H
Route 6	H >> T61 >> T13 >> T71 >> T95 >> T11 >> H
Route 7	H >> T62 >> T90 >> T66 >> T80 >> T2 >> H
Route 8	H >> T89 >> T59 >> T55 >> T54 >> T70 >> H
Route 9	H >> T94 >> T43 >> T44 >> T46 >> T82 >> H
Route 10	H >> T49 >> T72 >> T36 >> T15 >> T63 >> H
Route 11	H >> T20 >> T78 >> T86 >> T38 >> T21 >> H
Route 12	H >> T77 >> T32 >> T40 >> T69 >> T103 >> H
Route 13	H >> T83 >> T12 >> T19 >> T26 >> T25 >> H
Route 14	H >> T67 >> T16 >> T17 >> T58 >> T88 >> H
Route 15	H >> T10 >> T81 >> T9 >> T68 >> T30 >> H
Route 16	H >> T37 >> T85 >> T99 >> T65 >> T84 >> H
Route 17	H >> T3 >> T45 >> T31 >> T76 >> T73 >> H
Route 18	H >> T27 >> T42 >> T5 >> T52 >> T96 >> H
Route 19	H >> T98 >> T4 >> T93 >> T91 >> T53 >> H
Route 20	H >> T39 >> T48 >> T28 >> T47 >> T29 >> H
Unsch-tasks	H >> T101 >> T60 >> T18 >> T104 >> H

TABLE 5.3 THE OPTIMAL RESULTS OF ROUTE SCHEDULING WITH S2 STRUCTURE FOR INSTANCE 4

<i>In4</i>	
Minimum cost	497.63 km.
CPUtime	6.44 s.
Route 1	H >> T35 >> T20 >> T42 >> T39 >> T18 >> H
Route 2	H >> T17 >> T4 >> T24 >> T37 >> T41 >> H
Route 3	H >> T22 >> T8 >> T16 >> T38 >> T29 >> H
Route 4	H >> T12 >> T13 >> T3 >> T21 >> T19 >> H
Route 5	H >> T5 >> T2 >> T30 >> T32 >> T36 >> H
Route 6	H >> T7 >> T15 >> T1 >> T9 >> T14 >> H
Route 7	H >> T10 >> T23 >> T11 >> T6 >> T34 >> H
Route 8	H >> T28 >> T33 >> T27 >> T31 >> T25 >> H
Unsch-tasks	H >> T26 >> T40 >> H

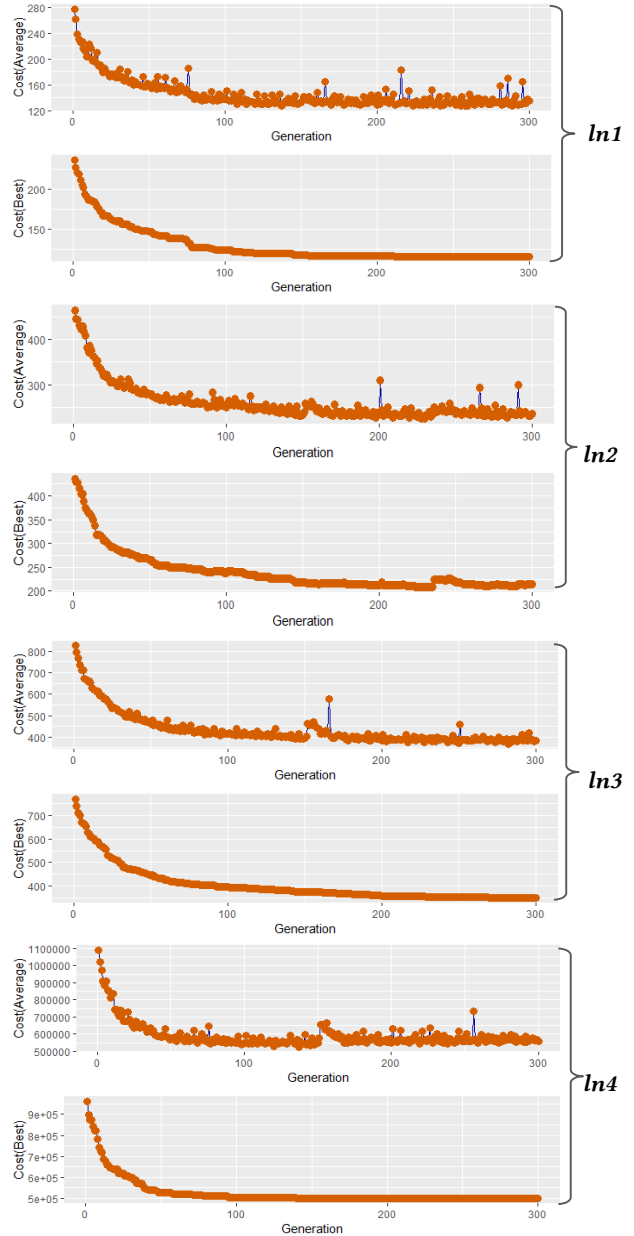


Fig. 7. Average and best fitness vs generation of S2 over 4 datasets

5 CONCLUSION

In this paper, a modified two-part chromosome representation is applied for encryption of the possible solutions involving s caregivers to offer homecare service for n patients. We also introduce a hybrid GA with three local searches including OPT, EKN and INN with the proper combination of strategy to enhance the performance of the original GA. Four instances consisting of two simulated cases; the adapted instance and the modified data of the case study were used to measure the performance of the proposed methods.

Our empirical study reveals that the *S11* gives the shortest service path with the total distance travelled by caregivers, compared to the original GA for three-four instances. Concerning the trade-off ratio between the percentage of improvement and the percentage of increased CPU time usage reveals that *S2* employed OPT only is the most suitable structure of the hybrid GA due to the greatest ratio.

This paper provides a potential basis for future research. On the one hand, we can continue to consider other details for HHC service such as dynamic scheduling and the requirement of multiple caregivers. This brings the model closer to the HHC practical activities. On the other hand, we may also try to design different algorithms such as clustering algorithms and parallel computing to obtain the optimal solution.

ACKNOWLEDGMENTS

This publication was made possible by grants from the EU Erasmus Mundus project SmartLink (552077-EM-1-2014-1-UK-ERA) to carry out this research at the University of the West of Scotland, UK.

REFERENCES

- [1] G. Felici, and C. Gentile, "A Polyhedral Approach for the Staff Rostering Problem," *Management Science*, vol. 50, no. 3, pp. 381-393, 2004.
- [2] J. F. Bard, and H. W. Purnomo, "Preference scheduling for nurses using column generation," *European Journal of Operational Research*, vol. 164, no. 2, pp. 510-534, 7/16/, 2005.
- [3] S. Topaloglu, and H. Selim, "Nurse scheduling using fuzzy modeling approach," *Fuzzy Sets and Systems*, vol. 161, no. 11, pp. 1543-1563, 2010/06/01, 2010.
- [4] 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, 2008.
- [5] A. A. Constantino, D. Landa-Silva, E. L. de Melo, C. F. X. de Mendonça, D. B. Rizzato, and W. Romão, "A heuristic algorithm based on multi-assignment procedures for nurse scheduling," *Annals of Operations Research*, vol. 218, no. 1, pp. 165-183, 2014//, 2014.
- [6] R. Redjem, and E. Marcon, "Operations management in the home care services: a heuristic for the caregivers' routing problem," *Flexible Services and Manufacturing Journal*, vol. 28, no. 1, pp. 280-303, 2016.
- [7] X. Cai, and K. N. Li, "A genetic algorithm for scheduling staff of mixed skills under multi-criteria," *European Journal of Operational Research*, vol. 125, no. 2, pp. 359-369, 9/1/, 2000.
- [8] T. Sinthamrongruk, and N. Harnpornchai, "Determination of optimal number of doctors and appointment interval for an appointment," *Information Engineering and Education Science, IRAICS Proceedings*, pp. 379-389: CRC Press, 2015.
- [9] M. Hadwan, M. Ayob, N. R. Sabar, and R. Qu, "A harmony search algorithm for nurse rostering problems," *Information Sciences*, vol. 233, pp. 126-140, 6/1/, 2013.
- [10] N. Todorovic, and S. Petrovic, "Bee Colony Optimization Algorithm for Nurse Rostering," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 43, no. 2, pp. 467-473, 2013.
- [11] C. Akjiratikarl, P. Yenradee, and P. R. Drake, "PSO-based algorithm for home care worker scheduling in the UK," *Computers & Industrial Engineering*, vol. 53, no. 4, pp. 559-583, 2007.
- [12] S.-C. Gao, and C.-W. Lin, "Particle Swarm Optimization Based Nurses' Shift Scheduling," *Proceedings of the Institute of Industrial Engineers Asian Conference 2013*, Y.-K. Lin, Y.-C. Tsao and S.-W. Lin, eds., pp. 775-782, Singapore: Springer Singapore, 2013.
- [13] G. Hiermann, M. Prandtstetter, A. Rendl, J. Puchinger, and G. R. Raidl, "Metaheuristics for solving a multimodal home-healthcare scheduling problem," *Central European Journal of Operations Research*, vol. 23, no. 1, pp. 89-113, 2015.
- [14] C.-C. Lin, J.-R. Kang, and T.-H. Hsu, "A Memetic Algorithm with Recovery Scheme for Nurse Preference Scheduling," *Journal of Industrial and Production Engineering*, vol. 32, no. 2, pp. 83-95, 2015/02/17, 2015.
- [15] 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.
- [16] A. Trautsamwieser, and P. Hirsch, "Optimization of daily scheduling for home health care services," *JAOR*, vol. 3, 2011.
- [17] M. Mutingi, and C. Mbohwa, "Multi-objective homecare worker scheduling: A fuzzy simulated evolution algorithm approach," *IIE Transactions on Healthcare Systems Engineering*, vol. 4, no. 4, pp. 209-216, 2014.
- [18] A. E. Carter, and C. T. Ragsdale, "A new approach to solving the multiple traveling salesperson problem using genetic algorithms," *European Journal of Operational Research*, vol. 175, no. 1, pp. 246-257, 2006.
- [19] U. o. Waterloo, "VLSI Data Sets - XQF131," University of Waterloo, 2018.