



# Scheduling of Home Health Care Services Based on Multi-agent Systems

Filipe Alves<sup>1</sup>(✉), Ana I. Pereira<sup>1,2</sup>, José Barbosa<sup>1</sup>, and Paulo Leitão<sup>1</sup>

<sup>1</sup> Research Centre in Digitalization and Intelligent Robotics (CeDRI),  
Instituto Politécnico de Bragança, Campus de Santa Apolónia,  
5300-253 Bragança, Portugal

{[filipealves](mailto:filipealves@ipb.pt), [apereira](mailto:apereira@ipb.pt), [jbarbosa](mailto:jbarbosa@ipb.pt), [pleitao](mailto:pleitao@ipb.pt)}@ipb.pt

<sup>2</sup> Algoritmi R&D Centre, University of Minho, Braga, Portugal

**Abstract.** Home Health Care (HHC) services are growing worldwide and, usually, the home care visits are manually planned, being a time and effort consuming task that leads to a non optimized solution. The use of some optimization techniques can significantly improve the quality of the scheduling solutions, but lacks the achievement of solutions that face the fast reaction to condition changes. In such stochastic and very volatile environments, the fast re-scheduling is crucial to maintain the system in operation. Taking advantage of the inherent distributed and intelligent characteristics of Multi-agent Systems (MAS), this paper introduces a methodology that combines the optimization features provided by centralized scheduling algorithms, e.g. genetic algorithms, with the responsiveness features provided by MAS solutions. The proposed approach was codified in Matlab and NetLogo and applied to a real-world HHC case study. The experimental results showed a significant improvement in the quality of scheduling solutions, as well as in the responsiveness to achieve those solutions.

**Keywords:** Multi-agent systems · Genetic algorithm  
Home Health Care · Optimization

## 1 Introduction

Over the last decade, Home Health Care (HHC) services have significantly increased [4, 10], namely in Portugal. The organization of HHC involves two main issues: medical decisions and organizational decisions. This paper considers the second issue and, more precisely, operational management problems related to human and material resources scheduling and optimization (considering, among others, vehicles and doctors). The scheduling and routing problems in this area are generally performed manually, are complex procedures, essentially because of its mathematical modelling complexity, and most importantly, unexpected events can occur resulting in uncertainty and disruptions requiring

real-time actions [15]. Typically, traffic jams, road accidents, and weather conditions should be taken into account when designing a real-world routing problem. In this context, HHC managers need decision support systems capable of solving large real-world situations in reasonable time and providing good solutions in a dynamic environment with numerous time constraints [12].

Multi-agent systems (MAS) [14] offers an alternative way to design and control systems, differing from the conventional approaches due to their inherent capabilities to adapt to emergence or disruptions without external intervention. MAS allow the decentralization of control over distributed structures, providing modularity, robustness and autonomy in systems [11] such as Health Care. On the other hand, Agent-Based Modelling (ABM) platforms are tools that allow the modelling, simulation or interactions of complex adaptive systems by using agents, providing a way to output the simulation results in a graphical manner according to several designed scenarios. These tools provide an easy and powerful simulation capability which enables a fast testing by using MAS [2]. In this context, systems, like HHC, can be viewed as a network of agents, e.g. vehicles or patients, with identifiable attributes, dynamic behaviour, sophistication in autonomous decision-making and alternatives to modify behavioural rules.

This paper addresses the routing problem in HHC, characterized by its dynamic context with uncertainties and random events. The proposed approach combines the optimization features provided by genetic algorithms, with the responsiveness features provided by MAS solutions. The agent-based model was implemented in the NetLogo tool that interact with the centralized genetic algorithm scheduling application codified in Matlab.

The rest of the paper is organized as follows: Sect. 2 describes the problem to be addressed and introduces the system architecture that combines MAS and genetic algorithms to improve the scheduling solutions for HHC services. Section 3 presents the problem formulation in MatLab and a brief description of the genetic algorithm. Section 4 describes the agent-based model, developed in NetLogo, for the dynamic decentralized solutions. Section 5 discusses the simulation results. Finally, Sect. 6 rounds up the paper with the conclusions and points out the future work.

## 2 Problem Description and System Architecture

This paper considers the problem of assigning tasks outside the Health Care Center of Bragança (HUB), Portugal, in particular, to find the schedule for home care visits for a certain day. The objective is to minimize the travel time spent by the vehicles during to perform all home visits. In this situation is not considered situations of external disruptions.

More concretely, it was considered the following assumptions, for the patients and vehicles:

- Each patient has his own location, i.e., his home.
- The time matrix of travel between all the localities is known in advance.

- The number of patients who need home care services, and assigned to a working day, is known in advance.
- The list and duration of the treatments are known for each patient.
- Caregivers that perform home care visits are known in advance.
- Each caregiver obtains a list of patients and a planned route from the HHC.
- All trips begin and end up at the Health Care Centre.

In this context, the system architecture for the HHC scheduling system integrates off-line and on-line modules, presented in Fig. 1, which can deal sequentially with the two sub-processes: optimized planning using genetic algorithms and real-time responsiveness solutions using MAS. The two modules are able to exchange information, balancing the decision-making according to the needs.

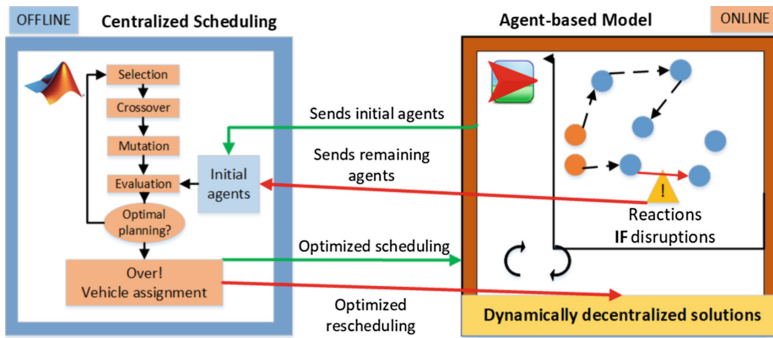


Fig. 1. Developed system architecture.

The left module performs the optimized scheduling for the vehicles routes, running off-line, for a given situation under study. The right module concerns to the dynamic re-scheduling, in response to disruptions or condition changes, e.g. a broken vehicle. In this module, the re-scheduling is obtained by the interaction of individual entities, each one reasoning about its own schedule.

### 3 Optimization of Centralized Scheduling

This section describes the approach embedded in the off-line module to perform the optimized scheduling.

#### 3.1 Problem Formulation in MatLab

Taking into account all the information, it was considered the follow parameters:

- $NP$  represents the total number of patients that need of home care visits.
- $NC$  is the total number of vehicles used for home care visits in a given day.

Consider the variable  $(np; nc) = (np_1, \dots, np_{NP}; nc_1, \dots, nc_{NP})$ , where the patient  $np_i$  will be visited by the vehicle  $nc_i$ , for  $i = 1, \dots, NP$ , and  $np \in \{1, \dots, NP\}^{NP}$  and  $nc \in \{1, \dots, NC\}^{NP}$ .

Then, for a given  $(np; nc)$  it is possible to define the route schedule and also the total time needed by each vehicle to finish its work defined in  $tt(np; nc)$ . So, the objective function is defined as

$$f(np; nc) = \max tt(np; nc) \quad (1)$$

which represents the maximum time spent by the vehicles to perform all visits and treatments by the nurses, including the returning journey to the Health Center. Then the constrained integer optimization problem will be defined as

$$\begin{aligned} \min & f(np; nc) \\ \text{s.t.} & 1 \leq np_i \leq NP, \quad i \in \{1, \dots, NP\} \\ & 1 \leq nc_j \leq NC, \quad j \in \{1, \dots, NP\} \\ & np_i \in \{1, \dots, NP\} \\ & nc_j \in \{1, \dots, NC\} \end{aligned} \quad (2)$$

where all the patients need to be treated  $\cup_{i=1}^{NP} np_i = \{1, \dots, NP\}$  and the vehicle  $nc_i$  needs to perform all the visits of the patients  $np_i$ , for  $i = 1, \dots, NP$ .

### 3.2 Genetic Algorithm

To solve the minimization problem presented in (2), a Genetic Algorithm was used. The Genetic Algorithm (GA) was proposed by Holland [8]. GA is a stochastic method, whose mechanism is based on simplifications of evolutionary processes observed in nature, namely selection, mutation and crossover [9]. As opposed to many other optimization methods, GA works with a population of solutions instead of one single solution [7]. The Algorithm 1 summarizes GA.

---

#### Algorithm 1. Genetic Algorithm

---

Generates a randomly population of individuals,  $\mathcal{P}^0$ , with dimension  $N_{pop}$ . Set  $k = 0$ .  
**while** stopping criterion is not met **do**  
    Set  $k = k + 1$ .  
     $\mathcal{P}' =$  Apply crossover procedure in population  $\mathcal{P}^k$ .  
     $\mathcal{P}'' =$  Apply mutation procedure in population  $\mathcal{P}^k$ .  
     $\mathcal{P}^{k+1} = NP$  best individuals of  $\{\mathcal{P}^k \cup \mathcal{P}' \cup \mathcal{P}''\}$ .  
**end while**

---

The iterative procedure terminates after a maximum number of iterations (number of generations) or after a maximum number of function evaluations [1, 3]. The information referred to above concerns centralized decisions, which will allow to calculate the optimized schedule that is passed to the on-line module.

## 4 Agent-Based Model

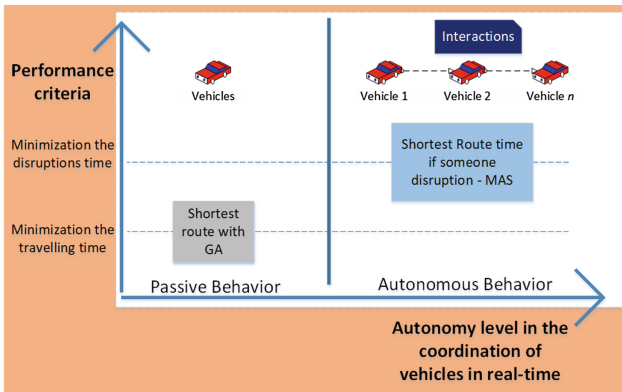
The on-line module considers the use of MAS to implement the dynamic and responsive re-scheduling in case of disruption. For this purpose, the agent-based model will be described in this section.

### 4.1 Behaviour of Agents

The agent-based model considers two types of agents:

- **Patient agent:** These agents represent HUB patients, providing the treatment needs to be performed. They are immobile, totally passive, and do not take the initiative to start the decision process.
- **Vehicle agent:** These agents represent the caregivers (1 caregiver uses 1 vehicle) that move around to provide the health care services at the patient agents location. Vehicles agents only interact with their own patients.

In this paper, only the global behaviour of vehicle agents will be described, since they are the ones that move and interact with the other agents. Figure 2 presents the two main categories of global vehicle behaviour: the passive and autonomous behaviours.



**Fig. 2.** Dynamic routing vehicle.

In the passive behaviour, the vehicles follow carefully the planned optimized route provided by the off-line module, i.e. using the GA codified in MatLab, without taking into account the disruptions or external problems. In the autonomous behaviour, the vehicle follows the planned route but is able to dynamically adapt the schedule in case of disruptions through the interaction with other vehicles to re-route the visits that were previously allocated to the broken vehicle.

## 4.2 Interaction Patterns in Autonomous Behaviour

In this work, only the dynamics of autonomous behaviour will be considered. Thus, although, initially, this dynamic involves passive behaviour, since it follows the planned route, it is subject to disruption tests or vehicle failures to evaluate interaction and autonomous cooperation between agents. This dynamic procedure involves decisions that are considered to help a vehicle in the event of disruption (which becomes unavailable) when choosing your help, in other words, the choice of another vehicle (available) that can take the remaining route of patients to visit. Each decision attempts to optimize the following criteria:

- The patients waiting time when a vehicle is interrupted on the route.
- Optimize travel time and its cost.

In this case, the vehicles follow the planned route defined by GA, when occurs a case of disruption in route, the procedure can opportunistically selects other available vehicle in on-line reactive manner (by comparing patients on routes), minimizing the waiting time to solve the problem and the time travelling. As depicted in Fig. 3, the initiator ask for available participants.

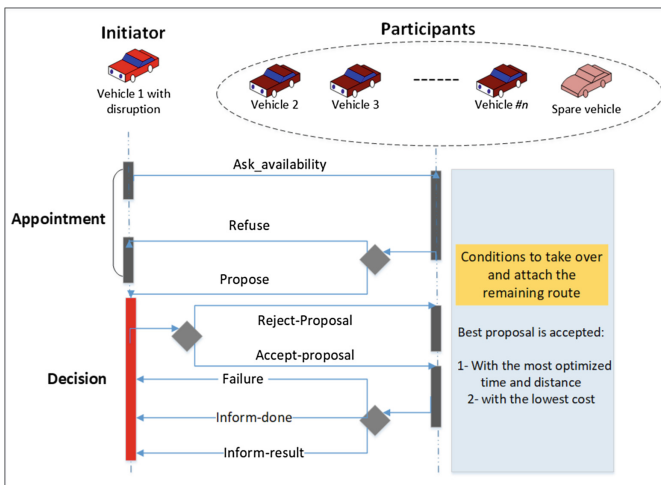


Fig. 3. Interactions and negotiations between vehicles.

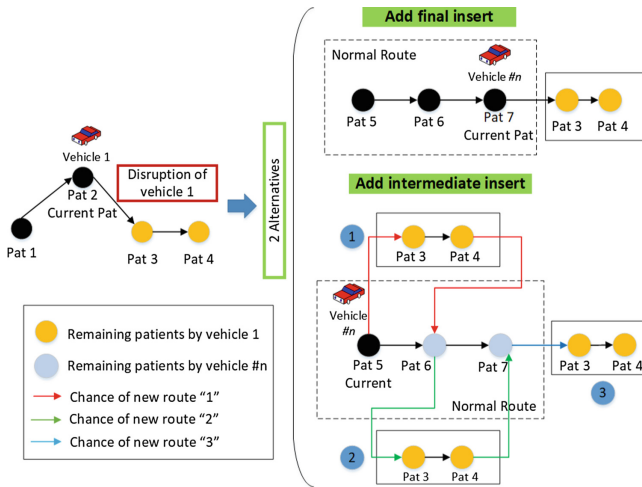
The Initiator calls the vehicles until it finds one or more available, in other words, the vehicle must be still on visits and be able until the care task is completed. As shown in the Fig. 3, the initiator scans the agents list and check if there are moving vehicles available to take over the remaining route. For each vehicle available, check time between the current or final patient in its route (depending on the alternative, which will be explained later) with the first patient in remaining route left by the initiator, where the total time for each available

participant will be calculated, to visit all the patients. Finally, when the best proposal with the most optimized time is found, the initiator decides and a participant will assume and update their route with the remaining route.

As previously mentioned, the autonomous behaviour includes the interaction with the vehicles and uses as a decision factor the comparison of the shortest distance and time between the remaining route and the next or final patient of the available vehicle route. It should be noted, that the remaining route is already optimized by the GA, so the comparison is done in a global way with the next patient of available vehicle. Therefore, this decision factor switches between the HHC goal (minimize the duration of travel and reduction costs) and the vehicles goal (minimize the time on travel).

For decision factor tests, two alternative methods (as shown in the Fig. 4) were compared:

- Put the remaining route in the final on the available vehicle;
- Put the remaining route in the middle on the available vehicle.



**Fig. 4.** Alternatives methods for decision factor.

Both alternatives use the decision factor, applied in different forms of comparison with the vehicle available. However the user will have the option to choose the alternative that it wants to show, depending on the needs, that is, may want to test the vehicle available only with the remaining route at the end (the shortest time, assumes the position), or test the remaining route at the various intervals in the route of available vehicle (tries to fit the remaining route in the range with the shortest distance and time). It should be noted that in the two alternatives, the code always look for the total route (vehicle route available

+ remaining route) with the most optimized time. As an example, the Fig. 4 shows that in the intermediate insertion method, the presented case may have three different routes, then the one that is more optimized will be choose.

### 4.3 Simulation Experiment Protocol

The described agent-based model was implemented in NetLogo [13], which is an agent-based modelling and simulation platform that allows to rapidly instantiate models to observe the behaviour agent-based systems. It provides an intuitive user interface where one can add buttons and control widgets to easily manipulate a model to view different scenarios [6].

Figure 5 presents the NetLogo interface for the implemented agent-based model.

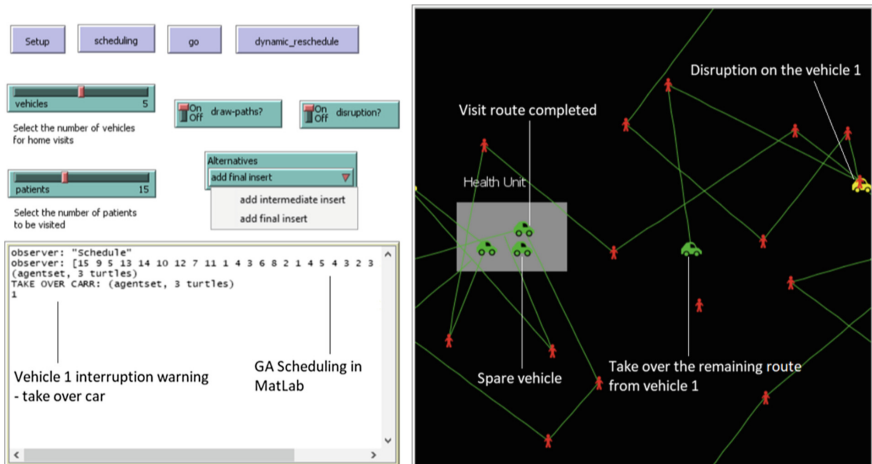


Fig. 5. Screen capture of the NetLogo interface.

As previously described, the agent-based model developed in NetLogo is connected with MatLab to allow the exchange of the optimized scheduling solution. It is often advantageous to implement separate portions of a model in the most appropriate language and to combine the results dynamically. The MatLab-NetLogo extension (MatNet) provides new functions within NetLogo that allow data passing between NetLogo and MatLab, and the calling of any valid, on-line MatLab commands from within NetLogo. The new tool presented herein facilitates future dynamic integration of these software platforms [5].

In each simulation, two parameters are used to vary the populations of the agents according to the database. For the case study, the simulation protocol involved three crucial steps, it can be applied to many other simulations within the database, thus:



- **Step 1:** The patients are generated from a random selection of the actual health centre database and their coordinates remained unchanged during the experiments. On the other hand, it is possible to select simulation with disruption (tested after certain ticks), or not, and which alternative to test.
- **Step 2:** The scheduling is performed by the user, which is transmitted data simulation and received the overall route of the GA for the vehicles planning (output viewer).
- **Step 3:** It is displayed the vehicle routes sequence and if there is disruption, we can update the dynamic reschedule by agents in their autonomous behaviour in coherence with the chosen alternative.

In conclusion, the trade-off between centralized decisions and dynamic interface provides easy visualization and handling for the independent user.

## 5 Discussion of the Simulation Results

This simulation tool was applied to test and experiment a case study presented in the data provided by the HUB. However only real locations and treatment times required by patients are used in order to confer the protection of confidential data. The Table 1 shows the parameters used in NetLogo and MatLab.

**Table 1.** Parameters for the study case

	Value
Number of vehicles agents	5
Number of patients agents	15
Number of localities	12
Number of treatments	5
Duration of treatments	[30,60,75,60,60]
Scene	50 $km^2$

For case study, simulations were carried out on a PC Intel(R) Core(TM) i7 CPU 2.2 GHz with 6.0 GB of RAM. The objective of this round of experiments (i.e., alternative #1 and alternative #2) was to highlight the efficiency of the proposed decision factor defined by the ability of the latter to minimize the vehicles travel time and automatically optimize disruptions or failures in vehicles.

GA, had 100% of successful rate since they found a feasible solution in all implementations. Initially the numerical result of the route obtained by the GA

**Table 2.** Route result of Netlogo output viewer and GA

Vehicles	Patients	Disruptions	Total Time (minutes)
Vehicle 0	{ 11, 7, 3 }		#
Vehicle 1	{ 13, 1 ~ 14, 10 }	X	#
Vehicle 2	{ 2, 5, 6, 12 }		#
Vehicle 3	{ 15, 8, 9, 4 }		#

in the case study, it is presented in Table 2 with only 4 vehicles, because the fifth is a spare vehicle, that is, it has no assigned route.

It is important to note that the scheduling of the total route obtained by the GA, and that will allow the vehicles agents to identify their own route, is obtained extremely fast, that is, it only took 24s. From the obtained route, which was tested with disruption, it is possible to verify that vehicle 1 was the one that suffered the stop or failure, leaving to visit the patients 14 and 10, respectively, that is, it will be our remaining route. The total time is not counted because there has been disruption of a vehicle and it is now expected that the autonomous behaviour of the agents can solve the problem. It should be noted again that the remaining route {14, 10} is already optimized by the GA so that it will be included in full and in the same order in another vehicle that may be available. A dynamic reschedule was then applied to each of the alternative methods developed. The Fig. 6 depicts the solutions of the obtained routes by each alternative.

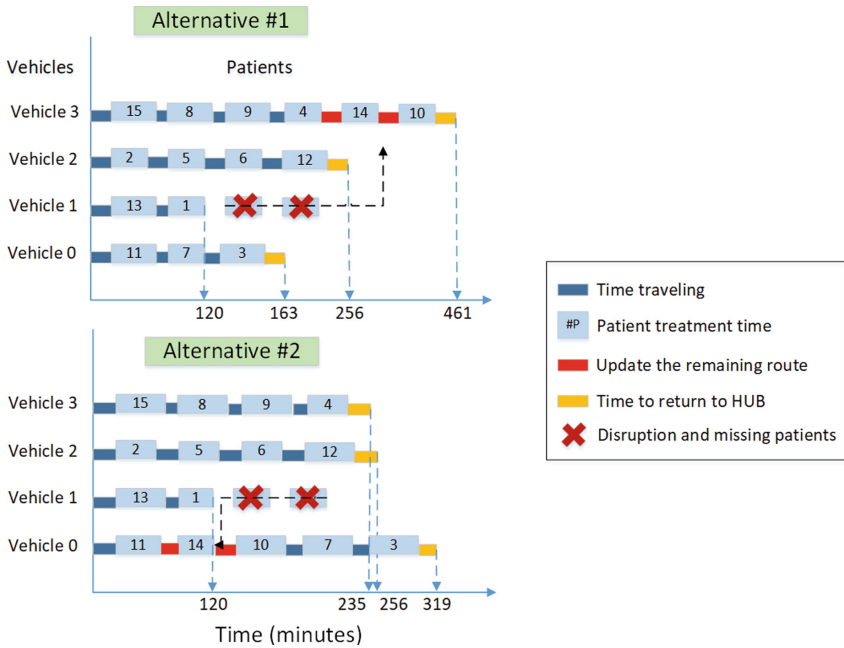


Fig. 6. Route simulation in the proposed alternatives.

Analysing Fig. 6, it is possible to see that the remaining route has been included in different vehicles depending on the alternatives. In Alternative 1, vehicle 3 was available and assumed the remaining route, including it at the end of its route, requiring 461 min to visit all patients and return to the origin point (HUB). In the same alternative, it was found that vehicle 1 only completed the

patients before disruption, accounting for only 120 min. The total spent time in minutes by all vehicles in this alternative was 1000 min. On the other hand, in alternative 2, vehicle 0 became available and took the remaining route, including it in its visits interval between patient 11 and patient 7, thus requiring 319 min to visit all patients and return to the HUB. Again, vehicle 1 only visited patients prior to disruption. Consequently, the total spent time in minutes by all vehicles in this alternative was 930 min.

This means that alternative 2 allowed significant reduction of 1 h and 10 min compared to alternative 1, producing an autonomous behaviour more organized and optimized than the first alternative. In this way, the model based on MAS combined with optimization features, allows to perform and visualize the optimum scheduling in a few seconds and still test disruptions in an autonomous and intuitive way for the user. The model also allows a decentralized strategy to support dynamic decision making of a complex problem.

## 6 Conclusion and Future Work

This paper addresses the operational-level HHC management problem in a dynamic context with numerous constraints, uncertainties, and random events. The system architecture has been proposed to solve the two sub-problems presented in a sequential manner. In this way, and in an attempt to optimize the process in case of disruption or external problems, it is necessary to use dynamics and strategies to minimize damage (such time spent) by each vehicle on home care routes, without, however, worsening the quality of the provided services.

The aim of this paper was to solve the routing problem (optimized scheduling by GA) in a dynamic context using an MAS based approach. For this, we developed and tested two alternatives to help vehicles on good decisions according to their level of autonomy and in a decentralized local context. The performance of these dynamic alternatives was assessed on a multi-agent platform. An experimental study was proposed based on a real case of a Health Unit in Bragança, that represented HHC services. It concludes that the alternative 2, is better than alternative 1, because it allowed to obtain the best time optimization proposal in the vehicle that assumed the remaining route as well as in the total time of all the vehicles in route. Therefore, it is verified that alternative 2 was the alternative that best simulated the dynamic behaviour among the agents, minimizing the response time to external disruptions.

Finally, it was demonstrated the pertinence, robustness, and implementation of distributed approach where the scheduling problem of vehicles was efficiently solved. Moreover, the created connection allowed to obtain a scheduling optimized by a meta-heuristic, in a fast way, with the possibility of interacting in an interface developed in NetLogo. In addition, it allows an independent user to use the platform to simulate multiple cases as well as the dynamic and autonomous interaction of the agents. The main contribution of this paper it was to provide a new way of solving a vehicle routing problem using the vehicles ability to dynamically design his own route in case of external disruptions.

For future work, it is possible to reformulate the problem and take into account a new alternative, which could contemplate a schedule to the remaining route and vehicles available by GA again. Another approach could be to use other meta-heuristics and increase the complexity of the problem.

## References

1. Alves, F., Pereira, A.I., Fernandes, F.P., Fernandes, A., Leitão, P., Martins, A.: Optimal schedule of home care visits for a health care center. In: Gervasi, O., et al. (eds.) ICCSA 2017. LNCS, vol. 10406, pp. 135–147. Springer, Cham (2017). [https://doi.org/10.1007/978-3-319-62398-6\\_10](https://doi.org/10.1007/978-3-319-62398-6_10)
2. Barbosa, J., Leitão, P.: Simulation of multi-agent manufacturing systems using agent-based modelling platforms. In: 9th IEEE International Conference on Industrial Informatics (INDIN), pp. 477–482 (2011)
3. Bento, D., Pinho, D., Pereira, A.I., Lima, R.: Genetic algorithm and particle swarm optimization combined with Powell method. In: AIP Conference Proceedings, vol. 1558, no. 1, pp. 578–581 (2013)
4. Benzarti, E., Sahin, E., Dallery, Y.: Operations management applied to home care services: analysis of the districting problem. *Decis. Support Syst.* **55**(2), 587–598 (2013)
5. Biggs, M.B., Papin, J.A.: Novel multiscale modeling tool applied to *Pseudomonas aeruginosa* biofilm formation. *PLoS ONE* **8**(10), e78011 (2013)
6. Chiacchio, F., Pennisi, M., Russo, G., Motta, S., Pappalardo, F.: Agent-based modeling of the immune system: NetLogo, a promising framework. *BioMed Res. Int.* (2014). <https://doi.org/10.1155/2014/907171>
7. Ghaheri, A., Shoar, S., Naderan, M., Hoseini, S.S.: The applications of genetic algorithms in medicine. *Oman Med. J.* **30**(6), 406 (2015)
8. Holland, J.H.: *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. MIT Press, Cambridge (1992)
9. Kumar, M., Husian, M., Upreti, N., Gupta, D.: Genetic algorithm: review and application. *Int. J. Inf. Technol. Knowl. Manag.* **2**(2), 451–454 (2010)
10. Nickel, S., Schrder, M., Steeg, J.: Mid-term and short-term planning support for home health care services. *Eur. J. Oper. Res.* **219**(3), 574–587 (2012)
11. Railsback, S.F., Grimm, V.: *Agent-Based and Individual-Based Modeling: A Practical Introduction*. Princeton University Press, Princeton (2011)
12. Rasmussen, M.S., Justesen, T., Dohn, A., Larsen, J.: The home care crew scheduling problem: preference-based visit clustering and temporal dependencies. *Eur. J. Oper. Res.* **219**(3), 598–610 (2012)
13. Wilensky, U., Evanston, I.: *NetLogo: center for connected learning and computer-based modeling*. Northwestern University, Evanston, IL 4952 (1999)
14. Wooldridge, M.: *An Introduction to Multiagent Systems*. Wiley, Hoboken (2009)
15. Yalçındağ, S., Matta, A., Şahin, E., Shanthikumar, J.G.: A two-stage approach for solving assignment and routing problems in home health care services. In: Matta, A., Li, J., Sahin, E., Lanzarone, E., Fowler, J. (eds.) *Proceedings of the International Conference on Health Care Systems Engineering*, pp. 47–59. Springer, Cham (2014). [https://doi.org/10.1007/978-3-319-01848-5\\_4](https://doi.org/10.1007/978-3-319-01848-5_4)