# Multi-objective Evolutionary Approach based on K-means clustering for Home Health Care Routing and Scheduling Problem

Mariem Belhor[1*,a,b,c], Adnen El-Amraoui[a], Abderrazak Jemai[b], François Delmotte[a]

[a] *Univ. Artois, Technoparc FUTURA, UR, 3926, 62400, Béthune, France*
[b] *Univ. Carthage, EPT, Marsa, 2078, Tunisia*
[c] *Univ. Manouba, National School of Computer Science, 2010, Manouba, Tunisia*

## Abstract

This paper suggests a bi-objective mathematical model, called HHC-MOVRPTW for the home health care routing and scheduling problem. HHC-MOVRPTW aims to minimize the overall service time while minimizing the total tardiness compared to the visiting time preferences. The considered problem is a NP hard, combining the Personnel Scheduling Problem and the Vehicle Routing Problem with time windows. Three solution approaches are proposed to solve it. Firstly, the HHC-MOVRPTW is solved with a non-scalar method, called Lexicographical method, in order to obtain a first solution for the problem. It is subsequently solved by two well-known multi-objective evolutionary algorithms, namely the Non-dominated Sorting Genetic Algorithm (NSGA-II) and the Strength Pareto Evolutionary Algorithm (SPEA2). Afterwards, a new hybridization approach, combined the evolutionary algorithm with K-means clustering technique is also suggested to improve the quality of the obtained Pareto sets. This is achieved by dividing the population of NSGA-II and SPEA-2 into sub-populations (clusters) and all sub-solutions have to be combined to find the final Pareto front. The computational experiments are performed using Solomon's bench- mark instances. Thus, results prove the effectiveness of the proposed approaches and their suitability with the problem.

*Keywords:* Home Health Care, Vehicle Routing Problem, Multiobjective Optimization, Evolutionary Algorithm, K-means Clustering

## 1. Introduction

Home Health Care (HHC) is defined as a set of caregivers providing a wide range of medical and paramedical services for patients in their homes such as nursing, drug delivery and medical assistance, etc. In France, HHC is gaining more importance within the European healthcare system. According to the official key numbers of Hospitalization, 128000 patients have benefited of HHC services and 5.9 million days of home hospitalization have been achieved in 285 HHC companies in 2019.

Reducing operating costs and providing high-quality of services are the main objectives of

[*] Corresponding author.
Email addresses: mariem.belhor@univ-artois.fr, mariem.belhor@gmail.com (Mariem Belhor), adnen.elamraoui@univ-artois.fr (Adnen El–Amraoui), abderrazak.jemai@insat.rnu.tn (Abderrazak Jemai), francois.delmotte@univ-artois.fr (François Delmotte)

HHC companies. Costs are mainly influenced by two parameters: the total travelled distance or time and the total time required by the HHC staff to perform medical tasks. Therefore, HHC companies face two problems, which are the staff scheduling and the routing problems. The combined routing and scheduling problem has become a relevant issue in several research fields, especially in HHC sector. Routing and scheduling problem are well known as combinatorial optimization problems. HHC decision-makers are confronted with multiple and complex decisions that are often contradictory, including the optimization of staff travelling distance and assignment of staff to patients. Several constraints are taking into account, such as, workload balancing and patient preferences, etc. Many studies have focused on healthcare optimization problems, in particular (Augusto & Xie, 2009) (Tlahig et al., 2009) (Ben Houria et al., 2016), etc. Inspired by this point, this work suggests a new bi-objective optimization model to the HHC routing and scheduling problem, called HHC- MOVRPTW. The considered problem is defined as a set of caregivers visiting patients at their homes to perform medical tasks with taking into consideration several constraints especially the visiting time preferences. This problem is considered as an extension of the famous vehicle routing problem with time windows. HHC-MOVRPTW aims to minimize the overall service time, required to perform all medical tasks by caregivers at homes of patients, while minimizing the total delays that represents the time difference between all services start- times and patients visiting-time preferences. In practice, both objectives tend to minimize operational costs and maximize the satisfaction of patients by arriving on time. Solving more than one objective consists of finding a set of good solutions (trade-offs), known as the set of Pareto-optimal solutions. In the literature, there are many works proposing exact and heuristic methods to solve multi-objective problems. In this paper, three solution approaches are proposed to solve HHC-MOVRPTW. First, a non-scalar method, lexicographic method is suggested in order to find a first solution for the problem. Then, two well-known multi-objective evolutionary algorithms, namely the Non-dominated Sorting Genetic Algorithm (Deb, 2001) and the Strength Pareto Evolutionary Algorithm (Zitzler et al., 2001) are also suggested to solve the proposed model with instances of large size, taken from the literature. The third solution approach is a hybridization between the two multi-objective evolutionary algorithms (NSGA-II and SPEA2) and the K-means clustering approach. The main idea of applying K-means clustering is to improve the quality of obtained Pareto sets in terms of convergence and diversity. the set of patients to be visited (input data) is divided into k similar clusters through K-means, where k is a predefined number of clusters, that corresponds to the number of caregivers. Evolutionary algorithms have to be simulated with each cluster data in order to obtain the optimal route of each caregiver. Afterwards, the obtained Pareto sub-set (partial solutions) have to be combined in order to form the global Pareto set.

This work is structured as follows. Section 2 summarizes the recent works on home health care problems and multi-objective optimization problems. The proposed optimization model is presented in section 3. Solution approaches are described in section 4. Section 5 provides and analysis the experimental results. Section 6 concludes the paper.

## 2. Literature Review

This section provides an overview of previous research and basic aspects on the home health care routing and scheduling problem, as well as the multi-objective optimization problems.

### *2.1. Home Health Care Routing and Scheduling Problem*

In home health care operations, a medical staff is scheduled and routed by planners to perform medical and paramedical services at patients' homes. This topic is called Home Health Care Routing and Scheduling Problem (HHCRSP). It is defined in the literature as a combination of two sub-problems, the vehicle routing problem and the personnel scheduling problem (Cheng & Rich, 1998) (Nickel et al., 2009) (Jemai et al., 2013) (Mankowska et al., 2014) (Yu & Gen, 2010) (Cococcioni et al., 2018) (Cholodowicz & Orlowski, 2017) (Mascolo et al., 2021). Such optimization problem has a significant impact since many decades, solving a HHCRSP consists of finding the shortest routes for a medical staff while respecting several operational constraints. However, the Vehicle Routing Problem (VRP) is a combinatorial optimization problem, proposed by Dantzig and Ramser in 1959. It consists of finding the optimal routes between a central depot and a set of customers, for a set of homogeneous vehicles. Each customer should be visited by exactly one vehicle and all vehicles must start and finish their routes at the depot (Bansal & Goel, 2018), (Elshaer & Awad, 2020). VRP was described as a generalized problem of travelling salesman problem (TSP) that consists of designing the shortest route for a salesman. The classical VRP has been extended in several variants by introducing additional real-life constraints. According to (Kumar & Panneerselvam, 2012) and (Shi-Yi & Wei-Chang, 2021), the most popular variants are VRP with Time Windows (VRPTW), VRP with Pickup and Delivery (VRPPD), Stochastic VRP (SVRP), the Multi-Depot VRP (MDVRP). Recently the Electric Vehicle Charging Scheduling (EVCS) is also considered among the most common variants as in (Liu et al., 2019). In VRPTW, each vehicle should deliver goods to customers within a specific time interval. In the literature, there are two types of VRPTW. VRPTW with soft time window (VRPSTW) where the time windows are allowed but associated with penalty costs (Li et al., 2015), (Iqbal et al., 2015) and VRP with hard time window (VRPHTW), in which any delay is allowed to the time window (Miranda & Conceição, 2016) (Hu et al., 2018). In the VRP with Pick-Up and Delivery (VRPPD), vehicles transport goods from a single depot to all customers and also from customers to the depot. Customers that receive goods from the depot are known as deliveries and customers who send goods are pickups. There are two sub-classes of VRPPD. Single demands, in which, customers are either pure deliveries or pure pickups (Nagy et al., 2015) and combined demands, where customers send and receive goods. (Kalayci & Kaya, 2016) (Avci & Topaloglu, 2016). The Stochastic VRP (SVRP) has one or several random components. it is an extension of VRP with stochastic (e.g. demand, customers, travel or service times) (Marinakis & Marinaki, 2015). In Multiple-Depots VRP (MDVRP), more than one depot is considered (Montoya-Torres et al., 2015). VRP and its variants have been widely studied and applied in many real-life applications, such as transportation, communication and healthcare, etc. (Elshaer & Awad, 2020), (Hameed, 2020) and (Hornstra et al., 2020). VRP is an NP-hard problem. Hence, exact methods are only efficient for small problem instances and may be difficult to solve these problems in accept- able process time. Therefore, Heuristics and metaheuristics are more suitable for real-world applications. On the other hand, Scheduling Personnel is a widely studying area and has an important role in manufacturing sector. It was introduced in the 1950s by (Dantzig, 1954) and (Edie, 1954). Personnel scheduling studies have focused on the assignment of tasks to employees, where the main objectives are to optimize the resource utilization and to ensure an equitably workload distribution. Scheduling problems are divided into static and dynamic problems. The Static scheduling cannot be changed over time. This may be different from employee to employee for each day of the week, but the shift start and end times are the same for each employee from week to week. While dynamic planning is variable over time (Özder et al., 2020). Personnel scheduling problems consist of various decisions that have to be taken, such as the assignment of tasks (e.g. employee A is assigned to job J). However, the group decision includes a number of different decisions where employees are combined into subsets, such as skills and different locations. For each decision, it is important to distinguish between individual personnel members and teams (Bergh et al., 2013). In HHC sector, personnel require transportation to perform services at different locations, such as caregivers visiting patients at their homes (Castillo-Salazar et al., 2012) (Castillo-Salazar et al., 2012) . These problems are referred in the literature as workforce routing and scheduling problems. Employees travel by diverse means of transportation, e.g. walking, car, public transport, bicycle, etc. Additionally, they have time window limitations regarding the daily workload or their starting and ending times (Castillo-Salazar et al., 2016). Many types of workforce scheduling and routing problem are discussed since the last decencies, whether for individual employees or teams (Gayraud et al., 2013) and (Issaoui et al., 2015) have focused on nurse scheduling problem while (Issabakhsh et al., 2018) and (Issabakhsh et al., 2018) have proposed optimization models for HHC staff (nurses, doctors, delivery men and technicians). Personnel scheduling is based on mathematical techniques and optimization approaches to improve resource allocation. The personnel scheduling problems are mainly solved by exact methods, heuristic and metaheuristic algorithms or both (Bergh et al., 2013). Unlike classical optimization approaches, the hybrid methods have achieved great success in a several areas due to their effectiveness in combining two or more components. The main objective of the hybrid approaches is to take advantage of the benefits of each component. In the literature, there are many hybrid approaches that have been proposed to deal with scheduling problems, such as in (Zhou et al., 2019), (Belhor et al., 2020), (Zhao et al., 2020), (Frazzon et al., 2018), (Zhao et al., 2021), (Arora & Banyal, 2022) (Belhor et al., 2023).

### *2.2. Multi-Objective Optimization Problem*

Modelling the appropriate mathematical formulation for an optimization problem is an important and critical step before optimization step itself (Branke et al., 2008; Coello et al., 2001). Optimization problem involves to find the best solution (the maximum or minimum value of some objective function) from all feasible solutions. Many decision and planning problems involve multiple conflicting objectives that should be considered simultaneously. Multi-objective optimization (MOO) involves more than one objective function to be simultaneously optimized. Solving a multi-objective optimization problem (MOOP) implies obtaining best solutions, so-called Pareto optimal solutions. MOOP received much attention in recent years due to their wide applicability in the most of real-world applications, especially in the HHC sector, where optimal decisions must be made, taking into account trade-offs between two or more conflicting objectives (Emmerich & Deutz, 2018). After finding the set of optimal solutions, HHC decision-makers choose the appropriate one according to their preferences (Pappalardo, 2008).

In the literature, many works have dealt with MOOP in HHC area. (Lin et al., 2015) have suggested a multi-objective optimization mathematical model for constructing weekly assignment for therapists, solved by an exact method. (Li et al., 2015) have proposed a multi-objective model for the HHCRSP which aims to maximize the satisfaction of care- givers and patients while minimizing the operating costs. (Braekers et al., 2016) have proposed a bi-objective optimization model for HHCRSP. They have solved the model by a new metaheuristic approach based on neighborhood search heuristic. (Alves et al., 2019) have worked on HHCRSP and proposed a multi-objective approach based on Tchebycheff method and a Genetic algorithm to solve the proposed model. (Decerle et al., 2019) have formulated a multi-objective model to deal with the HHCRSP, solved by a memetic algo- rithm. (Fathollahi-Fard et al., 2021) have suggested a multi-objective robust optimization of the HHC which is multi-depot, multi-period and multi-service. MOOP is based on the Pareto dominance concept, which consists of comparing two objective vectors in a precise sense (Cholodowicz & Orlowski, 2017). In MOOP, the concept of dominance is useful to compare two solution candidates $(x_1, x_2)$ and to determinate if one of these solutions is dominated by the other one. Researchers in MOOP aim to obtain good trade-offs so-called Pareto optimum (Coello et al., 2001; Király et al., 2011). Given two solutions $x_1$ and $x_2$ for a minimization problem. $x_1$ dominates $x_2$ denoted by $x_1, \succ x_2$, if and only if $x_1$ is better than solution $x_2$ in at least one objective.

$$\forall i \in \{1, \dots, M\}, f_i(x_1) \leq f_i(x_2) \;\; \wedge \;\; \exists j \in \{1, \dots, M\}, \; f_j(x_1) < f_j(x_2) \tag{1}$$

The non-dominated solutions are not dominated by any other solution. The Pareto set is a set of non-dominated solutions in the decision space. The Pareto front presents the image of Pareto set in the objective space. The selection of solutions is made according to the solutions ranking (e.g. objective 1 is more important than objective 2).

### 2.2.1. *Multiobjective Vehicle Routing Problem*

Multi-objective VRP (MOVRP) is a NP Hard problem in which at least two objectives have to be optimized. These objectives are often conflicting. Solving a MOVRP involves a set of non-dominated solutions. In the literature, there are three ways to formulated MOVRP. The first one consists of extending classical VRP by adding new objectives (Baños et al., 2013) (Chiang & Hsu, 2014). The second one is a generalization of VRP that consists of replacing some constraints by objective functions (e.g. the time window constraint is often replaced by an objective function) (Ayadi & Benadada, 2013), (Ghannadpour et al., 2014) and (Kaiwartya et al., 2015). While the third one defines new objectives from real-life cases such as in (Gupta et al., 2010), (Wang et al., 2016) and (Melián-Batista et al., 2014). There are two main approaches dealing with MOOPs. The first one is called conventional technique that converts the MOOP into single-objective and considers the rest of objectives as constraints (Jayamoorthi et al., 2017). While the second one is called evolutionary based technique that is well suited to solve several complex multi-objective problems. They generate non-dominated solutions after each iteration.

There is also another classification of solution approaches to solve MOVRP (Elshaer & Awad, 2020). The Pareto methods are the most popular and frequently used with evolutionary algorithms (Castro-Gutierrez et al., 2011), (Baños et al., 2013), (Chiang & Hsu, 2014). Scalar Methods consist of solving the MOVRP by an exact concept, such as weighted linear aggregation, ε-constraint methods, goal programming, etc. (Ghoseiri & Ghannadpour, 2010). The third category is called Non-Pareto and Non-Scalar Methods. These methods deal with several objectives separately. They are based on metaheuristic approaches such as Genetic Algorithms, Ant Colony etc. (Zou et al., 2013), (Chávez et al., 2016).

### 2.2.2. *Multiobjective Evolutionary Algorithms*

Evolutionary algorithms are shown as an important optimization and search methods in the last decade. They have a great success in solving widely used optimization problems and able to be successfully used in many applications of high complexity. Multiobjective Evolutionary algorithms (MOEAs) have the same success in both scientific research and engineering applications (Yu & Gen, 2010) (Castillo-Salazar et al., 2016). They are able to find several optimal solutions with a good spread in a single run of the algorithm.

The process of MOEA begins with a randomly initial population with fixed size. This population has to be evaluated by a fitness assignment value and efficient solutions have to be chosen after that. However, the reproduction of each solution is given by genetic operators especially the crossover and the mutation operators, in order to generate new offspring. At the final step, best solutions will be selected from the offspring and the parents. The MOEAs are mainly based on three components (Deb, 2001):

- The "fitness" assignment: allows better convergence towards optimal Pareto solutions. Among the best-known fitness assignment approaches, we can cite the "dominance-based approach".
- Diversity preservation: generate a diverse set of Pareto solutions. For example: the "nearest neighbor" method and the "Crowding Distance" method.
- Elitism: keeps the best solutions found during the search process.

In the literature, there are many MOEA algorithms, such as NPGA - Niched Pareto Genetic Algorithm (1994), NPGA II (2001), NSGA- Non-dominated Sorting Genetic Algorithm (1995), NSGA-II (2000), SPEA - Strength Pareto Evolutionary Algorithm (1998), SPEA 2 (2001), PESA - Pareto Envelope-based Selection Algorithm (2000), PESA II (2001) and MPGA - Multi-Population Genetic Algorithm (2003). A summary of these algorithms and a comparison study between them are provided by (Kunkle, 2005).

From the experiments conducted by (Zitzler et al., 2001) on various combinatorial and continuous optimization problems, it was proved that SPEA2 and NSGA-II achieved the best performance compared to other MOEAs. In addition, many works from various fields of application have also proven their effectiveness, including (Zitzler et al., 2001), (King et al., 2010), (Lee et al., 2011), (Bandyopadhyay & Bhattacharya, 2014), (Cholodowicz & Orlowski, 2017) and (Kaucic et al., 2019). However, there are similar recent works dealing with the problem of vehicle routing in the field of HAD, such as (Ait Haddadene et al., 2019) and (Xiang et al., 2021). In this paper, a new hybrid approach based on MOEAs is proposed to deal with the HHCRSP.

## 3. Problem Description and Mathematical Formulation

In this section, a bi-objective mathematical model, called HHC-MOVRPTW is proposed to deal with the HHCRSP that can be seen as a variant of VRP with soft time windows (VRPTW). The time window is defined by patients but caregivers can arrive before or after the visiting time preference with a penalty cost.

### *3.1. Problem Description*

For each HHC company, there is a set of caregivers $C = \{1, \dots, m\}$ available each day to perform specific tasks in patients' homes. The HHC-MOVRPTW is defined on a connected graph $G = (N, A)$, where $N = \{0, 1, \dots, n\}$ is a set of nodes. Node 0 represents the depot and nodes $N = \{1, 2, \dots, n\}$ represent the patients. And $A = \{(i, j) \mid i, j \in N, i \neq j\}$ is a set of arcs. Each arc $(i, j) \in A$ is associated with a travel time $T_{ij}$, measured in minutes. Care- givers are associated with homogenous vehicles (vehicles travel at the same speed and have the same capacity $Q$). Each caregiver visits a set of patients with maximum size $P$. Patients are equitably assigned between caregivers ($P = \frac{n}{m}$ if $n$ is an even number, otherwise $P = \frac{n}{m} + 1$). Where P refers to the maximum number of assigned patients. Each patient can be assigned to exactly one caregiver. Each patient requests a given quantity of drugs, denoted $q_i$. For each caregiver, the total quantity requested by his assigned patients should not exceed the capacity of his vehicle. All caregivers start and finish their services at HHC depot and they have the same daily workload (e.g. 5 working hours, from 8:00 am to 1:00 pm), denoted $V_c$. All caregivers visit a fixed number of sub-paths $B, (B = P - 1)$. Each patient $i \in N/\{0\}$ is associated with a service time $S_{ic}$, performed by caregiver c and served within a time window $[e_i, l_i]$. $e_i$ and $l_i$ represent successively the earliest and the latest service start time.

### *3.2. Mathematical Formulation*

To model the studied problem, three decision variables are defined. $X_{ic}$ is a binary scheduling variable. $\forall c \in C$, $\forall i \in N$, if node $i$ is assigned to caregiver $c, X_{ic} = 1$. Otherwise, $X_{ic} = 0$. The second decision variable is a binary routing variable, denoted $Y_{ijc}. \forall c \in C, i, j \in N$ if caregiver c travels from node $i$ to node $j$, $Y_{ijc} = 1$. Otherwise, $Y_{ijc} = 0. D_{ic}$ is the third decision variable that refers to the effective service start-time, performed by caregiver $c$ at node $i, \forall\, i \in N \setminus\{0\}, \forall\, c \in C$. HHC-MOVRPTW formulation is stated as follows:

$$\min Z_1 = \sum_{i,j \in N, i \neq j} \sum_{c \in C} \left(T_{ij} + S_{ic}\right). Y_{ijc} \tag{2}$$

$$\min Z_2 = \sum_{i \in N \setminus\{0\}} \sum_{c \in C} \left(\max\{(e_i - D_{ic}), 0\} + \max\{\left((D_{iw} + S_{ic}) - l_i\right), 0\}\right). X_{ic} \tag{3}$$

Subject to

$$\sum_{c \epsilon C} X_{0c} = m \tag{4}$$

$$\sum_{c \epsilon C} X_{ic} = 1, \forall\, i \in N \setminus\{0\} \tag{5}$$

$$\sum_{i \epsilon N \setminus\{0\}} X_{ic} \leq P, \ \forall\, c \in C \tag{6}$$

$$\sum_{i \epsilon N \setminus\{0\}} \sum_{c \epsilon C} \ Y_{i0c} = m \tag{7}$$

$$\sum_{j \epsilon N \setminus\{0\}} \sum_{c \epsilon C} \ Y_{ojc} = m \tag{8}$$

$$\sum_{c \in C} \ Y_{ijc} = 1, \forall\, i, j \in N \setminus\{0\}, \mathrm{i} \neq \mathrm{j} \tag{9}$$

$$\sum_{i,j \epsilon N \setminus\{0\}, \mathrm{i} \neq \mathrm{j}} Y_{ijc} \leq B \ , \forall\, c \in C \tag{10}$$

$$\sum_{i,j \epsilon N, i \neq j} \ (T_{ij} + S_{ic}). Y_{ijc} \leq \ V_c, \forall\, c \in C \tag{11}$$

$$\sum_{i \epsilon N} \ q_i. \ X_{ic} \leq \ Q, \forall\, c \ \in C \tag{12}$$

$$D_{jc} \ \geq \ (D_{ic} + \mathrm{T}_{ij} + S_{ic}). Y_{ijc}, \forall\, i, j \in N \setminus\{0\}, \mathrm{i} \neq \mathrm{j}, \forall\, c \ \in C \tag{13}$$

$$e_i \leq D_{ic} \leq l_i, \forall\, i \ \in N \setminus\{0\}, \forall\, c \ \in C \tag{14}$$

$$Y_{jic} - \ Y_{ijc} = 0, \forall\, i, j \in N, \ c \in C \tag{15}$$

$$X_{ic} \in \ \{0, 1\}, \forall\, i \in N, \forall\, c \ \in C \tag{16}$$

$$Y_{ijc} \in \ \{0, 1\}, \forall\, i, j \in N, i \neq j, \forall\, c \in C \tag{17}$$

$$D_{ic} \geq 0, \forall\, i \in N, \forall\, c \ \in C \tag{18}$$

HHC-MOVRPTW aims to optimize both objectives simultaneously. The first objective function aims to minimize the overall service time which include the travel times and the working times. The second objective function attempts to minimize the difference between start-time of services and visiting time preferences of patients. Constraint (4) guarantees that all caregivers visit the HHC depot (node 0). Constraint (5) ensures that each patient is assigned to exactly one caregiver. Constraint (6) forces each caregiver to be assigned to a limited and predefined number of patients. Constraint (7) and Constraint (8) guarantee that all caregivers start and finish their services at the HHC depot. Constraint (9) forces each caregiver to visit a given number of paths. Constraint (10) ensures that each caregiver crosses a path exactly once. Constraint (11) guarantees that caregiver's working time may not exceed the maximum workload. Constraint (12) ensures, for each caregiver, that the total quantity of drugs requested by his patients does not exceed his vehicle's capacity. For each two patients successively assigned to the same caregiver, constraint (13) ensures that the following service cannot begin until after the completion of the leading service. Constraint (14) guarantees that the service begins within an interval (the service cannot begin before the earliest start-time and after the latest start-time). Constraint (15) eliminates the sub- tours. Constraint (16) and Constraint (17) indicate that decision variables $X_{ic}$ and $Y_{ijc}$ are binary. Constraint (18) indicates that the effective service start-time $D_{ic}$ must be positive.

## 4. Solution Approaches

This section introduces the proposed solution approach. Three ways are suggested to deal with the HHC-MOVRPTW, as illustrated in Figure 1. The proposed solution approach is composed of three phases. In phase I, a non-scalar method, called lexicographic method, is suggested to solve the HHC-MOVRPTW.

In phase II, two well-known MOEAs, namely NSGA-II and the SPEA2, are implemented to solve the problem. Subsequently, each of these two MOEAs is combined with K-means clustering approach in phase III, to improve the quality of Pareto sets obtained in phase II. In fact, to solve a multi-objective optimization problem, there are two categories of resolution methods: conventional and evolutionary techniques. The lexicographic method belongs to conventional techniques that converts the MOOP into single-objective problem and considers the rest of objectives as constraints. It can be shown as an "a priori" approach in which the objectives are ranked according to the preferences of decision makers. The main advantage of this method is its simplicity and computational efficiency (Pinchera et al., 2017), (Cococcioni et al., 2018). While MOEAs belong to Pareto-based approach, that optimize simultaneously several objectives, often contradictory, by comparing and raking the solutions according to the concept of dominance.

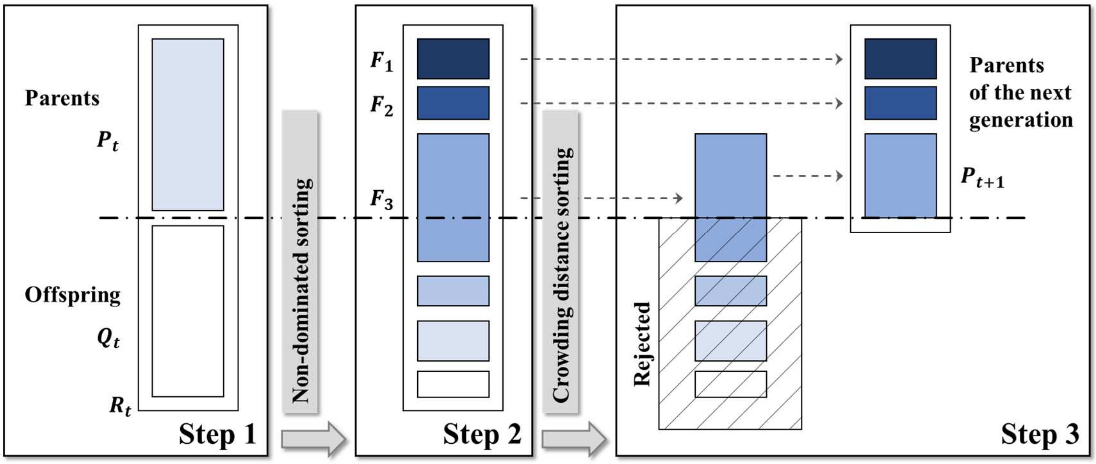


Figure 1: Structure of NSGA-II

### 4.1. *MOEA for HHC-MOVRPTW*

Based on the experiments carried out by (Zitzler et al., 2001) on various combinatorial and continuous optimization problems, it has been proven that SPEA2 and NSGA-II have achieved the best performance compared to other MOEAs. In addition, various related works have also proven their effectiveness, such as (Gadhvi et al., 2016), (Cholodowicz & Orlowski, 2017), (Kaucic et al., 2019), (Ait Haddadene et al., 2019) and (Xiang et al., 2021). For these reason, NSGA-II and SPEA2 are chosen to solve the HHC-MOVRPTW.

#### 4.1.1. *Non-dominated Sorting GA*

The non-dominated sorting GA (NSGA-II) was firstly proposed by (Deb, 2001). It is an improved version of non-dominated sorting GA (NSGA), proposed by (Srinivas & Deb, 1995). NSGA-II has successfully addressed multi-objective optimization on several benchmark problems. NSGA-II decomposes the population into Pareto fronts, using the notion of dominance, to classify the solutions. Each front is composed of solutions having the same rank. The first one is composed of all the non-dominated solutions, the second front contains the solutions which are only dominated by the solutions of the first front and the last front contains the solutions dominated by all other solutions of other fronts. This step is necessary to select the individuals for the reproduction step, which will allow to proceed to the next iteration of the algorithm. The popularity of NSGA-II is related to its computational complexity which is lower than that of NSGA: $O(MN^2) < O(MN^3)$, where M corresponds to the number of objectives and N is the size of the population (Deb, 2001). In the first generation, NSGA-II randomly generates a population of a fixed size. This population is composed of a set of chromosomes, each of which represents a solution to the problem. In this work, the chromosome is represented by a numerical vector, where each number inside corresponds to a given patient. The chromosome is consisting of the routes that vehicles must travel. In the present work, the chromosome representation is inspired by the multi-chromosome technique of (Király et al., 2011) as shown in Figure 2. Due to the problem constraints, each gene should appear exactly once in the chromosome. However, all routes must start and end at the depot (node 0). Thereby, all genes (0) are excluded from the chromosome, as shown in Figure 2, but they are taken into account when calculating the fitness function. Due to the workload balancing constraint, and due to the workload balancing constraint, patients have to be fairly divided between caregivers.

In each generation $t$, the offspring population called auxiliary $Q_t$ is first created by using the parent population $P_t$ and genetic operators. After that, the two populations are combined together to form a new population. Solutions on the first front $(F_1)$ are better than solutions on the second front $(F_2)$. In this case, the rank of the first front is better than that of the second one $(R_1 < R_2)$ Solutions having the same rank are evaluated using the "Crowing distance". Given m objective functions, the crowding distance of each objective is calculated by equation (19).

$$CD_{im} = \frac{f_m(x_{i+1}) - f_m(x_{i-1})}{f_m(x_{max}) - f_m(x_{min})} \tag{19}$$

The crowding distance for each solution $i$ is calculated as following:

$$CD_i = \sum_{m=1}^{M} CD_{im} \tag{20}$$

Given two solutions $i$ and $j$, $i$ has a rank $R_i$ and j has a rank $R_j$. Solution $i$ is preferred to solution $j$ if:

$$R_i < R_j \quad OR \quad (R_i = R_j \quad AND \quad CD_i > CD_j) \tag{21}$$

The boundary values of each front have the infinite distance value $(\infty)$. The fronts are sorted in ascending order and only the first $N$ solutions are kept for the next generation. The selection criterion is based on the comparison operator which requires both the rank and the $CD$. NSGA-II repeats these steps until a stopping criterion is satisfied (usually, maximum number of generations).

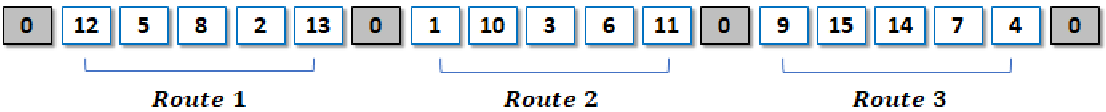


Figure 2: chromosome representation

### 4.1.2. *Strength Pareto Evolutionary Algorithm*

The Strength Pareto Evolutionary Algorithm 2 (SPEA2) was proposed by (Zitzler et al., 2001), extended of SPEA (Zitzler & Thiele, 1999). The SPEA2 starts with an initial population with fixed size and an empty archive (external set). Then, it separates the non-dominated set from the population solutions. The fitness of each solution is calculated according to the number of dominated and dominating solutions. It copies the front of non- dominated solutions in the archive and generates new solutions using the genetic operators. Finally, it selects the best candidates according to their density values. Each solution $i$ belongs to the population $P_t$ and the archive $Q$ has a strength value $S(i)$, represented by the number of dominating solutions (Zitzler et al., 2001). The strength value is calculated as following:

$$S(i) = |\{j \mid j \in P_t + Q \wedge i \succ j\}| \tag{22}$$

Where $|.|$, $+$ and $\succ$ denote respectively to the cardinality of a set, multiset union and the Pareto dominance relation. The raw fitness $R(i)$ of each individual $i$ is calculated as following:

$$R(i) = \sum_{j \in P_t + Q, j \succ i} S(j) \tag{23}$$

If $R(i) = 0$, the individual $i$ is a non-dominated solution and if individual $i$ has a high $R(i)$ value, it means that $i$ is dominated by many individuals. The fitness value of each individual $i$ is calculated by equation (23):

$$F(i) = R(i) + D(i) \tag{24}$$

Where $D(i)$ corresponds to the density of individual $i$ and defined as following:

$$D(i) = 1/\left(\sigma_i^k + 2\right) \tag{25}$$

For each individual $i$, a list containing the distances between this individual and all other individuals is defined and ordered in increasing way. The $k^{th}$ element of this list is denoted by $\sigma_k$. The selection process starts with copying the non-dominated solutions which have $[F(i) < 1]$, from the present population and archive list to the archive of the next generation. The process is stopped when the number of non-dominated solutions is equal to the archive list size. Otherwise, if the non-dominated front is too large compared to the archive, the individuals of the next generation are deleted iteratively by choosing at each iteration the individual who has the minimum distance from another individual. This procedure ends when the non-dominated front fits exactly into the archive. When the non-dominated front is too small compared to the archive, the best dominated individuals in the previous population are copied to the new archive until the non-dominated front has the same size compared to the archive. According to (Zitzler & Thiele, 1999), the complexity of the fitness evaluation is $O(M^2 \log M)$, $S(i)$ and $R(i)$ is $O(M^2)$. Knowing that $M = N + \overline{N}$, with $N$ is the population size and $N$ corresponds to the archive size. However, the worst run-time complexity of the truncation operation is $O(M^3)$.

### 4.2. *K-means Clustering Algorithm*

Clustering Algorithm is an unsupervised classification technique used in scientific and industrial applications. It is a technique that consists of classifying similar objects into groups, called clusters. K-means is one of the most popular clustering algorithms that aims to assign a given set of data points into k clusters (Mousa et al., 2017). The term "K-means" was first introduced by (Macqueen, 1967). The main steps of K-means are described as Figure 3. First, it fixes the number of clusters $(k)$ to partition a set of points. It starts by randomly choosing $k$ initial centroids for the $k$ clusters. Then, it calculates the Within-Cluster Sum of Square (WCSS) of all the points relative to the cluster's centroid. K-means assigns the points to the closest cluster, then it recalculates the new centroids. these steps are repeated until no point changes its cluster. In this work, K-means methods allows to divide the set of patients into similar clusters, where each of them corresponds to a caregiver. Each cluster contains a sub-set of population, which have to be visited by a given caregiver. The main objective is to find the minimum WCSS between patients and centroids. It is important to note that K-means can be defined as following equation:

$$WCSS = \sum_{i=1}^{K} \sum_{j=1}^{|C_i|} d(x_j - c_i) \tag{26}$$

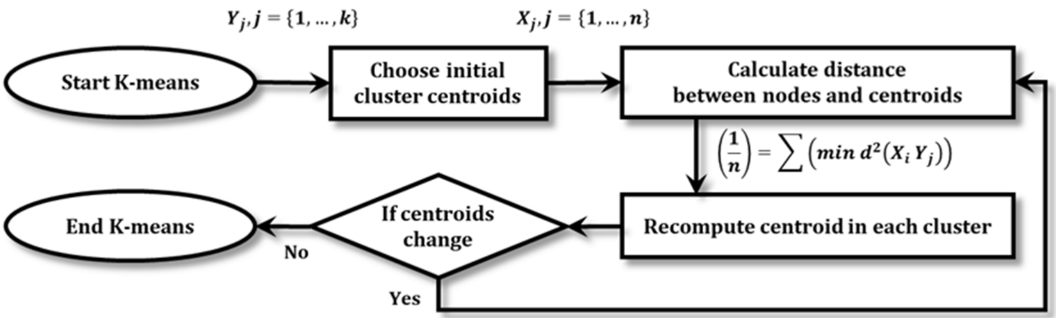


Figure 3: Flowchart of K-means clustering

Where K is the number of clusters, $C_i$ is the size of cluster, ci is the centroid of cluster $C_i$ and d is the Euclidean distance between each element $x_j$ and the centroid. In this paper, the main objective of using the K-mean method is to improve the quality of the Pareto fronts obtained by NSGA-II and SPEA2 in terms of convergence and diversity.

### 4.3. *Hybrid MOEA for HHC-MOVRPTW*

The new hybrid approach that combines MOEA with K-means, is called K-MOEA and described by Algorithm 1.

As illustrated by Figure 4, the K-MOEA is composed of two stages. In the decomposition stage, the set of patients (population) is divided into K clusters (sub-populations) through the K-means method. Knowing that K corresponds to the number of caregivers. In this work, the patients belonging to the same cluster have similar characteristics, such as geographical coordinates and visiting-time preferences. Thus, the K sub-populations are considered as input data for the NSGA-II and SPEA2.

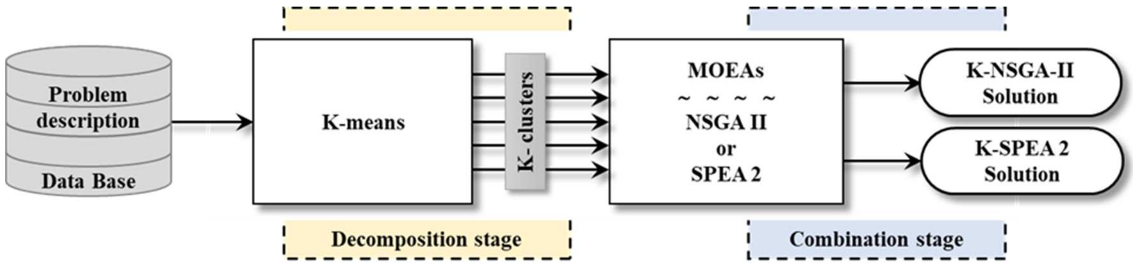


Figure 4: Structure of K-MOEA

The output of this stage is a K Pareto subsets, where each one corresponds to a sub-solution to the problem. In other words, each kth Pareto subset is composed of a set of non-dominated solutions for the kth caregiver, with $k = 1 \dots K$. Each non-dominated solution is represented by a pair of values. The first one shows the minimum service time (first objective) while the second indicates the total time difference compared to the patients' time preferences (second objective). In the combination stage, the K Pareto subsets have to be recombined to form the global Pareto front, consisting of non-dominated solutions for the problem. However, all Pareto subsets should have the same size. In order to obtain the global Pareto front, the K Pareto subsets should have the same size. Thus, according to the smallest size of the Pareto subsets, which is referred T, k new Pareto subsets of size T are randomly generated from the k current ones. In other words, T non-dominated solutions are randomly generated from each Pareto subset to form new ones of size T. Afterwards, all new Pareto subsets are combined by equation (28), that consists of combining the objective values, obtained by solutions of order t of all Pareto subsets, where $t = 1 \dots T$.

$$\forall\, t = 1..T,\ j = 1..m \qquad Q_j^t = \sum_{i=1}^{k} Z_{ji}^t \tag{28}$$

$Q$ is the global value of the objective function $j$ of the solution $t$, K is the number of Pareto subsets and m corresponds to the number of objective functions to be optimized. Figure 4 illustrates the structure of K-MOEA.

**Algorithm 1 : K-MOEA**

**Input :** $P$ : population size, $k$ : number of caregivers // number of clusters

**Output :** $Pareto\_F$ // Final Pareto set

**BEGIN**
  Initialize $Pareto\_F = \emptyset$
  $Pareto = \emptyset$ // list of Pareto subsets
  Generate $k$ random centroids
  Create $k$ empty lists $C = \{ C_1, \dots, C_k \}$ // each list corresponds to a cluster
  Assign $k$ centroids to the k clusters
  **Repeat**
    Calculate the Euclidean distance between each patient and each centroid
    **For** $i \leftarrow 1$ to $P$
      **For** $j \leftarrow 1$ to $k$
        **If** centroid $j$ is closest to $i$
          $C_j \leftarrow i$
        **End**
      **End**

```
        End
        Calculate new centroids
    Until no point changes its cluster.
    For j ← 1 to k
        Pareto[j] ← EA(C_j)        // Run the Evolutionary Algorithm
                                   // (NSGA-II or SPEA2) for each cluster j
    End
    Determine T                    // the smallest size of Pareto subsets
    Create k empty lists S = { S_1, ..., S_k } of size T        // the final Pareto subsets
    Pour i ← 1 to k
        Repeat
            Choose random solution x from Pareto[ j ]
            S_i ← x
        Until |S_i| = T
        For t ← 1 to T
            For b ← 1 to B         // B number of objectives
                Z_t ← Sum (S_i^t[b])
                solution [i] ← Z_t     // global solution
            End
        End
        Pareto_F ← solution [i]    // final Pareto set
    End
    Remove duplicated solutions from Pareto_F
    Return Pareto_F
END
```

## 5. Computational Experiments

In this section, experiments are carried out to validate the mathematical model and to evaluate the performance of the proposed solution approaches.

### 5.1. Implementation and Test instances

All proposed algorithms are implemented in Java language, using jMetal[1] library with the Integrated Development Environment Eclipse IDE -version 2021-12 and executed on a 11th Gen Intel(R) Core(TM) i7-1165G7 CPU computer under Windows 11 with 16 GB of RAM. The lexicographic method is encoded in java, using the Localsolver optimization library. In the literature, there is no standard benchmark for the considered problem. Therefore, 40 test instances are generated from Solomon's VRPTW benchmark[2] with different sizes (25, 50 and 100 customers) and different types (R, C and RC). In this work, customers correspond to patients. The service time is predefined for all instances and equal to 20 minutes. The travel time between patients is measured in minutes.

Firstly, the HHC-MOVRPTW is solved with the lexicographical method in order to obtain a first solution to the problem and then compare it with solutions obtained by Pareto methods. This

[1] http://jmetal.sourceforge.net/
[2] https://www.sintef.no/projectweb/top/vrptw/solomon-benchmark/

method is simulated with 2 orders. In order 1, the first objective is preferred over the second. While order 2 corresponds to the opposite case. Thus, the lexicographical method is simulated 2 times, using the 40 test instances and repeated 20 times in order to diversify the search space and to get better solutions. A limit CPU time is required for each run. So, the proposed model is simulated with 20 different CPU times. In total, $40 \times 2 \times 20 = 1600$ runs are performed with the lexicographical method. Afterwards, the proposed model is simulated with both algorithms NSGA-II and SPEA-2 which are configured with the same parameters. For each algorithm, a set of 20 runs are performed, using 40 test instances with an initial population of size=100, a crossover rate of 0.7 and a mutation rate of 0.2. The maximal number of generations is fixed to 1000 generations per run. In total, $40 \times 2 \times 20 = 1600$ runs have been carried out. For each test instance, 20 Pareto fronts are obtained, where each one contains a set of non-dominated solutions for the problem. Each solution is composed of two values. The first one indicates the minimum total service time whereas the second one indicates the minimum total difference between start-time of services and patients' visiting-time preferences. Subsequently, the proposed hybrid algorithms K- NSGA-II and K-SPEA2 are configured with the same parameters of NSGA-II and SPEA2, in order to ensure a fair comparison between the results obtained by the four algorithms. However, the number of instances being important, it cannot be possible to expose the results of all instances. Therefore, only 6 test instances of different size (25, 50 and 100) and type (C, R, RC) have been selected for testing. Thereby, each algorithm is simulated 20 times with 6 instances. In each run, the number of clusters (K) is changed in order to find the optimal one for the decomposition phase. The output of this phase is a K Pareto subsets, which have to be recombined to form the global Pareto front.

### 5.2. *Computational results*

Table 1 summarizes the results found by the lexicographical method. It indicates the best founded solutions in simulations of order 1 and order 2.

Table 1: Results of lexicographical method

| Instance | $CPU_{Avg}$ | Ordre 1 | | Ordre 2 | | GAP1 % | GAP2 % |
|---|---|---|---|---|---|---|---|
| | | $Z_1^*$ | $Z_2^*$ | $Z_1^*$ | $Z_2^*$ | | |
| C101.25 | 200000 | 687.45 | 57 | 687.45 | 57 | 0 | 0 |
| C101.50 | 500000 | 1359.5 | 123 | 1358.9 | 123 | 0.17 | 0 |
| C101.100 | 250000 | 2820.92 | 240.05 | 2819.56 | 240.05 | 0.17 | 0 |
| C102.25 | 370000 | 687.45 | 58.2 | 687.45 | 58.2 | 0 | 0 |
| C102.50 | 330000 | 1359.5 | 122.3 | 1362.2 | 122.3 | 0.74 | 0 |
| C102.100 | 370000 | 2820.9 | 235.7 | 2828.9 | 235.7 | 0.97 | 0 |
| C107.25 | 100000 | 687.45 | 59.78 | 691.81 | 59.78 | 2.27 | 0 |
| C107.50 | 245000 | 1358.9 | 118.4 | 1363.3 | 118.4 | 1.2 | 0 |
| C107.100 | 200000 | 2820.9 | 240.2 | 2828.9 | 240.2 | 0.97 | 0 |
| C201.25 | 350000 | 696.23 | 59 | 715.54 | 58.3 | 8.96 | 1.2 |
| C201.50 | 450000 | 1346.2 | 121.2 | 1445 | 121.2 | 22.2 | 0 |
| C201.100 | 270000 | 2581 | 244.4 | 2591.6 | 244.4 | 1.79 | 0 |
| C203.25 | 200000 | 696.23 | 59.63 | 723.31 | 59.63 | 12.13 | 0 |
| C203.50 | 200000 | 1346.2 | 126.9 | 1402.5 | 126.9 | 13.99 | 0 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| C203.100 | 450000 | 2581 | 221.6 | 2591.2 | 221.6 | 1.72 | 0 |
| C206.25 | 150000 | 696.23 | 57.85 | 785.39 | 57.85 | 31.24 | 0 |
| C206.50 | 300000 | 1346.2 | 119.3 | 1426.4 | 119.3 | 18.82 | 0 |
| C206.100 | 500000 | 2581 | 247.7 | 2588.5 | 247.7 | 1.27 | 0 |
| C207.25 | 240000 | 696.23 | 57 | 774.78 | 57 | 28.59 | 0 |
| C207.50 | 400000 | 1346.2 | 124.2 | 1426.2 | 124.2 | 18.78 | 0 |
| C207.100 | 500000 | 2581 | 247.5 | 2588.3 | 247.5 | 1.24 | 0 |
| R109.25 | 400000 | 835.27 | 58 | 974.5 | 57.3 | 29.34 | 1.22 |
| R109.50 | 350000 | 1526.7 | 121.4 | 1799.7 | 121 | 34.14 | 0.3 |
| R109.100 | 500000 | 2835.4 | 237.2 | 3195.6 | 238 | 30.13 | 0.34 |
| R111.25 | 500000 | 835.27 | 58.41 | 929.7 | 58 | 21.98 | 0.71 |
| R204.25 | 150000 | 813.34 | 57.93 | 896.79 | 57.93 | 21.03 | 0 |
| R204.50 | 150000 | 1465.3 | 121.6 | 1514.1 | 121.8 | 9.49 | 0.16 |
| R204.100 | 450000 | 2658.2 | 231.9 | 2759.8 | 232 | 13.37 | 0.06 |
| RC106.25 | 500000 | 794.99 | 58.2 | 846.51 | 58.2 | 14.87 | 0 |
| RC106.50 | 500000 | 1518.5 | 119.3 | 1724.7 | 119 | 28.45 | 0.28 |
| RC106.100 | 500000 | 2979.9 | 241.2 | 3439.1 | 242 | 31.91 | 0.32 |
| RC204.25 | 400000 | 726.13 | 56.96 | 827.33 | 56.96 | 30.92 | 0 |
| RC204.50 | 200000 | 1369.9 | 157.3 | 1479.9 | 157.3 | 22.91 | 0 |
| RC204.100 | 500000 | 2658.1 | 231.5 | 2866.2 | 231 | 24.02 | 0.23 |
| RC207.25 | 500000 | 726.13 | 57.5 | 988.51 | 57.5 | 53.71 | 0 |
| RC207.50 | 400000 | 1369.9 | 119.3 | 1666.3 | 119.4 | 44.48 | 0.03 |
| RC207.100 | 500000 | 2658.1 | 241.8 | 3053.1 | 242 | 37.51 | 0.05 |
| RC208.25 | 300000 | 726.13 | 58.96 | 809.6 | 58.96 | 26.96 | 0 |
| RC208.50 | 200000 | 1369.9 | 147 | 15860 | 147.5 | 36.87 | 0.33 |
| RC208.100 | 500000 | 2658.1 | 219.2 | 2803.3 | 219.2 | 18.08 | 0 |

Knowing that each solution is presented by 2 values $Z_1^*$ and $Z_2^*$, which correspond to the first and second objectives respectively (see equations (4.17) and (4.18)). The table indicates also for each instance, the execution time, the gap between $Z_1^*$ of the order 1 and $Z_1^*$ of order 2, which is referred by (GAP 1) and the gap between $Z_2^*$ of order 1 et $Z_2^*$ of order 2, which is noted by (GAP 2).

The computational results show that for the majority of instances, solutions of order 1 are better than those of order 2. Taking the example of instances C201.25, R109.100, and RC208.50, solutions of order 1 are strictly inferior than those of order 2: $Z^*_{1\,order1} < Z^*_{1\,order2}$ and $Z^*_{2\,order1} < Z^*_{2\,order2}$. For instances C101.100, RC207.25, and R204.25, we noticed that $Z^*_{1\,order1} < Z^*_{1\,order2}$ and $Z^*_{2\,order1} = Z^*_{2\,order2}$. However, C101.25 and C102.25 obtained the same results in both simulation orders. ($Z^*_{1\,order1} = Z^*_{1\,order2}$ and $Z^*_{2\,order1} = Z^*_{2\,order2}$). However, the lexicographical method divides the problem into multiple mono-objective problems to obtain a unique solution to the considered. Nevertheless, it tends to privilege one objective over the others, which makes the search space converge towards a particular direction. In addition to the lexicography method, K-NSGA-II and K-SPEA2 are implemented to solve the HHC-MOVRPTW, as well as their original versions NSGA-II and SPEA2. Unlike the lexicographic method, these algorithms provide a set of non-dominated solutions to the problem.

Table 2 presents for each test instance, the number of non-dominated solutions (size of the Pareto front |PF|) found by the two NSGA-II and SPEA2 algorithms, as well as the CPU time. The results are presented by minimum, maximum and average values which are found over 20 runs. According to these results, it was observed that NSGA-II takes about 4 to 15 seconds to find a solution to the problem, while SPEA2 takes between 6 and 20 seconds to solve it. However, SPEA2 is more expensive in run-time than NSGA-II with an average extra-time of around 5000 ms. This is due to the computational complexity of the archiving procedure of SPEA2. This observation was also highlighted in (Deb et al., 2003). In contrast, NSGA-II found more solutions on the Pareto front (in 62.5% of instances), compared to SPEA2. Regarding to K-NSGA-II and K-SPEA2, the results of the decomposition phase are shown in Table 3 that indicates for each instance, the number of clusters, the number of elements that compose it, the CPU time (minimum, maximum and average) as well as the sizes of Pareto fronts (minimum, maximum and average). Knowing that the first cluster of each instance contains a single element, corresponding to the HHC-depot.

Table 2: Results of NSGA-II and SPEA2

| Instance | NSGA-II | | | | | | SPEA2 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | CPU (ms) | | | \|FP\| | | | CPU (ms) | | | \|FP\| | | |
| | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg |
| C101.25 | 4946 | 5149 | 5047.5 | 2 | 10 | 6.4 | 7570 | 10086 | 8828 | 4 | 8 | 6.25 |
| C101.50 | 6025 | 6231 | 6128 | 3 | 8 | 5.6 | 8437 | 10961 | 9699 | 7 | 15 | 10.25 |
| C101.100 | 11085 | 14986 | 13035.5 | 5 | 10 | 8.8 | 13776 | 19903 | 16839.5 | 6 | 10 | 7.6 |
| C102.25 | 4795 | 5291 | 5043 | 6 | 9 | 8 | 7996 | 10279 | 9137.5 | 4 | 9 | 5.5 |
| C102.50 | 6155 | 6437 | 6296 | 3 | 10 | 6.8 | 9257 | 11380 | 10318.5 | 7 | 10 | 8.4 |
| C102.100 | 10376 | 11576 | 10976 | 2 | 8 | 3.2 | 13413 | 16430 | 14921.5 | 6 | 9 | 7.6 |
| C107.25 | 4025 | 4629 | 4327 | 2 | 14 | 6 | 7135 | 9308 | 8221.5 | 2 | 10 | 6.2 |
| C107.50 | 4278 | 4745 | 4511.5 | 2 | 15 | 8.8 | 6673 | 9559 | 8116 | 6 | 12 | 8.8 |
| C107.100 | 9511 | 9821 | 9666 | 7 | 11 | 9.5 | 12554 | 14543 | 13548.5 | 9 | 12 | 10 |
| C201.25 | 3931 | 4985 | 4458 | 7 | 15 | 11.4 | 6720 | 9731 | 8225.5 | 2 | 6 | 5 |
| C201.50 | 4331 | 5299 | 4815 | 9 | 15 | 12.4 | 7460 | 9990 | 8725 | 6 | 8 | 7.2 |
| C201.100 | 10602 | 12708 | 11655 | 2 | 15 | 7.6 | 13789 | 17623 | 15706 | 7 | 11 | 9 |
| C203.25 | 3978 | 4456 | 4217 | 2 | 9 | 6.2 | 6808 | 9162 | 7985 | 3 | 4 | 3.4 |
| C203.50 | 4231 | 5241 | 4736 | 4 | 14 | 8.6 | 6984 | 9927 | 8455.5 | 5 | 11 | 8.8 |
| C203.100 | 10512 | 13487 | 11999.5 | 2 | 9 | 6.4 | 13254 | 18482 | 15868 | 9 | 13 | 10.8 |
| C206.25 | 5715 | 5898 | 5806.5 | 7 | 14 | 11.4 | 8500 | 10783 | 9641.5 | 4 | 7 | 6.17 |
| C206.50 | 5936 | 6299 | 6117.5 | 7 | 12 | 9 | 8577 | 11137 | 9857 | 5 | 11 | 8.2 |
| C206.100 | 9144 | 11363 | 10253.5 | 4 | 14 | 8.2 | 12561 | 16051 | 14306 | 3 | 13 | 9.6 |
| C207.25 | 4436 | 4823 | 4629.5 | 7 | 9 | 7.6 | 7086 | 9676 | 8381 | 4 | 6 | 5.17 |
| C207.50 | 5198 | 5625 | 5411.5 | 3 | 9 | 6.4 | 7778 | 10565 | 9171.5 | 6 | 11 | 8 |
| C207.100 | 11700 | 14087 | 12893.5 | 6 | 12 | 8 | 14487 | 18965 | 16726 | 6 | 16 | 11.4 |
| R109.25 | 5159 | 5515 | 5337 | 5 | 9 | 7.25 | 8546 | 10511 | 9528.5 | 5 | 8 | 5.8 |
| R109.50 | 7248 | 7487 | 7367.5 | 7 | 15 | 11 | 10404 | 12381 | 11392.5 | 6 | 11 | 9 |
| R109.100 | 10474 | 10856 | 10665 | 5 | 15 | 10.8 | 13187 | 15507 | 14347 | 4 | 14 | 8.8 |
| R111.25 | 5198 | 5158 | 5178 | 8 | 15 | 11.4 | 8442 | 9871 | 9156.5 | 6 | 8 | 6.8 |
| R204.25 | 5198 | 5525 | 5361.5 | 6 | 10 | 8 | 8212 | 10375 | 9293.5 | 3 | 7 | 5 |
| R204.50 | 6463 | 7371 | 6917 | 5 | 10 | 8.8 | 9804 | 12148 | 10976 | 5 | 14 | 10 |
| R204.100 | 11068 | 13856 | 12462 | 7 | 13 | 10.4 | 13373 | 18808 | 16090.5 | 8 | 14 | 10.4 |

| | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| RC106.25 | 4701 | 5539 | 5120 | 9 | 13 | 10.25 | 7895 | 10530 | 9212.5 | 4 | 9 | 6.6 |
| RC106.50 | 4038 | 7192 | 5856.5 | 8 | 15 | 11.2 | 10323 | 9285 | 9804 | 6 | 10 | 8.6 |
| RC106.100 | 9396 | 11199 | 10297.5 | 10 | 15 | 12.8 | 12393 | 15854 | 14123.5 | 4 | 12 | 7.2 |
| RC204.25 | 5158 | 5501 | 5329.5 | 8 | 15 | 11.6 | 8229 | 10308 | 9268.5 | 5 | 8 | 6.6 |
| RC204.50 | 6540 | 7198 | 6869 | 7 | 8 | 7.2 | 9001 | 11965 | 10483 | 8 | 13 | 10 |
| RC204.100 | 9525 | 13801 | 11663 | 9 | 13 | 11.2 | 12835 | 18788 | 15811.5 | 6 | 12 | 9.8 |
| RC207.25 | 6087 | 6498 | 6292.5 | 6 | 15 | 10.6 | 9461 | 11331 | 10396 | 3 | 9 | 6.4 |
| RC207.50 | 7198 | 7488 | 7343 | 9 | 13 | 10.8 | 9674 | 12269 | 10971.5 | 6 | 12 | 10 |
| RC207.100 | 11356 | 11540 | 11448 | 5 | 13 | 9.4 | 14852 | 16393 | 15622.5 | 6 | 10 | 8.6 |
| RC208.25 | 5487 | 5865 | 5676 | 9 | 15 | 12.2 | 8404 | 10518 | 9461 | 2 | 9 | 5.4 |
| RC208.50 | 4768 | 7499 | 6133.5 | 10 | 13 | 11.4 | 7630 | 12455 | 10042.5 | 7 | 11 | 9.2 |
| RC208.100 | 8285 | 8785 | 8535 | 7 | 14 | 9 | 11571 | 13586 | 12578.5 | 5 | 10 | 8.4 |

In this work, we consider only $k-1$ clusters (except the first one). However, the depot is only taken into account when calculating the fitness function. Thus, in 56% of instances, the average size of Pareto $|PF|_{Avg}$ obtained by K-NSGA-II is larger than that obtained by K-SPEA2. In addition, it is observed that K-NSGA-II is faster than K-SPEA2 in all instances. Based on these results, it is important to choose the best Pareto fronts for the combi- nation phase. Therefore, a performance study is carried out to evaluate the quality of the Pareto fronts. In the literature, there are several multi-objective quality indicators (Zitzler et al., 2003) that can be distinguished according to the performance objective. Either the convergence towards the optimal Pareto front and/or the diversity of solutions on the Pareto front. In this work, three performance indicators are used to compare Pareto fronts. Two indicators evaluate the diversity of solutions, while the third one measures the convergence:

Table 3: Results of K-NSGA-II and K-SPEA2

| Instance | cluster | #cluster | size | K-NSGA II | | | | | | K-SPEA2 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | CPU (ms) | | | FP | | | CPU (ms) | | | FP | | |
| | | | | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg |
| C101.25 | 3 | C1 | 12 | 2103 | 2897 | 2500 | 3 | 7 | 5.00 | 2501 | 3587 | 3044 | 4 | 5 | 4.50 |
| | | C2 | 13 | 2128 | 2937 | 2532.5 | 4 | 7 | 5.50 | 2706 | 4002 | 3354 | 4 | 6 | 5.00 |
| C101.100 | 5 | C1 | 7 | 2034 | 2578 | 2306 | 2 | 6 | 4.00 | 2421 | 3527 | 2974 | 2 | 5 | 3.33 |
| | | C2 | 23 | 3256 | 4689 | 3972.5 | 5 | 7 | 5.50 | 5248 | 6021 | 5634.5 | 3 | 5 | 4.00 |
| | | C3 | 40 | 4789 | 5899 | 5344 | 6 | 11 | 7.75 | 6025 | 9025 | 7525 | 5 | 9 | 6.66 |
| | | C4 | 30 | 4032 | 4687 | 4359.5 | 3 | 7 | 5.75 | 5032 | 7868 | 6450 | 4 | 8 | 12.00 |
| C107.100 | 5 | C1 | 7 | 2005 | 2214 | 2109.5 | 4 | 5 | 4.50 | 2337 | 3782 | 3059.5 | 3 | 5 | 4.00 |
| | | C2 | 23 | 3520 | 5021 | 4270.5 | 5 | 6 | 5.50 | 5103 | 6115 | 5609 | 8 | 10 | 9.00 |
| | | C3 | 40 | 5762 | 6277 | 6019.5 | 6 | 7 | 6.50 | 6236 | 8311 | 7273.5 | 7 | 11 | 9.00 |
| | | C4 | 30 | 4122 | 5011 | 4566.5 | 3 | 7 | 5.00 | 5547 | 8001 | 6774 | 8 | 11 | 9.50 |
| C206.50 | 4 | C1 | 15 | 2301 | 2758 | 2529.5 | 4 | 5 | 4.50 | 3655 | 5981 | 4818 | 4 | 5 | 4.50 |
| | | C2 | 12 | 2037 | 2869 | 2453 | 3 | 4 | 3.50 | 3122 | 4955 | 4038.5 | 4 | 5 | 4.50 |
| | | C3 | 23 | 3444 | 4805 | 4124.5 | 5 | 6 | 5.50 | 3900 | 6958 | 5429 | 5 | 6 | 5.50 |
| R109.25 | 3 | C1 | 9 | 2504 | 2911 | 2707.5 | 4 | 7 | 5.50 | 2360 | 2822 | 2591 | 4 | 5 | 4.50 |
| | | C2 | 16 | 2237 | 2901 | 2569 | 3 | 6 | 4.50 | 2511 | 3738 | 3124.5 | 3 | 5 | 4.00 |
| RC106.50 | 4 | C1 | 12 | 2063 | 2972 | 2517.5 | 4 | 6 | 5.00 | 3417 | 4003 | 3710 | 2 | 6 | 4.00 |
| | | C2 | 24 | 3775 | 5047 | 4411 | 3 | 11 | 7.00 | 5207 | 7866 | 6536.5 | 6 | 10 | 8.00 |
| | | C3 | 14 | 2987 | 3022 | 3004.5 | 5 | 12 | 8.50 | 3689 | 6611 | 5150 | 4 | 8 | 6.00 |

- The metric $|PF|$ identifies the number of non-dominated solutions belonging to the first Pareto front. The larger the number, the Pareto has better diversity.
- The Spacing metric $(SP)$ is proposed by (Schott, 1995). It makes it possible to measure the diversity of non-dominated solutions belonging to the same Pareto, by comparing the uniform distribution and the deviation of the solutions. It calculates the relative distance between solutions as follows:

$$SP = \sqrt{\frac{1}{|FP|}\sum_{i=1}^{|FP|}\left(d_i - \bar{d}\right)^2} \tag{29}$$

where PF is the number of non-dominated solutions belonging to the same Pareto front, is the average distance between a solution $i$ and the rest of solutions on both objectives and $\bar{d} = \sum_{i=1}^{|FP|} d_i / |FP|$ is the average distance between all solutions on the two objectives. In our case, the objectives are not of the same magnitude. Thus, we normalized the values of two objectives over the interval [0,1] before determining the SP metric. Knowing that this metric is to be minimized, that is means if SP is low (tends to 0), the solutions are uniformly spaced. Therefore, the algorithm which allows to obtain the lowest value of SP is the better.
- The Hypervolume metric $(Hv)$ is proposed by (Zitzler et al., 2003), that measures the quality of the approximation in terms of convergence and diversity. It thus allows to calculate the multi-dimensional region separating a Pareto front and a reference point $(Z^{ref})$. The algorithm that provides a large value of $(H_v)$, implies better convergence. To calculate the hypervolume, the solutions are normalized on an interval [0,1]. This work deals with a minimization problem. So, the reference point is determined from the worst values obtained (upper bounds) for each objective $Z^{ref} = (\bar{Z}_1, \bar{Z}_2)$. The hypervolume is therefore can be determined through the following equation, Knowing that $Z_1(i) \leq Z_1(i+1)$:

$$Hv(PF, Z^{ref}) = \sum_{i=1}^{|FP|} |Z_1(i+1) - Z_1(i)| \left|Z_2^{ref} - Z_2(i)\right| \tag{30}$$

The results obtained from the performance analysis are summarized in Table 4 It indicates for each instance, the minimum, the maximum and the average values of the Spacing (SP) and hypervolume (Hv) indicators. It should be noted that in this paper, the objectives are not of the same magnitude. Thus, they have been normalized on the interval [0,1] before determining the performance metrics. According to Table 4, the average hypervolume value obtained by K-SPEA2 is higher than that obtained by K-NSGA-II, in 88.8% of instances. In addition, K-SPEA2 is better in terms of convergence and diversity compared to K-NSGA-II, in 39% of instances.

Therefore, it is clear to interpret that the Pareto fronts obtained by K-SPEA2 converge better towards the optimal Pareto front. In addition, K-SPEA2 has a better distribution capacity that can help it to convergence.

However, K-SPEA2 needs more time to find solutions, compared to K-NSGA-II. In fact, K-SPEA2 takes an almost negligible extra-time (about 13.78 ms). It should be noted that the two algorithms K-SPEA2 and K-NSGA-II are executed with small-size instances (clusters). Thereby, K-NSGA-II requires 2 to 6 seconds, while K-SPEA2 needs 2 to 9 seconds to solve the problem. By Comparing the two hybrid algorithms with their original versions, it can be seen that dividing the problem into sub-problems saves time (about 3000 ms). In addition, the results of hypervolume metric are illustrated graphically in Figure 5, using box-plots. So that, each box presents for each test instance, the 20 hypervolume values obtained from the 20 runs. By observing the behavior of the boxes, we can easily notice that K-NSGA-II and K-SPEA2 are very similar in terms of diversity in the most of instances. Nevertheless, the hypervolume values of K-

SPEA2 are higher than those of K-NSGA-II (in 60% of instances). It means that, K-SPEA2 is better in terms of convergence. It should be noted that the best Pareto front should have the best ratio between convergence towards the optimal front and the diversity of non-dominated solutions. Thus, a new weighted metric W is defined to measure both diversity and convergence of the Pareto fronts. Knowing that PF is also normalized in an interval [0,1] before calculating this metric.

$$W = \alpha\ |PF| + \beta\ (1 - SP) + \gamma\ Hv \tag{31}$$

where $\alpha + \beta + \gamma = 1$: $\alpha$, $\beta$ and $\gamma$ have the same value. For each test instance, the best Pareto front (Pareto subset) is the one with the highest value of $W$. Once this metric is determined for all clusters, the best Pareto subsets have to be combined to form the global Pareto front. Thus, the output of the combination stage is a global Pareto front. As the combination phase is done 20 times with both algorithms K-NSGA-II and K-SPEA2, we have obtained 20 global Pareto fronts by each of them. Now, in order to evaluate the performance of K-NSGA-II, K-SPEA2 and their original versions NSGA-II and SPEA2, we have done a performance study, presented by Table 5 through the previously used metrics.

According to the found results, it is observed that the hypervolume results obtained by SPEA2 are better (in 83% of instances) compared to those obtained by NSGA-II. Therefore, we can interpret that SPEA2 provides better convergence compared to NSGA-II.

Table 4: Performance study 1

| Instance | # Cluster | K-NSGAII | | | | | | K-SPEA2 | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SP | | | Hv | | | SP | | | Hv | | |
| | | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg | Min | Max | Avg |
| C101.25 | C1 | 0.019 | 0.227 | 0.097 | 0.306 | 0.846 | 0.564 | 0.165 | 0.284 | 0.206 | 0.611 | 1.031 | 0.855 |
| | C2 | 0.083 | 0.179 | 0.131 | 0.261 | 0.751 | 0.481 | 0.103 | 0.268 | 0.185 | 0.480 | 0.960 | 0.722 |
| C101.100 | C1 | 0.100 | 0.204 | 0.140 | 0.283 | 0.783 | 0.497 | 0.146 | 0.172 | 0.159 | 0.395 | 0.895 | 0.670 |
| | C2 | 0.224 | 0.471 | 0.335 | 0.579 | 1.039 | 0.853 | 0.049 | 0.140 | 0.110 | 0.757 | 1.098 | 0.944 |
| | C3 | 0.153 | 0.368 | 0.270 | 0.708 | 1.071 | 0.914 | 0.163 | 0.340 | 0.274 | 0.847 | 1.087 | 1.003 |
| | C4 | 0.104 | 0.414 | 0.279 | 0.552 | 1.082 | 0.903 | 0.169 | 0.225 | 0.195 | 0.808 | 1.078 | 0.998 |
| C107.100 | C1 | 0.060 | 0.177 | 0.119 | 0.381 | 0.871 | 0.619 | 0.256 | 0.268 | 0.262 | 0.627 | 1.027 | 0.844 |
| | C2 | 0.228 | 0.304 | 0.266 | 0.625 | 1.085 | 0.857 | 0.134 | 0.150 | 0.142 | 0.753 | 1.090 | 0.948 |
| | C3 | 0.079 | 0.149 | 0.114 | 0.745 | 1.096 | 0.998 | 0.156 | 0.284 | 0.220 | 0.782 | 1.063 | 0.953 |
| | C4 | 0.078 | 0.314 | 0.196 | 0.656 | 1.084 | 0.915 | 0.234 | 0.409 | 0.322 | 0.778 | 1.079 | 0.980 |
| C206.50 | C1 | 0.159 | 0.214 | 0.187 | 0.554 | 0.934 | 0.803 | 0.100 | 0.196 | 0.148 | 0.671 | 1.063 | 0.890 |
| | C2 | 0.259 | 0.374 | 0.317 | 0.466 | 0.896 | 0.666 | 0.023 | 0.220 | 0.121 | 0.494 | 0.994 | 0.727 |
| | C3 | 0.232 | 0.357 | 0.294 | 0.661 | 1.075 | 0.884 | 0.324 | 0.345 | 0.334 | 0.793 | 1.086 | 0.974 |
| R109.25 | C1 | 0.154 | 0.308 | 0.231 | 0.338 | 0.838 | 0.576 | 0.153 | 0.258 | 0.205 | 0.545 | 1.027 | 0.776 |
| | C2 | 0.165 | 0.172 | 0.170 | 0.435 | 0.965 | 0.746 | 0.266 | 0.278 | 0.272 | 0.402 | 0.992 | 0.654 |
| RC106.50 | C1 | 0.045 | 0.193 | 0.119 | 0.290 | 0.780 | 0.463 | 0.178 | 0.188 | 0.184 | 0.829 | 1.099 | 1.007 |
| | C2 | 0.048 | 0.280 | 0.164 | 0.695 | 1.097 | 0.887 | 0.199 | 0.268 | 0.234 | 0.788 | 1.091 | 0.993 |
| | C3 | 0.154 | 0.169 | 0.161 | 0.619 | 1.030 | 0.823 | 0.115 | 0.175 | 0.145 | 0.684 | 1.044 | 0.917 |

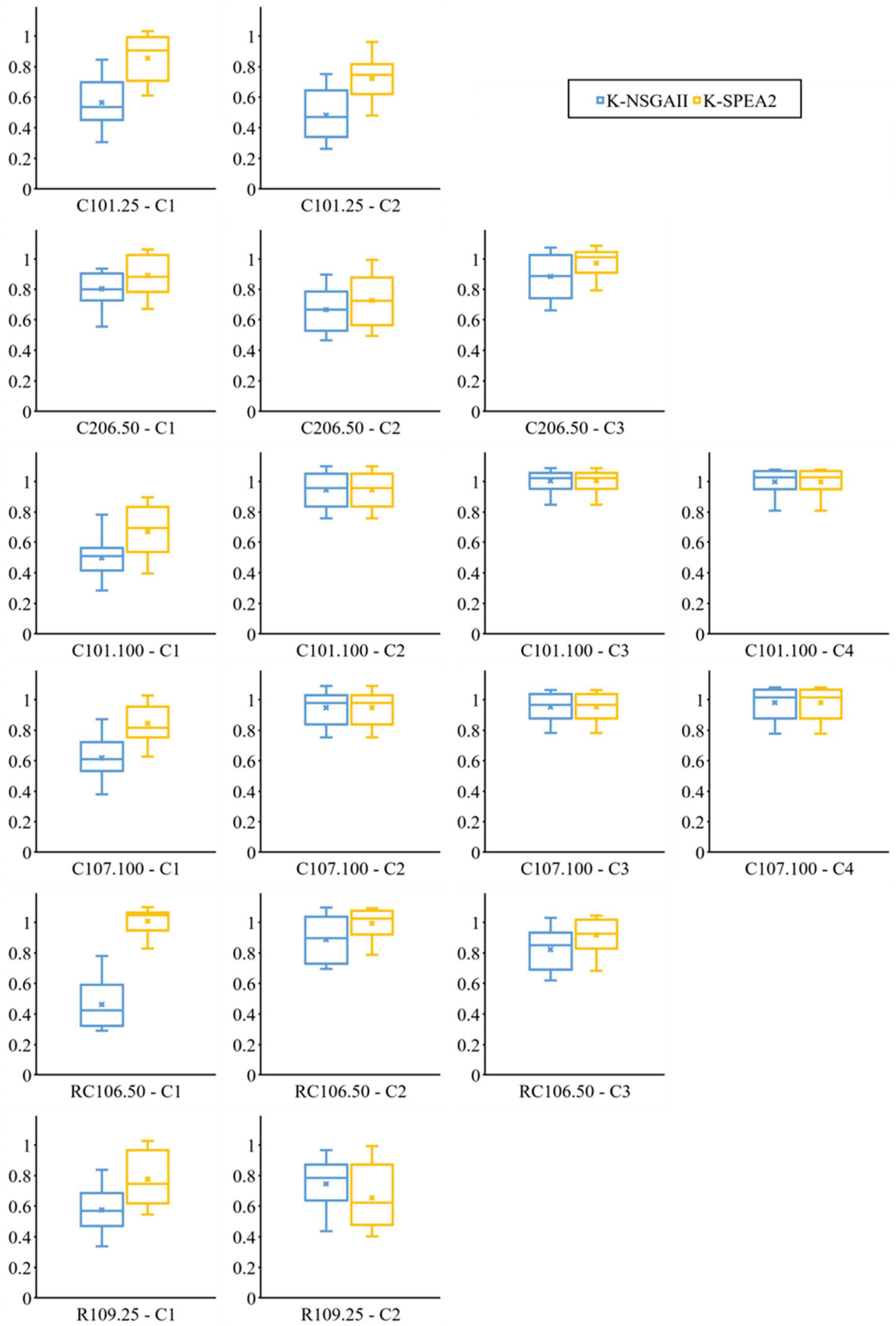

Figure 5: Results of Hypervolumes

On the other hand, good performance is obtained by NSGA-II in terms of SP metric (in 66%

of instances). Thus, the solutions of the Pareto fronts obtained by NSGA-II are evenly spaced and better distributed compared to those found using SPEA2. Moreover, it is noticed that both algorithms K-NSGA-II and K-SPEA2 also have good performance in terms of SP compared to their original version. In average, the SP values obtained by K-SPEA2 vary between 0.13 and 0.23, while those obtained by K-NSGA-II vary between 0.19 and 0.29. Furthermore, the average hypervolume values obtained by SPEA2 vary between 0.75 and 0.98, while those found with K-SPEA2 vary between 0.8 and 0.97. It means that both versions of SPEA provide better convergence towards the optimal Pareto front. However, K-SPEA2 slightly exceeds SPEA2 in terms of convergence, with an average deviation of 5%, but they are similar in terms of diversity. However, K-NSGA-II and K-SPEA2 are similar in terms of convergence and diversity.

Table 5: Performance study 2

| Instance | NSGA-II | | SPEA2 | | K-NSGA-II | | K-SPEA2 | |
|---|---|---|---|---|---|---|---|---|
| | $Hv_{\text{Avg}}$ | $SP_{\text{Avg}}$ | $Hv_{\text{Avg}}$ | $SP_{\text{Avg}}$ | $Hv_{\text{Avg}}$ | $SP_{\text{Avg}}$ | $Hv_{\text{Avg}}$ | $SP_{\text{Avg}}$ |
| C101.25 | 0.789 | 0.421 | 0.854 | 0.326 | 0.905 | 0.156 | 0.974 | 0.198 |
| C101.100 | 0.725 | 0.123 | 0.759 | 0.266 | 0.81 | 0.193 | 0.955 | 0.197 |
| C107.100 | 0.785 | 0.199 | 0.964 | 0.255 | 0.815 | 0.133 | 0.97 | 0.298 |
| C206.50 | 0.866 | 0.196 | 0.988 | 0.236 | 0.865 | 0.199 | 0.836 | 0.201 |
| R109.25 | 0.879 | 0.365 | 0.748 | 0.218 | 0.799 | 0.239 | 0.964 | 0.228 |
| RC106.50 | 0.882 | 0.134 | 0.899 | 0.163 | 0.802 | 0.203 | 0.817 | 0.226 |

In order to validate the performance of the four algorithms, a statistical analysis has been carried out through a non-parametric test, namely "Wilcoxon signed rank test". The statistical analysis is necessary to draw reliable conclusions about the performance of multi- objective algorithms. For each test instance, 20 hypervolumes and 20 values of SP are founded by NSGA-II, SPEA2, K-NSGA-II and K-SPEA2.

Let's remember that all algorithms share the same parameters for all runs, such as population size, crossover and mutation rates, etc. Thus, a pairwise comparison is performed, as explained in (Wilcoxon, 1992) and (Harris & Hardin, 2013) with a probability value $(p - value) \leq 0.05$. The pairs of algorithms are compared according to the hypervolume and the SP metrics. Therefore, the statistical test indicates for each test instance and depending on the indicator considered, whether a given algorithm is significantly better than the other algorithm or there is no significant difference between them. The test results are shown in Table 6. For each instance and according to the performance indicator, either the algorithm located at a given row is significantly better (**) than that in a given column, or it is significantly worse (x), or there is no significant difference between them (‖).

Table 6: Algorithms comparison

| Instance | Algorithme | NSGA-II | | SPEA2 | | K-NSGA-II | | K-SPEA2 | |
|---|---|---|---|---|---|---|---|---|---|
| | | Hv | SP | Hv | SP | Hv | SP | Hv | SP |
| C101.25 | K-NSGA-II | ** | ** | \|\| | ** | - | - | \|\| | \|\| |
| | K-SPEA2 | ** | x | \|\| | ** | \|\| | \|\| | - | - |
| C101.100 | K-NSGA-II | \|\| | x | ** | ** | - | - | x | \|\| |
| | K-SPEA2 | ** | x | ** | ** | ** | \|\| | - | - |
| C107.100 | K-NSGA-II | ** | ** | ** | ** | - | - | x | ** |
| | K-SPEA2 | ** | x | ** | \|\| | ** | x | - | - |
| C206.50 | K-NSGA-II | \|\| | \|\| | x | ** | - | - | ** | x |

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | K-SPEA2 | x | ** | x | ** | x | ** | - | - |
| R109.25 | K-NSGA-II | x | \|\| | \|\| | x | - | - | x | \|\| |
| | K-SPEA2 | ** | \|\| | ** | \|\| | ** | \|\| | - | - |
| RC106.50 | K-NSGA-II | x | x | x | x | - | - | x | \|\| |
| | K-SPEA2 | x | x | x | ** | ** | ** | - | - |

As discussed before, the hybrid algorithms are similar in term of performance. SPEA2 and K-SPEA2 are always better in terms of convergence but they require more time to give solutions to the problem. While NSGA-II and K-NSGA-II are the faster in computation and offer good performance in terms of convergence and diversity. Also, it is observed that the new hybrid algorithms K-NSGA-II and K-SPEA2 are faster than their original versions when dividing the considered problem into sub-problems. Nevertheless, in the overall process, they take more time to provide global solutions to the problem. Table 7 summarizes the best founded solutions of all algorithms. By comparing the founded results with those of lexicographical method, it is clear to observe that Pareto methods provide decision makers with a set of best solutions in a single algorithm run. They also allow to widen the search space thanks to the evolutionary process and therefore, offer a better convergence towards the optimal solutions. Unlike Pareto methods, the lexicography method provides a unique solution to the problem, favoring one objective over the others. However, it requires much more time to solve the problem (between 100 and 500 seconds) compared to the other algorithms. For all these reasons, we can conclude that the Pareto methods are more efficient than the lexicography method to solve the considered the problem, and that the hybrid algorithms K-NSGA-II and K-SPEA2 have succeeded in improving the quality of the Pareto fronts, thanks to the K-means clustering technique.

Table 7: Best solutions

| Instance | NSGA-II | | | SPEA2 | | | # Cluster | K-NSGA-II | | | K-SPEA2 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | \|FP\| | $Z_1^*$ | $Z_2^*$ | \|FP\| | $Z_1^*$ | $Z_2^*$ | | \|FP\| | $Z_1^*$ | $Z_2^*$ | \|FP\| | $Z_1^*$ | $Z_2^*$ |
| C101.25 | 10 | 622 | 58.14 | 6 | 623 | 58.18 | 1 | 5 | 336 | 26.58 | 4 | 326 | 28.97 |
| | | | | | | | 2 | 7 | 331 | 27.02 | 4 | 331 | 28.56 |
| **final solutions** | | | | | | | | 5 | 667 | 53.60 | 4 | 657 | 57.53 |
| C101.100 | 9 | 3875 | 240 | 6 | 3806 | 251.18 | 1 | 5 | 189 | 17.84 | 3 | 187 | 17.85 |
| | | | | | | | 2 | 5 | 603 | 56.50 | 3 | 594 | 56.50 |
| | | | | | | | 3 | 7 | 1209 | 96.69 | 5 | 1179 | 100.79 |
| | | | | | | | 4 | 3 | 941 | 73.07 | 8 | 860 | 72.80 |
| **final solutions** | | | | | | | | 3 | 2942 | 244.10 | 3 | 2820 | 247.94 |
| C107.100 | 11 | 3758 | 241.93 | 9 | 3753 | 259.10 | 1 | 5 | 321 | 50.92 | 3 | 321 | 59.60 |
| | | | | | | | 2 | 6 | 767 | 55.01 | 8 | 799 | 53.85 |
| | | | | | | | 3 | 7 | 1256 | 52.67 | 11 | 1311 | 55.01 |
| | | | | | | | 4 | 3 | 1020 | 52.66 | 8 | 1009 | 50.33 |
| **final solutions** | | | | | | | | 3 | 3364 | 211.26 | 3 | 3440 | 218.79 |
| C206.50 | 7 | 1598 | 120.88 | 11 | 1558 | 121.34 | 1 | 5 | 452 | 39.91 | 4 | 450 | 44.12 |
| | | | | | | | 2 | 3 | 406 | 40.55 | 5 | 406 | 42.09 |
| | | | | | | | 3 | 6 | 712 | 45.69 | 5 | 704 | 33.43 |

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **final solutions** | | | | | | | | 3 | 1570 | 126.15 | 4 | 1560 | 119.64 |
| R109.25 | 5 | 624 | 59.37 | 5 | 834 | 58.57 | 1 | 7 | 358 | 29.10 | 5 | 358 | 30.18 |
| | | | | | | | 2 | 3 | 532 | 28.70 | 5 | 532 | 29.83 |
| **final solutions** | | | | | | | | 3 | 890 | 57.8 | 5 | 890 | 60.01 |
| RC106.50 | 10 | 1756 | 128.12 | 10 | 1766 | 121.38 | 1 | 6 | 497 | 42.33 | 4 | 497 | 50.01 |
| | | | | | | | 2 | 11 | 774 | 49.69 | 10 | 829 | 32.56 |
| | | | | | | | 3 | 12 | 773 | 40.23 | 8 | 530 | 40.03 |
| **final solutions** | | | | | | | | 6 | 2044 | 132.25 | 4 | 1856 | 122.60 |

## 6. Conclusion

In this paper, a new bi-objective mathematical model is proposed to deal with the HHCRSP, which aims to minimize both total service time and the difference between ser- vices' start times and visiting-time preferences. Three methods are proposed to solve the problem. The first one is called Lexicographical method that belongs to the non-scalar resolution methods while the second and the third ones belong to Pareto method. In fact, two well-known MOEAs namely NSGA-II and SPEA2 are implemented to solve the problem, then they are combined with the well-known K-means clustering method, in order to improve the quality of the obtained Pareto fronts. All proposed approaches are evaluated through test instances, taken from the VRPTW Solomon's Benchmark. According to the experimental results, it is observed that K-SPEA2 has a better performance than SPEA2 by 8% of GAP in term of convergence and by 6% of GAP regarding the diversity indicator. While K-NSGA-II has 22% of GAP compared to NSGA-II in term of diversity and a GAP of 1.4% regarding the convergence indicator. By comparing Pareto methods with lexicographical method, it is clear to interpret that NSGA-II, SPEA2, K-NSGA-II and K-SPEA2 are faster in execution time than lexicographical method and provide the decision makers more than a single solution in just one run of the algorithm. The experimental results also prove the efficiency and the suitable of MOEAs in terms of CPU time and the quality of solutions facing the HHC-MOVRPTW.

## 7. Statements and Declarations

The authors have no relevant financial or non-financial interests to disclose. All authors have no conflicts of interest to declare that are relevant to the content of this article. The authors certify that they have no affiliations with or involvement in any organization or entity with any financial interest or non-financial interest in the subject matter or materials discussed in this manuscript.

## References

Ait Haddadene, S. R., Labadie, N., & Prodhon, C. (2019). Bicriteria vehicle routing problem with preferences and timing constraints in home health care services. Algorithms, 12 (8), 152.

Alves, F., Costa, L. A., Rocha, A. M., I Pereira, A., & Leitão, P. (2019). A Multi-objective Approach to the Optimization of Home Care Visits Scheduling. In and others (Ed.), 8th international conference on operations research and entreprise systems, 19-21 februay 2019, prague, insticc. ICORES.

Arora, N., & Banyal, R. K. (2022). Hybrid scheduling algorithms in cloud computing: A review. International Journal of Electrical & Computer Engineering (2088-8708), 12 (1).

Augusto, V., & Xie, X. (2009). Redesigning pharmacy delivery processes of a health care complex. Health care management science, 12 (2), 166–178.

Avci, M., & Topaloglu, S. (2016). hybrid metaheuristic algorithm for heterogeneous vehicle routing problem with simultaneous pickup and delivery. Expert Systems with Applications, 53, 160–171.

Ayadi, R., & Benadada, Y. (2013). Memetic Algorithm for a MultiObjective Vehicle Routing Problem with Multiple Trips. International Journal of Advanced Computer Science and Applications: IJCSA, 10 (2), 72–91.

Bandyopadhyay, S., & Bhattacharya, R. (2014). Solving a tri-objective supply chain problem with modified nsga-ii algorithm. Journal of Manufacturing Systems, 33 (1), 41–50.

Baños, R., Ortega, J., Gil, C., MáRquez, A. L., & De Toro, F. (2013). A Hybrid Meta-Heuristic for Multi-Objective Vehicle Routing Problems with Time Windows. Comput. Ind. Eng., 65 (2). https://doi.org/10.1016/j.cie.2013.01.007

Bansal, S., & Goel, R. (2018). Multi Objective Vehicle Routing Problem: A Survey. Asian Journal of Computer Science and Technology, 7 (3), 1–6.

Belhor, M., El-Amraoui, A., Delmotte, F., & Jemai, A. (2020). A new mip model and machine learning approach for home health care: Optimization of cancer treatment process by chemotherapy. 2020 5th International Conference on Logistics Operations Management (GOL), 1–7.

Belhor, M., El-Amraoui, A., Jemai, A., & Delmotte, F. (2023). Learning-based metaheuristic approach for home healthcare optimization problem. Computer Systems Science and Engineering, 45, 1–19.

Ben Houria, Z., Masmoudi, M., Hanbali, A. A., Khatrouch, I., & Masmoudi, F. (2016). Quantitative techniques for medical equipment maintenance management. European journal of industrial engineering, 10 (6), 703–723.

Bergh, J. V., Beliën, J., Bruecker, P. D., Demeulemeester, E., & Boeck, L. D. (2013). Personnel scheduling: A literature review. Eur. J. Oper. Res., 226, 367–385.

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

Branke, J., Deb, K., Miettinen, K., & Slowiński, R. (2008). Multiobjective Optimization. In and others (Ed.), Interactive and Evolutionary Approaches (1st ed.). Springer, Berlin, Heidelberg.

Castillo-Salazar, J. A., Landa-Silva, D., & Qu, R. (2012). A survey on workforce scheduling and routing prob- lems. 9th International Conference on the Practice and Theory of Automated Timetabling (PATAT 2012), 283–302.

Castillo-Salazar, J. A., Landa-Silva, D., & Qu, R. (2016). Workforce scheduling and routing problems: literature survey and computational study. Annual Operations Research, 239 (1), 39–67.

Castro-Gutierrez, J., Landa, Silva, D., & Pérez, J. M. (2011). Nature of real world multi-objective vehicle routing with evolutionary algorithms, Systems, Man and Cybernetics (SMC). IEEE, 257–264.

Chávez, J., Escobar, J., & Echeverri, M. (2016). A multi-objective Pareto ant colony algorithm for the Multi- Depot Vehicle Routing problem with Backhauls. International Journal of Industrial Engineering Computations, 7 (1), 35–35.

Cheng, E., & Rich, J. L. (1998). A Home Health Care Routing and Scheduling Problem.

Chiang, T. C., & Hsu, W. H. (2014). A knowledge-based evolutionary algorithm for the multi objective vehicle routing problem with time windows. Computers and Operations Research, 45, 25–37.

Cholodowicz, V., & Orlowski, E. (2017). Comparison of SPEA2 and NSGA-II Applied to Automatic Inventory Control System Using Hypervolume Indicator. Studies in Informatics and Control, 26, 67– 74.

Cococcioni, M., Pappalardo, M., & Sergeyev, Y. D. (2018). Lexicographic multi-objective linear programming using grossone methodology: Theory and algorithm. Applied Mathematics and Computation, 318, 298–311.

Coello, C. A. C., San, C., & México, P. (2001). A short tutorial on evolutionary multiobjective optimization. In and others (Ed.), International conference on evolutionary multi-criterion optimization (pp. 21– 40). Springer.

Dantzig, G. B. (1954). A comment on Edie's Traffic Delays at Toll Booths. Journal of the Operations Research Society of America, 2 (3), 339–341.

Deb, K. (2001). Multi-objective optimization using evolutionary algorithms. John Wiley; Sons, Inc.

Deb, K., Mohan, M., & Mishra, S. (2003). Towards a quick computation of well-spread pareto-optimal solutions. International Conference on Evolutionary Multi-Criterion Optimization, 222–236.

Decerle, J., O Grunder, Hassani, A. H. E., & O Barakat. (2019). A memetic algorithm for multi-objective optimization of the home health care problem. Swarm Evol. Comput, 44, 712–727.

Edie, L. C. (1954). Traffic delays at toll booths. Journal of the Operations Research Society of America, 2

(2), 107–138.
Elshaer, R., & Awad, H. (2020). A taxonomic review of metaheuristic algorithms for solving the vehicle routing problem and its variants. Comput. Ind. Eng, 140.
Emmerich, M. T. M., & Deutz, A. H. (2018). A tutorial on multiobjective optimization: fundamentals and evolutionary methods. Nat. Comput, 17 (3), 585–609.
Fathollahi-Fard, A. M., Ahmadi, A., & Karimi, B. (2021). Multi-Objective Optimization of Home Healthcare with Working-Time Balancing and Care Continuity. Sustainability, 13 (22), 12431–12431.
Frazzon, E. M., Albrecht, A., Pires, M., Israel, E., Kück, M., & Freitag, M. (2018). Hybrid approach for the integrated scheduling of production and transport processes along supply chains. International Journal of Production Research, 56 (5), 2019–2035.
Gadhvi, B., Savsani, V., & Patel, V. (2016). Multi-objective optimization of vehicle passive suspension system using nsga-ii, spea2 and pesa-ii. Procedia Technology, 23, 361–368.
Gayraud, F., Deroussi, L., Grangeon, L., & Norre, S. (2013). A New Mathematical Formulation for the Home Health Care Problem. Procedia Technol, 9, 1041–1047.
Ghannadpour, S. F., Noori, S., Tavakkoli-Moghaddam, R., & Ghoseiri, K. (2014). A multi-objective dynamic vehicle routing problem with fuzzy time windows: Model, solution and application. Applied Soft Computing, 14 (1), 504–527.
Ghoseiri, K., & Ghannadpour, S. (2010). FMulti-objective vehicle routing problem with time windows using goal programming and genetic algorithm. Applied Soft Computing, 30 (2), 286–296.
Gupta, R., Singh, & Pandey, D. (2010). Multi-objective fuzzy vehicle routing problem: a case study. International Journal of Contemporary Mathematical Sciences, 5 (29), 1439–1454.
Hameed, I. A. (2020). Multi-objective Solution of Traveling Salesman Problem with Time. Advances in Intelligent Systems and Computing, 2020, 921, 121–132.
Harris, T., & Hardin, J. W. (2013). Exact wilcoxon signed-rank and wilcoxon mann–whitney ranksum tests. The Stata Journal, 13 (2), 337–343.
Hornstra, R. P., Silva, A., Roodbergen, K. J., & Coelho, L. C. (2020). The vehicle routing problem with simultaneous pickup and delivery and handling costs. Computers Operational Research, 115.
Hu, C., Lu, J., Liu, X., & Zhang, G. (2018). Robust vehicle routing problem with hard time windows under demand and travel time uncertainty. Computer Operation Research, 94, 139–153.
Iqbal, S., Kaykobad, M., & Rahman, M. S. (2015). Solving the multi-objective Vehicle Routing Problem with Soft Time Windows with the help of bees. Swarm and Evolutionary Computation, 24, 50–64. Issabakhsh, M., Hosseini-Motlagh, S. .-., Pishvaee, M., & Nia, S. (2018). A Vehicle Routing Problem for Modeling Home Healthcare: a Case Study. International Journal of Transportation Engineering, 5 (3), 211–228.
Issaoui, B., I Zidi, Marcon, E., & Ghedira, K. (2015). new multi-objective approach for the home care service problem based on scheduling algorithms and variable neighborhood descent. Electron. Notes Discret. Math, 47, 181–188.
Jayamoorthi, K., Karunanidy, D., Jayavel, A., & Ramalingam, S. (2017). A Survey On Multi-Objective Travelling Salsman Problem. Iioab Journal, 8, 223–233.
Jemai, J., Chaieb, M., & Mellouli, K. (2013). The Home Care Scheduling Problem: A modeling and solving issue. 2013 5th International Conference on Modeling, Simulation and Applied Optimization.
Kaiwartya, O., Kumar, S., Lobiyal, D. K., Tiwari, P. K., Abdullah, A. H., & Hassan, A. N. (2015). Multi-objective Dynamic Vehicle Routing Problem and Time Seed Based Solution Using Particle Swarm Optimization. J. Sensors, 189832, 14–14.
Kalayci, C. B., & Kaya, C. (2016). An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery. Expert Systems with Applications, 66, 163–175.
Kaucic, M., Moradi, M., & Mirzazadeh, M. (2019). Portfolio optimization by improved nsga-ii and spea 2 based on different risk measures. Financial Innovation, 5 (1), 1–28.
King, R. A., Deb, K., & Rughooputh, H. (2010). Comparison of nsga-ii and spea2 on the multiobjective environmental/economic dispatch problem. University of Mauritius Research Journal, 16 (1), 485– 511.
Király, A., Abonyi, & J. (2011). Intelligent Computational Optimization in Engineering. In M. Köppen, G. Schaefer, A. Abraham, et al. (Eds.), Techniques and applications (pp. 241–269). Springer.
Kumar, S., & Panneerselvam, R. (2012). A Survey on the Vehicle Routing Problem and Its Variants. Intelligent Information Management, 4, 66–74.
Kunkle, D. (2005). A Summary and Comparison of MOEA Algorithms.

Lee, J. S., Choi, L. C., & Park, S. C. (2011). Multi-objective genetic algorithms, nsga-ii and spea2, for document clustering. International Conference on Advanced Software Engineering and Its Applications, 219–227.

Li, P., He, J., Zheng, D., Huang, Y., & Fan, C. (2015). Vehicle routing problem with soft time windows based on improved genetic algorithm for fruits and vegetables distribution. Discrete Dynamics in Nature and Society.

Lin, M. Y., Chin, K. S., & Tsui, K. L. (2015). A weighted multi-objective therapist assignment problem in Hong Kong. 2015 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), 1323–1327.

Liu, W.-L., Gong, Y.-J., Chen, W.-N., Liu, Z., Wang, H., & Zhang, J. (2019). Coordinated charging scheduling of electric vehicles: A mixed-variable differential evolution approach. IEEE Transactions on Intelligent Transportation Systems, 21 (12), 5094–5109.

Macqueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability, 1, 281–297.

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

Marinakis, Y., & Marinaki, M. (2015). Combinatorial Neighborhood Topology Bumble Bees Mating Optimization for the Vehicle Routing Problem with Stochastic Demands. Soft Computing, 19, 353– 373.

Mascolo, M. D., Martinez, C., & Espinouse, M. (2021). Routing and scheduling in Home Health Care: A literature survey and bibliometric analysis. Computers & Industrial Engineering, 158, 107255– 107255.

Melián-Batista, B., De Santiago, A., AngelBello, F., & Alvarez, A. (2014). A bi-objective vehicle routing problem with time windows: A real case in Tenerife. Applied Soft Computing, 17, 140–152.

Miranda, D., & Conceição, S. V. (2016). The vehicle routing problem with hard time windows and stochastic travel and service time. Expert Systems with Applications, 64, 104–116.

Montoya-Torres, J. R., Franco, J. L., Isaza, S. N., Jiménez, H. F., & Herazo-Padilla, N. (2015). A literature review on the vehicle routing problem with multiple depots. Computers & Industrial Engineering, 79, 115–129.

Mousa, A. A., El-Shorbagy, M. A., & Farag, M. A. (2017). K-means-Clustering Based Evolutionary Algorithm for Multi-objective Resource Allocation Problems. Applied Mathematics & Information Sciences, 11 (6), 1681–1692.

Nagy, G., Wassan, N. A., Speranza, M. G., & Archetti, C. (2015). The vehicle routing problem with divisible deliveries and pickups. Transportation Science, 49 (2), 271–294.

Nickel, S., Schröder, M., & Steeg, J. (2009). Planning for home health care services. Fraunhofer Institute for Industrial Mathematics.

Özder, E. H., Özcan, E., & Eren, T. (2020). A Systematic Literature Review for Personnel Scheduling Problems. Int. J. Inf. Technol. Decis. Mak., 19, 1695–1735.

Pappalardo, M. (2008). Multiobjective Optimization: A Brief Overview, Springer Optimization and Its Appli- cations. Altannar Chinchuluun & Panos M. Pardalos & Athanasios Migdalas & Leonidas Pitsoulis, 517–528.

Pinchera, D., Perna, S., & Migliore, M. (2017). A Lexicographic Approach for Multi-Objective Optimization in Antenna Array Design. Progress In Electromagnetics Research M, 59, 85–102.

Schott, J. R. (1995). Fault tolerant design using single and multicriteria genetic algorithm optimization (Doctoral dissertation). Massachusetts Institute of Technology.

Shi-Yi, T., & Wei-Chang, Y. (2021). The Vehicle Routing Problem: State-of-the-Art Classification and Review. Applied Sciences, 11 (21), 10295.

Srinivas, N., & Deb, K. (1995). Multiobjective function optimization using nondominated sorting genetic algorithms. Evol. Comput, 2 (3), 221–248.

Tlahig, H., Jebali, A., & Bouchriha, H. (2009). A two-phased approach for the centralisation versus decentral- isation of the hospital sterilisation service department. European Journal of Industrial Engineering, 3 (2), 227–246.

Wang, J., Zhou, Y., Wang, J., Zhang, C., Chen, P., & Z Zheng. (2016). Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: formulation, instances and algorithms. IEEE Transactions on Cybernetics, 46 (3), 582–594.

Wilcoxon, F. (1992). Individual comparisons by ranking methods. In S. Kotz & N. L. Johnson (Eds.), Breakthroughs in statistics: Methodology and distribution (pp. 196–202). Springer New York. https://doi.org/10.1007/978-1-4612-4380-9_16

Xiang, T., Li, Y., & Szeto, W. Y. (2021). The daily routing and scheduling problem of home health care: Based on costs and participants' preference satisfaction. International Transactions in Operational Research.

Yu, X., & Gen, M. (2010). Introduction to Evolutionary Algorithms. In and others (Ed.). Springer-Verlag. Zhao, F., He, X., & Wang, L. (2020). A two-stage cooperative evolutionary algorithm with problem-specific knowledge for energy-efficient scheduling of no-wait flow-shop problem. IEEE transactions on cy-bernetics, 51 (11), 5291–5303.

Zhao, F., Ma, R., & Wang, L. (2021). A self-learning discrete jaya algorithm for multiobjective energy-efficient distributed no-idle flow-shop scheduling problem in heterogeneous factory system. IEEE Transactions on Cybernetics.

Zhou, S., Xing, L., Zheng, X., Du, N., Wang, L., & Zhang, Q. (2019). A self-adaptive differential evolution algorithm for scheduling a single batch-processing machine with arbitrary job sizes and release times. IEEE transactions on cybernetics, 51 (3), 1430–1442.

Zitzler, E., Laumanns, M., & Thiele, L. (2001). Spea2: Improving the strength pareto evolutionary algorithm. eurogen 2001. evolutionary methods for design, optimization and control with applications to industrial problems.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: a comparative case study and the strength pareto approach. IEEE Transactions on Evolutionary Computation, 3 (4), 257–271.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on evolutionary computation, 7 (2), 117–132.

Zou, S., Li, J., & Li, X. (2013). A Hybrid Particle Swarm Optimization Algorithm for Multi-Objective Pickup and Delivery Problem with Time Windows. JCP, 8 (10), 2583–2589.