www.elsevier.com/locate/caor

A Decomposition Based Memetic Algorithm for Multi-objective Vehicle Routing Problem with Time Windows

Yutao Qi, Zhanting Hou, He Li, Jianbin Huang, Xiaodong Li

PII: S0305-0548(15)00089-1
DOI: http://dx.doi.org/10.1016/j.cor.2015.04.009
Reference: CAOR3765

To appear in: *Computers & Operations Research*

A Decomposition Based Memetic Algorithm for Multi-objective Vehicle Routing Problem with Time Windows

Yutao Qi[a], Zhanting Hou[a], He Li[b], Jianbin Huang[b], Xiaodong Li[c]

[a]School of Computer Science and Technology, Xidian University, Xi'an, China
[b]School of Software, Xidian University, Xi'an, China
[c]School of Computer Science and Information Technology, RMIT University, Melbourne, Australia

Abstract

Multi-objective evolutionary algorithm based on decomposition(MOEA/D) provides an excellent algorithmic framework for solving multi-objective optimization problems. It decomposes a target problem into a set of scalar sub-problems and optimizes them simultaneously. Due to its simplicity and outstanding performance, MOEA/D has been widely studied and applied. However, for solving the multi-objective vehicle routing problem with time windows (MO-VRPTW), MOEA/D faces a difficulty that many sub-problems have duplicated best solutions. It is well-known that MO-VRPTW is a challenging problem and has very few Pareto optimal solutions. To address this problem, a novel selection operator is designed in this work to enhance the original MOEA/D for dealing with MO-VRPTW. Moreover, three local search methods are introduced into the enhanced algorithm. Experimental results indicate that the proposed algorithm can obtain highly competitive results on Solomon's benchmark problems. Especially for instances with long time windows, the proposed algorithm can obtain more diverse set of non-dominated solutions than the other algorithms. The effectiveness of the proposed selection operator is also demonstrated by further analysis.

Keywords: Multi-objective optimization, memetic algorithm, decomposition, vehicle routing problem with time windows

1. Introduction

Vehicle routing problem with time windows (VRPTW) is an important variant of vehicle routing problem (VRP) which is an extensively studied combinational optimization problem and has been used in many real applications. The VRPTW can be defined as designing routes for a fleet of vehicles with limited capacity to serve a set of customers with known demands and predefined time window. Each route starts from and ends at the central depot. Each customer is visited once and only once by exactly one vehicle [1]. Due to its complexities and usefulness, VRPTW has been attracting much research effort. Many exact methods [2], heuristic and meta-heuristic methods [3] have been developed. However, most of existing works focus on VRPTW with single optimization objective. In real-life, there are several optimization objectives related to the tours of VRPTW, such as the number of used vehicles, total traveled distance, makespan or traveling time of the longest route, total waiting time, total delay time and so on [4]. Therefore, it is necessary to develop optimization techniques for multi-objective VRPTW (MO-VRPTW) to provide the decision makers with more comprehensive information about the target problems.

At present, many research efforts have been devoted to solving MO-VRPTW by using multi-objective optimization techniques. Hong and Park [5] constructed a linear goal programming model for the bi-objective VRPTW and developed a heuristic algorithm to reduce the computational burden inherent to the application of the model. Gehring et al. and Alvarenga et al. developed two-phase approaches for the MO-VRPTW, the first phase minimizes the number of vehicles and the second phase minimizes the total distance [6] or the total traveling time [7]. Rahoual et al. [8] solved the VRPTW by using the non-dominated sorting genetic algorithm (NSGA) with elitist and sharing strategy, which was the first work that employed the multi-objective evolutionary algorithm (MOEA)to solve MO-VRPTW. Ombuki et al. [9] investigated the effectiveness of solving VRPTW by using a multi-objective optimization model and multi-objective genetic algorithm. Tan et al. [10] introduced the heuristics methods into MOEA to perform local exploitation and proposed a hybrid multi-objective evolutionary algorithm for MO-VRPTW. Geiger et al. [11] proposed an interactive multi-objective approach with variable neighborhood search to solve various components of general vehicle routing problems including MO-VRPTW. Xu et al. [12] introduced a hybrid algorithm to solve tri-objective VRPTW. Garcia-najera et al. [13] first considered the similarity of solutions in solving MO-VRPTW. They designed a similarity measure between solutions and introduced it into multi-objective optimization approach to obtain better diversity[14]. Gong [15] developed multi-objective particle swarm optimization (PSO) algorithms for solving MO-VRPTW. Ghoseiri and Ghannadpour [16] investigated a bi-objective VRPTW and developed a multi-objective genetic algorithm for solving it. Chiang et al. [17] incorporated problem-specific knowledge into the genetic operators and developed an evolutionary algorithm for MO-VRPTW. Melian-Batista et al. [18] considered both the total traveled distance and traveling time balance, and developed a solution approach based on the scatter search metaheuristic for real-world MO-VRPTW. Banos et al. [19] combined evolutionary computation with simulated annealing and developed a hybrid algorithm for MO-VRPTW that considers both the traveled distance and the workload balance. The abovementioned works have illustrated the superiority of solving VRPTW by using multi-objective optimization models.

By evolving a population of solutions, multi-objective evolutionary algorithms (MOEAs) are able to approximate the Pareto optimal set in a single run. MOEAs have been very successful in solving multi-objective optimization problems, and a lot of research efforts have been made in this area [20]. More recently, Zhang and Li [21] combined decomposition methods and the evolutionary computation together, and proposed a multi-objective evolutionary algorithm based on decomposition (MOEA/D). MOEA/D provides an excellent general evolutionary algorithmic framework for multi-objective optimization. It has attracted increasing research interests. MOEA/D decomposes the multi-objective optimization problem into a set of scalar optimization problems, while evolutionary algorithm is applied to optimize the scalar subproblems simultaneously.

Due to its simplicity and outstanding performance, MOEA/D has been investigated widely and applied successfully on various continuous and discrete MOPs. MO-VRPTW is a special type of MOP which has one discrete target like the number of used vehicles, and some continuous targets like total traveled distance, total cost of routings, makespan or traveling time of the longest route. Because of this, MO-VRPTW has a discrete Pareto front which is composed of very few Pareto optimal solutions. The above features of MO-VRPTW imply that MOEA/D will have many subproblems with duplicated best solutions when solving MO-VRPTW. In this work, in order to obtain a good diversity of solutions, a novel selection operator is designed to enhance MOEA/D for dealing with MO-VRPTW.

On the other hand, it is a major advantage of MOEA/D that single objective local search methods can be used in the optimization of each subproblem in a natural way [20]. Although there are many suggested meta-heuristic methods for VRPTW, three effective methods are selected to form a memetic MOEA/D (M-MOEA/D) in this work. Memetic algorithm (MA) is a type of population based meta-heuristics algorithm. It is composed of an evolutionary framework and a set of local search methods that are activated within the generation cycle of the external framework [22]. It has been applied to solve a wide variety of optimization problems, including the classical combinatorial optimization problem VRPTW [23]. However, few efforts have been devoted to Memetic algorithm for MO-VRPTW [10][12].

The goal of this paper is to enhance MOEA/D for dealing with MO-VRPTW. We propose a memetic MOEA/D with a novel selection operator (M-MOEA/D) for MO-VRPTW. M-MOEA/D enhances the original MOEA/D in the following two aspects:

1. A specially designed selection operation for updating the current best solution of each subproblem is developed according to the characteristics of the MO-VRPTW.
2. Three types of local search methods which have different searching behaviors are employed periodically to from a memetic algorithm.

The rest of this paper is organized as follows. Section 2 describes the mathematical model of the MO-VRPTW. Section 3 presents the framework of the proposed M-MOEA/D. Based on the benchmarks of Solomon's 56 data sets, section 4 verifies the effectiveness of the proposed M-MOEA/D and presents further analysis on it. Finally, conclusion is given in section 5.

2. A Multi-objective Optimization Model for the VRPTW

The VRPTW can be defined as follows: n customers are waiting to be served, each of which requires a quantity of goods. The central depot has a fleet of vehicles to deliver the goods. Each vehicle has the same capacity, which must be greater or equal to the total of all demands on the route traveled by one vehicle. Besides, each customer must be visited once and only once by exactly one vehicle. The time window constraint is denoted by a predefined time interval which can be described by the earliest arrival time and the latest arrival time. The vehicles have to arrive at the customers before the latest arrival time, while arriving earlier than the earliest arrival time, waiting occurs. Each customer imposes an additional service time to the route, taking into consideration the loading or unloading time of goods. Each vehicle is supposed to complete its individual route within the time window of the depot. A solution of the VRPTW is a collection of routes in which a vehicle starts from the depot, visits customer nodes and then returns to depot under capacity and time window constraints. The number of routes in the network is equal to the number of vehicles used, one vehicle is dedicated to one route.

The objective of the MO-VRPTW is to obtain the minimum distance traveled by the vehicles and the minimum total number of vehicles used to serve the customers while the total customers have been served. The mathematical representation of the MO-VRPTW can be described as follows. Define $G = (V, A)$ as a directed complete graph, where $V = \{c_0, c_1, ..., c_n\}$ is the node set, and $A = \{\langle c_i, c_j \rangle | c_i, c_j \in V, i \neq j\}$ is the arc set. c_0 represents the depot, and $c_i (i = 1, 2, ...n)$ represents the customer. Each vehicle has the same capacity Q. Each node is associated with a demand quantity q_i among which node c_0 is associated with $q_0 = 0$. Each arc in the network represents a connection between two nodes and also indicates the direction it travels. Each arc is associated with a traveling time t_{ij}, which is proportional to the Euclidean distance d_{ij} between nodes c_i and c_j. A time window $[e_i, l_i]$ during the time when the service has to be started is considered. If the vehicle arrives at the node c_i before e_i, it will wait at the node until e_i, and then, a service time s_i is considered. For depot c_0, the time window is defined as that e_0 is the earliest start time, and l_0 is the latest return time of all the vehicles. When the vehicle arrives at customer c_i within the time windows, it has to stay at the location of customer c_i for a time interval at least s_i for service. Let K represents the number of the total used vehicles.

The optimization model for the MO-VRPTW can be mathematically defined as follows:

$$\begin{cases} \min F(x) = (f_1, f_2) \\ f_1 = K \\ f_2 = \sum_{i=1}^{N} \sum_{j=0, j\neq i}^{N} \sum_{k=1}^{K} d_{ij} x_{ijk} \end{cases} \tag{1}$$

Subject to:

$$x_{ijk} = \begin{cases} 1 & \text{if arc } \langle c_i, c_j \rangle \text{ is traversed by vehicle } k \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

$$\sum_{j=1}^{N} x_{0jk} = \sum_{i=1}^{N} x_{i0k} = 1, k \in \{1, 2, ..., K\} \tag{3}$$

$$\sum_{j=0, j\neq i}^{N} x_{ijk} = \sum_{j=0, j\neq i}^{N} x_{jik} \leq 1, i \in \{1, 2, ..., N\}, k \in \{1, 2, ..., K\} \tag{4}$$

$$\sum_{k=1}^{K} \sum_{i=0, i\neq j}^{N} x_{ijk} = 1, j \in \{1, 2, ..., N\} \tag{5}$$

$$\sum_{k=1}^{K} \sum_{j=0, j\neq i}^{N} x_{ijk} = 1, i \in \{1, 2, ..., N\} \tag{6}$$

$$\sum_{i=0}^{N} q_i \sum_{j=0, j\neq i}^{N} x_{ijk} \leq Q, k \in \{1, 2, ..., K\} \tag{7}$$

$$t_i + w_i + s_i + t_{ij} = t_j, i, j \in \{1, 2, ..., N\}, i \neq j \tag{8}$$

$$e_i \leq (t_i + w_i) \leq l_i, i \in \{0, 1, ..., N\} \tag{9}$$

Expression (1) refers to minimizing two goals, f_1 and f_2, which are the number of used vehicles and the total traveled distance respectively. x_{ijk} denotes whether the arc $\langle c_i, c_j \rangle$ is traversed by the k-th vehicle, as is defined in the expression (2). Constraints (3) and (4) represent that each vehicle starts the delivery task from the depot, visits customers one by one and then returns the depot. Constraint (5) and (6) define that each customer node is visited only once by one vehicle. Constraint (7) denotes that the quantity of goods that each vehicle carries could not exceed the capacity Q. Equation (8) describes how the travel time is computed, and constraint (9) defines the time window constraint. In these two equations, t_i is the time when vehicle arrives at customer i. w_i is the waiting time of the vehicle at customer's location, it can be determined by the time interval between t_i and e_i. s_i is the service time. t_{ij} is the traveling time between nodes c_i and c_j.

3. The M-MOEA/D Algorithm

MOEA/D provides an excellent general algorithmic framework for solving MOP. It decomposes the target MOP into a set of scalar subproblems with uniformly distributed aggregation weight vectors. MOEA/D minimizes these scalar subproblems simultaneously by evolving a population of solutions. At each generation, the population is composed of the current best solutions for every subproblem so far. Each subproblem has a neighborhood list, it can be optimized by using information from its several neighboring subproblems.

M-MOEA/D follows the framework of MOEA/D. It decomposes the target MO-VRPTW into a number of scalar subproblems by using the Tchebycheff approach. Given a reference point $z^* = (z_1^*, z_2^*, \cdots, z_m^*)$ and a weight vector $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_m)(\sum_{i=1}^{m} \lambda_i = 1, \lambda_i \geq 0, i = 1, \cdots, m)$, the Tchebycheff approach decomposes a MOP into the following single objective subproblem, in which Ω is the decision space.

$$\min_{\mathbf{x}\in\Omega} g^{te}(\mathbf{x}|\lambda, z^*) = \max_{1\leq i\leq m} \{\lambda_i |f_i(\mathbf{x}) - z_i^*|\} \tag{10}$$

3.1. Chromosome Representation

In M-MOEA/D , a variable-length chromosome representation for VRPTW which is developed by Tan et al. [10] is employed. In the coding method, each chromosome consists of several routes and each route is a sequence of customers to be served. Fig.1 illustrates the data structure of the chromosome representation. It can be seen that each chromosome encodes a complete solution including the number of routes and the customers served by these vehicles.

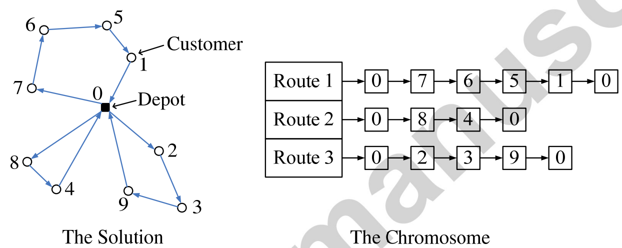

Figure 1: Illustration of the chromosome representation.

3.2. The Framework of M-MOEA/D

M-MOEA/D uses a specially designed selection operator and employs three local search methods to improve the quality of the solution to each scalar subproblem periodically. It works as follows:

Algorithm 1 M-MOEA/D Algorithm

Input:

The MO-VRPTW (1);
$MaxFunEvals$: The maximum number of function evaluations;
N: The number of the subproblems;
p_c: The crossover probability;
p_m: The mutation probability;
$\{\lambda^1, \lambda^2, \ldots, \lambda^N\}$: A set of uniformly distributed weight vectors;
T: The neighborhood list size;
EP: An external population.

Output:
The external population EP.

Step 1 Initialization:

1.1 Decompose the original MO-VRPTW into N scalar subproblems with weight set $\lambda^1, \lambda^2, \ldots, \lambda^N$ using the Tchebycheff approach;

1.2 Generate an initial population $POP = \{x^1, x^2, \ldots, x^N\}$ by the individual constructing process, evaluate initial individuals and perform objective normalization to get $FV^1, FV^2, \ldots, FV^N$, $FV^i = (FV_1^i, ..., FV_m^i)$ is the normalized target value vector of individual x^i. Each $FV_j^i (j = 1, ..., m)$ can be calculated by the following equation (11).

$$FV_j^i = \begin{cases} (f_j(x^i) - min_j)/(max_j - min_j) & \text{if } max_j \neq min_j \\ 1 & \text{otherwise} \end{cases} \tag{11}$$

In which $f_j(x^i)$ is the j-th target value of x^i, min_j and max_j are respectively the minimum and the maximum value of j-th target value of all the individuals in POP. Distribute each individual x^i to the i-th subproblem as its current best solution;

1.3 Calculate the Euclidean distances between any two weight vectors and then work out the T closest weight vectors to each weight vector. For each $i = 1, 2, \ldots, N$, set $B(i) = \{i_1, i_2, \ldots, i_T\}$, where $\lambda^{i_1}, \lambda^{i_2}, \ldots, \lambda^{i_T}$ are the T closest weight vectors to λ^i;

1.4 Set $z^* = (0, 0)$, set EP as an empty set.

Step 2 Reproduce: Construct a temporary population POP', and set it as empty. For $i = 1, 2, \ldots, N$, do

2.1 Randomly select two indices k,l from $B(i)$;

2.2 Apply the crossover operator on individuals x^{i_k} and x^{i_l} with probability p_c to generate individual $\widetilde{y}^i$;

2.3 Apply the mutation operator on $\widetilde{y}^i$ with probability p_m, giving rise to individual y^i;

2.4 Perform local search exploitation periodically. At the end of every 20 iterations, apply swap, lambda interchange or single route 2-opt local search on the individual y_i with the same probability to get a new generated individual $\widehat{y}^i$;

2.5 Evaluate y^i and $\widehat{y}^i$, normalize their objective values. If $g^{te}(\widehat{y}^i|\lambda^i, z^*) < g^{te}(y^i|\lambda^i, z^*)$, then let $y^i = \widehat{y}^i$;

2.6 $POP' = POP' \cup y^i$.

Step 3 Selection:

3.1 Let $POP' = POP' \cup POP$;

3.2 Update current best solution of each subproblem to form a new POP. First, select a subproblem at random, note as the i-th subproblem with weight $\lambda^i (i = 1, 2, \ldots, N)$. Choose the best individual of the i-th subproblem in sequence from current POP' and note it as y^i, that is $y^i = \arg\min_{y \in POP'} g^{te}(y|\lambda^i, z^*)$. If more than one individuals have the same values of $g^{te}(y|\lambda^i, z^*)$, let y^i be the one who has the fewest total traveled distance or the smallest number of used vehicles. Then replace current best solution of the i-th subproblem x^i with y^i and remove y^i from current POP'. This updating procedure repeats until all the subproblems have been updated.

3.3 Update of EP. Let $EP = EP \cup POP$, then remove all the dominated individuals from EP.

Step 4 Stopping Criteria: If stopping criteria is satisfied, then stop and output EP. Otherwise, go to Step 2.

In order to solve the MO-VRPTW which is an order-based multi-modal and multi-objective optimization problem, specialized initialization and genetic operators are employed in the proposed M-MOEA/D. M-MOEA/D also incorporates three local search operations which can contribute to intensification of the optimization results. The details of these components are provided in the following sections.

3.3. Population Initialization

At the population initialization step, initial individuals are generated by the following individual constructing process which guarantees the feasibility of the initial individuals. The process of constructing individual works as follows. First, a customer is selected randomly and it is placed as the first location to be visited on the first route. Then, a different random customer is chosen, if the capacity and time constraints are met, it is placed on the current route after the previous customer. If none of the constraints is satisfied, a new route is created and this customer is the first location to be visited on that route. This process is repeated until all customers have been assigned to a route.

3.4. Crossover Operation

M-MOEA/D employs the sequence based crossover (SBX)[24] as a crossover operator in step 2.2. Fig.2 illustrates how to apply SBX operator to create a new offspring individual.

Given two parent individuals P_1 and P_2, SBX operator randomly selects one route from each parent individual, noted as R_1 and R_2. Next, the randomly selected links are removed from R_1 and R_2 to split each of them into two sub-routes, say $R_1 = (R_{11}, R_{12})$ and $R_2 = (R_{21}, R_{22})$. Then, the customers that are serviced before the break point on the route of R_1 are linked to the customers that are serviced after the break point on the route of R_2 (c.f. the black nodes). Thus, the new generated route in the offspring individual is $R_{new} = (R_{11}, R_{22})$. If R_{new} meet the capacity and time window constraints, the old route R_1 in the parent individual P_1 is replaced by the new route R_{new}. Otherwise, another two randomly selected links in R_1 and R_2 are removed, the process is repeated to generate another R_{new}. If the recombination repeats more than 20 times before finding a feasible R_{new}, another two parent individuals are selected from the population and the crossover operation is reapplied. In a feasible solution, customers with early and late time windows are respectively scheduled at the beginning and the end of the route. By linking the first customers on route R_1 and the last customers on route R_2, the time window constraints are likely to be satisfied.

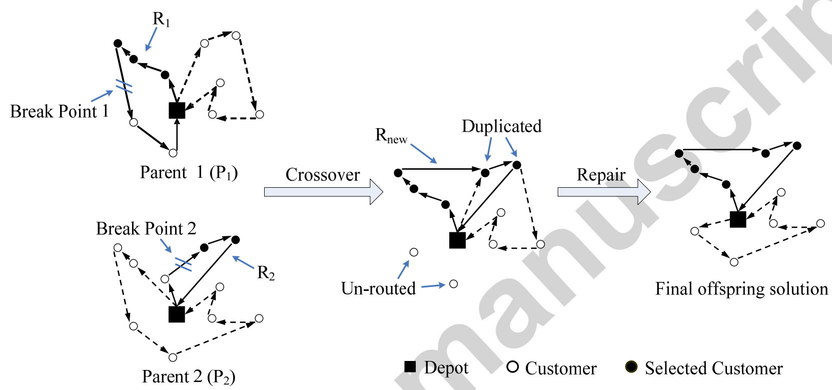

Figure 2: Illustration of the crossover operation.

The newly generated individual might be an invalid solution, as it is possible that customers are duplicated or un-routed in the new route. For example, in Fig.2 two customers appear more than once in two different routes, and two other customers are no longer served in the process. Thus, a repair operation is needed. If a customer appears twice in the new route, one of the two copies is removed from the route. If a customer appears in the new route and once old route at the same time, the customer is removed from the old route. If a customer is un-routed, the missing customer is reinserted into the new route. The deletion of customers from routes will only incur some waiting time before the next customer is serviced, and it will not violate any time window or capacity constraint. However, the re-insertion process may induce violation in capacity or time window constraints. Hence, each missing customer is reinserted into a position which will not cause any constraints violation. If there is no way to insert such a remaining customer into the existing routes without violating any constraint, a new route is created. If there are multiple positions satisfy the constraints, one of them will be chosen arbitrarily. Therefore, all chromosomes will remain feasible after the crossover operation in M-MOEA/D.

3.5. Mutation Operation

As for the mutation, M-MOEA/D employs the remove and reinsert mutation operator proposed by Garcia-Najera and Bullinaria in [14]. This mutation method involves three basic functions and three operators. The three basic functions are select route, select customer and insert customer. These functions are used by three mutation operators such as reallocation, exchange and reposition operator. The select route function stochastically selects a route from an individual, the routes with a larger travel distance and fewer customers are more likely to be chosen. The select customer function stochastically selects one customer from a specific route, the customers with longer associated travel distances are more likely to be selected. The insert customer function tries to insert a set of customers into a specific route where no violation of time window or capacity constraint is caused and the lowest travel distance is obtained. If no route is specified, all existing routes are tested.

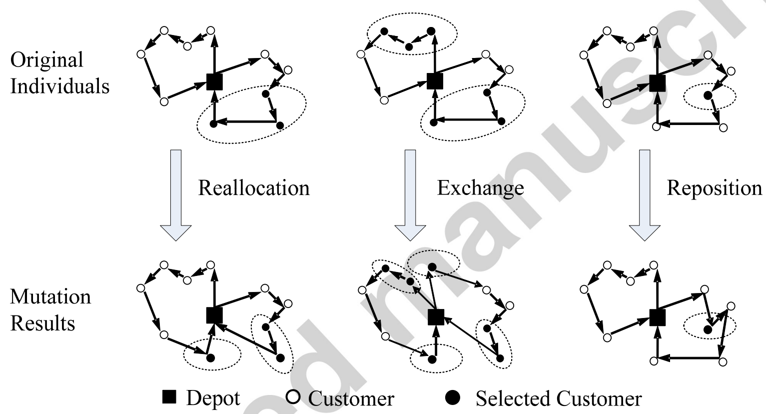

Figure 3: Illustration of the mutation operations.

Fig.3 is an illustration of the three mutation operators. The reallocation operator takes a number of customers from a given route and allocates them to another. In other words, two customers are selected from the route and removed from the route, along with all those customers in between them. Then, the removed customers are reallocated to the existing routes, including the one that they were removed from. The exchange operator swaps sequences of customers between two selected routes. First, two customers are selected from each route. The sequences of customers between them are then removed from their routes. Then, these removed customers are reallocated into the other routes. If one or more customers cannot be inserted into the other routes, the original routes are preserved. The reposition operator selects one customer from a specific route and reinserts it into the same route.

The mutation step works as follows: Two routes are selected using select route function. If they are the same route, the reallocation operator is performed, otherwise the exchange operator is executed. Then, another route is selected by using select route function and the reposition operator is performed.

The mutation operation removes some customers and reinserts them into another better position which will not cause any violation in capacity or time window constraints. If no such position can be found, the original route is not changed. Therefore, all chromosomes will remain feasible after the mutation operation of M-MOEA/D.

3.6. Local Search Operations

As has been investigated by Tan et al. [10], incorporating local search methods into multi-objective evolutionary algorithms for VRPTW can significantly improve the quality of individuals in the population. The local search methods can be used to enhance the multi-objective evolutionary algorithm for better convergence and discovering missing trade-off regions. Three effective local search methods are incorporated in the proposed M-MOEA/D to search for better routing solutions with the MO-VRPTW such as swap, lambda interchange [10] and single route 2-opt [25].

1. **Swap:** In the swap move, two candidate customers are selected from two different routes and their positions are exchanged. After the swapping is done, the feasibility of the newly generated routes is checked. If the two new routes are accepted, they will be updated as a part of the solution. Otherwise, the original routes will be restored. The procedure repeats until a pre-defined number of swapping is reached.
2. **Lambda interchange:** In the lambda interchange local search, a number of customers from its current position are extracted and inserted into a different route. Assuming that two routes A and B are selected to be changed, the lambda interchange local search starts from scanning customers in route A and moves a feasible customer into route B. The procedure repeats until a pre-defined number of customers is reached or the scanning ends at the last customer of route A.
3. **Single route 2-opt:** In the 2-opt for a single route, one route of the operated individual is selected randomly to perform the 2-opt move. After the 2-opt move is done, the feasibility of the new route is checked. If the new route is accepted, it will be updated as a part of the solution, otherwise the original route will be restored. The 2-opt move selects two edges which are not adjacent to each other from the operated route, thus the route breaks into two sub-routes, and then those two sub-routes are connected by reversing the direction of one sub-route. The 2-opt move exchanges two non-adjacent edges with another two new edges until the resulting route is feasible and better than the previous one. The exchange procedure repeats until a pre-defined number of 2-opt moving is reached or there is no feasible exchange can improve the current route.

There is no preference considered among the above local search methods. One of them will be randomly executed at the end of every 20 iterations for all individuals in the population to search better local solutions. From the description of the three local search methods, we can see that they will not induce any violation in capacity or time window constraints.

Moreover, the three local search methods used in M-MOEA/D have different searching behaviors. The swap move tries to exchange customers between different routes, and it helps to enhance the diversity of transport task allocation. The lambda interchange local search tries to move customers of a route to another one, it helps reduce the vehicle number. The single route 2-opt tries to exchange edges in a single route, it helps reduce the traveled distance of a single route. Therefore, the three local search strategies collaborate with each other, and they play different roles in the local search process of the M-MOEA/D.

3.7. Selection Operation

The selection operator in the original MOEA/D is based on the Tchebycheff metric which changes the target MOP into a series of scalar optimization problems by using a given set of weight vectors and a given reference point. MOEA/D with Tchebycheff approach performs well when dealing with MOPs with continuous objective functions [21]. However, the selection operator in the original MOEA/D does not work well for solving the MO-VRPTW, as it has a discrete optimization goal, that is the number of used vehicles.

Fig.4(a) is an example of the individuals of an evolving population distribution in the objective space. It can be seen that, many individuals have the same number of used vehicles. Solutions with small number of vehicles can violate the time window or capacity constraints, while solutions with large number of vehicles may have high total traveled distance. Thus, there are only few feasible solutions that are non-dominated individuals.

Fig.4(b) illustrates the contour lines of a scalar subproblem with weight $\lambda = (\lambda_1, \lambda_2)$ and the geometric relationship between its weight vector and optimal solution in the Tchebycheff approach. In this figure, we combine the objective space and the weight space together. According to Deb's analysis on MOEA/D using the Tchebycheff approach [26], the optimal solution of the scalar subproblem with weight vector λ is located on the lowest contour line. The optimal point is the intersection point of the target Pareto front and the straight line passes through the reference point $z*$ with the direction vector $\lambda' = (\frac{1}{\lambda_1}, \frac{1}{\lambda_2})$. Therefore, the direction vector λ' decides the location of the optimal solution in the objective space. The optimal solution of a scalar subproblem generated by Tchebycheff approach must be an non-dominated solution of the original multi-objective optimization problem.

As the MO-VRPTW has a small number of non-dominated solutions, the MOEA should not only maintains the non-dominated individuals in the population. As shown in Fig.4(c), two or more individuals may have the same objective values on the same scalar subproblem. On the other hand, two or more scalar subproblems may have the same optimal solution. Suppose A_1 and A_2 are two solutions with the same vehicle number, and A_1 has lower total traveled distance than A_2. Although solution A_1 weakly dominates A_2, they have the same objective values on the scalar subproblem whose direction vector is λ'_1. For the scalar subproblems with direction vectors λ'_2, λ'_3 and λ'_4, they have the same achievement scalarizing function value for A_1.

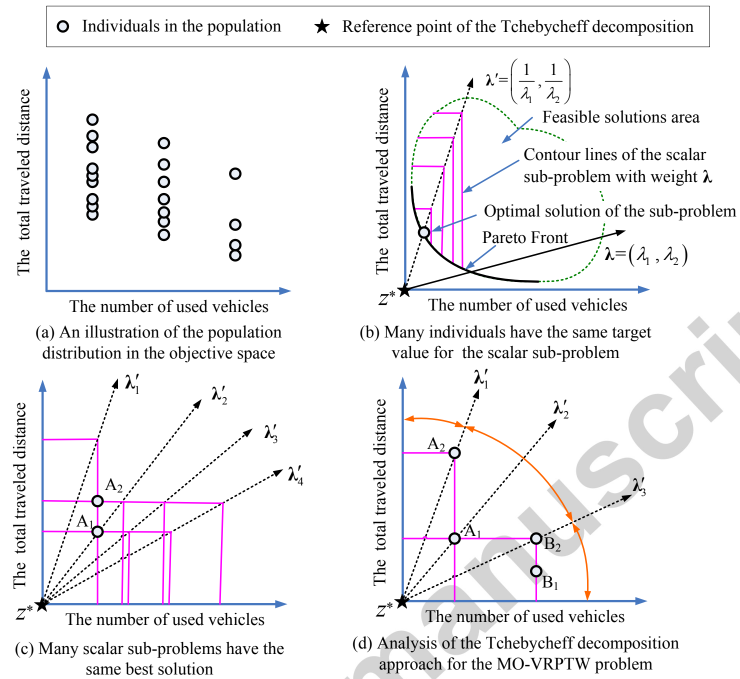

Figure 4: Analysis of the selection operation based on the Tchebycheff decomposition.

Fig.4(d) gives an analysis of the Tchebycheff decomposition approach for the MO-VRPTW. Let A_1, A_2 and B_2 are three solutions in the objective space. A_1 and A_2 have the same vehicle number, and A_1 has lower total traveled distance than A_2. A_1 and B_2 have the same total traveled distance, and A_1 has smaller vehicle number than A_2. Under these assumptions, we make a further discussion about the relationship between the optimal solution of a scalar subproblem and its direction vector λ'. Note that λ'_1, λ'_2 and λ'_3 are direction vectors that start from the reference point $z*$ and passes through the three solutions A_1, A_2 and B_2. Then, the scalar subproblems whose direction vectors lies between λ'_1 and λ'_3 will take A_1 as their optimal solution. When the direction vector of a scalar subproblem has higher slope than λ'_1, A_1 and A_2 will have the same achievement scalarizing function value on the subproblem. On the other hand, when the direction vector of a scalar subproblem has lower slope than λ'_3, A_1 and B_2 will have the same achievement scalarizing function value. M-MOEA/D selects A_1 which is a non-dominated solution as the optimal solution in all of the three cases described above. The selection operation selects the solution who has the lowest achievement scalarizing function value on the scalar subproblem as its optimal solution. If two or more individuals have the same achievement scalarizing function value, the non-dominated one will be selected. Therefore, if one solution dominates the other one, M-MOEA/D will select the non-dominated solution, which proves the correctness of the selection operation.

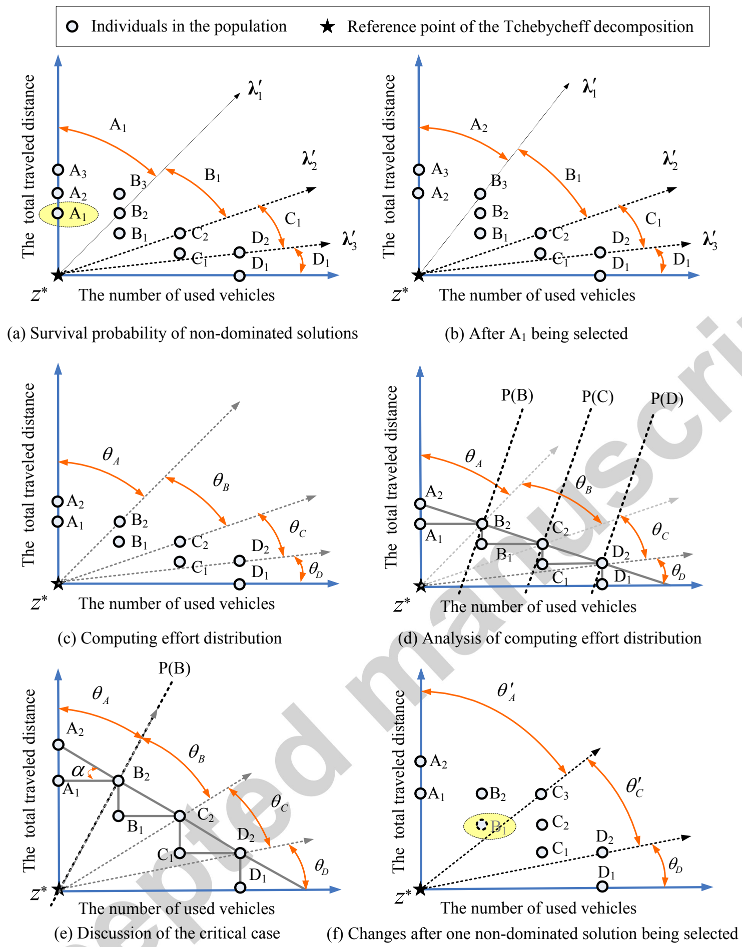

Figure 5: Analysis of the specially designed selection operation in M-MOEA/D.

Fig.5 gives a discussion about that the newly designed selection operation in M-MOEA/D is suitable for the MO-VRPTW. Suppose that $A = \{A_1, A_2, A_3, ...\}$, $B = \{B_1, B_2, B_3, ...\}$, $C = \{C_1, C_2, C_3, ...\}$, $D = \{D_1, D_2, D_3, ...\}$ ..., are sets of solutions, where the former set of solutions have fewer used vehicle number than those in the later set. A_1, B_1, C_1, D_1, ..., are non-dominated solutions with different used vehicle number. A_i and B_{i+1}, B_i and C_{i+1}, C_i and D_{i+1}, ..., $(i = 1, 2, ...)$ have the same total traveled distance. Fig.5(a) gives an illustration of the survival probability of the non-dominated solutions in the population. Subproblems with direction vectors lie between the vertical axis and λ_1' will take A_1 as their optimal solution. Subproblems will take B_1 as their optimal solution when their direction vectors lie between λ_1' and λ_2'. C_1 will be selected by subproblems whose direction vectors lie between λ_2' and λ_3'. D_1 will be selected by subproblems whose direction vectors lie between λ_3' and the horizontal axis. For ease of description, we call the two boundary vectors as the dominate boundary vectors of the non-dominated solution. As the direction vectors of scalar subproblems are evenly distributed in the first quadrant, the angle between the two boundary vectors of a non-dominated solution reflects its survival probability. The larger the angle, the higher the probability it survives, then the more computing effort will be devoted to it. As shown in this figure, A_1 has the largest survival probability. When A_1 is selected, the survival probability of B_1 becomes larger. The survival probability of A_2 is smaller than that of A_1, which is shown in Fig.5(b).

Fig.5(c-f) presents a theoretical analysis of the commutating effort distribution mechanism of the selection operation and its changes during the evolving process. In Fig.5(c), θ_A, θ_B, θ_C and θ_D are the angles between the two boundary vectors of non-dominated solutions A_1, B_1, C_1 and D_1 respectively. Note that $d(A, B)$ is the Euclidean distance between individuals A and B in the decision space, we can see that $d(A_1, B_2) = d(B_1, C_2) = d(C_1, D_2)$. To make it fair for each set of solutions which have different vehicle numbers, we suppose that $d(A_1, A_2) = d(B_1, B_2) = d(C_1, C_2) = d(D_1, D_2)$. Therefore, the three triangles $A_1A_2B_2$, $B_1B_2C_2$ and $C_1C_2D_2$ are congruent right triangles, A_2,B_2 ,C_2 and D_2 are collinear with each other. In Fig.5(d), we note the lines that are perpendicular to line A_2D_2 pass through B_2, C_2 and D_2 as $P(B)$, $P(C)$ and $P(D)$. It can be seen that if the reference point z^* is located at the A_2B_2 side of the line $P(B)$ rather than the B_2C_2 side, the angle θ_A will be larger than θ_B.If $\theta_A > \theta_B$, we can tell that z^* is located at the B_2C_2 side of $P(C)$ and the C_2D_2 side of $P(D)$, that is $\theta_B > \theta_C$ and $\theta_C > \theta_D$. Therefore, Fig.5(e) shows the critical case that line $P(B)$ passes though the reference point z^* and thus $\theta_A = \theta_B$. In this case, the triangles $z^*A_2B_2$ and $A_1A_2B_2$ are similar, therefore, the angle $\alpha = \theta_A$. With an increase of the angle α , the lines A_2D_2 and $P(B)$ rotate clockwise, z^* turns to the B_2C_2 side of the line $P(B)$, and thus the angle θ_A is smaller than θ_B. As shown in Fig.5(e), according to above analysis on this critical situation, $\alpha = \theta_A = \theta_B > \theta_C > \theta_D$. In addition $\theta_A + \theta_B + \theta_C + \theta_D = \pi/2$, we can conclude that $\alpha = \theta_A > \pi/8$. For ease of discussion, we term $d(A_1, A_2) = d(B_1, B_2) = d(C_1, C_2) = d(D_1, D_2)$ as the RC-gap between neighboring solutions in the solution set with the same used vehicles, and term $d(A_1, B_2) = d(B_1, C_2) = d(B_1, C_2) = d(C_1, D_2)$ as the NV-gap between neighboring solution sets. According to the objective normalization method described in Equation (11),the two optimization goals are normalized to a range [0,1]. Take the scenario in Fig.5(e) for example, the number of used vehicles has four different values, in this situation we can tell that the NV-gap $d(A_1, B_2) = 1/3$. The RC-gap $d(A_1, A_2) = d(A_1, B_2) \times \tan\alpha > 1/3 \times \pi/8 = 0.1381$, which means that the total traveled distance gap between two neighboring solutions is more than 13.81% of the range of total traveled distance values in current population. According to our observation, this situation rarely happens. Moreover, if the number of used vehicles has three different values, the NV-gap $d(A_1, B_2) = 1/2$, $\alpha > \pi/6$ and the RC-gap $d(A_1, A_2) = 0.2887$, and the percentage rises up to 28.87%. If the number of used vehicles has two different values, the NV-gap $d(A_1, B_2) = 1$, $\alpha > \pi/4$ and the RC-gap $d(A_1, A_2) = 1$, in this case, the percentage becomes 100%, which happen less frequently. If the number of used vehicles has only one value, then the problem becomes single objective optimization, this situation is not in the discussion of the selection operation for multi-objective optimization algorithm. It is reasonable to believe that $\theta_A > \theta_B > \theta_C > \theta_D$ holds true in most cases, which implies that the selection operation will devote more computing effort to solutions with smaller number of vehicles. That is what we want in solving the MO-VRPTW. When a non-dominated solution has been selected, the assignment of computing effort will be changed. As shown in Fig.5(f), after the non-dominated solution B_1 is selected, its remaining computing effort will be reallocated to its two neighbors A_1 and C_1. Thus, θ_A gets larger and becomes θ'_A, θ_C also becomes a larger angle θ'_C.

So far, we have discussed the correctness and effectiveness of the newly designed selection operation. One may ask why not use the fast non-dominated sorting procedure in the famous NSGA-II [26] to perform the selection? One reason is that the the fast non-dominated sorting procedure ranks individuals according to their dominance relationships and crowding distances. It does not tend to select individuals with smaller vehicle numbers, which are preferred by decision makers. Fig.6 gives an example to illustrate the difference between the proposed selection operator and the non-dominated sorting in NSGAII. In this figure, $E_3 \prec B_3$ means E_3 is more likely to survive than B_3. $Survival_A$ means the number of survivals in individual set A.

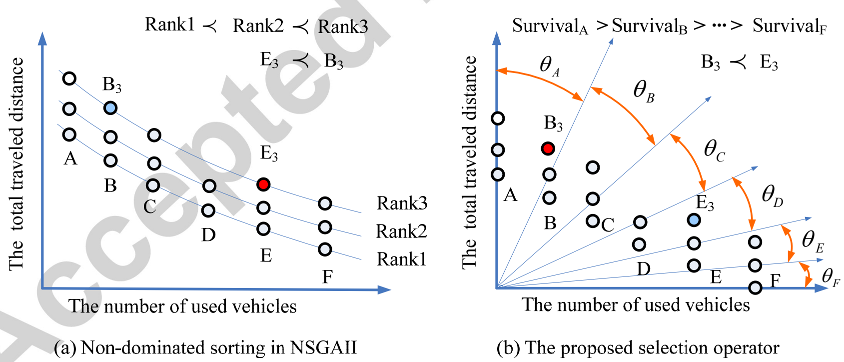

Figure 6: Comparison between the proposed selection operator and the non-dominated sorting in NSGAII.

The selection operator of the non-dominated sorting is based on the dominant relationship between solutions. Solutions with lower rank have higher survival probability. If two solutions have the same rank, the one with larger crowding distance (In Fig.6(a), E_3 has larger crowding distance than B_3) is more likely to survive. As for the proposed selection operator, as shown in Fig.6(b), it is a decomposition based selection method. The solution set with smaller number of used vehicles has larger number of survivals. Different from non-dominated sorting based selection operator, B_3 which has smaller number of used vehicles than E_3 is more likely to survive.

On the other hand, the newly designed selection operation in M-MOEA/D has lower time complexity than the fast non-dominated sorting. If we want to select m individuals from a population with size n, where $m < n$, the time complexity of the fast non-dominated sorting is $O(n^2)$. For the selection operation in M-MOEA/D, the time complexity is $O(m \times n)$. According to the step 3 of algorithm 1, $m = n/2$ in the proposed M-MOEA/D. That is to say, the time complexities of the two selection operations have the same order of magnitude. Despite of this, the actual execution time of the suggested selection operation is still fewer than that of the fast non-dominated sorting method. Therefore, there is a good reason to believe that the selection operation in M-MOEA/D is much more suitable to solve the MO-VRPTW.

4. Experimental Study

Solomon's benchmark [27] which has been widely used in these years is adopted to illustrate the superiority of the proposed M-MOEA/D. Solomon's problems consist of 56 data sets. All of the testing problems consist of 100 customers, fleet size, vehicle capacity, traveling time of vehicles, spatial and temporal distribution of customers. The traveling time from one customer to another is set as the corresponding Euclidean distance between them. Solomon's problems can be divided into six categories which is named as $C1$, $C2$, $R1$, $R2$, $RC1$ and $RC2$, based on the patterns of customers' locations and time windows. $C1$ and $C2$ problems have all customers located in geographical clusters. The problem category $R1$ and $R2$ refer to remote distribution of customers. $RC1$ and $RC2$ have a mix of remote and clustered customers. The problems in $C1$, $R1$ and $RC1$ generally have smaller time windows, while the problems in categories $C2$, $R2$ and $RC2$ often come with longer time windows. Solomon's benchmark problems have been studied in detail by Tan et al. [10]. They suggested that categories $C1$ and $C2$ have positively correlating objectives which means that the total traveled distance of a solution increases with the number of vehicles. They also found out that many of the instances in categories $R1$, $R2$, $RC1$ and $RC2$ have conflicting objectives.

In the experimental studies, we designed three groups of experiments to verify and analyze the effectiveness of the proposed M-MOEA/D. Firstly, we compared the solutions found by M-MOEA/D with the best known records to validate the feasibility and superiority of the proposed algorithm. Then, we compared M-MOEA/D with other two efficient and the latest developed multi-objective algorithms: MOGP by Ghoseiri et al. [16] and MOEA by Garcia-Najera [28] After that, the effectiveness of the two innovative components in

M-MOEA/D were analyzed in detail by comparing M-MOEA/D with its three variants. Finally, we made a comparison between the novel designed selection operator and the outstanding non-dominated sorting method. The results indicate that the proposed selection operator is more suitable for solving MOPs like MO-VRPTW which have one discrete optimization target and few Pareto optimal solutions.

The proposed M-MOEA/D was implemented in C++ and tested on a computer with dual-core 1.86GHz Intel CPU and 1.97GB memory. In this experimental studies, the maximum number of function evaluations $MaxFunEvals = 50,000$, the number of the subproblems $N = 100$, the crossover probability $p_c = 0.8$, the mutation probability $p_m = 0.2$, the neighborhood list size $T = 10$, and the external population size $EP_{size} = 100$.

4.1. Comparisons with the Best Known Solutions

In order to illustrate the effectiveness of the proposed M-MOEA/D, we compared it with the best known solutions. Table 1 shows the best known solution for each instance. In Table 1, NV is short for the number of used vehicles , TD means the total traveled distance of routings, Ref. is the references that published the best known solutions. min NV and min TD respectively represent the solutions with smallest NV and lowest TD.

M-MOEA/D is a multi-objective optimization algorithm which minimizes the number of the used vehicles and the total traveled distance simultaneously. We compared the results of M-MOEA/D with the best known results on these two objectives. Hence we listed the best known results with the smallest number of used vehicles (min NV) and lowest total traveled distance (min TD) in Table 1. In Table 1, problems in categories C1 and C2 have the same solutions on min NV and min TD. With the exception of R101, RC103 and RC104, all other problems in categories R1, R2, RC1 and RC2 have two different solutions. Prior researches have shown that the problems with all customers located in geographical clusters have few non-dominated solutions and problems with randomly distributed have more non-dominated solutions.

The comparisons between M-MOEA/D and the best known solutions are presented in Tables 2 and 3. In these two tables, the solutions of M-MOEA/D were the best results found by the algorithm within 30 independent runs. NV, TD, min NV and min TD have the same meanings as shown in Table 1. %NV and %TD are the percentage difference for M-MOEA/D in the number of used vehicles and in the total traveled distance with the best known solutions. The averages of them in each set category are also given in Table 2 and Table 3. The records shown in bold are the instances managed to find the best known solutions.

As shown in Tables 2 and 3, for C1 and C2 problems, M-MOEA/D obtained the best known solutions on both smallest number of used vehicles (min NV) and lowest total traveled distance (min TD). It performs well on these two categories of problems. The averages of %NV and %TD on R1 and RC1 problems are no more than 1.5% on the best known solutions in min TD. In min NV, the averages of %TD and %NV on R1 problems are 5.66 and 0.63 respectively. Those on

Table 1: Best-know results for Solomon's benchmark set.

Problems	min NV			min TD			Problems	min NV			min TD		
	NV	TD	Ref.	NV	TD	Ref.		NV	TD	Ref.	NV	TD	Ref.
C101	10	828.94	[29]	10	828.94	[29]	C201	3	591.56	[29]	3	591.56	[29]
C102	10	828.94	[29]	10	828.94	[29]	C202	3	591.56	[29]	3	591.56	[29]
C103	10	828.06	[29]	10	828.06	[29]	C203	3	591.17	[29]	3	591.17	[29]
C104	10	824.78	[29]	10	824.78	[29]	C204	3	590.60	[29]	3	590.60	[29]
C105	10	828.94	[29]	10	828.94	[29]	C205	3	588.88	[29]	3	588.88	[29]
C106	10	828.94	[29]	10	828.94	[29]	C206	3	588.49	[29]	3	588.49	[29]
C107	10	828.94	[29]	10	828.94	[29]	C207	3	588.29	[29]	3	588.29	[29]
C108	10	828.94	[29]	10	828.94	[29]	C208	3	588.32	[29]	3	588.32	[29]
C109	10	828.94	[29]	10	828.94	[29]							
R101	18	1613.59	[10]	18	1613.59	[10]	R201	4	1252.37	[30]	9	1144.48	[7]
R102	17	1486.12	[29]	18	1454.68	[10]	R202	3	1191.70	[31]	8	1034.35	[32]
R103	13	1292.68	[33]	14	1213.62	[29]	R203	3	939.54	[34]	6	874.87	[32]
R104	9	1007.24	[34]	10	974.24	[10]	R204	2	825.52	[35]	4	736.52	[32]
R105	14	1377.11	[29]	15	1360.78	[32]	R205	3	994.42	[31]	5	954.16	[9]
R106	12	1251.98	[34]	13	1240.47	[32]	R206	3	906.14	[36]	5	879.89	[32]
R107	10	1104.66	[37]	11	1073.34	[32]	R207	2	837.20	[38]	4	799.86	[32]
R108	9	960.88	[39]	10	947.55	[32]	R208	2	726.75	[34]	4	705.45	[32]
R109	11	1194.73	[30]	13	1151.84	[32]	R209	3	909.16	[40]	5	859.39	[32]
R110	10	1118.59	[34]	12	1072.41	[32]	R210	3	938.58	[16]	5	910.70	[32]
R111	10	1096.72	[31]	12	1053.50	[32]	R211	2	892.71	[35]	4	755.96	[32]
R112	9	982.14	[41]	10	953.63	[29]							
RC101	14	1696.94	[42]	15	1623.58	[29]	RC201	4	1406.91	[34]	6	1134.91	[10]
RC102	12	1554.75	[42]	14	1461.23	[32]	RC202	3	1365.65	[43]	8	1095.64	[32]
RC103	11	1261.67	[44]	11	1261.67	[44]	RC203	3	1049.62	[45]	5	928.51	[32]
RC104	10	1135.48	[46]	10	1135.48	[46]	RC204	3	798.41	[34]	4	786.38	[32]
RC105	13	1629.44	[39]	16	1518.58	[32]	RC205	4	1297.19	[34]	7	1157.55	[32]
RC106	11	1424.73	[39]	13	1371.69	[10]	RC206	3	1146.32	[40]	7	1054.61	[32]
RC107	11	1222.10	[16]	12	1212.83	[32]	RC207	3	1061.14	[35]	6	966.08	[32]
RC108	10	1139.82	[42]	11	1117.53	[32]	RC208	3	828.14	[47]	4	779.31	[32]

RC1 problems are 3.14 and 0.73 respectively. Although the performance of M-MOEA/D on R1 and RC1 problems is not as good as on C1 and C2 problems, the differences between the solutions obtained by M-MOEA/D and the best known solutions are very small. For R2 problems and RC2 problems in min TD, the category averages have increased no more than 4% in %TD, but they reduced about 10% in %NV. The average of NV on R2 problems has increased 16.67% and the average of TD on R2 problems has reduced 2.36% in min NV. Therefore, most of the solutions obtained by M-MOEA/D on R2 and RC2 problems are compromising comparing with the best known solutions.

Through analyzing the experimental results in Table 2 and 3, M-MOEA/D performs well on C1 and C2 problems, but not so well on R1 and RC1 problems. The results obtained from M-MOEA/D on R2 and RC2 problems are mostly comprising solutions compared with the best known solutions. Although M-MOEA/D hasn't achieved all the best solutions, it is comparable to the algorithms obtained the best known solutions.

4.2. Comparisons with Other Multi-objective Algorithms

In this experiment, the proposed M-MOEA/D is compared with other two outstanding multi-objective approaches such as MOGP [16] and MOEA [28]. The experimental data of the two comparing algorithms are obtained from corresponding literatures.

Tables 4-9 show the comparison results between M-MOEA/D and the other two comparing algorithms on C1, R1, RC1, C2, R2 and RC2 instances respectively.

In these tables, NV and TD have the same meaning as before. Num.Ns is short for the numbers of non-dominated solutions found by the comparing

Table 2: Comparisons between the best known results and the best results obtained by M-MOEA/D on C1, R1 and RC1 problems.

Problems	min NV						min TD					
	Best-know		M-MOEA/D				Best-know		M-MOEA/D			
	NV	TD	NV	TD	%NV	%TD	NV	TD	NV	TD	%NV	%TD
C101	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C102	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C103	10	828.06	**10**	**828.06**	0.00	0.00	10	828.06	**10**	**828.06**	0.00	0.00
C104	10	824.78	**10**	**824.78**	0.00	0.00	10	824.78	**10**	**824.78**	0.00	0.00
C105	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C106	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C107	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C108	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
C109	10	828.94	**10**	**828.94**	0.00	0.00	10	828.94	**10**	**828.94**	0.00	0.00
Catgegory average					0.00	0.00					0.00	0.00
R101	18	1613.59	19	1652.17	5.56	2.39	18	1613.59	20	1644.70	11.11	1.93
R102	17	1486.12	**17**	**1486.12**	0.00	0.00	18	1454.68	18	1473.73	0.00	1.31
R103	13	1292.68	13	1354.22	0.00	4.76	14	1213.62	**14**	**1213.62**	0.00	0.00
R104	9	1007.24	10	999.31	11.11	-0.79	10	974.24	11	991.91	10.00	1.81
R105	14	1377.11	14	1410.64	0.00	2.43	15	1360.78	15	1366.58	0.00	0.43
R106	12	1251.98	12	1265.99	0.00	1.12	13	1240.47	13	1249.22	0.00	0.71
R107	10	1104.66	10	1139.47	0.00	3.15	11	1073.34	11	1086.22	0.00	1.20
R108	9	960.88	10	965.52	11.11	0.48	10	947.55	10	965.52	0.00	1.90
R109	11	1194.73	12	1157.44	9.09	-3.12	13	1151.84	13	1155.38	0.00	0.31
R110	10	1118.59	11	1110.68	10.00	-0.71	12	1072.41	12	1106.03	0.00	3.13
R111	10	1096.72	11	1073.82	10.00	-2.09	12	1053.50	11	1073.82	-8.33	1.93
R112	9	982.14	10	981.43	11.11	-0.07	10	953.63	10	981.43	0.00	2.92
Catgegory average					5.66	0.63					1.06	1.46
RC101	14	1696.94	14	1758.17	0.00	3.61	15	1623.58	16	1646.65	6.67	1.42
RC102	12	1554.75	13	1509.18	8.33	-2.93	14	1461.23	15	1484.48	7.14	1.59
RC103	11	1261.67	11	1274.85	0.00	1.04	11	1261.67	11	1274.85	0.00	1.04
RC104	10	1135.48	10	1145.79	0.00	0.91	10	1135.48	10	1145.79	0.00	0.91
RC105	13	1629.44	14	1548.43	7.69	-4.97	16	1518.58	15	1528.61	-6.25	0.66
RC106	11	1424.73	12	1447.84	9.09	1.62	13	1371.69	13	1399.17	0.00	2.00
RC107	11	1222.10	11	1254.67	0.00	2.67	12	1212.83	12	1235.54	0.00	1.87
RC108	10	1139.82	10	1183.85	0.00	3.86	11	1117.53	11	1138.95	0.00	1.92
Category average					3.14	0.73					0.94	1.43

Table 3: Comparisons between the best known results and the best results obtained by M-MOEA/D on C2, R2 and RC2 problems.

Problems	min NV						min TD					
	Best-know		M-MOEA/D				Best-know		M-MOEA/D			
	NV	TD	NV	TD	%NV	%TD	NV	TD	NV	TD	%NV	%TD
C201	3	591.56	**3**	**591.56**	0.00	0.00	3	591.56	**3**	**591.56**	0.00	0.00
C202	3	591.56	**3**	**591.56**	0.00	0.00	3	591.56	**3**	**591.56**	0.00	0.00
C203	3	591.17	**3**	**591.17**	0.00	0.00	3	591.17	**3**	**591.17**	0.00	0.00
C204	3	590.60	**3**	**590.60**	0.00	0.00	3	590.60	**3**	**590.60**	0.00	0.00
C205	3	588.88	**3**	**588.88**	0.00	0.00	3	588.88	**3**	**588.88**	0.00	0.00
C206	3	588.49	**3**	**588.49**	0.00	0.00	3	588.49	**3**	**588.49**	0.00	0.00
C207	3	588.29	**3**	**588.29**	0.00	0.00	3	588.29	**3**	**588.29**	0.00	0.00
C208	3	588.32	**3**	**588.32**	0.00	0.00	3	588.32	**3**	**588.32**	0.00	0.00
Catgegory average					0.00	0.00					0.00	0.00
R201	4	1252.37	4	1253.23	0.00	0.07	9	1144.48	6	1185.79	-33.33	3.61
R202	3	1191.70	4	1081.82	33.33	-9.22	8	1034.35	5	1049.72	-37.50	1.49
R203	3	939.54	3	955.70	0.00	1.72	6	874.87	5	889.36	-16.67	1.66
R204	2	825.52	3	753.32	50.00	-8.75	4	736.52	5	743.29	25.00	0.92
R205	3	994.42	3	1017.96	0.00	2.37	5	954.16	5	954.48	0.00	0.03
R206	3	906.14	3	915.49	0.00	1.03	5	879.89	4	887.90	-20.00	0.91
R207	2	837.20	3	813.47	50.00	-2.83	4	799.86	4	809.51	0.00	1.21
R208	2	726.75	2	728.63	0.00	0.26	4	705.45	3	711.59	-25.00	0.87
R209	3	909.16	3	918.82	0.00	1.06	5	859.39	4	867.47	-20.00	0.94
R210	3	938.58	3	952.91	0.00	1.53	5	910.70	5	920.06	0.00	1.03
R211	2	892.71	3	774.68	50.00	-13.22	4	755.96	4	767.10	0.00	1.47
Catgegory average					16.67	-2.36					-11.59	1.28
RC201	4	1406.91	4	1421.88	0.00	1.06	6	1134.91	7	1289.94	16.67	13.66
RC202	3	1365.65	4	1161.29	33.33	-14.96	8	1095.64	5	1118.66	-37.50	2.10
RC203	3	1049.62	3	1097.40	0.00	4.55	5	928.51	5	940.55	0.00	1.30
RC204	3	798.41	3	801.90	0.00	0.44	4	786.38	4	792.98	0.00	0.84
RC205	4	1297.19	4	1327.09	0.00	2.30	7	1157.55	6	1187.48	-14.29	2.59
RC206	3	1146.32	3	1200.92	0.00	4.76	7	1054.61	5	1089.14	-28.57	3.27
RC207	3	1061.14	3	1107.71	0.00	4.39	6	966.08	5	987.88	-16.67	2.26
RC208	3	828.14	3	841.37	0.00	1.60	4	779.31	4	807.83	0.00	3.66
Category average					4.17	0.52					-10.04	3.71

algorithms. The non-dominated solutions of M-MOEA/D were found by the algorithms within 30 independent runs. Taking consideration the total set of solutions found by the comparing algorithms, the non-dominated solutions are bolded in these tables.

From Table 4, we can see that M-MOEA/D performs as well as the two comparing algorithms on C1 problems. All of them succeeded in finding the best known solutions on C1 problems which have all customers located in geographical location and tight time windows.

Table 4: Comparisons of non-dominated solutions found by other multi-objective algorithms (C1).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
C101	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C102	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C103	**10**	**828.06**	**10**	**828.06**	**10**	**828.06**
C104	**10**	**824.78**	**10**	**824.78**	**10**	**824.78**
C105	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C106	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C107	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C108	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C109	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
Num.Ns	**9**		**9**		**9**	

Table 5: Comparisons of non-dominated solutions found by other multi-objective algorithms (R1).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
R101	19	1652.17	19	1677.00	**19**	**1650.80**
	20	1644.70	20	1651.10	**20**	**1642.88**
R102	**17**	**1486.12**	18	1511.80	**17**	**1486.12**
	18	**1473.73**	19	1494.70	18	1474.19
R103	13	1354.22	14	1287.00	**13**	**1308.28**
	14	**1213.62**	15	1264.20	14	1219.37
R104	10	999.31	10	974.24	**10**	**990.79**
	11	991.91			**11**	**984.56**
R105	14	1410.64	15	1424.60	**14**	**1377.11**
	15	1366.58	16	1382.50	**15**	**1364.91**
R106	12	1265.99			**12**	**1261.52**
	13	1249.22	13	1270.30	**13**	**1241.65**
R107	**10**	**1139.47**			10	1154.38
	11	1086.22	11	1108.80	**11**	**1083.30**
R108					**9**	**984.75**
	10	965.52	10	971.91	**10**	**960.03**
R109	**12**	**1157.44**	12	1212.30	12	1157.76
	13	1155.38	14	1206.70	**13**	**1154.61**
R110	11	1110.68			**11**	**1094.75**
	12	1106.03	12	1156.50	**12**	**1088.61**
R111	11	1073.82	11	1111.90	**11**	**1061.37**
R112	10	981.43	11	1011.50	**10**	**980.83**
Num.Ns	5		0		**18**	

Table 6: Comparisons of non-dominated solutions found by other multi-objective algorithms (RC1).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
RC101	**14**	**1758.17**				
	15	1646.81	15	1690.60	**15**	**1625.26**
	16	1646.65	16	1678.90		
RC102	13	1509.18			**13**	**1501.11**
	14	1484.89	14	1509.40	**14**	**1480.26**
	15	1484.48	15	1493.20		
RC103	**11**	**1274.85**	12	1331.80	11	1278.19
RC104	10	1145.79	11	1177.20	**10**	**1144.39**
RC105	14	1548.43	15	1611.50	**14**	**1540.18**
	15	1528.61	16	1589.40	**15**	**1519.44**
RC106	12	1447.84	13	1437.60	**12**	**1395.70**
	13	1399.17	14	1425.30	**13**	**1379.68**
RC107	11	1254.67	**11**	**1222.10**	11	1234.49
	12	1235.54			**12**	**1215.06**
RC108	10	1183.85			**10**	**1158.22**
	11	1138.95	11	1156.50	**11**	**1122.98**
Num.Ns	2		1		**11**	

Table 7: Comparisons of non-dominated solutions found by other multi-objective algorithms (C2).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
C201	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**
C202	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**
C203	**3**	**591.17**	**3**	**591.17**	**3**	**591.17**
C204	**3**	**590.60**	3	599.96	**3**	**590.60**
C205	**3**	**588.88**	**3**	**588.88**	**3**	**588.88**
C206	**3**	**588.49**	3	588.88	**3**	**588.49**
C207	**3**	**588.29**	3	591.56	**3**	**588.29**
C208	**3**	**588.32**	**3**	**588.32**	**3**	**588.32**
Num.Ns	**8**		5		**8**	

Table 8: Comparisons of non-dominated solutions found by other multi-objective algorithms (R2).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
R201	**4**	**1253.23**	4	1351.40	4	1254.77
	5	1196.50			**5**	**1194.07**
	6	**1185.79**				
R202	**4**	**1081.82**	4	1091.22	4	1087.29
	5	**1049.72**			5	1050.41
R203	3	955.70	3	1041.00	**3**	**950.90**
	4	**904.46**	5	995.80	4	912.24
	5	**889.36**	6	978.50	5	905.34
R204	3	753.32	3	1130.10	**3**	**752.83**
	4	**745.96**	4	927.70		
	5	**743.29**	5	831.80		
			6	826.20		
R205	**3**	**1017.96**	3	1422.30	3	1040.29
	4	**960.33**	4	1087.80	4	968.09
	5	**954.48**				
R206	**3**	**915.49**	3	940.12	3	930.58
	4	**887.90**			4	899.83
R207	**3**	**813.47**	3	904.90	3	818.97
	4	**809.51**				
R208	**2**	**728.63**			2	736.90
	3	**711.59**	3	774.18	3	712.98
R209	**3**	**918.82**			3	921.97
	4	**867.47**	4	1008.00	4	878.05
R210	**3**	**952.91**	3	938.58	3	961.36
	4	**928.35**			4	936.68
	5	**920.06**				
R211	**3**	**774.68**	3	1310.40	3	785.97
	4	**767.10**	4	1101.50		
Num.Ns	**24**		0		3	

It can be seen from Tables 5 and 6 that, for R1 and RC1 problems, the number of the non-dominated solutions found by M-MOEA/D is smaller than that of MOEA but larger than that of MOGP. R1 and RC1 problems have smaller time windows than R2 and RC2 problems, thus they have smaller feasible solution spaces. MOEA performs better than M-MOEA/D on problems with tight time window constrains.

For C2 problems, as shown in Table 7, M-MOEA/D and MOEA obtained all the best known solutions and the number of MOGP is only five. So M-MOEA/D and MOEA perform better than MOGP on C2 problems which have all customers located in geographical location and long time windows.

For R2 and RC2 problems, as shown in Tables 8 and 9, the number of the non-dominated solutions found by M-MOEA/D are much larger than both MOEA and MOGP. From the results, we can see that MOEA/D has obvious advantage on R2 and RC2 problems comparing with MOEA and MOGP. Therefore, M-MOEA/D is more suitable for solving such complicated problems large and complex feasible solution spaces.

Table 9: Comparisons of non-dominated solutions found by other multi-objective algorithms (RC2).

Problems	M-MOEA/D		MOGP		MOEA	
	NV	TD	NV	TD	NV	TD
RC201	**4**	**1421.88**	4	1423.70	4	1438.43
	5	**1316.61**			5	1329.26
	6	**1297.47**			6	1316.25
	7	**1289.94**			7	1299.58
RC202	**4**	**1161.29**	4	1369.80	4	1165.57
	5	**1118.66**			5	1120.15
RC203	3	1097.40			**3**	**1061.47**
	4	**944.50**	4	1060.00	4	954.51
	5	**940.55**	6	1020.10		
RC204	**3**	**801.90**	3	901.46	3	802.71
	4	792.98			**4**	**792.84**
RC205	4	1327.09	4	1410.30	**4**	**1318.71**
	5	**1245.94**			5	1259.00
	6	**1187.48**			6	1214.49
					7	1205.06
RC206	3	1200.92			**3**	**1191.62**
	4	1092.70	4	1194.80	**4**	**1085.82**
	5	1089.14			**5**	**1077.48**
RC207	**3**	**1107.71**	4	1040.60	3	1133.27
	4	**1000.98**			4	1001.73
	5	**987.88**			5	1001.51
RC208	**3**	**841.37**	3	898.50	3	844.96
	4	807.83			**4**	**780.07**
Num.Ns	**15**		0		7	

Table 10 employs the hypervolume indicator proposed by Zitzler and Thiele [48] to compare the non-dominated solutions obtained by M-MOEA/D and the other two comparing algorithm in view of multi-objective optimization. Given an approximation set A, its hypervolume is the volume of the portion of the objective space that is dominated by the approximation set A by a reference point. This metric is an unary indicator which is strictly monotonic with Pareto dominance. It can provide a qualitative measure not only of convergence but also of diversity. For minimum multi-objective optimization problem, a higher hypervolume value means the corresponding PF has a better convergence and diversity.

In this study, the ranges of the two objective values differ so much. Therefore, we normalize the objective values to a range of 0 to 1 before calculating the hypervolume values. The normalization procedures are performed on non-dominated solutions for each instance. In the normalization procedure, the objective values NV and TD of solutions are divided by MaxNV and MaxTD respectively and normalized to a range of 0 to 1. In which, MaxNV and MaxTD are the maximum values of vehicle number and total traveled distance of all the non-dominated solutions found by all the compared algorithms for corresponding instance. As the normalized objective values are not larger than 1, the reference point is set as (1.5, 1.5).

Table 10 provides stronger evidences which support the same conclusions we have made above by using a comprehensive indicator for multi-objective optimization. It can be seen that non-dominated solutions obtained by the proposed M-MOEA/D have larger hypervolume values on all the R2 problems and six of the eight RC2 problems. According to the average hypervolume values, it is reasonable to come to the conclusion that M-MOEA/D outperforms MOEA on R2 and RC2 problems.

Table 10: A comprehensive comparison of the obtained non-dominated solutions using hypervolume indicator.

Problems	M-MOEA/D	MOGP	MOEA	Problems	M-MOEA/D	MOGP	MOEA
C101	**0.250**	**0.250**	**0.250**	C201	**0.257**	**0.257**	**0.257**
C102	**0.250**	**0.250**	**0.250**	C202	**0.257**	**0.257**	**0.257**
C103	**0.251**	**0.251**	**0.251**	C203	**0.257**	**0.257**	**0.257**
C104	**0.253**	**0.253**	**0.253**	C204	**0.258**	0.250	**0.258**
C105	**0.250**	**0.250**	**0.250**	C205	**0.259**	**0.259**	**0.259**
C106	**0.250**	**0.250**	**0.250**	C206	**0.260**	0.259	**0.260**
C107	**0.250**	**0.250**	**0.250**	C207	**0.260**	0.257	**0.260**
C108	**0.250**	**0.250**	**0.250**	C208	**0.260**	**0.260**	**0.260**
C109	**0.250**	**0.250**	**0.250**				
Category average	**0.250**	**0.250**	**0.250**	Category average	**0.258**	0.257	**0.258**
R101	0.285	0.283	**0.286**	R201	**0.546**	0.458	0.543
R102	**0.403**	0.365	**0.403**	R202	**0.631**	0.611	0.630
R103	**0.656**	0.596	0.654	R203	**0.865**	0.795	0.857
R104	0.908	**0.919**	0.913	R204	**0.976**	0.871	0.971
R105	0.547	0.505	**0.549**	R205	**0.821**	0.696	0.811
R106	0.679	0.631	**0.683**	R206	**0.872**	0.839	0.864
R107	0.851	0.797	**0.852**	R207	**0.930**	0.864	0.924
R108	0.924	0.920	**0.973**	R208	**1.164**	0.956	1.162
R109	**0.730**	0.702	**0.730**	R209	**0.884**	0.659	0.878
R110	0.798	0.729	**0.808**	R210	**0.848**	0.840	0.839
R111	0.817	0.795	**0.824**	R211	**0.960**	0.701	0.947
R112	**0.915**	0.852	**0.915**				
Category average	0.709	0.675	**0.716**	Category average	**0.863**	0.754	0.857
RC101	**0.348**	0.306	0.324	RC201	**0.544**	0.474	0.536
RC102	0.450	0.406	**0.452**	RC202	**0.666**	0.509	0.665
RC103	**0.630**	0.557	0.628	RC203	**0.891**	0.726	0.886
RC104	0.742	0.675	**0.743**	RC204	**1.016**	0.936	**1.016**
RC105	0.393	0.334	**0.397**	RC205	**0.607**	0.482	0.597
RC106	0.526	0.473	**0.536**	RC206	0.784	0.622	**0.792**
RC107	0.647	0.654	**0.657**	RC207	**0.858**	0.721	0.848
RC108	0.744	0.684	**0.752**	RC208	1.002	0.938	**1.020**
Category average	0.560	0.511	**0.561**	Category average	**0.796**	0.676	0.795

From above experimental results, M-MOEA/D found the best known solutions on C1 and C2 problems. It performs well on these problems which have all customers located in geographical clusters. For R1 and RC1 problems whose customers have small time windows and are randomly distributed, M-MOEA/D performs not as well as MOEA but much better than MOGP. M-MOEA/D has obvious advantage on R2 and RC2 problems, it is more suitable for solving the problems with long time windows and remote distribution of customers.

4.3. Contributions of the Two Innovative Components in M-MOEA/D

This experiment is designed to validate the effectiveness of the newly designed two components in M-MOEA/D. One component is the incorporation of three local search methods, noted as the component 1. The other one is the specially designed selection operator for MO-VRPTW, noted as component 2. We investigated the following variants of M-MOEA/D in this experiment.

- Variant-1: M-MOEA/D without component 1.
- Variant-2: M-MOEA/D without component 2.
- Variant-3: M-MOEA/D without component 1 and component2.

In algorithms Variant-2 and Variant-3, the selection operation in the original MOEA/D is adopted. Variant-3 is a variant of M-MOEA/D without local search methods and the newly designed selection operator is the prototype of MOEA/D with modified crossover and mutation operators. The hypervolume indicator is employed to compare the results of M-MOEA/D and its variants in view of multi-objective optimization. Before calculating the hypervolume values, normalization procedures are performed on non-dominated solutions obtained by M-MOEA/D and its variants for each instance. The normalization procedure is the same as that in section 4.2, in which MaxNV and MaxTD are set as the maximum values of vehicle number and total traveled distance of all the non-dominated solutions found by M-MOEA/D and its variants over 30 independent runs for corresponding instance. The reference point is also set as (1.5, 1.5).

Tables 11 - 16 compare the performance of M-MOEA/D and its three variants based on the average and the standard deviation of the hypervolume indicator over 30 independent runs. In these tables, the numbers in bracket are the standard deviation values, while the numbers above them are the average values. The numbers with largest average values or lowest standard deviation values are bolded.

Table 11: Comparisons of M-MOEA/D and its variants for C1 instances with hypervolume.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
C101	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	4.827E-01 (4.232E-02)
C102	**4.975E-01** **(0.000E+00)**	4.964E-01 (4.797E-03)	4.882E-01 (1.363E-02)	4.725E-01 (2.008E-02)
C103	**4.974E-01** **(2.347E-03)**	4.930E-01 (8.360E-03)	4.675E-01 (2.491E-02)	4.541E-01 (2.504E-02)
C104	**4.979E-01** **(3.894E-03)**	4.853E-01 (1.096E-02)	4.211E-01 (3.847E-02)	4.241E-01 (3.645E-02)
C105	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	4.922E-01 (9.567E-03)
C106	**4.975E-01** **(0.000E+00)**	4.953E-01 (6.670E-03)	4.945E-01 (9.511E-03)	4.810E-01 (1.773E-02)
C107	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	4.963E-01 (5.154E-03)	4.688E-01 (4.109E-02)
C108	**4.975E-01** **(0.000E+00)**	**4.975E-01** **(0.000E+00)**	4.889E-01 (1.073E-02)	4.669E-01 (3.568E-02)
C109	**4.963E-01** **(3.762E-03)**	4.863E-01 (1.175E-02)	4.479E-01 (3.696E-02)	4.286E-01 (4.763E-02)
Average	**4.974E-01** **(1.111E-03)**	4.940E-01 (4.726E-03)	4.777E-01 (1.549E-02)	4.634E-01 (3.062E-02)

Table 12: Comparisons of M-MOEA/D and its variants for C2 instances with hypervolume.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
C201	**5.609E-01** **(0.000E+00)**	**5.609E-01** **(0.000E+00)**	**5.609E-01** **(0.000E+00)**	**5.609E-01** **(0.000E+00)**
C202	**5.609E-01** **(0.000E+00)**	**5.609E-01** **(0.000E+00)**	**5.609E-01** **(0.000E+00)**	5.363E-01 (2.700E-02)
C203	**5.613E-01** **(0.000E+00)**	**5.613E-01** **(0.000E+00)**	5.510E-01 (7.906E-03)	5.188E-01 (3.009E-02)
C204	**5.600E-01** **(3.119E-03)**	5.554E-01 (8.919E-03)	5.387E-01 (2.407E-02)	5.142E-01 (2.570E-02)
C205	**5.634E-01** **(0.000E+00)**	**5.634E-01** **(0.000E+00)**	**5.634E-01** **(0.000E+00)**	5.596E-01 (1.189E-02)
C206	**5.638E-01** **(0.000E+00)**	**5.638E-01** **(0.000E+00)**	**5.638E-01** **(0.000E+00)**	5.532E-01 (2.558E-02)
C207	**5.640E-01** **(0.000E+00)**	**5.640E-01** **(0.000E+00)**	5.639E-01 (4.918E-04)	5.600E-01 (1.224E-02)
C208	**5.640E-01** **(0.000E+00)**	**5.640E-01** **(0.000E+00)**	**5.640E-01** **(0.000E+00)**	5.443E-01 (4.675E-02)
Average	**5.623E-01** **(3.899E-04)**	5.617E-01 (1.115E-03)	5.583E-01 (4.059E-03)	5.434E-01 (2.241E-02)

Table 13: Comparisons of M-MOEA/D and its variants for R1 instances with hypervolume.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
R101	**3.286E-01** **(2.974E-03)**	3.251E-01 (7.325E-03)	3.161E-01 (1.204E-02)	3.063E-01 (1.671E-02)
R102	4.413E-01 (1.307E-02)	**4.417E-01** **(1.231E-02)**	4.326E-01 (1.253E-02)	4.073E-01 (2.713E-02)
R103	**6.805E-01** (1.840E-02)	6.782E-01 (1.849E-02)	6.442E-01 (1.506E-02)	6.339E-01 **(1.397E-02)**
R104	**9.479E-01** **(6.164E-03)**	9.404E-01 (7.965E-03)	9.079E-01 (2.306E-02)	8.855E-01 (3.586E-02)
R105	**5.765E-01** (1.681E-02)	5.735E-01 (1.923E-02)	5.319E-01 **(1.379E-02)**	5.199E-01 (2.941E-02)
R106	**7.057E-01** (2.345E-02)	6.944E-01 (1.686E-02)	6.630E-01 **(1.091E-02)**	6.414E-01 (3.068E-02)
R107	**8.566E-01** **(9.742E-03)**	8.511E-01 (1.394E-02)	8.234E-01 (2.397E-02)	8.037E-01 (3.771E-02)
R108	**9.657E-01** (8.800E-03)	9.629E-01 **(7.791E-03)**	9.386E-01 (1.456E-02)	9.341E-01 (2.798E-02)
R109	**7.634E-01** (2.159E-02)	7.526E-01 **(2.084E-02)**	7.124E-01 (2.864E-02)	6.863E-01 (3.183E-02)
R110	**8.175E-01** (2.502E-02)	8.106E-01 (2.253E-02)	7.678E-01 **(1.526E-02)**	7.471E-01 (3.098E-02)
R111	**8.564E-01** **(1.080E-02)**	8.427E-01 (2.716E-02)	8.129E-01 (2.912E-02)	7.821E-01 (3.334E-02)
R112	**9.506E-01** **(1.492E-02)**	9.353E-01 (1.971E-02)	8.905E-01 (3.004E-02)	8.649E-01 (2.177E-02)
Average	**7.409E-01** **(1.431E-02)**	7.340E-01 (1.618E-02)	7.034E-01 (1.908E-02)	6.844E-01 (2.811E-02)

Table 14: Comparisons in hypervolume of M-MOEA/D and its variants for R2 instances.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
R201	**7.888E-01** (8.154E-03)	7.849E-01 **(7.035E-03)**	7.628E-01 (9.429E-03)	7.456E-01 (3.220E-02)
R202	8.602E-01 **(6.254E-03)**	**8.604E-01** (8.311E-03)	8.413E-01 (1.032E-02)	8.379E-01 (1.048E-02)
R203	**1.083E+00** **(6.789E-03)**	1.078E+00 (7.598E-03)	1.064E+00 (9.306E-03)	1.027E+00 (4.633E-02)
R204	**1.177E+00** **(3.983E-03)**	**1.177E+00** (4.901E-03)	1.158E+00 (1.327E-02)	1.156E+00 (1.029E-02)
R205	**1.034E+00** (1.284E-02)	1.028E+00 **(9.815E-03)**	1.010E+00 (1.294E-02)	9.425E-01 (6.212E-02)
R206	**1.086E+00** **(8.284E-03)**	1.079E+00 (1.254E-02)	1.063E+00 (1.125E-02)	1.037E+00 (2.118E-02)
R207	**1.135E+00** **(6.823E-03)**	1.133E+00 (8.858E-03)	1.107E+00 (1.352E-02)	1.105E+00 (1.650E-02)
R208	**1.343E+00** (3.546E-02)	1.339E+00 (3.281E-02)	1.334E+00 **(9.390E-03)**	1.218E+00 (7.210E-02)
R209	**1.092E+00** **(1.160E-02)**	1.089E+00 (1.175E-02)	1.063E+00 (1.187E-02)	1.026E+00 (5.202E-02)
R210	**1.066E+00** **(5.855E-03)**	1.063E+00 (7.798E-03)	1.043E+00 (1.026E-02)	1.026E+00 (3.084E-02)
R211	**1.152E+00** (1.371E-02)	1.150E+00 (1.423E-02)	1.126E+00 **(8.241E-03)**	1.108E+00 (2.141E-02)
Average	**1.074E+00** **(1.089E-02)**	1.071E+00 (1.142E-02)	1.052E+00 **(1.089E-02)**	1.021E+00 (3.413E-02)

Table 15: Comparisons in hypervolume of M-MOEA/D and its variants for RC1 instances.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
RC101	**3.820E-01** (1.627E-02)	3.678E-01 (1.879E-02)	3.349E-01 **(1.203E-02)**	3.273E-01 (1.617E-02)
RC102	**5.049E-01** (1.939E-02)	5.025E-01 (1.840E-02)	4.610E-01 **(1.389E-02)**	4.477E-01 (2.794E-02)
RC103	**6.557E-01** (3.182E-02)	6.365E-01 **(1.917E-02)**	6.257E-01 (2.417E-02)	6.026E-01 (2.974E-02)
RC104	**7.960E-01** (2.407E-02)	7.778E-01 (3.050E-02)	7.352E-01 (2.496E-02)	7.133E-01 **(2.160E-02)**
RC105	**4.547E-01** **(1.112E-02)**	4.477E-01 (1.582E-02)	4.130E-01 (2.561E-02)	3.939E-01 (1.721E-02)
RC106	**5.632E-01** **(1.445E-02)**	5.598E-01 (2.050E-02)	5.389E-01 (2.138E-02)	5.124E-01 (2.697E-02)
RC107	**6.906E-01** (2.934E-02)	6.557E-01 **(2.099E-02)**	6.345E-01 (2.663E-02)	6.059E-01 (3.738E-02)
RC108	**7.610E-01** (1.548E-02)	7.589E-01 **(1.156E-02)**	7.267E-01 (2.920E-02)	6.860E-01 (4.071E-02)
Average	**6.010E-01** (2.024E-02)	5.883E-01 **(1.947E-02)**	5.587E-01 (2.223E-02)	5.362E-01 (2.722E-02)

Table 16: Comparisons in hypervolume of M-MOEA/D and its variants for RC2 instances.

Problems	M-MOEA/D	Variant-1	Variant-2	Variant-3
RC201	**9.843E-01** **(6.727E-03)**	9.748E-01 (1.003E-02)	9.545E-01 (1.396E-02)	8.421E-01 (3.521E-02)
RC202	1.058E+00 (7.945E-03)	**1.059E+00** **(6.017E-03)**	1.038E+00 (1.242E-02)	1.031E+00 (1.487E-02)
RC203	**1.247E+00** (4.798E-02)	1.243E+00 (4.949E-02)	1.237E+00 **(1.208E-02)**	1.115E+00 (3.414E-02)
RC204	**1.341E+00** **(3.895E-03)**	**1.341E+00** (4.140E-03)	1.327E+00 (8.654E-03)	1.310E+00 (1.642E-02)
RC205	**1.021E+00** (1.034E-02)	1.011E+00 **(9.095E-03)**	9.897E-01 (1.227E-02)	9.304E-01 (5.760E-02)
RC206	**1.142E+00** (6.402E-02)	1.108E+00 (6.178E-02)	1.096E+00 (6.156E-02)	1.046E+00 **(1.525E-02)**
RC207	**1.142E+00** (5.630E-02)	1.139E+00 (5.650E-02)	1.126E+00 (5.826E-02)	1.088E+00 **(1.466E-02)**
RC208	1.317E+00 (1.489E-02)	**1.322E+00** (1.715E-02)	1.302E+00 **(1.453E-02)**	1.278E+00 (2.714E-02)
Average	**1.156E+00** (2.651E-02)	1.150E+00 (2.677E-02)	1.134E+00 **(2.422E-02)**	1.080E+00 (2.691E-02)

It can be seen from these tables that the average hypervolume values of M-MOEA/D are the largest on 49 of the 56 benchmark problems. M-MOEA/D has the largest average values on each category of problems among the compared algorithms. The diversity and convergence of M-MOEA/D are the best when comparing with its variants. With the exception of RC1 problems, the average of standard deviation value on each category in M-MOEA/D is the lowest comparing with its three variants. Therefore, the stability of M-MOEA/D is the best comparing with its variants.

For the three variants, the average values on each category of variant-1 is the largest, followed by variant-2, and the last variant-3. This shows that the diversity and convergence of variant-1 are the best, followed by variant-2 and the last variant-3. Variant-1 has the lowest averages of the standard deviation on each category, followed by variant-2, and the last variant-3. In other words, the stability of variant-1 is the best, followed by variant-2 and the last variant-3.

Through the comparisons among M-MOEA/D and its variants, we can find out that the two new components, the local search methods and the selection operator, have improved the convergence, diversity and stability of M-MOEA/D. The two components work cooperatively, and they play an important role in enhancing the performance of M-MOEA/D. The results of the hypervolume of variant-1 are better than that of variant-2 which indicates that the component 2 contributes more than the component 1 in view of multi-objective optimization.

4.4. Comparison of the Proposed Selection Operator with the Non-dominated Sorting Method

The last experiment was designed to make further investigation on the effectiveness of the newly designed selection operator in M-MOEA/D. We replaced our specially designed operator in M-MOEA/D with the effective non-dominated sorting selection operator which is widely used in many evolutionary multi-objective optimization algorithms and it is noted as Variant-4.

Tables 17-22 are respectively the comparisons between M-MOEA/D and the Variant-4 on C1, C2, R1, R2, RC1 and RC2 problems. The solutions in these tables are the optimal solutions within 30 independent runs according to the lowest number of used vehicles (min NV) and the shortest total traveled distance (min TD) respectively.

Table 17: Comparisons of M-MOEA/D and its variant-4 for C1 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
C101	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C102	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C103	**10**	**828.06**	**10**	**828.06**	**10**	**828.06**	**10**	**828.06**
C104	**10**	**824.78**	**10**	**824.78**	**10**	**824.78**	**10**	**824.78**
C105	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C106	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C107	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C108	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
C109	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**	**10**	**828.94**
Average	**10**	**828.38**	**10**	**828.38**	**10**	**828.38**	**10**	**828.38**

Table 18: Comparisons of M-MOEA/D and its variant-4 for C2 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
C201	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**
C202	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**	**3**	**591.56**
C203	**3**	**591.17**	**3**	**591.17**	**3**	**591.17**	**3**	**591.17**
C204	**3**	**590.60**	**3**	**590.60**	**3**	**590.60**	**3**	**590.60**
C205	**3**	**588.88**	**3**	**588.88**	**3**	**588.88**	**3**	**588.88**
C206	**3**	**588.49**	**3**	**588.49**	**3**	**588.49**	**3**	**588.49**
C207	**3**	**588.29**	**3**	**588.29**	**3**	**588.29**	**3**	**588.29**
C208	**3**	**588.32**	**3**	**588.32**	**3**	**588.32**	**3**	**588.32**
Average	**3**	**589.86**	**3**	**589.86**	**3**	**589.86**	**3**	**589.86**

Table 19: Comparisons of M-MOEA/D and its variant-4 for R1 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
R101	**19**	**1652.17**	**19**	**1652.17**	20	1644.70	**20**	**1644.25**
R102	**17**	**1486.12**	**17**	**1486.12**	18	1473.73	**18**	**1472.81**
R103	13	1354.22	**13**	**1311.26**	**14**	**1213.62**	14	1219.98
R104	**10**	**999.31**	10	1000.47	**11**	**991.91**	11	999.64
R105	14	1410.64	**14**	**1386.77**	**15**	**1366.58**	15	1366.69
R106	**12**	**1265.99**	12	1291.71	**13**	**1249.22**	13	1251.47
R107	10	1139.47	**10**	**1133.52**	11	1086.22	**12**	**1079.99**
R108	10	965.52	**10**	**962.90**	10	965.52	**10**	**962.90**
R109	**12**	**1157.44**	12	1159.80	**13**	**1155.38**	12	1159.80
R110	11	1110.68	**11**	**1090.21**	12	1106.03	**12**	**1077.14**
R111	11	1073.82	**11**	**1062.06**	11	1073.82	**11**	**1062.06**
R112	10	981.43	**10**	**970.36**	10	981.43	**10**	**970.36**
Average	**12.42**	1216.40	**12.42**	**1208.95**	**13.17**	1192.35	**13.17**	**1188.92**

Table 20: Comparisons of M-MOEA/D and its variant-4 for R2 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
R201	**4**	**1253.23**	4	1262.16	6	1185.79	**7**	**1180.21**
R202	**4**	**1081.82**	4	1082.20	5	1049.72	**5**	**1043.59**
R203	3	955.70	**3**	**949.96**	5	889.36	**5**	**887.02**
R204	3	753.32	**3**	**751.77**	5	743.29	**4**	**741.86**
R205	**3**	**1017.96**	3	1030.13	**5**	**954.48**	5	975.01
R206	**3**	**915.49**	3	919.34	4	887.90	**5**	**884.89**
R207	**3**	**813.47**	3	818.16	4	809.51	**4**	**807.71**
R208	**2**	**728.63**	2	731.19	3	711.59	**3**	**708.05**
R209	3	918.82	**3**	**917.47**	**4**	**867.47**	4	878.38
R210	**3**	**952.91**	3	955.07	**5**	**920.06**	4	923.06
R211	**3**	**774.68**	3	775.44	**4**	**767.10**	4	774.31
Average	**3.09**	**924.18**	**3.09**	926.63	**4.55**	**889.66**	**4.55**	891.28

Table 21: Comparisons of M-MOEA/D and its variant-4 for RC1 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
RC101	**14**	**1758.17**	15	1635.46	16	1646.65	**15**	**1635.46**
RC102	**13**	**1509.18**	13	1515.19	**15**	**1484.48**	14	1486.97
RC103	**11**	**1274.85**	12	1307.34	**11**	**1274.85**	12	1307.34
RC104	10	1145.79	**10**	**1142.83**	10	1145.79	**10**	**1142.83**
RC105	14	1548.43	**14**	**1542.29**	15	1528.61	**16**	**1518.60**
RC106	12	1447.84	**12**	**1384.81**	13	1399.17	**13**	**1381.39**
RC107	**11**	**1254.67**	11	1260.47	**12**	**1235.54**	12	1241.04
RC108	**10**	**1183.85**	11	1135.77	11	1138.95	**11**	**1135.77**
Average	**11.88**	1390.35	12.25	**1365.52**	**12.88**	1356.76	**12.88**	**1356.18**

Table 22: Comparisons of M-MOEA/D and its variant-4 for RC2 instances.

Problems	min NV				min TD			
	M-MOEA/D		Variant-4		M-MOEA/D		Variant-4	
	NV	TD	NV	TD	NV	TD	NV	TD
RC201	4	1421.88	**4**	**1413.52**	7	1289.94	**8**	**1285.98**
RC202	**4**	**1161.29**	4	1166.07	5	1118.66	**6**	**1115.52**
RC203	3	1097.40	**3**	**1089.96**	5	940.55	**5**	**939.65**
RC204	3	801.90	**3**	**800.35**	4	792.98	**4**	**792.42**
RC205	4	1327.09	**4**	**1322.78**	**6**	**1187.48**	6	1191.97
RC206	3	1200.92	**3**	**1199.10**	**5**	**1089.14**	4	1091.42
RC207	**3**	**1107.71**	3	1131.10	5	987.88	**5**	**986.43**
RC208	**3**	**841.37**	3	859.39	**4**	**807.83**	4	822.12
Average	**3.38**	**1119.95**	**3.38**	1122.78	**5.13**	**1026.81**	5.25	1028.19

As shown in Tables 17-22, M-MOEA/D performs as good as variant-4 on C1 and C2 problems. This is because the optimal solutions of C1 and C2 problems are unitary. For R2 and RC2 problems, the solutions found by M-MOEA/D are better than those of Variant-4 on both the smallest number of used vehicles (min NV) and the lowest total traveled distance (min TD). For R1 problems, the average TD of variant-4 is lower than that of M-MOEA/D. However, the average NV is the same in both min NV and min TD. For RC1 problems, the average NV of M-MOEA/D is lower than that of variant-4 in min NV. Although M-MOEA/D performs not as good as variant-4 in min TD, it has the same average NV with variant-4 and the average TD of M-MOEA/D is only 0.58 larger than that of variant-4. This illustrates that the proposed selection operator considers the number of used vehicles first, and then the total traveled distance, which had been analyzed in section 4.7.

5. Conclusion

In this paper, we investigated the multi-objective vehicle routing problem with time windows (MO-VRPTW) which simultaneously minimize the number of used vehicles and the total traveled distance. By incorporating local search methods into MOEA/D, we proposed a memetic MOEA/D (M-MOEA/D) to solve MO-VPRTW. The key improvements of M-MOEA/D are obtained by its two newly developed components. One is the incorporation of three local search methods. Another one is a specially-designed selection operator. With the help of the two improving components, M-MOEA/D obtains highly competitive performance on Solomon's VRPTW 100-customer instances.

We also investigated the contributions of the two new components in M-MOEA/D. Experimental results demonstrate that both components play an important role in improving the performance. Moreover, the selection operator is the biggest contributor. We have also illustrated the effectiveness of the newly designed selection operator which considers the number of used vehicles first and then the total traveled distance.

In the field of VRPTW, many research efforts have been carried out to solve Solomon's benchmark problems in which the numbers of customers are less than 100. As far as we know, many real-world transportation problems have more than 100 customers to be served. Therefore, it will be our future work to apply the proposed M-MOEA/D to some MO-VRPTW with large customer sizes.

Acknowledgements

This work was supported by the National Natural Science Foundation of China under Grant Nos. 61303119 and 61373043, the Fundamental Research Funds for the Central Universities under Grant Nos. JB140304 and JB140317 .

References

[1] L. H. Lee, K. C. Tan, K. Ou, Y. H. Chew, Vehicle capacity planning system: A case study on vehicle routing problem with time windows, IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans 33 (2) (2003) 169–178.

[2] R. Baldacci, A. Mingozzi, R. Roberti, Recent exact algorithms for solving the vehicle routing problem under capacity and time window constraints, European Journal of Operational Research 218 (1) (2012) 1–6.

[3] N. A. El-Sherbeny, Vehicle routing with time windows: An overview of exact, heuristic and metaheuristic methods, Journal of King Saud University (Science) (22) (2010) 123–131.

[4] J. Castro-Gutierrez, D. Landa-Silva, J. Moreno Perez, Nature of real-world multi-objective vehicle routing with evolutionary algorithms, in: IEEE International Conference on Systems, Man, and Cybernetics, 2011, pp. 257–264.

[5] S. C. Hong, Y. B. Park, A heuristic for a bi-objective vehicle routing with time window constraints, International Journal of Production Economics 62 (3) (1999) 249–258.

[6] H. Gehring, J. Homberger, Parallelization of a two-phased metaheuristic for routing problems with time windows, Asia-Pacific Journal of Operational Research 18 (2001) 35–47.

[7] G. B. Alvarenga, G. R. Mateus, T. G. D., A genetic and set partitioning two-phase approach for the vehicle routing problem with time windows, Computers & Operations Research 34 (6) (2007) 1561–1584.

[8] M. Rahoual, B. Kitoun, H. M. Mabed, V. Bachelet, F. Benameur, Multicriteria genetic algorithms for the vehicle routing problem with time windows, in: Proceedings to the fourth Metaheuristic International Conference, 2001, pp. 527–532.

[9] B. Ombuki, B. J. Ross, F. Hanshar, Multi-objective genetic algorithms for vehicle routing problem with timewindows, Applied Intelligence 24 (1) (2006) 17–30.

[10] K. C. Tan, Y. H. Chew, L. H. Lee, A hybrid multiobjective evolutionary algorithm for solving vehicle routing problem with time windows, Computational Optimization and Applications 34 (1) (2006) 115–151.

[11] M. J. Geiger, W. Wenger, W. Habenicht, Interactive utility maximization in multi-objective vehicle routing problems: A "decision maker in the loop"-approach, in: Proceedings of the IEEE Symposium on Computational Intelligence in Multicriteria Decision Making, 2007, pp. 178–184.

[12] H. Xu, W. Fan, T. Wei, L. Yu, An Or-opt NSGA-II algorithm for multi-objective vehicle routing problem with time windows, in: Proceedings of the fourth IEEE Conference on Automation Science and Engineering Key Bridge Marriott, 2008, pp. 23–26.

[13] A. Garcia-Najera, J. A. Bullinaria, A multi-objective density restricted genetic algorithm for the vehicle routing problem with time windows, in: Proceedings of UK workshop on computational intelligence, 2008, pp. 47–52.

[14] A. Garcia-Najera, J. A. Bullinaria, An improved multi-objective evolutionary algorithm for the vehicle routing problem with time windows, Computers & Operations Research 38 (1) (2011) 287–300.

[15] Y. J. Gong, J. Zhang, O. Liu, Optimizing the vehicle routing problem with time windows: A discrete particle swarm optimization approach, IEEE Transactions on Systems, Man, and Cybernetics - Part C 42 (2) (2012) 254–267.

[16] K. Ghoseiri, S. F. Ghannadpour, Multi-objective vehicle routing problem with time windows using goal programming and genetic algorithm, Applied Soft Computing 10 (4) (2010) 1096–1107.

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

[18] B. Melian-Batista, A. D. Santiago, F. AngelBello, A. Alvarez, A bi-objective vehicle routing problem with time windows: A real case in tenerife, Applied Soft Computing 17 (2014) 140 – 152.

[19] R. Banos, J. Ortega, C. Gil, A. L. Marquez, F. de Toro, A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows, Computers & Industrial Engineering 65 (2) (2013) 286 – 296.

[20] A. Zhou, B.-Y. Qu, H. Li, S.-Z. Zhao, P. N. Suganthan, Q. Zhang, Multi-objective evolutionary algorithms: A survey of the state of the art, Swarm and Evolutionary Computation 1 (1) (2011) 32–49.

[21] Q. Zhang, H. Li, MOEA/D: A multiobjective evolutionary algorithm based on decomposition, IEEE Transactions on Evolutionary Computation 11 (6) (2007) 712–731.

[22] F. Neri, C. Cotta, Memetic algorithms and memetic computing optimization: A literature review, Swarm and Evolutionary Computation 2 (2012) 1–14.

[23] X. Chen, Y.-S. Ong, M.-H. Lim, K. C. Tan, A multi-facet survey on memetic computation, IEEE Transactions On Evolutionary Computation 15 (5) (2011) 591–607.

[24] J. Potvin, S. Bengio, The vehicle routing problem with time windows part II: Genetic search, Informs Journal on Computing 8 (2) (1996) 165–172.

[25] R. Tavakkoli-Moghaddam, A. Saremi, M. Ziaee, A memetic algorithm for a vehicle routing problem with backhauls, Applied Mathematics and Computation 181 (2) (2006) 1049–1060.

[26] K. Deb, H. Jain, An improved NSGA-II procedure for many-objective optimization, part I: Solving problems with box constraints, Tech. Rep. 2012009, Indian Institute of Technology (June 2012).

[27] M. M. Solomon, Algorithms for vehicle routing and scheduling problem with time window constraints, Operations Research 35 (2) (1987) 254–265.

[28] A. Garcia-Najera, Multi-objective evolutionary algorithms for vehicle routing problems, Ph.D. thesis, University of Birmingham (December 2010).

[29] Y. Rochat, E. Taillard, Probabilistic diversification and intensification in local search for vehicle routing, Journal of Heuristics 1 (1) (1995) 147–167.

[30] J. Homberger, H. Gehring, Two evolutionary metaheuristics for the vehicle routing problem with time windows, Information Systems and Operational Research 37 (3) (1999) 297–318.

[31] L. M. Rousseau, M. Gendreau, G. Pesant, Using constraint-based operators to solve the vehicle routing problem with time windows, Journal of Heuristics 8 (1) (2002) 43–58.

[32] S. Jung, B. R. Moon, A hybrid genetic algorithm for the vehicle routing problem with time windows, in: Genetic and Evolutionary Computation, Morgan Kaufmann, 2002, pp. 1309–1316.

[33] H. Li, A. Lim, Local search with annealing-like restarts to solve the VRPTW, European Journal of Operational Research 150 (1) (2003) 115–127.

[34] D. Mester, An evolutionary strategies algorithm for large scale vehicle routing problem with capacitate and time windows restrictions, Working paper, Institute of Evolution, University of Haifa, Israel (2002).

[35] R. Bent, P. V. Hentenryck, A two-stage hybrid local search for the vehicle routing problem with time windows, Transportation Science 38 (4) (2004) 515–530.

[36] G. Schrimpf, J. Schneider, H. Stamm-Wilbrandt, G. Dueck, Record breaking optimization results using the ruin and recreate principle, Journal of Computational Physics 159 (2) (2000) 139–171.

[37] P. Shaw, A new local search algorithm providing high quality solutions to vehicle routing problems, Working paper, University of Strathclyde, Glasgow, United Kingdom (1997).

[38] A. L. Bouthillier, T. G. Crainic, A cooperative parallel meta-heuristic for the vehicle routing problem with time windows, Computers & Operations Research 32 (7) (2005) 1685–1708.

[39] J. Berger, M. Barkaoui, O. Braysi, A route-directed hybrid genetic approach for the vehicle routing problem with time windows, Information Systems and Operational Research 41 (2) (2003) 179–194.

[40] J. Homberger, Verteilt-parallele metaheuristiken zur tourenplanung, Ph.D. thesis, Fern University, Hagen, Germany (2000).

[41] L. M. Gambardella, E. Taillard, G. Agazzi, New Ideas in Optimization, McGraw-Hill, 1999.

[42] E. Taillard, P. Badeau, M. Gendreau, F. Guertin, J. Potvin, A tabu search heuristic for the vehicle routing problem with soft time windows, Transportation Science 31 (2) (1997) 170–186.

[43] P. P. Repoussis, C. D. Tarantilis, G. Ioannou, Arc-guided evolutionary algorithm for the vehicle routing problem with time windows, IEEE Transactions on Evolutionary Computation 13 (3) (2009) 624–647.

[44] P. Shaw, Using constraint programming and local search methods to solve vehicle routing problems, in: Principles and Practice of Constraint Programming, Vol. 1520 of Lecture Notes in Computer Science, Springer Berlin Heidelberg, 1998, pp. 417–431.

[45] Z. Czech, P. Czarnas, Parallel simulated annealing for the vehicle routing problem with time windows, in: Parallel, Distributed and Network-based Processing, 2002. Proceedings. 10th Euromicro Workshop on, 2002, pp. 376 –383.

[46] J. F. Cordeau, G. Laporte, A. Mercier, A unified tabu search heuristic for vehicle routing problems with time windows, Journal of the Operational Research Society 52 (8) (2001) 928–936.

[47] T. Ibaraki, S. Imahori, M. Kubo, T. Masuda, T. Uno, M. Yagiura, Effective local search algorithms for routing and scheduling problems with general time-window constraints, Transportation Science 39 (2) (2005) 206–232.

[48] E. Zitzler, L. Thiele, Multiobjective optimization using evolutionary algorithms - A comparative case study, Springer Berlin Heidelberg, 1998.