Hindawi
Mathematical Problems in Engineering
Volume 2018, Article ID 2571380, 21 pages
https://doi.org/10.1155/2018/2571380

Research Article

A Bee Evolutionary Algorithm for Multiobjective Vehicle Routing Problem with Simultaneous Pickup and Delivery

Guiliang Gong,[1] **Qianwang Deng**,[1] **Xuran Gong**,[1] **Like Zhang**,[1] **Haibin Wang**,[2] **and He Xie**[2]

[1]*State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University, Changsha 410082, China*
[2]*CRRC Zhuzhou Institute Co., Ltd., Zhuzhou 412200, China*

Correspondence should be addressed to Qianwang Deng; deng_arbeit@163.com

Received 15 September 2017; Accepted 19 March 2018; Published 19 June 2018

Academic Editor: Nunzio Salerno

Copyright © 2018 Guiliang Gong et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A new closed-loop supply chain logistics network of vehicle routing problem with simultaneous pickups and deliveries (VRPSPD) dominated by remanufacturer is constructed, in which the customers are originally divided into three types: distributors, recyclers, and suppliers. Furthermore, the fuel consumption is originally added to the optimization objectives of the proposed VRPSPD. In addition, a bee evolutionary algorithm guiding nondominated sorting genetic algorithm II (BEG-NSGA-II) with a two-stage optimization mechanism is originally designed to solve the proposed VRPSPD model with three optimization objectives: minimum fuel consumption, minimum waiting time, and the shortest delivery distance. The proposed BEG-NSGA-II could conquer the disadvantages of traditional nondominated sorting genetic algorithm II (NSGA-II) and algorithms with a two-stage optimization mechanism. Finally, the validity and feasibility of the proposed model and algorithm are verified by simulating an engineering machinery remanufacturing company's reverse logistics and another three test examples.

1. Introduction

Recycling and remanufacturing is an integral system by which the old or discarded products are recycled and then processed, recovered, and sold as like-new ones in the markets [1], and the environmental impacts are also reduced by maintaining the geometrical form of the products, thus preserving its environmental and economic values [2, 3]. It is one of the important ways to realize the green sustainable development [4, 5]. For example, in recent years, China offers subsidies on a range of vehicles such as small- and medium-sized old vehicles, the rural bus, and yellow-sticker vehicles subsidy in its "automotive replacement" policy, to encourage vehicle owners to submit vehicles to officially recognized end-of-life vehicle (ELV) dismantlers [6]. Reverse logistics is a central problem of recycling and remanufacturing of waste products [7–9]. Its path planning has a great influence on distribution efficiency and environmental protection efficiency, which usually includes three main models: vehicle routing problem that delivers the goods before pickups, vehicle routing problem that delivers and pickups the goods in a mixed way, and vehicle routing problem with simultaneous pickups and deliveries (VRPSPD) of the goods, etc. [10]. The model of VRPSPD is better than other ones in environmental protection and saving cost, for it can make full use of the vehicle load space [11]. Therefore, the VRPSPD problem attracts more and more researchers in recent years.

The VRPSPD problem was first introduced by Min [12], who developed a solution procedure to solve a public library distribution with two vehicles. Based on a set of covering formulations, Klose et al. [13] implemented a Branch and Price approach for the VRPSPD. Li and Lim [14] proposed a tabu-embedded simulated annealing algorithm which restarted a search procedure from the current best solution after several nonimproving search iterations. Crispim and Brandão [15] proposed a hybrid heuristic algorithm which combined the tabu search with the falling variable field method for solving the vehicle routing problem with backhauls. Ropke and Pisinger [16] designed an adaptive large neighborhood search heuristic composed of a number of competing subheuristics, which improved many of the best known solutions. Montané and Galvao [17]

used tabu search algorithm and local optimization method for the VRPSPD problem with maximum travel constraint. Based on GLNPSO, a PSO algorithm with multiple social structures, Ai and Kachitvichyanukul [18] developed a new PSO algorithm for VRPSPD. Çatay [19] proposed an ant colony algorithm by employing a new saving-based visibility function and pheromone updating procedure. Mingyong and Erbao [20] solved the VRPSPD problem with time windows and travel constraints by improving the differential evolution algorithm. Wu et al. [21] proposed a hybrid chaotic quantum evolutionary algorithm which successfully reduced the number of vehicles and the distance of the distribution according to the data in the literature [15]. Subramanian et al. [22] proposed a branch-cut-and-price approach for the VRPSPD problem. By combining a local search named variable neighborhood descent algorithm into PSO, Goksal et al. [23] presented a heuristic approach for VRPSPD and improved several best known solutions. By analysing the vehicle load fluctuation characteristics, Zhang et al. [24] designed a heuristic factor and solved the VRPSPD problem with vehicle travel constraints based on an improved ant colony algorithm. Then Zhang et al. [25] studied the VRPSPD with time dependent vehicle routing problems and developed a hybrid algorithm that integrated both ant colony system and tabu search algorithms for solving it. Polat et al. [26] proposed a perturbation based variable neighborhood search heuristic for solving the VRPSPD problem with time limit. Avci and Topaloglu [27] developed a hybrid local search algorithm in which a nonmonotone threshold adjusting strategy is integrated with tabu search and effectively solved the VRPSPD problem with different types of vehicles. With the deterioration of the environment and the enhancement of environmental protection consciousness, the scholars have begun to take the vehicle fuel consumption as one of the optimization objectives in the past two years. Jin and Pei-Hua [28] used multistarting tabu search algorithm to study the heterogeneous vehicle low-carbon routing problem based on energy consumption and carbon emissions, then proposed a variable domain random search method, and solved the low energy vehicle routing problem about time windows [29]. By analysing the solution space and complexity of fuel-consumption-minimizing capacitated vehicle routing problem (FCM-CVRP) and capacitated vehicle routing problem (CVRP), Wu et al. [30] found that the FCM-CVRP was more difficult to solve, and they proposed a two-phase algorithm to solve it. Considering the factors of road grade, Rao et al. [31] proposed a low-carbon vehicle routing problem model which has capacity constraints and objective of minimum fuel consumption. Ombuki et al. [32] used a multiobjective evolutionary genetic algorithm based on the Pareto sort for the Vehicle Routing Problem with Time Windows (VRPTW), and its two optimization objectives are minimum number of vehicles and shortest delivery distance. Lau et al. [33] took the shortest delivery distance and the shortest travel time as the optimization objectives and used NSGA-II algorithm guided by fuzzy logic for the routing problem with multiple warehouses, multiple customers, and a variety of products. Baños et al. [34] put forward a model with two objectives of shortest distance and minimum unbalance load, which considered the workload of the vehicle, and proposed a hybrid heuristic algorithm composed of genetic algorithm and simulated annealing algorithm for it. Hsueh [35] studied the vehicle routing problem with the objectives of minimizing the sum of the fixed costs and the expected fuel consumption costs. Jemai, Zekri, and Mellouli [36] studied the Multiobjective GA for green vehicle routing problem with different metrics: computation time, traveled distance, emissions volume, generational distance, spacing, entropy, and contribution. Soleiman Chaharlang and Ghaderi [37] proposed a multiobjective nonlinear programming model for the green vehicle routing problem (GVRP), including original and remanufactured products distribution (both delivery and pickup) of end-of-life (EOL) products.

Although there is a stream of research on the VRPSPD problem and related algorithms, we find that there are still some important problems which have been overlooked and needed to be solved. It is indicated that the traditional VRPSPD model only considers a single type of customer. However, in a real recycling and remanufacturing system, customers consist of various types, such as distributors, recyclers, and suppliers. In addition, although the optimization objective of fuel consumption has been studied in some models such as FCM-CVRP and CVRP, it has not been considered in the existing literatures with the traditional VRPSPD. However, the fuel consumption is actually a key factor affecting the resource consumption and environmental destruction in VRPSPD. With regard to the overlooked problems mentioned above, in this paper the customers are originally divided into three types which are distributors, recyclers, and suppliers, and a new closed-loop supply chain logistics network of VRPSPD dominated by remanufacturer is constructed, which is more consistent with the actual situation of recycling and remanufacturing. Furthermore, for the purpose of resource saving and environment protection, the fuel consumption is originally added to the optimization objectives of VRPSPD. Besides, inspired by the literature [38], a bee evolutionary algorithm guiding nondominated sorting genetic algorithm II (BEG-NSGA-II) with a two-stage optimization mechanism is originally designed to solve the proposed VRPSPD model with three optimization objectives: minimum fuel consumption, minimum waiting time, and the shortest delivery distance. The optimal solution should meet the travel constraints as well as load constraints. The proposed BEG-NSGA-II could conquer the disadvantages (premature convergence to local solution and low searching efficiency) of the traditional NSGA-II and the disadvantage (inability to gain stable and high quality initial population in the first stage) of the existing algorithms with two-stage optimization mechanism when it is used to solve the problems with rigor constraints. Finally, the validity and feasibility of the proposed model and algorithm are verified by simulating an engineering machinery remanufacturing company's reverse logistics and another three test examples.

The rest of this paper is organized as follows. The proposed VRPSPD model is constructed in the next section. In Section 2, the multiobjective optimization model of the proposed VRPSPD is proposed. In Section 3, the implementation details of the proposed BEG-NSGA-II are presented.

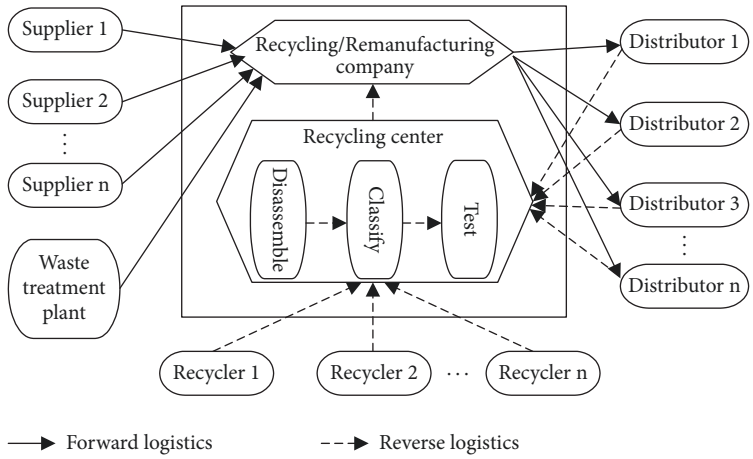


Figure 1: The proposed logistics structure.

Afterwards, experimental studies and discussions are made in Section 4. Finally, the conclusions are described in Section 5.

2. The Multiobjective Optimization Model of the Proposed VRPSPD

2.1. The Proposed VRPSPD Model. The proposed VRPSPD model in the paper is shown in Figure 1. This model is described as follows: a group of vehicles start from the recycling and remanufacturing plant with new products; then the vehicles distribute the new products to each distributor, take waste products back from each recycling business, or take new components from supply business during the set time. Some requirements must be met: the load of vehicles with waste products backed from the service points must meet the needs of all distributor points; the load of every service point cannot exceed the allowable values; each service point is serviced only once in a logistics delivery cycle.

The following assumptions are made for the network structure:

(i) Due to the uncertainty of quantity and quality of recycled products, each service point (except waste treatment plan) in the model should establish an information-sharing platform at the beginning of each logistics cycle; then, according to the information, recycling and remanufacturing factory arranges distribution vehicles.

(ii) Distributor points are sale points of remanufacturing products as well as recycling points of waste products.

(iii) Recycling process center and recycling and remanufacturing plant are located in the same location.

2.2. Fundamental Assumptions and Parameter Description. The VRPSPD model proposed in this paper is based on the following assumptions:

(i) Every vehicle moves at a constant speed between any two customers; namely, the acceleration speed is zero.

(ii) Other energy consumption of distribution vehicles (such as air conditioning) is zero, which means $Pacc = 0$ in (2).

(iii) Every vehicle departs from the recycling company and returns to it in the end.

(iv) All vehicles' maximum payload is identical and they have the same weight as well.

(v) The fuel consumption for each vehicle is only affected by three factors, namely, travel distance, payload, and travel speed.

(vi) The road slope (θ) between any two customers is set to zero, i.e., without considering the road slope of any two customers.

By analysing the operating process of recycling and remanufacturing reverse logistics network, recycling points and suppliers can be regarded as special distribution points. So the vehicle routing problem of the proposed model can be changed into VRPSPD. $V = \{v_0, v_1, v_2, \ldots, v_n, r_1, r_2, \ldots, r_m, p_1, p_2, \ldots, p_s\}$ is used to express the set of all service points, in which v_0 is the recycling and remanufacturing plant, $\{v_1, v_2, \ldots v_n\}$is the set of distributors, $\{r_1, r_2, \ldots, r_m\}$ is the set of recycling points, and $\{p_1, p_2, \ldots, p_s\}$ is the set of suppliers. Use the corresponding sequence number of each element of V to form a new set V'; i.e., $V' = \{0, 1, \ldots, n, n+1, \ldots n+m, \ldots, n+m+s\}$.

The parameters used in the definition of multiobjective VRPSPD are as follows:

(i) G_m: the vehicle capacity

(ii) G_0: the weight of an empty vehicle

(iii) L: the maximum travel distance of a vehicle

(iv) d_{ij}: the distance between any two service points i and j

(v) v_{ij}: the speed between any two service points i and j

(vi) d_i: the demand quantity of service point i

(vii) p_i: the pickup quantity that service point i has

(viii) K: the set of vehicles

(ix) k: the vehicle of k, $k \in K$

(x) n_k: the number of vehicles

Table 1: The meanings of letter symbols in (2).

Letter symbols	Meanings
φ	Ratio of fuel to air weight
N	Engine speed
k	Engine friction factor
V_s	Engine exhaust volume
g	Acceleration of gravity
η_{tf}	Transmission efficiency
η	Engine efficiency
A	Truck windward area
C_d	Air resistance coefficient
ρ	Air density
d	Travel distance
P_{acc}	Other energy needs
C_r	Rolling resistance coefficient
a	Vehicle acceleration
M	Mass of vehicle and products
v	Vehicle travel speed
θ	The road slope

(xi) p_{ik}: the pickup quantity taken by k vehicle from service point i, where $p_{ik} = p_i$

(xii) q_{ijk}: the load of vehicle k after it accessed i before it accesses j

(xiii) e_0: the start time of vehicle from V_0

(xiv) S_i: the serve time of point i

(xv) L_i: the start time of point i required for delivery.

If the real delivery time for point i is later than L_i, then there exists waiting time which can be formulated as follows:

$$W_{\mathrm{i}} = \max\left(\mathrm{S_i} - L_{\mathrm{i}}, 0\right) \tag{1}$$

2.3. The Model of Fuel Consumption. Many scholars have carried out a lot of researches on the model of vehicle energy consumption and have made a series of research achievements. One of the classic models for the energy consumption of vehicle engine was based on the research of Barth and Boriboonsomsin [39]. It is shown in (2) and the meanings of the letter symbols in (2) are shown in Table 1.

$$\begin{aligned} FR &= \varphi\left(\mathrm{kN}V_s + \frac{P}{\eta}\right) \\ P &= \frac{P_{tract}}{\eta_{tf}} + P_{acc} \\ P_{tract} &= \frac{Mv}{1000}\left(\mathrm{a} + g\sin\theta + gC_r\right) + \frac{0.5C_d\rho A v^3}{1000} \end{aligned} \tag{2}$$

In (2), FR denotes fuel consumption rate, which means energy consumption for a vehicle per unit time.

According to the assumption, we can get the vehicle fuel consumption from service point i to point j by

$$\mathrm{e}_{ij} = FR_{ij} \times \frac{d_{ij}}{v_{ij}} \tag{3}$$

2.4. The Model of Travel Distance. According to the assumption above, all vehicles' travel distance can be calculated by

$$D = \sum_{k=1}^{K}\sum_{i\in V}\sum_{j\in V} x_{ijk}d_{ij}, \quad k \in K \tag{4}$$

2.5. Objective Functions. The optimization objectives and constraints in our multiobjective optimization model are defined as follows:

$$\min \quad W_t = \sum_{i=1}^{n+m+s} \max\left(\mathrm{S}_i - L_i, 0\right) \tag{5}$$

$$\min \quad Q = \sum_{k=1}^{K}\sum_{i=0}^{n+m+s}\sum_{j=0}^{n+m+s}\left(x_{ijk}\cdot e_{ij}\right) \tag{6}$$

$$\min \quad D = \sum_{k=1}^{K}\sum_{i=0}^{n+m+s}\sum_{j=0}^{n+m+s}\left(x_{ijk}d_{ij}\right) \tag{7}$$

$$\text{Subject to:} \quad \sum_{k\in K}\sum_{i\in V} x_{ijk} = 1, \quad \forall j \in V \tag{8}$$

$$\sum_{i\in V} x_{ijk} - \sum_{i\in V} x_{jik} = 0, \quad \forall j \in V,\ \forall k \in K \tag{9}$$

$$\sum_{i\in V} x_{ik} \le 1, \quad k \in K \tag{10}$$

$$\sum_{j\in V} q_{ijk} = \sum_{i\in V}\sum_{j\in V} x_{ijk}d_i, \quad \forall k \in K \tag{11}$$

$$\sum_{k\in K}\sum_{i\in V} q_{ijk} - d_j = \sum_{k\in K}\sum_{i\in V} q_{jik} - p_j, \quad \forall j \in V \tag{12}$$

$$\sum_{i\in V} q_{ijk} = \sum_{i\in V}\sum_{j\in V} x_{ijk}p_j, \quad \forall k \in K \tag{13}$$

$$q_{ijk} + p_{jk} - d_{jk} \le G_m, \quad \forall k \in K,\ \forall i \in V,\ \forall j \in V \tag{14}$$

$$\sum_{i\in V}\sum_{j\in V} x_{ijk}\cdot d_{ij} \le L, \quad \forall k \in K \tag{15}$$

$$x_{ijk} = \begin{cases} 1, & \text{Vehicle access to } j \text{ after access } i \\ 0, & \text{Othervise} \end{cases} \tag{16}$$

Equations (5), (6), and (7) indicate the three optimization objectives which are minimum waiting time, fuel consumption, and delivery distance. Equation (8) ensures that each service point must be served once and only once. Equation (9) ensures that each service point must be served by one and only one vehicle.

Equation (10) ensures that each vehicle can only be used at most once. Equation (11) ensures that the vehicle's load starting from V_0 is equal to the demand of all the customers that it serves. Equation (12) is the vehicle's load change equation that indicates the load after vehicle k severed i before accessing j. Equation (13) ensures that the load of vehicle k equals the pickup quantity of all customers which are severed by vehicle k. Equation (14) ensures that each vehicle's load cannot exceed its maximum load, and (15) ensures that each vehicle's travel distance is not more than the maximum distance.

3. The Proposed Algorithm BEG-NSGA-II

3.1. Framework. In this paper, a two-stage optimization mechanism is used to solve the proposed VRPSPD with strict constraints. In the first stage, we use an improved bee evolutionary genetic algorithm to optimize the initial population. The optimization efficiency of the first stage is improved by optimizing the selection operators, selecting the different crossover operators according to the similarity of individuals' parent chromosomes, and selecting the different mutation operators according to the performance of individual's parent chromosome. In the second stage, an improved NSGA-II is used to optimize the proposed VRPSPD model. Based on the improved crossover and mutation operators, we construct the methods of deleting duplicate individuals, introducing new individuals, and using elite population instead of the parent population to improve the population diversity. To deal with the strict constraints, we introduce an external auxiliary population. During the optimizing process, if the constraint violation degree of the infeasible solutions is smaller than a given value, these infeasible solutions are copied to the external auxiliary population and used to evolve in the next generation. By using this external auxiliary population, the infeasible solutions gradually evolve to the boundary of feasible solutions, and the convergence speed is accelerated. The framework of the proposed algorithm is shown in Figure 2. More details of steps are described as follows.

The first stage includes the following.

Step 1. Randomly generate N different chromosomes to generate the initial population, and set $t = 1$, where N is the number of chromosomes and t is the current generation.

Step 2. Calculate the constraint violation value of each chromosome with (17). If the number of individuals that meets the constraint conditions is greater than the specified value or $t > T$ (T is the maximal iteration time in the first stage), jump over the first phase; otherwise go to *step 3*.

$$Const(p) = \sum_{i=1}^{I} \max\left[0, Drift_{pi} - Cap_i\right] \quad (i = 1, 2, \ldots, I) \tag{17}$$

where $Drift_{pi}$ denotes the *ith* constraint condition value of chromosome P after decoding; Cap_i denotes the upper bound of the *ith* constraint condition.

Step 3. Calculate each chromosome's fitness based on the degree of constraint violation using (18); then select the best fitness individual as the queen; if $t > 1$, compare the fitness value of the queen with the parent queen, and select the better one as the new queen; if the number of feasible solutions is more than one, use nondominated sorting and congestion degree computing to deal with the feasible solutions to ensure that the queen selected from the parent population is the best one, then compare it with the parent queen, and select the better one as the new queen.

$$fitness(p) = \frac{1}{(Const(p) + 1)} \tag{18}$$

Step 4. Select γ individuals from the population using roulette wheel method. The number of γ is calculated by (19), where round [ ] means the operation of round; for example, round [2.3] equals 2.

$$\gamma = \text{round}\left[\frac{t}{T} \times \frac{N}{2}\right] \tag{19}$$

Step 5. Randomly generate β individuals different from each other, and combine them with the individuals generated from *step 4*. The number of β is calculated by

$$\beta = \frac{N}{2} - \gamma \tag{20}$$

Step 6. The new queen takes crossover and mutation with other individuals generated from *step 5* to produce the offspring population.

Step 7. Delete the redundant repeating individuals and randomly generate M individuals different from each other into the population, where M equals the number of redundant repeating individuals that were deleted.

Step 8. Return to *step 2*, and set $t = t + 1$.

The second stage includes the following.

Step 9. Take fast nondominated sorting and crowding degree calculation on population.

Step 10. Carry out the following three kinds of operation for the population generated by *step 9*: select the top S (S=0.3N) individuals as the elite population; copy the individuals whose constraint violation values are less than a given value to the external auxiliary population S^*; use a binary tournament selection method to select crossover individuals and they were taken crossover according to the corresponding selected crossover methods to generate offspring.

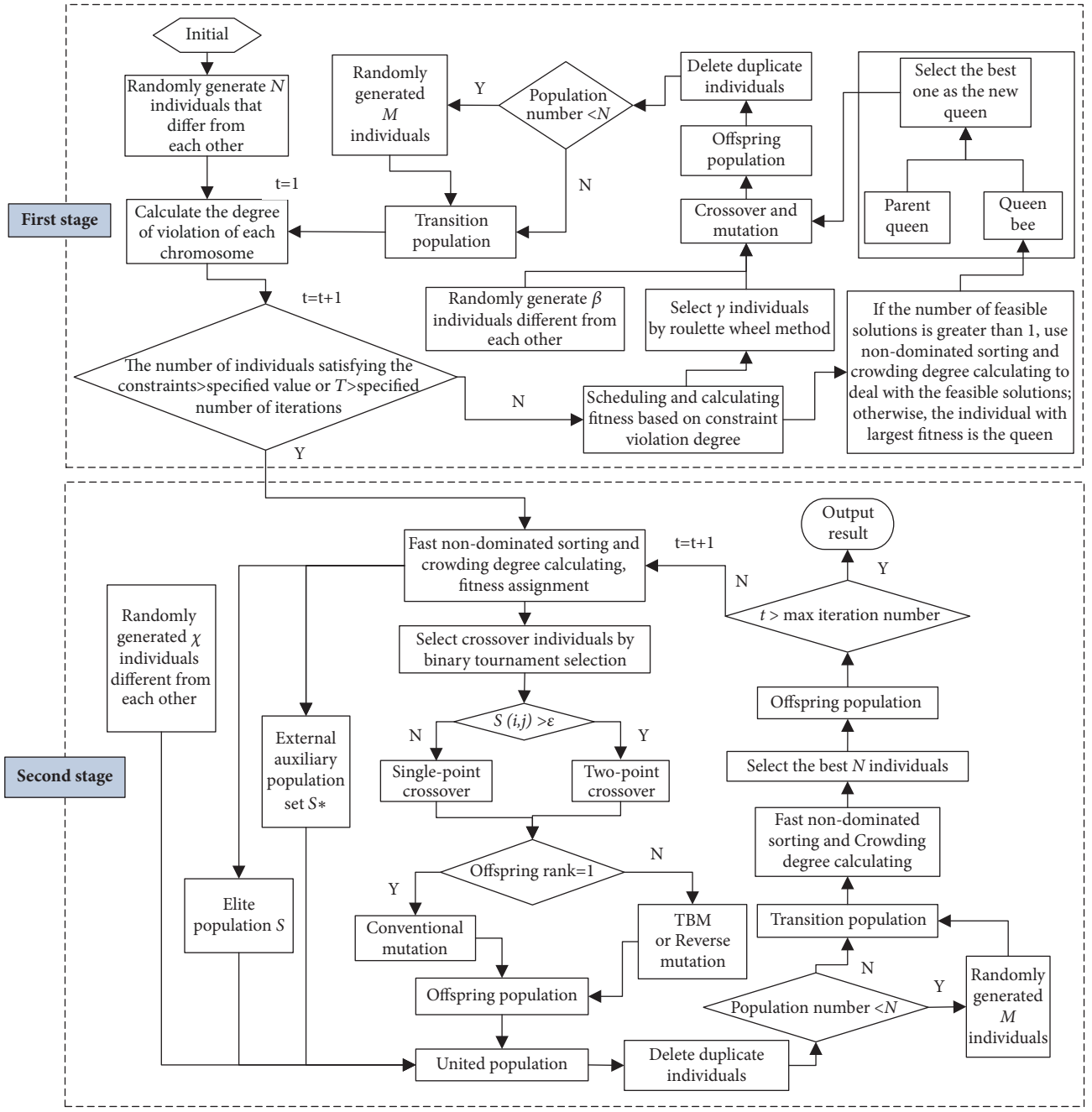


Figure 2: The framework of BEG-NSGA-II for VRPSPD.

Step 11. The individuals take mutation with the probability of 0.1 after crossover. If the offspring rank = 1, then select the conventional mutation; otherwise the two-binding mutation (TBM) or reverse mutation is selected randomly.

Step 12. Randomly generate χ individuals different from each other. The number of χ is calculated by

$$\chi = \frac{N}{2} - \text{round}\left[\frac{t}{GEN} \times \frac{N}{2}\right] \tag{21}$$

where t denotes the current iterations and GEN denotes the total iterations. Round [ ] denotes rounding operating; if $\chi < 5$, set $\chi = 5$.

Step 13. Combine the individuals of S, $S*$, and χ and the individuals taken from *step 11*.

Step 14. Use *step 7* and *step 9* to deal with the individuals from *step 13*, and select top N individuals as the offspring population.

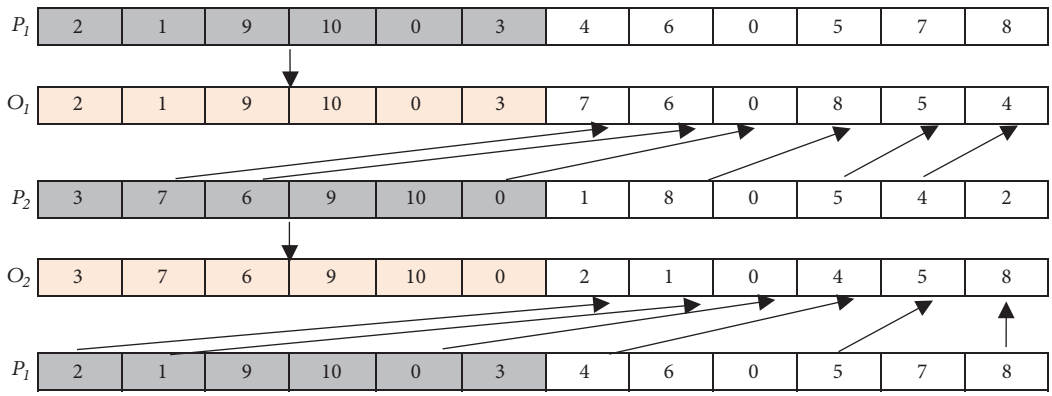


Figure 3: Singe-point crossover.

Step 15. If $t > GEN$, output the result; otherwise, $t = t + 1$, and return *step 10*.

3.2. Chromosome Representation. According to the characteristics of logistics path planning, a string with different integers (the length equals L, where L is the number of customers) is used to express a chromosome. Then n_k - 1 zeros are inserted into this string, where n_k is the number of distribution vehicles and zeros express the distribution centers. For example, if there are 6 customers and three vehicles to serve them, the string of 25603104 expresses the distribution paths which are path 0-2-5-6-0, path 0-3-1-0, and path 0-4-0.

3.3. Crossover Operators. In this paper, in order to enhance the searching space and avoid prematurity of local optimal solutions, we use two types of crossover operators (single-point crossover and two-point crossover operator) in this proposed algorithm. The single-point crossover operator helps to play excellent high genetic characteristics and to improve the convergence speed. Nevertheless, when the two parent generations become identical or similar, the crossover offspring almost get no change especially in the middle and later periods of the evolution. Hence, a two-point crossover operator is proposed for its ability to improve the mentioned problems. Equation (22) is a difference degree function. The crossover operator is chosen by (23), where L is the length of the chromosome and a_{il} denotes the first gene of *ith* chromosome.

$$S(i,j) = \frac{\sum_{l=1}^{L} \left| \min\left[\max\left[0, \left| a_{il} - a_{jl} \right| \right] - 1, 0 \right] \right|}{L} \tag{22}$$

$$\begin{aligned} &S(i,j) \le \varepsilon, \quad \text{Single-point crossover} \\ &S(i,j) > \varepsilon, \quad \text{Two-point crossover} \end{aligned} \tag{23}$$

The procedure of single-point crossover is described as follows (P_1 and P_2 are used to denote two parents; O_1 and O_2 are used to denote two offspring).

Step 1. A random parameter k that meets the inequality $0<k<L$ (the length of the chromosome) is generated to determine the position of the crossover.

Step 2. The elements from 1 to k in P_1 are duplicated to O_1 in the same positions; and the elements from 1 to k in P_2 are duplicated to O_2 in the same positions.

Step 3. The elements in P_2 but not in O_1 are duplicated to the remaining empty positions in O_1 from left to right.

Step 4. The elements in P_1 but not in O_2 are duplicated to the remaining empty positions in O_2 from left to right.

The procedure of two-point crossover is described as follows (P_1 and P_2 are used to denote two parents; O_1 and O_2 are used to denote two offspring).

Step 1. Two random parameters k_1 and k_2 that meet the inequality $0 < k_1 < k_2 < L$ as well as $k_1 \neq k_2$ are generated to determine the positions of crossover.

Step 2. The elements from k_1 to k_2 in P_1 are appended to the leftmost positions of O_1; and the elements from k_1 to k_2 in P_2 are appended to the leftmost positions of O_2.

Step 3. The elements in P_2 but not in O_1 are duplicated to the remaining empty positions in O_1 from left to right.

Step 4. The elements in P_1 but not in O_2 are duplicated to the remaining empty positions in O_2 from left to right.

The examples of singe-point crossover and two-point crossover are, respectively, shown in Figures 3 and 4.

3.4. Mutation Operators. In this paper, we have adopted three different mutation operators for the purpose of expending the solution space as well as maintaining the good solutions, which are conventional mutation operator, reverse mutation operator, and TBM operator. The examples of these mutation operators are shown in Figures 5, 6, and 7. At this stage, every offspring inherits the excellent traits of the queen for it is generated by the mutation of individuals and queen. So we

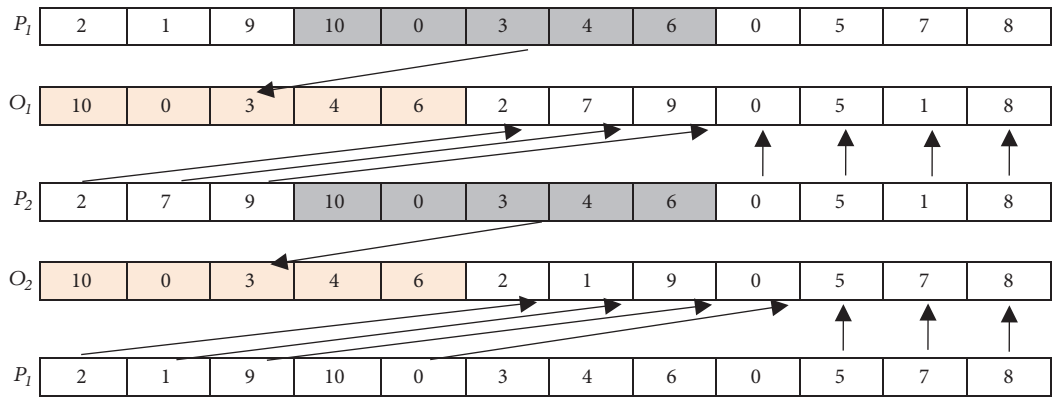


Figure 4: Two-point crossover.

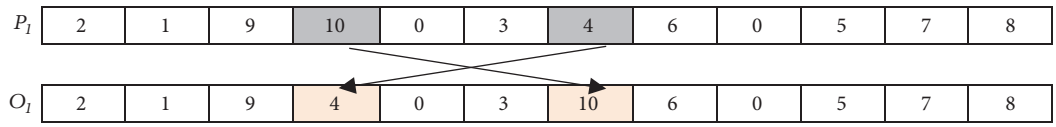


Figure 5: Conventional mutation.

randomly chose a mutation operator with P_m probability at this stage. At the second stage, if the rank of offspring equals 1, we select the conventional mutation; otherwise we randomly select reverse or TBM mutation.

The main procedure of conventional mutation operator is described as follows (P_1 and O_1 are used to denote a parent and offspring, respectively).

Step 1. Randomly select two positions in P_1.

Step 2. Swap the elements in the selected positions to generate O_1.

The main procedure of reverse mutation operator is described as follows (P_1 and O_1 are used to denote a parent and offspring, respectively).

Step 1. Randomly select two positions in P_1.

Step 2. Reverse the numbers between the two selected positions to generate O_1.

The main procedure of TBM operator is described as follows (P_1 and O_1 are used to denote a parent and offspring, respectively).

Step 1. A random parameter m ($m < L - 3$, where L is the length of the chromosome) is generated in P_1.

Step 2. Exchange the elements m with $m+3$ and $m+1$ with $m+4$ in P_1 to generate O_1.

4. Experimental Results

The proposed BEG-NSGA-II algorithm was coded in MATLAB R2014a and implemented on a computer configured with Intel Core i3 CPU with 2.67 GHz frequency and 8GB RAM. Three sets of examples are used to illustrate the performance of the proposed algorithm, in which three of the same trucks were used to deliver and pick up the goods. The first one is the VRPSPD problem of an actual engineering machinery remanufacturing company (with 12 customers). To further verify the effectiveness of the proposed algorithm, another two instances with 22 customers and 42 customers are used to simulate the problem of VRPSPD.

Because the problem of this paper has no benchmarks, we generate the test instances based on the SCA3-0 instance from Dethloff [40]. The generating standards are described as follows: n customers are randomly generated that are distributed in a square area whose length is 100 km; d_i is randomly generated within $[0.5ton, 5ton]$; $p_i = 0.5(d_i * r+1)$, where r is randomly generated within $[0, 1]$; the number of distributors, recyclers, and suppliers is randomly generated within $[d_i, 2*d_i]$, and the total number of distributors, recyclers, and suppliers equals the total customers that we set (such as 12 customers).

The values of parameters are shown in Table 2 [30].

In addition, G_m=25ton, G_0=10ton, and β=1, and the best value of v_{ij} is 17m/s.

The adopted parameters of the proposed algorithm are set as follows: $N = 100$, $p_c = 1$, $p_m = 0.1$, $T = 100$, and $GEN = 1000$. In order to demonstrate the effectiveness and performance of the proposed algorithm, the traditional algorithm and the proposed algorithm are run 10 times, respectively.

4.1. Test Example 1. Table 3 shows an example of VRPSPD problem of an actual engineering machinery remanufacturing company with 12 customers. The first column shows the information of the customers (there are 1 recycling center, 6 distributers, 3 recyclers, and 3 suppliers). In the second column, in order to facilitate encoding and decoding in

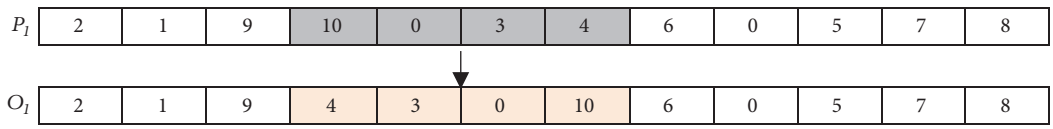


Figure 6: Reverse mutation.

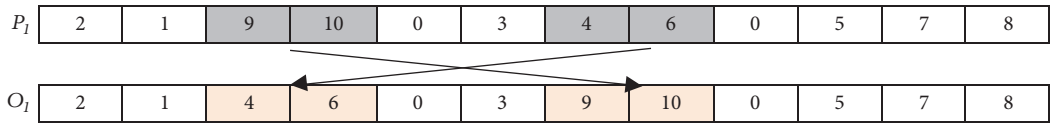


Figure 7: Two-binding mutation.

Table 2: The values of parameters used in the proposed VRPSPD model.

Parameter	value
φ	1
N	25r/m
k	0.2
V_s	$8L$
g	9.8N/kg
η_{tf}	*0.45*
η	0.6
A	0.4
C_d	0.01
ρ	$1.2041[kg/m^3]$
C_r	$4.5m^2$
P_{acc}	0
d	-
a	0
M	-
v	-
θ	0

the algorithm, all customers were numbered as a series continuous integer (recycling center 0 is numbered as 0; distributors 1, 2, ..., and 6 are numbered as 1, 2, ..., and 6; recyclers 1, 2, and 3 are numbered as 7, 8, and 9; suppliers 1, 2, and 3 are numbered as 10, 11, and 12.). The third and fourth columns show the X/km and Y/km coordinates of the customers, respectively. The amounts of demand/ton and recycle/ton of the customers are shown in columns 5 and 6. What is more, column 7 determines the service time of the customers (for example, 10 am-3 pm means that the service time of the customer is needed from 10 am until 3 pm).

In order to demonstrate the performance of the proposed algorithm and research the effect of the duplicate individuals, test example 1 was tested by the traditional NSGA-II (T-NSGA-II), BEG-NSGA-II without eliminating the duplicate individuals (W-BEG-NSGA-II), and BEG-NSGA-II, respectively. Table 4 shows the optimization results of test example 1 using these three algorithms. From the optimization results, we can gain the following information:

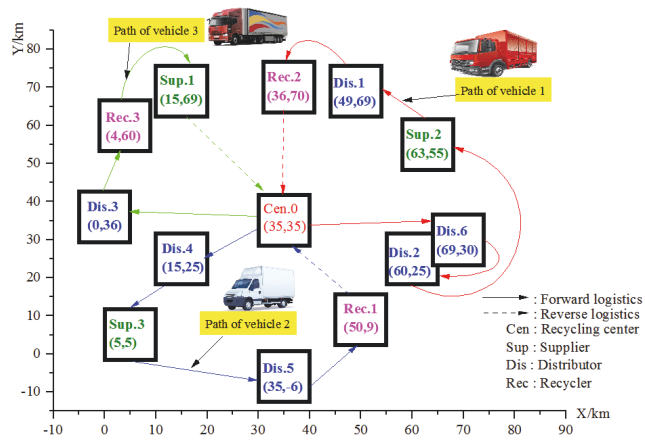


Figure 8: Solution 1 of test example 1 (fuel consumption=418.99kg, waiting time = 95.5 min, and distance =371.69km).

(i) The algorithms of T-NSGA-II, W-BEG-NSGA-II, and BEG-NSGA-II could find 5, 6, and 6 optimal Pareto solutions with average computing time of 1.2, 0.9, and 0.5 minutes correspondingly during 10 run times.

(ii) The algorithm of BEG-NSGA-II could find 2 Pareto solutions (bold and marked as "a" in Table 4) that dominate the results of T-NSGA-II and 6 Pareto solutions (bold and marked as "b" in Table 4) that dominate the results of W-BEG-NSGA-II.

From the information mentioned above, we can draw the following conclusions:

(i) The algorithms of T-NSGA-II and W-BEG-NSGA-II may generate amount of duplicate individuals, especially after 100 iterations. These duplicate individuals may cause the problems of inefficiency, slow convergence, and converging to local Pareto optimal solutions.

(ii) The proposed algorithm of BEG-NSGA-II is more effective than the algorithms of T-NSGA-II and W-BEG-NSGA-II to solve the VRPSPD problem with rigor constraints.

Figures 8 and 9 show two optimization Pareto solutions (0-2-6-11-1-8-0 0-4-12-5-7-0 0-3-9-10-0; 0-6-2-11-1-8-0 0-4-12-5-7-0 0-3-9-10-0) of BEG-NSGA-II. Taking the solution (0-2-6-11-1-8-0 0-4-12-5-7-0 0-3-9-10-0), for example, the

Table 3: The basic information of test example 1 with 12 customers.

Customer	Number	X/km	Y/km	Demand/ton	Recycle/ton	Service time
Cen. 0	0	35	35	-	-	-
Dis. 1	1	49	69	8	2.5	8 am-10 am
Dis. 2	2	60	25	6	2	10 am-12 am
Dis. 3	3	0	36	12	6	8 am-10 am
Dis. 4	4	15	25	9	5	8.5 am-10.5 am
Dis. 5	5	35	-6	9	4	9.5 am-11.5 am
Dis. 6	6	69	30	11	5	9 am-12 am
Rec.1	7	50	9	0	8	10 am-3 pm
Rec.2	8	36	70	0	8	10 am-3 pm
Rec.3	9	4	60	0	10	10 am-3 pm
Sup.1	10	15	69	0	6.3	10 am-3 pm
Sup.2	11	63	55	0	6.3	10 am-3 pm
Sup.3	12	5	5	0	6.3	10 am-3 pm

Note: (a) Cen., Dis., Rec., and Sup. are the abbreviations of recycling center, distributors, recycler, and supplier, respectively.
(b) "-" means that these data do not need to be set in advance.

Table 4: Comparisons of the optimization results of test example 1 with 12 customers.

Algorithms	Vehicle path			Fuel consumption/kg	Waiting time/min	Distance/km	Average computing time/min
T-NSGA-II	0-3-1-8-10-0	0-4-12-5-7-0	0-2-11-6-9-0	546.39	0.74	489.70	1.2
	0-3-1-8-10-0	0-4-12-5-7-0	0-11-2-6-9-0	548.06	0.74	481.77	
	0-3-8-1-10-0	0-4-12-5-7-0	0-2-11-6-9-0	546.33	31.28	493.11	
	0-3-8-1-10-0	0-4-12-5-7-0	0-11-2-6-9-0	548.00	31.28	485.18	
	0-3-10-1-8-0	0-4-12-5-7-0	0-2-11-6-9-0	539.77	50.13	475.41	
W-BEG-NSGA-II	0-4-5-7-10-0	0-2-6-11-1-8-0	0-3-9-12-0	512.69	95.5	476.94	0.9
	0-4-5-7-10-0	0-6-2-11-1-8-0	0-9-3-12-0	506.21	147.4	458.14	
	0-4-5-7-10-0	0-6-2-11-1-8-0	0-3-9-12-0	525.85	73.8	488.82	
	0-4-5-7-10-0	0-6-2-11-1-8-0	0-9-3-12-0	519.36	125.6	470.02	
	0-5-7-4-10-0	0-2-6-11-1-8-0	0-9-3-12-0	519.20	180.9	452.91	
BEG-NSGA-II	0-2-6-11-1-8-0	0-4-12-5-7-0	0-3-9-10-0	418.99	95.5	371.69	0.5
	0-6-2-11-1-8-0	**0-4-12-5-7-0**	**0-3-9-10-0**	**432.14**	**73.7**	**383.57[b]**	
	0-6-2-1-8-11-0	**0-4-12-5-7-0**	**0-3-9-10-0**	**449.21**	**63.0**	**409.26[b]**	
	0-6-2-1-9-10-0	**0-4-12-5-7-0**	**0-3-8-1-11-0**	**475.03**	**31.3**	**444.04[b]**	
	0-6-2-1-10-9-0	**0-4-12-5-7-0**	**0-3-8-1-11-0**	**477.91**	**31.3**	**441.32[b]**	
	0-1-6-2-11-8-0	**0-4-12-5-7-0**	**0-3-9-10-0**	**479.98**	**0**	**427.85[a,b]**	
	0-1-2-6-11-8-0	**0-4-12-5-7-0**	**0-3-9-10-0**	**484.08**	**0**	**424.94[a,b]**	

Note: (a) "[a]"meant that the marked result dominates the results of T-NSGA-II.
(b) "[b]" meant that the marked result dominates the results of W-BEG-NSGA-II.
(c) "[a,b]" meant that the marked result dominates the results of T-NSGA-II and W-BEG-NSGA-II simultaneously.

path of vehicle 1 is 0-2-6-11-1-8-0; i.e., vehicle 1 starts from recycling center (numbered 0), passes by distributor 2 (numbered 2), distributor 6 (numbered 6), supplier 2 (numbered 11), distributor 1 (numbered 1), and recycler 2 (numbered 8), and then goes back to recycling center (numbered 0); the path of vehicle 2 is 0-4-12-5-7-0; i.e., vehicle 2 starts from recycling center (numbered 0), passes by distributor 4 (numbered 4), supplier 3 (numbered 12), distributor 5 (numbered 5), and recycler 1 (numbered 7), and goes back to recycling center (numbered 0); the path of vehicle 3 is 0-3-9-10-0; i.e., vehicle 3 starts from recycling center (numbered 0), passes by distributor 3 (numbered 3), recycler 3 (numbered 9), and supplier 1 (numbered 10), and then goes back to recycling center (numbered 0). And its fuel consumption/kg, waiting time/min, and distance/km are 418.99 kg, 95.5 min, and 371.69 km, respectively. Figure 10 shows the number of duplicate individuals which changes with the number of iterations in test example 1.

Table 5: The basic information of test example 1 with 22 customers.

	Number	X/km	Y/km	Demand/ton	Recycle/ton	Service time
Cen. 0	0	50	50	-	-	-
Dis. 1	1	41	60	3	1.5	8 am-10 am
Dis. 2	2	39	40	2	1	10 am-12 am
Dis. 3	3	60	58	4	2	8 am-10 am
Dis. 4	4	58	42	3.5	2	8.5 am-10.5 am
Dis. 5	5	18	86	3	0.5	9.5 am-11.5 am
Dis. 6	6	4	12	3	1.5	9 am-12 am
Dis. 7	7	90	6	2	2	10 am-3 pm
Dis. 8	8	92	86	4	2	10 am-3 pm
Dis. 9	9	25	76	2.5	2	10 am-3 pm
Dis. 10	10	30	40	3	2	10 am-3 pm
Dis. 11	11	72	36	2.5	1	10 am-3 pm
Dis. 12	12	68	80	3	1.5	10 am-3 pm
Dis. 13	13	22	69	3.5	2.5	10 am-3 pm
Dis. 14	14	28	44	2	1.5	8.5 am-10.5 am
Dis. 15	15	56	36	4	1.5	9.5 am-11.5 am
Rec.1	16	64	62	0	6	9 am-12 am
Rec.2	17	32	80	0	7	10 am-3 pm
Rec.3	18	28	36	0	7	10 am-3 pm
Sup.1	19	62	40	0	6	10 am-3 pm
Sup.2	20	63	55	0	6.3	10 am-3 pm
Sup.3	21	14	70	0	6.3	10 am-3 pm
Sup.4	22	20	30	0	6.3	10 am-3 pm

Note: (a) Cen., Dis., Rec., and Sup. are the abbreviations of recycling center, distributors, recycler, and supplier, respectively.
(b) "-" means that these data do not need to be set in advance.

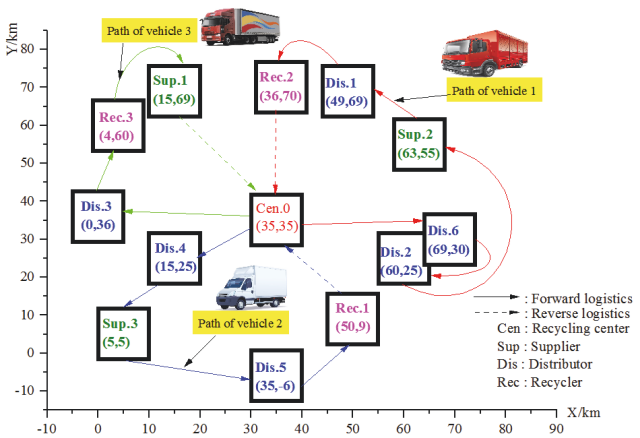


Figure 9: Solution 2 of test example 1 (fuel consumption=432.14kg, waiting time = 73.7 min, and distance = 383.57km).

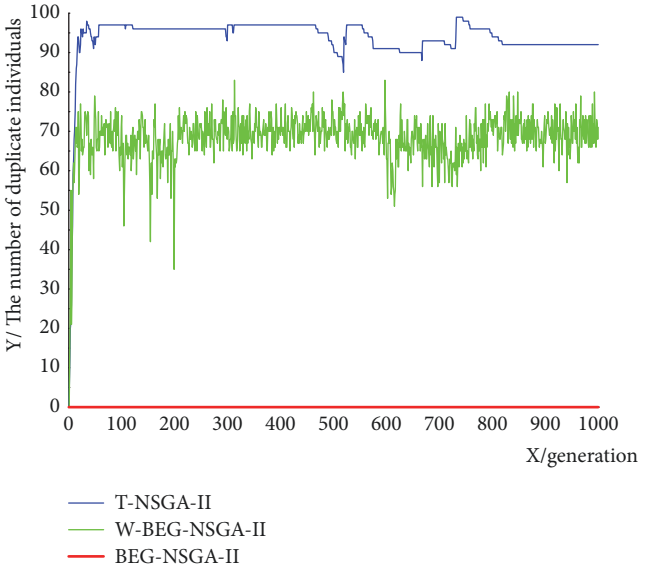


Figure 10: The number of duplicate individuals in test example 1 with 12 customers.

4.2. Test Example 2. Table 5 shows an example of VRPSPD problem with 22 customers. The first column shows the information of the customers (there are 1 recycling center, 15 distributers, 3 recyclers, and 4 suppliers). In the second column, in order to facilitate encoding and decoding in the algorithm, all customers were numbered as a series continuous integer (recycling center 0 is numbered 0; distributors 1, 2, ..., and 15 are numbered 1, 2, ..., and 15; recyclers 1, 2, and 3 are numbered 16, 17, and 18; suppliers 1, 2, and 3 are numbered 20, 21, and 22.). The third and fourth columns show the X/km and Y/km coordinates of the customers, respectively. The amounts of demand/ton and recycle/ton of the customers

Table 6: Comparisons of the optimization results of test example 2 with 22 customers.

Algorithms	Vehicle path			Fuel consumption/kg	Waiting time/min	Distance/km	Average computing time/min
T-NSGA-II	0-1-4-7-11-19-13-2-6-22-0	0-18-14-21-5-9-20-0	0-15-3-16-12-8-10-17-0	702.56	109.37	708.43	2.5
	0-4-15-7-11-19-2-13-6-22-0	0-18-14-21-5-9-16-0	0-1-3-20-12-8-10-17-0	654.17	125.09	676.71	
	0-4-15-7-11-19-2-13-6-22-0	0-18-14-21-5-9-16-0	0-1-20-3-12-8-10-17-0	652.72	141.55	678.08	
	0-4-15-11-7-19-2-13-6-22-0	0-18-14-21-5-9-16-0	0-1-3-20-12-8-10-17-0	650.68	150.43	680.65	
	0-4-15-11-19-1-13-2-6-22-0	0-18-14-21-5-9-20-0	0-3-7-16-12-8-10-17-0	698.33	96.99	712.65	
	0-4-15-19-11-1-13-2-6-22-0	0-18-14-21-5-9-20-0	0-3-7-16-12-8-10-17-0	695.59	122.56	714.06	
	0-4-15-19-11-7-2-13-6-22-0	0-18-14-21-5-9-16-0	0-1-3-20-12-8-10-17-0	644.90	161.72	676.88	
	0-4-15-19-11-7-2-13-6-22-0	0-18-14-21-5-9-16-0	0-1-20-3-12-8-10-17-0	643.45	178.18	678.25	
	0-4-15-19-11-7-2-13-6-22-0	0-18-14-21-9-5-16-0	0-1-3-20-12-8-10-17-0	644.72	168.08	683.36	
	0-4-15-19-11-7-2-13-6-22-0	0-18-14-21-9-5-16-0	0-1-20-3-12-8-10-17-0	643.26	184.54	684.73	
W-BEG-NSGA-II	**0-3-8-12-17-9-13-21-0**	**0-4-7-11-19-20-16-0**	**0-1-5-14-10-2-6-22-15-18-0**	**597.48**	**70.17**	**591.05**[a]	1.6
	0-3-8-12-17-9-13-21-0	**0-4-7-11-19-20-16-0**	**0-1-14-5-10-2-6-22-15-18-0**	**631.29**	**66.56**	**620.02**[a]	
	0-3-8-12-17-9-13-21-0	**0-4-11-7-19-16-20-0**	**0-1-5-14-10-2-6-22-15-18-0**	**590.77**	**70.17**	**593.94**[a]	
	0-3-8-12-17-9-13-21-0	**0-4-11-7-19-16-20-0**	**0-1-14-5-10-2-6-22-15-18-0**	**624.58**	**66.56**	**622.91**[a]	
	0-3-8-12-17-9-13-21-0	**0-4-19-7-11-16-20-0**	**0-1-5-14-10-2-6-22-15-18-0**	**599.79**	**73.82**	**588.29**[a]	
	0-3-8-12-17-9-13-21-0	**0-4-19-11-7-16-20-0**	**0-1-5-14-10-2-6-22-15-18-0**	**585.67**	**75.06**	**589.56**[a]	
	0-3-8-12-17-9-13-21-0	**0-4-19-11-7-20-16-0**	**0-1-5-14-10-2-6-22-15-18-0**	**585.60**	**82.61**	**588.27**[a]	
	0-4-11-7-19-16-20-0	**0-3-8-12-17-9-13-21-0**	**0-1-5-14-10-2-6-22-15-18-0**	**590.77**	**70.17**	**593.94**[a]	
	0-4-11-7-19-16-20-0	**0-3-8-12-17-9-13-21-0**	**0-1-14-5-10-2-6-22-15-18-0**	**624.58**	**66.56**	**622.91**[a]	
	0-4-19-7-11-20-16-0	**0-3-8-12-17-9-13-21-0**	**0-1-5-14-10-2-6-22-15-18-0**	**599.76**	**80.99**	**586.63**[a]	
	0-4-19-11-7-16-20-0	**0-3-8-12-17-9-13-21-0**	**0-1-5-14-10-2-6-22-15-18-0**	**585.67**	**75.06**	**589.56**[a]	
	0-4-19-11-7-16-20-0	**0-3-12-8-17-9-13-21-0**	**0-1-5-14-10-2-6-22-15-18-0**	**585.60**	**81.80**	**594.75**[a]	
BEG-NSGA-II	0-4-15-2-10-6-22-18-14-0	0-1-17-16-3-11-19-0	0-9-13-21-5-12-8-7-20-0	544.83	147.38	569.64	0.9
	0-4-15-2-10-6-22-18-14-0	0-9-17-3-16-11-19-0	0-1-13-21-5-12-8-7-20-0	543.80	147.15	570.21	
	0-4-15-2-10-6-22-18-14-0	0-9-17-16-3-11-19-0	0-1-13-21-5-12-8-7-20-0	540.38	159.78	569.18	
	0-4-15-2-10-6-22-18-14-0	0-17-9-16-3-11-19-0	0-1-13-21-5-12-8-7-20-0	537.94	164.41	572.82	
	0-4-15-2-10-14-6-22-18-0	**0-1-17-3-16-11-19-0**	**0-9-13-21-5-12-8-7-20-0**	**546.79**	**47.91**	**572.21**[a,b]	
	0-4-15-2-10-14-6-22-18-0	**0-1-17-16-3-11-19-0**	**0-9-13-21-5-12-8-7-20-0**	**543.37**	**60.54**	**571.17**[a,b]	
	0-4-15-2-10-14-6-22-18-0	**0-3-9-17-16-11-19-0**	**0-1-13-21-5-12-8-7-20-0**	**542.87**	**0**	**583.29**[a,b]	
	0-4-15-2-10-14-6-22-18-0	**0-3-16-17-9-11-19-0**	**0-1-13-21-5-12-8-7-20-0**	**539.74**	**0**	**584.11**[a,b]	
	0-4-15-2-10-14-6-22-18-0	**0-3-17-9-16-11-19-0**	**0-1-13-21-5-12-8-7-20-0**	**537.20**	**0**	**584.27**[a,b]	
	0-4-15-2-10-14-6-22-18-0	**0-9-17-3-16-11-19-0**	**0-1-13-21-5-12-8-7-20-0**	**542.34**	**60.31**	**571.75**[a,b]	

Note: (a) "[a]" meant that the marked result dominates the results of T-NSGA-II.
(b) "[b]" meant that the marked result dominates the results of W-BEG-NSGA-II.
(c) "[a,b]" meant that the marked result dominates the results of T-NSGA-II and W-BEG-NSGA-II simultaneously.

are shown in columns 5 and 6. What is more, column 7 determines the service time of the customers (for example, 10 am-3 pm means that the service time of the customer is needed from 10 am until 3 pm).

In order to demonstrate the performance of the proposed algorithm and research the effect of the duplicate individuals, test example 2 was also tested by T-NSGA-II, W-BEG-NSGA-II, and BEG-NSGA-II, respectively. Table 6 shows the optimization results of test example 2 using these three algorithms. From the optimization results, we can gain the following information:

(i) The algorithms of T-NSGA-II, W-BEG-NSGA-II, and BEG-NSGA-II could find 10, 12, and 10 optimal Pareto solutions with average computing time of 2.5, 1.6, and 0.9 minutes correspondingly during 10 run times.

(ii) All of the results (bold and marked as "[a]" in Table 6) gained by T-NSGA-II dominate the ones gained by W-BEG-NSGA-II.

(iii) The algorithm of BEG-NSGA-II could find 6 Pareto (bold and marked as "[a]" or "[b]" in Table 6) solutions that dominate the results of T-NSGA-II as well as W-BEG-NSGA-II.

From the information mentioned above, we can draw the following conclusions:

(i) The algorithms of T-NSGA-II and W-BEG-NSGA-II may generate amount of duplicate individuals, especially after 100 iterations. These duplicate individuals may cause the problems of inefficiency, slow convergence, and converging to local Pareto optimal solutions.

(ii) The proposed algorithm of BEG-NSGA-II is more effective than the algorithms of T-NSGA-II and W-BEG-NSGA-II to solve the VRPSPD problem with rigor constraints.

Figures 11 and 12 show two optimization Pareto solutions (0-4-15-2-10-6-22-18-14-0 0-1-17-16-3-11-19-0 0-9-13-21-5-12-8-7-20-0; 0-4-15-2-10-6-22-18-14-0 0-9-17-3-16-11-19-0 0-1-13-21-5-12-8-7-20-0) of BEG-NSGA-II. The specific path descriptions of vehicles 1, 2, and 3 can refer to the test example 1. Figure 13 shows the number of duplicate individuals which changes with the number of iterations in test example 2.

4.3. Test Example 3. Table 7 shows an example of VRPSPD problem with 42 customers. The first column shows the information of the customers (there are 1 recycling center, 30 distributers, 6 recyclers, and 6 suppliers). In the second column, in order to facilitate encoding and decoding in the algorithm, all customers were numbered as a series continuous integer (recycling center 0 is numbered 0; distributors 1, 2, ..., and 30 are numbered 1, 2, ..., and 30; recyclers 1, 2, ..., and 6 are numbered 31, 32, ..., and 36; suppliers 1, 2, ..., and 6 are numbered 37, 38, ..., and 42). The third and fourth columns show the X/km and Y/km coordinates of the customers, respectively. The amounts of demand/ton and recycle/ton of the customers are shown in columns 5 and 6. What is more, column 7 determines the service time of the customers (for example, 10 am-3 pm means that the service time of the customer is needed from 10 am until 3 pm).

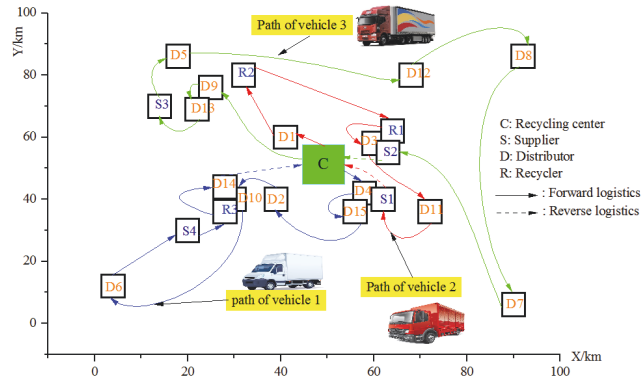


Figure 11: Solution 1 of test example 2 (fuel consumption=544.83kg, waiting time = 147.38 min, and distance = 569.64km).

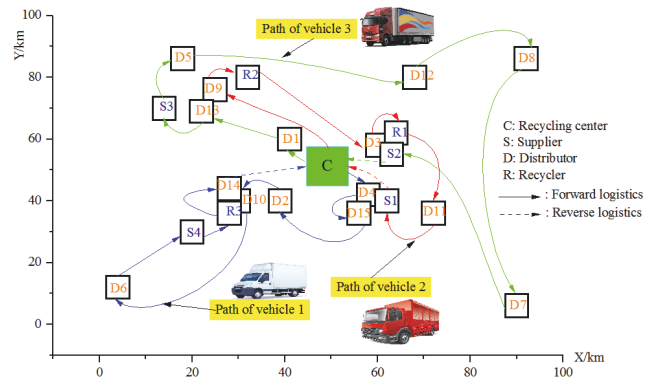


Figure 12: Solution 2 of test example 2 (fuel consumption=543.80kg, waiting time = 147.15 min, and distance = 570.21km).

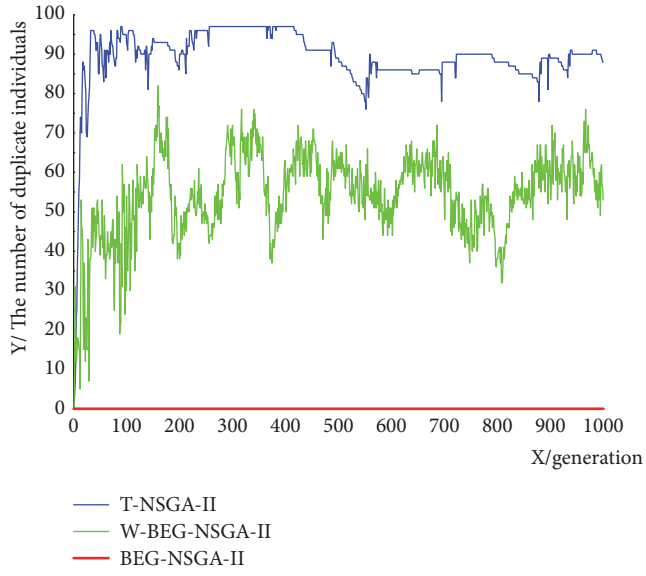


Figure 13: The number of duplicate individuals in test example 2 with 22 customers.

Table 7: The basic information of test example 3 with 42 customers.

	Number	X/km	Y/km	Demand/ton	Recycle/ton	Service time
Cen. 0	0	50	50	-	-	-
Dis. 1	1	41	60	2	1	8 am-10 am
Dis. 2	2	39	40	1.5	0.8	10 am-12 am
Dis. 3	3	60	58	3	1.2	8 am-10 am
Dis. 4	4	58	42	2.5	1	8.5 am-10.5 am
Dis. 5	5	18	86	2	0.5	9.5 am-11.5 am
Dis. 6	6	4	12	2	0.8	9 am-12 am
Dis. 7	7	90	6	1.5	0.6	10 am-3 pm
Dis. 8	8	92	86	3	1.5	10 am-3 pm
Dis. 9	9	25	76	2	1	10 am-3 pm
Dis. 10	10	30	40	2	1	10 am-3 pm
Dis. 11	11	72	36	2	0.7	10 am-3 pm
Dis. 12	12	68	80	2	1.2	10 am-3 pm
Dis. 13	13	22	69	2	1.8	10 am-12 am
Dis. 14	14	28	44	1.5	0.9	8.5 am-10.5 am
Dis. 15	15	56	36	2.5	1.1	9.5 am-11.5 am
Dis. 16	16	9	56	1	0.8	9 am-12 am
Dis. 17	17	8	40	1.7	0.8	10 am-3 pm
Dis. 18	18	85	12	1	0.7	10 am-3 pm
Dis. 19	19	88	84	0.9	0.6	10 am-3 pm
Dis. 20	20	40	80	1	0.8	10 am-3 pm
Dis. 21	21	45	13	1.5	1	10 am-3 pm
Dis. 22	22	70	25	1.1	1	10 am-3 pm
Dis. 23	23	69	77	1.2	0.6	10 am-12 am
Dis. 24	24	24	81	1	0.7	8.5 am-10.5 am
Dis. 25	25	50	30	1.2	0.6	9.5 am-11.5 am
Dis. 26	26	48	56	0.9	0.5	9 am-12 am
Dis. 27	27	22	60	1.4	0.6	10 am-3 pm
Dis. 28	28	6	50	0.5	0.2	10 am-3 pm
Dis. 29	29	48	10	0.6	0.4	10 am-3 pm
Dis. 30	30	92	50	0.8	0.4	10 am-3 pm
Rec.1	31	64	62	0	2.5	10 am-3 pm
Rec.2	32	32	80	0	2	10 am-3 pm
Rec.3	33	28	36	0	3	10 am-12 am
Rec.4	34	62	40	0	1.8	8.5 am-10.5 am
Rec.5	35	76	70	0	3.5	9.5 am-11.5 am
Rec.6	36	72	41	0	5	9 am-12 am
Sup.1	37	12	63	0	4	10 am-3 pm
Sup.2	38	21	19	0	4.2	10 am-3 pm
Sup.3	39	63	55	0	4	10 am-3 pm
Sup.4	40	14	70	0	3	10 am-3 pm
Sup.5	41	20	30	0	7	10 am-3 pm
Sup.6	42	80	53	0	4.9	10 am-3 pm

Note: (a) Cen., Dis., Rec., and Sup. are the abbreviations of recycling center, distributors, recycler, and supplier, respectively.
(b) "-" means that these data do not need to be set in advance.

In order to demonstrate the performance of the proposed algorithm and research the effect of the duplicate individuals, test example 3 was also tested by T-NSGA-II, W-BEG-NSGA-II, and BEG-NSGA-II, respectively. Table 8 shows the optimization results of test example 3 using these three algorithms. From the optimization results, we can gain the following information:

(i) The algorithms of T-NSGA-II, W-BEG-NSGA-II, and BEG-NSGA-II could find 8, 6, and 11 optimal Pareto solutions with average computing time of 3,

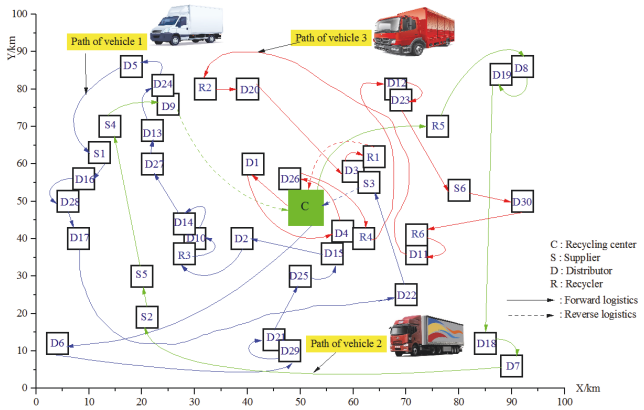


Figure 14: Solution 1 of test example 3 (fuel consumption=945.59kg, waiting time = 466.37min, and distance = 959.02km).

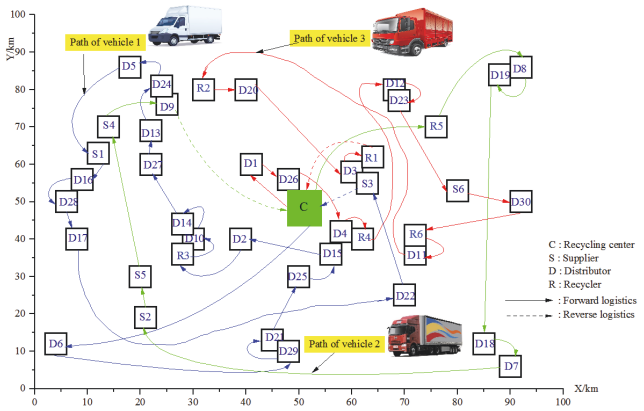


Figure 15: Solution 2 of test example 3 (fuel consumption=945.59kg, waiting time = 463.29 min, and distance = 992.51km).

2.6, and 2.1 minutes correspondingly during 10 run times.

(ii) The algorithm of BEG-NSGA-II could find 9 Pareto solutions (bold and marked as "a" in Table 8) that dominate the results of T-BEG-NSGA-II.

From the information mentioned above, we can draw the following conclusions:

(i) The algorithms of T-NSGA-II and W-BEG-NSGA-II may generate amount of duplicate individuals, especially after 100 iterations. These duplicate individuals may cause the problems of inefficiency, slow convergence, and converging to local Pareto optimal solutions.

(ii) The proposed algorithm of BEG-NSGA-II is more effective than the algorithms of T-NSGA-II and W-BEG-NSGA-II to solve the VRPSPD problem with rigor constraints.

Figures 14 and 15 show two optimization Pareto solutions (0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0, 0-35-8-19-18-7-38-41-40-9-0, 0-1-4-26-34-12-23-42-30-36-11-32-20-3-31-0; 0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0, 0-35-8-19-18-7-38-41-40-9-0, 0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0) of BEG-NSGA-II. The specific path descriptions of vehicles 1, 2, and 3 can refer

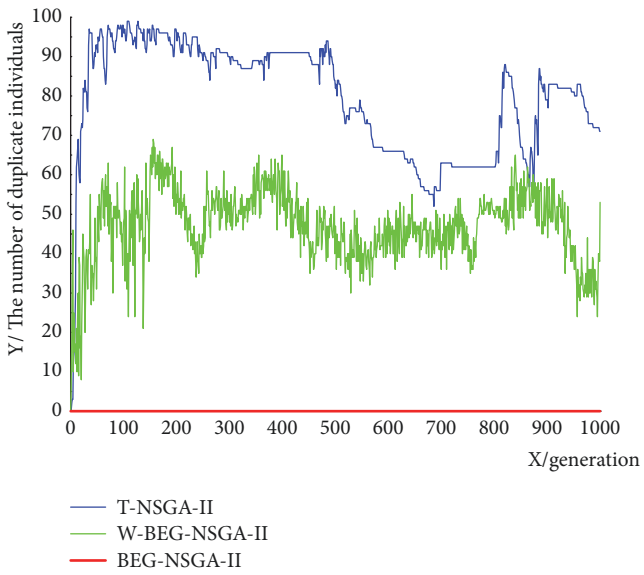


Figure 16: The number of duplicate individuals in test example 3 with 42 customers.

to test example 1. Figure 16 shows the number of duplicate individuals which changes with the number of iterations in test example 3.

4.4. Test Example 4. Dethloff's benchmark [40] which has been widely used in these years is adopted to illustrate the pervasiveness and superiority of the proposed BEG-NSGA-II. To the best of our knowledge, the best known upper bounds for Dethloff's benchmark [40] were found by the following algorithms:

PILS: parallel iterated local search [41]

VLBR: variable length bone route [42]

HPSO: hybrid particle swarm optimization [23]

ACS: ant colony system [43]

ACSEVNS: ant colony system empowered variable neighborhood search algorithm [44].

Therefore, we compare the performance of the proposed BEG-NSGA-II against these algorithms reported in the literature. Table 9 shows the optimization results of test example 4 using these algorithms and our algorithm. The optimization objective of PILS, VLBR, HPSO, and ACS is vehicle distance, by which the best known solution (BKS) for Dethloff's benchmark [40] is found. The optimization objective of ACSEVNS is also vehicle distance, which shows the best solutions (Best), average solution over 10 replications (Avg.), and average computation time (T). Because the waiting time of customers is not considered in Dethloff's benchmark [40], our proposed BEG-NSGA-II only optimizes the objectives of vehicle distance and fuel consuming (Fuel). From the optimization results, we can gain the following information:

(i) The algorithm of BEG-NSGA-II could find 9 BKSs in 40 instance.

Table 8: Comparisons of the optimization results of test example 3 with 42 customers.

Algorithms	Vehicle path	Fuel consumption/kg	Waiting time/min	Distance/km	Average computing time/min
T-NSGA-II	0-16-13-27-34-35-36-11-18-7-21-40-37-0 0-26-1-32-12-8-19-29-2-41-17-6-28-33-38-0 0-4-15-22-25-14-9-24-5-20-23-30-42-39-10-3-31-0	1035.36	1006.49	1120.63	3
	0-16-13-27-34-35-36-11-18-7-21-40-37-0 0-26-1-32-12-8-19-29-2-41-17-6-28-33-38-0 0-14-4-15-22-25-9-24-5-20-23-30-42-39-10-3-31-0	1089.13	943.67	1156.32	
	0-16-13-27-34-35-36-11-18-7-21-40-37-0 0-26-1-32-12-19-8-29-2-41-17-6-28-33-38-0 0-4-15-22-25-14-9-24-5-20-23-30-42-39-10-3-31-0	1037.93	1003.33	1119.99	
	0-16-13-27-34-35-36-11-18-7-21-40-37-0 0-26-1-32-12-19-8-29-2-41-17-6-28-33-38-0 0-14-4-15-22-25-9-24-5-20-23-30-42-39-10-3-31-0	1091.70	940.52	1155.68	
	0-16-13-27-34-35-36-11-18-7-21-40-37-0 0-26-1-32-12-19-8-29-2-41-17-6-28-33-38-0 0-14-4-15-25-22-9-24-5-20-23-30-42-39-10-3-31-0	1099.02	926.88	1162.02	
	0-16-13-34-35-36-11-18-7-21-27-40-37-0 0-26-1-32-12-8-19-29-2-41-17-6-28-33-38-0 0-4-15-22-25-14-9-24-5-20-23-30-42-39-10-3-31-0	1050.16	987.16	1116.56	
	0-16-13-34-35-36-11-18-7-21-27-40-37-0 0-26-1-32-12-19-8-29-2-41-17-6-28-33-38-0 0-14-4-15-22-25-9-24-5-20-23-30-42-39-10-3-31-0	1106.49	921.19	1151.61	
	0-27-40-13-37-16-34-36-11-18-7-21-35-0 0-26-1-32-12-8-19-29-2-41-17-6-28-33-38-0 0-4-15-22-25-14-9-24-5-20-23-30-42-39-10-3-31-0	983.25	1239.32	1065.36	
W-BEG-NSGA-II	0-26-39-11-22-25-2-27-16-28-17-6-41-10-31-19-32-0 0-1-15-14-5-24-40-36-30-29-21-38-37-0 0-4-3-8-35-23-12-20-13-9-34-18-7-33-42-0	1031.87	508.12	1185.27	2.6
	0-26-39-11-22-25-2-27-16-28-17-6-41-10-31-19-32-0 0-1-15-14-5-24-40-36-30-29-21-38-37-0 0-4-34-3-8-35-23-12-20-13-9-18-7-33-42-0	1040.27	369.66	1191.60	
	0-26-39-11-22-25-2-27-16-28-17-6-41-10-32-19-31-0 0-1-15-14-5-24-40-36-30-29-21-38-37-0 0-4-3-8-35-23-12-20-13-9-34-18-7-33-42-0	1022.91	525.40	1168.28	
	0-26-39-11-22-25-2-27-16-28-17-6-41-10-32-19-31-0 0-1-15-14-5-24-40-36-30-29-21-38-37-0 0-34-4-3-8-35-23-12-20-13-9-18-7-33-42-0	1034.94	386.94	1176.93	
	0-26-39-11-22-25-2-27-16-28-17-6-41-10-32-19-31-0 0-34-4-14-5-24-40-36-30-29-21-38-37-0 0-15-3-8-35-23-12-20-13-9-1-18-7-33-42-0	1031.05	614.06	1157.60	
	0-26-39-11-22-25-2-27-16-28-17-6-41-10-32-19-31-0 0-34-4-14-5-24-40-36-30-29-21-38-37-0 0-15-3-12-8-35-23-20-13-9-1-18-7-33-42-0	1030.05	628.25	1161.22	

Table 8: Continued.

Algorithms	Vehicle path	Fuel consumption/kg	Waiting time/min	Distance/km	Average computing time/min
BEG-NSGA-II	0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0 0-35-8-19-18-7-38-41-40-9-0 0-1-4-26-34-12-23-42-30-36-11-32-20-3-31-0	983.70	463.29	992.51	2.1
	0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0 **0-35-8-19-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**945.59**	**466.37**	**959.02[a]**	
	0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0 0-35-19-8-18-7-38-41-40-9-0 0-1-4-26-34-12-23-42-30-36-11-32-20-3-31-0	984.27	463.29	990.59	
	0-6-29-21-25-15-2-33-10-14-27-13-24-5-37-16-28-17-22-39-0 **0-35-19-8-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**946.15**	**466.37**	**957.10[a]**	
	0-6-29-21-25-22-15-2-33-10-14-27-13-24-5-37-16-28-17-39-0 **0-35-8-19-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**955.73**	**621.80**	**951.37[a]**	
	0-6-29-21-25-22-15-2-33-10-14-27-13-24-5-37-16-28-17-39-0 **0-35-19-8-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**956.30**	**621.80**	**949.45[a]**	
	0-25-29-21-22-15-2-33-10-14-27-13-24-5-37-16-28-17-6-39-0 **0-35-8-19-18-7-38-41-40-9-0** **0-1-4-26-34-12-23-42-30-36-11-32-20-3-31-0**	**917.76**	**709.52**	**954.92[a]**	
	0-25-29-21-22-15-2-33-10-14-27-13-24-5-37-16-28-17-6-39-0 **0-35-8-19-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**879.65**	**712.60**	**921.44[a]**	
	0-25-29-21-22-15-2-33-10-14-27-13-24-5-37-16-28-17-6-39-0 **0-35-19-8-18-7-38-41-40-9-0** **0-1-4-26-34-12-23-42-30-36-11-32-20-3-31-0**	**918.33**	**709.52**	**953.00[a]**	
	0-25-29-21-22-15-2-33-10-14-27-13-24-5-37-16-28-17-6-39-0 **0-35-19-8-18-7-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**880.21**	**712.60**	**919.52[a]**	
	0-25-29-21-22-15-2-33-10-14-27-13-24-5-37-16-28-17-7-39-0 **0-35-19-8-18-6-38-41-40-9-0** **0-1-26-4-34-12-23-42-30-36-11-32-20-3-31-0**	**929.96**	**680.24**	**984.30[a]**	

Note: (a) "[a]" means that the marked result dominates the results of T-NSGA-II.

Table 9: Computational results for Dethloff's benchmark.

Instance	Best so far know solution in the literature		ACSEVNS			BEG-NSGA-II			
	Ref.	BKS	Best	Avg.	T	Best	Avg.	T	**Fuel**
SCA3-0	PILS, VLBR, HPSO, ACS	**635.62**	**635.62**	635.62	4.77	732.12	745.36	10.21	684.23
SCA3-1	PILS, VLBR, HPSO, ACS	**697.84**	**697.84**	697.84	5.24	821.13	838.39	11.32	761.96
SCA3-2	PILS, VLBR, HPSO, ACS	**659.34**	**659.34**	659.34	7.47	**659.34**	659.34	12.65	612.62
SCA3-3	PILS, VLBR, HPSO, ACS	**680.04**	**680.04**	680.34	5.20	762.36	776.61	9.69	715.37
SCA3-4	PILS, VLBR, HPSO, ACS	**690.50**	**690.50**	690.50	4.96	765.58	770.67	13.52	726.27
SCA3-5	PILS, VLBR, HPSO, ACS	**659.91**	**659.91**	659.91	5.18	675.69	680.21	12.38	606.39
SCA3-6	PILS, VLBR, HPSO, ACS	**651.09**	**651.09**	651.11	4.68	**651.09**	651.09	14.32	610.92
SCA3-7	PILS, VLBR, HPSO, ACS	**659.17**	**659.17**	659.17	6.06	763.47	773.6	12.21	723.26
SCA3-8	PILS, VLBR, HPSO, ACS	**719.48**	**719.48**	719.56	4.51	768.36	782.65	10.36	716.28
SCA3-9	PILS, VLBR, HPSO, ACS	**681.00**	**681.00**	681.00	7.08	**681.00**	681.00	9.36	621.23
SCA8-0	PILS, VLBR, HPSO, ACS	**961.50**	**961.50**	961.50	5.33	1012.38	1026.54	15.63	973.62
SCA8-1	PILS, VLBR, HPSO, ACS	**1049.65**	**1049.65**	1049.65	5.62	1099.63	1129.78	16.22	1038.53
SCA8-2	PILS, VLBR, HPSO, ACS	**1039.64**	**1039.64**	1041.62	6.05	1126.53	1139.25	16.37	1079.92
SCA8-3	PILS, VLBR, HPSO, ACS	**983.34**	**983.34**	983.34	8.39	1016.29	1035.69	13.23	968.37
SCA8-4	PILS, VLBR, HPSO, ACS	**1065.49**	**1065.49**	1065.49	6.07	**1065.49**	1065.49	11.32	1016.27
SCA8-5	PILS, VLBR, HPSO, ACS	**1027.08**	**1027.08**	1027.14	6.96	1089.36	1096.75	10.63	1023.63
SCA8-6	PILS, VLBR, HPSO, ACS	**971.82**	**971.82**	971.82	7.76	985.78	996.34	9.89	928.56
SCA8-7	PILS, VLBR, HPSO, ACS	**1051.28**	**1051.28**	1051.28	8.14	1156.36	1173.95	12.35	1103.93
SCA8-8	PILS, VLBR, HPSO, ACS	**1071.18**	**1071.18**	1071.22	7.06	1210.35	1242.25	18.69	1129.46
SCA8-9	PILS, VLBR, HPSO, ACS	**1060.50**	**1060.50**	1060.50	5.29	1126.52	1153.34	19.99	1086.95
CON3-0	PILS, VLBR, HPSO, ACS	**616.52**	**616.52**	616.52	6.80	712.45	732.32	15.36	672.97
CON3-1	PILS, VLBR, HPSO, ACS	**554.47**	**554.47**	554.47	5.01	589.76	593.67	11.25	521.36
CON3-2	PILS, VLBR, HPSO, ACS	**518.00**	**518.00**	518.00	7.55	562.98	583.53	16.87	512.39
CON3-3	PILS, VLBR, HPSO, ACS	**591.19**	**591.19**	591.19	5.75	**591.19**	591.19	13.63	538.13
CON3-4	PILS, VLBR, HPSO, ACS	**588.79**	**588.79**	588.79	3.90	623.82	643.78	9.36	579.24
CON3-5	PILS, VLBR, HPSO, ACS	**563.70**	**563.70**	563.70	6.86	**563.70**	563.70	12.36	513.21
CON3-6	PILS, VLBR, HPSO, ACS	**499.05**	**499.05**	499.05	8.54	**499.05**	499.05	12.21	432.63
CON3-7	PILS, VLBR, HPSO, ACS	**576.48**	**576.48**	576.48	4.26	596.35	601.28	10.86	536.52
CON3-8	PILS, VLBR, HPSO, ACS	**523.05**	**523.05**	523.05	3.89	556.31	571.36	10.35	506.37
CON3-9	PILS, VLBR, HPSO, ACS	**578.25**	**578.25**	578.25	6.33	623.67	635.73	13.62	576.23
CON8-0	PILS, VLBR, HPSO, ACS	**857.17**	**857.17**	857.17	5.40	926.46	942.39	15.36	876.56
CON8-1	PILS, VLBR, HPSO, ACS	**740.85**	**740.85**	740.85	8.46	**740.85**	740.85	15.78	695.35
CON8-2	PILS, VLBR, HPSO, ACS	**712.89**	**712.89**	712.89	4.79	763.56	781.22	10.36	713.69
CON8-3	PILS, VLBR, HPSO, ACS	**811.07**	**811.07**	811.07	7.21	862.35	873.68	14.36	812.38
CON8-4	PILS, VLBR, HPSO, ACS	**772.25**	**772.25**	772.25	6.70	812.65	832.47	15.69	756.24
CON8-5	PILS, VLBR, HPSO, ACS	**754.88**	**754.88**	754.88	5.74	**754.88**	754.88	12.63	703.63
CON8-6	PILS, VLBR, HPSO, ACS	**678.92**	**678.92**	678.92	4.36	723.52	736.61	12.53	676.29
CON8-7	PILS, VLBR, HPSO, ACS	**811.96**	**811.96**	813.46	8.38	866.76	873.69	20.32	813.53
CON8-8	PILS, VLBR, HPSO, ACS	**767.53**	**767.53**	767.53	6.16	813.65	826.63	12.36	763.96
CON8-9	PILS, VLBR, HPSO, ACS	**809.00**	**809.00**	809.00	7.19	901.23	923.57	14.23	863.24
G. Avg.					6.13			13.25	
BKS found		40	40			9			

Note: the best solution of each problem instance is highlighted in bold.
BKS: the best known solution. **Ref.:** the best solution reference. **Best**: the best solution in 10 replications.
Avg.: average solution over 10 replications. **T:** average computation time. **G. Avg.:** average of 40 instances.
BKS found: number of best solutions found.

(ii) The objective of vehicle distance and computation time obtained from algorithm of BEG-NSGA-II is little worse than the ones obtained from ACSEVNS.

(iii) In addition to the objective of vehicle distance and computation time, the optimization objective of fuel consumption that is not considered in the compared algorithm is also obtained in the proposed algorithm of BEG-NSGA-II.

From the information mentioned above, we can draw the following conclusions:

(i) The proposed algorithm of BEG-NSGA-II can obtain the BKS or near BKS in reasonable computation time considering the objectives of vehicle distance and fuel consumption simultaneously.

(ii) The reason why some results gained from proposed BEG-NSGA-II are little worse than the compared algorithms is that we consider the optimization objectives of vehicle distance and computation simultaneously, while the compared ones only consider the single objective of vehicle distance.

(iii) From the simulations of examples 1-4, we can see that our proposed algorithm of BEG-NSGA-II has high performance in multiobjective optimization as well as meeting the single objective optimization.

5. Conclusions

In this paper, we construct a new closed-loop supply chain logistics network of VRPSPD dominated by remanufacturer, in which the customers are originally divided into three types: distributors, recyclers, and suppliers. The proposed model is more consistent with the actual situation of recycling and remanufacturing. For the purpose of resource saving and environment protection, we originally add the fuel consumption as an optimization objective into VRPSPD. In order to solve the proposed model and conquer the disadvantages of the traditional NSGA-II and the existing algorithms with two-stage optimization mechanism, a bee evolutionary algorithm guiding nondominated sorting genetic algorithm II (BEG-NSGA-II) is proposed with the optimization objectives of minimum fuel consumption, minimum waiting time, and the shortest delivery distance while meeting the travel constraints and load constraints at the same time. To verify the effectiveness of the proposed algorithm, test examples are used to simulate the proposed VRPSPD and the traditional VRPSPD, respectively. From the simulation results of test examples 1-3, we can see that our proposed algorithm BEG-NSGA-II is superior to the traditional NSGA-II (T-NSGA-II) and the proposed BEG-NSGA-II without eliminating the duplicate individuals (W-BEG-NSGA-II). From the simulation results of test example 4, we can see that the proposed algorithm of BEG-NSGA-II can obtain the best known solution (BKS) or near BKS in reasonable computation time considering the objectives of vehicle distance and fuel consumption simultaneously. That is to say, our proposed algorithm of BEG-NSGA-II works better in multiobjective optimization as well as meeting the single objective optimization. It would be prosperous to apply our algorithm to other multiobjective optimization problems, especially in the logistics distribution field.

Conflicts of Interest

The authors confirm that this work does not have any conflicts of interest with other work.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant no. 71473077), the National Science-Technology Support Plan Projects of China (2015BAF01B00), the National High Technology Research and Development Program of China (863 Program) (Grant no. 2013AA040206), and Project of State Key Laboratory of Advanced Design and Manufacturing for Vehicle Body, Hunan University (Grant no. 71775004).

References

[1] J. Östlin, E. Sundin, and M. Björkman, "Product life-cycle implications for remanufacturing strategies," *Journal of Cleaner Production*, vol. 17, no. 11, pp. 999–1009, 2009.

[2] T. Zhang, J. Chu, X. Wang, X. Liu, and P. Cui, "Development pattern and enhancing system of automotive components remanufacturing industry in China," *Resources, Conservation & Recycling*, vol. 55, no. 6, pp. 613–622, 2011.

[3] D. L. Diener and A.-M. Tillman, "Component end-of-life management: Exploring opportunities and related benefits of remanufacturing and functional recycling," *Resources, Conservation & Recycling*, vol. 102, pp. 80–93, 2015.

[4] X. Bin-Shi, "State of the art and future development in remanufacturing engineering," *Transactions of Materials & Heat Treatment*, vol. 31, no. 1, pp. 10–14, 2010.

[5] X. Chang, Y. Zhong, Y. Wang, and Z. Chen, "Research of low-carbon policy to promote automotive parts remanufacturing in china: a case study of auto engine remanufacturing," *Systems Engineering -Theory & Practice*, vol. 33, no. 11, pp. 2811–2821, 2013.

[6] L. Wang and M. Chen, "Policies and perspective on end-of-life vehicles in China," *Journal of Cleaner Production*, vol. 44, pp. 168–176, 2013.

[7] I. Konstantaras, "Optimal control of production and remanufacturing in a reverse logistics model with backlogging," *Mathematical Problems in Engineering*, vol. 2010, Article ID 320913, 19 pages, 2010.

[8] M. Zarei, S. Mansour, A. Husseinzadeh Kashan, and B. Karimi, "Designing a Reverse Logistics Network for End-of-Life Vehicles Recovery," *Mathematical Problems in Engineering*, vol. 2010, Article ID 649028, 16 pages, 2010.

[9] S. Li, N. Wang, Z. He, A. Che, and Y. Ma, "Design of a multiobjective reverse logistics network considering the cost and service level," *Mathematical Problems in Engineering*, vol. 2012, Article ID 928620, 21 pages, 2012.

[10] F. H. Wang and D. A. Qing-Li, "The multiple-vehicle transportation policy in closed-loop chain with split pick-ups," *Systems Engineering -Theory & Practic*, vol. 27, no. 2, pp. 105–111, 2007.

[11] Y. Ding, X. Li, and Y. Gao, "Multi-period and multi-objective dynamic location model for remanufacturing logistics network," *Chinese Journal of Management*, vol. 11, no. 3, pp. 428–433, 2014.

[12] H. Min, "The multiple vehicle routing problem with simultaneous delivery and pick-up points," *Transportation Research Part A: General*, vol. 23, no. 5, pp. 377–386, 1989.

[13] A. Klose, M. G. Speranza, and L. V. Wassenhove, *Quantitative Approaches to Distribution Logistics and Supply Chain Management*, Springer, Berlin, Germany, 2002.

[14] H. Li and A. Lim, "A Metaheuristic for the Pickup and Delivery Problem with Time Windows," *International Journal on Artificial Intelligence Tools*, vol. 12, no. 02, pp. 173–186, 2003.

[15] J. Crispim and J. Brandão, "Metaheuristics applied to mixed and simultaneous extensions of vehicle routing problems with backhauls," *Journal of the Operational Research Society*, vol. 56, no. 11, pp. 1296–1302, 2005.

[16] S. Ropke and D. Pisinger, "An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows," *Transportation Science*, vol. 40, no. 4, pp. 455–472, 2006.

[17] F. A. T. Montané and R. D. Galvão, "A tabu search algorithm for the vehicle routing problem with simultaneous pick-up and delivery service," *Computers & Operations Research*, vol. 33, no. 3, pp. 595–619, 2006.

[18] T. J. Ai and V. Kachitvichyanukul, "A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery," *Computers & Operations Research*, vol. 36, no. 5, pp. 1693–1702, 2009.

[19] B. Çatay, "A new saving-based ant algorithm for the vehicle routing problem with simultaneous pickup and delivery," *Expert Systems with Applications*, vol. 37, no. 10, pp. 6809–6817, 2010.

[20] L. Mingyong and C. Erbao, "An improved differential evolution algorithm for vehicle routing problem with simultaneous pick-ups and deliveries and time windows," *Engineering Applications of Artificial Intelligence*, vol. 23, no. 2, pp. 188–195, 2010.

[21] B. Wu, C. H. Qian, M. Dong, and Q. H. Xie, "Chaos quantum evolutionary algorithm for vehicle routing problem with simultaneous delivery and pickup," *Control and Decision*, vol. 25, no. 3, pp. 5097–5101, 2010.

[22] A. Subramanian, E. Uchoa, A. A. Pessoa, and L. S. Ochi, "Branch-cut-and-price for the vehicle routing problem with simultaneous pickup and delivery," *Optimization Letters*, vol. 7, no. 7, pp. 1569–1581, 2013.

[23] F. P. Goksal, I. Karaoglan, and F. Altiparmak, "A hybrid discrete particle swarm optimization for vehicle routing problem with simultaneous pickup and delivery," *Computers & Industrial Engineering*, vol. 65, no. 1, pp. 39–53, 2013.

[24] T. Zhang, WX. Tian, YJ. Zhang, and SX. Liu, "The improved ant colony algorithm for the VRPSPD with maximum distance constraint," *Systems Engineering-Theory Practice*, vol. 28, no. 1, pp. 132-130, 2008.

[25] T. Zhang, W. A. Chaovalitwongse, and Y. Zhang, "Integrated ant colony and tabu search approach for time dependent vehicle routing problems with simultaneous pickup and delivery," *Journal of Combinatorial Optimization*, vol. 28, no. 1, pp. 288–309, 2014.

[26] O. Polat, C. B. Kalayci, O. Kulak, and H.-O. Günther, "A perturbation based variable neighborhood search heuristic for solving the vehicle routing problem with simultaneous pickup and delivery with time limit," *European Journal of Operational Research*, vol. 242, no. 2, pp. 369–382, 2015.

[27] M. Avci and S. Topaloglu, "A hybrid metaheuristic algorithm for heterogeneous vehicle routing problem with simultaneous pickup and delivery," *Expert Systems with Applications*, vol. 53, pp. 160–171, 2016.

[28] L. I. Jin and F. U. Pei-Hua, "Heterogeneous fixed fleet low-carbon routing problem and algorithm," *Computer Integrated Manufacturing Systems*, vol. 19, no. 6, pp. 1351–1362, 2013.

[29] J. Li and P.-H. Fu, "Model and simulation for vehicle routing problem with time windows based on energy consumption," *Journal of System Simulation*, vol. 25, no. 6, pp. 1147–1154, 2013.

[30] L. Wu, X. P. Hu, and W. Z. Rao, "capacity-vehicle-routing-problem model and algorithm for reducing fuel consumption," *Transaction of Systems Engineering*, vol. 28, no. 6, pp. 804–811, 2013.

[31] W.-Z. Rao, C. Jin, X.-H. Wang, and F. Liu, "A model of low-carbon vehicle routing problem considering road gradient and its solving strategy," *System Engineering Theory and Practice*, vol. 34, no. 8, pp. 2092–2105, 2014.

[32] B. Ombuki, B. J. Ross, and F. Hanshar, "Multi-objective genetic algorithms for vehicle routing problem with time windows," *Applied Intelligence*, vol. 24, no. 1, pp. 17–30, 2006.

[33] H. C. W. Lau, T. M. Chan, W. T. Tsui, F. T. S. Chan, G. T. S. Ho, and K. L. Choy, "A fuzzy guided multi-objective evolutionary algorithm model for solving transportation problem," *Expert Systems with Applications*, vol. 36, no. 4, pp. 8255–8268, 2009.

[34] R. Baños, J. Ortega, C. Gil, A. L. Márquez, and F. De Toro, "A hybrid meta-heuristic for multi-objective vehicle routing problems with time windows," *Computers & Industrial Engineering*, vol. 65, no. 2, pp. 286–296, 2013.

[35] C. F. Hsueh, "A vehicle routing problem with consideration of green transportation," *Journal of Management & Sustainability*, vol. 7, no. 4, pp. 89–97, 2017.

[36] J. Jemai, M. Zekri, and K. Mellouli, "Multi Objective Genetic Algorithms for the Green Vehicle Routing Problem: a Comparative Study," *Journal of Theoretical and Applied Information Technology*, vol. 95, no. 23, pp. 6597–6607, 2017.

[37] H. Soleimani, Y. Chaharlang, and H. Ghaderi, "Collection and distribution of returned-remanufactured products in a vehicle routing problem with pickup and delivery considering sustainable and green criteria," *Journal of Cleaner Production*, vol. 172, pp. 960–970, 2018.

[38] Q. Deng, G. Gong, X. Gong, L. Zhang, W. Liu, and Q. Ren, "A bee evolutionary guiding nondominated sorting genetic algorithm ii for multiobjective flexible job-shop scheduling," *Computational intelligence and neuroscience*, vol. 2017, Article ID 5232518, p. 20, 2017.

[39] M. Barth and K. Boriboonsomsin, "Energy and emissions impacts of a freeway-based dynamic eco-driving system," *Transportation Research Part D: Transport and Environment*, vol. 14, no. 6, pp. 400–410, 2009.

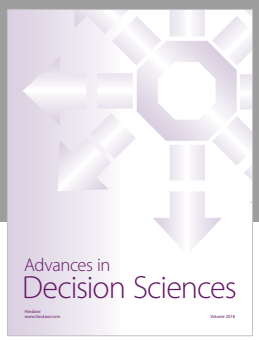
[40] J. Dethloff, "Vehicle routing and reverse logistics: the vehicle routing problem with simultaneous delivery and pick-up," *OR Spectrum*, vol. 23, no. 1, pp. 79–96, 2001.

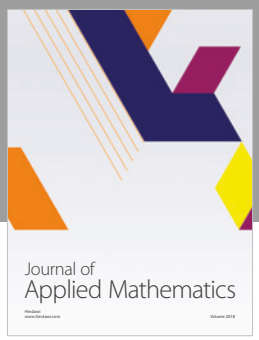
[41] A. Subramanian, L. M. A. Drummond, C. Bentes, L. S. Ochi, and R. Farias, "A parallel heuristic for the vehicle routing problem with simultaneous pickup and delivery," *Computers & Operations Research*, vol. 37, no. 11, pp. 1899–1911, 2010.

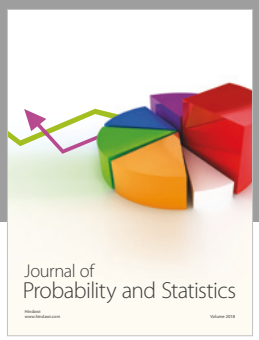
[42] E. E. Zachariadis, C. D. Tarantilis, and C. T. Kiranoudis, "An adaptive memory methodology for the vehicle routing problem with simultaneous pick-ups and deliveries," *European Journal of Operational Research*, vol. 202, no. 2, pp. 401–411, 2010.

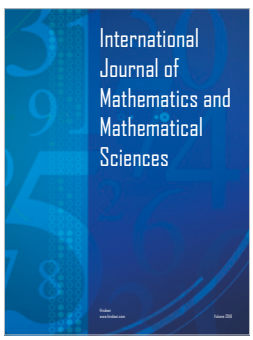
[43] Y. Gajpal and P. Abad, "An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup," *Computers & Operations Research*, vol. 36, no. 12, pp. 3215–3223, 2009.

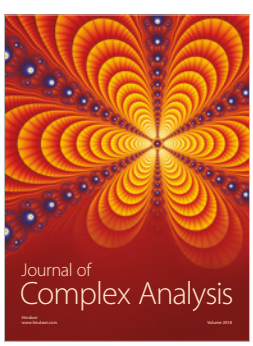
[44] C. B. Kalayci and C. Kaya, "An ant colony system empowered variable neighborhood search algorithm for the vehicle routing problem with simultaneous pickup and delivery," *Expert Systems with Applications*, vol. 66, pp. 163–175, 2016.

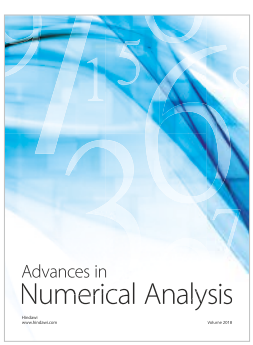


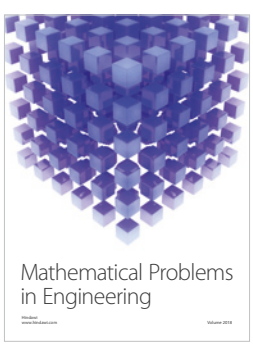


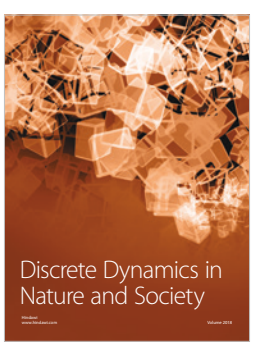


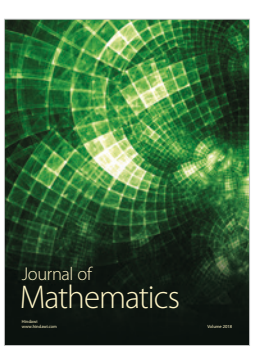


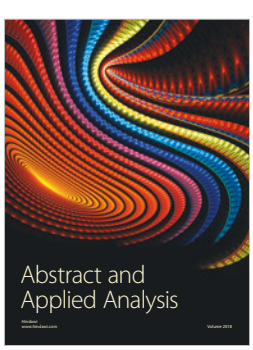


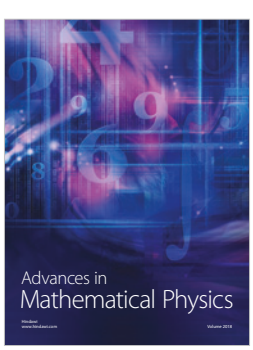












Hindawi

Submit your manuscripts at
www.hindawi.com