COMPLEX SYSTEM MODELING AND SIMULATION
ISSN 2096-9929 03/06 pp 202−219
Volume 3, Number 3, September 2023
DOI: 10.23919/CSMS.2023.0008

Energy-Efficient Multi-Trip Routing for Municipal Solid Waste Collection by Contribution-Based Adaptive Particle Swarm Optimization

Xiaoning Shen*, Hongli Pan, Zhongpei Ge, Wenyan Chen, Liyan Song, and Shuo Wang

Abstract: Waste collection is an important part of waste management system. Transportation costs and carbon emissions can be greatly reduced by proper vehicle routing. Meanwhile, each vehicle can work again after achieving its capacity limit and unloading the waste. For this, an energy-efficient multi-trip vehicle routing model is established for municipal solid waste collection, which incorporates practical factors like the limited capacity, maximum working hours, and multiple trips of each vehicle. Considering both economy and environment, fixed costs, fuel costs, and carbon emission costs are minimized together. To solve the formulated model effectively, contribution-based adaptive particle swarm optimization is proposed. Four strategies named greedy learning, multi-operator learning, exploring learning, and exploiting learning are specifically designed with their own searching priorities. By assessing the contribution of each learning strategy during the process of evolution, an appropriate one is selected and assigned to each individual adaptively to improve the searching efficiency of the algorithm. Moreover, an improved local search operator is performed on the trips with the largest number of waste sites so that both the exploiting ability and the convergence accuracy of the algorithm are improved. Performance of the proposed algorithm is tested on ten waste collection instances, which include one real-world case derived from the Green Ring Company of Jiangbei New District, Nanjing, China, and nine synthetic instances with increasing scales generated from the commonly-used capacitated vehicle routing problem benchmark datasets. Comparisons with five state-of-the-art algorithms show that the proposed algorithm can obtain a solution with a higher accuracy for the constructed model.

Key words: municipal solid waste collection; energy conservation; multi-trip; contribution; particle swarm optimization

1 Introduction

With the growth of global population and acceleration of urbanization, the population gradually migrated to the city, and living standards and environment have been greatly changed, which lead to more and more household waste. For instance, according to the estimates released by the Waste Engineering Institute

• Xiaoning Shen is with the Collaborative Innovation Center of Atmospheric Environment and Equipment Technology, and also with the Jiangsu Key Laboratory of Big Data Analysis Technology, School of Automation, Nanjing University of Information Science and Technology, Nanjing 210044, China. E-mail: sxnystsyt@sina.com.
• Hongli Pan, Zhongpei Ge, and Wenyan Chen are with the School of Automation, Nanjing University of Information Science and Technology, Nanjing 210044, China. E-mail: 1546498146@qq.com; 965159334@qq.com; 2978133743@qq.com.
• Liyan Song is with the Guangdong Provincial Key Laboratory of Brain-Inspired Intelligent Computation, the Department of Computer Science and Engineering, Southern University of Science and Technology, Shenzhen 518055, China. E-mail: songly@sustech.edu.cn.
• Shuo Wang is with the School of Computer Science, University of Birmingham, Birmingham, B15 2TT, UK. E-mail: s.wang.2@bham.ac.uk.
∗ To whom correspondence should be addressed.
※ This article was recommended by Associate Editor Rui Wang.
Manuscript received: 2023-02-02; revised: 2023-03-22; accepted: 2023-04-17

of Japan in 2020, the amount of waste produced in the world will reach 32 billion t a year by 2050, 4.2 times the amount in 2000. The accumulation of waste results in the rapid deterioration of the global environment, and carbon emissions are also on the rise. It is reported by International Energy Agency (IEA) that the increase of global carbon emissions in 2019 was about 33 billion t. Facing such a huge challenge, the world attaches great importance to it, and many local governments begin to implement Integrated Solid Waste Management (ISWM)[1]. ISWM includes generation, source-separation, storage, collection, processing, recovery, and disposal[2]. Among all steps of ISWM, waste collection is an important issue[3], whose typical process involves vehicles starting from the depot and traveling in certain routes to collect waste by visiting all waste sites. Waste collection costs a large amount of budget. For example, the total annual waste management costs in the United States are about 20 billion US dollar, among which waste collection costs have exceeded 10 billion US dollar[4]. The main reason for the high costs is that the vehicles have not been reasonably scheduled. In real life, once a vehicle is used, the corresponding driver salaries and maintenance costs are generated. To make full use of the vehicle and driver resources, each vehicle should make multiple trips to collect waste within the working hours of its driver. In this way, the number of required vehicles is also reduced. To our knowledge, multi-trip vehicle routing for waste collection has not been reported in the existing studies. Moreover, road traffic is one of the biggest contributors to large amounts of daily greenhouse gas emissions. In particular, heavy vehicles such as buses or garbage trucks account for a small proportion of the vehicle population but a large proportion of total tailpipe emissions[5]. To sum up, it is necessary to study the energy-efficient multi-trip vehicle routing problem for municipal solid waste collection to reduce both transportation costs and environmental pollution.

The main contributions of this work are summarized as follows. (1) We establish a constrained optimization model for the Energy-efficient Multi-trip Vehicle Routing Problem of Municipal Solid Waste Collection (EMVRPMSWC), which is to minimize both the transportation costs and the carbon emissions within the limited vehicle capacity and working hours. The distinctive characteristic of the model is to allow each vehicle to have multiple trips among the waste sites, the depot, and the disposal station. Additionally, the carbon emission pollution caused by driving vehicles is considered. (2) We propose Contribution-based Adaptive Particle Swarm Optimization (CAPSO) for solving the model. Frist, a novel decoding method is designed according to the features of the model, which follows the principle of maximizing the vehicle utilization and guaranteeing the full load of each vehicle. Second, four different learning strategies with specific search tasks, namely, rapid convergence, diversity enhancement, exploration of new regions, and exploitation of the optimum, are developed. On the basis of their contribution changing with the progress of the evolution, the most suitable learning strategy can be adaptively assigned to each individual at different stages. Third, an improved local search operator is performed on the trips that contain the most waste sites, which reduces the risk of rapid population assimilation caused by excessive mining.

2 Related Work

Since the notion of municipal solid waste collection was first introduced by Beltrami and Bodin[6], many researchers have studied on the modeling and solving of this problem, and achieved promising results[7].

2.1 Models of municipal solid waste collection

In recent work on modeling the municipal solid waste collection problem, more and more attentions are paid to the environmental protection, in addition to the traditional objectives like transportation costs, distance, or time. Wu et al.[8] considered uncertainty of waste generation rate and constructed a chance-constrained model for the low-carbon Vehicle Routing Problem (VRP) in waste management system. Molina et al.[9] took eco-efficiency as a performance indicator to design a waste collection routing model for a single landfill to reduce carbon emissions. Wu et al.[10] constructed a model for the green VRP in a waste management system, considering the greenhouse gas emission costs. In terms of the practicability, various realistic factors have been introduced into the waste collection model. Markov et al.[11] constructed a recyclable waste collection model, which extended the VRP with intermediate facilities by considering heterogeneous fleet and flexible multi-destination allocation. Shi et al.[12] modeled waste collection as a multi-depot VRP considering multiple waste depots in cities. Tirkolaee et al.[13] proposed a waste collection model in consideration of robust capacity constraints and uncertain amount of waste. Although the existing work on the municipal solid waste collection have modeled the realistic factors to some extent, they are all based on the VRP with a single trip. Actually, driver salaries and maintenance costs would be required once a vehicle is used in real life. With the aim of saving costs and improving resource utilization, one vehicle should go out and collect wastes multiple times as long as the maximum working hours of its driver are not exceeded.

In this study, an energy-efficient multi-trip vehicle routing model is constructed, which differs from the existing work in that: (1) More practical factors are considered, including multiple trips of each vehicle and the maximum working hours of the drivers. And (2) carbon dioxide (CO_2) emissions are transformed into costs and optimized with fixed costs and fuel costs together so that the environmental pollution caused by moving vehicles is reduced.

2.2 Methods for municipal solid waste collection

Since waste collection is an NP-hard problem[14], exact algorithms cannot get the optimal solution within polynomial time. Therefore, more and more heuristic and meta-heuristic algorithms are applied to this problem. Heuristic algorithms construct feasible solutions quickly based on intuitive experiences, but the quality of the solutions cannot be guaranteed. Louati et al.[15] proposed a heuristic smart routing algorithm for municipal solid waste collection. Shi et al.[12] solved the multi-depot vehicle routing problem in the waste collection system using a sector combination optimization algorithm. Meta-heuristic algorithms make an improvement on heuristic algorithms, which combine random algorithms and local search methods, and exhibit better effectiveness and applicability in finding solutions to the complex problems. Wei et al.[16] studied a waste collection problem considering the midway disposal pattern and proposed an artificial bee colony algorithm to handle this problem. Expósito-Márquez et al.[17] applied a greedy randomized adaptive search algorithm to the recyclable waste collection model. Qiao et al.[18] used a two-stage algorithm, which combined particle swarm optimization and tabu search to solve the Capacitated Vehicle Routing Problem (CVRP) in the waste collection system. Hannan et al.[19] determined the optimal routes of the CVRP in waste collection by a modified particle swarm optimization. Although some meta-heuristic methods have been adopted for solving the municipal solid waste collection, some problems still exist to be handled, e.g., insufficient use of the problem-specific information and a lack of variety when generating the new individuals.

Evolutionary algorithms with adaptive learning and variable neighborhood search are often used to solve vehicle routing problems. Islam et al.[20] proposed a hybrid meta-heuristic algorithm combining particle swarm optimization and variable neighborhood search to solve the clustered vehicle routing problem. Zheng et al.[21] proposed an evolutionary algorithm using variable local search based mutation to balance global exploration and local exploitation for vehicle routing of shared e-bicycle battery replacement and recycling. Wang et al.[22] constructed a dynamic multi-objective evolutionary algorithm based on ensemble learning for dynamic vehicle routing problem with time window. Shen et al.[23] adopted a region enhanced discrete multi-objective fireworks algorithm for low-carbon vehicle routing problem. Zheng et al.[24] adopted an evolutionary algorithm to route vehicles to distribute vaccines from the depots to the inoculation spots after obtaining the actual demands. Shen et al.[25] developed an adaptive neighborhood search heuristic to solve the electric vehicle routing problem with time windows. All these works neglect different search requirements at different evolutionary stages, and the variable neighborhood search might cause assimilation of the population.

To cover shortages of the existing methods, contribution-based adaptive particle swarm optimization is proposed in this paper. This algorithm is different from the existing ones in that: (1) A novel decoding method is designed with the idea of maximizing the vehicle utilization within the working hours and guaranteeing the full load of each vehicle. Meanwhile, it can ensure the feasibility of the decoded solutions. (2) A contribution-based adaptive learning strategy is presented to improve the search efficiency. And (3) an improved local search operator is given to further increase the solution accuracy.

3 Problem Descriptions and Formulations

This section gives a description of the considered problem EMVRPMSWC and constructs a mathematical model for it.

3.1 Problem descriptions

EMVRPMSWC can be defined as a directed network

G=(V, E), where V is the set of points including the depot, waste sites, and the disposal station, and E={(i, j)| $i, j \in V$} is an edge set.

To minimize objectives concerning both transportation costs and environmental pollution, EMVRPMSWC involves determining the minimal number of vehicles and finding the most proper routes for each selected vehicle, while satisfying the constraints of drivers' working hours and the vehicle capacity. Figure 1 shows an example of EMVRPMSWC. Each vehicle may have more than one trip in a daily work, and the number of trips depends on the capacity constraint and the maximum available time of the vehicle. Before finishing a day's work, the vehicle will drive to the disposal station to unload the waste and then return to the depot.

3.2 Mathematical model

The transportation costs for collecting municipal solid wastes mainly include fixed costs and fuel costs. In addition, carbon emission costs are also evaluated by a carbon tax. The parameters and variables in the EMVRPMSWC model are shown in Table 1.

(1) Fixed costs

In the process of waste collection, each vehicle used will incur the corresponding driver salaries and maintenance costs. In this paper, the fixed costs C_{fixed} can be expressed as Eq. (1).

$$C_{\text{fixed}} = C_{\text{f}} \cdot \sum_{k \in K} U_k \tag{1}$$

(2) Fuel costs

Fuel costs are incurred by fuel consumption during driving, and the fuel consumption efficiency is affected by the driving speed, road conditions, vehicle loads,

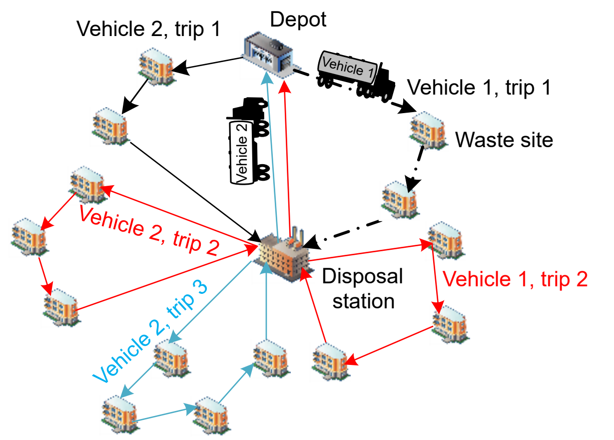


Fig. 1 An example of the considered problem EMVRPMSWC.

Table 1 Notations of the EMVRPMSWC model.

Variable	Description
i,j	Index of the waste site
k	Index of the vehicle
b	Index of the trip
K	Set of available vehicles
B_k	Set of all trips for vehicle k
N	N={1, 2, …, n}, 1: depot; 2: disposal station; 3−n: waste sites
Q	Capacity of each vehicle
z	Weight of the vehicle itself
q_i	Weight of wastes at waste site i
v	Average speed of each vehicle
$T_{\max}$	The maximum working hours of a driver
d_{ij}	Travel distance between waste site i and j
t_{ij}	Travel time between waste site i and j
l_{ij}	Load of the vehicle from waste site i to j
C_e	Carbon tax
C_m	Unit fuel costs of a vehicle
C_{f}	Fixed costs of a vehicle
S_{ij}	Carbon emission of the vehicle from waste site i to j
y_{ik}^b	$y_{ik}^b = \begin{cases} 1, & \text{if vehicle } k \text{ serves waste site } i \text{ in the trip } b; \\ 0, & \text{else} \end{cases}$
x_{ijk}^b	$x_{ijk}^b = \begin{cases} 1, & \text{if vehicle } k \text{ passes through the route}(i, j) \text{ in the trip } b; \\ 0, & \text{else} \end{cases}$
U_k	$U_k = \begin{cases} 1, & \text{if vehicle } k \text{ is used;} \\ 0, & \text{else} \end{cases}$

and other factors. Similar to Qiao et al.[18], we assume that the unit fuel costs C_m of the vehicle is fixed. Therefore, the fuel costs (C_{fuel}) generated by all vehicles after completing all the trips are shown in Eq. (2).

$$C_{\text{fuel}} = C_m \cdot \sum_{k \in K} \sum_{b \in B_k} \sum_{i \in N} \sum_{j \in N \& j \neq i} d_{ij} \cdot x_{ijk}^b \tag{2}$$

(3) Carbon emission costs

Because of the aggravation of the greenhouse effect, many countries have formulated a carbon tax policy to control the CO_2 emission. The carbon tax charges on the amount of CO_2 emitted. Therefore, the carbon emission costs (C_{carbon}) produced by all vehicles after completing all the trips are shown in Eqs. (3) and (4).

$$S_{ij} = F \cdot H_{ij} \tag{3}$$

$$C_{\text{carbon}} = C_e \cdot \sum_{k \in K} \sum_{b \in B_k} \sum_{i \in N} \sum_{j \in N \& j \neq i} S_{ij} \cdot x_{ijk}^b \tag{4}$$

where F is the fuel emission parameter, and H_{ij} is the fuel consumption of the vehicle traveling from waste site i to waste site j, which were referred to Bektas and Laporte[26].

In order to minimize the transportation costs and carbon emissions generated by vehicles in the process of municipal solid waste collection, the mathematical model of EMVRPMSWC is established as follows:

$$\min C_{\text{total}} = C_{\text{fixed}} + C_{\text{fuel}} + C_{\text{carbon}} \tag{5}$$

$$\text{s.t.} \sum_{k\in K}\sum_{b\in B_k}\sum_{j\in N\backslash\{1\}} x_{ijk}^b = \sum_{k\in K} U_k,\ i=1 \tag{6}$$

$$\sum_{b\in B_k}\sum_{j\in N\backslash\{1\}} x_{ijk}^b = 1,\ i=1, \forall k\in K \& U_k = 1 \tag{7}$$

$$\sum_{k\in K}\sum_{b\in B_k} y_{ik}^b = 1,\ \forall i\in N\backslash\{1,2\} \tag{8}$$

$$\sum_{i\in N}\sum_{j\in N\backslash\{1,2\}} x_{ijk}^b = y_{jk}^b,\ \forall b\in B_k, \forall k\in K \tag{9}$$

$$\sum_{i\in N\backslash\{1,2\}}\sum_{j\in N} x_{ijk}^b = y_{jk}^b,\ \forall b\in B_k,\ \forall k\in K \tag{10}$$

$$l_{ij} = z,\ i=2,\ \forall j\in N\backslash\{2\} \tag{11}$$

$$\sum_{i\in N}\sum_{j\in N \& j\neq i} q_j\cdot x_{ijk}^b \leqslant Q,\ \forall b\in B_k, \forall k\in K \tag{12}$$

$$\sum_{b\in B_k}\sum_{j\in N}\sum_{j\in N \& j\neq i} t_{ij}\cdot x_{ijk}^b \leqslant T_{\max},\ \forall k\in K \tag{13}$$

where Eq. (5) is the optimization objective which is to minimize the total costs of municipal solid waste collection, including fixed costs, fuel costs, and carbon emission costs. Constraint (6) means that the number of vehicle trips starting from the depot to other waste sites is equal to the number of vehicles used. Constraint (7) means that if the vehicle k is used, it should have one and only one trip starting from the depot. Thus, Constraints (6) and (7) ensure that all vehicles start from the depot and only start once. Constraint (8) guarantees that each waste site is served only once by just one vehicle. Constraints (9) and (10) indicate that when each waste site is served, there must be a vehicle driving from another point to it. After finishing the work, the vehicle leaves the waste site. Constraint (11) illustrates that all the vehicles will empty the wastes at the disposal station. Constraints (12) and (13) assure the load and the working hours of each vehicle will not exceed its specified maximum value.

4 CAPSO for EMVRPMSWC

With the aim of solving EMVRPMSWC, contribution-based adaptive particle swarm optimization named CAPSO is proposed. To expand the searching area of the algorithm, a contribution-based adaptive learning mechanism is designed. Additionally, an improved local search operator is adopted, which extends the searching depth of the algorithm.

4.1 Framework of CAPSO

The main framework of CAPSO is given in Algorithm 1. It can be found that CAPSO consists of four components: (1) initialization; (2) selection of the learning strategy based on contribution adaptively and generation of the offspring individual; (3) improved local search; and (4) update of pbest (the personal best) and gbest (the global best).

4.2 Encoding and decoding of individuals

EMVRPMSWC is a combinatorial optimization problem so that the integer encoding is used in the proposed algorithm CAPSO. For a problem containing n points (1: depot, 2: disposal station, and 3−n: waste sites), each individual is encoded as a sequence of integers between 3−n. In the context of municipal solid waste collection, individuals are decoded sequentially so that the vehicle utilization is maximized within the working hours and the full load of each vehicle is guaranteed. This mechanism of decoding can ensure that the obtained solutions are feasible since both the capacity and the time constraints are satisfied.

Figure 2 gives an illustration of the proposed encoding and decoding method. Assume there are ten waste sites numbered 3−12. The weight of wastes at each waste site is placed above the site number, and the value on each arrow represents the time required to pass through the route. The maximum capacity and working hours of each vehicle are assumed to be 20 t and 8 h, respectively. The mechanism of decoding is detailed as follows. First, Vehicle 1 starts its first trip (Trip 1) from the depot and serves in order of the waste

Algorithm 1 CAPSO framework

1: Initialization of the swarm
2: **while** termination criteria are not met
3: **for** each individual
4: Select the learning strategy based on contribution adaptively and generate the offspring individual // In Section 4.3
5: Improved local search // In Section 4.4
6: Update pbest and gbest
7: **end for**
8: **end while**

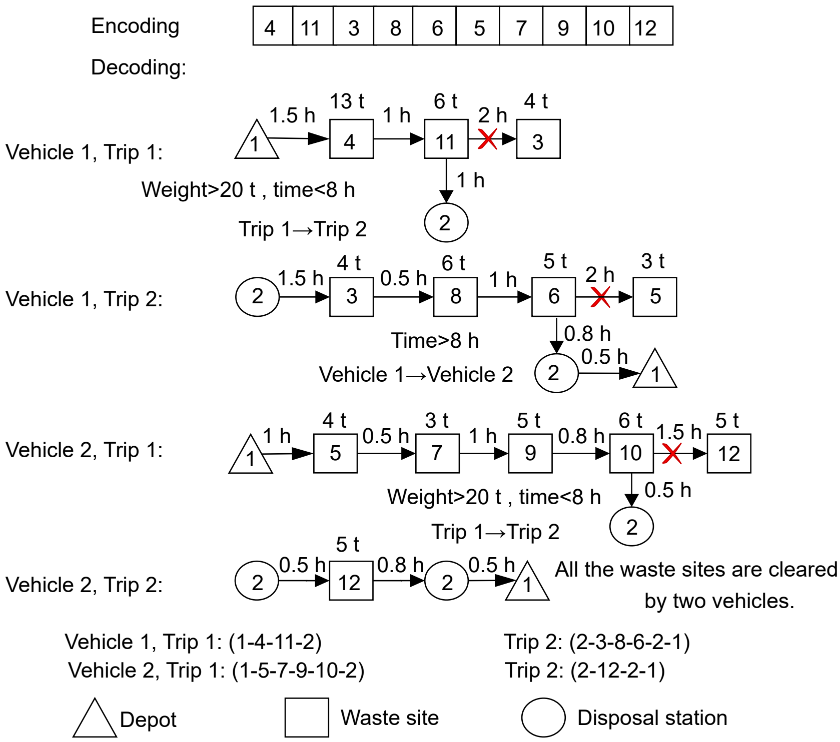


Fig. 2 Illustration of the proposed encoding and decoding method.

sites in the encoding. When Vehicle 1 visits waste site 3, the load of Vehicle 1 will be 23 t if it serves the site, exceeding the maximum capacity of the vehicle. Therefore, Vehicle 1 will give up serving waste site 3 and drive to the disposal station for unloading. Then Vehicle 1 starts its second trip (Trip 2) at the disposal station. The waste site 3 which has not been served previously due to the capacity limitation, will be the first site to be served by Vehicle 1 in Trip 2, and the remaining waste sites will be handled in sequence. If Vehicle 1 continues to serve waste site 5 after completing waste site 6, the vehicle will work for 8.5 h, exceeding the maximum working hours of its driver. Therefore, after serving waste site 6, Vehicle 1 will go to the disposal station for unloading and then return to the depot to finish its daily work. Vehicle 2 visits the remaining waste sites in the same way. In this example, all the waste sites are cleared by two vehicles.

4.3 Contribution-based adaptive learning strategy

Since EMVRPMSWC is a discrete problem, the standard Particle Swarm Optimization (PSO)[27] is no longer suitable for it. Therefore, the commonly-used crossover and mutation operators in genetic algorithm are introduced to reflect the learning mechanism in the standard PSO. Furthermore, individuals at different stages may have distinct states and searching tasks, so it is better for the learning strategy to vary with each individual in different periods. Therefore, a Contribution-based Adaptive Learning (CAL) strategy is proposed. Firstly, four learning strategies with specific roles are designed in the light of four purposes, which are summarized as rapid convergence to the global optimum, increases of the population diversity, exploration of new regions, and exploitation of the optimal individual position. Additionally, the contribution of each strategy is defined, and the most suitable learning strategy is selected for each individual adaptively based on the contribution value, which maximizes the searching efficiency of individuals at different stages.

4.3.1 Four different learning strategies

(1) A greedy learning strategy

At the early stage of evolution, the population is in the state of rapid convergence so that it can approach the extreme point quickly. Therefore, the greedy learning strategy is designed according to the characteristics of EMVRPMSWC. First, considering that EMVRPMSWC is an asymmetric problem[28, 29], the costs generated on the same route in the opposite direction may be very different. Thus, the reversion mutation operator[30] is selected to replace the “inertia” part of the velocity update formula in standard PSO. In the operator, the routes between two randomly selected points in the individual are arranged in reverse order, while the unreversed parts are preserved as inertia. Then, the greedy crossover operator which tracks the extremum is used to reflect the “self-learning ability” and “social learning ability” in the velocity update formula. Since the crossover operator allows the information interaction between two discrete individuals, it can make the individual learn from pbest and gbest. Moreover, inspired by the idea of greedy search, the information of vehicle capacity and distance is introduced into the crossover operator so that the transportation costs and carbon emissions of the waste sites served in sequence are relatively small. The proposed greedy learning strategy not only makes PSO suitable for solving the discrete problems like EMVRPMSWC, but also accelerates the convergence of the algorithm.

The procedure of the greedy learning strategy is described as follows. First, the mutated individual is generated through the reversion mutation operator. Next, Individual 1 is generated by performing the greedy crossover operator on the mutated individual and pbest. Last, Individual 1 and gbest are crossed in the same way. Due to the space limitation, the pseudo code of the greedy crossover operator is presented in Algorithm A1 in Appendix A.

(2) A multi-operator learning strategy

When the population has evolved for a certain number of generations through the greedy learning strategy, most of the individuals converge to the same region rapidly due to the utilization of heuristic information. In this case, the searching range of the population is very limited, resulting in the premature convergence. Therefore, at this stage of evolution, learning strategies of individuals should be adjusted, which can keep population diversity while producing new individuals. With this idea, a multi-operator learning strategy is designed. The “inertia” part of the standard PSO is replaced by the multi-mutation operator, and the partial mapped crossover operator[31] is used to realize the information interaction between the individual and pbest (or gbest).

The procedure of the multi-operator learning strategy is presented as follows. First, the mutated individual is generated by the multi-mutation operator, where one mutation operator is selected uniformly at random from the three candidate ones of swap mutation, reverse mutation, and insertion mutation. Second, Individual 1 is produced by performing the partial mapped crossover operator on the mutated individual and pbest. Last, Individual 1 and gbest are crossed in the same way to obtain a new individual. Compared with the greedy learning strategy, the multi-operator learning strategy has richer ways to create new individuals, which increases the population diversity effectively.

(3) An exploring learning strategy

When assimilation of the population is serious, comprehensive learning of the neighborhood information is beneficial to explore a wider range of potential space, so as to achieve global optimization. It was proved in Ref. [32] that it is more conducive to global search by making each dimension of an individual learn randomly from its own pbest or other individual’s pbest. For this, an exploring learning strategy is proposed, which adopts the multi-mutation operator to retain inertia, and the partial mapped crossover operator to realize the learning process, as done in the multi-operator learning strategy. The difference between them lies in that in the exploring learning strategy, each dimension of the current individual can learn stochastically and independently from the personal best of any other individual in the population, as long as it has a better fitness than the current one. However, the multi-operator learning strategy just takes gbest and the individual’s own pbest as the learning objects. The exploring learning strategy enriches the sources of information interaction among individuals, which diversifies the searching directions of population while guaranteeing the quality of individuals.

(4) An exploiting learning strategy

With the aim of balancing exploration and exploitation, local search ability of the algorithm needs to be further enhanced. With this in mind, an exploiting learning strategy is proposed, in which each individual only learns from pbest by the partial mapped crossover operator. The “inertia” part of standard PSO is still replaced by the multi-mutation operator. This strategy makes full use of the valuable information of the individual itself, which is beneficial to find a better solution around pbest.

4.3.2 Definition of the contribution

At different stages of the evolution, individuals in various states need to select appropriate learning strategies adaptively according to their search roles, so as to maximize the utility of each individual. In this section, the contribution of each learning strategy in Section 4.3 is defined on the basis of its improvement to the performance of the algorithm during the evolution.

(1) Assume that there are m learning strategies.

When initializing the population, the contribution of each learning strategy is set to be $C_k=0$, $k=1, 2, \ldots, m$.

(2) In a certain evolutionary generation, a new population is created through distinct learning strategies. The learning strategy selected by the i-th individual is recorded as learn(i). G new individuals are ranked from the best to the worst according to the objective value, and the ranking $r(i)$ of the i-th new individual is obtained. The weight w_i is assigned to the i-th new individual as shown in Eq. (14). It can be seen that the better the ranking of the individual is, the greater the weight is assigned.

$$w_i = \frac{\log(G - r(i) + 1)}{\log(1) + \log(2) + \cdots + \log(G)},\ i = 1,2,...,G \tag{14}$$

(3) The contribution of the k-th learning strategy is updated according to Eq. (15).

$$C_k = \begin{cases} C_k + f_i \cdot w_i, & \text{if learn}(i) = k,\ k = 1,2,...,m, \\ C_k, & \text{else} \end{cases} \tag{15}$$

where f_i is the reward factor. If the objective value of the i-th individual is improved through the k-th learning strategy, $f_i=1$; otherwise, $f_i=0$.

4.3.3 Contribution-based adaptive learning mechanism

In order to reduce the blind search which causes the waste of computational resources, the algorithm should have the ability to search autonomously and have a definite purpose at a certain evolutionary stage. Thus, a contribution-based adaptive learning mechanism is designed, where the selection probability of each learning strategy is adjusted dynamically with the change of the contribution. Set the initial selection probability of the k-th learning strategy to be $p_k=1/m$ ($k=1, 2, \ldots, m$). After the selection, probabilities of all the learning strategies are updated as shown in Eq. (16), they are normalized as Eq. (17).

$$p_k = C_k + \varepsilon,\ \ k = 1,2,...,m \tag{16}$$

$$P_k = \frac{p_k}{\sum\limits_{j=1}^{m} p_j},\ k = 1,2,...,m \tag{17}$$

where a small constant $\varepsilon = 0.01$ is added to ensure that each learning strategy has a chance to be selected. Pseudo code of the contribution-based adaptive learning mechanism is given in Algorithm A2 in Appendix A.

4.4 Improved local search operator

In order to perform a finer search around the decoded individuals, an Improved Local Search (ILS) operator is designed based on the classical 2-opt operator[33, 34]. Compared with the 2-opt operator, the ILS operator has the following characteristics. (1) To prevent the loss of population diversity caused by excessive exploitation, ILS is only conducted on the trips that contain the largest number of waste sites in each individual. In contrast, the original 2-opt operator is performed on all the trips of each vehicle, which may lead to the rapid population assimilation. And (2) when updating the individual after a local search, the total costs (Eq. (5)) rather than the shortest distance is adopted as the indicator. In this way, not only the distance (related to the transportation costs) but also the load of each vehicle (related to the carbon emissions) is considered. ILS can realize a deep mining in local regions, which improves both the solution accuracy and the population diversity.

Implementation of ILS is illustrated in Fig. 3. First, the individual X is decoded according to Section 4.2 to obtain a vehicle routing solution, where two vehicles are needed, and each one has two trips. Then, the vehicle indices (V) and the trip indices (T) corresponding to the trips that contain the most waste sites (excluding depots and disposal stations) are found, where $V_1=1$, $T_1=2$ and $V_2=2$, $T_2=1$. Next, the 2-opt operator is applied to the second trip of Vehicle 1 and the first trip of Vehicle 2 to effectively find new trips with lower costs. Finally, the optimized trips {(2-6-8-11-4-9-2-1), (1-13-16-10-12-14-2)} and the unselected trips {(1-3-5-7-2), (2-18-15-17-2-1)} are recombined sequentially to form a new individual. In the process of recombination, there are two situations based on the destination of the trip. (1) When the destination of the trip is the disposal station, it indicates that the vehicle will not go to the depot in the current trip. Therefore, only the starting point and the ending point (disposal station) of the trip are removed. And (2) when the destination of the trip is the depot, it means that the vehicle must go to the disposal station for unloading before returning to the depot. As a result, the starting point, the disposal station, and the ending point (depot) of the trip need to be removed. If the original individual X is worse than the new generated X_{new} in terms of the total costs (Eq. (5)), then it is replaced by X_{new}. The above ILS procedure is performed Y times around each decoded individual.

4.5 Complexity analysis of CAPSO

Time complexity of the standard PSO is $O(GD)$, where G is the population size, and D is the individual

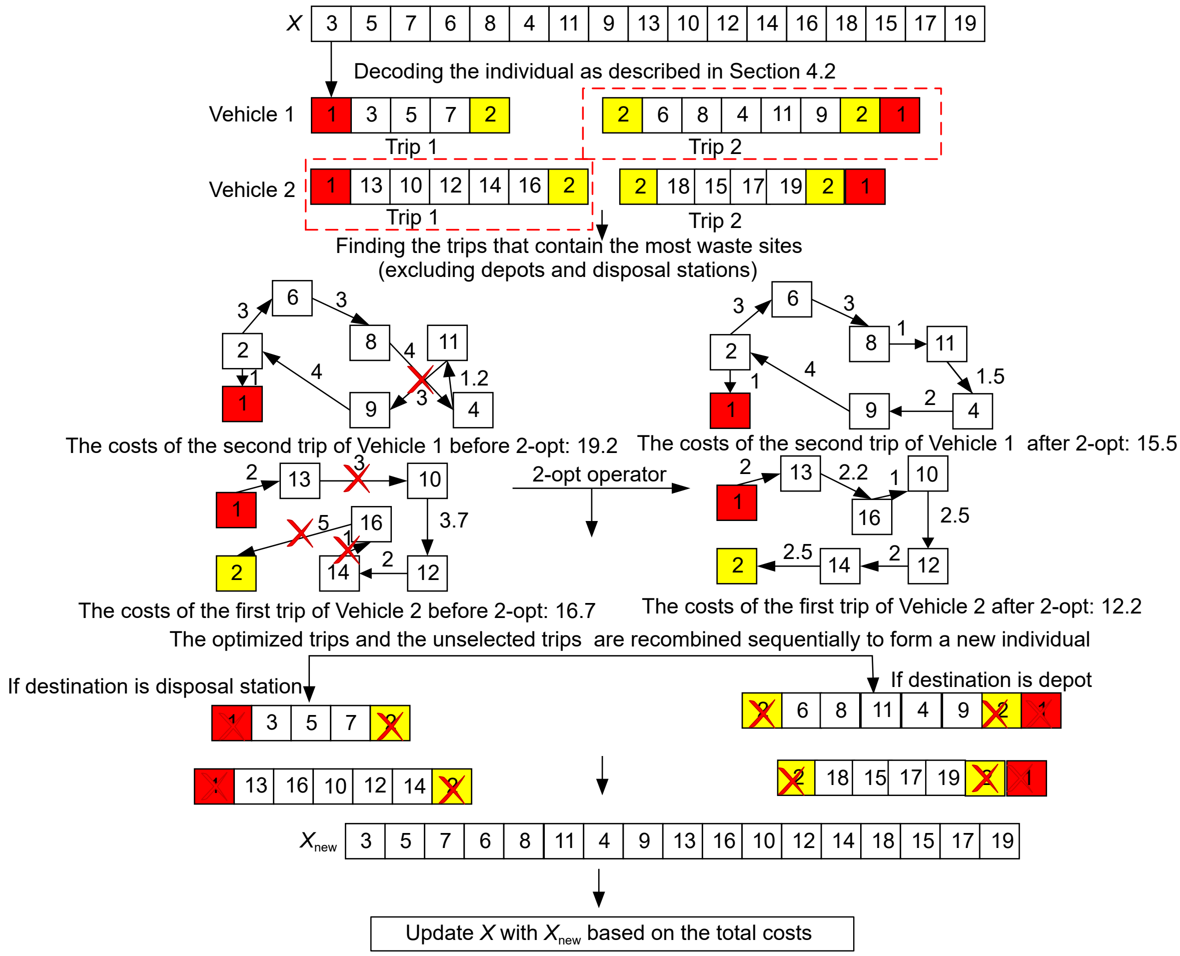


Fig. 3 Implementation of the improved local search operator.

dimension. For the proposed algorithm CAPSO, the CAL strategy is employed to generate new individuals, which only changes the individual learning mechanism, and its time complexity is consistent with the individual update method of the standard PSO. At the same time, the ILS operator is adopted to search Y times around the neighborhood of each decoded individual, and its time complexity is $O(GDY)$. To sum up, the total time complexity of CAPSO is $O(GDY)$. Although the time complexity of CAPSO is Y times higher than the standard PSO due to the introduction of the local search, its searching accuracy has improved significantly. Since all the original individuals are replaced by the updated ones in CAPSO, no additional memory space is required. Thus, the space complexity of CAPSO is $O(GD)$, which is the same as that of the standard PSO.

5 Experimental Study

In this paper, three groups of experiments on nine synthetic instances and one real case are conducted. The first group of experiments validates the effectiveness of the two new strategies. The overall performance of the proposed algorithm is analyzed in the second group. In the third group, the optimal solution to EMVRPMSWC and the corresponding vehicle routes are presented in a real-world case. All the experiments are implemented in MATLAB running on a personal computer with Intel (R) Core (TM) i5-5500U.

5.1 Instances and parameter settings

Considering that EMVRPMSWC is a real-world problem, we made an investigation on the Green Ring Company of Jiangbei New District in Nanjing, China. First, by querying on a map, the longitude and latitude coordinates of 54 residential areas, one depot, and one disposal station in Jiangbei New District were obtained, which were then transformed to the plane coordinates by the COORD coordinate converter. Next, the daily waste amount of each residential area was obtained by the standing book of Green Ring Company, and the capacity of each vehicle was set to 10 t. Finally, we got a real instance on municipal solid waste collection by integrating all the data.

Furthermore, in order to measure the scalability of the proposed algorithm to problems with different sizes, nine synthetic instances ranging from 17 to 201 customer points were selected from the commonly-used CVRP benchmark datasets named Set A, Set B, Set E, and Set CMT (http://vrp.atd-lab.inf.puc-rio.br/index.php/en/). To make the synthetic instances conform to the actual situation of municipal solid waste collection, we processed the data of the original datasets, which are shown in Appendix B in detail. In above, a total of ten real and synthetic instances were employed in the experiments, which were named Pro-*n* each (*n* is the problem scale, i.e., the total number of residential areas, the depot and the disposal station, and Pro-56 is the real instance mentioned above). Parameter settings[35] of the EMVRPMSWC model are listed in Table A1 in Appendix B.

In CAPSO, the population size G is set to 200 and the number of local search iterations Y is set to 10, which are commonly-used in the standard PSO and the 2-opt operator[32, 33]. To give a fair comparison, all the algorithms perform 30 independent runs on each instance, and the maximum number of the objective evaluations FE_{max} is set to 100 000 in each run.

5.2 Validating the effectiveness of the two new strategies

Effectiveness of the CAL strategy and the ILS operator is verified in this section.

5.2.1 Validating the effectiveness of the CAL strategy

The CAL strategy in CAPSO is replaced by each of the four learning strategies presented in Section 4.3.1, which produces four algorithms named PSO_learn1, PSO_learn2, PSO_learn3, and PSO_learn4. Besides, a multi-strategy mechanism which randomly selects one of the four learning strategies for each individual is also adopted to replace the CAL strategy, resulting in an algorithm named RSPSO. They are compared with CAPSO to analyze the influence of the CAL strategy on the performance of the algorithm. The results are shown in Table 2, where "Best" and "Mean" represent the best value and the mean value of the results obtained in 30 runs, respectively. Furthermore, to significantly compare different algorithms, Wilcoxon rank sum test with the significance level of 0.05 is employed, where the sign of "+/−/=" in CAPSO vs. B indicates that CAPSO is significantly better than B, significantly worse than B, or there is no significant difference, respectively.

It can be seen from Table 2 that CAPSO is better than the five comparison algorithms on all instances except the small-scale instances Pro-17 to Pro-33, no matter the "Best" values or the "Mean" values. Wilcoxon rank sum test results also show that CAPSO is significantly better than the five comparison algorithms on these instances. The above results indicate that the CAL strategy is feasible and effective, which integrates four learning strategies with distinct advantages adaptively, and enables the algorithm to search more precisely based on the evolutionary state at different stages.

In order to graphically display the selection of learning strategies of CAPSO at each stage of the evolution, one large-scale instance Pro-152 is taken as an example, and the times of each strategy chosen at different stages are counted, as shown in Fig. 4. FE_{max} is divided into three equal parts to reflect the early, middle, and later stages of the evolution. It can be seen from Fig. 4 that the greedy learning strategy is chosen by individuals most often at the early stage. The reason is that it can make the individuals capture the valuable information of pbest and gbest quickly, which makes a great contribution to the early convergence of the population. Therefore, the greedy learning strategy has a higher probability to be selected. With the progress of the evolution, it is difficult for individuals to generate new individuals with better objective values through the greedy learning strategy, so its contribution changes little. With the contribution of other strategies increasing, the selection probability of the greedy learning strategy decreases, and its advantage loses gradually. When the population is trapped into the local optimum, diversity needs to be introduced to expand the searching area and create individuals in new regions. The multi-operator learning strategy enriches the way to generate individuals, and the exploring learning strategy brings in the new learning objects, both of which help the population jump out of the local optimum. The contributions of these two strategies also increase with their improvement to the individual objective value.

Therefore, at the middle stage of the evolution, these two strategies are selected by more individuals gradually. Finally, when the population evolves to the later stage, the region where the optimal solution locates has been found roughly. At this moment, the

Table 2 Statistical test results before and after using improved strategies on the 10 EMTRMSWC instances.

Problem	Indicator	PSO_learn1	PSO_learn2	PSO_learn3	PSO_learn4	RSPSO	CAPSO-ILS	CAPSO
Pro-17	Best	**1.47×10³**	**1.47×10³**	**1.47×10³**	**1.47×10³**	**1.47×10³**	**1.47×10³**	**1.47×10³**
	Mean	1.48×10³	**1.47×10³**	**1.47×10³**	**1.47×10³**	**1.47×10³**	1.48×10³	**1.47×10³**
	Wilcoxon	+	=	=	=	=	=	—
Pro-23	Best	**2.27×10³**	**2.27×10³**	**2.27×10³**	**2.27×10³**	**2.27×10³**	**2.27×10³**	**2.27×10³**
	Mean	2.30×10³	**2.28×10³**	**2.28×10³**	**2.28×10³**	2.29×10³	2.29×10³	**2.28×10³**
	Wilcoxon	+	=	+	+	+	+	—
Pro-33	Best	2.96×10³	**2.95×10³**	2.98×10³	3.07×10³	**2.95×10³**	**2.95×10³**	**2.95×10³**
	Mean	3.03×10³	3.02×10³	3.08×10³	3.24×10³	2.97×10³	3.00×10³	**2.96×10³**
	Wilcoxon	+	+	+	+	=	+	—
Pro-56	Best	6.23×10³	6.40×10³	6.63×10³	7.12×10³	6.09×10³	6.28×10³	**6.01×10³**
	Mean	6.57×10³	6.85×10³	6.99×10³	7.38×10³	6.25×10³	6.55×10³	**6.13×10³**
	Wilcoxon	+	+	+	+	+	+	—
Pro-65	Best	6.67×10³	7.08×10³	7.31×10³	7.92×10³	6.44×10³	6.65×10³	**6.37×10³**
	Mean	7.13×10³	7.67×10³	7.67×10³	8.52×10³	6.67×10³	6.99×10³	**6.57×10³**
	Wilcoxon	+	+	+	+	+	+	—
Pro-79	Best	8.51×10³	9.89×10³	9.81×10³	1.28×10³	8.01×10³	8.66×10³	**7.93×10³**
	Mean	9.88×10³	1.15×10³	1.10×10³	1.38×10³	8.64×10³	9.78×10³	**8.14×10³**
	Wilcoxon	+	+	+	+	+	+	—
Pro-81	Best	1.04×10⁴	1.21×10⁴	1.20×10⁴	1.44×10⁴	9.84×10³	1.06×10⁴	**9.77×10³**
	Mean	1.11×10⁴	1.40×10⁴	1.38×10⁴	1.52×10⁴	1.01×10⁴	1.12×10⁴	**9.97×10³**
	Wilcoxon	+	+	+	+	+	+	—
Pro-102	Best	1.59×10⁴	1.94×10⁴	1.90×10⁴	2.23×10⁴	1.59×10⁴	1.62×10⁴	**1.53×10⁴**
	Mean	1.79×10⁴	2.10×10⁴	2.08×10⁴	2.35×10⁴	1.69×10⁴	1.81×10⁴	**1.61×10⁴**
	Wilcoxon	+	+	+	+	+	+	—
Pro-152	Best	2.27×10⁴	3.02×10⁴	2.76×10⁴	3.38×10⁴	2.12×10⁴	2.25×10⁴	**2.04×10⁴**
	Mean	2.39×10⁴	3.21×10⁴	2.98×10⁴	3.48×10⁴	2.21×10⁴	2.43×10⁴	**2.19×10⁴**
	Wilcoxon	+	+	+	+	+	+	—
Pro-201	Best	2.85×10⁴	4.02×10⁴	3.54×10⁴	4.46×10⁴	2.66×10⁴	2.96×10⁴	**2.47×10⁴**
	Mean	3.12×10⁴	4.34×10⁴	3.81×10⁴	4.77×10⁴	2.75×10⁴	3.27×10⁴	**2.71×10⁴**
	Wilcoxon	+	+	+	+	+	+	—
Total	+/=/−	10/0/0	8/2/0	9/1/0	9/1/0	8/2/0	9/1/0	—

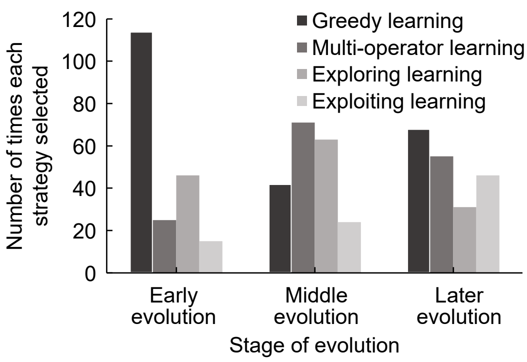


Fig. 4 Number of times each strategy selected at different evolutionary stages.

local search ability of the algorithm needs to be enhanced to further improve the solution accuracy. The greedy learning strategy makes full use of the problem features to perform a fine search around the individuals. The exploiting learning strategy is beneficial to mine better solutions near the personal best. The multi-operator learning strategy is responsible for maintaining diversity of the population. As a result, the above three strategies are chosen evenly by individuals at the later stage of the evolution. The results in Fig. 4 indicate that the proposed CAL strategy can enable the individuals at different stages to select an appropriate learning strategy correctly, which helps the algorithm make a good balance between exploration and exploitation. Similar results can be obtained from other instances.

5.2.2 Validating the effectiveness of the ILS operator

In order to validate the effectiveness of the ILS operator given in Section 4.4, it is removed from the proposed algorithm CAPSO to obtain the algorithm CAPSO-ILS, and a comparison between CAPSO-ILS and CAPSO is carried out. The results are shown in Table 2.

As shown in Table 2, CAPSO-ILS has the same “Best” values as CAPSO only on the small-scale instances (Pro-17 to Pro-33). On the other instances, the values of “Best” found by CAPSO are better than those of CAPSO-ILS. In terms of the “Mean” values, CAPSO is superior to CAPSO-ILS on all the instances. The reason is that the ILS operator provides a great help for the fine search of the algorithm. ILS not only enhances the ability to mine the area near the individual, but also prevents excessive exploitation of the individual neighborhood. Therefore, it can improve the solution accuracy and reduce the risk of rapid assimilation of the population.

To observe the convergence of all the algorithms visually, the convergence curves on instance Pro-152 are given in Fig. 5, which shows how the optimal objective values found by each algorithm change with the number of objective evaluations. According to the curves, it can be clearly seen that both the solution accuracy and the convergence speed of the proposed algorithm CAPSO are superior to the other comparison algorithms on Pro-152. CAPSO jumps out of the local optimum for several times and finally obtains a solution with higher precision. Similar results can be obtained from the other instances.

5.3 Validating the performance of the proposed algorithm

In order to evaluate the overall performance of the proposed algorithm in solving EMVRPMSWC, we re-

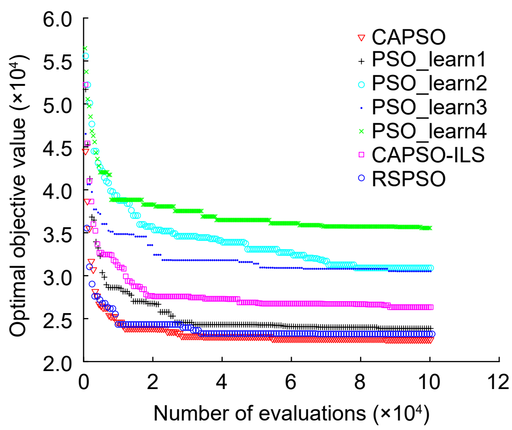


Fig. 5 Convergence curves of the seven algorithms on instance Pro-152 when validating effectiveness of the strategies.

implement five state-of-the-art meta-heuristic algorithms and compare them with CAPSO. As mentioned in Section 2.1, all the existing models on the municipal solid waste collection are based on the vehicle routing problem with a single trip, while our EMVRPMSWC is modeled based on the Multi-Trip Vehicle Routing Problem (MTVRP), which are different from each other. Thus, only one meta-heuristic algorithm for solving the waste collection problem[19] is selected as a comparison algorithm. In their work, PSO is adopted, which is real-encoded and mapped into discrete numbers by sigmoid function and rounding function. To adapt it to our multi-trip model, each individual is decoded into routes by our decoding method introduced in Section 4.2. Considering the multi-trip nature of our model, a Hybrid Genetic Algorithm (HGA)[36] and Tabu Search (TS)[37] used to solve MTVRP (not a waste collection problem) are adopted as the second and the third comparison algorithms. HGA is based on traditional GA and a reorder routine is adapted to accelerate HGA. In TS, the nearest-neighbor heuristic is used to obtain an initial solution and the Tabu search heuristic is proposed to improve the quality of the solution. Since there is also little research on MTVRP, two methods handling the multi-trip arc routing problem are also selected as two additional comparison algorithms. One is an Improved Max-Min Ant System (IMMAS)[38], which also uses the Taguchi parameter design method. The other is a hybrid algorithm combining Simulated Annealing and Genetic Algorithm (SA+GA)[39], where a simulated annealing algorithm is applied to generate initial solutions and a genetic algorithm is then used to generate the best possible solution.

For the population size, IMMAS takes the value of 50 according to its original literature (Ref. [38]), while PSO, HGA, and SA+GA take 200, which is the same as the proposed algorithm. All the algorithms stop after 100 000 objective evaluations in one run, and other parameters are set in accordance with the literature. Comparison results (the best, mean, and worst values of the results obtained in 30 runs) of the six algorithms are shown in Table 3.

As shown in Table 3, the “Best” values found by CAPSO are better than the other five comparison algorithms on most of the medium-scale and large-scale instances (Pro-56 and the above), and it is only worse than IMMAS on the instance Pro-102. The “Worst” values found by CAPSO are also better than the other five algorithms on most instances. Wilcoxon

Table 3 Statistical test results of the six algorithms on the 10 EMVRPMSWC instances.

Problem	Indicator	HGA	SA+GA	IMMAS	TS	PSO	CAPSO
Pro-17	Best	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$
	Mean	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	1.48×10^{3}	1.48×10^{3}	$\mathbf{1.47\times10^{3}}$
	Worst	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	$\mathbf{1.47\times10^{3}}$	1.49×10^{3}	1.49×10^{3}	$\mathbf{1.47\times10^{3}}$
	Wilcoxon	=	=	=	+	+	—
Pro-23	Best	$\mathbf{2.27\times10^{3}}$	$\mathbf{2.27\times10^{3}}$	2.28×10^{3}	$\mathbf{2.27\times10^{3}}$	$\mathbf{2.27\times10^{3}}$	$\mathbf{2.27\times10^{3}}$
	Mean	2.28×10^{3}	$\mathbf{2.27\times10^{3}}$	2.29×10^{3}	2.30×10^{3}	2.29×10^{3}	2.28×10^{3}
	Worst	2.31×10^{3}	$\mathbf{2.27\times10^{3}}$	2.32×10^{3}	2.36×10^{3}	2.34×10^{3}	2.34×10^{3}
	Wilcoxon	=	=	+	+	=	—
Pro-33	Best	$\mathbf{2.95\times10^{3}}$	$\mathbf{2.95\times10^{3}}$	2.99×10^{3}	$\mathbf{2.95\times10^{3}}$	$\mathbf{2.95\times10^{3}}$	$\mathbf{2.95\times10^{3}}$
	Mean	2.98×10^{3}	2.99×10^{3}	3.03×10^{3}	3.01×10^{3}	3.02×10^{3}	$\mathbf{2.96\times10^{3}}$
	Worst	3.07×10^{3}	3.07×10^{3}	3.14×10^{3}	3.11×10^{3}	3.09×10^{3}	$\mathbf{3.03\times10^{3}}$
	Wilcoxon	=	+	+	+	+	—
Pro-56	Best	6.14×10^{3}	6.20×10^{3}	6.51×10^{3}	6.94×10^{3}	6.26×10^{3}	$\mathbf{6.01\times10^{3}}$
	Mean	6.43×10^{3}	6.63×10^{3}	6.64×10^{3}	7.76×10^{3}	6.78×10^{3}	$\mathbf{6.13\times10^{3}}$
	Worst	6.92×10^{3}	7.09×10^{3}	7.24×10^{3}	7.92×10^{3}	7.13×10^{3}	$\mathbf{6.48\times10^{3}}$
	Wilcoxon	+	+	+	+	+	—
Pro-65	Best	6.57×10^{3}	6.89×10^{3}	6.79×10^{3}	7.12×10^{3}	6.95×10^{3}	$\mathbf{6.37\times10^{3}}$
	Mean	6.99×10^{3}	7.28×10^{3}	6.97×10^{3}	7.87×10^{3}	7.64×10^{3}	$\mathbf{6.57\times10^{3}}$
	Worst	7.45×10^{3}	7.58×10^{3}	7.05×10^{3}	8.02×10^{3}	7.89×10^{3}	$\mathbf{6.96\times10^{3}}$
	Wilcoxon	+	+	+	+	+	—
Pro-79	Best	8.50×10^{4}	9.54×10^{3}	8.34×10^{3}	1.07×10^{4}	9.59×10^{3}	$\mathbf{7.93\times10^{3}}$
	Mean	9.77×10^{3}	1.04×10^{4}	8.42×10^{3}	1.30×10^{4}	1.10×10^{4}	$\mathbf{8.14\times10^{3}}$
	Worst	1.08×10^{4}	1.12×10^{4}	9.79×10^{3}	1.52×10^{4}	1.34×10^{4}	$\mathbf{9.63\times10^{3}}$
	Wilcoxon	+	+	+	+	+	—
Pro-81	Best	1.03×10^{4}	1.12×10^{4}	9.91×10^{3}	1.41×10^{4}	1.59×10^{4}	$\mathbf{9.77\times10^{3}}$
	Mean	1.16×10^{4}	1.24×10^{4}	1.29×10^{4}	1.51×10^{4}	1.75×10^{4}	$\mathbf{9.97\times10^{3}}$
	Worst	1.55×10^{4}	1.43×10^{4}	1.38×10^{4}	1.77×10^{4}	1.98×10^{4}	$\mathbf{1.03\times10^{4}}$
	Wilcoxon	+	+	+	+	+	—
Pro-102	Best	1.69×10^{4}	1.83×10^{4}	$\mathbf{1.52\times10^{4}}$	2.29×10^{4}	1.82×10^{4}	1.53×10^{4}
	Mean	1.84×10^{4}	1.95×10^{4}	$\mathbf{1.57\times10^{4}}$	2.51×10^{4}	2.00×10^{4}	1.61×10^{4}
	Worst	1.98×10^{4}	2.08×10^{4}	$\mathbf{1.59\times10^{4}}$	2.79×10^{4}	2.23×10^{4}	1.77×10^{4}
	Wilcoxon	+	+	−	+	+	—
Pro-152	Best	2.61×10^{4}	2.76×10^{4}	2.29×10^{4}	3.25×10^{4}	2.77×10^{4}	$\mathbf{2.04\times10^{4}}$
	Mean	2.79×10^{4}	3.01×10^{4}	2.37×10^{4}	3.50×10^{4}	3.09×10^{4}	$\mathbf{2.19\times10^{4}}$
	Worst	3.01×10^{4}	3.26×10^{4}	2.58×10^{4}	3.84×10^{4}	3.21×10^{4}	$\mathbf{2.33\times10^{4}}$
	Wilcoxon	+	+	+	+	+	—
Pro-201	Best	3.43×10^{4}	3.50×10^{4}	3.02×10^{4}	4.47×10^{4}	3.78×10^{4}	$\mathbf{2.47\times10^{4}}$
	Mean	3.71×10^{4}	3.91×10^{4}	3.21×10^{4}	4.95×10^{4}	4.11×10^{4}	$\mathbf{2.71\times10^{4}}$
	Worst	4.02×10^{4}	4.24×10^{4}	3.55×10^{4}	5.29×10^{4}	4.33×10^{4}	$\mathbf{2.85\times10^{4}}$
	Wilcoxon	+	+	+	+	+	—
Total	+/=/−	7/3/0	8/2/0	8/1/1	10/0/0	9/1/0	—

rank sum test results show that CAPSO is significantly better than HGA, SA+GA, IMMAS, TS, and PSO on seven, eight, eight, ten, and nine instances, respectively. CAPSO is only significantly worse than IMMAS on the instance Pro-102. The above results show that the overall performance of CAPSO is better than all kinds of comparison algorithms when solving EMVRPMSWC with different sizes, which indicates that it has a promising scalability to the problem size.

The reason for the good performance of the proposed CAPSO is the effective coordination of the two novel strategies. The ILS operator adopts the improved 2-opt operator to mine the neighborhood of the individual after decoding, which extends the searching depth of the algorithm. However, by using ILS, the algorithm tends to fall into the local optimum and fail to converge to the true global optimal solution. Therefore, when generating new individuals, not only the useful information about the population should be exploited, but also the searching diversity needs to be taken into account. For this, the CAL strategy is designed to serve the generation of new individuals from two perspectives such as the variety of learning objects and the multiplicity of learning mechanisms, which broadens the searching area of the algorithm. The two strategies complement each other and achieve a complete and sophisticated search of the search space along many different directions, which ensures that the algorithm makes a good balance between exploration and exploitation.

Convergence curves of the six algorithms compared on instance Pro-201 are shown in Fig. 6, which indicates that both the solution accuracy and the convergence speed of the proposed algorithm CAPSO are superior to the other comparison algorithms.

5.4 Vehicle routes of EMVRPMSWC

To further validate the feasibility and effectiveness of CAPSO for solving EMVRPMSWC, the optimal vehicle routes on the real-world case (Pro-56) are presented in Fig. 7. In Fig. 7, the black dot, the red star, and the green square denote the waste sites, the depot, and the disposal station, respectively. The black dotted line and the red solid line give the trips traveled by Vehicle 2 and Vehicle 1, respectively. It can be found from Fig. 7 that a total of two vehicles are needed to complete all the tasks on the instance. Vehicle 1 has four trips, which are (1-21-14-28-29-31-32-30-27-22-25-50-2), (2-41-40-37-39-52-46-42-2), (2-4-43-53-34-33-38-36-56-35-2), and (2-54-49-55-26-48-47-51-45-44-2-1). The schedule of Vehicle 2 is (1-13-3-20-10-9-6-5-8-12-2) and (2-11-23-15-16-19-17-18-7-24-2-1). In

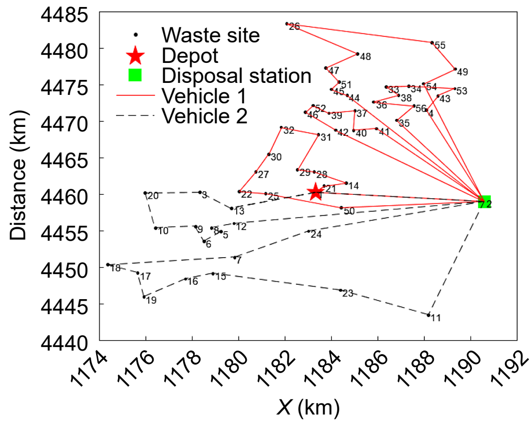


Fig. 7 Vehicle routing solution of the proposed algorithm on the real instance Pro-56.

EMVRPMSWC, the objective is to minimize the transportation costs and carbon emissions, and there is no high requirement for the completion time, which means the tasks only need to be completed within the maximum working hours of the driver. Therefore, reducing the number of vehicles is helpful to save transportation costs and carbon emissions. In addition, after the last waste site 24 is served by Vehicle 2 in the second trip, all the tasks have been completed. Thus, Vehicle 2 has only two trips. Figure 8 gives the comparison results of the six algorithms with respect to the fixed costs, fuel costs, carbon emission costs, and total costs. As shown in Fig. 8, the proposed CAPSO obtains the lowest carbon emission costs and transportation costs among all the comparison algorithms. Compared with HGA and SA+GA, although CAPSO has a slightly higher fuel costs, its improvement in the carbon emission costs is much more than its deterioration in the fuel costs. So the total costs of CAPSO are also lower. The reason for the lower carbon emission costs obtained by CAPSO is

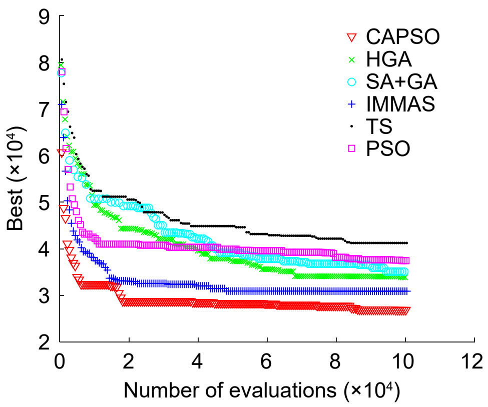


Fig. 6 Convergence curves of the six algorithms on instance Pro-201 when validating the overall performance of CAPSO.

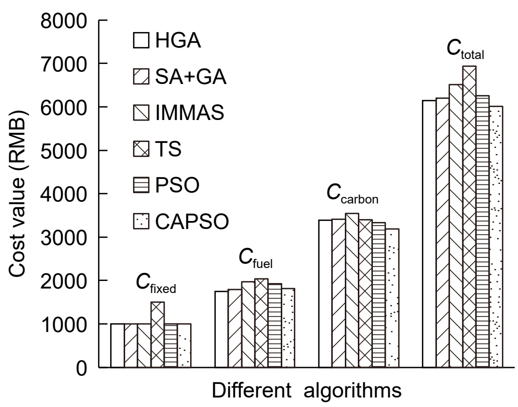


Fig. 8 Comparison results of the optimal solutions obtained by the six algorithms in terms of each cost.

that the fuel costs only involve the distance information, while the carbon emission costs are related to both the distance and load. Distance and load are utilized as the heuristic information to guide the search in CAPSO (see Section 4.4), whereas only the distances between waste sites are considered in the other comparison algorithms. As stated above, CAPSO can minimize the transportation costs and carbon emissions by reasonably scheduling the vehicles and routes, which achieves the goal of “energy saving and emission reduction”.

6 Conclusion

This work aims to study the vehicle routing problem for collecting municipal solid wastes in the residential areas. To achieve this, we construct an energy-efficient multi-trip vehicle routing model for it, which makes full use of the vehicle resources. Then contribution-based adaptive particle swarm optimization is proposed to solve the model, where four different learning strategies with distinct search roles are designed and the most suitable one is chosen for each individual adaptively in accordance with the contribution value. Additionally, an improved local search operator is designed to only perform on the trips that contain the most waste sites, which avoids overexploitation. Experimental results on one real-world waste collection problem and nine synthetic instances indicate that compared with five state-of-the-art algorithms, the proposed algorithm has significant superiorities in solving the formulated model with respect to the solution accuracy and the convergence speed.

Although some realistic factors like multiple trips of each vehicle, working hours of the drivers, and the limited vehicle capacity are considered in EMVRPMSWC, it still has a distance from the real municipal solid waste collection situation. For example, the break time for drivers during a working day is not taken into account, and our current work assumes that the amount of waste at each waste site is known beforehand and invariable. However, there may be changes in the actual situation. How to establish a model of the waste collection in an uncertain environment is one of the research directions.

Appendix

A Pseudo codes and their descriptions

The interaction between an individual and its pbest is taken as an example. Algorithm A1 is described as follows. The first waste site S is selected from the non-depot and non-disposal stations, and then its left and right waste sites in X and X_{pbest} form the candidate set $\{S_{\text{Lp}}, S_{\text{Rp}}, S_{\text{LX}}, S_{\text{RX}}\}$ (Lines 2 and 3). The next waste site S' to be visited is chosen from the candidate set which has the least transportation cost to the current site S (Lines 4−7). Line 8 distinguishes two cases. One is that the next waste site S' is on the left side of S. Since S is deleted from X_{pbest} and X (Lines 5 and 6), the right side of S' is empty. When S' is set as the current visited waste site S, a new waste site S' to be served can only be chosen from the left candidate set $\{S_{\text{Lp}}, S_{\text{LX}}\}$ of S (Lines 9−11). The other is that S' is selected from the right side of S. Thus, the left side of S' is empty. When S' is set as the current site S, a new site S' to be served can only be chosen from the right candidate set $\{S_{\text{Rp}}, S_{\text{RX}}\}$ of S (Lines 12−14). Each waste site that has been selected is removed from X_{pbest} and X to avoid repeated access to one site (Lines 17 and 18). Lines 8−19 are repeated until a new complete individual is generated.

Algorithm A1 Greedy_Crossover (X_{pbest}, X, D)

Input: X_{pbest} (pbest), X (individual), and D (dimension of the individual)

Output: X_{new} (new individual)

1: $X_{\text{new}} \leftarrow [\ ]$;
2: Select the first waste site S from $\{3, 4, \ldots, D\}$;
3: Add the left and right waste sites of S in X and X_{pbest} into the candidate set $\{S_{\text{Lp}}, S_{\text{Rp}}, S_{\text{LX}}, S_{\text{RX}}\}$;
4: $X_{\text{new}} \leftarrow [X_{\text{new}}, S]$;
5: $X_{\text{pbest}} \leftarrow X_{\text{pbest}} \backslash \{S\}$;
6: $X \leftarrow X \backslash \{S\}$;
7: Select the next waste site S' to be visited from the candidate set which has the least transportation cost to the current waste site S;
8: **while** X is not empty **do**
9:   **if** $S' \in \{S_{\text{Lp}}, S_{\text{LX}}\}$
10:     Set S' to be the current visited waste site S;
11:     Select a new waste site S' to be visited from the left candidate set $\{S_{\text{Lp}}, S_{\text{LX}}\}$ of S which has the least transportation cost to the current waste site S;
12:   **else**
13:     Set S' to the current visited waste site S;
14:     Select a new next visited waste site S' from the right candidate set $\{S_{\text{Rp}}, S_{\text{RX}}\}$ of S which has the least transportation cost to the current waste site S;
15:   **end if**
16:   $X_{\text{new}} \leftarrow [X_{\text{new}}, S]$;
17:   $X_{\text{pbest}} \leftarrow X_{\text{pbest}} \backslash \{S\}$;
18:   $X \leftarrow X \backslash \{S\}$;
19: **end while**
20: **Output** X_{new}.

Algorithm A2 is described as follows. First, the learning strategy learn(i) is determined by the roulette selection for each individual i in the population. A new individual $N_{\text{pop}}(i)$ is generated according to learn(i), and the objective value of $N_{\text{pop}}(i)$ is evaluated (Lines 1−5). Next, the individuals in the new population are sorted based on the objective value from the best to the worst, and the ranking vector r of the individuals is obtained (Line 6). Last, the contribution C_k of each learning strategy is updated through Eqs. (14) and (15). Based on the new contribution vector C, the selection probability vector P is updated and normalized according to Eqs. (16) and (17), respectively (Lines 7−15).

B Processing methods for synthetic instances and the related parameter settings

In the VRP datasets, the distances between custom points and the demands required by customers are generally large, which do not conform to the actual situation of waste collection and transportation between communities. Thus, the coordinates and demands in the nine synthetic instances need to be adjusted. In each instance, all the coordinates were scaled by a multiple, the value of which was determined according to the original coordinate values and the data acquired from the investigation to the real company. It not only reduced the distances between customers, but also kept their relative positions unchanged. The same operation was performed on the demands of customers to obtain the weight of wastes at each waste site. The depot and the disposal station were marked as Point 1 and Point 2, respectively, which were generated randomly within the range of adjusted coordinates.

Parameter settings of the EMVRPMSWC model are listed in Table A1.

Algorithm A2 CA_ learning (pop, *m*, *P*, *C*, *G*)

Input: pop (swarm population), m (number of learning strategies), $P=[p_1, p_2, \ldots, p_m]$ (the selection probability vector of learning strategies), $C=[C_1, C_2, \ldots, C_m]$ (the contribution vector), and G (population size)

Output: G_{pop} (new population), G_{popobj} (objectives of new population), P (the updated selection probability vector), and C (the updated contribution vector)

1: **for** i=1 to G **do**
2: learn(i)←Roulette(P);
3: A new individual $G_{\text{pop}}(i)$ is generated by the i-th individual pop(i) based on the learning strategy learn(i);
4: $G_{\text{popobj}}(i)$ ← evaluate ($G_{\text{pop}}(i)$); // Objective value is evaluated according to Eq. (5)
5: **end for**
6: r ← rank (G_{popobj});
7: **for** i=1 to G **do**
8: **for** k=1 to m **do**
9: **if** learn(i)==k
10: Update C_k by Eq. (15);
11: **end if**
12: **end for**
13: **end for**
14: Update P based on the updated C by Eq. (16)
15: Normalize P by Eq. (17);
16: **Output** G_{pop}, G_{popobj}, P, and C.

Acknowledgment

This work was supported by the Guangdong Provincial Key Laboratory (No. 2020B121201001), National Natural Science Foundation of China (NSFC) (Nos. 61502239 and 62002148), Natural Science Foundation of Jiangsu Province of China (No. BK20150924), and Shenzhen Science and Technology Program (No. KQTD2016112514355531).

References

[1] F. R. McDougall, P. R. White, M. Franke, and P. Hindle, *Integrated Solid Waste Management: A Life Cycle Inventory*. Oxford, UK: Blackwell Science, 2001.

[2] E. C. Rada, M. Grigoriu, M. Ragazzi, and P. Fedrizzi, Web oriented technologies and equipments for MSW collection, in *Proc. International Conference on Risk Management, Assessment, and Mitigation*, Bucharest, Romania, 2010, pp. 150–153.

[3] T. E. Kanchanabhan, J. A. Mohaideen, S. Srinivasan, and V. L. K. Sundaram, Optimum municipal solid waste collection using geographical information system (GIS)

Table A1 Parameter settings of the EMVRPMSWC model.

Notation and meaning	Value
z: weight of the vehicle itself	1 t
Q: capacity of each vehicle	10 t
$T_{\max}$: the maximum working hours of a driver	8 h
v: speed of each vehicle	30 km/h
C_e: carbon tax	5 RMB/t
C_m: unit fuel cost of a vehicle	2 RMB/km
C_f: fixed cost of a vehicle	500 RMB
a: acceleration of a vehicle	0 m/s^2
g: gravitational acceleration	9.81 m/s^2
θ: slope of the road	0°
F: fuel emission parameter	2.621×10^{-6} t/L
C_d: traction coefficient	0.7
C_r: rolling resistance coefficient	0.3
A: front surface area of the vehicle	5 m^2
ρ: air density	1.204 kg/m^3

and vehicle tracking for Pallavapuram municipality, *Waste Manag. Res.*, vol. 29, no. 3, pp. 323–339, 2011.

[4] B. G. Wilson and B. W. Baetz, Modeling municipal solid waste collection systems using derived probability Distributions. I: Model development, *J. Environ. Eng.*, vol. 127, no. 11, pp. 1031–1038, 2001.

[5] V. Wagner and D. Rutherford, Survey of best practices in emission control of in-use heavy-duty diesel vehicles, https://theicct.org/sites/default/files/publications/ICCT_HDV_in-use_20130802.pdf, 2013.

[6] E. J. Beltrami and L. D. Bodin, Networks and vehicle routing for municipal waste collection, *Networks*, vol. 4, no. 1, pp. 65–94, 1974.

[7] M. A. Hannan, M. S. H. Lipu, M. Akhtar, R. A. Begum, M. A. A. Mamun, A. Hussain, M. S. Mia, and H. Basri, Solid waste collection optimization objectives, constraints, modeling approaches, and their challenges toward achieving sustainable development goals, *J. Clean. Prod.*, vol. 277, p. 123557, 2020.

[8] H. Wu, F. Tao, Q. Qiao, and M. Zhang, A chance-constrained vehicle routing problem for wet waste collection and transportation considering carbon emissions, *Int. J. Environ. Res. Public Health*, vol. 17, no. 2, p. 458, 2020.

[9] J. C. Molina, I. Eguia, and J. Racero, Reducing pollutant emissions in a waste collection vehicle routing problem using a variable neighborhood tabu search algorithm: A case study, *TOP*, vol. 27, no. 2, pp. 253–287, 2019.

[10] H. Wu, F. Tao, and B. Yang, Optimization of vehicle routing for waste collection and transportation, *Int. J. Environ. Res. Public Health*, vol. 17, no. 14, p. 4963, 2020.

[11] I. Markov, S. Varone, and M. Bierlaire, Integrating a heterogeneous fixed fleet and a flexible assignment of destination depots in the waste collection VRP with intermediate facilities, *Transp. Res. B*: *Methodological*, vol. 84, pp. 256–273, 2016.

[12] Y. Shi, L. Lv, F. Hu, and Q. Han, A heuristic solution method for multi-depot vehicle routing-based waste collection problems, *Appl. Sci.*, vol. 10, no. 7, p. 2403, 2020.

[13] E. B. Tirkolaee, I. Mahdavi, and M. M. S. Esfahani, A robust periodic capacitated arc routing problem for urban waste collection considering drivers and crew's working time, *Waste Manag.*, vol. 76, pp. 138–146, 2018.

[14] H. R. Lewis, Computers and intractability: A guide to the theory of NP-completeness, *The Journal of Symbolic Logic*, vol. 48, no. 2, pp. 498–500, 1983.

[15] A. Louati, L. H. Son, and H. Chabchoub, Smart routing for municipal solid waste collection: A heuristic approach, *J. Ambient Intell. Humaniz. Comput.*, vol. 10, no. 5, pp. 1865–1884, 2019.

[16] Q. Wei, Z. Guo, H. C. Lau, and Z. He, An artificial bee colony-based hybrid approach for waste collection problem with midway disposal pattern, *Appl. Soft Comput.*, vol. 76, pp. 629–637, 2019.

[17] A. Expósito-Márquez, C. Expósito-Izquierdo, J. Brito-Santana, and J. A. Moreno-Pérez, Greedy randomized adaptive search procedure to design waste collection routes in La Palma, *Comput. Ind. Eng.*, vol. 137, p. 106047, 2019.

[18] Q. Qiao, F. Tao, H. Wu, X. Yu, and M. Zhang, Optimization of a capacitated vehicle routing problem for sustainable municipal solid waste collection management using the PSO-TS algorithm, *Int. J. Environ. Res. Public Health*, vol. 17, no. 6, p. 2163, 2020.

[19] M. A. Hannan, M. Akhtar, R. A. Begum, H. Basri, A. Hussain, and E. Scavino, Capacitated vehicle-routing problem model for scheduled solid waste collection and route optimization using PSO algorithm, *Waste Manag.*, vol. 71, pp. 31–41, 2018.

[20] M. A. Islam, Y. Gajpal, and T. Y. ElMekkawy, Hybrid particle swarm optimization algorithm for solving the clustered vehicle routing problem, *Appl. Soft Comput.*, vol. 110, p. 107655, 2021.

[21] Y. -J. Zheng, X. Chen, H. -F. Yan, and M. -X. Zhang, Evolutionary algorithm for vehicle routing for shared e-bicycle battery replacement and recycling, *Appl. Soft Comput.*, vol. 135, p. 110023, 2023.

[22] F. Wang, F. Liao, Y. Li, X. Yan, and X. Chen, An ensemble learning based multi-objective evolutionary algorithm for the dynamic vehicle routing problem with time windows, *Comput. Ind. Eng.*, vol. 154, p. 107131, 2021.

[23] X. Shen, J. Lu, X. You, L. Song, and Z. Ge, A region enhanced discrete multi-objective fireworks algorithm for low-carbon vehicle routing problem, *Complex System Modeling and Simulation*, vol. 2, no. 2, pp. 142–155, 2022.

[24] Y. -J. Zheng, X. Chen, Q. Song, J. Yang, and L. Wang, Evolutionary optimization of COVID-19 vaccine distribution with evolutionary demands, *IEEE Trans. Evol. Comput.*, vol. 27, no. 1, pp. 141–154, 2023.

[25] Y. Shen, L. Yu, and J. Li, Robust electric vehicle routing problem with time windows under demand uncertainty and weight-related energy consumption, *Complex System Modeling and Simulation*, vol. 2, no. 1, pp. 18–34, 2022.

[26] T. Bektaş and G. Laporte, The pollution-routing problem, *Transp. Res. B*: *Methodological*, vol. 45, no. 8, pp. 1232–1250, 2011.

[27] J. Kennedy and R. Eberhart, Particle swarm optimization, in *Proc. ICNN'95—Int. Conf. Neural Networks*, Perth, Australia, 1995, pp. 1942–1948.

[28] V. Leggieri and M. Haouari, A matheuristic for the asymmetric capacitated vehicle routing problem, *Discrete Appl. Math.*, vol. 234, pp. 139–150, 2018.

[29] H. Li, K. Gao, P. -Y. Duan, J. -Q. Li, and L. Zhang, An improved artificial bee colony algorithm with Q-learning for solving permutation flow-shop scheduling problems, *IEEE Trans. Syst. Man Cybern.*, vol. 53, no. 5, pp. 2684–2693, 2023.

[30] I. Sbai, S. Krichen, and O. Limam, Two meta-heuristics for solving the capacitated vehicle routing problem: The case of the Tunisian Post Office, *Oper. Res.*, vol. 22, no. 1, pp. 507–549, 2022.

[31] V. Singh and S. Choudhary, Genetic algorithm for traveling salesman problem: Using modified partially-mapped crossover operator, in *Proc. 2009 Int. Multimedia,*

Signal Processing and Communication Technologies, Aligarh, India, 2009, pp. 20–23.

[32] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions, *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, 2006.

[33] S. Hougardy, F. Zaiser, and X. Zhong, The approximation ratio of the 2-opt heuristic for the metric traveling salesman problem, *Oper. Res. Lett.*, vol. 48, no. 4, pp. 401–404, 2020.

[34] Y. Pan, K. Gao, Z. Li, and N. Wu, Solving biobjective distributed flow-shop scheduling problems with lot-streaming using an improved jaya algorithm, *IEEE Trans. Cybern*., doi: 10.1109/TCYB.2022.3164165.

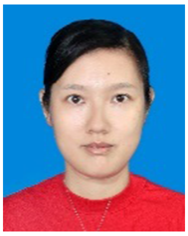
[35] J. Chen, B. Dan, and J. Shi, A variable neighborhood search approach for the multi-compartment vehicle routing problem with time windows considering carbon emission, *J. Clean. Prod.*, vol. 277, p. 123932, 2020.

[36] L. Zhen, C. Ma, K. Wang, L. Xiao, and W. Zhang, Multi-depot multi-trip vehicle routing problem with time windows and release dates, *Transportation Research Part E: Logistics and Transportation Review*, vol. 135, p. 101866, 2020.

[37] Y. Sun, D. Wang, M. Lang, and X. Zhou, Solving the time-dependent multi-trip vehicle routing problem with time windows and an improved travel speed model by a hybrid solution algorithm, *Clust. Comput.*, vol. 22, no. 6, pp. 15459–15470, 2019.

[38] E. B. Tirkolaee, M. Alinaghian, A. A. R. Hosseinabadi, M. B. Sasi, and A. K. Sangaiah, An improved ant colony optimization for the multi-trip capacitated arc routing problem, *Comput. Electr. Eng.*, vol. 77, pp. 457–470, 2019.

[39] E. B. Tirkolaee, A. A. R. Hosseinabadi, M. Soltani, A. K. Sangaiah, and J. Wang, A hybrid genetic algorithm for multi-trip green capacitated arc routing problem in the scope of urban services, *Sustainability*, vol. 10, no. 5, p. 1366, 2018.



Xiaoning Shen received the BS and PhD degrees from Nanjing University of Science and Technology in 2003 and 2008, respectively. She is currently a professor at the School of Automation, Nanjing University of Information Science and Technology, Nanjing, China. Her research interests include computational intelligence and multi-objective optimization and scheduling. She has published over 50 academic papers in the area of evolutionary algorithms and their applications.

Hongli Pan received the BS degree from Wanjiang University of Technology in 2019 and the master degree from Nanjing University of Information Science and Technology, Nanjing, China in 2022. Her research interest includes swarm intelligence algorithms and its applications.

Shuo Wang received the PhD degree in computer science from University of Birmingham in 2011, sponsored by the Overseas Research Students Award (ORSAS) from the British Government. She is an associate professor at the School of Computer Science, University of Birmingham (UoB), UK. She was previously an assistant professor and a research fellow at the Centre of Excellence for Research in Computational Intelligence and Applications (CERCIA), UoB. Her research interests include data stream classification, class imbalance learning, and ensemble learning approaches in machine learning, and their applications in social media analysis, software engineering, and fault detection. Her work has been published in internationally renowned journals and conferences, such as *IEEE Transactions on Knowledge and Data Engineering* and International Joint Conference on Artificial Intelligence (IJCAI).

Wenyan Chen received the BS degree from Xu Hai College of China University of Mining and Technology in 2022. She is currently pursuing the master degree at Nanjing University of Information Science and Technology, Nanjing, China. Her research interests include hyperheuristic algorithm and its application in software project scheduling.

Liyan Song received the BEng and MS degrees from Harbin Institute of Technology in 2009 and 2011, respectively. She received the PhD degree from University of Birmingham (UK), working on SEE using machine learning methods. She is currently a research assistant professor (RAP) at Southern University of Science and Technology (SUSTech). Her main research interests focus on predictive software engineering including software defect prediction and software effort estimation using machine learning techniques, classical machine learning (such as Bayesian model, unsupervised machine learning, online learning, and class imbalance), and deep learning. She has published high quality papers in *TOSEM*, *FSE*, and *ICSE* in the area of software engineering and *ACML* and *PAKDD* in the area of machine learning.

Zhongpei Ge received the BS degree from Guangzhou Huali College in 2020. He is currently pursuing the master degree at Nanjing University of Information Science and Technology, Nanjing, China. He is currently engaged in the research of swarm intelligence computing and its application.