



# A mathematical method for solving multi-depot vehicle routing problem

Fang wan<sup>1</sup> · Haixiang Guo<sup>1,2,3</sup> · Wenwen Pan<sup>1</sup> · Jundong Hou<sup>1,2</sup> · Shengli Chen<sup>3</sup>

Accepted: 11 June 2023 / Published online: 9 July 2023

© The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2023

## Abstract

Because logistics companies usually have multiple depots to serve their many dispersed customers, multi-depot vehicle routing problem (MDVRP) has gained significant research attention. To solve an MDVRP model, this paper develops a hybrid ant colony optimization based on a polygonal circumcenter (BPC-HACO). Furthermore, because ACO has been found to fall easily into the local optimum, simulated annealing and three local optimization operations are introduced to encourage the ACO to improve the algorithm's optimization ability. Finally, MDVRP benchmarks and data sets of other papers are employed to verify the effectiveness of the BPC-HACO in solving MDVRP (In 23 instances, BPC-HACO finds 14 BSKs, and 3 results are better than the BSKs), MDVRP with distance constraints (Compared to other papers, the route length is reduced by an average of 17.94%) and dynamic MDVRP (In 10 instances, BPC-HACO finds 3 BSKs, and 2 results are better than the BSKs). Finally, fitness landscape analysis has been applied to analyze the structural features of MDVRP to choose the most appropriate algorithm for MDVRP.

**Keywords** Multi-depot vehicle routing problem · Polygonal circumcenter · Ant colony optimization · Simulated annealing

## 1 Introduction

Logistics distribution impacts all enterprises as it is vital to service quality, transport costs, and revenue. The multi-depot vehicle routing problem (MDVRP) is an extension of VRP. Classic VRP has only one depot, while MDVRP includes multiple depots. The MDVRP has been applied to practical problems. For example, Wang et al. (2023) studied a real-world logistics distribution network with 4 depots and 160 customers. Soeanu et al. (2020) studied a small-scale risk-constrained multi-depot vehicle routing problem with 16 nodes and 3 depots. Kronmueller et al.

(2023) investigated a grocery delivery in Amsterdam with 20 depots and 10,000 orders.

The MDVRP is a typical NP-hard problem, the complexity and significance of which has attracted significant research attention.

Current solutions to multi-depot vehicle routing problems (MDVRP) can be roughly divided into 4 categories.

(1) The part method clusters customers with the depot as the center and then optimizes the routes for each depot. That is, turn the multi-depot VRP into single-depot VRP, as shown in Fig. 1a. Chen et al. (2023) and Xue et al. (2023) first assigned each customer to its nearest depot, after which hybrid algorithms were introduced to optimize the vehicle routes. Kim et al. (2023) combined ant colony optimization with a k-means clustering algorithm to solve garbage collection on a large scale.

(2) The overall method is establishing a virtual center so all vehicles start from the virtual center. That is, turn MDVRP into MDVRP with a virtual center (V-MDVRP), as shown in Fig. 1b. Yu et al. (2011) established a virtual center, turned an MDVRP into a V-MDVRP, and proposed an improved ACO to solve the V-MDVRP. Yan et al. (2017) analyzed the advantages and disadvantages of the

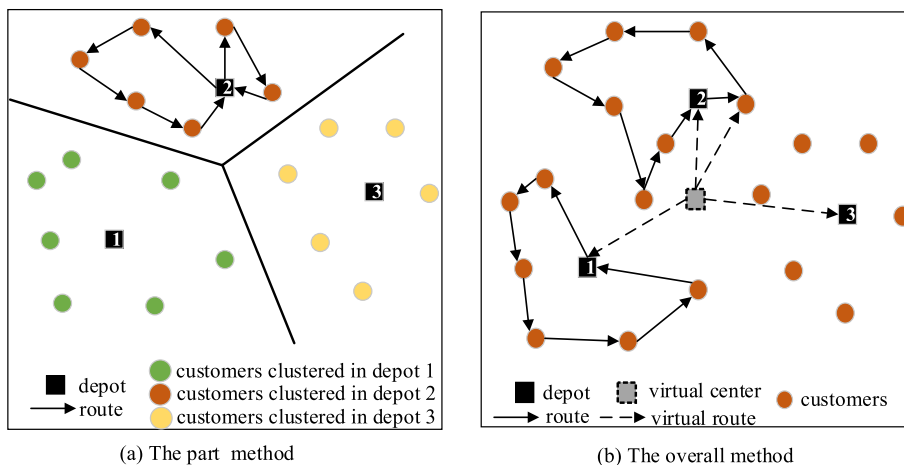
✉ Fang wan  
1140293340@qq.com

<sup>1</sup> School of Economics and Management, China University of Geosciences, Wuhan, People's Republic of China

<sup>2</sup> Mineral Resource Strategy and Policy Research Center of China University of Geosciences, Wuhan, People's Republic of China

<sup>3</sup> School of Management, Xi'an University of Finance and Economics, Xi'an, China

**Fig. 1** The part and overall method for MDVRP



part method, the overall method, and heuristic algorithms in solving MDVRP. Then, they combined the overall method and a heuristic algorithm to solve the MDVRP. In the overall method, there is no standard for setting the location of the virtual center; although it becomes a single depot problem, it still needs to assign the route to the nearest depot according to the distance.

(3) The coding method expresses the corresponding position of the depot and the customer using different coding methods and then designs the algorithm to solve the MDVRP, as shown in Fig. 2. Azuero-Ortiz et al. (2023) and Ge et al. (2023) adopted coding methods and hybrid algorithms to solve MDVRP. Lavigne et al. (2023) provided a Memetic Algorithm with a Sequential Split procedure (MASS) for solving a real-life waste collection problem in Brussels with multiple depots.

The part method optimizes the routes in each part but does not optimize the whole, and while the overall method considers all customers every time and seeks to optimize the whole, it does not optimize the part. Further, it isn't easy to design suitable and efficient coding methods for practical MDVRP.

Regardless of the overall method, the part method, or the coding method, the distance-based clustering method will be involved in selecting the depot. To avoid the distance-based clustering method, the virtual center is set at the circumcenter of the polygon formed by the depot. The algorithm can adaptively complete the selection of the

virtual center to the depot without considering the influence of distance.

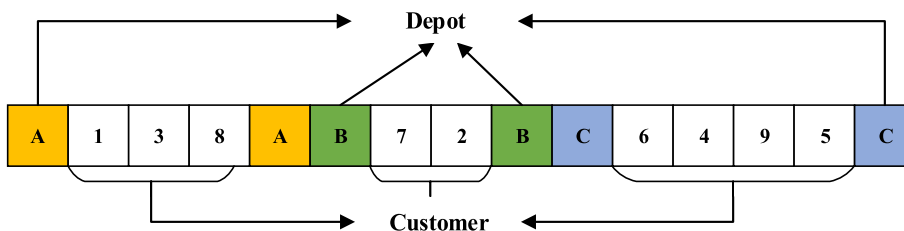
(4) In other methods, Wang et al. (2021) proposed a GA based on column generation to solve multi-depot electric VRP, which first generates a set of columns (each referring to a route), and then uses the GA to select a subset of columns to construct the final solution. Wang et al. (2022) propose a branch-and-price (BAP) algorithm for solving multi-depot GVRP with time windows to reduce the total carbon emissions, but it is very time-consuming in solving large-scale problems.

Few kinds of the literature integrate algorithm principles and mathematical ideas to solve MDVRP. For example, Yücenur et al. (2011) proposed a geometric shape (circle)-based genetic clustering algorithm for solving MDVRP. Draw a circle with the depot as the center, and assign the customers in the circle to the current depot. ACO has many advantages, such as excellent global search abilities, distributed computing, and easy combination with other algorithms, all of which can contribute to solving MDVRPs (Londoñoa et al. (2023); Zheng et al. (2023)).

This paper makes three major contributions:

1. Combines the advantages of the part and overall method, integrating ACO principle and mathematical ideas, then proposes the BPC-HACO to solve the MDVRP.

**Fig. 2** The coding method for MDVRP



2. MDVRP benchmarks and data sets from other papers are employed to verify the effectiveness of the BPC-HACO in solving various types of MDVRP.
3. FLA has been applied to analyze the structural features of MDVRP to choose the appropriate algorithm.

The remainder of this paper is structured as follows. Section 2 describes the DMDVRP model and defines the problem, Sect. 3 details the hybrid BPC-HACO, Sect. 4 proves the effectiveness of the BPC-HACO algorithm, and Sect. 5 concludes the paper and outlines future research directions.

## 2 MDVRP model

MDVRP includes multiple depots and multiple vehicles in each depot. It is assumed that each vehicle starts its travel from a depot, and upon completion of service to customers, it has to return to the depot. The notations and the mathematical model are as follows:

Sets:  $i = 1, 2, \dots, I$ : Set of all depots  $j = 1, 2, \dots, J$ : Set of all customers  $k = 1, 2, \dots, K$ : Set of all vehicles

Parameters:

$C_{ij}$ : Distance between points  $i$  and  $j$ ,  $i, j \in I \cup J$ .

$V_i$ : Maximum throughput at the depot  $i$ ,  $d_j$ : Demand of customer  $j$ ,

$Q_k$ : Capacity of vehicle  $k$ ,

Decision variables:

$$x_{ijk} = \begin{cases} 1, & \text{when vehicle } k \text{ serves point } j \text{ after } i \\ 0, & \text{otherwise} \end{cases}$$

Mathematical model:

$$\min \sum_{i \in I \cup J} \sum_{j \in I \cup J} \sum_{k \in K} C_{ij} x_{ijk} \tag{1}$$

$$\sum_{k \in K} \sum_{i \in I \cup J} x_{ijk} = 1, j \in J \tag{2}$$

$$\sum_{j \in J} d_j \sum_{i \in I \cup J} x_{ijk} \leq Q_k, k \in K \tag{3}$$

$$\sum_{i \in I \cup J} x_{ijk} - \sum_{i \in I \cup J} x_{jik} = 0, k \in K, j \in I \cup J \tag{4}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} \leq 1, k \in K \tag{5}$$

$$\sum_{j \in J} d_j * x_{ijk} \leq V_i, i \in I \tag{6}$$

$$\sum_{i \in J} \sum_{j \in J} x_{ijk} \leq |J| - 1, k \in K \tag{7}$$

Equation (1) minimizes the total cost. Each customer has to be assigned a single route according to Eq. (2). The capacity constraint for a set of vehicles is given by Eq. (3). Equation (4) shows the new sub-tour elimination constraint. The flow conservation constraints are expressed in Eq. (5). Each route can be served almost once according to

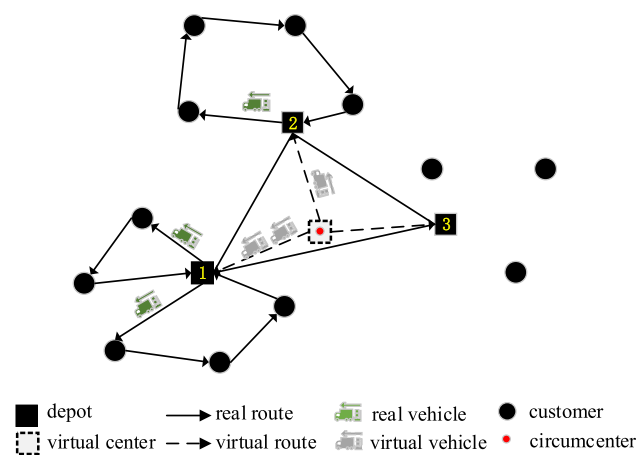


Fig. 3 An example of MDVRP with three depots

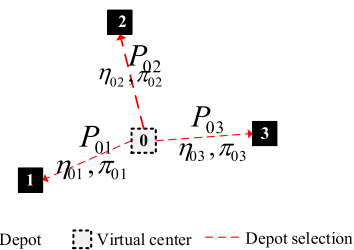


Fig. 4 “Virtual center-Depot” selection method

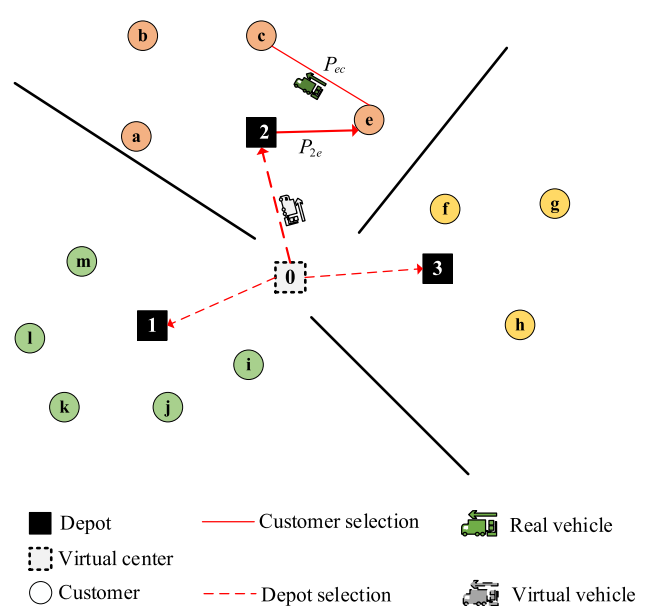


Fig. 5 “Depot-Customer” selection method

Eq. (6). The capacity constraints for the depots are given in Eq. (7).

### 3 Hybrid ant colony optimization based on polygonal circumcenter (BPC-HACO)

ACO has some excellent properties, such as strong search ability, easy parallel operations, and robustness. However, it also has some disadvantages, such as slow convergence and falling easily into a local optimum (Wu et al. (2023); Ren et al. (2023)). The simulated annealing (SA) algorithm, which is an intelligent algorithm developed in the 1980s, has certain advantages for local searches as it uses the Metropolis criterion to ensure that solutions with better fitness are accepted or that solutions with poor fitness are accepted with a certain probability (Vincent et al. (2023)). Both the ACO and SA have been widely used in optimization problems. For example, Li et al. (2019) adapted the ACO to solve a multi-depot, multi-objective vehicle routing problem and proved that the algorithm performed well. However, the efficiency decreased when the algorithm was used to solve large-scale problems. Nia et al. (2023) and Wang et al. (2023) combined the advantages of the SA and ACO to enhance local search abilities. They proved that they could better solve larger-scale combinatorial optimization problems.

#### 3.1 Ant colony optimization

##### 3.1.1 Determine a virtual center

Suppose there are customers,  $M$  depots, and a virtual center at the circumcenter of a polygon formed by the depots, as shown in the MDVRP with three depots in Fig. 3. There is no vehicle in the virtual center, and the vehicle starts from the actual depot. For example, if the virtual center selects

```

//Notation
T: Current temperature
Tend: Termination temperature
M: Number of iterations at each temperature
W: Vehicle capacity
w: Vehicle load
p: Probability parameter
Alpha: Cooling index
//Execution
Input: Local optimum solution x and f(x) of ACO
While T>=Tend
  for k =1: M
    X ← "Select-Insert-Optimize-Delete" on x
    Calculate f(X) and vehicle load w for solution X
    If w<=W
      E=f(X)-f(x)
      If E<=0 or Exp(-E/T)>p
        x,f(x) ← X,f(X)
      End If
    End If
  End For
  Cool down: T=T*alpha
End While
Output: x and its fitness value f(x)
    
```

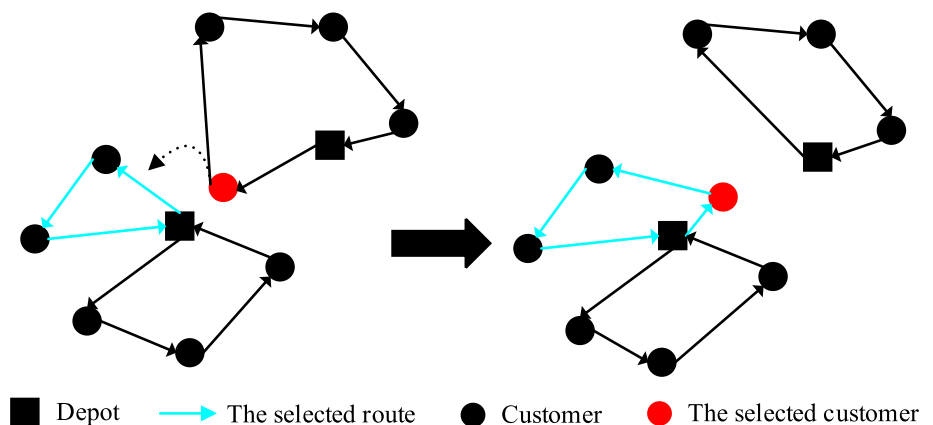
Fig. 7 SA algorithm pseudo-code

depot 1, then depot 1 schedules a vehicle to serve the corresponding customers.

##### 3.1.2 Initialize pheromone matrix and heuristic information

The initial pheromone matrix is set as a constant matrix  $(N + M + 1) \times (N + M + 1)$ . The heuristic information matrix is also set as  $(N + M + 1) \times (N + M + 1)$ , which is calculated using the distance between two customers.

Fig. 6 Neighborhood operation mechanism



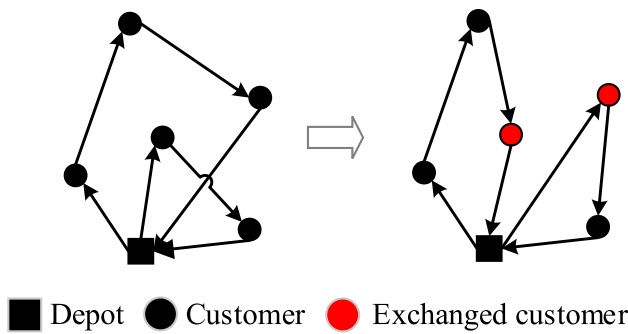


Fig. 8 Inter-route optimization

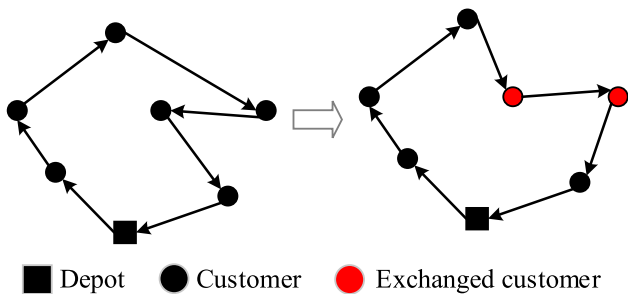


Fig. 9 Intra-route optimization

### 3.1.3 Select the depot

The virtual center is set at the circumcenter of a polygon formed by the depots to ensure that the distances between the virtual centers and the depots are equal (as Eq. (8)), which ensures that the heuristic information between the virtual center and different depots is equal (as Eqs. (9–10)). The probability ( $P_{0j}(j = 1, 2, 3)$ ) of the virtual center selecting the depot is only influenced by the pheromone (as Eqs. (11–12)), as shown in Fig. 4. With the increase in iterations, the adaptive selection of the depot is completed.

$$d_{01} = d_{02} = d_{03} \tag{8}$$

$$\eta_{0i} = \frac{1}{d_{0i}}, i = 1, 2, 3 \tag{9}$$

$$\eta_{01} = \eta_{02} = \eta_{03} = A \tag{10}$$

$$P_{0i}(t) = \frac{[\pi_{0i}(t)]^\alpha [\eta_{0i}]^\beta}{\sum_{i \in \text{depot}} [\pi_{0i}(t)]^\alpha [\eta_{0i}]^\beta}, i \in \{1, 2, 3\} \tag{11}$$

$$P_{0i}(t) = \frac{[\pi_{0i}(t)]^\alpha [A]^\beta}{\sum_{i \in \text{depot}} [\pi_{0i}(t)]^\alpha [A]^\beta} = \frac{[\pi_{0i}(t)]^\alpha}{\sum_{i \in \text{depot}} [\pi_{0i}(t)]^\alpha}, i \in \{1, 2, 3\} \tag{12}$$

### 3.1.4 Select the customer

After the depot is selected, the vehicle at the depot chosen selects customers (as Eq. (13)), which completes the “Depot-Customer” selection, then continues to select other customers (as Eq. (14)), as shown in Fig. 5.

As a whole, the method clusters the customers with the depots. As part, the method optimizes the customer combinations within each depot, which combines the overall method and part method advantages.

$$P_{2j}(t) = \begin{cases} \frac{[\pi_{2j}(t)]^\alpha [\eta_{2j}]^\beta}{\sum_{j \in \text{Customer}} [\pi_{2j}(t)]^\alpha [\eta_{2j}]^\beta}, & \text{if } j \in \{a, b, \dots, m\} \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

$$P_{ej}(t) = \begin{cases} \frac{[\pi_{ej}(t)]^\alpha [\eta_{ej}]^\beta}{\sum_{j \in \text{Customer}} [\pi_{ej}(t)]^\alpha [\eta_{ej}]^\beta}, & \text{if } j \in \{a, \dots, d, f, \dots, m\} \\ 0, & \text{otherwise} \end{cases} \tag{14}$$

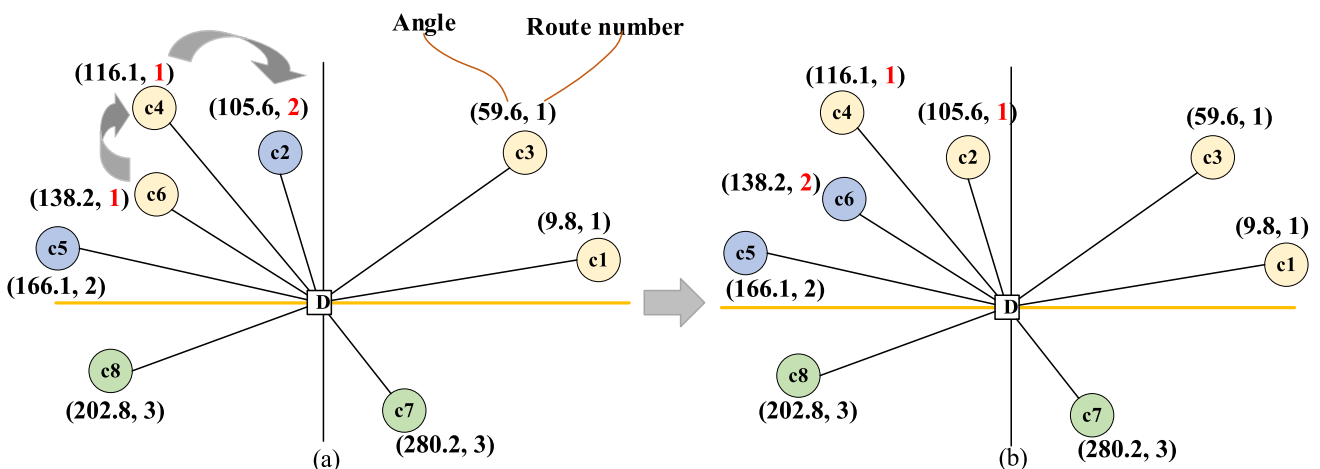
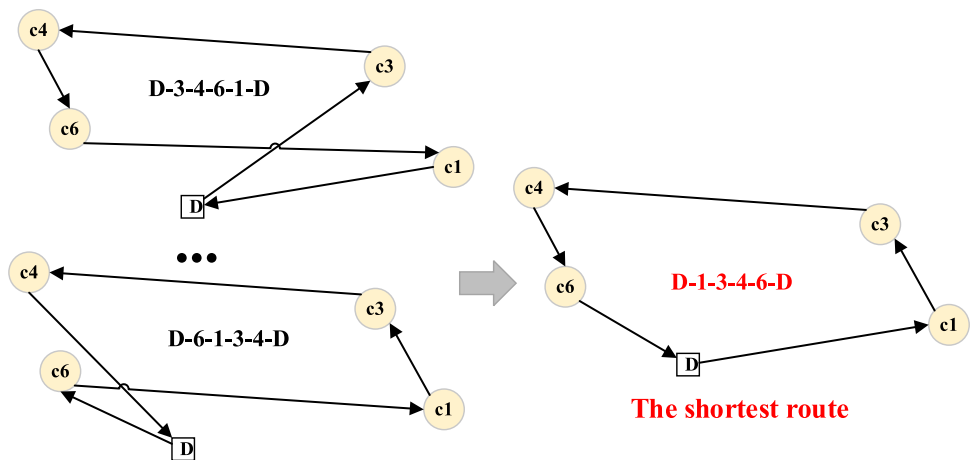
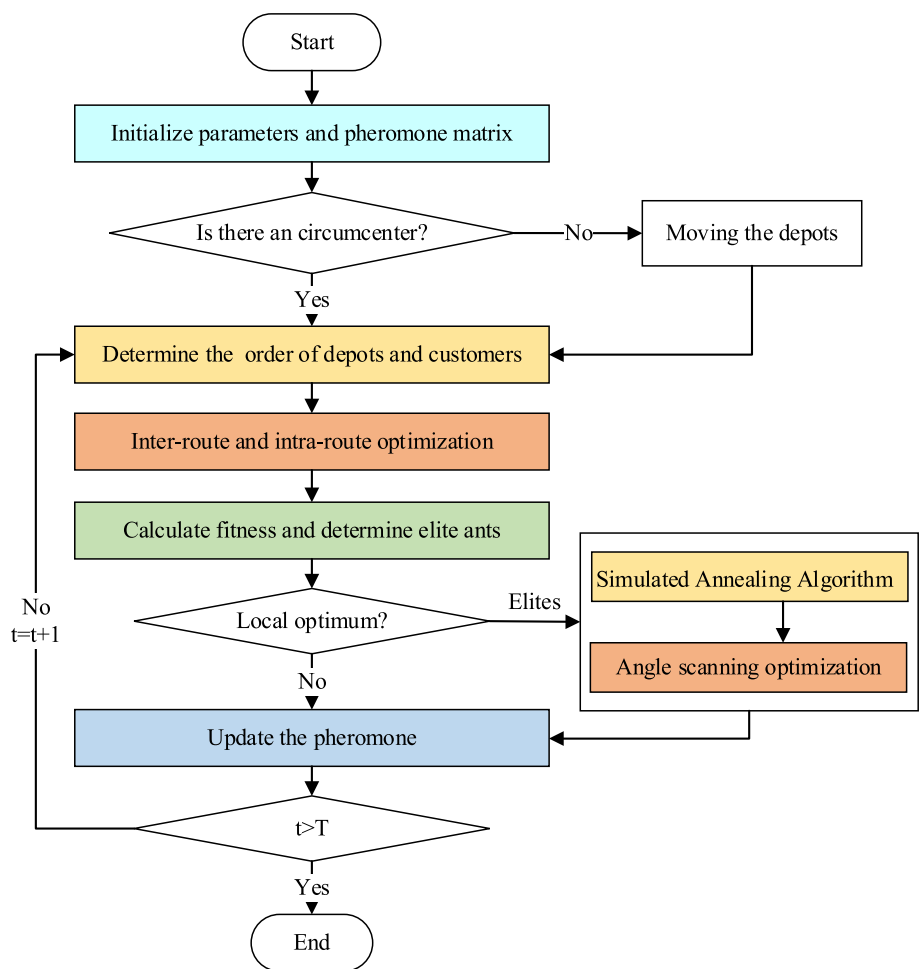


Fig. 10 Angle scanning optimization

**Fig. 11** The shortest route for customers 1, 3, 4, and 6



**Fig. 12** The steps of the BPC-HACO algorithm



### 3.1.5 Update the pheromone

After the routes are constructed, formula (15) updates all pheromones and is calculated using formula (16), where  $Q$

is a known constant, and  $\rho$  is the pheromone evaporation rate.

$$\pi_{ij}(t + 1) = (1 - \rho)\pi_{ij}(t) + \sum_{k=1}^m \Delta\pi_{ij}^k(t) \tag{15}$$

**Table 1** Detailed information for the instances

No	Customers	Depots	Capacity	Distance
1	50	4	80	–
2	50	4	160	–
3	75	5	140	–
4	100	2	100	–
5	100	2	200	–
6	100	3	100	–
7	100	4	100	–
8	249	2	500	–
9	249	3	500	–
10	249	4	500	–
11	249	5	500	–
12	80	2	60	–
13	80	2	60	200
14	80	2	60	180
15	160	4	60	–
16	160	4	60	200
17	160	4	60	180
18	240	6	60	–
19	240	6	60	200
20	240	6	60	180
21	360	9	60	–
22	360	9	60	200
23	360	9	60	180

### 3.2 Simulated annealing algorithm

The SA algorithm, which has strong local optimization ability, is introduced to optimize the local optimum solution for the ACO and encourage it to converge to the global optimum solution. The neighborhood operation mechanism in the SA algorithm is “Select-Insert-Optimize-Delete”; that is, it randomly selects a customer in one route, inserts them into another route, minimizes the increased length of the route, and then deletes the customer from the original route, as shown in Fig. 6. Figure 7 shows the pseudo-code for the SA algorithm.

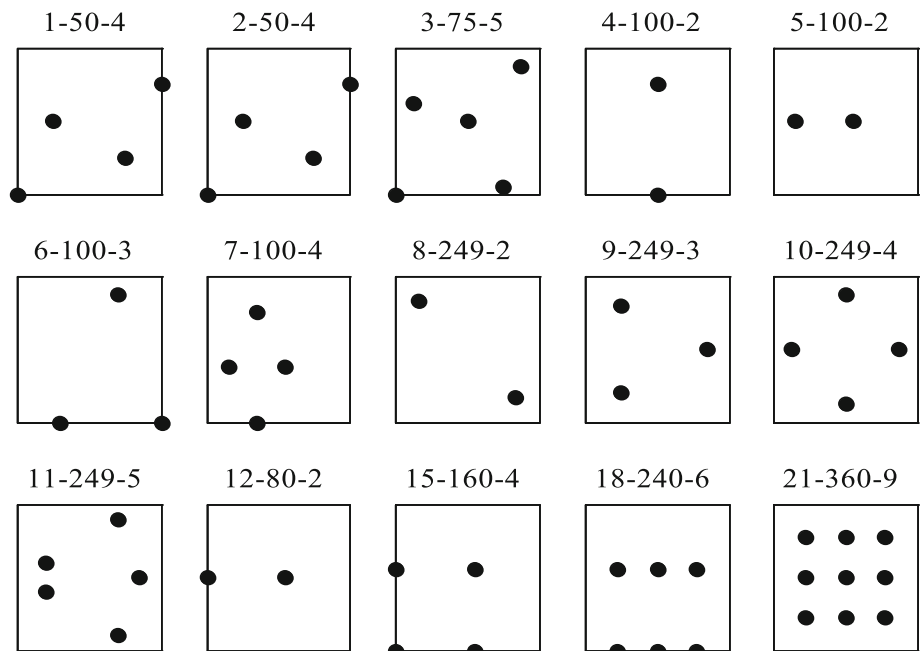
As the temperature decreases, the probability gradually decreases that the solutions with poor fitness are accepted, which guides the ant colony to evolve toward the global optimum.

### 3.3 Local interchange operation

2-Opt is a common, classical local optimization algorithm that usually involves two exchanges: exchanging customers on different routes and exchanging customers within the route (Manullang et al. (2023)). Three local optimization operations are adopted to avoid local optimization and obtain a better route: (1) inter-route optimization, (2) intra-route optimization, and (3) angle scanning optimization, which includes both inter-route and intra-route optimization (Asefi et al. (2019)).

$$\Delta\pi_{ij}^k(t) = \begin{cases} Q/fk(t), & \text{if link}(i,j) \text{ on } k\text{th route} \\ 0, & \text{otherwise} \end{cases} \quad (16)$$

**Fig. 13** Depot distribution before the movement of instances 1 to 21



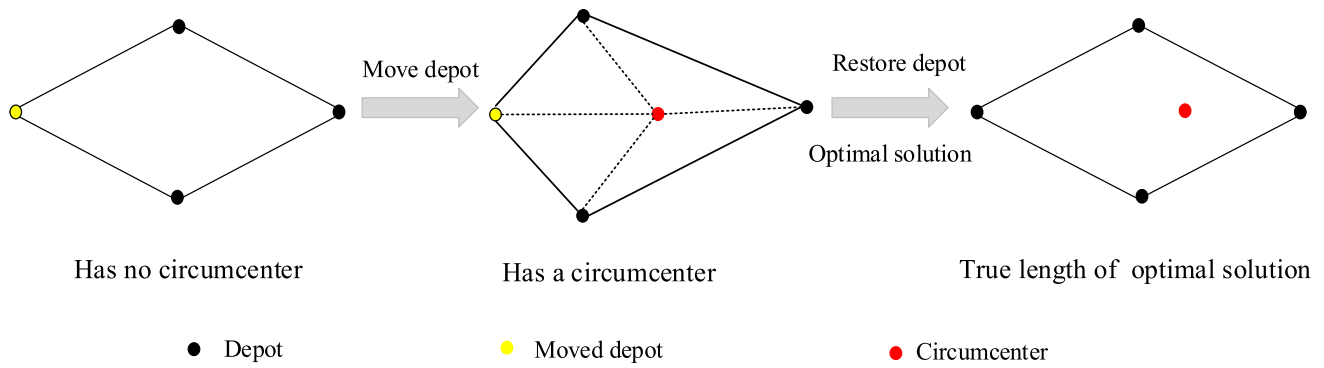


Fig. 14 Depot setting for the instances without a circumcenter

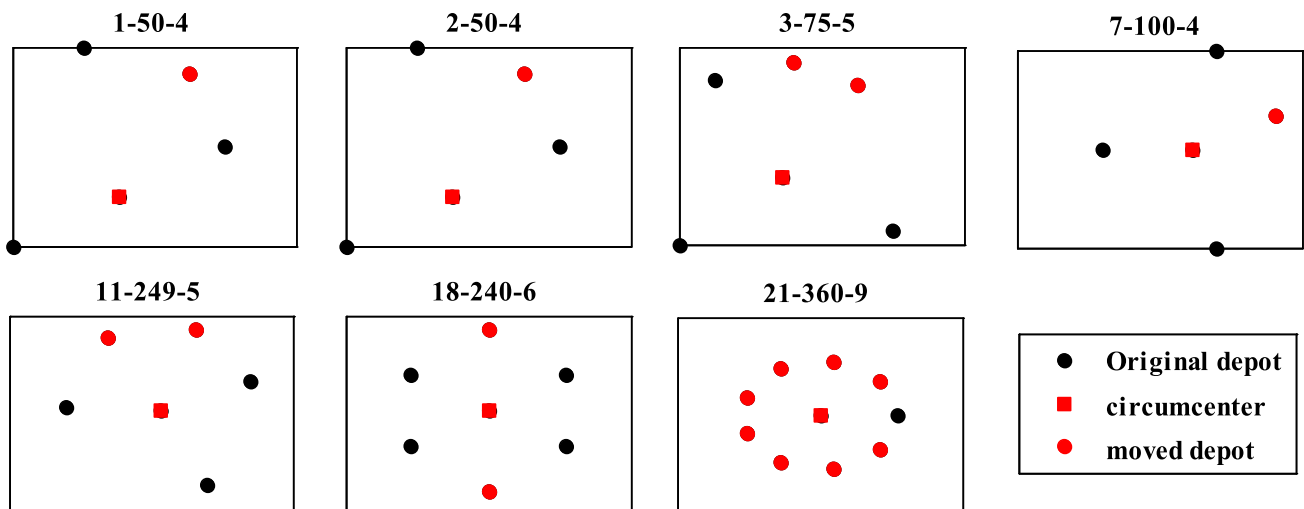


Fig. 15 Depot distribution after the movement of instances 1,2,3,7,11,18, and 21

Table 2 The parameters of the BPC-HACO algorithm

Algorithm	Parameter	Description	Value
ACO	$m$	Number of ants	100
	$\alpha$	Weight of pheromone	1
	$\beta$	Weight of visibility	5
	$\rho$	Pheromone evaporation rate	0.75
	$T$	Number of iterations	200
	$w$	Number of elite ants	3
	SA	$M$	Number of iterations at each temperature
$\alpha$		Cooling index	0.95
$T_{end}$		Termination temperature	1
$T_0$		Initial temperature	100

### 3.3.1 Inter-route optimization

To determine the optimal situation, inter-route optimization randomly selects two routes, exchanges the customers on the two routes, and ensures the vehicle capacity limit, as shown in Fig. 8.

### 3.3.2 Intra-route optimization

To determine the optimal situation, intra-route optimization randomly selects a route and exchanges two customers, as shown in Fig. 9.



**Table 3** Results from the 30 runs

No	BSK	Sbest	Savg	E_Sbest	E_Savg	SD
1	576.87	<b>576.87</b>	582.42	0.00%	0.96%	4.76
2	473.53	<b>473.53</b>	482.36	0.00%	1.87%	7.00
3	641.19	<b>641.19</b>	652.50	0.00%	1.76%	8.07
4	1001.59	1003.73	1010.91	0.21%	0.93%	7.26
5	750.03	<b>750.03</b>	760.06	0.00%	1.34%	9.42
6	876.50	<b>876.5</b>	885.24	0.00%	1.00%	8.31
7	885.80	<b>884.66</b>	894.31	- 0.13%	0.96%	6.08
8	4437.68	<b>4406.68</b>	4426.52	- 0.70%	- 0.25%	12.46
9	3900.22	<b>3897.6</b>	3908.78	- 0.07%	0.22%	9.37
10	3663.02	<b>3663.02</b>	3672.58	0.00%	0.26%	7.67
11	3554.18	3580.8	3590.26	0.75%	1.02%	9.37
12	1318.95	<b>1318.95</b>	1328.35	0.00%	0.71%	7.94
13	1318.95	<b>1318.95</b>	<b>1325.87</b>	0.00%	0.52%	6.84
14	1360.12	<b>1360.12</b>	<b>1370.69</b>	0.00%	0.78%	9.43
15	2505.42	<b>2505.42</b>	2513.26	0.00%	0.31%	5.65
16	2572.23	<b>2572.23</b>	<b>2591.40</b>	0.00%	0.75%	9.73
17	2709.09	2731.37	<b>2750.54</b>	0.82%	1.53%	10.68
18	3702.85	3741.99	3751.04	1.06%	1.30%	9.15
19	3827.06	3863.9	3901.37	0.96%	1.94%	12.14
20	4058.07	4097.06	4139.11	0.96%	2.00%	11.55
21	5474.84	5575.79	5596.32	1.84%	2.22%	11.54
22	5702.16	5718	5818.34	0.28%	2.04%	9.65
23	6095.46	6145.58	6283.17	0.82%	3.08%	14.23
Avg	2669.82	2682.78	2705.89	0.30%	1.18%	9.06

Bold values represent the same or better results than BSKs

### 3.3.3 Angle scanning optimization

If there is a crossover between two routes, the solution can be improved using local optimization to eliminate this crossover in the solution (Wu et al. (2023)). While enumeration can be used to identify the crossover, it reduces the efficiency and practicality of the algorithm, especially when trying to solve a large-scale VRP. Therefore, to improve the local optimization efficiency, angle scanning optimization can effectively reduce the crossover in the solution (Cai et al. (2014)), as shown in Fig. 10.

Figure 10 illustrates the coordinate system with the depot as the center, in which the right direction is the reference direction. The angle between all customers and the reference direction is calculated, and the route number they belong to is recorded. For example, in (59.6, 1), “59.6” represents the angle, and “1” represents the route number.

Then, it is necessary to judge whether the customer’s angle on each route becomes larger in turn; if not, it needs to be optimized. For example, the angles for customers 1, 3, 4, and 6 on route 1 are 9.8, 59.6, 116.1, and 138.2, all of

which become larger. Therefore, the shortest route for customers 1, 3, 4, and 6 is “D-1-3-4-6-D,” which reduces the crossovers between the routes, as shown in Fig. 11.

Then, it is necessary to judge whether the customer angles between the different routes become larger; if not, there is a route crossover. For example, the angle for customer 2 (105.6) on route 2 is smaller than that for customer 4 (116.1) on route 1, indicating a crossover between routes 1 and 2, and then exchanging customers 2 and 4. After the exchange, the angle of customer 4 (116.1) on route 2 is smaller than the angle of customer 6 (138.2) on route 1, and then exchange customers 4 and 6, as shown in Fig. 10b. After these two exchanges, as the routes in the depot do not cross, the routes are optimal.

### 3.4 BPC-HACO flowchart

The BPC-HACO flowchart is shown in Fig. 12.

## 4 Experimental results and discussion

This section first introduces some MDVRP benchmarks, then the algorithm’s running results, and some comparative experiments to verify the algorithm’s effectiveness in solving various types of MDVRP. Finally, FLA is performed on the MDVRP data set to know their structural features and select suitable algorithms for solving MDVRP.

### 4.1 Benchmark experiments 1

To illustrate the effectiveness of the BPC-HACO in solving an MDVRP, several experiments based on the benchmark are simulated for the instances available at <http://www.bernabe.dorronsoro.es/vrp/>, the detailed information for which is given in Table 1. The first column gives the instance name, the second and third columns show the number of customers and the number of depots, the fourth column indicates the vehicle capacity constraint, and the fifth column indicates the maximum vehicle delivery distance constraint, where “-” means no distance constraint.

Figure 13 shows the distribution of the original depots for instances 1–23, in which the subtitle “1–50–4” refers to instance 1, which has 50 customers and 4 depots. The depot locations for instances 12, 13 and 14 are the same; the depot locations for instances 15, 16 and 17 are the same; the depot locations for instances 18, 19 and 20 are the same; and the depot locations of instances 21,22 and 23 are the same.

From Fig. 13, it can be seen that instances 1, 2, 3, 7, 11, 18, and 21 do not have circumcenters, which are solved as follows. First, some depots are moved to ensure that the

**Table 4** Comparison results based on benchmark 1

No	BSK	Bezerra (2018)	Yao (2019)	Sadati (2020)	Zhang (2020)	Gu (2022)	Torres (2022)	BPC-HACO
1	576.87	582.34	<b>576.86</b>	<b>576.87</b>	<b>576.87</b>	595.47	646.86	<b>576.87</b>
2	473.53	473.87	<b>473.53</b>	<b>473.53</b>	<b>473.53</b>	482.19	535.49	<b>473.53</b>
3	641.19	<b>641.19</b>	<b>641.19</b>	<b>641.19</b>	<b>641.19</b>	692.72	699.44	<b>641.19</b>
4	1001.59	1008.66	<b>1001.49</b>	1008.62	<b>1010.57</b>	1127.58	1090.06	1003.73
5	750.03	752.97	750.26	752.04	751.15	837.5	835.47	<b>750.03</b>
6	876.5	878.02	<b>876.5</b>	882.71	880.57	971.13	958.3	<b>876.5</b>
7	885.8	890.46	<b>885.69</b>	896.01	<b>881.97</b>	913.17	–	<b>884.66</b>
8	4437.68	–	4482.44	<b>4417.34</b>	4516.75	5079.62	4611.03	<b>4406.68</b>
9	3900.22	–	3912.23	3940.56	3939.52	4901.76	4204.66	<b>3897.6</b>
10	3663.02	–	<b>3663</b>	3696.31	3724.93	4393.91	4003.74	<b>3663.02</b>
11	3554.18	–	3648.95	3578.14	3624.67	4203.38	3948.98	3580.8
12	1318.95	<b>1318.95</b>	<b>1318.95</b>	<b>1318.95</b>	<b>1318.95</b>	–	1651.02	<b>1318.95</b>
13	1318.95	–	<b>1318.95</b>	<b>1318.95</b>	<b>1318.95</b>	–	–	<b>1318.95</b>
14	1360.12	–	1365.68	<b>1360.12</b>	<b>1360.12</b>	–	–	<b>1360.12</b>
15	2505.42	2525.85	<b>2505.29</b>	2538.79	<b>2505.42</b>	–	<b>3113.4</b>	<b>2505.42</b>
16	2572.23	–	2587.87	<b>2572.23</b>	<b>2572.23</b>	–	–	<b>2572.23</b>
17	2709.09	–	<b>2708.99</b>	2731.37	<b>2709.09</b>	–	–	2731.37
18	3702.85	3796.04	3781.04	3798.58	3749.34	–	4623.47	3741.99
19	3827.06	–	<b>3827.06</b>	<b>3827.06</b>	<b>3827.06</b>	–	–	3863.9
20	4058.07	–	<b>4058.07</b>	4097.06	<b>4058.07</b>	–	–	4097.06
21	5474.84	–	<b>5474.84</b>	5643.55	5619.95	–	6743.06	5575.79
22	5702.16	–	<b>5702.06</b>	5708.36	<b>5702.16</b>	–	–	5718
23	6095.46	–	<b>6095.46</b>	6145.58	<b>6078.75</b>	–	–	6145.58
Avg	–	<b>0.61%</b>	<b>0.31%</b>	<b>0.64%</b>	<b>0.49%</b>	<b>11.78%</b>	<b>13.83%</b>	<b>0.30%</b>

polygon formed by these depots has a circumcenter, after which these depots are input to the BPC-HACO to determine the optimal solution. Finally, the original depots are reintroduced into the optimal solution to obtain the true route lengths, as shown in Fig. 14. Figure 15 is the depot distribution for instances 1, 2, 3, 7, 11, 18, 21 without circumcenters.

The parameters of the BPC-HACO algorithm are shown in Table 2. The algorithm is implemented in MATLAB R2014a, Windows 7 ( $\times$  64).

All instances are run 30 times, the result from which are shown in Table 3. The second column gives the Best Solution Known (BSK), the third column shows the best solution from all 30 runs, the fourth column gives the average solution from all 30 runs, and the fifth column shows the error between the best solution and the BSK, the sixth column shows the error between the average solution and the BSK, and the seventh column shows the standard deviation for the 30 runs. The  $E\_Sbest$ ,  $E\_Savg$ , and  $SD$  are determined using Eqs. (17)–(19).

$$E\_Sbest(\%) = (Sbest - BSK)/BSK \times 100\% \quad (17)$$

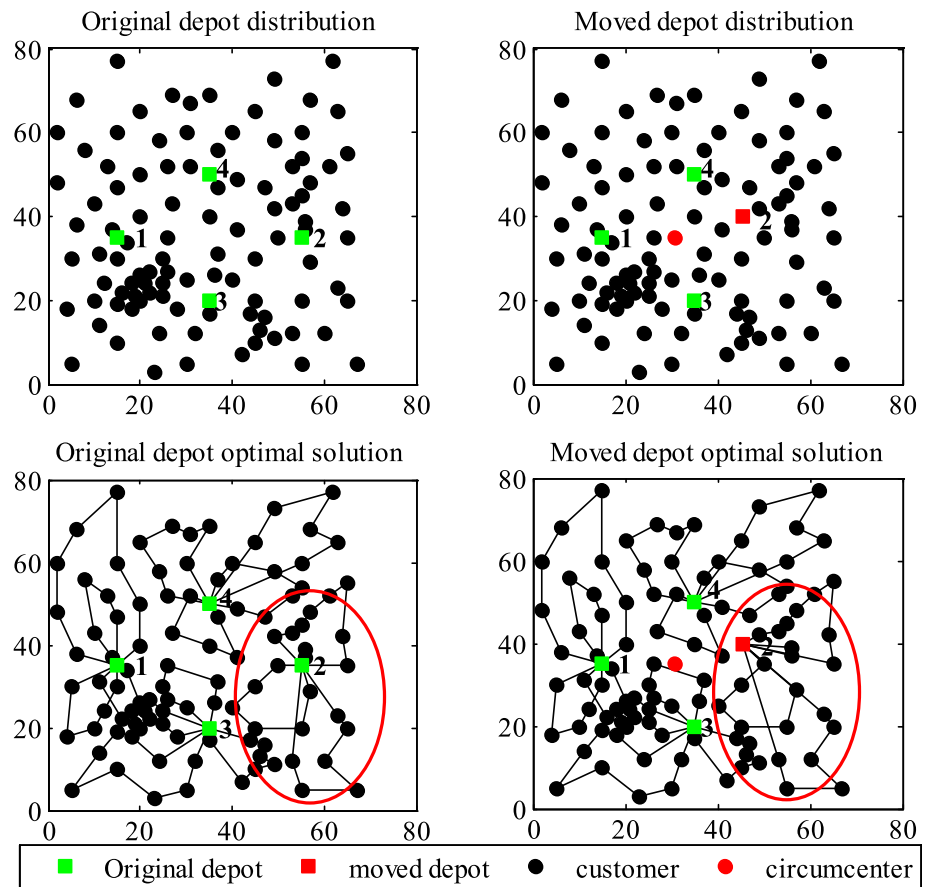
$$E\_Savg(\%) = (Savg - BSK)/BSK \times 100\% \quad (18)$$

$$SD = \sqrt{\frac{\sum_{i=1}^N (Si - Savg)^2}{N}} \quad (19)$$

In Table 4, the  $E\_Sbest$  for 23 instances is no greater than 2.0%, the  $E\_Savg$  is no greater than 3.1%, and the  $SD$  is no greater than 15, which indicates that the algorithm is stable and robust, and the algorithm performs balanced in each instance. The best solutions are compared with Bezerra et al. (2018); Yao et al.(2019); Sadati et al.(2020); Zhang et al.(2020); Gu et al.(2022); Torres et al. (2022), as shown in Table 4.

The bold numbers in Table 4 represent the number of BSKs found or the instances that are better than the BSK. The last row in the table shows the average  $E\_Sbest$  for the 23 instances. The BPC-HACO algorithm finds 14 optimal solutions; meanwhile, instances 7, 8, and 9 have better results than the BSK. The numbers of depots for instances 7, 8, and 9 are 4, 2, and 3, and the numbers of customers

**Fig. 16** Depot and optimal solution distribution for instance 7



are 100, 249 and 249, indicating that the method remains feasible when there are more or fewer customers and depots.

Instances 1, 2, 3, 7, 11, 18, and 21 need to move some depots, where instances 1, 2, and 3 find the BSK, and instance 7 finds a better result than the BSK. Figure 16 shows the distributions for the original depots and the moved depots and the distributions for the optimal solution in the original depots and the moved depots for instance 7. The distributions for the optimal solution differ only in depot 2 for the original depots and the moved depots. No difference is found between the routes for the other three depots. Therefore, the optimal solution is found by moving some depots. The errors for instances 11 and 18 are 0.75% and 1.06%, and the error for instance 21 is 1.84%, considered relatively large and mainly due to the significant changes in the depot position in instance 21.

## 4.2 Benchmark experiments 2

The performance of the BPC-HACO algorithm in solving MDVRP with distance constraints and the dynamic multi-depot vehicle routing problem (DMDVRP) is evaluated using a series of experiments based on another benchmark,

which is available at <http://neo.lcc.uma.es/vrp/vrpinstances/multiple-depot-vrp-instances/>, the detailed information for which is shown in Table 5.

### 4.2.1 BPC-HACO for solving DMDVRP

DMDVRP is an extension of MDVRP. In MDVRP, all customers are known. In DMDVRP, some customers are known (known customers). Other customers dynamically appear during the service period (new customers), and the objective is to find the shortest route for both the known and new customers (Hu et al.(2023)). The performance of the BPC-HACO algorithm solving for DMDVRP is evaluated based on instances 1–13 in Table 5; the results are compared with Yu et al. (2013) and Xu et al. (2018), as shown in Table 6.

Yu et al. (2013) introduced a distance-based clustering algorithm to cluster each customer to the nearest depot. To plan the routes, they proposed an improved ant colony optimization (IACO) with ant colony weight and mutation operations. Xu et al. (2018) used K-means to cluster the customers and introduced a hybrid ant colony optimization (HACO) with mutation operations and local optimization. Both papers randomly selected 30% of the customers to be

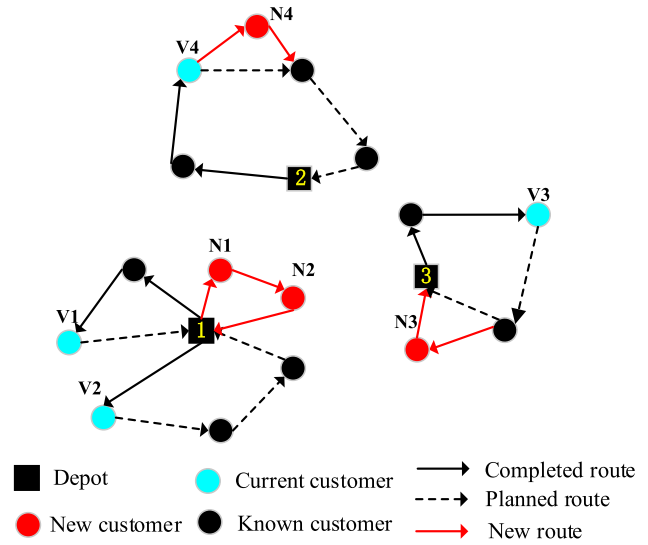
**Table 5** Detailed information on the instances

No	Depots	Customers	Capacity	Distance
1	4	50	80	–
2	4	50	160	–
3	5	75	140	–
4	2	100	100	–
5	2	100	200	–
6	3	100	100	–
7	4	100	100	–
8	2	249	500	–
9	3	249	500	–
10	4	249	500	–
11	5	249	500	–
12	2	80	60	–
13	4	160	60	–
R1	4	48	200	500
R2	4	96	195	480
R3	4	144	190	460
R4	4	192	185	440
R5	4	240	180	420
R6	4	288	175	400
R7	6	72	200	500
R8	6	144	190	475
R9	6	216	180	450
R10	6	288	170	426

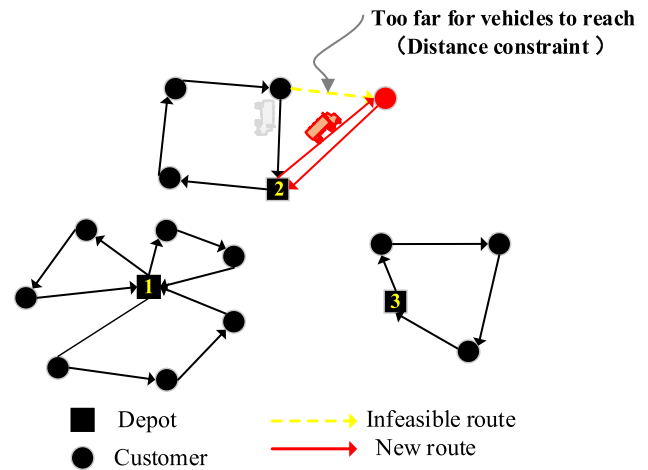
**Table 6** The comparison results of DMDVRP

No	IACO		HACO		BPC-HACO	
	Length	Time(s)	Length	Time(s)	Length	Time(s)
1	686.46	56	811.72	60.45	<b>634.373</b>	50.07
2	568.24	59	599.6	46.62	<b>526.872</b>	52.31
3	772.62	109	1033.07	89.86	<b>748.674</b>	75.10
4	1158.77	124	1482	158.92	<b>1144.6</b>	129.42
5	865.23	146	1255.38	130.16	<b>863.734</b>	134.55
6	1043.04	132	1274.6	137.51	<b>1025.7</b>	130.94
7	1053.97	148	1292.97	123.6	<b>1052.6</b>	139.59
8	5147.59	678	7873.47	599.77	<b>4683.78</b>	372.85
9	4543.65	602	6279.87	453.87	<b>4178.51</b>	365.25
10	4267.4	598	6076.75	384.96	<b>3986.5</b>	366.96
11	4186.71	611	6101.29	366.24	<b>3851.6</b>	370.46
12	1529.98	93	1524.62	106.05	<b>1508.9</b>	93.52
13	2936.2	306	3100.88	214.5	<b>2874.9</b>	185.57
Avg	<b>2212.3</b>	<b>281.692</b>	<b>2977.4</b>	<b>220.962</b>	<b>2083.13</b>	189.73

Bold values represent the same or better results than other algorithms



**Fig. 17** An example of DMDVRP



**Fig. 18** An example of MDVRP with distance constraints

**Table 7** The comparison results of MDVRP with distance constraints

No	BSK	Sbest	Savg	E_Sbest	E_Savg	T(S)
R1	861.32	<b>861.32</b>	876.85	0.00%	1.80%	27.55
R2	1307.61	<b>1307.61</b>	1310.68	0.00%	0.24%	65.24
R3	1806.6	1891.3	1908.71	4.69%	5.65%	88.94
R4	2072.52	2125.63	2150.85	2.56%	3.78%	127.62
R5	2385.77	<b>2372.67</b>	2400.33	-0.55%	0.61%	177.36
R6	2723.27	2799.73	2812.27	2.81%	3.27%	212.64
R7	1089.56	<b>1089.56</b>	1099.33	0.00%	0.90%	45.51
R8	1666.6	1716.9	1717.21	3.02%	3.04%	97.58
R9	2153.1	2238.31	2264.16	3.96%	5.16%	152.75
R10	2921.85	<b>2902.59</b>	2915.48	-0.66%	-0.22%	225.14
Avg	1898.82	1930.562	1945.59	1.58%	2.42%	122.03

Bold values represent the same or better results than BSKs

**Table 8** The comparison results of small-scale MDVRP

Data sets	Customer	Depot	Capacity	Length	Time(s)	BPC-HACO		
						Length	Route	Time(s)
Wang et al. (2012)	25	3	20	67.44	–	<b>66.14</b>	1,3,17,10,27,3,1 1,3,24,14,19,12,3,1 1,3,6,20,28,15,23,3,1 1,4,9,25,21,4,1 1,2,5,13,26,2,1 1,4,11,16,7,4,1 1,2,29,18,22,8,2,1	4.15
Luo et al. (2014)	25	3	20	65.56	–	<b>65.56</b>	1,3,25,18,10,6,3,1 1,2,12,11,28,2,1 1,4,13,16,17,4,1 1,3,21,23,22,27,8,3,1 1,4,14,7,29,4,1 1,3,9,26,15,3,1 1,2,19,5,20,24,2,1	4.29
Ma et al. (2014)	30	3	4,6,8	1464.4	6.55	<b>756.07</b>	1,2,15,33,32,17,2,1 1,3,25,21,3,1 1,4,8,14,11,4,1 1,3,31,9,16,29,20,3,1 1,3,23,24,27,28,6,13,3,1 1,4,34,18,26,4,1 1,3,30,22,7,3,1 1,3,5,12,19,10,3,1	5.07
Yan et al. (2017)	15	3	5	475.67	–	<b>430.20</b>	1,2,8,14,7,9,2, 1 1,2,15,11,13,2, 1 1,4,17,16,19,4, 1 1,3,10,18,3, 1 1,3,12,6,5,3,1	1.38
Zhang et al. (2009)	15	4	4	<b>458.21</b>	1.5	458.73	1,4,17,16,19,4,1 1,2,13,11,15,2,1 1,2,18,10,8,2,1 1,2,14,7,9,2,1 1,3,5,6,12,3,1	1.41
Ling et al. (2017)	24	3	4	641.3	–	<b>565.37</b>	1,2,7,22,13,2,1 1,3,6,28,5,3,1 1,3,17,25,3,1 1,4,24,19,10,4,1 1,4,23,18,9,4,1 1,3,16,26,21,12,3,1 1,2,27,14,11,20,15,2,1 1,4,8,4,1	4.13

Bold values represent the same or better results than other algorithms

new customers and then used “The nearest addition method” to deal with the new customers, as shown in Fig. 17. After the new customers (N1, N2, N3, N4) appear, the distances are calculated between the new customers and

the customers (V1, V2, V3, V4) that are being or have been served. Then, the new customers (N3, N4) are added to the nearest route, on which customers V3 and V4 are located. While N1 and N2 are close to V1 and V2, the vehicles in

**Table 9** Information and Statistical measures for 23 instances of landscapes

NO	$H(\epsilon)$		$M(\epsilon)$		$\epsilon^*$		$rFD$		$\tau$		$\delta$	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std	Mean	Std
1	0.28	0.001	0.33	0.005	391.91	9.58	0.006	0.003	0.35	0.006	62.17	3.40
2	0.26	0.009	0.30	0.004	384.32	7.70	0.040	0.004	0.26	0.008	64.64	2.85
3	0.32	0.010	0.35	0.009	408.56	10.49	-0.040	0.000	0.28	0.002	75.88	4.04
4	0.31	0.003	0.40	0.005	636.33	8.87	0.013	0.004	0.35	0.006	87.35	4.67
5	0.30	0.007	0.37	0.002	600.76	10.20	0.015	0.004	0.33	0.002	85.59	4.05
6	0.32	0.001	0.39	0.005	645.64	11.40	0.012	0.005	0.31	0.008	94.69	3.36
7	0.42	0.007	0.38	0.000	700.48	9.69	-0.023	0.002	0.28	0.001	99.10	4.99
8	0.49	0.006	0.40	0.010	883.84	10.53	0.011	0.003	0.38	0.005	124.20	4.07
9	0.50	0.009	0.45	0.001	897.67	12.34	-0.003	0.005	0.33	0.005	135.71	4.01
10	0.52	0.002	0.45	0.009	906.66	11.90	-0.014	0.002	0.28	0.003	137.13	3.66
11	0.55	0.005	0.43	0.005	920.76	12.16	-0.048	0.002	0.26	0.007	147.32	3.41
12	0.21	0.003	0.27	0.004	404.17	7.53	0.013	0.005	0.31	0.008	75.24	2.49
13	0.20	0.004	0.29	0.010	412.63	9.50	0.016	0.001	0.34	0.003	78.42	3.68
14	0.19	0.001	0.28	0.002	434.11	8.22	0.017	0.002	0.36	0.003	70.44	3.78
15	0.41	0.007	0.48	0.007	749.52	9.04	0.037	0.003	0.32	0.001	112.19	4.06
16	0.43	0.002	0.40	0.003	718.48	8.71	-0.030	0.002	0.31	0.007	105.32	5.26
17	0.40	0.003	0.45	0.006	798.99	10.78	0.040	0.005	0.33	0.000	123.15	4.14
18	0.55	0.000	0.44	0.003	1020.77	11.07	-0.010	0.004	0.24	0.008	125.02	5.56
19	0.54	0.004	0.46	0.004	1134.03	10.94	0.029	0.004	0.27	0.007	122.48	4.46
20	0.56	0.002	0.41	0.004	1057.74	11.83	-0.047	0.002	0.19	0.004	134.03	5.53
21	0.63	0.009	0.42	0.000	1415.26	10.21	0.041	0.000	0.15	0.006	144.86	4.08
22	0.66	0.002	0.43	0.002	1529.08	12.48	-0.025	0.004	0.12	0.002	146.62	5.11
23	0.62	0.010	0.44	0.005	1479.71	11.59	-0.048	0.005	0.13	0.008	153.99	5.00

these routes have no remaining capacity. Therefore, depot 1 needed to arrange new vehicles to serve new customers N1 and N2.

As can be seen from Table 6, the BPC-HACO performs better, especially when the customer and depot numbers are relatively large (instances 8, 9, 10, and 11). This is because the BPC-HACO selects the depot and the customers based on probability. Then, the algorithm adaptively completes the “depot-customer-vehicle” selection as the number of iterations increases.

BPC-HACO regards the depot and the customer as a whole and then optimizes the whole. In contrast, the IACO and HACO first clustered the customers to the depot and then optimized the routes inside the depot. However, as previously discussed, optimizing the part does not optimize the whole. The number of ants and the number of iterations of the three algorithms are the same (100 iterations and 30 ants). When the number of customers or depots is large, the time is shorter because BPC-HACO does not need to select the depot by other means, such as clustering, thus saving time.

#### 4.2.2 BPC-HACO for solving MDVRP with distance constraints

MDVRP with distance constraints has one more vehicle constraint than MDVRP. Therefore, as well as considering vehicle capacity, it is also necessary to consider a vehicle travel distance less than the maximum distance constraint (Yuan et al.(2020)), as shown in Fig. 18.

To demonstrate that the BPC-HACO is also effective in solving MDVRP with distance constraints, experiments are conducted based on instances R1-R10 in Table 5, after which the results are compared with the Best Solution Known (BSK), as shown in Table 7. The third column shows the best solution over the 30 runs, and the fourth column shows the average solution for the 30 runs, which are determined using Eqs. (12)–(14), and the seventh column “T” is the average running time (seconds).

The E\_Sbest for the 10 instances is not greater than 5.0%, the E\_Savg is not greater than 6.0%, and the T is not greater than 230 s. The BPC-HACO is found to be able to effectively solve the MDVRP with distance constraints within a reasonable running time. The E\_Sbest for instances R5 (240 customers, 4 depots) and R10 (288 customers, 6 depots) are, respectively, -0.55% and -

**Table 10** Results of Best, Mean and Std. for 5 optimization algorithms

No	ACO			GA			PSO			DE			TS		
	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std	Best	Mean	Std
1	855.93	1019.16	5.17	876.40	906.77	10.72	884.02	1016.46	13.27	600.03	891.73	11.12	732.11	844.62	9.59
2	<u>521.28</u>	<b>578.30</b>	14.12	737.39	786.92	14.29	612.87	751.64	16.19	526.83	<b>603.99</b>	15.95	642.29	683.04	15.82
3	<u>644.01</u>	<b>826.49</b>	11.06	785.48	977.81	12.38	790.36	1041.29	9.74	979.18	1106.35	9.51	784.23	<b>843.76</b>	10.55
4	<u>1106.48</u>	<b>1274.91</b>	12.69	1112.79	1390.61	12.75	1227.61	1357.21	14.52	1186.93	1350.53	9.99	1239.48	<b>1315.56</b>	13.31
5	852.34	<b>974.95</b>	10.74	1086.96	1168.83	16.19	936.25	1018.97	13.66	906.68	<b>934.96</b>	17.00	<u>767.53</u>	1025.48	10.91
6	1007.09	1141.62	13.79	893.86	1178.25	11.26	997.90	1287.95	17.58	947.95	<b>1097.64</b>	10.49	912.87	<b>1070.37</b>	11.52
7	<u>958.76</u>	<b>1007.72</b>	13.13	1095.60	1120.25	14.14	1050.03	<b>1095.16</b>	13.13	1105.78	1149.78	13.24	1097.95	1364.31	15.20
8	<u>4418.44</u>	<b>4516.53</b>	17.29	4424.39	4577.46	16.75	4574.51	4661.74	18.97	4562.49	<b>4574.96</b>	20.31	4669.47	4816.11	18.31
9	4196.69	4348.28	15.31	4071.11	<b>4161.55</b>	18.23	4209.64	4405.56	12.04	4101.02	4193.19	13.08	<u>3992.07</u>	<b>4184.84</b>	18.97
10	3858.29	4005.85	13.73	<u>3766.32</u>	<b>3778.10</b>	22.02	3932.82	<b>3932.98</b>	20.35	4006.17	4223.85	16.40	3903.36	4185.66	17.94
11	3781.47	<b>3819.09</b>	9.43	3927.44	4004.54	16.06	<u>3624.84</u>	<b>3667.95</b>	11.80	3740.79	4027.97	13.91	3719.36	3928.22	13.90
12	1623.51	1730.67	13.11	1514.24	1732.07	18.24	1497.80	<b>1646.47</b>	14.88	<u>1386.90</u>	<b>1609.66</b>	18.20	1590.79	<b>1625.66</b>	10.47
13	1452.02	<b>1605.41</b>	10.05	1627.78	1834.94	13.41	1473.95	<b>1639.21</b>	10.87	1605.18	1767.11	13.91	1544.87	1843.25	12.45
14	1497.73	<b>1528.19</b>	12.97	1363.51	<b>1477.81</b>	16.23	1521.98	1595.29	13.87	1586.07	1665.35	17.69	1464.63	<b>1524.39</b>	11.90
15	2630.50	<b>2735.11</b>	8.08	<u>2598.60</u>	<b>2654.96</b>	17.29	2749.48	2779.70	17.71	2619.75	2809.98	20.55	2631.54	2784.07	10.78
16	2817.11	<b>2843.07</b>	15.14	2761.74	2928.97	11.98	<u>2749.80</u>	2939.81	16.50	2799.54	3023.09	18.49	2763.78	<b>2786.13</b>	16.60
17	3000.59	3180.73	16.66	<u>2836.96</u>	<b>3115.84</b>	18.10	2980.00	<b>3108.34</b>	17.31	3003.54	3160.28	16.77	2938.26	<b>3113.39</b>	13.37
18	<u>3757.86</u>	<b>3769.07</b>	11.74	3953.18	3977.04	16.75	3784.36	3816.20	15.30	4068.05	4108.96	15.35	3820.77	<b>3830.97</b>	17.87
19	4138.52	<b>4211.71</b>	13.89	<u>3976.44</u>	<b>4127.39</b>	17.29	4117.34	4245.11	14.39	4119.55	<b>4121.77</b>	15.35	4026.18	<b>4127.57</b>	17.29
20	4339.11	4356.20	12.54	4116.26	<b>4123.78</b>	18.37	4162.99	4265.80	13.32	4300.76	<b>4529.81</b>	14.47	4175.82	<b>4221.98</b>	13.71
21	5883.78	<b>5979.56</b>	18.93	<u>5742.54</u>	<b>5957.26</b>	19.03	5900.34	6072.23	18.61	5806.82	6057.81	15.06	5759.79	<b>5876.49</b>	11.91
22	5891.33	<b>5987.31</b>	11.67	5904.85	<b>5971.91</b>	23.53	6023.00	6216.99	22.63	5959.31	6038.23	12.94	<u>5837.47</u>	6089.75	15.07
23	6291.53	6391.33	19.27	6245.36	<b>6290.23</b>	28.56	6279.29	6374.94	29.00	6181.54	6405.51	22.04	<u>6152.71</u>	<b>6246.45</b>	20.64



**Table 11** Summary of landscape features and most appropriate algorithm

Customer/ Depot	NO	Landscape Features	Ranking of the most appropriate Algorithm
Cus:50,75 Dep:4,5	1 2 3	The landscape structure of each instance is rugged, multi-modal, and the presence of some needle-like funnels	1st: ACO 2nd: DE
Cus:100 Dep:2,3,4	4 5 6 7	The landscape structure of each instance is rugged, multi-modal, and the presence of some needle-like funnels. Also, the number of local optimum increases as the number of depots	1st: ACO 2nd: TS
Cus:249 Dep:2,3,4,5	8 9 10 11	The landscape structure of each instance is rugged, multi-modal, deceptive, and the presence of many needle-like funnels. The number of local optimum increases as the number of depots	1st: GA 2nd: ACO
Cus:80 Dep:2	12 13 14	The landscape structure of each instance is rugged, multi-modal, and the presence of some needle-like funnels	1st: ACO 2nd: GA, DE
Cus:160 Dep:4	15 16 17	The landscape structure of each instance is highly rugged, multi-modal, deceptive, and the presence of many needle-like funnels	1st: GA 2nd: PSO
Cus:240 Dep:6	18 19 20	The landscape structure of each instance is highly rugged, multi-modal, deceptive, and the presence of many needle-like funnels. Also, the number of local optimum is approximately same due to same number of customers and depots	1st: GA 2nd: ACO
Cus:360 Dep:9	21 22 23	The landscape structure of each instance is highly rugged, multi-modal, deceptive, and the presence of many needle-like funnels. Also, the number of local optimum is approximately same due to same number of customers and depots	1st: TS 2nd: GA

0.66%, indicating that the BPC-HACO algorithm can achieve better solutions for large-scale problems. The  $E\_Sbest$  for instances R1 (48 customers, 4 depots), R2 (96 customers, 4 depots), and R7 (72 customers, 6 depots) is 0%, indicating that the BPC-HACO algorithm can achieve a BSK for small-scale problems.

### 4.3 BPC-HACO for solving small-scale MDVRP

The BPC-HACO is also effective for solving small-scale MDVRP (MDVRP with a small number of customers and depots). The experimental data come from Wang et al. (2012), Luo et al. (2014), Ma et al. (2014), Yan et al. (2017), Zhang et al. (2009), Ling et al. (2017), where the number of depots is 3, and the number of customers ranged from 15 to 30. The detailed information for the data sets and the solution results are shown in Table 8.

The fifth column shows the lengths, the sixth and seventh columns show the lengths and the routes solved by the BPC-HACO, and in the seventh column, 1 is the virtual center, 2, 3 and 4 are the depots, 5 is customer 1, 6 is customer 2, and so on. The BPC-HACO can get better solutions than Wang et al. (2012), Ma et al. (2014), Yan

et al. (2017), and Ling et al. (2017) and finds the same result as Luo et al. (2014). The length of the BPC-HACO is 0.52 more than Zhang et al. (2009), but it had one less vehicle, demonstrating that the BPC-HACO has advantages for solving small-scale MDVRP.

### 4.4 Fitness landscape analysis of the algorithm

There are many metaheuristic algorithms for solving MDVRP, such as Ant Colony Optimization (ACO), Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Differential Evolution (DE), and Tabu Search (TS).

Hence, a fundamental question arises: how to select a best-suited heuristic? Fitness landscape analysis (FLA) has been applied successfully (Zhou et al. (2023); Rodríguez et al. (2023); Dokeroglu et al. (2023)), which provides structural features of the problem to choose the most appropriate algorithm for solving the problem. Fitness landscape generated by random walks in search spaces obtained using different local operators on MDVRP. The fitness landscape is a surface in the search space that defines the fitness for each sample.



#### 4.4.1 Landscape analysis

FLA measures the ruggedness of a landscape in two ways: 1. Information analysis measure ( $H(\varepsilon)$ ,  $M(\varepsilon)$ ,  $\varepsilon^*$ ) 2. Statistical analysis measure ( $rFD$ ,  $\tau$ ,  $\delta$ ). The calculation methods of these 6 parameters refer to Jana et al. (2017). Table 9 summarizes the mean and standard deviation (Std.) of the parameters of the landscape measure for 23 benchmark instances (detailed information in Table 1) over 20 runs.

$H(\varepsilon)$  increases with the number of customers or depots, and the Entropy measure indicates that all instances' landscapes are rugged.  $M(\varepsilon)$  does not change significantly and indicates the existence of multi-modality on the instance landscape structure.  $\varepsilon^*$  increases with the complexity of the instance. The more complex the instance, the greater the fitness difference between samples. To make the landscape structure becomes flat, the greater  $\varepsilon^*$  will be.  $rFD$  is small for all instances, indicating that the samples' fitness and distances to the global minimum are uncorrelated. Moreover, the negative value indicates that many local optimums exist on the landscape structure, which is deceptive.  $\tau$  decreases with an increase in the number of customers or depots, which indicates that the degree of nonlinear correlation between neighbor samples increases with the complexity of the instance.  $\delta$  indicates a significant increase in dispersion with length, which indicates the existence of multi-funnels in the landscape structure and the best solution is far away from the possible solutions.

#### 4.4.2 Performance analysis for algorithms

We consider the 5 widely used algorithms for solving MDVRP (ACO, GA, PSO, DE, TS) and only use the standard versions of these algorithms. The parameter settings of these algorithms refer to (Jana et al. (2017)). For fair comparisons, the algorithm's parameters are consistent, such as population size and the number of iterations.

Table 10 reports the best and mean results and standard deviation of 20 runs for each algorithm on 23 instances. Wilcoxon's rank sum test is conducted to judge whether has a statistical difference between the best algorithm and the competitors (Bo et al. (2023)). The mean results on each row are in boldface, while 'italics' indicate no significant difference between the best algorithm. The best result is underlined.

From the above results and discussions, we summarize the landscape features and the most appropriate algorithm for solving MDVRP in Table 11.

ACO performs well when the number of customers or depots is small (instances 1 ~ 14), and the standard deviation does not change much, indicating that ACO is robust. GA performs well when the number of customers or

depots is large (instances 15 ~ 21), but GA has large standard deviations in some instances. TS achieved the best results on instances 5, 9, 22 and 23 and performed on average similarly to ACO and GA. The standard deviation shows that TS is also robust in some instances. DE provided the best results for instances 1 and 12, and no significant performance improvement except for instances 1 and 12. DE performance decreases as the complexity of the instance increases. In general, the success of DE depends heavily on mutation and crossover operators. PSO obtains the best results on instances 11 and 16. PSO performance degrades as the complexity of the instance increases, similar to DE.

In conclusion, ACO, GA and TS algorithms perform well in solving MDVRP. For other algorithms, some improvements are made to get better results.

## 5 Conclusions and future work

This paper proposes a hybrid ant colony optimization (BPC-HACO) to solve the MDVRP model based on the polygonal circumcenter. As the BPC-HACO avoids clustering and uses pheromone accumulations to guide the algorithm to select depots and customers intelligently, it can optimize the routes in each depot and the total route of all depots to achieve a unity of quality and efficiency. Furthermore, local search and elite strategies are added to increase the population diversity and retain better solutions. Two kinds of MDVRP benchmarks and six small-scale data sets are employed to verify the effectiveness of the BPC-HACO in solving MDVRP, MDVRP with distance constraints, DMDVRP and small-scale MDVRP. Finally, FLA technology is used to analyze the structural features of MDVRP to choose a suitable algorithm to solve it. Further research could consider an MDVRP with customer priority, distance, and time asymmetries, closer to the needed practical applications.

**Acknowledgements** National Natural Science Foundation of China [Grant No. 72074198, 71874165, 71573237]; the National Social Science Fund of China [Grant No. 21AZD074]; Young Talents Foundation of The Central Propaganda Department [Grant No. 2020084007]; the Research Foundation of Philosophy and Social sciences of Ministry of Education of China [Grant No. 20JHQ094]; Soft science research project of technological innovation in Hubei Province [Grant No. 2019ADC154]; Young Talents Foundation of The Central Propaganda Department; Hubei Provincial Natural Science Foundation of China [Grant No. 2020CFB822].

**Author contributions** WF contributed to conceptualization, methodology, software, visualization, investigation, and writing—original draft. GH was involved in funding acquisition, project administration, writing—review and editing, supervision, and data curation. PW contributed to data curation, writing—review and editing,

supervision, and validation. HJ and CS were involved in writing—review and editing and supervision.

**Funding** National Natural Science Foundation of China [grant numbers 72074198, 71874165, 71573237]; the National Social Science Fund of China [grant numbers 21AZD074]; Young Talents Foundation of The Central Propaganda Department [grant number 2020084007].

**Data availability** The datasets used and/or analysed during the current study available from the corresponding author on reasonable request.

## Declarations

**Conflict of interest** The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

**Ethical approval** No conflict of interest exists in submitting this manuscript, and all authors approve the manuscript for publication.

**Informed consent** On behalf of my co-authors, I would like to declare that the work described was original research that has not been published previously and is not under consideration for publication elsewhere, in whole or in part.

## References

- Asefi H, Shahparvari S, Chhetri P, Lim S (2019) Variable fleet size and mix VRP with fleet heterogeneity in integrated solid waste management. *J Clean Prod* 230:1376–1395
- Azuero-Ortiz J, Gaviria-Hernández M, Jiménez-Rodríguez V, Vale-Santiago E, González-Neira E (2023) Design of a hybridization between Tabu search and PAES algorithms to solve a multi-depot, multi-product green vehicle routing problem. *Decis Sci Lett* 12(2):441–456
- Bezerra SN, Souza SRD, Souza MJF (2018) A GVNS algorithm for solving the multi-depot vehicle routing problem. *Electron Notes Discrete Math* 66:167–174
- Bo Q, Cheng W, Khishe M (2023) Evolving chimp optimization algorithm by weighted opposition-based technique and greedy search for multimodal engineering problems. *Appl Soft Comput* 132:109869
- Cai W, Wang Y, Yu B (2014) Improved ant colony algorithm for period vehicle routing problem. *Oper Res Manag Sci* 23(5):70–77
- Chen CM, Lv S, Ning J, Wu JMT (2023) A genetic algorithm for the waitable time-varying multi-depot green vehicle routing problem. *Symmetry* 15(1):124
- Dokeroglu T, & Ozdemir YS (2023) A new robust Harris Hawk optimization algorithm for large quadratic assignment problems. *Neural Comput Appl*, 1–14.
- Ge X, Jin Y, Zhang L (2023) Genetic-based algorithms for cash-in-transit multi depot vehicle routing problems: economic and environmental optimization. *Environ Dev Sustain* 25(1):557–586
- Gu, Z., Zhu, Y., Wang, Y., Du, X., Guizani, M., & Tian, Z. (2022). Applying artificial bee colony algorithm to the multidepot vehicle routing problem. *Software: Practice and Experience*, 52(3), 756–771.
- Hu H, Li X, Ha M, Wang X, Shang C, Shen Q (2023) Multi-depot vehicle routing programming for hazmat transportation with weight variation risk. *Transportmetrica B Trans Dyn* 11(1):1136–1160
- Jana ND, Sil J, Das S (2017) Selection of appropriate metaheuristic algorithms for protein structure prediction in AB off-lattice model: a perspective from fitness landscape analysis. *Inf Sci* 391:28–64
- Kim J, Manna A, Roy A, & Moon I (2023) Clustered vehicle routing problem for waste collection with smart operational management approaches. *Int Transact Oper Res*.
- Kronmueller M, Fielbaum A, & Alonso-Mora J (2023) Pooled grocery delivery with tight deadlines from multiple depots. *arXiv preprint arXiv:2303.11804*.
- Lavigne C, Inghels D, Dullaert W, Dewil R (2023) A memetic algorithm for solving rich waste collection problems. *Eur J Oper Res* 308(2):581–604
- Li Y, Soleimani H, Zohal M (2019) An improved ant colony optimization algorithm for the multi-depot green vehicle routing problem with multiple objectives. *J Clean Prod* 227:1161–1172
- Ling H, Gu J (2017) Study on multi-depot open vehicle routing problem with soft time windows. *Comput Eng Appl* 53(14):232–239
- Londoñoa AA, Gonzalez WG, Giraldob ODM, & Willmer J (2023) A new matheuristic approach based on Chu-Beasley genetic approach for the multi-depot electric vehicle routing problem. *Int J Ind Eng Comput*, 14 (2023).
- Luo HB (2014) Study on multi-depots and multi-vehicles vehicle scheduling problem based on improved particle swarm optimization. *Comput Eng Appl* 50(7):251–253
- Ma Y, Yao T, Zhang F (2014) Multi-depot multi-type vehicle scheduling problem and its genetic algorithm. *Math Pract Theory* 2:107–114
- Manullang MJC, Priandana K, & Hardhienata MKD (2023) Optimum trajectory of multi-UAV for fertilization of paddy fields using ant colony optimization (ACO) and 2-opt algorithms. In *AIP conference proceedings* (Vol. 2482, No. 1, p. 020004). AIP Publishing LLC.
- Nia AR, Awasthi A, & Bhuiyan N (2023). Integrate exergy costs and carbon reduction policy in order to optimize the sustainability development of coal supply chains in uncertain conditions. *Int J Prod Econ*, 108772.
- Ren T, Luo T, Jia B, Yang B, Wang L, Xing L (2023) Improved ant colony optimization for the vehicle routing problem with split pickup and split delivery. *Swarm Evol Comput* 77:101228
- Rodríguez-Maya NE, Flores JJ, Verel S, & Graff M (2023). Models to classify the difficulty of genetic algorithms to solve continuous optimization problems. *Natl Comput*, 1–21.
- Sadati MEH, Aksen D, & Aras N (2020) A Trilevel r-Interdiction selective multi-depot vehicle routing problem with depot protection. *Comput Oper Res*, 104996.
- Soeanu A, Ray S, Berger J, Boukhtouta A, Debbabi M (2020) Multi-depot vehicle routing problem with risk mitigation: model and solution algorithm. *Expert Syst Appl* 145:113099
- Torres-Pérez I, Rosete A, Sosa-Gómez G, Rojas O (2022) New heuristics for assigning in the multi-depot vehicle routing problem. *IFAC-PapersOnLine* 55(10):2228–2233
- Vincent FY, Aloina G, Jodiawan P, Gunawan A, Huang TC (2023) The vehicle routing problem with simultaneous pickup and delivery and occasional drivers. *Expert Syst Appl* 214:119118
- Wang Z, Wu Y (2023) An ant colony optimization-simulated annealing algorithm for solving a multiload AGVs workshop scheduling problem with limited buffer capacity. *Processes* 11(3):861
- Wang C, Guo C, Zuo X (2021) Solving multi-depot electric vehicle scheduling problem by column generation and genetic algorithm. *Appl Soft Comput* 112:107774

- Wang S, Han C, Yu Y, Huang M, Sun W, Kaku I (2022) Reducing carbon emissions for the vehicle routing problem by utilizing multiple depots. *Sustainability* 14(3):1264
- Wang Y, Wei Y, Wang X, Wang Z, Wang H (2023) A clustering-based extended genetic algorithm for the multidepot vehicle routing problem with time windows and three-dimensional loading constraints. *Appl Soft Comput* 133:109922
- Wang T, Wu K (2012) Study on multi-depot vehicle routing problem with time windows based on particle swarm optimization. *Jisuanji Gongcheng Yu Yingyong (comput Eng Appl)* 48(27):27–30
- Wu D, Li J, Cui J, Hu D (2023) Research on the time-dependent vehicle routing problem for fresh agricultural products based on customer value. *Agriculture* 13(3):681
- Wu H, & Gao Y (2023) An ant colony optimization based on local search for the vehicle routing problem with simultaneous pickup-delivery and time window. *Appl Soft Comput*, 110203.
- Xu H, Pu P, & Duan F (2018) A hybrid ant colony optimization for dynamic multidepot vehicle routing problem. *Discrete Dynam Nat Soc*, 2018.
- Xue S (2023) An adaptive ant colony algorithm for crowdsourcing multi-depot vehicle routing problem with time windows. *Sustain Oper Comput*.
- Yan K, Li A, Guo J, Chen B (2017) Research on multi-depot vehicle routing problem based on time window. *Geospatial Inform* 5:11
- Yao B, Chen C, Song X, Yang X (2019) Fresh seafood delivery routing problem using an improved ant colony optimization. *Ann Oper Res* 273(1–2):163–186
- Yu B, Yang ZZ, Xie JX (2011) A parallel improved ant colony optimization for multi-depot vehicle routing problem. *J Oper Res Soc* 62(1):183–188
- Yu B, Ma N, Cai W, Li T, Yuan X, Yao B (2013) Improved ant colony optimization for the dynamic multi-depot vehicle routing problem. *Int J Log Res Appl* 16(2):144–157
- Yuan, X., Zhang, Q., Liu, H., & Wu, L. (2020). Solving MDVRP with grey delivery time based on improved quantum evolutionary algorithm. *J Grey Syst*, 32(3).
- Yücenur GN, Demirel NÇ (2011) A new geometric shape-based genetic clustering algorithm for the multi-depot vehicle routing problem. *Expert Syst Appl* 38(9):11859–11865
- Zhang W, Gajpal Y, Appadoo SS, Wei Q (2020) Multi-depot green vehicle routing problem to minimize carbon emissions. *Sustainability* 12(8):3500
- Zhang J, Tang JF, Pan ZD (2009) A scatter search algorithm for multi-depot vehicle routing problem. *Syst Eng* 27(6):83
- Zheng YJ, Chen X, Yan HF, & Zhang MX (2023). Evolutionary algorithm for vehicle routing for shared e-bicycle battery replacement and recycling. *Appl Soft Comput*, 110023.
- Zhou X, Song J, Wu S, Wang M (2023) Artificial bee colony algorithm based on online fitness landscape analysis. *Inf Sci* 619:603–629

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.