

A hybrid optimization approach for the heterogeneous vehicle routing problem with multiple depots cooperative operation^①

Liu Jiansheng (刘建胜)^{②*}, Tan Wenyue^{*}, Jiang Hai^{**}, Yu Gong^{***}

(^{*} School of Mechanical and Electronic Engineering, Nanchang University, Nanchang 330031, P. R. China)

(^{**} School of Mechanical Engineering, Shanghai Jiaotong University, Shanghai 200240, P. R. China)

(^{***} Southampton Business School, University of Southampton, Southampton, SO171 BJ, UK)

Abstract

With the challenge of great growing of transport diversity for the automobile enterprises, the heterogeneous vehicle routing problem with multiple depots, multiple types of finished vehicles and multiple types of transport vehicles in finished vehicle logistics (HVRPMD) is modelled and solved. A multi-objective optimization model for HVRPMD is presented considering loading constraints to minimize the total cost and minimize the number of transport vehicles. Then a hybrid heuristic algorithm based on genetic algorithm and particle swarm optimization (GA-PSO) is developed. Moreover, a case study is used to evaluate the effectiveness of this algorithm. By comparing the GA-PSO algorithm with the traditional GA algorithm, the simulation results demonstrate the proposed GA-PSO algorithm is able to better support the HVRPMD problem in practice. Contributions of the paper are the modelling and solving of a complex HVRPMD in logistics industry.

Key words: finished vehicle logistics (FVL), vehicle routing problem (VRP), hybrid heuristic algorithm, multiple factory depot

0 Introduction

With the development of automobile industry, most of automobile enterprises focus on automobile production, and separate the finished transportation business to some independent logistics company. So the 3rd party finished vehicle logistics (3PFVL) has become trend of transport logistics service for the automobile enterprises and the dealers. The 3PFVL is an important component of the automobile industry which can be characterized as follows: the dealers' orders of finished vehicles are from manufacturers, and then some finished vehicle logistics companies are assigned to transport these finished vehicles to dealers. In order to get the maximum profits, the vehicle loading and routing schedule should be optimized.

Vehicle logistics optimization belongs to vehicle routing problem (VRP). The VRP was firstly proposed in Ref. [1]. It is a combinatorial optimization and integer programming problem to minimize the total route cost. In recent decades, it develops very quickly and considerable number of variants has been considered. VRP has a variety of classification. If classified by task

characteristics, it can be divided into loading problems, unloading problems and handling problems. If classified according to vehicle load conditions, there are full load problems and non-full load problems^[2]. With the different number of depots, it can be classified into single-depot problems^[3] and multiple-depot problems^[4,5]. If classified by the vehicles, it can be classified into single-vehicle routing problems^[6,7] and multiple-vehicle routing problems^[8,9]. From another point of view, the VRP can be classified into open-loop problems (vehicles may not return to the yard) and vehicle closed-loop problems (vehicles must return to the depot) according to classification of relations. And if classified by restrictions, the VRP concludes the capacitated VRP^[10-12], the VRP with time windows (VRPTW), the VRP with pickup and deliveries^[13,14], periodic VRP^[15], dynamic VRP (DVRP)^[16], periodic VRP (PVRP)^[17], split delivery vehicle routing problem (SDVRP). Ref. [18] analyzed why the finished vehicle logistics costs in China are high and discussed how to lower the costs. Ref. [19] analyzed the development trend of FVL in China and put forward a strategy of resource integration.

As a variant of the VRP, the finished vehicle rou-

① Supported by the National Natural Science Foundation of China (No. 51565036).

② To whom correspondence should be addressed. E-mail: liujiansheng@ncu.edu.cn

Received on Feb. 21, 2019

ting problem (FVRP) is a practical routing optimization problem with more special strict constraints. Different from the traditional VRP, both the finished vehicles and transport vehicles are irregular three dimensional shapes in the FVRP. Simplified geometry models of the finished vehicle and transport vehicle are defined in the author's previous paper^[20]. Moreover, with the city development, many industrial parks are built in different districts in one city, and the automobile enterprises build some sub-factories in these different industrial parks. Each sub-factory will produce own specified type of finished vehicles, which leads to a phenomenon that different types of finished vehicles are located in different depots. So the 3PFVL needs assign transport vehicles to load the corresponding type of finished vehicle from different multiple depots, and delivery them to the destination to satisfy customers. So in the VRP for distribution centers, transport vehicles generally dispatch from a main depot^[21], however in the problem, the optimization between depots must be considered. Finally, in the traditional VRP, the transport cost optimization is generally simplified to minimize the travelled distance if the distance charge rate is the same. However, there are different highway toll rates in different loading conditions in the different district areas. For example, in the Jiangxi Province of China, the toll rate will be 0.08 Yuan/km if the loading weight is less than 10 t. If the loading weight is between 10 t and 49 t, the toll rate will be decreased linearly from 0.08 Yuan/km to 0.04 Yuan/km. These make the problem more challenging.

In conclusion, there are no existing methodologies that can be directly used to solve the problem without substantial modification. To the best of our knowledge, this is exploratory research work to consider multi-objective optimization with multiple factory depots, multiple types of vehicles, opened-loop routing, different routing cost weight and the irregular geometry constraints. The main contribution of this work is to solve a heterogeneous vehicle routing problem with many complex practical conditions, such as multiple depots, multiple types of finished vehicles, multiple types of transport vehicles, irregular geometry structure for finished vehicles and loading space, different cost rules and so on.

1 Problem description

The 3PFVL receives some logistics orders which contain different types of finished vehicles located in different depots. To get the maximum profits, the 3PFVL needs an optimization transport solution to finish the orders with the minimum number transport vehicles. Firstly, transport vehicles will go to different depots to load these finished vehicles and then ship them to every customer point till finishing all distribution tasks. Finally transport vehicles will go to the ending depot and wait for next task. As shown in Fig.1, distribution tasks are completed by three transport vehicles. The optimized solution is to select the ordered finished vehicles to load and the transport vehicles into which to load them, and determine a set of routes satisfying all

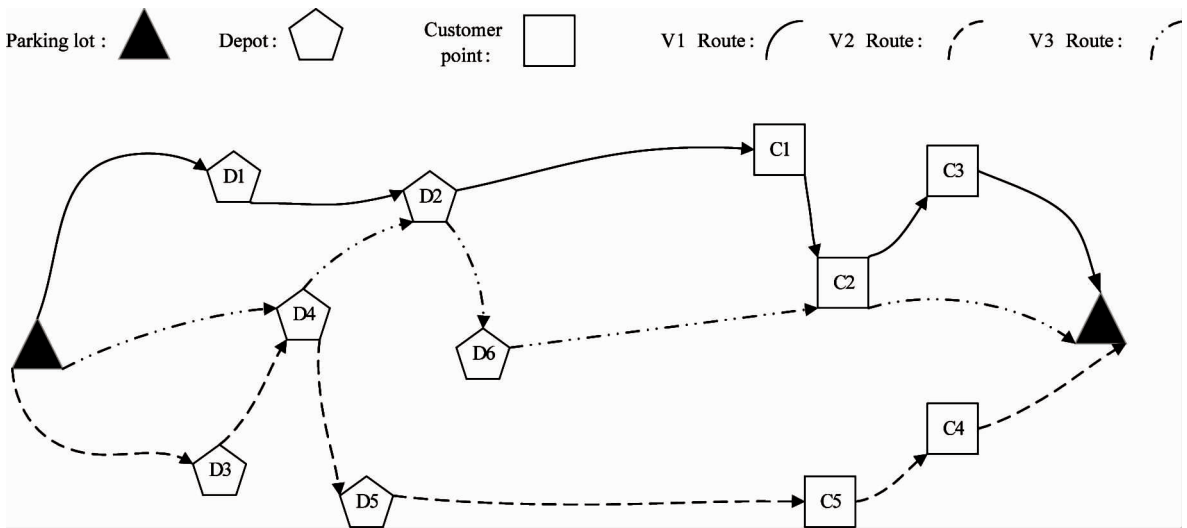


Fig. 1 Transport routes of three transport vehicles

customers under some constraints. Here, each type of transport vehicle can load a limited weight, and can be allowed to visit more than one customer point if the

space is enough. Moreover, each type of transport vehicle has different routing cost according to their self-weight and loading weight. The objective is to find a

set of vehicle routes with minimal total cost and minimal number of transport vehicles. The problem is termed the heterogeneous vehicle routing problem with multiple depots, multiple type of finished vehicles and multiple type of transport vehicles in finished vehicle logistics (HVRPMD).

2 Mathematic model

For the HVRPMD problem, there are many optimization factors, such as routing distance, total cost, total time and loading rate. In practice, many logistics companies generally set the lowest cost and minimum number of transport vehicles as two most important indicators. The optimization model of the HVRPMD is proposed as follows:

$$f_1 = \min K \quad (1)$$

$$f_2 = \min \sum_{k=1}^K [f_{21} + f_{22} + f_{23}] \quad (2)$$

$$f_{21} = \sum_{i=0}^{n_1} \sum_{j=0}^{n_1} \alpha_{ijk} a_{ij} (C_{ijr_k}^1 + C_{r_k})$$

$$f_{22} = d_{p_k q_k} (C_{p_k q_k r_k} + C_{r_k})$$

$$f_{23} = \sum_{i=0}^{n_2} \sum_{j=0}^{n_2} \beta_{ijk} b_{ij} (C_{ijr_k}^2 + C_{r_k})$$

s. t.

$$\alpha_{ijk} (\alpha_{ijk} - 1) = 0, i, j \in [0, n_1], k \in [1, K] \quad (3)$$

$$\beta_{ijk} (\beta_{ijk} - 1) = 0, i, j \in [0, n_2], k \in [1, K] \quad (4)$$

$$\alpha_{iik} = \beta_{jjk} = 0, i \in [0, n_1], j \in [0, n_2], k \in [1, K] \quad (5)$$

$$p_k \in [1, n_1], q_k \in [1, n_2], p_k, q_k \in Z, k \in [1, K] \quad (6)$$

$$\sum_{j=1}^{n_1} \alpha_{0jk} = \sum_{i=0}^{n_1} \alpha_{ipk} = 1, k \in [1, K] \quad (7)$$

$$\sum_{j=0}^{n_1} \alpha_{jik} = \sum_{j=1}^{n_1} \alpha_{ijk}, i \in [1, n_1], i \neq p_k, k \in [1, K] \quad (8)$$

$$S_{ik} (\sum_{j=0}^{n_1} \alpha_{jik} - 1) = 0, i \in [1, n_1], k \in [1, K] \quad (9)$$

$$K = \sum_{i=1}^m k_i \quad (10)$$

$$r_k = \begin{cases} 1 & k \in [1, k_1] \\ 2 & k \in [k_1 + 1, k_1 + k_2] \\ \dots & \dots \\ m & k \in [\sum_{i=1}^{m-1} k_i + 1, \sum_{i=1}^m k_i], \end{cases} m \in N \quad (11)$$

$$\sum_{i=1}^{n_1} S_{ik} w_i \leq G_{r_k}, k \in [1, K] \quad (12)$$

$$\sum_{i=1}^{n_1} l_i S_{ik} \leq L_{r_k}, k \in [1, K] \quad (13)$$

$$y_{ik} (y_{ik} - 1) = 0, i \in [1, n_2], k \in [1, K] \quad (14)$$

$$\sum_{k=1}^K y_{ik} = 1, i \in [1, n_2], k \in [1, K] \quad (15)$$

$$\sum_{j=1}^{n_2} \beta_{jik} = y_{ik}, i \in [1, n_2], i \neq q_k \quad (16)$$

$$\sum_{j=0}^{n_2} \beta_{ijk} = y_{ik}, i \in [1, n_2], k \in [1, K] \quad (17)$$

$$\sum_{j=0}^{n_2} \beta_{qjk} = 1, k \in [1, K] \quad (18)$$

$$\sum_{i=1}^{n_2} \beta_{iok} = 1, k \in [1, K] \quad (19)$$

$$S_{ik} = \sum_{j=1}^{n_2} D_{ji} y_{jk}, i \in [1, n_1], k \in [1, K] \quad (20)$$

Notation:

n_1 : The number of depots.

n_2 : The number of customer points.

N : The number of transport vehicle types

r_k : The type of K th transport vehicle.

K : The total selected number of transport vehicles.

p_k : The last depot that the k th transport vehicle arrived at.

q_k : The first customer point that the k th transport vehicle arrived at.

$$\alpha_{ijk} = \begin{cases} 1 & \text{if the } k\text{th transport vehicle goes to load finished vehicles from depot } i \text{ to depot } j \\ 0 & \text{otherwise} \end{cases}$$

$$\beta_{ijk} = \begin{cases} 1 & \text{if the } k\text{th transport vehicle goes from customer } i \text{ to customer } j \\ 0 & \text{otherwise} \end{cases}$$

a_{ij} : The distance between depot i and j .

b_{ij} : The distance between customer point i and j .

$d_{p_k q_k}$: The distance between depot p_k last arrived at and customer point q_k first arrived at for the K th transport vehicle.

$C_{ijr_k}^1$: The toll per unit distance from depot i to j for the transport vehicles of model r_k .

$C_{ijr_k}^2$: The toll per unit distance from customer point i to j for the transport vehicles of model r_k .

C_{r_k} : The gas bill per unit distance for the transport vehicles of model r_k .

S_{ik} : The number of finished vehicles that the K th transport vehicle loaded at the depot i .

w_i : The weight of finished vehicles of model i .

G_{r_k} : The limit weight of transport vehicles of model r_k .

l_i : The length of finished vehicles of model i .

L_{r_k} : The limit length of transport vehicles of model r_k

$$y_{ik} = \begin{cases} 1 & \text{if the } K\text{th transport vehicle is} \\ & \text{responsible for order point } i \\ 0 & \text{otherwise} \end{cases}$$

D_{ij} : The demand quantity of finished vehicles of model j for customer i .

Eq. (1) and Eq. (2) are two objective functions. Objective Eq. (1) minimizes the number of transport vehicles. Objective Eq. (2) minimizes the total cost under the conditions of constraints.

Constraints (3) and (4) define two 0 – 1 variables. Constraint (5) defines that α_{ijk} and β_{ijk} are equal to zero if i is equal to j . Constraint (6) indicates that p_k takes values in 1, 2, ..., n_1 and q_k takes values in 1, 2, ..., n_2 . Constraint (7) determines the depot where transport vehicles start from and at which depot they last arrive. Constraint (8) indicates the flow conservation of each depot, which means that the number of transport vehicles that arrive at each depot is equal to the one that leave from this depot. Constraint (9) indicates that only if transport vehicles pass one depot, can they load corresponding finished vehicles. Constraint (10) shows the total number of transport vehicles. Constraint (11) shows the relationship between serial number of transport vehicles and their model. Constraint (12) is the weight limitation of transport vehicles. Constraint (13) is the length limitation of transport vehicle. Constraint (14) defines y_{ik} as a 0 – 1 variable. Constraint (15) limits that for each customer point, there is only one transport vehicle arriving and completing its distribution task. Constraint (16) and Constraint (17) state the flow conservation of each customer point, which means that the number of transport vehicles that arrive at each customer point is equal to the number of transport vehicles that leave from this customer point. Constraints (18) and (19) determine the customer point transport vehicles first arrive at and the ending depot. Constraint (20) ensures the balance between supply and demand.

3 Proposed algorithm

It is very difficult to solve HVRPMD problem with accurate algorithm. This problem is a multi-objective optimization problem. The object that must be met first is minimum transport vehicles. The possible minimum value is shown as Eq. (21).

$$K_{\min} = \left\lceil \max \left\{ \frac{\sum l}{L_{\max}}, \frac{\sum w}{W_{\max}} \right\} \right\rceil + 1 \quad (21)$$

$\sum l$ and $\sum w$ respectively represent the total length and total weight of all finished vehicles. L_{\max} is maximum loading length of the transport vehicle and W_{\max} is maximum loading weight. The square brackets represent taking the integer part towards minus infinity. First set $K = K_{\min}$, but now it still cannot make sure that this K is the minimum number of transport vehicles. The correct K value cannot be determined until the step to initialize the population, which will be mentioned later. Next it needs to design the algorithm for getting the loading schemes and delivery routes.

As a heuristic search algorithm, the genetic algorithm (GA) can achieve good results in dealing with problems such as multimodal, discontinuous, non-differentiable and non-convex. The GA does not depend on the specific area of the problem and has strong robustness. Therefore, it is widely used to solve various complex optimization problems. What's more, genetic algorithm has a good global search ability, which can quickly search out the global solution in the solution space without falling into the fast decline trap of the local optimal solution. However, it also has obvious shortcomings, such as slow convergence, poor local search ability and many control variables. Particle swarm optimization (PSO)^[22] is an evolutionary computation. Similar to genetic algorithm, PSO is an iterative optimization algorithm. Compared with genetic algorithms, the standard PSO has fewer control variables. It performs the extremum optimization by following the individual extremum and global extremum, which is easy to operate and converges quickly. However, with the increasing number of iterations, while the population converges, the particles are more and more similar, and may not jump out of the local optimal solution.

The particle swarm algorithm was originally proposed to solve the continuous optimization problem, and its evolutionary operator is designed for continuous objects. In order to solve the discrete optimization problem, the original evolutionary operator and the operation rules of the particle swarm optimization algorithm must be changed to solve the discrete optimization calculation problem^[23]. To solve the HVRPMD, a hybrid algorithm GA-PSO is proposed. In GA-PSO, PSO shares the individual extremum (pBest) and global extremum (gBest) with GA, and the genetic operator searches and evolves on these feasible solutions. Instead of updating the particle position by tracking the extremum in the traditional particle swarm optimization

algorithm, both the crossover and mutation operations in genetic algorithm are adopted in GA-PSO. It searches for the optimal solution by crossing with the individual extremum and global extremum. This algorithm combines the global optimization of the GA with the fast local search performance of the PSO. It enriches the

search behavior and enhances the search ability. Besides, it also improves the convergence rate of the local region, avoids the stagnation phenomenon in the later phase of the basic PSO algorithm and improves the search accuracy of GA. The steps of GA-PSO hybrid algorithm are shown in Fig. 2.

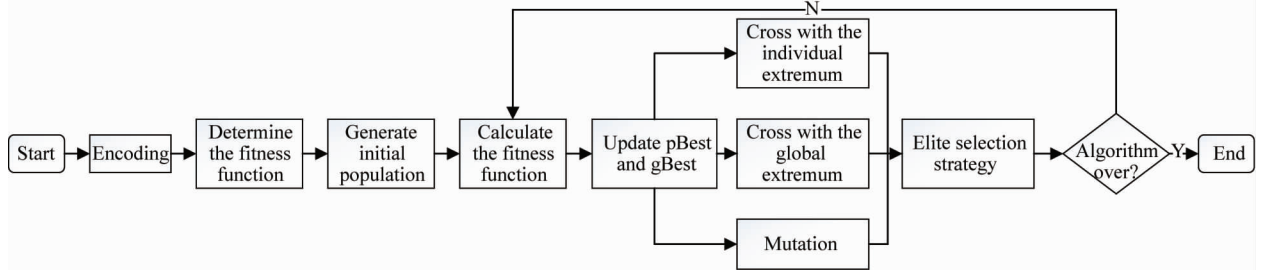


Fig. 2 Flow chart of the GA-PSO algorithm

The GA-PSO algorithm can be represented by pseudo code as follows.

```

Initial population;
Do
    For each chromosome
        Calculate fitness values by Eq. (22);
        If (the current fitness value of the chromosome
is better than the pBest)
            Then update pBest;
        End
        If (the current fitness value of the chromosome
is better than the gBest)
            Then update gBest;
        End
    End
    For each chromosome
        [ value1 , parent1 ] ← do cross-operation with
pBest and calculate fitness values;
        [ value2 , parent2 ] ← do cross-operation with
gBest and calculate fitness values;
        [ value3 , parent3 ] ← do mutate-operation and
calculate fitness values;
        The value of this chromosome ← min ( value1 ,
value2 , value3 );
        Replace the corresponding chromosome in the
population;
    End

```

The chromosomes with large fitness value are replicated twice, and the chromosomes with small fitness value are eliminated; When the fitness minimum error standard or the maximum number of iterations is not reached.

1) Encoding

For vehicle routing problems, the natural number coding is generally used. According to the problem, the encoded array must contain three parts of information: depots, customer points and the model of transport vehicles. Only in this way can we determine the route planning of each transport vehicle. Furthermore, in order to facilitate programming, some digital processing needs to be done, such as ‘-3 -1 -2 -6 -4 16 10 M+1’. This array shows that a transport vehicle of model ‘1’ sets out from the starting depot, then goes to depots 3, 1, 2, 6 to load finished vehicles in turn, ships them to customer point 16, 10 to complete distribution tasks, and finally goes back to the ending depot.

To distinguish depots and customer points, a minus is added to the front of the serial number of each depot. ‘M’ represents a negative number, which has large absolute value. For one hand, it can be distinguished from the number of customer points, for another hand, it can divide the numbers that represent the routes of two vehicles and it’s easy to find ‘M’, the point of division from the array. Among the array only the serial number of customer points is positive, in order to facilitate to find the delivery sequence of customer points.

If this string of number above is regarded as a group, then a chromosome encoding that represents a feasible solution is composed of several groups connected together. A group corresponds to the transport routes of a transport vehicle and the number of groups is equal to the number of transport vehicles. So it is easy to get the information from the encoding.

2) Fitness function

The genetic algorithm uses fitness function values to evaluate individual performance and guide the

search. Therefore, the selection of fitness function is very important. Poor performance fitness functions often lead to cheat problems. The selection criteria of fitness function are: normative (single value, continuous, strictly monotonous), rationality (small amount of calculation) and universality. The 'ranking()' function in Sheffield's Genetic Algorithm Toolbox is used to get the fitness. After such processing, the individual which costs less will has greater fitness and the fitness function values are more evenly distributed in the interval (0,2). The fitness value is calculated by

$$FitnV(Pos) = 2 - sp + 2 \times (sp - 1) \times \frac{Pos - 1}{Nind - 1} \quad (22)$$

This algorithm sorts the objective function in descending order. Individuals with minimum fitness are placed at the first position in the ordered list of objective function values. The *Nind* is the size of the population, the *Pos* is the individual's position in a sorted population. The *sp* specifies the pressure drop and set *sp* = 2 in the hybrid algorithm.

3) Initialization

It needs to consider that how to generate a population of a certain size, which can meet all constraints at the same time. Customer points should be generated first, and then the relevant depots and demands can be generated. The process of generating the individual has the following steps.

Step 1 Generate a disordered array from 1 to 20 by the function 'randperm' in Matlab (Suppose there are 20 customer points).

Step 2 Select the model of transport vehicle through a judgment statement, the codes are shown as below:

```
u = rand;
if 0 < u & u < = pa
u = 1;
else if pa < u & u < = pb
u = 2;
else
u = 3;
end
```

'u' represents the type of transport vehicle. 'pa' and 'pb' are transport vehicle selection parameters. In order to achieve the objective to minimize the number of transport vehicles, the capacity of transport vehicles selected should be as large as possible. So the parameter value 'pa' and 'pb' should be determined according to the capacity of each type of transport vehicles.

Step 3 Separate this group from the later part. As long as the loading weight or length exceeds the

limit of the transport vehicle, just insert the number 'M + u'.

Step 4 Determine the depots based on the customers, then generate a disordered array and insert it in front of the customer points after taking opposite numbers.

Step 5 Repeat Step2 – Step4 until there is no customer point left.

According to the steps above, an initial population of a certain size can be generated through a loop structure. Meanwhile, the minimum number of transport vehicles is determined. When an individual (a feasible solution) is generated, it should get the number of transport vehicles and determine whether this value is equal to the *K* got before. If there is still no such individual that the number of transport vehicles is equal to *K* after a certain number of cycles reached, then to a large extent, it indicates that the value of *K* is not large enough. So *K* should be added 1 and continue the loop. Specific codes are as follows:

```
If t > 1000 & num = = 0
vn = vn + 1;
end
```

The *vn*(*K*) is the minimum number of transport vehicles.

4) Update the particle

It should update the individual extremum and the global extremum in each iteration of the population.

5) Crossover operator

Individuals are updated by crossing with the individual extremum and crossing with the global extremum separately. Cross-operation using integers crossing method. Since the coding method used in this algorithm is natural number coding, the way of cross-encoding is shown as follows:

First, remove all inserted numbers, leaving only 20 digits representing the order of the customer points. Then the chromosomes are crossed. Crossover point and exchange length are determined randomly. In order to avoid duplication encoding, duplicate point should be deleted before exchanging. For example:

A = 12 20 2[14 15 10 13] 18 8 9 1 5 17 11 7 6
16 3 4 19

B = 15 14 5[20 11 18 2] 1 7 6 16 19 8 13 12 9 10
3 4 17

Then, remove the duplicate numbers 20 11 18 2 in array A and remove 14 15 10 13 in array B. The followings are got:

A = 12[14 15 10 13] 8 9 1 5 17 7 6 16 3 4 19

B = 5[20 11 18 2] 1 7 6 16 19 8 12 9 3 4 17

Next move exchange segments to the front of A and B respectively and the followings are got:

A = 20 11 18 2 12 14 15 10 13 8 9 1 5 17 7 6 16
3 4 19
B = 14 15 10 13 5 20 11 18 2 1 7 6 16 19 8 12 9
3 4 17

Then, repeat the initializing operation to insert figures that represent depots and models of transport vehicles. Finally, a pair of new chromosomes is generated.

6) Mutation operator

The mutation operations that genetic algorithm commonly used are: Simple Mutation, Uniform Mutation, Boundary Mutation and Gaussian Mutation and so on. In the proposed algorithm, the Multi-point Mutation is used, which is one of the Simple Mutation. The specific operation is as follows: First, a chromosome p is randomly selected in the population, and then 3 gene points are randomly selected on the chromosome. After that, the two chromosome segments between these 3 points are interchanged. Thus, a new chromosome can be obtained and the mutation operation is completed. For example:

P = 8 9 17 14 18【12】2 10 16 20【3】7 19 4 1
【13】11 5 15 6
P' = 8 9 17 14 18 7 19 4 1【13】【12】2 10 16 20
【3】11 5 15 6

7) Elitist strategy

Chromosomes are sorted by fitness value from large to small. The individuals with high fitness value are selected and transmitted to the next generation while the remaining individuals will be discarded. However, some improvements are made by introducing the elite selection strategy. The chromosome which has the maximum fitness in this generation is defined as the elite individual. Among the populations obtained through crossover and mutation operation, individuals with large fitness values may be selected twice and individuals

with lower fitness values will be discarded. Then a new population is generated and continues to evolve. This adaptive improved method can avoid precocity and in-breeding. The formula is

$$N(i) = round(\frac{m(i)}{\sum_{i=1}^{num} m(i)} \times num) \tag{23}$$

where, $m(i)$ represents the fitness value of the i th chromosome, num represents the size of the population. $N(i)$ indicates the number of copies of the chromosome.

4 Computational experience

4.1 A case study

Using the data of a medium-sized logistics company in China, there are 3 models of transport vehicles and the number is sufficient in the logistics company. The relevant data of transport vehicles are listed in Table 1. The depots coordinates and finished vehicle specifications are listed in Table 2. Some orders are listed in Table 3. There are 5 depots located in different industrial parks, corresponding to 5 models of finished vehicles located from A to E.

The optimization objective is to minimize cost. The transport cost consists of two parts: the tolls and gas bills. Gas bill per unit distance relates to the model of the transport vehicle. The toll per unit distance depends on the model of the transport vehicle and the specific routes. In other words, for the same model of transport vehicles, the tolls per unit distance on different roads may be different. From Table 1, the toll of each type transport vehicles is different among order city points. It is not a constant value and it is changed dynamically based on the vehicle models and routes.

Table 1 Relevant data of transport vehicles

Models	Self-weight (t)	Limit total weight (t)	Upper floor length(m)	Upper floor row number	Lower floor length (m)	Gas bill (Yuan/km)	Toll (Yuan/km)
1	27	49	33	2	29	2.2	1.92 – 2.40
2	25	43	28	1	27	2.0	1.68 – 2.10
3	23	35	28	1	27	1.8	1.44 – 1.80

Table 2 The information of each depot and finished vehicles

Depots	X-axis(km)	Y-axis(km)	Finished vehicle	Length(mm)	Weight(kg)
1(Xiaolan)	75	55	A	4 545	1 280
2(Jiaoqiao)	90	120	B	4 731	1 550
3(Yinbin)	80	80	C	4 630	1 495
4(Changdong)	110	83	D	3 850	1 045
5(Wangcheng)	50	77	E	4 370	1 050

Table 3 Customer order information

Customers	X-axis(km)	Y-axis(km)	Orders
1	400	100	1A + 1B + 2C
2	300	80	3B + 2C + 3D
3	255	220	4C + 4D
4	230	144	2B + 2C + E
5	280	224	5A
6	260	180	1C + 2D + 1E
7	330	280	1A + 1D + 2E
8	430	160	1B + 1D + 1E
9	460	295	5A + 2C
10	480	160	2B + 3C + 2E
12	200	170	5B
13	220	195	6D + 2E
11	280	110	3C + 3D
14	410	125	1A + 2C + 1D
15	363	275	5B + 1E
16	290	245	2A + 2B + 2D
17	355	230	3D + 1E
18	368	65	1A + 1B + 1C
19	341	138	4A + 3E
20	475	234	2B + 1C + 1E

Here it is created by the random function. If the number of points is N , then the number of calculation weights which represent toll per unit distance is

$$1 + 2 + \cdots + N - 2 + N - 1 = \frac{1}{2}N(N - 1)$$

(24)

The presented algorithm is coded in Matlab R2015b and tested on a PC with an Intel 3.7 GHz processor,

4.00 GB RAM , and the Microsoft Windows 7 operating system. The parameter values are shown in Table 4.

Table 4 Algorithm parameters

Parameters	Values
$Nind$	100
Gen	1 000
pa	0.7
pb	0.9
P_c	0.8
P_m	0.1

The routing solution is shown in Fig. 3. The calculation total cost is 17 214.463 Yuan and total number of transport vehicles is 7. Moreover, the detail routes are listed in Table 5 and the optimization iteration process is shown in Fig. 4(a).

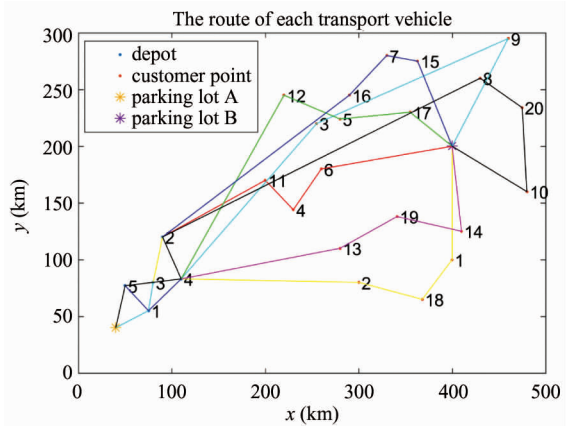


Fig. 3 The optimization vehicle routing results (GA-PSO)

Table 5 The detail routes

Transport vehicles	Transport vehicle type	Routes	Finished vehicle
V_1	1	S-5-3-4-2--11-4-6-E	7B + 3C + 2D + 2E
V_2	1	S-1-3-2-4--2-18-1-E	2A + 5B + 5C + 3D
V_3	1	S-5-1-3-4--13-19-14-E	5A + 5C + 4D + 3E
V_4	1	S-1-3-4--3-9-E	5A + 6C + 4D
V_5	1	S-5-1-4--12-5-17-E	5A + 9D + 3E
V_6	1	S-5-1-4-2--16-7-15-E	3A + 7B + 3D + 3E
V_7	1	S-5-3-4-2--8-20-10-E	5B + 4C + 1D + 4E

4.2 Performance analysis and algorithm comparison

In order to verify the effectiveness of the GA-PSO algorithm, the GA-PSO algorithm and the genetic algorithm are calculated 10 times respectively. In two algorithms, the same encoding is taken, and the initial population size is set as 100. In genetic algorithms, the crossover rate of genetic algorithm is 0.8 and the

mutation rate is 0.05. The best optimization iteration processes of GA-PSO and GA are shown separately in Fig. 4. The optimal result is obtained in the 341th generation of GA-PSO algorithm simulation. The corresponding result comparison between two algorithms is shown in Table 6. Meanwhile, it lists the worst and average values over the 10 runs for the case.

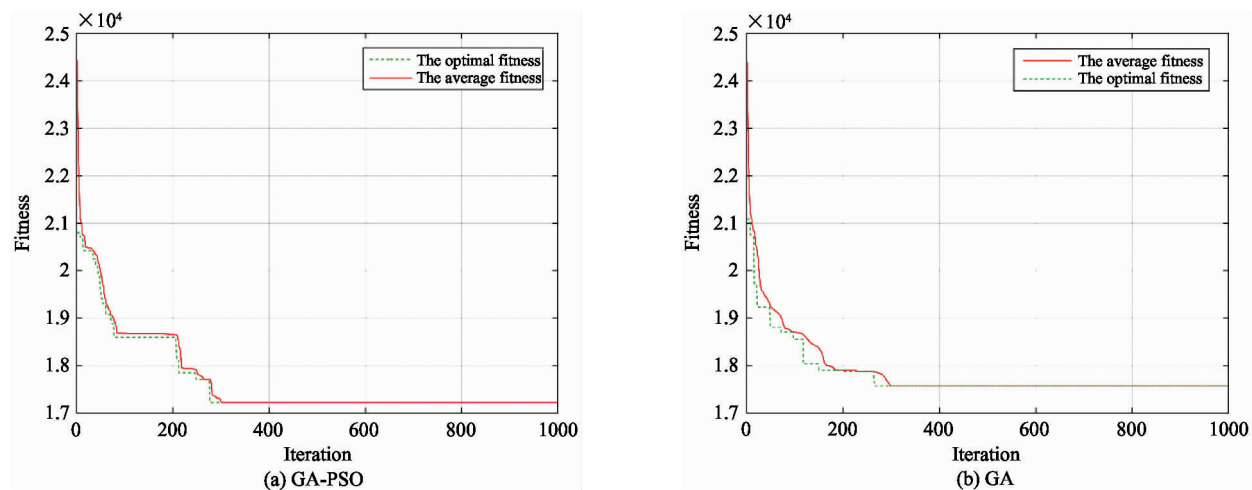


Fig. 4 Optimization iteration figure

Table 6 Algorithm performance comparison

Algorithm	The number of transport vehicles	Best (Yuan)	Worst (Yuan)	Average (Yuan)	CPU time(s)
GA-PSO	7	17 214. 647	17 584. 474	17 385. 1282	284. 693
GA	7	17 503. 613	17 915. 765	17 730. 7257	2 732. 1742

Comparing two algorithms, the GA-PSO algorithm converges faster and the computation result is better. The most importance is that the GA-PSO runs only 10.42% of the time of GA on average. Experiments show that the proposed PSO-GA algorithm has good performance and can quickly find the optimal solution or approximate optimization solution.

In order to fully evaluate the proposed algorithm,

more experiments are conducted. The model and assumptions in these test examples are unchanged. The location map of each tests are still generated randomly. As the problem size increases (number of depots and number of orders), the difference in cost and calculation time between these two algorithms become more apparent, shown in Table 7.

Table 7 Comparison of optimization results

Tests scale (number of depot \times customer)	GA-PSO		GA			
	Cost (Yuan)	CPU time(s)	Convergence	Cost(Yuan)	CPU time(s)	Convergence
1 \times 10	4 671. 733	587. 514	120	4 671. 733	8 258. 2832	180
1 \times 15	5 593. 726	528. 9293	133	5 593. 726	9 080. 6625	126
1 \times 20	7 533. 995	532. 5051	216	7 726. 284	14 846. 9903	280
3 \times 10	8 907. 600	890. 1411	283	9 292. 113	16 838. 7635	289
3 \times 15	12 668. 605	802. 5614	310	13 061. 753	18 953. 1801	274
3 \times 20	16 772. 728	564. 0031	280	17 138. 769	21 692. 0707	140
5 \times 10	13 356. 440	797. 8346	250	13 876. 606	16 198. 0045	272
5 \times 15	18 744. 221	785. 6334	110	18 791. 038	17 057. 7963	187
5 \times 20	22 574. 238	973. 6945	298	23 767. 684	21 546. 1627	301

5 Conclusion

This paper addresses a real transportation problem in finished vehicle logistics, and develops a multi-objective optimization model for the HVRPMD problem by considering the minimum routing cost, the minimum

number of transport vehicles and specified constraints on transport vehicle availability. From the view of a practical contribution, the HVRPMD problem is modeled, and the GA-PSO algorithm is proposed. The process and application of the proposed method are demonstrated using actual cases. Simulation results show the proposed hybrid heuristic approach can match

requirements by transport service with high efficiency in finished vehicle routing problem.

References

- [1] Dantzig G B , Ramser J H . The truck dispatching problem[J]. *Management Science* , 1959 , 6(1) :80-91
- [2] Wei G S , Hong X Z , Ling T . Genetic algorithm for VRP of non-full loads with time windows[J]. *Acta Simulata Systematica Sinica* , 2005(11) :24-28
- [3] Khachay M , Zaytseva H . Polynomial time approximation scheme for single-depot euclidean capacitated vehicle routing problem[J]. *Combinatorial Optimization and Applications* , 2015 , 9486:178-190
- [4] Oliveira F B D , Enayatifar R , Sadaei H J , et al . A cooperative coevolutionary algorithm for the multi-depot vehicle routing problem[J]. *Expert Systems with Applications An International Journal* , 2016 , 43(C) :117-130
- [5] Wang J Y , Lin C K . A new model and heuristic algorithms for the multiple-depot vehicle scheduling problem [J]. *Journal of the Chinese Institute of Engineers* , 2010 , 33(2) :287-299(In Chinese)
- [6] Azi N , Gendreau M , Potvin J Y . An exact algorithm for a single-vehicle routing problem with time windows and multiple routes[J]. *European Journal of Operational Research* , 2007 , 178(3) :755-766
- [7] Gribkovskaia I , Halskau Ø , Laporte G , et al . General solutions to the single vehicle routing problem with pickups and deliveries[J]. *European Journal of Operational Research* , 2007 , 180(2) :568-584
- [8] Lu Q , Dessouky M . An exact algorithm for the multiple vehicle pickup and delivery problem[J]. *Transportation Science* , 2004 , 38(4) :503-514
- [9] Salhi S , Nagy G . A cluster insertion heuristic for single and multiple depot vehicle routing problems with backhauling[J]. *Journal of the Operational Research Society* , 1999 , 50(10) :1034-1042
- [10] Baldacci R , Toth P , Vigo D . Exact algorithms for routing problems under vehicle capacity constraints[J]. *Annals of Operations Research* , 2010 , 175(1) :213-245
- [11] Laporte G , Nobert Y . Exact algorithms for the vehicle routing problem [J]. *Annals of Discrete Mathematics* , 1987 , 31 :147-184
- [12] Asano T , Katoh N , Kawashima K . A new approximation algorithm for the capacitated vehicle routing problem on a tree[J]. *Journal of Combinatorial Optimization* , 2001 , 5 (2) :213-231
- [13] Liu R , Xie X , Augusto V , et al . Heuristic algorithms for a vehicle routing problem with simultaneous delivery and pickup and time windows in home health care[J]. *European Journal of Operational Research* , 2013 , 230 (3) : 475-486
- [14] Gajpal Y , Abad P . An ant colony system (ACS) for vehicle routing problem with simultaneous delivery and pickup[J]. *Computers & Operations Research* , 2009 , 36 (12) :3215-3223
- [15] Solomon M M , Desrosiers J . Time window constrained routing and scheduling problems[J]. *Transportation Science* , 1988 , 22(1) :1-13
- [16] Psaraftis H N . Dynamic vehicle routing: status and prospects[J]. *Annals of Operations Research* , 1995 , 61(1) : 143-164
- [17] Vanderbeck M M F , Vanderbeck A F . The periodic vehicle routing problem: classification and heuristic[J]. *RAIRO - Operations Research* , 2006 , 40(2) :169-194
- [18] Jing L I , Zhang B X . Study on the cause and countermeasures of high-cost finished vehicle logistics in China [J]. *Logistics Technology* , 2007(8) :48-50
- [19] Wu B F , Liu Z Y . A study on finished vehicle logistics in China: its development trend and resource integration [J]. *Automotive Engineering* , 2005 , 27(3) : 367-371
- [20] Liu J S , Smith A E , Qian D . The vehicle loading problem with a heterogeneous transport fleet[J]. *Computers & Industrial Engineering* , 2016 , 97(C) :137-145
- [21] Liu J , Luo Z , Duan D , et al . A GA approach to vehicle routing problem with time windows considering loading constraints[J]. *High Technology Letter* , 2017 , 23 (1) : 54-62
- [22] Kennedy J , Eberhart R C . Particle swarm optimization [C] // IEEE International Conference on Neural Networks , Piscataway , USA , 1995 : 1942-1948
- [23] Wang K P , Huang L , Zhou C G , et al . Particle swarm optimization for traveling salesman problem [C] // Proceedings of the 2003 International Conference on Machine Learning and Cybernetics , Xi ' an , China , 2003 (3) : 1583-1585

Liu Jiansheng, born in 1978. He received his Ph. D degree from Nanchang University in 2009. He studied in Auburn University in USA as visiting scholar granted by Chinese Scholarship Council. His research interests focus on digital and intelligent manufacturing, facility layout and logistics management and optimization.