

Research on Hybrid Immune Algorithm for Solving the Location-Routing Problem With Simultaneous Pickup and Delivery

Xiaowei Wang, Information Management Center, Physical Education College of ZhengZhou University, China*

ABSTRACT

In the simultaneous pickup and delivery problem, every customer has both delivery demand and pick-up demand, and both demands need to be served simultaneously. Under this condition, a location-routing problem with simultaneous pickup and delivery model was established to minimize the sum of location cost, routing cost, and transportation cost. For solving this model, a hybrid immune algorithm was developed. The initial solution was generated by greedy clustering algorithm, the antibody was evaluated and sorted by the original immune algorithm, and the immune operation of the original algorithm was improved by the neighborhood search operation. Finally, the feasibility of the model and the effectiveness of the algorithm were verified by using the hybrid immune algorithm, the original immune algorithm, the simulated annealing algorithm, and the ant colony algorithm.

KEYWORDS

Hybrid Immune Algorithm, Location-Routing Problem, Simultaneous Pickup and Delivery

1 INTRODUCTION

Location-routing problem (LRP) includes the location decision and the path decision, which depend on and influence each other. LRP is a NP difficult problem, and the calculation time increases exponentially as the scale of the problem increases. To this end, intelligent optimization algorithm which can give approximate optimal solution in finite computational time has become the main solution of the LRP problem. Watson-Gandy and Dohrn firstly considered warehouse location in the routing problem (Mu Zhou, Yanmeng Wang & Zengshan Tian, et al. 2019) (Prodhon C., & Prins C., 2014). Prodhon, Prins, Schneider and Drexl focus on LRP field (Schneider M., & Drexl M., 2017) (Drexl M., Schneider M., 2015).

The LRP problem has been widely studied, and the relevant researchers have deepened the variant LRSPD (Location-routing problem with Simultaneous Pickup and Delivery) problem of the LRP problem. The LRSPD problem also belongs to the NP difficult problem as a branch of the LRP problem, which is an extension of the Mosheiov introduction of the transport salesman location problem in the number of warehouses and vehicle capacity (Qiao J. F., Li F., & Yang S. X., et al., 2020). Karaoglan proposed model of simultaneous pickup and delivery, and solved the problem using hybrid genetic algorithm, branch bound method and heuristic algorithm (Mu Zhou, Yaohua Li,

DOI: 10.4018/JCIT.295253

*Corresponding Author

This article published as an Open Access article distributed under the terms of the Creative Commons Attribution License (<http://creativecommons.org/licenses/by/4.0/>) which permits unrestricted use, distribution, and production in any medium, provided the author of the original work and original publication source are properly credited.

& Yiwen Wang, et al, 2021.) (Karaoglan I., Altiparmak F., & Kara I., et al, 2012) (Yu V. F., & Lin S. W., 2014). Yu put forward an intelligent algorithm combining multi-starting point mountain climbing algorithm and simulated annealing algorithm (Yu V. F., & Lin S. Y., 2016) (Wang X. F., 2014). Wang considered several constraints of service modes, fuzzy time windows, and used TS (tabu search) based heuristic algorithm to solve the problem (Zhang Xiaonan, Fan Houming, & Li Jianfeng, 2015). For the positioning-route problem of simultaneous delivery, Zhang Xiaonan designed a variable neighborhood decentralized search algorithm to solve (Sun Qingwei, & Zhang Yang, 2017). Sun Qingwei considered the location-path problem of multiple models and simultaneous delivery, and designed an improved GA for solution (Leng Longlong, Zhao Yanwei, & Zhang Chun miao, et al, 2019). Leng Longlong proposed a quantum superheuristic algorithm to solve the LRPSPD problem of low carbon and multi-vehicle types (Zhang B., Pan Q., & Gao L., et al, 2017).

Based on the LRPSPD problem, this paper considers the cooperation of logistics companies and third-party logistics enterprises, and puts forward a hybrid immune algorithm. Initial solution was generated by greedy clustering algorithm. Besides, the evaluation and rank of antibody was realized by original immune algorithm, and immune operation was improved by neighborhood search algorithm. Taguchi method is used to set the important parameters in the algorithm, and the optimal parameter combination is determined. A set of numerical examples are obtained by using the SN separation method to improve the Prodhon standard database.

2 LOCATION-ROUTING PROBLEM OF SIMULTANEOUS AND PICKUP OF GOODS

The main difference between LRPSPD and LRP is that the customer demand are two different and need simultaneous service and the demand cannot be split. For LRPSPD problems, the previous articles usually calculated the delivery demand and pickup demand separately but the article improved the location-routing problem model of simultaneous delivery, integrate the demand into the load of the vehicle and determine the service order according to the dynamic change of the vehicle load. Prevent the lack of customer service due to the capacity of the vehicle.

2.1 LRPSPD Problem Description

The LRPSPD problem can be defined as: each customer has two needs, likes pickup and delivery demand in a transportation network with M candidate candidates, N customers and K delivery vehicles. The excellent warehouse needs to be selected from the potential M seat candidate warehouse to open and the vehicles start from their warehouses to serve a series of customers in turn. Noting that the vehicle cannot exceed the vehicle capacity and the maximum driving distance of the vehicle and return to the assigned customer with the best warehouse location and quantity, determine the number of vehicles and their route.

This article mainly realizes the following three decisions: ① Location decisions, In a number of candidate warehouses to determine the appropriate warehouse to open. ② Customer decisions, assign warehouse to customers and determine the proper warehouse ownership. ③ Vehicle decisions, determine the number of dispatched vehicles and the corresponding transport route.

2.2 Model Construction

This article builds the problem model of simultaneous delivery and pickup location-routing decision. The model is based on the following assumptions: a) Each customer can only be assigned only once. That is the customer can be served by one warehouse by one vehicle; b) Each vehicle only serves one path and only be enabled once; c) The maximum capacity limit, the total customer demand allocated under must not exceed the warehouse capacity and each customer demand must be met; d) Each vehicle must return to the last customer serving the route; e) The warehouse is the same facility; f) Any two warehouses do not communicate.

By reading a large number of relevant literature, the article found that most of this type of models calculate the pickup demand and delivery demand respectively, and send the corresponding pickup vehicles and delivery vehicles to meet the different needs of customers respectively. Such a model is not only complex but also increases logistics costs. First, the waste of vehicle loading, where the delivery vehicles are empty vehicles when the return journey, pickup vehicles are empty vehicles, empty vehicles vehicle loading resources idle. Second, the waste of driving path and the model assumes that each customer has two needs which requires two types of car service to each customer which requires two types of vehicles to serve each customer. The two vehicles' driving paths have great repeatability, resulting in increased transportation costs. In real life, many vehicles are dispatched in both the pickup business and delivery business. Aiming at the too complex disadvantages of the previous model, the article calculates the customer delivery demand uniformly, linking it to the real-time load weight of the vehicle, jointly forming the load constraint of the vehicle. In this way, the vehicles on the route are both pickup and delivery business, each customer demand is met once, the appropriate users will be allocated according to the load situation of the vehicle, to ensure that each customer demand is met, greatly avoiding the occurrence of empty vehicles, and the repetition of the pickup vehicles has been improved.

The parameters and variables used in the article mean as below: I is the warehouse collection ($i \in 1, 2, 3, \dots, m$); J is the customer collection ($j \in J, j=1, 2, 3, \dots, n$); K indicates the vehicle collection; $V = I \cup J$ represents a node collection; F represents the warehouse construction cost; W_i indicates the maximum i warehouse capacity; Q indicates the maximum vehicle capacity; C_0 means the vehicle activation cost; d_{ij} represents the distance of the node i from the j ; p_j represents the j customer pickup demand; d_j represents j customer delivery needs; W_{ijk} represents the additional variables, Vehicle k in the Arc (i, j) Load weight; CI means unit cost of distance; UIK represents the auxiliary variables for eliminate sub-circuits; X_{ijk} indicates the vehicle is arc (i, j) for 1 otherwise is 0; y_i means that the warehouse i is enabled for 1 Otherwise is 0; Z_{ij} indicates the customer j by a warehouse i service of 1 Otherwise is 0.

The model main function is as follows:

$$\min \sum_{i \in I} F y_i + \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} C_0 X_{ijk} + \sum_{i \in I} \sum_{j \in J} \sum_{k \in K} C_1 d_{ij} X_{ijk} \quad (1)$$

$$\sum_{j \in J} z_{ij} d_j \leq W_i y_i, \quad \forall i \in I \quad (2)$$

$$\sum_{j \in J} z_{ij} p_j \leq W_i y_i, \quad \forall i \in I \quad (3)$$

$$\sum_{j \in J} X_{ijk} d_j \leq Q, \quad \forall i \in I, k \in K \quad (4)$$

$$\sum_{j \in J} X_{ijk} p_j \leq Q, \quad \forall i \in I, k \in K \quad (5)$$

$$0 \leq W_{ijk} \leq Q, \quad \forall i \in I, j \in V, k \in V \quad (6)$$

$$\sum_{i \in V} W_{ijk} \cdot X_{ijk} - d_j + p_j = \sum_{i \in V} W_{ijk} \cdot X_{ijk} \quad (7)$$

$$\sum_{i \in I} \sum_{k \in V} X_{ijk} = 1, \quad \forall j \in J \quad (8)$$

$$\sum_{i \in I} \sum_{j \in J} X_{ijk} = 1, \quad \forall k \in K \quad (9)$$

$$\sum_{i \in I} Z_{ijk} = 1, \quad \forall j \in J \quad (10)$$

$$X_{ijk} = 0, \quad \forall i \in I, j \in J, k \in K \quad (11)$$

$$\sum_{i \in V} x_{ijk} = \sum_{i \in V} x_{jik}, \quad \forall j \in V, k \in K \quad (12)$$

$$U_{ik} - U_{jk} + N \cdot X_{jik} \leq N, \quad \forall i, j \in J, k \in K \quad (13)$$

$$\sum_{p \in J} x_{ipk} + \sum_{p \in \{V, I, J\}} x_{pjk} \leq 1 + Z_{ij}, \quad \forall i \in I, j \in J, k \in K \quad (14)$$

$$\sum_{j \in J} Z_{ij} \geq y_i, \quad \forall i \in I \quad (15)$$

$$Z_{ij} \leq y_i, \quad \forall j \in J, i \in I \quad (16)$$

$$\sum_{k \in K} \sum_{j \in J} x_{ijk} \geq y_i, \quad \forall i \in I \quad (17)$$

$$\sum_{j \in J} x_{ijk} \leq y_i, \quad \forall i \in I, k \in K \quad (18)$$

$$X_{ijk} \in \{0, 1\} \quad (19)$$

$$y_i \in \{0, 1\} \quad (20)$$

$$Z_{ij} \in \{0, 1\} \quad (21)$$

Target function(1)indicates the minimum cost, Including the warehouse construction costs, vehicle activation cost and vehicle transportation costs; Formula (2)(3) indicates that the delivery and pickup needs of the customers of the warehouse shall not exceed the capacity of the warehouse; Formula(4)(5) indicates that the delivery and pickup requirements of the vehicle serving shall not exceed the capacity of the vehicle; Formula(6)(7) represents the node supply and demand balance constraint and the warehouse service of each customer when the capacity of the vehicle can't exceed the capacity of the vehicle; Formula(8) means that there is and only one car for each customer; Formula(9) means that each vehicle is enabled up to once per vehicle; Formula(10) means that each customer can only be served by one warehouse; Formula(11) means that any two warehouses have no lines; Formula(12) represents the node flow balance; Formula (13) represents elimination sub-loop; Formula(14) indicates that only this customer is assigned to the warehouse will vehicles pass through the customer; Formula(15)~(18) represents the decision relationship between variables and the(16) means that the customer will be served if the warehouse is open. Formula (17) and (18) indicate that cars from the warehouse is open. Formula(19)~(21) means that the decision variable should be between 0 and 1.

3 SOLVING A HYBRID IMMUNE ALGORITHM FOR LRSPD

Immune algorithm is based on the intelligent optimization method of the biological immune system. The proposed algorithm targets the diversity of immune system mechanisms to maintain population diversity, so that it can overcome the “precocity” problem in the general optimization process, and the mechanism of its unique memory bank can eventually obtain the global optimal solution. Under the choice of the antigen, the affinity between the antibody and the antigen is increasing and it eventually can produce the most effective antibodies against the antigen. From a problem-solving perspective, if the antibody is regarded as the result of the problem-solving problem then the process of immune optimization is the process of constantly seeking the antibodies with the greatest affinity. Immune algorithms have been widely used to solve various site selection problems therefore the article tries to improve the proposed algorithm used to solve the LRSPD problem to verify the efficiency of the hybrid immune algorithm.

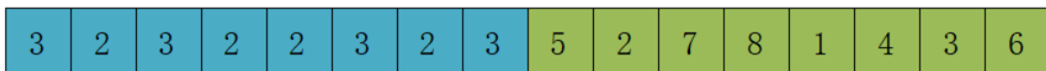
According to the specific features of the LRPSPD model and by combining the original immune algorithm with the greedy clustering algorithm and neighborhood search, we design a hybrid immune algorithm for solving LRPSPD. Firstly, the greedy clustering algorithm is used to replace the generation of initial solution to generate relatively high quality initial solution; Then the original immune algorithm is used to evaluate and rank the antibodies; Finally, the immune operation of the original algorithm is improved by the neighborhood search operation to seek a better solution. According to the large changes in neighborhood search, the mechanism features of the initial solution and the specific operation under each neighborhood are different. One neighborhood operation is mainly for the customer warehouse and the other for the customer distribution order. Both operations influence each other and play a crucial role in the improvement of the solution. Keep the cross-operation of the algorithm, increase population diversity, and prevent premature convergence.

The hybrid immune algorithm designed in the article mainly consists of three parts: a) Initialization. The article uses greedy clustering strategy to cluster customer groups. The warehouse is then allocated according to the center of each cluster and served by the vehicles in the warehouse. Each feasible solution is an antibody, which uses the strategy to generate the initial antibody group, and then combined with the characteristics of the LRPSPD model, a certain number of antibodies will meet the conditions, so that the algorithm can better optimize the iteration. b) Antibody evaluation. The article evaluates the antibody by calculating the affinity between the antibody and the antigen which includes the degree of matching between the antibody and the antigen, and the degree of similarity between the antibody and the antibody. The higher the matching degree of the antibody and the antigen, the lower the similarity between the antibodies, the higher the antibody quality, several high quality of antibodies are stored in the memory bank, so that the algorithm can quickly converge to the optimal solution. c) Immune operation. In immune operations, the algorithm has designed two operations to produce new individuals with large differences. Mainly for cross-operation and neighborhood search operations. This preserves the diversity of individuals in the antibody population and prevents precocious convergence.

3.1 Representation of Solution

At the same time, the delivery site route problem can form a solution of $n \times 2$, the solution has two main parts, respectively marked as X and Y, with different colors. Among them, X(gray) represents the customer ownership relationship with the warehouse, Y(orange) represents the initial order of customers. This article assumes $m=3$, $n=8$, then there are 3 warehouses, 8 customers, the solution length of 16. The other operations of the solution are based on the initial solution. Figure 1 is an initial solution.

Figure 1.



From this solution, warehouse #1 of the #3 warehouses is not open, Eight customers were assigned to warehouses #2 and #3, Customer number 1368 is served by Warehouse #3, Customer number: 2457 is served by Warehouse #2; The customer service order under Warehouse 3 is 8136, Customer service order of Warehouse #2 is 5274; By this way of encoding, Customer order, Customer ownership, Whether the warehouse is open is clearly visible, Build distribution routes based on the order of the customers served by each warehouse, The enabling depot then dispatches the vehicle for service.

3.2 Generation of the Initial Solution

The quality of the initial solution plays a crucial role in the algorithm convergence, and the initial solution is generated according to the greedy clustering algorithm. The specific steps are as follows:

(1) Clustering of customers

The article uses the idea of the greedy algorithm. Depending on the capacity of the vehicle, customer delivery needs and pickup needs. And the distance between the customers to cluster the customers. A basic clustering process can be described as: a) Generates an empty cluster (the cluster capacity should be less than or equal to the capacity of the vehicle). Select one customer from all customers to join the cluster and eliminate the customer in the customer index; b) Select the customer with the closest distance from the remaining customers to judge whether the vehicle capacity constraint is satisfied, and if it is satisfied, the customer will be added to the current cluster; Otherwise, the current cluster does not consider this customer; The c) repeat b) process continues to select the customer closest to the last customer included in the cluster. And to determine whether the vehicle's capacity constraints are met. Until the remaining capacity of the current cluster is insufficient to serve the new customer. A cluster complete. The next cluster allocation process is based on process a)~c) until all customers are assigned.

(2) Select an open warehouse and assign customer clustering

First, the cluster center of gravity is calculated by formula (22) from the coordinates of each customer (22), and formula (23) the distance of each warehouse (23).

$$G(X_c, Y_c) = \left(\frac{\sum_{j \in C} X_j}{n_c}, \frac{\sum_{j \in C} Y_j}{n_c} \right) \quad (22)$$

$$dist_{iG} = \sum_{c \in j} \sqrt{(X_i - X_c)^2 + (Y_i - Y_c)^2}, i \in I \quad (23)$$

After that, a warehouse opening is then randomly selected and the cluster nearest to it, the next cluster and so on. Until the warehouse capacity is insufficient, the assigned cluster is marked and selected clusters excluded in the cluster index; and the next warehouse is randomly selected and open to the above process until all clusters are assigned. Finally, all the open warehouses are found to randomly sort the customers under their ownership, and to generate the initial solution.

3.3 The Antibody was Evaluated and Sorted

The antibody was evaluated by caculating affinity of the antibody and antigen. The calculation steps are as follows:

(1) The affinity of the antibody and antigen is calculated

According to LRSPD model, reciprocal of target function is selected as affinity function, which indicates identification degree of antibody to antigen. Furthermore, affinity function is minimum cost function, and the smaller the cost, the higher the quality of the solution.

$$Affinity = \frac{1}{\min f} \quad (24)$$

(2) The concentration among antibody is determined

The similarity among antibodies is acquired according to its affinity, and antibody similarity is calculated by RCB matching algorithm. In RCB method, if r continuous bit of two antibodies is similar, two antibodies are considered similar antibodies. The similarity can be acquired in terms of Eq. (25) and Eq. (26). Where, $S_{\alpha,\beta}$ is the same bit number of two antibody, and len is the length of antibody. Besides, r is threshold value of similarity.

$$ASim_{\pm,\beta} = \frac{S_{\pm,\beta}}{len} \quad (25)$$

$$BSim_{\pm,\beta} = \begin{cases} 1, & ASim_{\pm,\beta} > r \\ 0, & Others \end{cases} \quad (26)$$

The concentration of antibody α is the similarity of antibody α and antibody population, and it can be determined by Eq. (26). Where, Num is the sum of antibody.

$$Con_{\pm} = \frac{\sum_{j \in J} BSIm_{\alpha,j}}{Num} \quad (27)$$

(3) Expected reproduction probability of antibody is calculated

The expected reproduction probability is important standard to judge antibody quality, and it depends on affinity (the antibody and antigen) and antibody concentration. Where, μ is assessment parameter of population diversity.

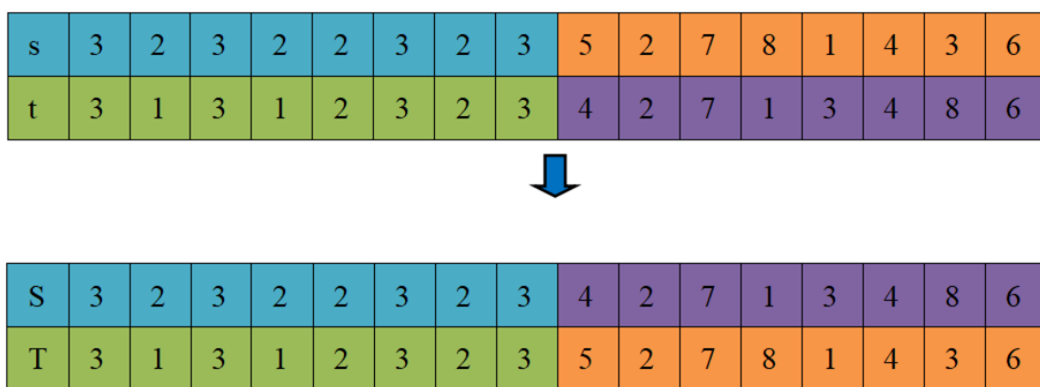
$$Con_{\pm} = \mu \frac{Affinity}{\sum Affinity} + (1 - \mu) \frac{Con_{\alpha}}{\sum Con_{\alpha}} \quad (28)$$

3.4 Immune Operation

(1) Cross

In order to find the balance between local search capability and global search capability for the algorithm, there are two cross strategies, including cross operation of antibody population and its father and internal cross in father population. The probability of the antibody performing each cross strategy is 50%. For the cross operation of antibody population and its father, One individual is respectively selected from the memory base and antibody base in terms of roulette strategy. X part of the selected antibody is remain unchanged, and Y part of the antibody finish cross operation, which is shown as Fig.2.

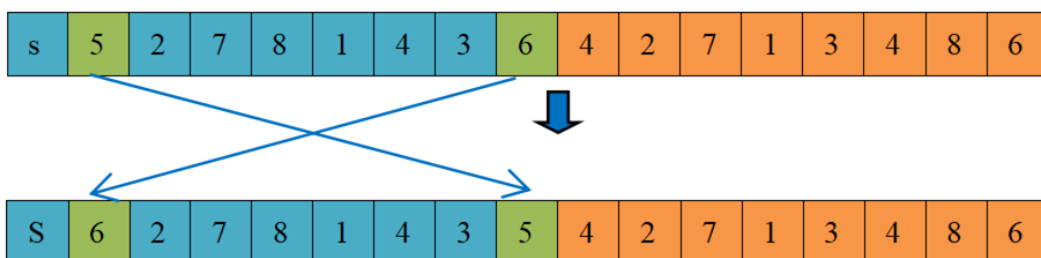
Figure 2. Cross operation



- Step1. Antibody s is selected from memory base, and antibody t is selected from antibody polulation. X part and Y part of each selected are marked with different colour.
 - Step2. Antibody s and t are divided into two part accorting to color mark, and the Y part of two finsh exchage.
 - Step3. X part of antibody s and Y part of antibody t form new antibody S .
 - Step4. X part of antibody t and Y part of antibody s form new antibody T .
- (2) Neighborhood search mechanism

An antibody is selected from father population, and its neighborhood is searched if the antibody corresponding random number exceeds mutation probability. The specific neighborhood serch mechanism includes X neighborhood search (customer and warehouse) and Y neighborhood search (customer service sequence). There are swap, insert, reverse, single point mutation, multipoint mutation

Figure 3. Swap operation



and depot abandonment in X neighborhood search. Y neighborhood search is consist of swap, insert and reverse. Swap operation is shown as Fig.3. X part of an antibody is selected randomly, and two positions are selected from X part. In Fig.3, green point is selected customers, which are respectively 2th and 8th.

4. CASE STUDY

In order to verify the effectiveness of the hybrid immune algorithm in solving the LRSPD model, this paper generates two groups of small-scale data by imitating the Srivastava86-8x2 data in literature (Wang X., F.,2014). The number of customers and warehouses is set. The location information of specific data points is randomly generated by the program and saved as the test data set. The customer demand generation method is as follows: the original data demand is d , $0.5d$ is selected as the lower bound of demand, $1.5d$ as the upper bound of demand, and then an integer is randomly generated in the range of $[0.5d, 1.5d]$ to replace the original data as the new delivery demand. After that, the SN separation method is used to generate a new pickup demand p . Other relevant data are shown in Table 1.

SN separation method, namely the separation method of Salhi and Nagy. In the calculation instance, there are the coordinates of the customer and the warehouse, and the demand of the customer. The method of SN separation to generate the demand for delivery is as follows: firstly, the minimum ratio of horizontal and vertical coordinates of each customer is calculated, that is, $r_i = \min(x_i/y_i, y_i/x_i)$, $\forall i \in N$; then, the actual delivery demand q_i is used to generate the instance delivery demand d_i , and the generation mode is $d_i = q_i \cdot r_i$, and then calculate the instance pickup demand $p_i = q_i \cdot d_i$. This separation method is called X-type separation method, and another Y-type separation method is realized by replacing X-type pickup and delivery demand. The paper mainly uses the X-type instance improved by the SN separation method for testing.

Table 1. The relative data of tested instances

Number of customers	Number of candidate warehouses	Vehicle capacity	Warehouse capacity	Warehouse activation cost	Vehicle activation cost
8	2	200	1000	35	20
15	5	200	1000	40	20

4.1 Calculation instance parameter setting

Set the relevant parameters of the hybrid immune algorithm, the length of the antibody: $\text{length} = n * 2$, which is determined by the number of customers n . In this paper, the hybrid immune algorithm is used to solve the Prodhon instance of CLRP. The number of iteration is set to 1000, and the experiment is repeated 10 times. It is found that the algorithm converges at the 600th iteration, which is close to the optimal solution, so the iteration number is set to 600. The threshold value of similarity between antibodies is $r = 0.4$, which aims to ensure the diversity of the population and the quality of the antibodies. In addition, the parameters that profoundly affect the performance of the algorithm are discussed and sensitivity analyzed through the Taguchi Design of Experiment (DOE) (Prins C., Prodhon C., & Calvo R., W.,2006). Using the orthogonal experiment table L16 to carry out the experiment, solve the 20-5-1 instance of the Prodhon standard instance, and give 5 key parameters and 4 reasonable level values for each parameter. These five parameters are the population size (Size-pop), the memory storage capacity (Over-best), the diversity evaluation parameter (p-s), the

Table 2. Parameter level

Parameters	Levels			
	1	2	3	4
Size-pop	20	30	40	50
Over-best	4	6	8	10
p-s	0.4	0.5	0.6	0.7
p-cross	0.3	0.4	0.5	0.6
p-mutation	0.6	0.7	0.8	0.9

Table 3. Response value of each parameter

Levels	Parameters				
	Size-pop	Over-best	p-s	p-cross	p-mutation
1	322.84	329.14	314.64	365.22	361.84
2	360.25	315.48	316.69	314.14	312.03
3	313.13	359.88	323.17	319.06	319.74
4	315.05	306.76	356.75	312.83	317.65
Range	47.12	53.12	42.11	52.38	49.41
Level	4	1	5	2	3

Figure 4. Over-best Parametric level analysis



cross probability (p-cross), and the mutation probability (p-mutation). After several experiments, the parameter level and response value data were recorded in a table, as shown in Table 2-4.

According to the response value of significance level of each parameter in Table 3, the horizontal trend chart of the impact factor is obtained. Because there are too many analysis parameters, the paper mainly displays the horizontal analysis chart of parameter Overbest, as shown in Fig.4.

Table 4. Orthogonal array and AVG statistics

Parameter combination number	Levels					AVG
	Size-pop	Over-best	p-s	p-cross	p-mutation	
1	1	1	1	1	1	339.31
2	1	2	2	2	2	315.95
3	1	3	3	3	3	326.81
4	1	4	4	4	4	309.28
5	2	1	2	3	4	336.36
6	2	2	1	4	3	306.20
7	2	3	4	1	2	487.76
8	2	4	3	2	1	310.67
9	3	1	3	4	2	325.07
10	3	2	1	3	1	309.62
11	3	3	4	2	4	314.16
12	3	4	2	1	3	303.66
13	4	1	4	2	3	315.81
14	4	2	3	1	4	330.15
15	4	3	2	4	4	310.79
16	4	4	1	3	2	303.45

Table 5. Test case solution

The size of the calculation instance	The lower bound of the solution		Operation hours	
	HIM	CPLEX	HIM	CPLEX
8-2	886.29	873.58	17.53	55.23
15-5	456.32	456.32	23.22	80.96

As can be seen from Table 3, the range of the parameter Over-best is the largest, which indicates that the selection of memory storage capacity has the greatest impact on the algorithm performance, and it proves that the excellent antibodies in memory storage play an important role in the early convergence of the algorithm. Considering comprehensively, the parameter combination proposed in the paper is Size-pop=50, Over-best=6, p-s=0.6, p-cross=0.4, p-mutation=0.7 .

4.2 Calculation Instance Results and Analysis

The hybrid immune algorithm is set according to the above parameters. The experimental environment is Intel(R)Corei5-8250uCPU@1.60GHz, 8.00GB memory, 64-bit Windows10 operating system, and MatlabR2020 is used for programming. The solution results of the two groups of test cases obtained in the instance are shown in Table 5.

Figure 5. Comparison of objective function values of algorithms

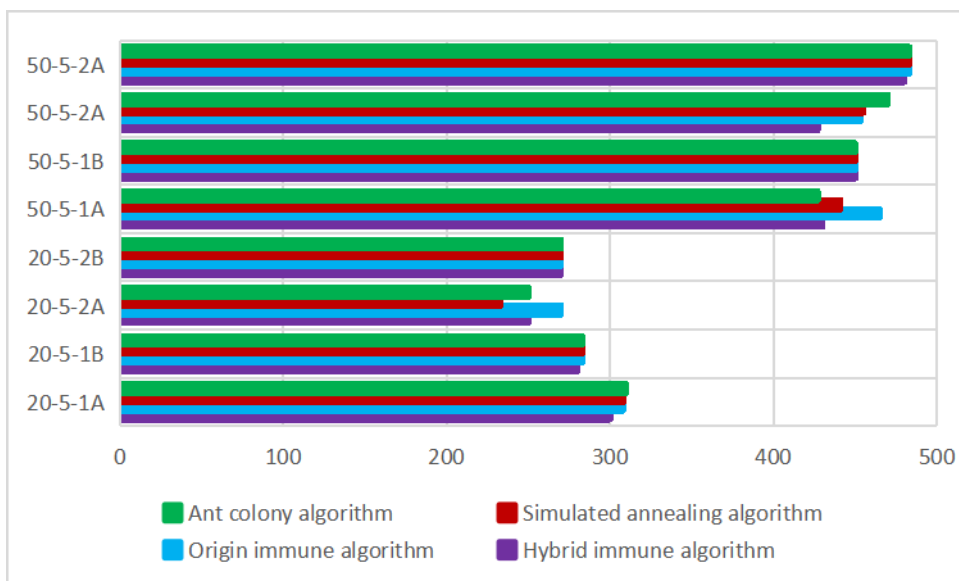
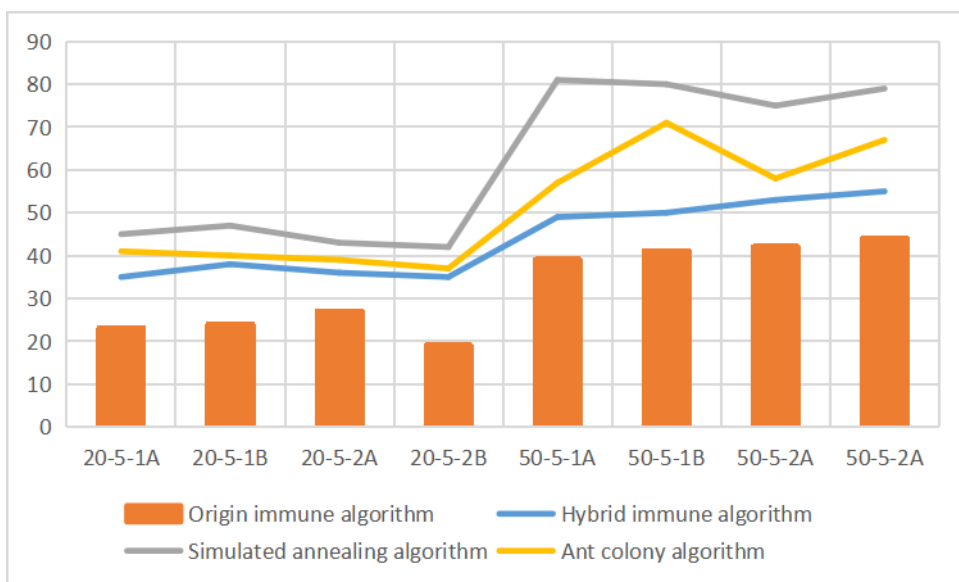


Figure 6. Algorithm run-time comparison



It is not difficult to see from Table 5 that the performance of this algorithm is superior in solving small-scale instances, and it can find a relatively good optimal solution in a relatively short time. At the same time, the CPLEX program designed according to the model in this paper runs smoothly, and the lower bound of the solution can be obtained in a good time. The improved HIM algorithm designed in this paper shows excellent performance in small-scale examples, and the advantages of the meta-heuristic algorithm over the precise algorithm gradually appear. In order to test the effectiveness of the algorithm for the medium-sized instances, the paper solves the medium-sized instances of 20 customers from 5 warehouses, and fully verifies the performance of the hybrid immune algorithm by solving the two kinds of small and medium-sized instances.

The medium-sized instance adopted in this paper is the ProDHON instance in the CLRP standard instance library, which is improved to adapt to the solution of LRSPD problem, and the SN separation method mentioned in this paper is used to carry out the instance transformation. After that, the hybrid immune algorithm, the original immune algorithm, the simulated annealing algorithm and the ant colony algorithm were used to solve the problem respectively. Each example was solved 20 times, and the optimal solution and the time spent were recorded. The solution results were shown in Table 6, and the comparison of some key values was shown in Figure 5 and 6.

Table 6. Comparison of results of numerical examples

Number	The instance name	Hybrid immune algorithm		Original immune algorithm		Simulated annealing algorithm		Ant colony algorithm	
		Total cost	Operation hours	Total cost	Operation hours	Total cost	Operation hours	Total cost	Operation hours
1	20-5-1a	311.12	33.90	320.36	28.84	300.59	44.85	310.53	41.66
2	20-5-1b	283.09	35.60	295.15	29.34	285.09	46.82	289.34	39.24
3	20-5-2a	275.37	33.92	290.60	31.20	260.58	45.57	279.63	38.32
4	20-5-2b	291.83	33.94	300.42	28.34	292.87	45.12	294.36	36.63
5	50-5-1a	453.37	51.92	493.82	48.36	465.66	86.62	436.34	60.51
6	50-5-1b	483.50	52.57	488.23	49.34	482.72	87.11	484.26	76.42
7	50-5-2a	464.96	56.72	500.32	52.69	511.12	83.77	516.32	65.77
8	50-5-2b	517.60	59.37	544.36	53.46	538.04	86.23	540.21	69.35

Because the number of iterations is not a good description of the performance of the algorithm. This paper analyzes the results of instance from two aspects: the quality of solution and the running time of algorithm. It is not difficult to see from the above figure that the quality of the solutions obtained by the four algorithms is excellent for a small instance of 20 customers. The solution time of the two immune algorithms is obviously better than that of the simulated annealing algorithm, and the performance of the ant colony algorithm is relatively stable. The original immune algorithm has the fastest solution speed, while the hybrid immune algorithm has a relatively slower solution speed, because the execution of the immune operation of the hybrid immune algorithm takes more time. It can be seen from the solution of the medium instance for 50 customers that the quality of the solution obtained by the hybrid immune algorithm is the best, which is obviously better than other algorithms(Ferreira K. M., & De Queiroz T. A.,2018). In terms of the solution time of the algorithm, the original immune algorithm is still the fastest.

Through the comparison of 8 instances and 4 algorithms, it can be seen that the hybrid immune algorithm can get a good solution in a relatively short time, and the performance of the algorithm

Figure 7. Operation hours comparison of algorithm

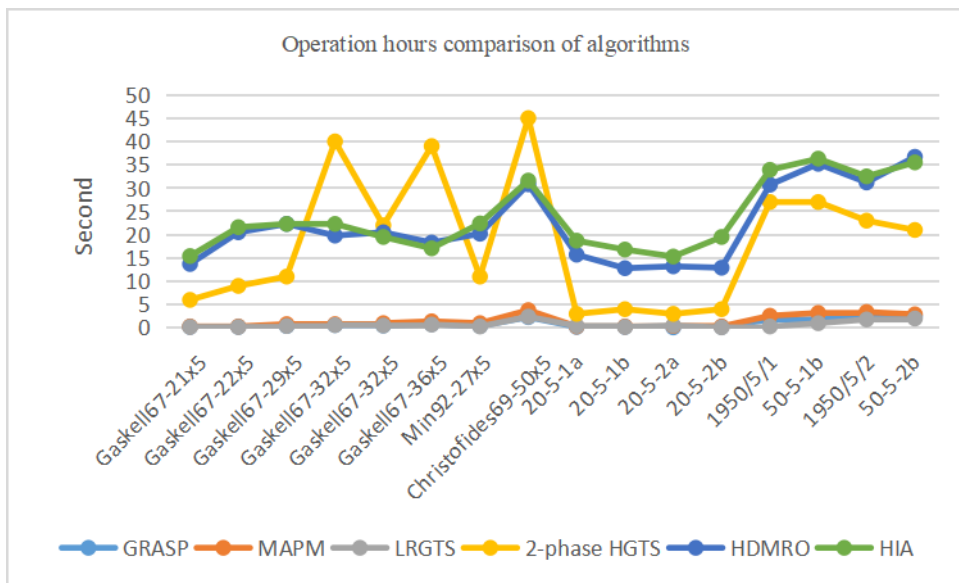


Table 7. Algorithm efficiency comparison

Data	GRASP	MAPM	LRGTS	2-phase HGTS	HDMRO	HIA
Gaskell67-21x5	429.6	424.9	424.9	424.9	424.9	424.9
Gaskell67-22x5	585.1	611.8	587.4	585.1	585.1	585.1
Gaskell67-29x5	515.1	512.1	512.1	512.1	512.1	516.2
Gaskell67-32x5	571.9	571.9	584.6	562.2	568.5	562.2
Gaskell67-32x5	504.3	534.7	504.8	504.3	504.3	504.3
Gaskell67-36x5	460.4	485.4	476.5	460.4	479.5	479.5
Min92-27x5	3062.0	3062.0	3065.2	3062.0	3082.0	3082.0
Christofides69-50x5	599.1	565.6	586.4	580.4	565.6	565.6
20-5-1a	55021	54793	55131	54793	54793	54793
20-5-1b	39104	39104	39104	39104	39104	39104
20-5-2a	48908	48908	48908	48945	48908	48908
20-5-2b	37542	37542	37542	37542	37542	37542
50-5-1	90632	90160	90160	90402	90160	90540
50-5-1b	64761	63242	63256	64073	66468	66468
50-5-2	88786	88298	88715	89342	89809	89903
50-5-2b	68042	67893	67698	68479	67340	68340
Optimal times	7	10	6	10	11	9

is quite excellent. In order to further verify the solving efficiency of the hybrid immune algorithm, this paper selects 16 groups of data from the CLRP standard instance library for solving, and the comparison effect with the other four algorithms is shown in Fig.7 and Table 7.

According to the data in Table 7, it is relatively intuitive to compare and analyze the algorithm from two aspects. On the one hand, according to the quality of their solutions, the paper can get 9 groups of optimal solutions. GRASP(Qiao J. F., Li F., & Yang S. X., et al,2020)(greedy random adaptive search algorithm) can get 7 groups of optimal solutions. MAPM(Rahmani Y., Cherif-khettaf W. R., & Oulamara A.,2016)(cultural gene algorithm based on crowd management) can get 10 groups of optimal solutions, which is an algorithm with the best quality. LRGTS (tabu search algorithm based on Lagrange relaxation) can obtain 6 groups of optimal solutions, and 2-phase HGTS (two-stage heuristic scatter tabu search) can also obtain 10 groups of optimal solutions. On the other hand, due to the differences in computer hardware and compilation software, the solving time is also relatively different, and there is no unified standard to measure. The first three algorithms quoted in the paper are all coded by C++, while the improved algorithm in the paper is programmed by MATLAB. Compared with the hybrid discrete fuzzy algorithm, it can be seen that the solving time is no different, but it is worthy of recognition that HIM algorithm can solve relatively excellent solutions in effective time. Considering comprehensively, the performance of the algorithm is convincing.

5. CONCLUSION

The paper try to solve location-routing problem of simultaneous delivery and pickup. Initial solution was generated by greedy clustering algorithm. Furthermore, the evaluation and rank of antibody was realized by original immune algorithm, and immune operation was improved by neighborhood search algorithm. The hybrid immune algorithm can improve the quality of the solution while ensuring the diversity of the population, and obtain a good solution to avoid falling into the local optimum. By comparing and analyzing the optimization results of the original immune algorithm, the solution quality, convergence speed and decision-making effect of the algorithm have been significantly improved. The solution solved by the new algorithm can reduce the cost of distribution, improve the utilization efficiency of vehicle capacity, and reduce the waste of resources. It has a strong reference value for logistics companies that have simultaneous delivery and retrieval services.

Hybrid discrete fuzzy algorithm has come into public view in recent years, and it has shown good performance in solving LRP problems. In the future research, we can try to compare and analyze it with hybrid immune algorithm, which has great reference significance for future research work.

FUNDING

The article was Supported by Science and Technology Project of Henan Province (No. 202102310323; No. 212102310264)

REFERENCES

- Drexl, M., & Schneider, M. (2015). A survey of variants and extensions of the location-routing problem. *European Journal of Operational Research*, 241(2), 283–308. doi:10.1016/j.ejor.2014.08.030
- Ferreira, K. M., & De Queiroz, T. A. (2018). Two effective simulated annealing algorithms for the Location-Routing Problem. *Applied Soft Computing*, 70, 389–422. doi:10.1016/j.asoc.2018.05.024
- Karaoglan, I., Altiparmak, F., & Kara, I. (2012). The Location-routing Problem with Simultaneous Pickup and Delivery. *Formulations and a Heuristic Approach*, 40, 465–477.
- Leng, L., & Zhao, Y., & Zhang, C. (2019). Quantum heuristic for solving the low-carbon location-routing problem. *Jisuanji Jicheng Zhizao Xitong*, 8, 1–22.
- Prins, C., Prodhon, C., & Calvo, R. W. (2006). Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *A Quarterly Journal of Operations Research*, 4, 221–238.
- Prodhon, C., & Prins, C. (2014). A survey of recent research on location-routing problems. *European Journal of Operational Research*, 238(1), 1–17. doi:10.1016/j.ejor.2014.01.005
- Qiao, J. F., Li, F., Yang, S. X., Yang, C., Li, W., & Gu, K. (2020). An adaptive hybrid evolutionary immune multi-objective algorithm based on uniform distribution selection. *Information Sciences*, 512, 446–470. doi:10.1016/j.ins.2019.08.032
- Qiao, J. F., Li, F., Yang, S. X., Yang, C., Li, W., & Gu, K. (2020). An adaptive hybrid evolutionary immune multi-objective algorithm based on uniform distribution selection. *Information Sciences*, 512, 446–470. doi:10.1016/j.ins.2019.08.032
- Rahmani, Y., Cherif-khettaf, W. R., & Oulamara, A. (2016). The two-echelon multi-products location-routing problem with pickup and delivery. *International Journal of Production Research*, 54(4), 999–1019. doi:10.1080/00207543.2015.1040127
- Schneider, M., & Drexl, M. (2017). A survey of the standard location-routing problem. *Annals of Operations Research*, 259(1-2), 389–414. doi:10.1007/s10479-017-2509-0
- Sun, Q., & Yang, Z. (2017). Study on multi-vehicle routing problem with simultaneous pickup and delivery. *Journal of Transportation Engineering and Information*, 15, 100–104.
- Wang, X. F. (2014). A location-routing problem with simultaneous pickup and delivery in urban-rural dual-directions logistics network. *Advanced Materials Research*, 756, 3423–3429.
- Yu, V. F., & Lin, S. W. (2014). Multi-start simulated annealing heuristic for the location routing problem with simultaneous pickup and delivery. *Applied Soft Computing*, 24, 284–290. doi:10.1016/j.asoc.2014.06.024
- Yu, V. F., & Lin, S. Y. (2016). Solving the location-routing problem with simultaneous pickup and delivery by simulated annealing. *International Journal of Production Research*, 24(2), 1–24. doi:10.1080/00207543.2015.1085655
- Zhang, B., Pan, Q., Gao, L., Zhang, X., Sang, H., & Li, J. (2017). An effective modified migrating birds optimization for hybrid flow shop scheduling problem with lot streaming. *Applied Soft Computing*, 52, 14–2. doi:10.1016/j.asoc.2016.12.021
- Zhang, X., Fan, H., & Li, J. (2015). Variable neighborhood decentralized search algorithm for simultaneous pickup location routing problem. *Jisuanji Jicheng Zhizao Xitong*, 21, 2535–2548.
- Zhou, M., Li, Y., Wang, Y., Pu, Q., Yang, X., & Nie, W. (2021). Device-to-device cooperative positioning via matrix completion and anchor selection. *IEEE Internet of Things Journal*, 1. Advance online publication. doi:10.1109/JIOT.2021.3109291
- Zhou, M., Wang, Y., Tian, Z., Lian, Y., Wang, Y., & Wang, B. (2019). Calibrated data simplification for energy-efficient location sensing in internet of things. *IEEE Internet of Things Journal*, 6(4), 6125–6133. doi:10.1109/JIOT.2018.2869671