Improved Multi-objective Genetic Algorithm for Two-echelon Vehicle Routing Problem with Soft Time Windows

Xuanying Zhang[1,a,*]

[1]*School of Information Engineering and Automation, Kunming University of Science and Technology, Kunming, China*
a. zhangxuanyingkust@163.com
**corresponding author: Xuanying Zhang*

Keywords: Two-echelon vehicle routing problem, soft time windows, multi-objective, improved genetic algorithm.

Abstract: For the two-echelon vehicle routing problem with soft time windows (2E-VRPSTW). Aiming at minimizing the total distance and maximizing the on-time rate as the goals, a multi-objective genetic algorithm was proposed to solve the problem, and the goal was optimized by bottom-up approach. In the second-echelon vehicle routing, a variable neighborhood search is integrated on the basis of improved multi-objective genetic algorithm(IMOGA)to enhance the local search ability of the algorithm; in the first-echelon vehicle routing, each satellite is determined according to the customers served in the second level The volume of satellite is solved using the nearest neighbor principle and optimized using local search.Experimental results verify the effectiveness of the algorithm.

1. Introduction

With the implementation of "manufacturing power strategy", it is necessary to promote the high-quality development and deep integration of modern logistics industry and advanced manufacturing industry. An effective logistics scheduling optimization algorithm is an important way to create low-cost and efficient logistics services, and is an important research area of the logistics industry. In logistics scheduling, the vehicle routing problem(VRP) is one of the most important problems. Traditional VRP only considers the optimization of the route between the distribution center and the customer. However, in the transportation process, long-distance trucks caused road congestion and increased noise, and caused air pollution . In addition, major cities in China have proposed regulations on traffic restrictions. Therefore, in urban transportation, the goods must first be transported from the depot to the satellite by the heavy trucks, and then transported to the customer by the lower approved truck, thus forming a two-echelon vehicle routing problem.

For 2E-VRP, Rainic et al.[1] first proposed the mathematical model of the problem in 2009 and solved it with an exact algorithm. Meihua et al.[2] proposed a hybrid ant colony optimization algorithm (HACO) for 2E-VRP and transformed the problem into VRP through customer classification. Find the optimal solution for some test sets in a short time. Hemmelmayr et al.[3] proposed an adaptive large neighbourhood search (ALNS) heuristic algorithm for 2E-VRP, and added a new local search operation to the algorithm to enhance its performance. Cihan et al.[4] established a TSVRPATWs model and a data set for the two-stage vehicle routing problem with arc time windows (TSVRPATWs) considering route time windows, and proposed a cultural genetic algorithm to solve A better solution was found in a reasonable time. Kergosien et al.[5] used the logistics system of a French hospital as the background, and used the improved genetic algorithm and tabu search algorithm for 2E-VRP to expand the data set based on the actual data. Good solution to this problem.

This paper addresses the two-echelon vehicle routing problem with soft time windows (2E-VRPSTW). With the goal of minimizing the total distance and maximizing the punctuality, an improved multi-objective genetic algorithm is proposed to solve this problem. Experiments prove the effectiveness of the algorithm.

2. Problem Description

An example of the 2E-VRPSTW describes as follows: D is the central warehouse, S1 and S2 are the satellites, and c1 to c9 are the customer points. In the first-echelon of the route, the first-echelon vehicle departs from the depot and returns to each satellites; in the second-echelon route, the second-echelon vehicle departs from one satellite and serves the customer, and then returns to the same satellite.

In order to describe the model more clearly, we make the following assumptions: (1)There is only one depot. (2)The driving speed of each vehicle is equal to 1, that is, the Euclidean distance between two points is equal to the driving time of the two places. (3)The customer's goods cannot be divided, and each customer is served by only one car.

The mathematical model is as follows:

$$\min f_1 = \sum_{i \in V_d \cup V_s} \sum_{j \in V_d \cup V_s} c_{ij} x_{ij} + \sum_{l \in V_s \cup V_c} \sum_{m \in V_s \cup V_c} c_{lm} y_{lm} \tag{1}$$

$$\max f_2 = \frac{ontimec}{totalc} \tag{2}$$

s.t.

$$\sum_{j \in N_d \cup N_s} x_{ij} = 1, \forall i \in V_s \tag{3}$$

$$\sum_{j \in V_d \cup V_s} x_{ij} = \sum_{j \in V_d \cup V_s} x_{ji}, \forall i \in V_s \tag{4}$$

$$0 \le Q_{dsi} \le C_{V1}, \forall i \in V_d \cup V_s \tag{5}$$

$$\sum_{j \in V_s} s_{ij} = 1, \forall i \in V_c \tag{6}$$

$$\sum_{j \in V_s \cup V_c} y_{ij} = 1, \forall i \in V_c \tag{7}$$

$$\sum_{j \in V_s \cup V_c} y_{ij} = \sum_{j \in V_s \cup V_c} y_{ji}, \forall i \in V_c \tag{8}$$

$$0 \le Q_{sci} \le C_{V2}, \forall i \in V_s \bigcup V_c \quad (9)$$

$$0 \le d_i \le C_{V2}, \forall i \in V_c \quad (10)$$

$$t_i = t_j + st_j + c_{ij}, \forall i \in V_s \bigcup V_c, \forall j \in V \quad (11)$$

$$ontimec = \sum_{i \in N_c} otr_i \quad (12)$$

Formula (1) and formula (2) are optimization goals, formula (1) means minimizing the total vehicle path in the first and second stages, and formula (2) means maximizing the punctuality rate of vehicles reaching the customer; (3) to equation (14) are constraints,. Equation (3) indicates that each satellite can only be served once, Equation (4) indicates the node's vehicle flow balance constraints, Equation (5) indicates that the first-level vehicles cannot be overloaded during transportation, and Equation (6) indicates each customers can only be assigned to one satellite. Equation (7) indicates that each customer can only be served once. Equation (8) indicates customer point flow balance constraints. Equation (9) indicates that secondary vehicles cannot be overloaded during transportation , Equation (10) indicates that the customer's pickup demand cannot exceed the maximum load of the second-level vehicle, Equation (11) indicates the time when the vehicle reaches point i, and Equation (12) indicates the calculation method for the total number of customers arriving within the time window.The symbol table is shown in Table 1.

Table 1: Symbol description.

Symbol	Description
c_{ij}	Distance between points i, j.
x_{ij}	In the first-echelon, if the vehicle passes i, j is 1, otherwise it is 0.
y_{lm}	In the second-echelon, if the vehicle passes l, m is 1, otherwise it is 0.
ontimec	Number of cities served in time window.
totalc	Total city numbers.
Qds_i	Carrying capacity after the vehicle leaves point i in one-echelon.
CV_1	Maximum load of vehicles in one-echelon.
Qsc_i	Carrying capacity after the vehicle leaves point i in second-echelon.
CV_2	Maximum load of vehicles in two-echelon.
s_{ij}	if satellite j serves customer i, is 1otherwise 0

3. Improved Multi-objective Genetic Algorithm

2E-VRPSTW is a NP-hard problem. Solving such problems often uses a heuristic algorithm. The genetic algorithm searches for the optimal solution by simulating biological evolution. It has good global optimization capabilities. The variable neighborhood descent algorithm can search the solution space in detail. Can make up for the shortcomings of genetic algorithms. Based on multi-objective genetic algorithm and variable neighborhood search, this paper designs a multi-objective hybrid genetic algorithm to solve the problem.

3.1. Encoding and Decoding

Because the route in 2E-VRPSTW consists of two phases, and each satellite serves a part of customers, the solution consists of two parts, π_1 and π_2. 2E-VRPSTW obtains the initial solution in the following way:

Step 1: Assign customers to satellites based on nearest neighbor rules.

Step 2: For each satellite, follow the constraints of the second-echelon to generate a route for the second-echelon $\pi_{2.1} \cdots \pi_{2,N_s}$.

Step 3: According to the customer classification of step 1, get the delivery volume of each satellite, and generate the first-echelon route π_1 according to the nearest neighbor principle and the constraints of the second-echelon.

For example, for the problem of $N_s = 2$, $N_c = 10$,a solution is expressed as follows:

$$\pi_1 = \begin{bmatrix} 0 & 1 & 2 & 0 \end{bmatrix}$$
$$\pi_2 = \begin{bmatrix} \pi_{2.1} & \pi_{2.2} \end{bmatrix}$$
$$\pi_{2.1} = \begin{bmatrix} 0 & 1 & 2 & 3 & 0 & 4 & 5 & 6 & 0 \end{bmatrix}$$
$$\pi_{2.2} = \begin{bmatrix} 0 & 7 & 8 & 9 & 0 & 10 & 0 \end{bmatrix}$$

The first part of π_1 indicates that in the first-echelon of the route, the first-echelon vehicle departs from the depot, serves the satellite1 and satellite2 in turn and returns to the depot; the second section of π_2, the route $\pi_{2.1}$ indicates that in the second-echelon of the route, vehicle1 is from the satellite1 and then serve customers 1, 2, and 3 in turn and return to satellite1. Vehicle2 departs from satellite1 and serve customers 4, 5, and 6 in turn and return to satellite1. The other paths in the second part are interpreted by analogy.

3.2. Population Initialization and Fitness Calculation

This section must be in one column.In the first-generation population, the encoding method described in Section 3.1 generates a random initial solution, which improves the diversity of the population while generating a feasible solution. In the selection process, the fitness of each individual needs to be calculated. Since the multi-objective problem cannot measure the quality of the solution with one value, and the magnitude of each target value is different, the fitness of the solution is calculated using the following formula:

$$f(i) = \frac{otr_i}{otr_{\max}} + \frac{length_{\min}}{length_i} \tag{13}$$

Among them, otr_i is the time window punctuality rate of the ith solution in the population (non-inferior solution set), $otr_{\max}$ is the maximum time window punctuality rate in the population (non-inferior solution set), and $length_i$ is the population of the generation (non-inferior solution set). The total travel distance of the ith solution in the set), $length_{\min}$ is the minimum total travel distance of the generation population (non-inferior solution set). In summary, the fitness of the individuals in the population are all less than 2, and the larger the individual fitness value, the greater the probability of being selected.

3.3. Select

This paper uses the roulette method to select operators. Select_p is the selection probability. The selected pseudo-code as follows:

```
Algorithm:Select
begin:
rand1=a random value between 0 and 1
rand2=a random value between 0 and 1
if rand1<Select_p:
Calculate the fitness of the Pareto set
Normalize fitness values
Choose a solution with roulette method in Pareto set as Parent1
else:
Calculate the fitness of the population set
Normalize fitness values
Choose a solution with roulette method in population set as Parent1
if rand2<Cross_ s:
Calculate the fitness of the Pareto set
Normalize fitness values
Choose a solution with roulette method in Pareto set as Parent2
else:
Calculate the fitness of the population set
Normalize fitness values
Choose a solution with roulette method in population set as Parent2
```

3.4. Crossover

During the crossover process, the two parents selected according to the selection process are crossed as shown in the figure1 below. Remove the 0 in the parent sequence to get the order in which the cities get services. Select two locations in the parent 2 city set (position1< position2), and put the cities before position 1 and after position 2 into the offspring city set. The cities that have not been served in the parent 1 are placed in the city set of the children in the order, and the child city set is decoded according to the constraints,and then according to the delivery of each satellite The first-echelon route is generated according to the nearest neighbor principle.

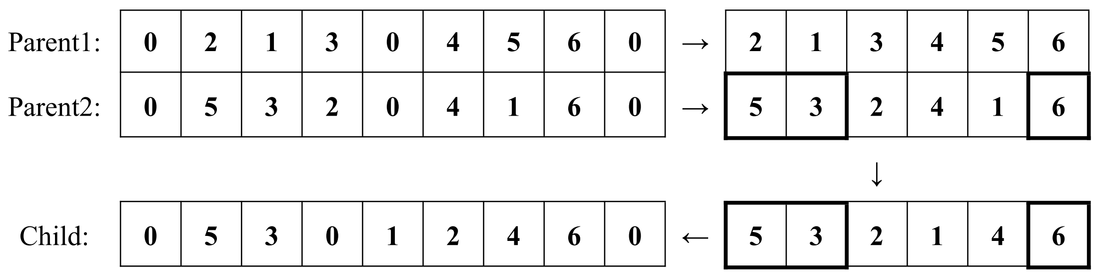

Figure 1: Crossover.

3.5. Mutation

This paper uses a two-point interchange mutation operator to select a solution with a certain probability. Two cities are randomly selected for two-point exchange, and then the city sequence is decoded into a feasible solution. Compare and update non-inferior solution set.

0	**2**	**1**	**3**	**0**	**4**	**5**	**6**	**0**	→	**2**	**1**	**3**	**4**	**5**	**6**
												↓			
0	**2**	**6**	**3**	**0**	**4**	**5**	**1**	**0**	←	**2**	**6**	**3**	**4**	**5**	**1**

Figure 2: Mutation.

3.6. Local search

Non-dominated set have higher probability to find better quality solutions. Therefore, this paper uses a variable neighborhood descent method to optimize the Pareto solution set.

LS1:Swap(1,1):In a satellite, two cities are selected in two routes for exchange.

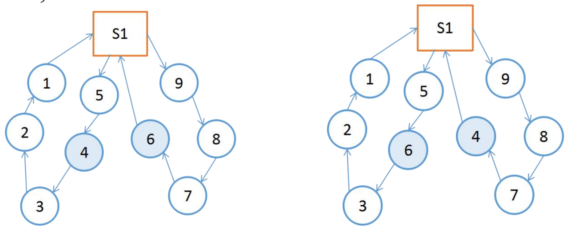

Figure 3: Swap(1,1).

LS2: Shift (1,0): In a satellite, insert a customer from one car into another.

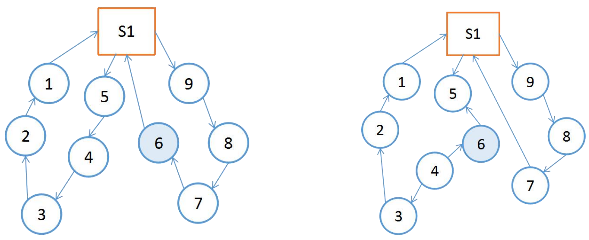

Figure 4: Shift (1,0).

LS3: Exchange: As shown in the figure below, select two cities in a route to exchange.

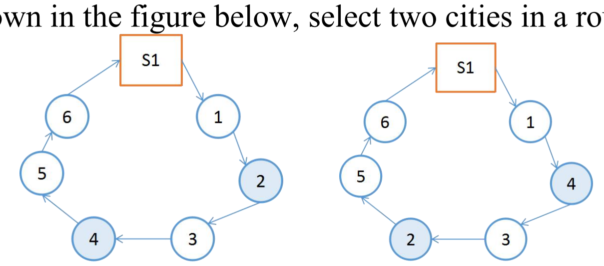

Figure 5: Exchange.

LS4: 2-opt: As shown in the figure, select a section of customers served by a car in a two-phase route ,and reverse it.

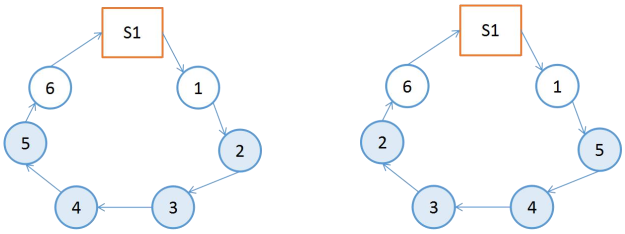

Figure 6: 2-opt.

The process of variable neighborhood decline is as follows:

Step1: Let Neighbourhood structures set = [LS1, LS2, LS3, LS4], and disrupt the order of the set, and make Pareto temp set equal to Pareto set.

Step2: For the corresponding solution in each Pareto temp set, follow the order of the local search operations in the Neighbourhood structures set to perform local operations on the solution, each operation is repeated 5 times, and then update the Pareto set.

4. Performance Metrics

The method proposed by Ishibuchi et al.[6] was used to evaluate the non-dominated solution set. The evaluation index is shown in formula (14).

$$R\_NDS(S_j) = \frac{\left|S_j - \{x \in S_j \,|\, \exists y \in S : y \prec x\}\right|}{|S_j|} \tag{14}$$

S represents the set of non-dominated solution sets generated by all algorithms. $y \prec x$ represents that the solution x is dominated by y. $|S_j|$ represents the number of solutions in the non-dominated solution set. The average AVG of R_NDS is used as the evaluation index for experiments.

The experiment uses Python 3.6 programming to achieve simulation, run on a PC with Intel 7700HQ 2.80 GHz. The test parameters are as follows:Select_p=0.9, Mutation_p=0.1, Popsize=100, For fair comparison, IMOGA and VND_LS and HEDA are run the same time and 20 times.

Table 2: Comparisons of IMOGA and VND_LS and HEDA.

Instance	VND_LS	HEDA	IMOGA
rcdp25-1	0.17	0.13	1.0
rcdp25-2	0.42	0.50	0.70
rcdp50-1	0.13	0.56	0.65
rcdp50-2	0.61	0.32	1.0
rcdp100-1	0.08	0.43	0.80
rcdp100-2	0.11	0.22	1.0
rcdp200-1	0.33	0.43	0.72
rcdp200-2	0.29	0.24	0.87

Considering the 8 different size instances, we compare the IMOGA with the VND_LS and HEDA. For each instance, we run both the same time and obtain the RND_S. Table 2 summarizes the results grouped by each instance.

From Table 2, it can be seen that the results obtained by IMOGA are much better than those of VND_LS and HEDA for every instances. Most of the solution of VND_LS and HEDA are dominated by the solution of IMOGA. Thus, it is concluded that IMOGA has the better performance for 2E-VRPSTW in the given time.

4.1. Conclusions

In this paper, a multi-objective hybrid genetic algorithm is designed to solve the two-echelon vehicle routing problem with soft time windows in urban logistics, and a variable neighborhood descent method is designed to solve it. Aiming at a wide range of genetic algorithms, the initial solution has high diversity, and the local search operator is used to enhance the optimization performance of the variable neighborhood descent algorithm. The experimental results of the extended standard examples show the proposed model and optimization Effectiveness of the algorithm. In addition, because the implementation process of the algorithm is relatively simple, it has more practical significance in the process of urban logistics optimization decision-making.

References

[1] Crainic T G , Ricciardi N , Storchi G . Models for Evaluating and Planning City Logistics Systems[J]. Transportation Science, 2009, 43(4):432-454.

[2] Meihua W , Xuhong T , Shan C , et al. Hybrid Ant Colony Optimization Algorithm for Two Echelon Vehicle Routing Problem[J]. Procedia Engineering, 2011, 15(none):3361-3365.

[3] Hemmelmayr V C , Cordeau J F , Crainic T G . An adaptive large neighborhood search heuristic for Two-Echelon Vehicle Routing Problems arising in city logistics[J]. Computers & Operations Research, 2012, 39(12):3215-3228.

[4] Cihan Çetinkaya, Karaoglan I , Hadi Gökçen. Two-stage vehicle routing problem with arc time windows: A mixed integer programming formulation and a heuristic approach[J]. European Journal of Operational Research, 2013, 230(3):539-550.

[5] Kergosien Y , Lenté, Ch, Billaut J C , et al. Metaheuristic algorithms for solving two interconnected vehicle routing problems in a hospital complex[J]. Computers & Operations Research, 2013, 40(10):2508-2518.

[6] Ishibuchi H, Yoshida T, Murata T. Balance between genetic search and local search in memetic algorithms for multiobjective permutation flowshop scheduling [J]. IEEE transactions on evolutionary computation, 2003, 7(2): 204-223.