

## Research Article

# Models and Methods for Two-Echelon Location Routing Problem with Time Constraints in City Logistics

Peng Yang<sup>1</sup> and Lining Zeng<sup>2</sup>

<sup>1</sup>Hunan University of Finance and Economics, Changsha, Hunan, China

<sup>2</sup>College of Information Science and Engineering, Hunan University, Changsha, Hunan, China

Correspondence should be addressed to Peng Yang; pengyang947@gmail.com

Received 22 September 2018; Accepted 26 November 2018; Published 13 December 2018

Academic Editor: Honglei Xu

Copyright © 2018 Peng Yang and Lining Zeng. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper focuses on the two-echelon location routing problem with time constraints in city logistics system. The aim is to define the structure of a system which can optimize the location and the number of two different kinds of logistics facilities as well as the related routes on each echelon. A mathematic model considering the problem characteristics has been set up. Based on probability selection principle, this paper first puts forward a metaheuristic algorithm with comprehensive consideration of time and space accessibility to solve it. Then, random instances of different sizes are generated to verify the effectiveness of our method.

## 1. Introduction

Many cities find it challenging to set up a high-efficiency city logistics system to increase the freight efficiency of city freight vehicles and decrease the impact of city distribution on city living conditions [1]. In China, more and more cities have been making great effort to reduce traffic in the city center, by creating peripheral logistic platforms or transfer stations, from which only smaller vehicles are allowed to go downtown. Therefore, the bigger-capacity trucks are usually prohibited from entering the city center, and the smaller trucks provide most of the distribution activities within the city. Multiechelon, especially two-echelon logistics system, has been becoming the most common version in practice [2]. The service providers in this system usually need to deliver on or before the time requested by the customer. In consequence, how to select locations for the node facilities of the system, how to dispatch vehicles, and how to plan appropriate vehicle routes have become the key problems in the city logistics system optimization [3].

Transportation route optimization is a long-term research hotspot in the logistics system optimization field. Early in 1959, Dantzig [4] put forward Vehicle Routing Problem (VRP). Afterwards, various problems extended from

VRP attracted extensive attention, such as time constraints, abnormal shape, open type, and backhauls. Two-Echelon Vehicle Routing Problem (2E-VRP) is an extension, aiming at the transportation routing problem of two-echelon logistics system. It was raised by Jacobsen early in 1980 [5]. Since 2E-VRP is an NP problem, which is more complicated than classical VRP, its solution algorithm has got wide attention. It includes deterministic algorithm and heuristic algorithm. In 2008, Crainic et al. [6] and Taniguchi et al. [7] put forward a heuristic algorithm based on cluster, by which they applied two single echelon vehicle routing problems to replace 2E-VRP problem and then got iterative solution of the two subproblems. In 2009, Perboli et al. [8] put forward some effective inequality constraints and applied branch and bound methods to solve 2E-VRP problem; however, such deterministic algorithm is only applicable to small-scale problem. In 2011, Perboli et al. [9] have additionally built two heuristic algorithms to solve 2E-VRP problem. In brief, seeking high-quality and high-efficiency algorithm has become the striving direction in this field.

The above problems only refer to tactical planning decision-making. When a problem extends to the decision-making on both strategic and tactical planning, 2E-LRP problem is put forward, but few scholars have carried out

research on this issue. Alumur and Kara [10] once built a multiple-target MIP model for the two-echelon LRP problem of a hazardous substance and got a solution through CPLEX. Sterle [11] and Boccia [12] applied tabu search algorithm to solve 2E-LRP problem. Nguyen et al. [13, 14] applied four present constructive heuristics and a hybrid metaheuristic to solve 2E-LRP problem. Considering a 2E-LRP problem with capacity limitation, Contardo et al. [15] built two-subscript vehicle follow model, applied branch and cut algorithm for solution, and got lower bound of the model. They put forward self-adaption large-scale neighborhood search algorithm and got upper bound of the model. Zhao [16] built a heterogeneous fleet two-echelon capacitated location-routing model for joint delivery arising in city logistics.

The existing algorithms aiming to solve the 2E-LRP problem mentioned above mainly focused on tabu search algorithm and large-scale neighborhood search algorithm. There are few heuristic algorithms taking the problem structure characteristics into account. Considering two-echelon location routing problem with time constraints in city logistics system and its characteristics, this paper has set up relevant data structure and mathematic model and put forward a metaheuristic algorithm with comprehensive consideration of time and space accessibility and application of probability selection principle. To verify effectiveness of the algorithm, this paper has built relevant calculation instances for solution. The remainder of this paper is organized as follows: the description of the problem and a mathematical model are presented in Section 2; the proposed algorithm is introduced in Section 3. Experimentation and concluding remarks are discussed in Sections 4 and 5, respectively.

## 2. Problem Description

The problem can be described as follows. The two-echelon city logistics network normally consists of a series of known distribution centers, potential transfer stations, and known customers. Due to practical problems, such as city transportation environment and transportation efficiency, goods are distributed first from distribution centers, via transfer stations, and finally sent to customers within given time. In the system, distribution centers, transfer stations, and relevant routes constitute first-echelon city logistics network; transfer stations, customers, and relevant routes constitute second-echelon city logistics network. Under the conditions such as known customers' geographical position and quantity demand, distribution centers and potential transfer stations' geographical position, and capacity limitation and vehicles' capacity limitation in the logistics network at all echelons, we try to solve problems, including facilities locating, rationing, and vehicle routing in the two-echelon logistics network, to minimize the total cost.

An illustrative example of the two-echelon logistics system is given in Figure 1. The overall system is expressed as graph  $G = (N, E)$ , in which node set  $N = P \cup S \cup C$  means set of distribution centers, transfer stations, and customers and  $E$  means set of edges. Let  $C$  be a set of customer demands. The demand of each customer  $i$  ( $1 \leq i \leq k$ ) is expressed as two-tuple combination  $c_i = (w_i, d_i)$ , in which  $w_i \in R+$  means this

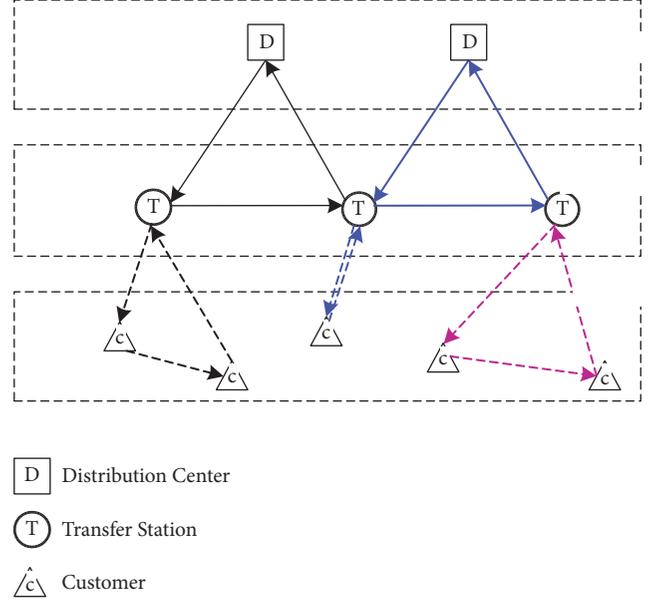


FIGURE 1: Example of the two-echelon logistics system.

customer's demands quantity;  $d_i \in R+$  means this customer's required date for receiving the goods before  $d_i$ ; that is, the time window of customer demand is  $[0, d_i]$ .

Distribution center set  $P$  consists of  $m$  distribution centers; any distribution center  $p_i$  ( $1 \leq i \leq m$ ) is expressed as two-tuple combination  $p_i = (pc_i, pl_i)$ , in which  $pc_i \in R+$  means fixed cost of the distribution center; once the distribution center is used, the fixed cost occurs;  $pl_i \in R+$  means maximum load of the distribution center; the distribution center cannot allow overload; that is, the loading of distribution center cannot exceed  $pl_i$ .

Transfer station set  $S$  consists of  $n$  transfer stations, any transfer station  $s_i$  ( $1 \leq i \leq n$ ) is expressed as two-tuple combination  $s_i = (sc_i, sl_i)$ , in which  $sc_i \in R+$  means fixed cost of transfer station; once the transfer station is used, the fixed cost occurs;  $sl_i \in R+$  means maximum load of the transfer station; that is, the transferred goods quantity of transfer station cannot exceed  $sl_i$ .

The transportation between distribution centers and transfer stations is the first-echelon transportation, and the limited capacity of each first-echelon vehicle is represented as  $L_1$ . Correspondingly, the transportation between transfer stations and customers is the second-echelon transportation, and the limited capacity of each second-echelon vehicle is expressed as  $L_2$ ; obviously, considering the practical conditions, there is  $L_1 > L_2$ .

The edge between any two nodes  $N_1, N_2 \in P, S \cup C$  is expressed as  $E_{N_1}^{N_2} = (c_{N_1}^{N_2}, t_{N_1}^{N_2})$ , in which  $c_{N_1}^{N_2} \in R+$  means the transportation cost of single vehicle at the transportation echelon;  $t_{N_1}^{N_2} \in R+$  means the time for finishing the transportation between the two nodes. When  $N_1, N_2 \in P \cup S$ , the first-echelon transportation occurs; when  $N_2 \in C \cap N_1 \in S \cup C$ , the second-echelon transportation occurs. If there is no direct transportation route between two nodes,

it is expressed as  $E_{N_1}^{N_2} = 0$ . Since direct transportation cannot be realized between distribution centers and customers, it is obvious that when  $N_1 \in P \wedge N_2 \in C$ ,  $E_{N_1}^{N_2} = E_{N_2}^{N_1} = 0$  occurs.

Vehicle  $T_i$  can be expressed as  $T_i = \{path_i, \{C_{T_i}\}\}$ , in which  $path_i = \{E_{N_1}^{N_2}, E_{N_2}^{N_3}, \dots, E_{N_k}^{N_{k+1}}\}$  means that vehicle  $T_i$  must pass  $k$  edges of  $E_{N_1}^{N_2}, E_{N_2}^{N_3}, \dots, E_{N_k}^{N_{k+1}}$  and nodes including  $N_1, N_2, \dots, N_{k+1}$  to finish the transportation task;  $\{C_{T_i}\} = \{C_{T_i}^1, C_{T_i}^2, \dots, C_{T_i}^k\}$  means that the goods of vehicle  $T_i$  contain the transportation demand of  $k$  customers. The time of vehicle  $T_i$  passing node  $N_j$  is expressed as  $t_{T_i}^{N_j}$ , which represents the total time of vehicle  $T_i$  that starts from  $N_1$ , passes  $N_2, N_3, \dots, N_{j-1}$ , and reaches  $N_j$ ; the total time of vehicle  $T_i$  is expressed as  $t_{T_i}$ ; obviously,  $t_{T_i} = t_{T_i}^{N_{k+1}}$ . The total goods weight of vehicle  $T_i$  is expressed as  $W_{T_i} = \sum_{c_j \in \{C_{T_i}\}} w_j$ .

To meet any customer's demand  $c_i$ , the goods must be transported by two different vehicles, which are responsible for the first-echelon and second-echelon transportation tasks, respectively. Therefore, the path of demand  $c_i$  can be expressed as  $path_{c_i} = (p_{c_i}, s_{c_i}, T_i^1, T_i^2)$ , which means that the goods path of  $c_i$  starts from distribution center  $p_{c_i}$ , transported by vehicle  $T_i^1$  to transfer station  $s_{c_i}$ , after being transferred by vehicle  $T_i^2$  to the customer. In the process,  $c_i$  must pass all edges of  $T_i^1$  before arriving at  $s_{c_i}$  and must pass all edges of  $T_i^2$  before arriving at the customer. In this paper, we set that the customer demands cannot be split.

Since customer requires finishing delivering task within a certain time constraint, it is necessary to consider the time consumed in the whole transportation process. The total consumed time includes three parts: (1) time for the first-echelon transportation, expressed as  $t_{T_i^1}^{s_{c_i}}$ ; (2) time for the second-echelon transportation, expressed as  $t_{T_i^2}^{c_i}$ ; and (3) waiting time at transfer station  $s_{c_i}$ . The waiting time of vehicle means that, in the process of the first-echelon transportation, the goods to be transported by vehicle  $T_i^2$  may be from different first-echelon transporting vehicles, and the arrivals of these vehicles are not at the same time. Although the goods of  $c_i$  arrive at transfer station  $s_{c_i}$ , it is required to wait for arrival of other goods and then to realize the dispatch; this period constitutes the waiting time. The length of waiting time depends on the time taken for the arrival of the goods transported by the  $c_i$ -used same vehicle to the transfer station during the second-echelon transportation; the waiting time is expressed as  $wt_{c_i} = \max\{t_{T_j^2}^{s_{c_i}} \mid T_j^2 = T_i^2\} - t_{T_i^1}^{s_{c_i}}$ . It needs to be noticed that all goods demands are known, and the goods of distribution center can be prepared in advance. Therefore, no waiting time is consumed in the distribution center.

The plan of the given system is a set of all customer demands and  $c_i$ -related vehicles and paths, which is expressed as follows.

Set  $UP = \{up_1, up_2, \dots, up_m\}$  is expressed as the use condition of distribution center, in which  $up_i$  ( $1 \leq i \leq m$ ) is a 0-1-variable.  $up_i = 1$  means that the distribution center is used; otherwise, the distribution center is not used.

Set  $US = \{us_1, us_2, \dots, us_n\}$  is expressed as the use condition of transfer station, in which  $us_i$  ( $1 \leq i \leq n$ ) is a 0-1-variable.  $us_i = 1$  means that the transfer station is used; otherwise, the transfer station is not used.

Set  $R = \{r_{N_1}^{N_2}\}$  means the set of all paths in the result, in which  $r_{N_1}^{N_2} \in N+$  means the number of vehicles from node  $N_1$  to node  $N_2$ .

Finally, the cost of the whole plan can be expressed as the sum of fixed cost of distribution centers, fixed cost of transfer centers, and path-to-path transportation cost; that is,

$$Z = \sum_{p_i \in P} up_i * pc_i + \sum_{s_i \in S} us_i * sc_i + \sum_{N_1, N_2 \in \{P, S, C\}} r_{N_1}^{N_2} * c_{N_1}^{N_2} \quad (1)$$

A correct plan must satisfy the following constraints:

$$t_{T_i^1}^{s_{c_i}} + t_{T_i^2}^{c_i} + wt_{c_i} \leq d_i \quad (2)$$

$$W_{T_i} \leq L \quad (3)$$

$$\sum_{T_i \in \text{level } 1} W_{T_i} = \sum_{T_i \in \text{level } 2} W_{T_i} \quad (4)$$

$$\sum_{p_j \in path_{T_i}} W_{T_i} \leq pl_j \quad (5)$$

$$\sum_{s_j \in path_{T_i}} W_{T_i} \leq sl_j \quad (6)$$

Constraints (2) imply that all customer demands must be completed within the specified time constraints. Constraints (3) ensure that any vehicle cannot be overloaded, in which  $L$  depends on the echelon on which the vehicle lies. As for the first-echelon vehicle,  $L = L_1$ ; otherwise,  $L = L_2$ . Constraints (4) indicate that the total amount brought to all transfer stations by the bigger-capacity trucks must be equal to the total demand of customers assigned to these transfer stations. Constraints (5) and Constraints (6) state that any goods through distribution centers and transfer stations cannot exceed the load limit.

### 3. Algorithm

**3.1. Algorithm Framework.** Since the above problems are NP-hard problems, this paper has designed a Heuristic Algorithm Based on Probability (HABP) for solution. The main ideas of the algorithm are the following:

(a) In line 1, HABP seeks initial solution by function InitialSolution(G), which takes the Graph G as input. InitialSolution(G) computes a solution by FirstFit(FF) mechanism and return.

(b) Weight number distribution: in line 2, according to the node-to-node comprehensive space-time distance consisting of single-vehicle transportation cost and transportation time, the function STDistance(node) makes initial processing of graph G in the system to distribute initial value for each node.

(c) Iterative solution: in lines 3 to 5, based on the current solution and initial value, the HABP chooses customer

```

Input: two-echelon routing graph G
Output: New solution V
1.  $V = \text{InitialSolution}(G)$ ;
2. for every node in G :  $\text{value}(\text{node}) = \text{STDistance}(\text{node})$ ;
3. for 1 to  $\text{maxIterationNum}$  do
4.    $V = \text{RandomRePath}()$ ;
5. endfor;
6. return V;

```

ALGORITHM 1: HABP algorithm framework.

nodes for optimization and replaces the current solution when finally better solution is found, where the function  $\text{RandomRePath}()$  is the core function of the algorithm, which seeks better solution in the iteration.  $\text{RandomRePath}()$  will be explained in detail in Section 3.3.

(d) Algorithm termination: when the number of iterations reaches the designated value, stop the search.

The pseudocodes of HABP algorithm framework are shown in Algorithm 1.

**3.2. Composition of Initial Solution.** A given two-echelon routing graph  $G$  with time constraints consists of  $m$  distribution centers,  $n$  transfer stations, and  $k$  customer demands. At the beginning of heuristic algorithm, we search for an initial solution one by one at first and then take it as the starting point for optimizing. For the algorithm in this paper, we adopt FirstFit (FF) algorithm to generate initial solution.

The main ideas of FF algorithm are as follows: for the demand of each customer, we successively search for all potential routes. The first route satisfying the requirements is selected as the route for the demand. Here, we put aside the concentration of goods, assuming that all goods directly start off from the distribution center, arrive at the transfer station, and finally get to the customer. The optimizing is done in the later process of algorithm. In the process of searching for initial route, the two conditions should be met:

(a) As for route  $p_{c_i} \rightarrow s_{c_i} \rightarrow c_i$  satisfying any customer demand  $c_i$ , this route is interconnected; that is,  $E_{p_{c_i}}^{s_{c_i}} \neq 0, E_{s_{c_i}}^{c_i} \neq 0$ .

(b) This route transportation can meet the time requirement of customer demand; that is,  $t_{p_{c_i}}^{s_{c_i}} + t_{s_{c_i}}^{c_i} \leq d_i$ .

The algorithm of initial solution is shown in Algorithm 2.

**3.3. Optimizing Strategy.** For a given two-echelon routing graph  $G$  and initial solution, it is required to utilize optimizing strategy to seek better feasible solution and replace current solution. In the optimizing process, heuristic algorithm usually adopts random method to generate new solution and compare it with current solution. In this process, random strategy directly influences algorithm convergence rate. In HABP, problem solution means a vehicle operating plan satisfying all customer demands in  $G$  in transportation quantity and time. For two different solutions, on the condition that the time demand of all customers is satisfied, the solution with less total cost is the better one. In the process of seeking

```

Input: two-echelon routing graph G
Output: An initial solution V
1. for every customer demand do
2.   for  $i = 1$  to  $m$  do
3.     for  $j = 1$  to  $n$  do
4.       if  $(E_{p_{c_i}}^{s_{c_i}} \neq 0 \wedge E_{s_{c_i}}^{c_i} \neq 0 \wedge t_{p_{c_i}}^{s_{c_i}} + t_{s_{c_i}}^{c_i} \leq d_i)$ 
5.          $r_{p_{c_i}}^{s_{c_i}} + 1; r_{s_{c_i}}^{c_i} + 1;$ 
6.         break;
7. return V;

```

ALGORITHM 2: FirstFit.

better solution, in order to improve convergence rate, it is required to choose a random value with bigger probability to obtain better solution when new solution is generated. Therefore, in the optimizing process, HABP puts forward the concept of selective probability and puts forward an optimizing mode based on probability. This mode contains the following steps:

*Step 1.* Allocate initial value for all nodes.

*Step 2.* Distribute probability for all demands and select optimizing node according to probability, based on the initial value and current solution of nodes.

*Step 3.* Replan for the selected node, generate new solution, and accept new solution with certain probability according to the comparison of new and old solutions.

*Step 4.* Return the optimal solution obtained in the process as a result.

**3.3.1. Initial Value of Node.** In the two-echelon location routing environment, it is not completely equivalent between nodes. For example, for a customer demand, there are two transfer station nodes  $s_1$  and  $s_2$  for choice, and the two routes have the same cost, but  $s_1$  has better connectivity than  $s_2$ ; that is,  $s_1$  is connected with more other nodes and has better cost; it is obvious that  $s_1$  is a better choice for this demand, for it has more possibility to share the transportation resource with other demands and bring optimization of total cost. Here, connectivity is measured from the two aspects of cost and time. About initial value, we consider that the node with less average cost and average time shall have higher probability for route accession.

The cost and time initial values of various nodes are calculated in the formula below.

About distribution center node  $p_i$ , its initial value is the mean value of the route of all transfer stations connected with it; that is,

$$\begin{aligned}
 vc_{p_i} &= \frac{\sum_{E_{p_i}^{s_j} \neq 0} c_{p_i}^{s_j}}{\text{num}_{p_i}}, \\
 vt_{p_i} &= \frac{\sum_{E_{p_i}^{s_j} \neq 0} t_{p_i}^{s_j}}{\text{num}_{p_i}}
 \end{aligned} \tag{7}$$

In the formula,  $num_{p_i}$  means the number of transfer station nodes with route to connect node  $p_i$ ;  $vc_{p_i}$  means the initial cost of distribution center node  $p_i$ , whose value is the mean cost value of the route connecting transfer stations; similarly  $vt_{p_i}$  means the initial time of distribution center node  $p_i$ , whose value is the mean time value of the route connecting transfer stations.

Correspondingly, for distribution center node  $s_i$ , its initial value is the mean value of the route of all distribution centers connected with it; that is,

$$\begin{aligned} vc_{s_i} &= \frac{\sum_{E_{s_i}^{p_j} \neq 0} c_{s_i}^{p_j}}{num_{s_i}}, \\ vt_{s_i} &= \frac{\sum_{E_{s_i}^{p_j} \neq 0} t_{s_i}^{p_j}}{num_{s_i}} \end{aligned} \quad (8)$$

For distribution center node  $c_i$ , its initial values and mean values of all potential direct routes include the values from  $c_i$  to transfer station node and from transfer station node to distribution center, since the values from transfer station node to distribution center can be expressed as  $vc_{s_i}$  and  $vt_{s_i}$ , which are formulated as follows:

$$\begin{aligned} vc_{c_i} &= \frac{\sum_{E_{c_i}^{s_j} \neq 0} c_{c_i}^{s_j} + vc_{s_j}}{num_{c_i}}, \\ vt_{c_i} &= \frac{\sum_{E_{c_i}^{s_j} \neq 0} t_{c_i}^{s_j} + vt_{s_j}}{num_{c_i}} \end{aligned} \quad (9)$$

It should be noted that, for transfer station node and customer node, the initial value does not take the path between the same nodes into account. This is because the path between the same nodes does not represent the actual capacity of the node but only reflects the convenience in combining goods.

**3.3.2. Selection of Optimizing Nodes.** For a given graph  $G$  and a current solution  $V$ ,  $V$  consists of distribution center utility state set  $UP$ , transfer center utility state set  $US$ , and route set  $R$ . The optimizing process means seeking a new solution with cost and time advantages by adjusting the content of current solution. For the algorithm in this paper, the optimizing process consists of two parts: determination of optimizing node and construction of new solution, in which selection of optimizing node is to determine the adjustment of current solution. In this algorithm, we assign probability to each customer demand, take random selection of customer node, and replan its route for optimization, based on node initial value and current solution.

At first, we should define the cost of all demands in the current solution. For any demand, finishing the transportation may include four kinds of routes: (1)  $P \rightarrow S$ ; (2)  $S \rightarrow S$ ; (3)  $S \rightarrow C$ ; and (4)  $C \rightarrow C$ . Among the four routes, route 1 and route 3 are the unique paths that the goods must pass, while route 2 and route 4 are not unique. For any of the routes, the vehicle may carry goods from multiple customers.

Therefore, the transportation costs are shared by multiple customers. Those costs are accounted on one-route vehicle  $T$ ; proportion of demand  $c_i$  is formulated as  $\lambda_i$ :

$$\lambda_i = \frac{w_i}{W_T} \quad (10)$$

In the above formula,  $W_T$  means load of vehicle  $T$  on this route. For demand  $c_i$  of  $path_{c_i}$ , its  $cost_i$  can be formulated as follows:

$$cost_i = \sum_{E_{N_1}^{N_2} \in path_{c_i}} \lambda_i * c_{N_1}^{N_2} \quad (11)$$

Considering the time cost of  $c_i$ , in accordance with the definition, the transfer of all vehicles only occurs in the transfer from the first-echelon transportation to the second-echelon transportation; therefore,  $c_i$  transportation shall and shall only be realized through the two vehicles  $T_i^1$  and  $T_i^1$ , with the total time as follows:

$$t_i = t_{T_i^1}^{s_{c_i}} + t_{T_i^2}^{c_i} + wt_{c_i} \quad (12)$$

According to the above parameters, we can assign the probabilities that are selected for optimization to different demands. We should make an analysis on the aspects of cost and time. As for the cost, a demand that requires high cost in the current solution, it must get distribution of higher probability, for such demand may have higher possibility for optimization. As a reference, the ratio of initial value to current cost  $cost_i/vc_{c_i}$  can be considered as one of basic parameters for probability assignment. As for the time, any demand must meet the requirement of deadline; any delivery within the deadline shall be consistent in effect. However, generally, it is hoped to realize the delivery ahead of the deadline, under the condition that the cost is not raised. In fact, it can also improve the customer satisfaction. In this paper, we define the time impact factor of  $c_i$  as  $\chi_i$ , which is formulated as follows:

$$\chi_i = \begin{cases} 1 & t_i \leq vt_{c_i} \\ \frac{t_i}{vt_{c_i}} & t_i > vt_{c_i} \end{cases} \quad (13)$$

In formula (13), when the time consumption of demand  $c_i$  is higher than the mean value in the current solution, a time impact factor ( $>1$ ) is assigned to it, so that it has higher probability to be selected for optimization.

To sum up, in the HABP algorithm, the probability  $p_i$  that a demand  $c_i$  is selected for optimization is formulated as follows:

$$p_i = \frac{cost_i * \chi_i}{\sum_{c_j \in \{C\}} cost_j * \chi_j} \quad (14)$$

**3.3.3. Construction and Acceptance Strategy of New Solution.** After all demands are assigned with probabilities, we make a random selection of a demand for optimization on the probability base. The optimizing method is to cancel existing

route and reselect a new route for this demand. In the selection process, the following factors should be considered:

(a) Access point: for the route of a demand  $c_i$ , in the final section of the route, two possibilities may occur: starting from the transfer station node or starting from the customer node. It is clear that, in view of cost, “vehicle pooling” is useful for control of comprehensive cost. Therefore, during the planning of new route, “vehicle pooling” should be realized as much as possible under the premise that time and load constraints are not exceeded.

(b) Treatment of original route: for the generation of new route, it is required to cancel the original route. Considering the following conditions, the vehicle transporting for a certain demand  $c_i$  on the second-echelon route has a route in the current solution, expressed as  $\{s_0, c_1, c_2\}$ . When  $c_1$  is selected as the optimization node, route  $E_{s_0}^{c_1}$  becomes unnecessary; at this time, if  $E_{s_0}^{c_2} \neq 0$ , we should make a comparison of the cost and time results between original route  $\{s_0, c_1, c_2\}$  and new route  $\{s_0, c_2\}$ ; the better one shall be selected.

(c) Load: during the construction of new route, load is an important factor for consideration. At first, it should be ensured that no overload occurs. Here, “overload” includes overload of all nodes and overload of all vehicles in the whole transportation. For the overload route, probability should not be assigned. With no overload, we hope to select the route loaded with bigger loading capacity, because this means a reduction of unit transportation cost.

(d) Time: the time consumption must be lower than the time constraint of the demand.

(e) Selection of transfer station and distribution center nodes. During the construction of new route, we should select the transfer station and distribution center route which can bring better solution.

Considering the above constraints, the construction of new route can be implemented in the following steps:

(a) Selecting access point: at first, we should select access point for the demand. In this algorithm, we also apply the mode based on probability; that is, the node with shorter distance shall get higher probability to become the access point for node optimization. For the set  $\{N_j \mid E_{N_j}^{c_i} \neq 0\}$  of the given to-be-optimized demand  $c_i$  and all nodes connected with it, the probability that  $N_j$  is selected as the access point is formulated as follows:

$$p_{N_j} = \frac{1/c_{N_j}^{c_i}}{\sum_{E_{N_j}^{c_i} \neq 0} 1/c_N^{c_i}} \quad (15)$$

(b) Constructing route: after the access point is selected, we shall reconstruct the route. During the route construction, we should consider the influence of different access points. When the selected access point is the customer node, in fact, it is unnecessary to reconstruct a new route; it is only required to add an edge based on the route of the access point. Three conditions need to be met: the load is sufficient; the time constraint is satisfied; the access point is the end point of the vehicle path. If and only if the three conditions are satisfied, we add the edge between the access point and demand node on the vehicle route of the access point, which is regarded as

the new route of the demand. When the selected access point is the transfer station node, the route shall be constructed in the following steps.

At first, we assign probability to the transfer station node. Similar to the preceding method, the route construction also applies probability-based method. The probability that transfer station node  $s_j$  is selected is formulated as follows:

$$p = \frac{1/c_{c_i}^{s_j} * (1 + p_{s_j})/2}{\sum_{E_{c_i}^{s_k} \neq 0} 1/c_{c_i}^{s_k} * (1 + p_{s_k})/2} \quad (16)$$

$$= \frac{(1 + p_{s_j}) * \sum_{E_{c_i}^{s_k} \neq 0} c_{c_i}^{s_k}}{c_{c_i}^{s_j} * \sum_{E_{c_i}^{s_k} \neq 0} (1 + p_{s_k})}$$

In Formula (16),  $p_{s_j}$  means the probability that the node  $S_j$  is chosen for optimization.  $(1 + p_{s_j})/2$  is used to standardize the probability and balance the probability of all nodes. It can avoid the local optimization where the top of the formula means the result of standard probability that the node can be chosen. The bottom of the formula means the total result of all nodes, so the formula presents the final probability that the note  $S_j$  can be chosen.

Formula (16) shows that the probability that transfer station node  $s_j$  is inversely proportional to the distance between  $c_i$  and  $s_j$ . In addition, since transfer station's fixed cost occupies a fairly large proportion in the comprehensive cost, we reduce by half the probability that unused transfer station node is selected. In random method, we are able to select transfer station node. Similarly, after selection of transfer station node, we can also search for distribution center node and finally construct a complete route.

**3.3.4. Acceptance of Route.** After a route is completed, we should judge whether the route is acceptable. Firstly, the route must satisfy the load and time constraints; if not, such route should be given up. Secondly, after the route is constructed, if the total cost is better than the current solution, such current solution shall be replaced. However, we should not simply give up the new solution whose total cost fails to be better than the current solution. Otherwise, the algorithm may fall into local optimum. Here we assign accepted probability to the new route. In the algorithm operating process, with the increase of iterations, the current solution is continuously approaching the optimal solution. Thus, in accordance with the relations between current iterations and maximum iterations, we define acceptance probability as follows:

$$\lambda = \frac{I}{I_{max}} * \alpha \quad (17)$$

In the above formula,  $\lambda$  means acceptance probability,  $I$  means current iterations,  $I_{max}$  means maximum iterations of the system, and  $\alpha$  is a constant, which means the datum acceptance probability of the system. On the basis of probability, if the system accepts the route mentioned above, the current solution shall be replaced; if not, this route shall be given up.

```

Input: Two-echelon routing graph  $G$ , Current solution  $V$ 
Output: New solution  $V'$ 
1. Compute the initial value for every node by Formula (15)
2. for 1 to  $maxIterationNum$  do
3.    $c = ChooseCustomer()$ ;
4.    $n = ChooseLinkNode(c)$ ;
5.   if ( $n \in C$ )
6.      $AddPath(c, n)$ ;
7.     if( $acceptPath() == true$ )
8.        $uptate(V)$ ;
9.     Endif;
10.  else
11.     $RePath(c, n)$ ;
12.    if( $acceptPath() == true$ )
13.       $uptate(V)$ ;
14.    Endif;
15.  Endif;
16. Endfor;
14. return  $V$ ;

```

ALGORITHM 3: RandomRePath.

3.3.5. *Improvement Process.* Algorithm 3 shows the pseudocodes in the improvement process.

In Algorithm 3, line 1 computes the initial value of all nodes. The optimization iteration is executed in lines 2-16. In line 3, the function `ChooseCustomer()` chooses a customer to be optimized randomly as described in Section 3.3.2. The function `ChooseLinkNode()` chooses the previous link node based on the connection graph in line 4, as described in Section 3.3.3. In line 5, if the previous link node is a customer node, the algorithm constructs a new path and compares the result with the current result by function `acceptPath()`. If `acceptPath()` returns “true,” the algorithm takes the new result as the current result. In lines 10-14, if the previous link node is a satellite node, the algorithm uses function `RePath()` to construct a new result and compares it to decide whether to accept it as described in Section 3.3.4.

## 4. Computational Experiments

In this section, we present the results of the computational testing. To the best of our knowledge, there is no suitable standard benchmark for the problem considered here. Therefore, the main experiments were carried out on a number of randomly generated test instances, specially designed to reflect the sizes and peculiarities of real-world distribution problems.

4.1. *The Generation of the Instances.* The main data generated by the instances include (a) the load and fixed cost of the distribution center, (b) the load and fixed cost of the satellite, (c) the weight and time constraints of the customer, and (d) the connected graph  $G$ , where each element contains the distance between two nodes and the transportation time.

Other major parameters that need to be set autonomously in the experiment include (a) the number of parking lots  $P$ , (b) the number of satellites  $S$ , (c) the number of customers  $C$ ,

(d) the capacity of the first-floor vehicles which is set as 1500, (e) the capacity of the second layer vehicle which is set as 900, and (f) the number of algorithm iterations which is set as 1000 times.

It should be noted that the scale of the experiment in this paper is set from 0 to 500. Generally speaking, for any optimization algorithm, one must supply some termination conditions specifying when the algorithm halts. In the experiment of this paper, we have tested that, after 1000 iterations of the system, the optimization effect becomes weaker. That is, it can obtain a better solution. In addition, the system sets a uniform number of iterations, which can eliminate noises in different iterations and pay more attention to the optimization effect of the algorithm itself. Therefore 1000 is set as the number of iterations. When the system needs to deal with large-scale problems, it only needs to add the termination condition to the algorithm. The system will automatically terminate when the convergence speed is reduced. All randomly generated values follow the following methods:

$$value = a + b * rand() * s \quad (18)$$

where  $a$  is the lowest value and  $rand()$  is the random value, whose values are real numbers between 0 and 1.  $b$  represents the coefficients of two random values;  $s$  denotes the scaling factor which is used to expand the random range and calculated by  $s = c \pm d * rand()$ , where  $c$  and  $d$  represent the minimum size coefficient and the scale factor parameter, respectively. Therefore, the value of  $s$  is a real number whose value is between  $c - d$  and  $c + d$ . When  $c = 1$  and  $d = 0$ , it means that the value does not need to be scaled. The values of the parameters are shown in Table 1.

4.2. *Experiments and Results.* In this paper, the purpose of proposing the HABP algorithm is to solve problems of insufficient cost optimization and long calculation time. Therefore, this paper mainly compares the change of cost optimization and algorithm running time after using the HABP algorithm. In the simulation experiment, we analyzed the total cost and running time of HABP when using the FirstFit algorithm and the HABP algorithm in instances of different scales. In the algorithm, the following conditions need to be met:

- (a) The capacity constraints of the vehicle
- (b) Load constraints of the distribution center
- (c) The load constraints of the transfer station
- (d) Customer’s time constraints
- (e) Path access constraints

All tests were performed on an Intel i5-3230M 2.6GHz processor. The algorithm is implemented in C++. The results of the algorithms are shown in Table 2.

In the actual environment, the number of distribution centers or transfer stations is normally fixed, while the number of customers usually varies. Therefore, in the simulation experiment, the value of  $C$  is variable, ranging from 50 to

TABLE 1: Parameter values for generating instances.

Value	$a$	$b$	$c$	$d$
$pl_i$	2000	2000	$k/m$	0
$pc_i$	1000	1000	1	0.15
$sl_i$	1000	1000	$k/n$	0
$sc_i$	500	1000	1	0.15
$w_i$	50	80	1	0
$d_i$	150	170	1	0
$c_{N_1}^{N_2}$	30	30	1	0
$t_{N_1}^{N_2}$	20	30	1	0.2

TABLE 2: Results of the algorithms.

Computational instances	Number of distribution centers	Number of transfer stations	Number of customers	Name of instances	Final costs	Run time (s)	Initial solution cost	Optimization (%)
1	3	8	20	P3-S8-C10	3051.93	0.09	4473.99	31.785
2	3	8	50	P3-S8-C50	6078.23	0.12	7866.92	22.737
3	3	8	100	P3-S8-C100	9274.95	0.16	12489.2	25.737
4	3	8	150	P3-S8-C150	12799.8	0.19	17820.7	28.174
5	3	8	200	P3-S8-C200	14959.7	0.22	19332.7	22.580
6	3	8	250	P3-S8-C250	19391.5	0.23	26043.9	25.543
7	3	8	300	P3-S8-C300	23544.5	0.23	31092	26.198
8	3	8	350	P3-S8-C350	25819.1	0.27	32973.6	21.698
9	3	8	400	P3-S8-C400	31814.9	0.26	41883.6	24.04
10	3	8	450	P3-S8-C450	34776.4	0.29	46474.6	25.171
11	3	8	500	P3-S8-C500	35206.1	0.31	44403.3	20.713
12	3	8	550	P3-S8-C550	43403.2	0.33	56309.2	22.92
13	3	8	600	P3-S8-C600	44031.1	0.35	56052	21.446
14	3	8	650	P3-S8-C650	51651.6	0.34	67129.7	23.057
15	3	8	700	P3-S8-C700	56875.8	0.35	72771.3	21.843
16	3	8	750	P3-S8-C750	61195.3	0.36	77920.5	21.464
17	3	8	800	P3-S8-C800	62861	0.39	79422.5	20.852
18	3	8	850	P3-S8-C850	69820.9	0.41	89022.5	21.569
19	3	8	900	P3-S8-C900	68511.9	0.43	85313.8	19.694
20	3	8	950	P3-S8-C950	78655.7	0.44	97806.6	19.581
21	3	8	1000	P3-S8-C1000	83872.3	0.46	104377	19.645

1000 with a span of 50. From the experimental results, the following conclusions can be drawn.

(a) Compared with the FirstFit algorithm, HABP can effectively optimize the cost, and the optimized ratio is over 20%. What is worth mentioning is that the FirstFit (FF) algorithm is a typical optimization algorithm in computer operating system, which is widely used in industry and is thought to be efficient. The simulation that used FF as the comparison object is meaningful. The experiment was based on the statistical result. We generated multiple graphs of given scale, optimized the result, and took the average result as the final result.

(b) The optimization result of HABP is slightly reduced when the size of customer increases, but the decrease is

not obvious. In the experiment, the effect of the number of iterations on the experimental results is more reflected. This is because the setting is 1000 times. Under the premise of the number of iterations, the larger the scale, the lower the coverage of the randomly selected optimization node.

(c) The convergence speed of HABP is fast. When the number of iterations is determined, similar optimization values can be obtained for different scale instances, indicating that the algorithm can obtain a relatively better solution within a small number of iterations.

(d) HABP has extremely fast operation speed. This speed is not affected by scale. Therefore, a significant advantage of HABP is that it can be applied to large-scale calculations,

which cannot be realized by most current researches and algorithms.

## 5. Conclusion

In this paper, a mixed-integer linear programming model is put forward for a two-echelon location routing problem with time constraints in city logistics. To solve this problem, we propose a metaheuristic algorithm with comprehensive consideration of time and space accessibility and application of probability selection principle. The methods are tested using randomly generated instances with up to 1000 customers. Experimental results show that the proposed algorithm is effective in terms of quality of solutions and computation times in most of the solved instances. In future research, we aim to consider more real-life constraints to deal with some other realistic problems, such as uncertain demand of customers and drop shipping measure. We are also interested in improving the algorithm with other metaheuristic techniques and an iterative process.

## Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

## Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This research was supported by the National Natural Science Foundation of China (Grant no. 71502053). It was also supported by the Natural Science Foundation of Hunan Province (Grant no. 2018JJ2012).

## References

- [1] E. Taniguchi, R. G. Thompson, and T. Yamada, "New opportunities and challenges for city logistics," *Transportation Research Procedia*, vol. 12, pp. 5–13, 2016.
- [2] R. Cuda, G. Guastaroba, and M. G. Speranza, "A survey on two-echelon routing problems," *Computers & Operations Research*, vol. 55, pp. 185–199, 2015.
- [3] T. G. Crainic, N. Ricciardi, and G. Storchi, "Advanced freight transportation systems for congested urban areas," *Transportation Research Part C: Emerging Technologies*, vol. 12, no. 2, pp. 119–137, 2004.
- [4] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.
- [5] S. K. Jacobsen and O. B. G. Madsen, "A comparative study of heuristics for a two-level routing-location problem," *European Journal of Operational Research*, vol. 5, no. 6, pp. 378–387, 1980.
- [6] T. G. Crainic, S. Mancini, and G. Perboli, "Clustering-based heuristics for the two-echelon vehicle routing problem," Montreal: Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation, 2008.
- [7] E. Taniguchi, R. G. Thompson, and T. Yamada, "Recent Trends and Innovations in Modelling City Logistics," *Procedia - Social and Behavioral Sciences*, vol. 125, pp. 4–14, 2014.
- [8] G. Perboli, R. Tadei, and E. Fadda, "Valid inequalities for the two echelon capacitated vehicle routing problem," Montreal: Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation, 2009.
- [9] G. Perboli, R. Tadei, and D. Vigo, "The two-echelon capacitated vehicle routing problem: models and math-based heuristics," *Transportation Science*, vol. 45, no. 3, pp. 364–380, 2011.
- [10] S. Alumur and B. Y. Kara, "A new model for the hazardous waste location-routing problem," *Computers & Operations Research*, vol. 34, no. 5, pp. 1406–1423, 2007.
- [11] C. Sterle, "Location-routing models and methods for freight distribution and infomobility in city logistics," Naples. Italy: Universita Degli Studi di Napoli Federico II, 2010.
- [12] M. Boccia, T. G. Crainic, A. Sforza, and C. Sterle, "A metaheuristic for a two echelon location-routing problem," *Lecture Notes in Computer Science*, vol. 6049, pp. 288–301, 2010.
- [13] V.-P. Nguyen, C. Prins, and C. Prodhon, "Solving the two-echelon location routing problem by a GRASP reinforced by a learning process and path relinking," *European Journal of Operational Research*, vol. 216, no. 1, pp. 113–126, 2012.
- [14] V.-P. Nguyen, C. Prins, and C. Prodhon, "A multi-start iterated local search with tabu list and path relinking for the two-echelon location-routing problem," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 1, pp. 56–71, 2012.
- [15] C. Contardo, V. Hemmelmayr, and T. G. Crainic, "Lower and upper bounds for the two-echelon capacitated location-routing problem," *Computers & Operations Research*, vol. 39, no. 12, pp. 3185–3199, 2012.
- [16] Q. W. Contardo, W. Wang, and R. D. Souza, "A heterogeneous fleet two-echelon capacitated location-routing model for joint delivery arising in city logistics," *International Journal of Production Research*, vol. 56, pp. 5062–5080, 2018.



**Hindawi**

Submit your manuscripts at  
[www.hindawi.com](http://www.hindawi.com)

