Final Report For Logistics Distribution System Location-Routing Modeling

516030910487 席奇 516030910483 邱柏崴

1. Motivation

1.1Background

With the development of market economy and logistics technology, physical distribution business developed rapidly.

1.2Problem

Consumers need to know the instant logistics information in order to predict the arrival time and pick-up time of express.

1.3Drawbacks

Logistics information updates provided by the shopping websites are not timely or the estimated arrival time is inaccurate.

1.4What to do

(1)Determine the relationship between logistics distance and approximate time required.

(2) Determine the distribution vehicle scheduling scheme after arriving in the destination city.

(3) Determine the location of distribution points and whether the carrying capacity meets the requirements.

2. **Problem formulation**

2.1The relationship between distance and time

In physical distribution business, the logistics time from departure to destination is mainly determined by distance and means of conveyance and can be revised by the average timeliness data of the express company in the past few months. The distance and total time spent in the history of express delivery can be counted as a correction parameters.

2.2Vehicle scheduling problem

After the express arrives the destination city, the vehicle scheduling problem affects mostly on raising service quality and timelines.

2.3Distribution center location and load quantity

After the express arrives the destination city, the vehicle scheduling problem affects mostly on raising service quality and timelines.

3. **Brief introduction of the algorithm**

Our algorithm is about how to model a logistics distribution system including Location and routing to achieve the optimal transportation efficiency and the minimum cost.

4. **Analysis**

①Logistics distribution center allocation(based on TSP)

+

②Vehicle routing selection(based on VRP)

For the above two classical problems, we already have genetic algorithm ant colony algorithm and some other algorithms to solve. But there are some obvious shortcomings in these algorithms. For example, genetic algorithm cannot guarantee the maximum probability convergence to the global optimal solution; ant colony algorithm and particle swarm algorithm are easy to produce local convergence slow convergence speed. Therefore, we are trying to design an improved ant colony algorithm with high speed convergence.

5. **Mathematical modeling**

5.1Logistics distribution center allocation

I is the set of production enterprises (1, 2, 3,... N);
H is the collection of product categories (1, 2, 3,... Q);
J is the collection of distribution centers (1, 2, 3,... M);
K is the set of users or customers (1, 2, 3,... P);
A_{hij} refers to the unit freight of product h from the ith production enterprise to the jth distribution center.
B_{hjk} is the unit delivery cost of product h delivered from the jth distribution center to the kth user;
F_j represents the average fixed administrative cost of the product during the period of the jth distribution center;
V_{hj} is the unit variable cost incurred by the jth distribution center for the storage of the h product;
$D_{hk}(T_{hk})$ is the loss due to delay time (T) when delivers the hth product to the kth user;
Q_{hk} is the number of product h required by the kth user;
W_j is the cargo storage capacity of the jth distribution center;
S_j is the infrastructure investment cost after the selection of the jth distribution center;
Y_{hi} is the production capacity of the ith enterprise to produce the hth product.
$I_j \sum_{hjk} X_{hijk}$ is the maximum inventory quota of products distributed by each production enterprise to all users through the jth distribution center.
$Z_j = \sum_j X_{hj}$ is the number of the hth products passing through the jth distribution center.

$$\min F(X) = \sum_{hijk}(A_{hij} + B_{hjk}) X_{hijk} + \sum_{hj} V_{hj}(Z_j)^{\theta} + \sum_{j=1}^{m} F_j R_j + \sum_{hk} D_{hk}(T_{hk}) + \sum_{j=1}^{m} S_j$$

$$\sum_{ij} X_{hijk} = Q_{hk}$$

$$\sum_{jk} X_{hijk} \leqslant Y_{hi}$$

$$I_j \sum_{hjk} X_{hijk} \leqslant W_j$$

$$X_{hijk}, R_j \geqslant 0$$

$$i \in I, h \in H, k \in K$$

5.2Vehicle routing selection

(i) ***Parameter setting***

The distribution center is set to have M cars and the load capacity of the kth car is Q_k （k=1，2，L，C）.

The maximum driving distance of a single delivery is D_k, which requires delivery to L demand points.

The demand at each demand point is q_i （i=1，2，L，L），

The time window is $[e_i, u_i]$. Where e_i is the earliest start time allowed for task i. If the vehicle arrives earlier than e_i, it needs to wait at i. The u_i allows the latest start time for task i, and if the vehicle arrives later than the u_i, task i will be delayed.

Let n_k be the demand points distributed by the kth car (n_k=0 means unused kth car), and set R_k to represent the driving path of the kth car, where r_{ki} represents a demand point, and the order of the path of this demand point R_k is i, and r_{k0}=0 means the distribution center. t_{rki} is also set to represent the moment when the kth vehicle reaches point i on the driving path R_k, w_{rki} represents the time required for the kth vehicle to complete task i (such as acceptance inspection, order signing and unloading, etc.). In addition, c_k is used in the objective function to represent the unit transport cost of vehicle k, p_e is the opportunity cost per unit time to reach the demand point i before ei, p_u is the penalty cost per unit time to reach the demand point i after u_i.

Formula (1) ensures that the sum of demands at each demand point on each path does not exceed the weight of the car;

Formula (2) ensures that the length of each distribution path does not exceed the maximum driving distance of a single distribution;

Formula (3) and Formula(4) ensure that the total working time of the vehicle does not exceed the maximum working time;

Formula (6) indicates that the demand points on each path get the distribution service;

Formula (7) is the composition of demand points of each path;

Formula (8) restricts that each demand point can only be delivered by one car;

Formula (9) describe that,
①when the number of customers served by the KTH car is greater than or equal to 1, it means that the car participated in the distribution, then sign(n_k) =1;
②when the number of customers served by the KTH car is less than 1, it means that the car is not used, and sign（ n_k ）=0.

$$\min z=\sum_{k=1}^{M} c_k\left[\sum_{i=1}^{n_k} d_{r_{k(i-1)}r_{ki}}+d_{r_{kn_k}r_{k0}}\mathrm{sign}(n_k)\right]+p_e\sum_{k=1}^{M}\sum_{i=1}^{n_k}\max(e_i-t_{r_{ki}},0)+p_u\sum_{k=1}^{M}\sum_{i=1}^{n_k}(t_{r_{ki}}-u_i,0)$$

$$\sum_{i=1}^{n_k} d_{r_{k(i-1)}r_{ki}}+d_{r_{kn_k}r_{k0}}\mathrm{sign}(n_k)\leq D_k$$

$$\max(t_{r_{ki}}+w_{r_{ki}}+t_{r_{ki}r_{k0}})\leq u_0 \quad i=1,2,L,n_k\ \forall k$$

$$\min(t_{r_{ki}}-t_{r_{k0}r_{ki}})\geq e_0 \quad i=1,2,L,n_k\ \forall k$$

$$0\leq n_k\leq L$$

$$\sum_{k=1}^{c} n_k=L$$

$$R_k=\{r_{ki}|r_{ki}\in\{1,2,L,L\},i=1,2,L,n_k\}$$

$$R_{ki}IR_{kj}=\phi,\ \forall k_i\neq k_j$$

$$\mathrm{sign}(n_k)=\begin{cases}1 & n_k\geq 1\\ 0 & \text{otherwise}\end{cases}$$

6. **Algorithm design**

Step1: Initialization, setting undetermined parameters and maximum evolutionary algebra;

Step2: Randomly select the location of each ant;

Step3:

Figure out where each ant k is going to move, suppose j, and suppose i was the last position. According to the priority principle of short waiting time and narrow time window required, the time window width of the next intersection j and the time when the intersection i reaches the next intersection j are calculated. Then, the transition probability is calculated according to the path length and the amount of information on the path to the next intersection j. The formula is:

$$pP^{k}_{ij}=\begin{cases}\tau^{\alpha}_{ij}(t)\cdot\eta^{\beta}_{ij}(t)/\sum\limits_{s\in \mathrm{allowed}_k}\tau^{\alpha}_{is}(t)\cdot\eta^{\beta}_{is}(t) & j\in \mathrm{tabu}_k\\ 0 & \text{otherwise}\end{cases}$$

$$\eta_{ij}(t)=\frac{1}{d_{ij}}$$

There are m junctions and n ants, and d_{ij} =(i,j=1,2,…,m) represents the distance between the intersection i and j. and the density of pheromones on the connection between the junction i and j. $\tau_{ij}(t)$means at the time t (initial moment, the same concentration of pheromones on each path). Ant k (k = 1, 2, …, n) in the process of movement, according to various path pheromone concentration decided to transfer the direction, $tabu_k$ (k = 1, 2,…, n) has come to an ant k intersection collection, at the beginning of the$tabu_k$, there is only one element in the ant k start crossing, following an evolutionary $tabu_k$, element in the growing, with the passage of time, on every path pheromone before gradually faded, expressed in parameter 1-ρ pheromone volatilization of degree.

After m moments, the ant complete a cycle, the pheromone on the path gradually disappear, with the parameter 1-ρ pheromone volatility degree, and the pheromone concentration on each path is adjusted according to formula below:

$$\tau_{ij}(t+m)=\rho*\tau_{ij}(t)+\Delta\tau_{ij}$$

$$\Delta\tau_{ij}=\sum_{k=1}^{n}\Delta\tau^{k}_{ij} \quad \rho\in[0,1]$$

Step4: Calculate the newly generated pheromone concentration on path ij as follows:

$$\Delta\tau^{k}_{ij}=\frac{Q}{d_{ij}}$$

Q is the pheromone intensity, which is a constant of the number of tracks left by ants, and it affects the convergence degree of the algorithm.

Step5: Calculate the concentration of pheromone diffused from i to il in other paths as follows:

$$\Delta\tau^{k}_{ij}=D^{k}_{il}$$

$$D^{k}_{il}=\begin{cases} r*Q/d_{ij}(1-d_{il}*(d_{il})^{w}\tan\theta/d^{w+1}), \text{if } d_{il}<d^{-w+1}/(d_{ij})^{w}*\tan\theta \\ 0, \quad \text{otherwise} \end{cases}$$

w is an adjustable constant greater than 1, d is the average distance of cities, r is an adjustable constant less than 1, and the parameter θ is an acute angle.

Step6: Calculate the concentration of pheromone diffused from j to jl in other paths as follows:

$$\Delta\tau^{k}_{il}=D^{k}_{jl}$$

$$D^{k}_{jl}=\begin{cases} r*Q/d_{ij}(1-d_{jl}*(d_{jl})^{w}\tan\theta/d^{w+1}), \text{if } d_{jl}<d^{-w+1}/(d_{ij})^{w}*\tan\theta \\ 0, \quad \text{otherwise} \end{cases}$$

Step7: If every ant in this cycle has implemented Step3~Step6, go to Step8, otherwise go to Step3.

Step8: Update the pheromone concentration on each path according to equation below, where m=1.

$$\tau_{ij}(t+m)=\rho*\tau_{ij}(t)+\Delta\tau_{ij}$$

$$\Delta\tau_{ij}=\sum_{k=1}^{n}\Delta\tau^{k}_{ij} \quad \rho\in[0,1]$$

Step9: If each ant has completed a complete path, turn to Step10, otherwise turn to Step3.

Step10: Judge whether the specified evolutionary algebra has been reached or the obtained solution has not been significantly improved in recent generations, if so, go to Step11, otherwise go to Step3.

Step11: Output the optimization results.

7. Code

```
L_best=inf;
T_best=0;
tau0=1/(n*L_nn);
tau=ones(n,n)*tau0;
ant_path=zeros(m,n+1);

antpath(:,1)=randint(m,1,[1,1]);

current_node=ant_path(k,s-1);
visited=ant_path(k,:);
to_visit=setdiff([1:n],visited);
c_temp=length(to_visit);
if c_temp~=0
  p=zeros(1,c_temp);
  for i=1:c_temp
    p(i)=(tau(current_node,to_visit(i)))^alpha*(1/d(current_node,to_visit(i)))^beta;
  end
  sum_p=sum(p);
  q0=rand;
  select=to_visit(c_temp);
```

```
        if(q0<=0.9)
            [y i]=max(p(i));
        select=to_visit(i);
    else p=p/sum_p;
            [y i]=max(p(i));
        select=to_visit(i);
    end
    if c_temp==1
        select=to_visit(c_temp);
    end
    ordinal_of_vehicle=find(ant_path(k,:)==1);
    last_vehicle=ordinal_of_vehicle(length(ordinal_of_vehicle));
        for l=last_vehicle:n+20
            if (antpath(k,l)~=1) & (antpath(k,l)~=0)
                total_load=total_load+load(ant_path(k,l));
            end
        end
    if (total_load+load(select))>capacity_limit
        select=1;
    end
    total_load=0;
        city_to_visit=select;
        ant_path(k, s)=city_to_visit;
    end
    tau(current_node,city_to_visit)=(1-rho)*tau(current_node,city_to_visit)+tau0;
    tau(Tour_min(i),Tour_min(i+1))=(1-rho)*tau(Tour_min(i),Tour_min(i+1))+rho/L_gb;
```

8. Experiment

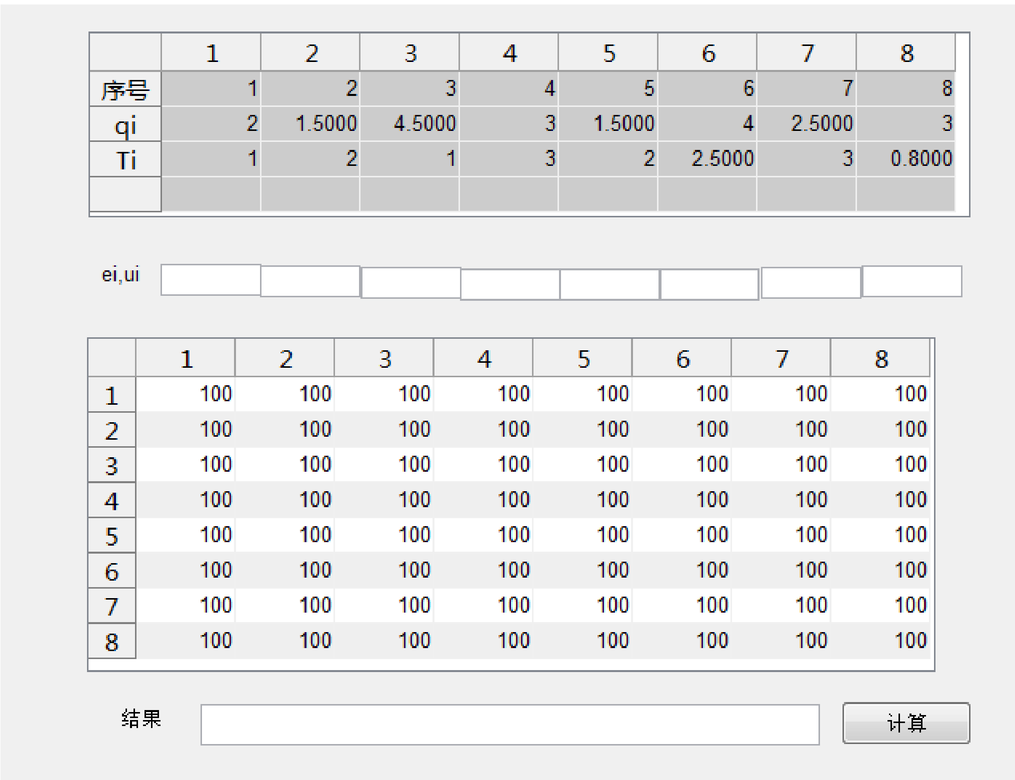
8.1Consumer location information

	1	2	3	4	5	6	7	8
q_i	2	1.5	4.5	3	1.5	4	2.5	3
T_i	1	2	1	3	2	2.5	3	0.8
$[e_i, u_i]$	[1.5 ,4]	[4 , 6]	[1 ,2]	[4 , 7]	[3 , 5.5]	[2 ,5]	[5 ,8]	[1 ,4]

8.2Distance between task points and center and between task points

	0	1	2	3	4	5	6	7	8
0	0	40	60	75	90	200	100	160	80
1	40	0	65	40	100	50	75	110	100
2	60	65	0	75	100	100	75	75	75
3	75	40	75	0	100	50	90	90	150
4	90	100	100	100	0	100	75	75	100
5	200	50	100	50	100	0	70	90	75
6	100	75	75	90	75	70	0	70	100
7	160	110	75	90	75	90	70	0	100
8	80	100	75	150	100	75	100	100	0

8.3Experiment Comparison Results



	1	2	3	4	5	6	7	8
序号	1	2	3	4	5	6	7	8
qi	2	1.5000	4.5000	3	1.5000	4	2.5000	3
Ti	1	2	1	3	2	2.5000	3	0.8000

ei,ui

	1	2	3	4	5	6	7	8
1	100	100	100	100	100	100	100	100
2	100	100	100	100	100	100	100	100
3	100	100	100	100	100	100	100	100
4	100	100	100	100	100	100	100	100
5	100	100	100	100	100	100	100	100
6	100	100	100	100	100	100	100	100
7	100	100	100	100	100	100	100	100
8	100	100	100	100	100	100	100	100

结果

计算

Algorithm	Search success rate	Average driving cost	Average time cost
Genetic Algorithm	24.1%	993.6	17.90
Ant Colony Algorithm	39.7%	951.5	11.28
New Algorithm	46.9%	941.7	9.361

9. Summary

An improved ant colony algorithm based on mutation and dynamic pheromone updating is applied to logistics vehicle distribution routing problem with time window constraints. M ants are evenly placed in n edge distribution points, and the nearest neighbor node selection principle is adopted. On this basis, the time window constraints of each intersection are satisfied. At the same time, dynamic local pheromone updating is carried out and mutation algorithm is used to accelerate local optimization and convergence speed. Compared with other algorithms such as genetic algorithm and basic ant colony algorithm, the result of operation on the same computer has obvious advantages. Therefore, the algorithm described in this paper can be used to optimize the vehicle routing of logistics distribution with time window constraints, and the approximate optimal solution can be obtained quickly under the condition of meeting the demand of time windows at each point, which can provide some reference for solving the vehicle routing optimization problem of logistics distribution with time window constraints in the future.

When an ant searches for a path, if it finds a short subpath (subsolution), it releases a corresponding concentration of pheromone. On the one hand, the pheromone directly affects the ants at two points of the subsolution; on the other hand, it will diffuse outward with the path as the center, affecting the behavior of other ants near the path, so that they will have a greater probability in finding the path. Select the path in the next step. At the same time, under the constraints of time windows, ants open new paths to follow the following rules: starting from the last intersection of the current path, to the first intersection of all the unvisited intersections that have the earliest service time. Only when the service time exceeds the time window of the intersection, can we actively open up a new path, and re-restrict the starting point from the untouched intersection, and make the first intersection of all untouched intersections to serve as the first intersection of the latest path. Through this collaborative approach based on pheromone diffusion under time window constraints, on the one hand, it guides ants to open up the next path in meeting the time window of a new intersection, on the other hand, the interference of other ants in choosing the optimal path when choosing the next intersection will be reduced. Thus, the convergence degree of the algorithm is greatly improved while meeting the time window requirements of each intersection, and the search efficiency and success rate of the algorithm are improved.

We programmed the new algorithm with MATLAB, and we found programs using traditional ant colony algorithm. These programs are operated on the same computer. The results of our algorithm are compared with those of genetic algorithm and traditional ant colony algorithm. It is found that the former is obviously better than the latter.

In this algorithm, a new dynamic information strategy is adopted under the premise of meeting the time window constraints of individual points to ensure that each ant contributes to the search in each search. At the same time, a unique mutation strategy is adopted to search each search result in order to search the results of each search. But the logistics distribution center allocation part based on TSP we haven't found a better algorithm. So in the future we will try to design an appropriate way to solve it.