Based on Ant Colony Blurred Time Vehicle Scheduling Algorithm

Zuo Baohe  
Dept. School of Computer Science & Engineering,  
South China University of Technology  
Guangzhou China.510006.  
e-mail:zuobh@scut.edu.cn

Lin Shaozhu  
Dept. School of Computer Science & Engineering,  
South China University of Technology  
Guangzhou China.510006.  
e-mail:lszlsz44@163.com

Abstract—With the acceleration of global economic integration and the rapid development of e-commerce logistics become more and more important. How to build a logistics distribution mode to meet both Low-cost and client satisfaction become an important and urgent research topics. In this paper we proposed a based on ant colony blurred time vehicle scheduling algorithm (BVSA). Blurred time vehicle scheduling problem is a good extension of traditional vehicle scheduling problem(VSP). In BVSA we studied many problems such as client satisfaction, vehicle waiting time, vehicle traffic time and vehicle distance. We use ant colony algorithm to solve these problems and the experiment show good results.

Keywords—Ant Colony Algorithm; Blurred Time Windows; VSP; VRP; Logistics;

I. INTRODUCTION

With the development of market economy the impact of Modern Logistics that as "the third profit source" on the modern economy become more and more important. Nowadays Logistics has become to the most important areas of competition and the areas of future competition in the market. Logistics will play an important role in the future [1].

Logistics Company how to sent the goods to client with Low-cost, high efficiency, fast and accurate become the important problem of Logistics problem. With the continuous development of modern society and Consumers continue to improve the quality of service, study how to improve logistics efficiency, reduce logistics costs and improve customer service quality has important meaning of the times.

So the problem of Logistics scheduling has attracted many experts’ attention. Foreign experts defined this problem as Vehicle Routing Problem(VRP) or Vehicle Scheduling Problem(VSP). Logistics distribution problem first proposed by Dantzig and Ramser on 1959[2], later N Christofides summarized[3]. In the paper of Christofides[4],Golden and Assad[5] have detailed description.

Base on the classic VRP ,this problem have many different Extension and change patterns such as Capacitated Vehicle Routing Problems(CVRP)[8],Vehicle Routing Problems with Time Windows(VRPTW)[9],Vehicle Routing Problems with Defined Time(VRPDT),Fleet Size and mix Vehicle Routing Problems(FSVRP)[10],Vehicle Routing Problems with Multiple Use of Vehicle(VRPM), Vehicle Routing Problems with Backhauls(VRPB)[12],Vehicle Routing Problem with Stochastic Demand(VRPSD) [15],Dynamic Vehicle Routing Problem (DVRP)[16 17 18 19],Multiple-Depot Vehicle Scheduling Problem with Time Windows(MDVSPTW) [6 7], Vehicle Routing Problem with Split Deliveries(VRPSD) [13], Open Vehicle Routing Problem(OVRP) [14].

VSP problem is a NP-hard problem. At present the solutions of this problem at home and abroad are exact optimization methods, heuristic methods, sub-optimal heuristic algorithm and intelligent methods. We proposed a based on ant colony blurred time vehicle scheduling algorithm.

II. PROBLEM DESCRIPTION

Vehicle Scheduling Problem (VSP) can describe as an Undirected Complete Graph \( G=(C,E) \) [20]. The set of vertices is \( C = \{c_1,c_2,...,c_i\} \), \( c_i \) is the central of distribution; \( \{c_1,c_2,...,c_i\} \) is the set of clients; The set of edges \( E = \{c_i>c_j;i\neq j, c_i,c_j\in C\} \) denote as the route set; \( D = \{\varphi_{ij};i,j=0,1,...,k\} \) denote as the distance between \( i \) and \( j \). We are defined \( l \) as client number; \( \omega_i \) as the weight of client \( i \); \( t_{ij} \) as the time between \( i \) and \( j \); \( m \) as the total number of vehicles; \( q_{ik}(k=1,2,3,...,m) \) as the vehicle load; \( \nu_{ik}(k=1,2,3,...,m) \) as the average speed; \( g_i \) as the weight of client \( i \) needed; \( \{ET_l',LT_l'\} \) as the Tolerable time windows and \( \{ET_l,LT_l\} \) as the expected time windows. The two time windows denote as blurred time windows [21] (see as Figure 1).

![Blurred time windows](image)

Figure 1 Blurred time windows

Based on the theory of blurred setting we defined the satisfaction as. \( \mu_i(t) \) (see formula 1) [21].

The 2nd International Conference on Computer Application and System Modeling (2012)

The 2nd International Conference on Computer Application and System Modeling (2012)

We are also defined the relationship $y_{ia}$ (see formula 2) and $x_{ia}$ (see formula 3).

$y_{ia} \begin{cases} 1 & \text{vehicle } k \text{ distribute client } i \\ 0 & \text{else} \end{cases}$ (2)

$x_{ia} \begin{cases} 1 & \text{vehicle } k \text{ go from } i \text{ to } j \\ 0 & \text{else} \end{cases}$ (3)

Vehicle Scheduling Problem (VSP) must meet the next conditions:

(1) Client satisfaction must bigger than 0 (see formula 4).

$\mu_{ia}(t_i) \geq 0, i = 1, 2, 3, ..., l$ (4)

(2) Each vehicle can't overload (see formula 5).

$\sum_j g_{ja}x_{ia} \leq q_j, k = 1, 2, 3, ..., l$ (5)

(3) Should meet the blurred time windows (see formula 6).

$E_r^{i'} - S_i^{i'} \leq L_r^{i'} - E_r^{i'}$ (6)

(4) Variable mutual constraints (see formula 7,8).

$\sum_{i=1}^{m} x_{ia} = y_{ia}, j = 1, 2, 3, ..., l, k = 1, 2, 3, ..., m$ (7)

$\sum_{i=1}^{m} x_{ia} = y_{ia}, i = 1, 2, 3, ..., l, k = 1, 2, 3, ..., m$ (8)

To simplify the study we defined next supposes:

(1) Goods can be mixed;

(2) Speed unchanged;

(3) Any client can only by one vehicle at one time to distribute.

The target of Vehicle Scheduling Problem (VSP) is to resolve the next 4 target: minimize vehicle time; minimize waiting time; minimize total distance and maximize client satisfaction.

III. ALGORITHM DESIGN AND IMPLEMENTATION

A. Algorithm Analysis

BVSA based on ant colony algorithm give full consideration to the specific requirements of VSP. It improved the selection mechanism, update mechanism and coordination mechanism of ant colony algorithm, added the adaptive shift strategy and pheromone update strategy.

First, set the basic ant colony algorithm:

$\tau_{ij}$ denote as the pheromone of route; $\eta_{ij}$ denote as the visibility of route, define as $\eta_{ij} = 1/t_{ij}$; $\xi_{ij}$ denote as reciprocal of waiting time, define as $\xi_{ij} = 1/w_{ij}$;

So we can compute the Transition probability like (9) [22].

$$p^i_j = \frac{[\tau_{ij}]^{\alpha} \cdot [\xi_{ij}]^{\beta} \cdot [\xi_{ij}]^{\gamma}}{\sum_i [\tau_{ij}]^{\alpha} \cdot [\eta_{ij}]^{\beta} \cdot [\xi_{ij}]^{\gamma}}$$ (9)

$\alpha$ is pheromone Heuristic factor, $\beta$ is expectations Heuristic factor, $\lambda$ and $\theta$ are the weight of $\tau_{ij}$ and $\gamma_{ij}$.

Based on ant colony algorithm, if the target values did not change obviously after N iterations, we update the Volatile factor as formula 10.

$$\rho = \begin{cases} 0.95 \rho (t-1) & \text{if } 0.95 \rho (t-1) > \rho_{min} \\ \rho_{min} & \text{otherwise} \end{cases}$$ (10)

To enhance the globe search of ant and to prevent excessive local converge we update the pheromone of route $(i, j)$ at time $t$ (see formula 11,12).

$$\tau_{ij}(t) = \tau_{ij}(t-1) + \Delta \tau_{ij}(t)$$ (11)

$$\Delta \tau_{ij}(t) = \sum_{k=1}^{n} \Delta \tau_{ij}^{k}$$ (12)

$\Delta \tau_{ij}(t, n)$ Denote as the pheromone of route $(i, j)$ after $n$ iterations.

$\xi_{ia} \in [0, 1]$ is a Coefficient; $\sum_{k=1}^{n} \Delta \tau_{ij}(t, u)$ denote as the sum pheromone of the previous $n - 1$ iterations on the route $(i, j)$.

See formula 11 $\sum_{k=1}^{n} \Delta \tau_{ij}^{k}$ denote as the sum pheromone of all the ant on the route $(i, j)$.

In this paper we defined minimize vehicle time as formula 14; minimize waiting time as formula 15; minimize total distance as formula 13 and maximize client satisfaction as formula 16.
\[
\min F_D = \sum_{i=0}^{n} \sum_{j=0}^{n} d_{ij} x_{ij} F_{D} \tag{13}
\]
\[
\min F_T = \sum_{i=0}^{n} \sum_{j=0}^{n} t_{ij} x_{ij} F_{T} \tag{14}
\]
\[
\min F_w = \sum_{i=0}^{n} W_i (t_i) F_{w} \tag{15}
\]
\[
\max F_s = \sum_{i=0}^{n} \omega_i \mu_i (t_i) F_{s} \tag{16}
\]

So we have the final objective function as formula 17.

\[
\begin{align*}
\text{Min } Z &= k_{s1} \times (F_{D} / F_{D_{max}}) + k_{s2} \times (F_{T} / F_{T_{max}}) + k_{s3} \times (F_{w} / F_{w_{max}}) + k_{s4} \times (1 - F_{s} / (\omega_{max} n \times f_{s_{max}})) \\
&= k_{s1} + k_{s2} + k_{s3} + k_{s4} = 1; F_{D_{max}} \text{ denote as maximum travel time; } n_i \text{ denote as the number of client; } f_s \text{ denote as maximum waiting time; } \omega_{max} \text{ denote as maximum weight of client; } f_{s_{max}} \text{ denote as the maximum client satisfaction; } F_{T_{max}} \text{ denote as maximum travel distance.}
\end{align*}
\]  

B. The Implement of BVSA Algorithm

Step 1: Initialize;
Step 2: When the cycles less than the specified number, do next steps;
Step 3: Initialize pheromone matrix, visibility matrix, distance matrix, the time matrix, customer appointment schedule; put ant on the distribution central and initialize TabuList[i] (the set of Tabu[i]);
Step 4: For each ant \(i, (1 \leq k \leq m)\) do next steps;
Step 5: If there has a client j in TabuList[i] that ant k can meet the constraints and has the biggest \(p_{ij}^k\).

Select client j and put ant to j. Using formulas 13, 14, 15, 16 to compute \(F_D\), \(F_T\), \(F_w\), \(\max F_s\). Then update pheromone.

Else go to Step 5.
Step 6: Using formula 17 to compute \(\min Z\), and update Information concentration of route;

Save the loop times and if loop times bigger than N, using formula 10 to update Volatile factor;
Go to Step 2.
Step 7: Output the result and end;

IV. EXPERIMENT AND COMPARISON

To test the performance of our algorithm that denote as BVSA, we do experiment with the data set ch130 to compare BVSA to Min-Max [24] Algorithm. For our experiment we set iterations number is 500, ant number 100, \(\rho = 0.95, Q = 15000, \alpha = 1, \beta = 0, \lambda = 5, \theta = 0\). TABLE I and TABLE II show the experimental results.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Iterations 1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max-Min</td>
<td>415</td>
<td>426</td>
<td>...</td>
<td>445</td>
<td>418</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>BVSA</td>
<td>398</td>
<td>396</td>
<td>...</td>
<td>376</td>
<td>382</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TABLE II LENGTH OF Max-Min AND BVSA

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>OptimalValue</th>
<th>AVGvalue</th>
<th>variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max-Min</td>
<td>6328.34</td>
<td>6458.01</td>
<td>89.54</td>
</tr>
<tr>
<td>BVSA</td>
<td>6211.76</td>
<td>6348.90</td>
<td>75.59</td>
</tr>
</tbody>
</table>

To testify the feasibility of the BVSA, this paper will simulate the distribution process in Guangzhou city. The distances among the locations in this paper are got by calling Google Map API. TABLE III list the time and goods of 20 client. The NO 0 denote as distribution central.

<table>
<thead>
<tr>
<th>NO</th>
<th>Time (ET, ET, LT, LT')</th>
<th>Weight (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>[40,70,100,125]</td>
<td>3000</td>
</tr>
<tr>
<td>2</td>
<td>[30,60,100,120]</td>
<td>2000</td>
</tr>
<tr>
<td>3</td>
<td>[130,160,190,207]</td>
<td>3000</td>
</tr>
<tr>
<td>5</td>
<td>[60,80,120,135]</td>
<td>5000</td>
</tr>
<tr>
<td>6</td>
<td>[70,90,110,130]</td>
<td>1000</td>
</tr>
<tr>
<td>7</td>
<td>[90,120,180,205]</td>
<td>3000</td>
</tr>
<tr>
<td>8</td>
<td>[80,100,150,170]</td>
<td>2000</td>
</tr>
<tr>
<td>9</td>
<td>[190,220,260,290]</td>
<td>4000</td>
</tr>
<tr>
<td>10</td>
<td>[210,230,280,320]</td>
<td>3000</td>
</tr>
<tr>
<td>11</td>
<td>[300,320,390,410]</td>
<td>4000</td>
</tr>
<tr>
<td>13</td>
<td>[400,420,480,520]</td>
<td>3000</td>
</tr>
<tr>
<td>14</td>
<td>[510,530,580,590]</td>
<td>3000</td>
</tr>
<tr>
<td>15</td>
<td>[520,530,570,590]</td>
<td>2000</td>
</tr>
<tr>
<td>16</td>
<td>[210,240,260,290]</td>
<td>4000</td>
</tr>
<tr>
<td>17</td>
<td>[180,210,260,280]</td>
<td>2000</td>
</tr>
<tr>
<td>18</td>
<td>[200,230,270,290]</td>
<td>4000</td>
</tr>
<tr>
<td>19</td>
<td>[410,440,490,510]</td>
<td>1000</td>
</tr>
<tr>
<td>20</td>
<td>[50,80,130,160]</td>
<td>3000</td>
</tr>
</tbody>
</table>

For our experiment we set \(\alpha = 2, \beta = 3, \theta = 1, \lambda = 2, Q = 250, k_{s1} = 0.7, k_{s2} = 0.10, k_{s3} = 0.1, k_{s4} = 0.1; \rho = 0.95, \omega_{max} = 0.6, \) service time is 20 minutes. When it has the same value 30 times, update \(\rho\). When the iteration number is 150 set \(\rho = 0.8\). Form the experiment we have the result as TABLE IV:

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>OptimalValue</th>
<th>AVGvalue</th>
<th>variance</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max-Min</td>
<td>6328.34</td>
<td>6458.01</td>
<td>89.54</td>
</tr>
<tr>
<td>BVSA</td>
<td>6211.76</td>
<td>6348.90</td>
<td>75.59</td>
</tr>
<tr>
<td>Avgsatisfaction</td>
<td>Length</td>
<td>Totaltime</td>
<td>Waitingtime</td>
</tr>
<tr>
<td>-----------------</td>
<td>--------</td>
<td>-----------</td>
<td>-------------</td>
</tr>
<tr>
<td>0.792</td>
<td>216.79</td>
<td>1255</td>
<td>383</td>
</tr>
</tbody>
</table>

For the experiment result this distribution task require 4 vehicle, the detail distribution as:

Vehicle 1: 0->6->8->7->17->19->14->15->0, load 15000kg goods.
Vehicle 2: 0->2->20->5->3->4->9->0, load 21000kg goods.
Vehicle 3: 0->1->18->12->0, load 7000kg goods.
Vehicle 4: 0->16->10->11->0, load 11000kg goods.

REFERENCES