An Improved Tabu Search Approach to Vehicle Routing Problem

Hongmei Jia*, Yang Li, Bo Dong, Hongying Ya
Hebei United University, No.46 Xihuaxi Street, Tangshan, 063000, China

Abstract

Vehicle Routing Problems have very important applications in the area of distribution management. VRP is both of theoretical and practical interest (due to its real world applications), which explains the amount of attention given to the VRP by researchers in the past years, and since VRP is an NP-Hard problems. In this paper, we designed and realized a new Tabu Search by introducing mutation and mixed local searching tactics for overcoming the weaknesses of the current TS. Here we are concerned with algorithm strategies and parameters and how they affect the performance of the designed Tabu Search. After comparing the Improved Tabu Search with other algorithms, the excellent performance of the Improved Tabu Search is shown. First, the qualities of solutions of the Improved Tabu Search to VRP are very good whether the size of problems is big or small; second, the Improved Tabu Search algorithm is very stable; finally, the convergent speed is fast, with high calculating efficiency.

© 2013 The Authors. Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
Selection and peer-review under responsibility of Chinese Overseas Transportation Association (COTA).

Keywords: vehicle routing problems; Tabu Search; the Improved Tabu Search; solution valuing; termination criteria

1. Vehicle Routing Problems statement

Vehicle Routing Problems, which is introduced by Dantzig and Ramser in 1959, have very important applications in the area of distribution management. As a consequence, they have become some of the most studied problems in the combinatorial optimization literature and large number of papers and books deal with the numerous procedures that have been proposed to solve them.

The Vehicle Routing Problem (VRP) is a complex combinatorial optimization problem, which can be seen as a merge of two well-known problems: the Traveling Salesperson Problem (TSP) and the Bin Packing Problem (BPP). It can be described as follows: given a fleet of vehicles with uniform capacity, a common depot, and several costumer demands, finds the set of routes with overall minimum route cost which service all the demands.

2. Literature review

* Corresponding author. Tel.: +86-315-2594045.
E-mail address: 654578863@qq.com
VRP is NP-Hard problems, and therefore difficult to solve. The fact that VRP is both of theoretical and practical interest (due to its real world applications), explains the amount of attention given to the VRP by researchers in the past years. The Vehicle Routing Problems have been studied by many researchers for the last decades. Bodin, Golden et al. (1983) enumerated more than 700 literatures in their papers. The studies in this field are gone to particulars in the sets of thesis of Christofides(1985), Golden and Assad (1988), and in summarizing papers of Altinkemer and Gavish (1991), Laporte (1992), Salhi (1993). The main researchers in this field are Bodin, Christofides, Golden, Assad, Ball, Laporte, Rinnooy Kan, Lenstra, Desrosiers, Desrochers et al.

There are many researchers whose studied the complexity after Vehicle Routing Problems be put forwarded, that is the foundation of solving algorithms studies. This problem has shown to be NP hard by Lenstra and Rinnooy Kan in the literature: Complexity of vehicle routing and scheduling problems. The complexity of VRP is proved O (n') by Dantzig and Fuikerson (1954); Savelsbergh (1985) and Solomon (1986) put forward VRP with Time Windows is more complex than the standard VRP; Lenstra and Rinnoopy Kan proved mostly all kinds of VRP are NP-hard problems.

As a NP-hard problem, the distribution routing projects of the physical distribution vehicle routing problem will increase exponentially along with the adding of customers.

Here there are techniques used for solving Vehicle Routing Problems. Near all of them are heuristics and meta-heuristics because no exact algorithm can be guaranteed to find optimal tours within reasonable computing time when the number of cities (customers) is large. This is due to the NP-Hardness of the problem.

3. MIP formulations of the VRP in this paper

From a graph theoretical point of view the VRP may be stated as follows: Let \( G = (C, L) \) be a complete graph with node set \( C = \{c_0, c_1, c_2, \ldots, c_n\} \) and arc set \( L = \{(c_i, c_j) : c_i, c_j \in C, i \neq j\} \). In this graph model, \( c_0 \) is the depot and the other nodes are the customers to be served. Each node is associated with a fixed quantity \( q_i \) of goods to be delivered (a quantity \( q_0 \) is associated to the depot \( c_0 \)). To each arc \( (c_i, c_j) \) is associated a value \( d_{ij} \) representing the travel distance between \( c_i \) and \( c_j \). \( n_k \) is the number of customers by \( k \)th vehicle services ( \( n_k = 0 \) means that \( k \)th vehicle is idle). The collection of \( k \)th route is denoted by \( R_k \), the component \( r_{ki} \) represents that the order is \( i \) of customer \( r_{ki} \) in route \( k \) (the depot is not included), \( r_{ki} = 0 \) means the depot. The goal is to find a set of tours of minimum total travel costs (distance). Each tour starts from and terminates at the depot \( c_0 \), each node \( c_k (i = 1, \ldots, n) \) must be visited exactly once, and the quantity of goods to be delivered on a route should never exceed the vehicle capacity \( Q \).

Then VRP can then be formulated as an MIP:

\[
\min \sum_{k=1}^{K} \left[ \sum_{i=1}^{n} d_{r_{(i-1)n_k} + d_{kn_k} \cdot \text{sign}(n_k)} \right]
\]

s.t.

\[
\sum_{i=1}^{n} q_{ki} \leq Q_k
\]

\[
0 \leq n_k \leq L
\]

\[
\sum_{k=1}^{K} n_k = L
\]

\[
R_k = \{r_{ki} | r_{ki} \in \{1, 2, \ldots, L\}, i = 1, 2, \ldots, n_k \}
\]

\[
R_k \cap R_{k'} = \emptyset \forall k \neq k'
\]

\[
\text{sign}(n_k) = \begin{cases} 
1 & n_k \geq 1 \\
0 & \text{others}
\end{cases}
\]
The objective function (1) expresses the total travel distance. Constraints (2) ensure that the total load of each truck not exceed the truck capacity. Constraints (3) ensure the number of customers not exceed the total number of customers. Constraints (4) guarantee that all customers are visited. Constraints (5) are the customers of each route. Constraints (6) guarantee that each customer just only is visited by one truck. Constraints (7) expresses: truck kth joy in the distribution if the number of customers $\geq 1$ serviced by the truck, then $\text{sign}(n_k) = 1$; truck kth doesn’t joy in the distribution if the number of customers serviced by the truck $< 1$, then $\text{sign}(n_k) = 0$.

4. The design of the Improved Tabu Search Algorithm to VRP

As an extension of Local search, Tabu Search relates to some key elements such as the solution indicating method, solution valuing, local searching tactics and termination criterion which decide the performance of the algorithm. Furthermore, we need some elements such as tabus, tabu length, candidate list and so on.

4.1. The solution indicating method

When we solve VRP by heuristics algorithm, the solution indicating method is the key work which decides the performance of the algorithm. In our search, we use the array of customers and dummy depots at same time as the solution indicating method. We transform Vehicle Routing Problem to Traveling Salesman Problem by adding dummy depots. This solution indicating method is put forward based on the above thought. Set 0 as depot, 1, 2, ..., L as customers. There are K vehicles be available in the depot, so there are K route at most, each route start from the depot and terminates of the depot. K-1 depots are added for explain all routes in one solution. Set $L+1, L+2, ..., L+K-1$ as dummy depots. So one array of $1, 2, ..., L+K-1$ is a solution, corresponding to a distribution scheme. For example, there are 7 customers served by 3 truck, we can express the distribution scheme by the stochastic array of $1, 2, ..., 9$ (8, 9 are dummy depot). Solution 129638547 expresses the distribution scheme: the route of truck 1: 0-1-2-9(0), the route of truck 2: 9(0)-6-3-8(0), the route of truck 3: 8(0)-5-4-7-0, there are three routes, three trucks joy in the distribution; Solution 573894216 expresses the distribution scheme: the route of truck 1: 0-5-7-3-8(0), truck 2 doesn’t joy in the distribution, the route of truck 3: 9(0)-4-2-1-6-0, there are two routes, two trucks joy in the distribution. The distribution scheme is fixed that the number of routes less than or equals the number of available trucks.

4.2. Solution valuing

Based on the description of VRP formulations, we valve the quality of solution by: firstly, find the distribution scheme corresponding to the solution, then justify whether the scheme is satisfied with the constraints or not, at the same time, calculate the valve of the objective function, the distribution scheme is better and the solution quality is better which the valve of objective function is better with promising the solution be satisfied with the constraints.

The solutions and the corresponding distribution scheme gained by the array of customers and dummy depots at same time are satisfied with the constraints of each customer be serviced by and only by one truck. To each distribution scheme, we need justify the total load of every route and ensure the total load of any route not exceed the truck capacity. If the solution is not satisfied with the constraints, set the scheme is unfeasible scheme, but still need calculate the objective value. To a solution, set the number of unfeasible routes is M (M = 0, the solution is feasible solution), set the valve of the objective function is Z, the penalty power is $P_w$ (the power is set as the appropriate integer recording to the value space of the objective function), then solution value E is calculated by formula (8) (the quality of solution is better with smaller E). The method of solution valuing embodies the idea of handling constraints by penalty function.

$$E = Z + M \cdot P_w$$ (8)
4.3. Local searching tactics

In our research, a random number which is between 0 and 3 is generated by computer, then one of three operators: “swap”, “invert” and “insert” is adopted at the same possibility according to the generated random number. After the operators, we must ensure the neighbors of the trial solution be still the component of solution space, means the solution is validity. Local searching tactics are related to the solution indicating method. We use the array of customers and dummy depots at same time as the solution indicating method.

4.4. Termination criteria

As a heuristics algorithm, Tabu Search need termination criterion in the iterations in order to get the solution in the reasonable time. In practice, obviously, the search has to be stopped at some point. There are four most commonly used stopping criteria in TS. The first is to stop after a fixed number of iterations (or a fixed amount of CPU time); the second is to stop after some number of iterations without an improvement in the objective function value (the criterion used in most implementation); the third is to stop when the objective reaches a pre-specified threshold value; the forth is the search is usually stopped after completing a sequence of phases, the duration of each phase being determined by one of the above criteria. In this paper, we adopt the second criteria.

4.5. Tabus

Tabus are one of the distinctive elements of TS when compared to LS. Tabus are used to prevent cycling when moving away from local optima through non-improving moves. Tabu (disallowing) moves that reverse the effect of recent moves is done to prevent the search from tracing back its steps to where it came from.

In our research, we adopt tabus to vectors change of solution. For example, the solution $S = 123456789$, the operator is swap. The exchanging points are point 4 and point 7, the neighbor of the trial solution after the operator: $S' = 123756489$. The $S'$ which is the solution after the swap between point 4 and point 5 may be local optimum. For forbidding local optimum, Tabus is the forbidden swap between point 4 and point 5 in the next several steps of iterations. Tabus are: select the new solution in $S$ neighborhood if the new solution is better than $S'$, select the inferior solution in $S$ neighborhood if $S'$ is local optimum.

4.6. Tabu length

Tabu length is the steps number of tabus. Giving tabus $x$ a number $l$ (tabu length) that means tabus $x$ is forbidden in the next $l$ iteration steps, tabu list($x$) = 1. tabu list($x$) = 1-1. After one step of iteration, tabus $x$ is released until tabu list($x$) = 0. The follows is the several methods of choosing tabu length:

- $l$ is a const, such as $l = 10$, $l = \sqrt{n}$ (n is the number of neighbors in the neighborhood). This method easy to realize;
- $l \in [l_{min}, l_{max}]$. $l$ is a variable depending on the value of objective function and neighborhood structure. $l_{min}, l_{max}$ are const and depend on the size of problem, limiting changing space $[a\sqrt{N}, b\sqrt{N}](0 < a < b)$; changing space $[aN, bN](0 < a < b)$ is also set by the number of neighbors in neighborhood $n$.

The selection of tabu length is closed to the characteristics of the practical problems and the experience of algorithm designer. Tabu length is too short leading to the cycling, tabu length is too long leading to the increasing calculating time.

4.7. Probabilistic TS and candidate list
In “regular” TS, one must evaluate the objective for every element of the neighborhood \( N(S) \) of the current solution. This can prove extremely expensive from the computational standpoint. An alternative is to instead consider only a random sample \( N'(S) \) of \( N(S) \), thus reducing significantly the computational burden. Another attractive feature of the alternative is that the added randomness can act as an anti-cycling mechanism; this allows one to use shorter tabu lists than would be necessary if a full exploration of the neighborhood was performed. One the negative side, it must be noted that, in that case, one may miss excellent solutions. Probabilities may also be applied to activating tabu criteria.

Another way to control the number of moves examined is by means of candidate list strategies, which provide more strategic ways of generating a useful subset \( N'(S) \) of \( N(S) \).

4.8. Aspiration criteria

While central to TS, tabus are sometimes too powerful: they may prohibit attractive moves, even when there is no danger of cycling, or they may lead to overall stagnation of the searching process. These are called aspiration criteria. The simplest and most commonly used aspiration criterion (found in almost all TS implementations) consists in allowing a move, even if it is tabu, if it results in a solution with an objective value better than that of the current best-know solution.

4.9. Mutation operator

As the difference from traditional TS, mutation tactics is a very important step in the Improved Tabu Search which decides the improved degree of the performance of the algorithm. After mutation operators, we must ensure the neighbors of the trial solution be still the component of solution space, means the solution is validity.

The mutation will be happened when the generated probability is less than the probability of mutation \( P_m \). If mutation operator is happened, generating the swap times \( J \), swapping the orders of selected components for \( J \) times (the mutation points are generated randomly by computer).

5. Algorithm structure of the Improved Tabu Search to VRP

We design the algorithm structure based on the strategies of Tabu Search.

{Input the conditions of VRP, including the coordinates of depot, the number and coordinates of customers and the demand of each customer;

Input the parameters of the algorithm, including the number \( T \) of steps at termination, the penalty power \( P_w \), and the mutation probability \( P_m \) and so on;

Initialize tabu list \( H \);

Generate the initial solution as the current solution \( S \), iteration step \( t = 0 \);

Calculate the solution value of \( S \) by the solution valuing;

The current best solution \( S_{best} = S \);

The evaluating value of the current best solution \( E_{best} = S \);

while ( \( t < \text{steps at termination} \ T \) ) do

\{ The number of neighbors in the iteration \( n = 0 \);

Give the evaluating value of the best current solution in the iteration \( E_{localbest} \) a big positive number \( E_{localbest} = \exp(100) \);

While(\( n < N \)) do

\{ r1 = \text{random}[0,3];

If (\( r1 < 1 \)) Swap the order of two elements of \( S \), get a neighbor \( S^1 \);

If (\( 1 \leq r1 < 2 \)) Invert the order of two elements of \( S \), get a neighbor \( S^2 \);
If \((r_1 \geq 2)\) Insert the order of two elements of \(S\), get a neighbor \(S^3\);
if \((S^i(1 \leq i \leq 3)\) isn’t the elements in tabu list \(H\) or \(S^i\) is satisfied with aspiration criteria)
{ Calculate the evaluating value of \(S^i\) by the solution valuing;
if (the evaluating value of \(S^i < E_{\text{localbest}}\))
{ \(\text{S_{localbest} = s^i}\);
    \(E_{\text{localbest}} = \text{the solution value of } s^i\);
    \(N=N+1;\)}
\(r_2 = \text{random}[0,1]\);
if ( \(r_2 \leq \text{mutation probability } P_n\)) mutate \(\text{S_{localbest}}\) to \(\text{S_{localbest}'}\), take \(\text{S_{localbest}'}\) as the initial solution of the next iteration, \(S = \text{S_{localbest}'}\);
else \(S = \text{S_{localbest}}\);
\(n=n+1;\)
} 
if ( \(E_{\text{best}} < E_{\text{localbest}}\))
{ \(\text{S_{best} = S_{localbest}}\);
    \(E_{\text{best}} = E_{\text{localbest}};\)}
\(S = \text{S_{localbest}}\);
Release the first element of tabu list \(H\), put \(S_{\text{localbest}}\) in tabu list \(H\) as the last element of tabu list;
\(t = t + 1;\}
Output the distribution scheme and the value of the objective function of \(S_{\text{best}}\);

6. Numerical study and the analysis of the Tabu Search

To compare the performance of the Improved Tabu Search and other heuristics to VRP, we designed 3 instances to analyze and evaluate the performance of the Improved Tabu Search. The instances are as the follows.

<table>
<thead>
<tr>
<th>Case</th>
<th>Customer number</th>
<th>Truck number</th>
<th>Truck capacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>6</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>5</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>100</td>
<td>20</td>
<td>8</td>
</tr>
</tbody>
</table>

6.1. The impact of algorithm strategies and parameters on the performance of Tabu Search.

As a heuristic algorithm, the performance of Tabu Search embodies two aspects: firstly, the effect of the algorithm, namely the quality of the solutions; secondly, the efficiency of the algorithm, namely the capability of getting the optimum solution or a good solution in reasonable calculating time. The performance not only relates to the algorithm strategy but also to the parameters of the algorithm.

Now we will study the impact of some algorithm strategies and the parameters on the performance of Tabu Search to VRP.

The impact of tabu length: as shown in table 2 the average distance of Tabu Search are very similar in five tabu length. When tabu length increase to 20 and 30, the calculating time is longer than the time which tabu length is 15, but the result is worse. So the impact of tabu length is relative light when solving VRP by Tabu Search, but the better solution can’t be achieved if tabu length is too short, oppositively, the quality of solution doesn’t always increase and the calculating time increases if tabu length is too long.
The impact of iteration tactics: in Table 3, in the five iteration tactics, the results of $T=800$, $N=20$ and $T=400$, $N=40$ are better than others. So we can see that the different iteration tactics have the influence on the performance. When the searching time is a constant, the number of neighbors $N$ is too big or too small that goes against the performance of algorithm. To instance 3.2, the better solutions can be achieved when $N$ is between 20–40 and the searching time is 16000.

Table 2. The impact of tabu length on the performance of algorithm

<table>
<thead>
<tr>
<th>Tabu length</th>
<th>Average of total distance (km)</th>
<th>Average number of used trucks</th>
<th>The average steps to reach the best solution first</th>
<th>Average calculating times ($s \times 10^{-3}$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>110.9</td>
<td>3.7</td>
<td>253.9</td>
<td>4.53</td>
</tr>
<tr>
<td>5</td>
<td>108.4</td>
<td>3.9</td>
<td>274.8</td>
<td>4.60</td>
</tr>
<tr>
<td>10</td>
<td>109.4</td>
<td>3.8</td>
<td>298.0</td>
<td>4.60</td>
</tr>
<tr>
<td>15</td>
<td>110.7</td>
<td>3.8</td>
<td>280.9</td>
<td>4.68</td>
</tr>
<tr>
<td>20</td>
<td>109.9</td>
<td>3.7</td>
<td>300.9</td>
<td>5.23</td>
</tr>
<tr>
<td>30</td>
<td>110.1</td>
<td>3.9</td>
<td>253.3</td>
<td>6.01</td>
</tr>
</tbody>
</table>

The impact of local searching tactics: the all results of four iteration tactics of the Improved Tabu Search are better than that of TS, because we adopted the mutation tactics. In the four iteration tactics, the result of Tree-union operator is distinctly better than other tactics, the calculating time of four kinds of operators are very closer. So we can see that the different iteration tactics have some influence on the performance. The reason of the result of three-union operator being better is that the searching space is enlarged and the diversity of solutions is increased because of more operators.

Table 3. The impact of iteration tactics on the performance of algorithm

<table>
<thead>
<tr>
<th>Iteration tactics</th>
<th>Average of total distance (km)</th>
<th>Average number of used trucks</th>
<th>The average steps to reach the best solution first</th>
</tr>
</thead>
<tbody>
<tr>
<td>$T=1600$, $T=10$</td>
<td>115.2</td>
<td>4</td>
<td>8243</td>
</tr>
<tr>
<td>$T=800$, $N=20$</td>
<td>109.5</td>
<td>3.9</td>
<td>9806</td>
</tr>
<tr>
<td>$T=400$, $N=40$</td>
<td>108.4</td>
<td>3.9</td>
<td>10992</td>
</tr>
<tr>
<td>$T=200$, $N=80$</td>
<td>114.8</td>
<td>4</td>
<td>11912</td>
</tr>
<tr>
<td>$T=100$, $N=160$</td>
<td>112.7</td>
<td>3.8</td>
<td>8048</td>
</tr>
</tbody>
</table>

The searching process of Tabu Search: as figure 1 is shown, at the start of iterations (0~190), the improvement of the solution quality is very fast, the total distance is decreased form 212.44km to 104.6km rapidly. In the stage 190~2300 of iteration, the quality of solution is stagnant. The quality of solution which is the global optimum by now reach the highest at step 2300. In the whole process, we can see the convergent speed of tITS is faster than the speed of TS, the quality of solution is better than the quality of TS at the same step.

The impact of the mutation probability: in Table 5, the quality of the solution is increased with the increasing of mutation probability, but when mutation probability is bigger than 0.09, the quality of solutions stops increasing with the increasing of mutation probability, even the quality decreases sometime. So we can see that mutation operator is helpful to increase the quality of solution with increasing the diversity of solution and
enlarging the searching space. But the performance of algorithm will decrease when mutation probability is too big.

![Fig.1 the searching process of the Improved Tabu Search of instance 2](image)

**Table 5. the impact of the times of multi-invert operators on the performance of algorithm**

<table>
<thead>
<tr>
<th>The mutation probability</th>
<th>Average of total distance(km)</th>
<th>Average number of used trucks</th>
<th>The average steps to reach the best solution first</th>
<th>Average calculating times(s*10^-3)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>106.8</td>
<td>3.9</td>
<td>385.8</td>
<td>8.48</td>
</tr>
<tr>
<td>0.05</td>
<td>105.1</td>
<td>4</td>
<td>437.7</td>
<td>8.46</td>
</tr>
<tr>
<td>0.09</td>
<td>104.6</td>
<td>4</td>
<td>422.4</td>
<td>10.28</td>
</tr>
<tr>
<td>0.2</td>
<td>104.6</td>
<td>4</td>
<td>453.3</td>
<td>11.12</td>
</tr>
<tr>
<td>0.6</td>
<td>104.9</td>
<td>4</td>
<td>446.1</td>
<td>11.28</td>
</tr>
<tr>
<td>0.8</td>
<td>104.7</td>
<td>4</td>
<td>453.9</td>
<td>10.58</td>
</tr>
<tr>
<td>1.0</td>
<td>105.0</td>
<td>4</td>
<td>447.6</td>
<td>11.56</td>
</tr>
</tbody>
</table>

The following characteristics of tITS can be summarized by studying the searching process: ① the quality of solution is improving with the steps of iteration. In start stage of iteration, the improvement of the quality of solution is very fast. The speed of improvement is slower and slower with the increase of the iteration steps; ② the performance of the Improved Tabu Search is very better than the performance of TS considering the quality of solution and convergent speed. It is proved that the Improved Tabu Search algorithm to VRP has stronger capability to search the global optimum than Tabu Search.

6.2. The comparison of the Improved Tabu Search and other heuristics to VRP

To compare the performance of the Improved Tabu Search and other heuristics to VRP, the author put the results of the Improved Tabu Search and other heuristics to VRP, so we can analyze and compare the performance of these heuristics algorithms.

As shown in table 6, the quality of result of the Improved Tabu Search to VRP is best, Clarke and Wright Savings is followed, Genetic Algorithm is worst. In the stability of the results, the Improved Tabu Search to VRP is best, Tabu Search is followed, Hill Climbing is worst. In the number of the used trucks, six algorithms is similar. In the calculating times, Clarke and Wright Savings is shortest, Tabu Search is followed, Genetic Algorithm is longest.

Through above numerical study, we can summarize the following characteristics of the Improved Tabu Search to VRP:
• The qualities of solutions of the Improved Tabu Search to VRP are very good whether the size of problems is big or small;
• The Improved Tabu Search algorithm is very stable;
• The convergent speed is fast, the calculating efficiency is high.

### Table 6. the comparison of different algorithms

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Total distance (km)</th>
<th>Standard deviation of solutions</th>
<th>Number of used trucks</th>
<th>Times of searching that the first time to reach the best solution</th>
<th>Calculating times (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clarke and Wright Savings</td>
<td>107.8</td>
<td>-</td>
<td>4</td>
<td>-</td>
<td>0.93*10^{-3}</td>
</tr>
<tr>
<td>Hill Climbing</td>
<td>128.0</td>
<td>8.34</td>
<td>3.9</td>
<td>893.5</td>
<td>0.33*10^{-3}</td>
</tr>
<tr>
<td>Simulated Annealing</td>
<td>109.5</td>
<td>5.99</td>
<td>3.9</td>
<td>12012</td>
<td>1.76</td>
</tr>
<tr>
<td>Genetic Algorithm</td>
<td>140.1</td>
<td>4.30</td>
<td>4</td>
<td>295.3</td>
<td>2.44</td>
</tr>
<tr>
<td>Tabu Search</td>
<td>108.4</td>
<td>3.14</td>
<td>3.9</td>
<td>10992</td>
<td>4.60*10^{-3}</td>
</tr>
<tr>
<td>The Improved Tabu Search</td>
<td>104.6</td>
<td>1.41</td>
<td>4</td>
<td>60720</td>
<td>10.28*10^{-3}</td>
</tr>
</tbody>
</table>

### 7. Conclusions

In this paper, we designed and realized a new Tabu Search by introducing mutation and mixed local searching tactics for overcoming the weaknesses of the current TS, here we concern how algorithm strategies and parameters affect the performance of the designed Tabu Search. And after comparing the Improved Tabu Search with other algorithms, the excellent performance of the Improved Tabu Search is shown. First, the qualities of solutions of Improved Tabu Search to VRP are very good whether the size of problems is big or small; second, the Improved Tabu Search algorithm is very stable; final, the convergent speed is fast, the calculating efficiency is high.

### Acknowledgements

This research was conducted at Hebei United University. We acknowledge the financial support of the Social Sciences Foundation of Hebei province (HB10XGL187): Study on comprehensive transportation assessment and management system of Hebei province, Science and Technology Research and Development Program of Tangshan(111302011b): Research on Urban Comprehensive Traffic Structure Optimization Mechanism.

### References


