Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

# Particle Swarm Optimization for Open Vehicle Routing Problem with Time Dependent Travel Time

**ZhaoYanwei* , Wu Bin****
**Wang Wanliang**, Zhang Jingling***

** MOE Key Laboratory of Mechanical Manufacture and Automation, Zhejiang University of Technology, Hangzhou, 310032, China (e-mail:zyw@zjut.edu.cn)*
*** College of Information Engineering, Zhejiang University of Technology, Hangzhou, 310032, China(e-mail:seasimen@yahoo.com.cn)*

**Abstract:** Open Vehicle Routing Problem with Time Dependent Travel Time (OVRPTD) is different from most variants of vehilce routing problems from the liteture in that the vehicle dosen't return to the depot after serving the last customer and the travle time is time dependent. The travle time is presented by a continuous dynamic network time dependent function. Particle Swarm Optimization with self-adaptive inertia weight is presented. Each particle regulates its inertia weight according to the corresponding position with itself and the best particle in the population. Different updating rules are applied to the excellence particles and the inferior particles. For the excellence particles, compute their information entropy after server iterations, and update their position according to the new position updating function. And for the inferior particles, record them in the bulletin board, then after several iteration, use the new particles displace the inferior according the appearance frequency in the board. In the experiment, the influence of the population, iteration, inertia weight for the optimization result is discussed. By the experiment, give the field of the parameter. Compare the particle swarm optimization with other algorithms by the benchmark. The result shows the algorithm in the paper is the efficiency for the OVRPTD.

## 1. INTRODUCTION

The Vehicle Routing Problem (VRP) is an important problem occurring in many distribution systems. Vehicle routing problem is an NP problem, it has therefore attracted considerable research attention and a number of algorithms have been proposed for its solution. There are two general assumptions in the past research. First, the travel time or cost of the route is assumed to be static and wouldn't change in the process. Second, vehicles start from the depot and finished service, they return to the depot. However, vehicle in the actual process, because traffic flow, accidents, weather changes and other factors, will always be at a constant speed of change. On the other hand, the vehicles do not need to return to the depot in many cases. For example, third-party logistics companies leasing vehicles, the schools bus and so on. These two types of problems currently have a separate study, the first problem is called time dependent vehicle routing problem, the second problem is called open vehicle routing problem. But these two issues combine the study has not been reported, in the paper, the problem is researched.

Time dependent vehicle routing problem gradually become a hot research after entering the 21st century, Joojung (2000),Yang-Byung(2000), Soumia(2003) and AlbertoV. Donati (2006), have researched genetic algorithm, tabu search, ant colony algorithm optimizing the problem. Ali Haghani(2005) and Potvin(2006), research the dynamic customer demand combine the time dependent vehicle routing problem. Due to time-dependent problems too complex, precision algorithms can not be applied, and some algorithms with the operation of exchange and insertion can not be applied directly. There are two problems existed in solution for OVRP. First, the applied methods are singleness. Brandao(2004),Fu.Z(2005) research tabu search algorithm for OVRP, Tarantilis(2004)research the threshold algorithm for OVRP. Many algorithms have not been attempted, such as particle swarm optimization, ant colony optimization and genetic algorithm. Second, the model of OVRP is simple. Some constraints in the practice have not considered in the model.

## 2. TRAVEL TIME-DEPENDENT FUNCTION

In vehicle routing problems and more generally in the transportation field, an important area thatremains very challenging is the conception of efficient models to achieve a good trade-off between the implementation requirements and the ability to reflect the complexity of real-world conditions such as fluctuations in travel times. In the paper, the figure 1 presents the speed of travel time-dependent model. In the model, the travel speed smoothly changes. And according to actual situation in the traffic section, such as peak morning and evening peak situation can be embodied. This can be broadly reflecting the actual speed road conditions change. The model also meets FIFO criteria.

978-1-1234-7890-2/08/$20.00 © 2008 IFAC

10.3182/20080706-5-KR-1001.2843

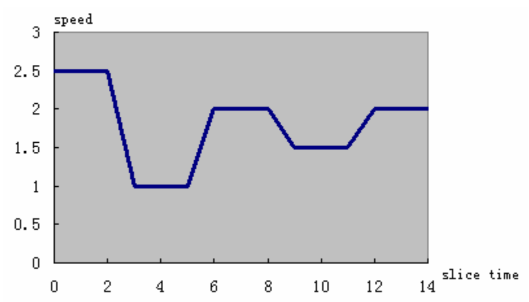



Fig.1. Travel time-dependent function

## 3. PARTICLE SWARM OPTIMIZATION FOR OVRPTD

### *3.1 Particle Swarm Optimization with Adaptive Inertia Weight Adjustment*

The Particle Swarm Optimization (PSO) algorithm is an adaptive algorithm based on a social-psychological metaphor. It is originally proposed by J.Kennedy. A population of individuals adapts by returning stochastically toward previously successful regions in the search space, which is influenced by the successes of their topological neighbours. The research of PSO algorithm shows, the inertia weight $\omega$, and $c_1, c_2$ which are called cognitive and social parameter respectively, have a very significant impact on the performance of optimization. The inertia weight $\omega$ is employed to control the impact of the previous history of velocities on the current one. Accordingly, the parameter $\omega$ regulates the trade-off between the global (wide-ranging) and local (nearby) exploration abilities of the swarm. Eberhart(2005), Zhang Xuanping(2005), Chatterjee (2006) ,and Liu Bo(2005) present a variety of inertia weights improvement. Several such inertia weight adjustment methods, and are targeted at the entire group. And each iteration, all the particles using the same parameters. However, the actual activities of human society, every individual in the operation will be governed by their own experience or around the impact of the choices the next time action, according to their own circumstances and the environment parameters of regulation. Therefore, this paper presents an adaptive regulation of each particle's inertia weight method. According to the PSO algorithm's updating formula, the particle's velocity of the next iteration, are made up of three components: the current speed, the distance between current location and the history optimal position, and the distance between current location and it's the population optimal position. From the research results have shown that the inertia is bigger, the particle's velocity is faster, and a more conducive particle global search, on the contrary, it will help local search. Well, the particles based on the current location and its optimal location and the optimal location of the population; adjust the inertia weight value to expand the global search, or for local search; In other words , the particle close to the population optimal position, or close to the history optimal location of their own.

Definition1: $|P_g - X_i|$ presents the distance between the particle $i$ and the population optimal position $P_g$ . $|P_i - X_i|$ Presents the distance between the particle $i$ and its history optimal position.

Definition2:
$d_i = \dfrac{|P_g - X_i|}{|P_i - X_i|}$ $\quad \forall |P_g - X_i| \neq 0, |P_i - X_i| \neq 0$ presents the particle $i$ 's location offsets. Then each particle's inertia weight can be computed as following:

$$\omega_i = \begin{cases} \omega * d_i & 0 < d_i \leq 1 \\ \omega * \dfrac{d_i^2}{1+d_i^2} & d_i > 1 \end{cases} \tag{1}$$

$\omega$ presents the initial value of weight, as a general set 1.By the PSO algorithm convergence theorem the value of $\omega$ in the range of [0,1]. So, $d_i$ >1, regulate its value into [0, 1]. Constraints $|P_g - X_i| \neq 0, |P_i - X_i| \neq 0$ presents when the particles in the optimal location or populations optimal position for the last movement of incentives, inertia weight remained unchanged.

### *3.2 Procedure of PSO for OVRPTD*

Adaptive weight adjustment PSO algorithm for open vehicles routing problem with time dependent travel time the process as shown in Figure 2.

(1)Encoding Method

PSO encoding method is based on real number encoding method, which is present by Wang Wanliang(2006). In the method, the integer part of each dimension or element in the vector represents the vehicle. Thus, the same integer part represents the customer in the same vehicle. The fractional part represents the sequence of the customer in the vehicle.

(2) Initial Procedure

The initial particle positions are produced randomly. Firstly, customers' permutation is randomly generated and then every customer is assigned to the vehicles. Secondly, the delivery plan is mapped to the particle's position according to the encoding rules. Repeat the procedure until it produces the initial population.

(3)Strategy of Updating

After decoding, the improvement nearest neighbour which is presents by Malandraki is used to improve the solution. In every generation, after computing particles' fitness, sequence the particles according to fitness. Then, record the best and the worst of particles on the bulletin board. Every a few generations, calculated the best particles' Information Entropy, Information Entropy is defined as follows: $F(X_i)$ Presents the particle's fitness, $E(X)$ presents the particle swarm's Information Entropy. If the information entropy of the best particles remain the same or increasing steadily in several times, it shows premature convergence. The best particle will update their position according to the formula(3) presented by F. Bergh.

$$p_i = F(X_i) / \sum_1^m F(X_i)$$
$$E(X) = \frac{-\sum_1^m p_i \ln(p_i)}{\ln m} \tag{2}$$

$$\begin{aligned} v_{gd}(t+1) &= -x_{gd}(t) + p_{gd}(t) + \omega v_{gd}(t) + \rho(t)(1-2Rand()) \\ x_{gd}(t+1) &= p_{gd}(t) + \omega v_{gd}(t) + \rho(t)(1-2Rand()) \end{aligned} \tag{3}$$

For the worst particles, count its frequency in the bulletin board, the highest frequency of particles is removed, and then randomly generate new particles added to the population. Meanwhile, in order to ensure the diversity of species, the cross-operator algorithm is added into the algorithm. The Framework of PSO for OVRPTD is shown in Figure2.

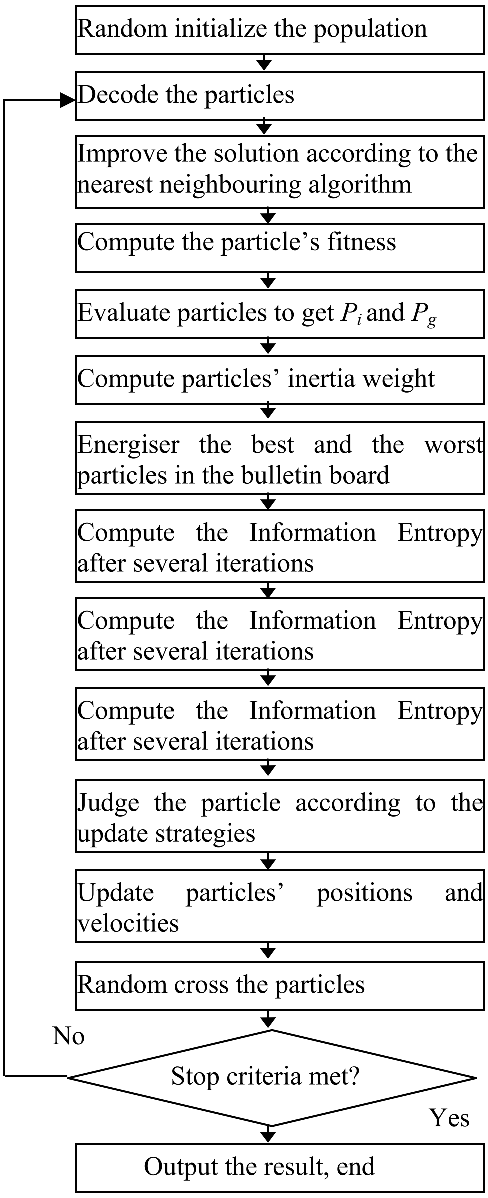



Figure2 Procedure of PSO for OVRPTD

*3.3 Analysis of Algorithm Time Complexity*

The time complexity of the PSO in the paper can be analysis as following: when the population size is $P$ ,the iteration number is $N$, the problem size is $L$. In the decoding phase, the main computing time spent in sort and search, and the time complexity of the two algorithms is $O(L^2)$. The heurist of improved the solution is $O(L^2)$ also. So, the total time complexity is $O(3P*L^2)$. The time complexity of computing fitness is $O(P*L)$. The time complexity of evaluating the best particles is $O(P)$. The time complexity of computing inertia weight is $O(3P*L)$. The time complexity of recording the particle the bulletin board is $O(P^2)$. The time complexity of computing Information Entropy is $O(P)$. The time complexity of updating the particles is $O(3P*L)$. The time complexity of crossing the particle is $O(4P*L)$. So, the whole algorithm's time complexity is:

$$T(P,N,L) \approx N*P*O(L^2)$$

## 4. COMPUTATIONAL EXPERIMENTS

*4.1 Experiments Data*

In the experiment, the revised Solomon benchmard is used. In the benchmark, there are three types data: heap distribution (C), the random distribution (R), semi-heap distribution (RC) . Five instances are selected in every type. In the test cases, the road is divided into three categories, respectively, main roads, normal roads and secondary roads. Discrete time divided into five sections, as shown in Figure3.

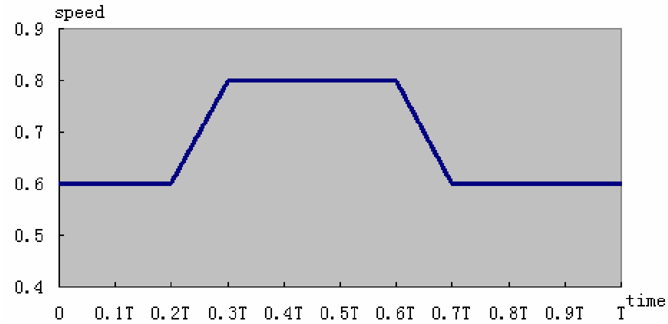



Figure.3. Speed time dependent function

*4.2 The Algorithm is Affected by Populations*

For choosing the algorithm's parameter of population,the C101,R101,RC101 are tested. The number of particle, respectively 5,10 .... As shown in the first column. The iteration is 500, and the algorithm randomly run 50 times. The results of the mean and standard deviation are shown in the following table 1. As can be seen from the table, when the algorithm is smaller populations, the results of optimization are poor, and the algorithm is instability, a lot of ups and downs. This is mainly because a small population of particles, the lack of diversity, easily trapped into local optima. As the population size increases, the algorithm's result is better and the algorithm is more stable. But population size reaches a certain level; the results of the optimization stabilized and are no longer having changed significantly with the increase of population size. As for the three issues, when population sizes over 80, the results of the optimization algorithm have no obvious impact. Figure 4 is the statistics result of three problems.

Table1. Result of the algorithm with different populations

| | C101 | | R101 | | RC101 | |
|---|---|---|---|---|---|---|
| | Mean | S.D. | Mean | S.D. | Mean | S.D. |
| 5 | 378.2 | 76.7 | 636.7 | 71.2 | 483.9 | 65.3 |
| 10 | 272.6 | 52.4 | 583.8 | 58.4 | 437.5 | 57.6 |
| 20 | 192.7 | 18.6 | 532.4 | 35.3 | 385.3 | 30.2 |
| 30 | 168.5 | 10.8 | 482.3 | 26.1 | 336.7 | 19.3 |
| 50 | 164.6 | 9.3 | 477.3 | 18.4 | 332.8 | 16.7 |
| 80 | 157.3 | 8.7 | 474.7 | 16.8 | 327.5 | 14.2 |
| 100 | 156.4 | 9.1 | 475.3 | 15.3 | 328.9 | 15.5 |
| 200 | 157.8 | 8.9 | 473.7 | 14.7 | 326.1 | 14.3 |

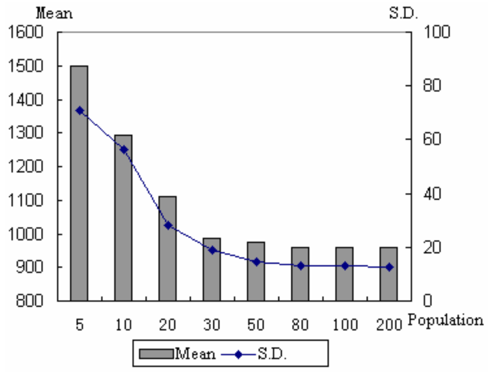



Figure.4. Statistics result of different populations

*4.3 The Algorithm is Affected by the number of iteration*

The population size is 50, randomly run 50 times, the number of iteration is 50,100.... as shown in the first column, the mean result and the excellent rate shown in the following table 2. The criteria of the excellent rate is the error in the optimal results within 5%.In table 2,it shows that with the number of iteration increasing, the algorithm gets better results, and the excellent rates increase high probability. PSO can converge to the optimal results from the theoretical analysis, but it need endless iteration time. In the actual situation , the calculation can not be unlimited, but increased the number of iteration, the result can improve. From PSO principle, as the iteration increasing, the particles can more fully and more clearly understand in the search space, and the exchange of information between the particles are more fully. It improve the quality of the result Figure 5 represents the three problems' statistics result , they also reflecte the same results.

Table 2 Result of the algorithm with different iterations

| | C101 | | R101 | | RC101 | |
|---|---|---|---|---|---|---|
| | Mean | R.G. | Mean | R.G. | Mean | R.G. |
| 50 | 282.6 | 0% | 584.6 | 0% | 439.2 | 0% |
| 100 | 229.3 | 20% | 537.3 | 16% | 375.3 | 24% |
| 300 | 178.8 | 56% | 516.4 | 48% | 356.8 | 60% |
| 500 | 164.7 | 78% | 473.2 | 70% | 336.4 | 80% |
| 800 | 162.4 | 80% | 471.4 | 80% | 331.8 | 80% |
| 1000 | 158.7 | 86% | 472.3 | 80% | 328.5 | 84% |
| 1500 | 156.2 | 92% | 470.3 | 84% | 326.3 | 86% |
| 2000 | 154.6 | 92% | 469.2 | 88% | 321.4 | 86% |

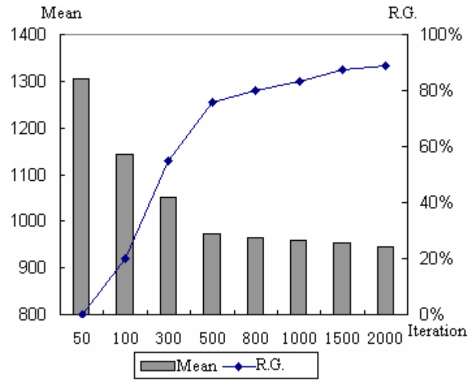



Figure.5. Statistics result of different iterations

*4.4 Inertia Weight Adjustment Comparison*

The C101, R102, RC101 are used to compare the inertia weight adjustment methods. Figure 6, 7, 8 represent each of a variety of algorithms to solve the problems of the mean result and and the optimum result. AFPSO represents the adaptive inertia weight adjustment method, LPSO represents Liu Bo, who proposed method, CPSO represents Chatterjee and others presented the method, and PSO represents Eberhart and Shi, who presented linear adjustment programs. As can be seen from the figures, the method that we present in the paper has strong search capabilities to search for a better solution, and the algorithm is stable. LPSO is the second method, only slighter inferior than AFPSO. And the Chatterjee's nonlinear method has improved slightly than Eberhart and Shi's linear methods, but have a gap compared with above two methods. Because AFPSO and LPSO adjust inertia weight based on the particles' fitness, which the adjustment has the aim, rather than blindly adjustments. And they can use the population information reasonablely, and adjust the inertia weight according to the trend of particle movement. The difference between AFPSO and LPSO is that LPOS adjust inertia weight for the swarm, but AFPSO adjust inertia weight for each particle, taking into account the movement of particles of different trends. However, the latter two methods are changed the inertia wegiht based on the process of iterative algorithm, and not too many other factors to consider. So the change is "unconscious, without purpose", therefore, the impact of the algorithm is relatively small. Figure 9 is the optimization process of C101, Figure 10 is the optimization process of R102.

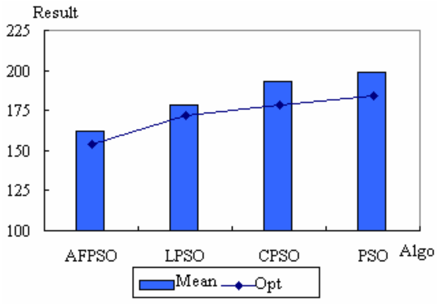



Figure.6. Comparison about C101

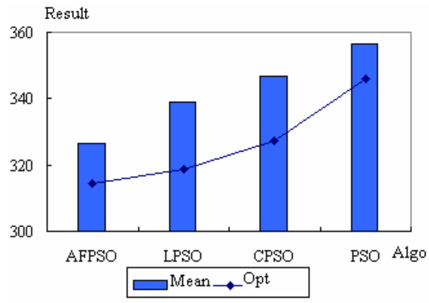



Figure.7. Comparison about R102

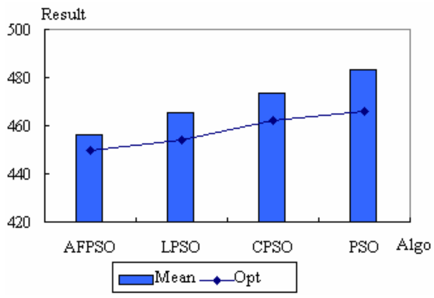



Figure.8. Comparison about RC101

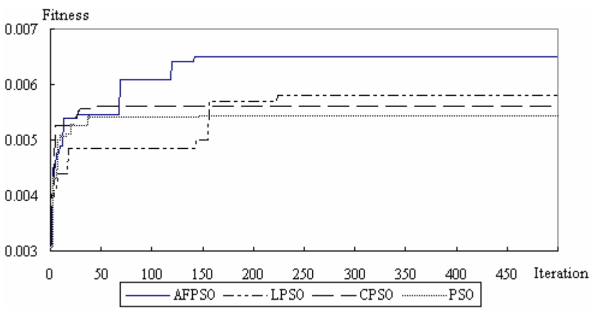
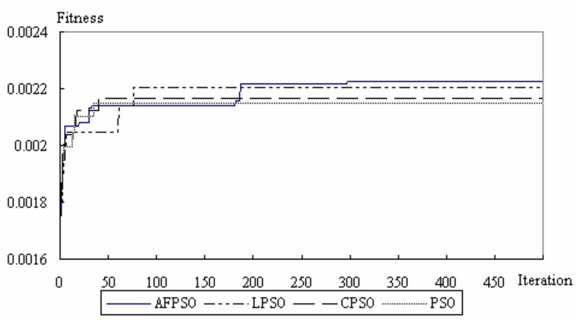



Figure.9 Evolution curve of several algorithms C101




Figrue.10 Evolution curve of several algorithms R102

### 4.5 Update Strategy Comparison

Figure.11 shows the comparison of the using update strategy that is presented in the paper and not using the strategy. As can be seen from figure, the mean results and standard deviation which is used update strategy have larger improvement than not using the update strategy. The proposed strategy includes two parts, one is for the fine particles, mainly to avoid premature convergence while maintaining stability algorithm. The second part for the relatively poor performance of particles through the re-initialization algorithm to ensure diversity. Figure 12, Figure 13 is C102 and RC103 two algorithms' evolutionary curve.

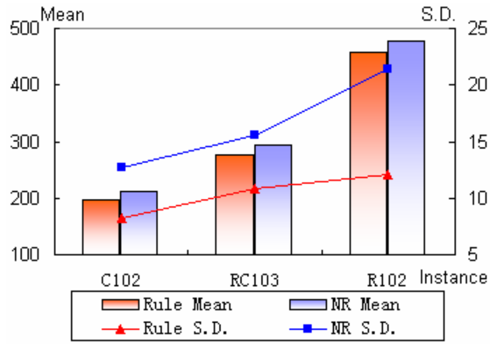



Figure.11 Statistics result of the algorithms

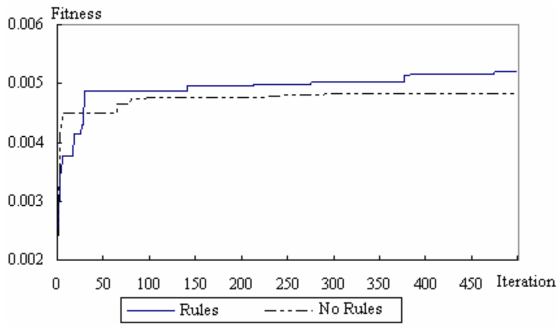
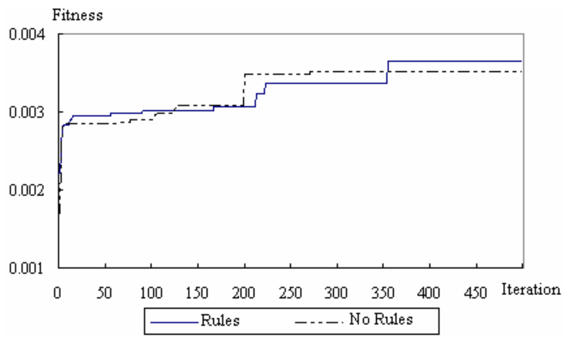



Figure.12 Evolution curve of several algorithms C102




Figrue.13 Evolution curve of several algorithms RC103

### 4.6 Comparison with other Algorithms

Table 3 shows the AFPSO, and the PSO (PSO) and the improved nearest neighbouring (NNA) to solve the Solomon 15 instance. The table shows that the results of AFPSO are much better than the results of nearest neighbouring heuristic method. And AFPSO comparison with PSO also have dramatically increased.

Table 3 Comparison of AFPSO with other algorithms

| | AFPSO | PSO | NNA |
|---|---|---|---|
| C101 | 153.8 | 184.3 | 283.7 |
| C102 | 190.1 | 215.6 | 271.4 |
| C103 | 169.4 | 189.3 | 251.3 |
| C104 | 198 | 225.4 | 254.9 |
| C105 | 151.6 | 182.7 | 246.3 |
| R101 | 464.3 | 482.5 | 578.4 |
| R102 | 449.5 | 465.7 | 553.1 |
| R103 | 447.8 | 454.2 | 521.6 |
| R104 | 432.6 | 471 | 573.2 |
| R105 | 443.2 | 498 | 530.5 |
| RC101 | 314.5 | 346.1 | 491.1 |
| RC102 | 311.3 | 342.4 | 421.4 |
| RC103 | 268.7 | 321.3 | 407.8 |
| RC104 | 269.1 | 324.6 | 423.6 |
| RC105 | 308.6 | 336 | 435.3 |

## 6. CONCLUSIONS

This paper studies the PSO for open vehicle routing problem with time dependent vehicle routing problem. A continuous speed time-dependent model is presented. The improvement PSO with adaptive adjustment of inertia weight strategies and updating rules based on information entropy is presented. In the experiments, parameters affected the optimization result is discussed, such as the population size of the algorithm, the numbers of iteration. The experimental results showed that with the increase of population size, the optimization results become better, and more stable. But population size reaches a certain level; the results of the optimization stabilized and are no longer changed significantly with the increase of population size. With increasing the number of iteration, the result is better and search excellent solution to the increasingly high probability. We compare the Adaptive inertia weight adjustment with other weight adjustment method. Because our methods fully use the algorithms' information, optimize results improved markedly. The updated strategies effectively increase the diversity of the particles and improve the optimization results. Finally, compare the particle swarm optimization with other algorithms by the benchmark. The result shows the algorithm that the paper present is the efficiency method for the open vehicle routing problem with time dependent.

## REFERENCES

Jung S.A(2000), Genetic algorithm for vehicle routing problem with time dependent travel times.

Yang-Byung Park(2000), A solution of the bicriteria vehicle scheduling problems with time and area-dependent travel speeds, *Computers & Industrial Engineering*, 38(1):173-187.

Ichoua S, Gendreau M, Potvin J-Y(2003),Vehicle dispatching with time-dependent travel times . *European Journal of Operational Research*, 144(2): 379-396.

AlbertoV. Donati, Roberto Montemanni, and Norman Casagrande(2006),Time dependent vehicle routing problem with a multi ant colony system, *European Journal of Operational Research*,2006,online.

Ali Haghani and Soojung Jung(2005), A dynamic vehicle routing problem with time-dependent travel times, *Computers & Operations Research*, .32(11):2959-2986

Jean-Yves Potvin,Ying Xu,Ilham Benyahia(2006),Vehicle routing and scheduling with dynamic travel times, *Computers & Operations Research*,.33(4):1129-1137.

Jose Brandao(2004), A tabu search algorithm for the open vehicle routing problem, *European Journal of Operational Research* ,157(3), 552-564

Fu, Z., Eglese, R., Li(2005), A new tabu search heuristic for the open vehicle routing problem. *Journal of the Operational Research Society*, 57(8), 267-274.

Tarantilis C, Ioannou G, Kiranoudis C(2004). A threshold accepting approach to the open vehicle routing problem. *RAIRO Operations Research*, 38, 345-360.

Wu Qi-Di, Wang Lei(2005), *Intelligent Particle Swarm Optimization Algorithm Research and Application*, Jiang Su Education Publisher.

Zhang Xuanping,Du Yuping,Qin Guoqiang(2005), Adaptive Particle Swarm Algorithm with Dynamically Changing Inertia Weight, *Journal of Xi'an Jiaotong University*, 39(10) :1039-1042.

A.Chatterjee and P.Siarry(2006), Nonlinear Inertia Weight Variation for Dynamic Adaptation in Particle Swarm Optimization, *Computers & Operations Research*, 33(3): 859-871.

Bo Liu, Ling Wang, Yi-Hui Jin,et.al(2005). Improved particle swarm optimization combined with chaos, *Chaos, Solitons & Fractals*, 25(5):1261-1271.

Wang Wanliang, Wu Bin, Zhao Yanwei(2006), Particle Swarm Optimization for Open Vehicle Routing Problem, Lecture Notes in Artificial Intelligence,Vol.4114. 999-1007.

## Acknowledgement

This project is supported by the National Natural Science Foundation of China (Grant No.60573123), The National 863 Program of China (Grant No.2007AA04Z155), the Plan of Zhejiang Province Science and Technology (Grant No. 2007C21013).