International Journal of Hybrid Information Technology
Vol.9, No.1 (2016), pp. 367-378
http://dx.doi.org/10.14257/ijhit.2016.9.1.31

Regional Traffic Timing Plan Optimization Based on Improved Particle Swarm Algorithm

Zhongyu Li, Bing Li, Keli Chen, Tao Yang and Honge Li

School of Mathematics and Computer Engineering, Xihua University, Chengdu, China
[1]125938422@qq.com, [2]libing_lyl@163.com

Abstract

The modern traffic control timing plan aims at raising transportation efficiency of vehicles by making full use of existing road resources. So we built a region traffic control model; its performance of model is measured by minimum average travel time of vehicles, its average delay model for single crossroad is derived from the revised HCM2000; and then it is transformed into objective function of optimization. To enhance the global and local searching ability, the improved exponential inertia weight factor is taken for the particle swarm algorithm. So basing MATLAB simulation tool, we calculated timing plan on under-saturation traffic flow condition and saturation traffic flow condition by the classic particles swarm optimization algorithm and improved particle swarm algorithm respectively. The simulation results show that the improved particle swarm algorithm can effectively avoid that the particles fall into local optimal solution; at the same time, the improved algorithm can reasonably distribute green signal time for every phase, the timing plan can effectively shorten average vehicle delay time and improve the transportation efficiency of road.

Keywords: *regional traffic control model, the average travel time, particle swarm algorithm, exponential inertia weight factor, HCM200*

1. Introductions

The fast urbanization pace is accompanied by serious traffic congestion problems. To solve traffic congestion problem, the most effective way is to build more new roads and improve optimal management of the existing roads. So on the condition of existing road, the optimization of traffic control to ease traffic jams is one of the feasible and effective solutions. Now, urban traffic control mode is mainly divided into single crossroad control, trunk control and regional control [1]. With the expansion of urban, the insular crossroad control mode and trunk mode are not enough to meet with real traffic scene. Therefore, the regional traffic control model [2, 4] has become a research hotspot.

In abroad, famous Webster equation used in calculating delay time is a milestone [3, 15]; then Australia scholars introduced profit compensation coefficient in it and combined the vehicle delay and the number of vehicle stopping together. In addition, in the most famous American Road Traffic Capacity Manual, HCM2000 model about the traffic capacity and the evaluation model of the delay can describe delay time of the real crossroad more accurately [4, 13]. At home, scholar Zhang hui and some others proposed urban traffic coordination control based on distributed Q-learning, and also proved that this control method is more effective than the timing control method [5]. Lei Lei and some other scholars proposed oversaturated crossroad group system model, and the system cost includes the all crossroad group delay times of main roads in the model [6]. Liu qin and some other scholars proposed traffic regional coordinated control model, its performance index is maximum numbers of departing vehicle in a

ISSN: 1738-9968 IJHIT

region and minimum numbers of awaiting vehicle in all crossroads[7]. Basing on above evaluation models, this paper proposed an evaluation model based the shortest average travel time in region to meet the improved particle swarm algorithm.

Particle swarm optimization (*PSO*) is proposed by Dr. E-Berhart and Dr. Kennedy, this kind of intelligent optimization algorithm is based on the feeding behavior of birds. Particle swarm optimization algorithm[10-12] can realize the multidimensional complex space search and at last get the optimal solution through cooperation and competition between particles [8, 9, 16]. This paper improved the *PSO* algorithm from angle of convergence speed and avoiding local optimal solution, the improved key is inertia weight factor of particle swarm algorithm; namely, the decreasing composite exponential function is taken for the inertia weight factor. In this paper, initial population still uses random initialization, so that the diversity of population is subject to certain restrictions, but at the same time, the timing plan of crossroad is closely related to parameter settings of the established regional traffic model. The experimental results show that the improved particle swarm optimization algorithm in solving region traffic can avoid local optimal solution effectively to a certain extent, improve traffic conditions, and alleviate traffic pressures.

2. Mathematical Model

For the region traffic model, vehicle delay time and the number of stops are very important evaluation indexes, minimizing the average delay time of vehicles is a key index in a lot of research results usually [6]. This paper use the average travel times as the evaluation index of each crossroad; and then set up a corresponding mathematical model. Delay time in HCM2000 average delay model [4] is composed of delay time of lane group, average delay times of crossroad can be obtained by the sum of lane group delay times. In fact, HCM2000 model only need simple computing, it is easy to be understood. But it is not suitable for region because it is based on the single crossroad delay, so can only be used for single crossroad [14]. So we need to improve the model and set up a corresponding regional model. In this paper, the model of the minimum regional average travel times is derived from this idea.

2.1. HCM2000 Model

The vehicles average delay equation of crossroad given by the American Highway Capacity Manual, *i.e.*, HCM2000 delay time model considers a variety of different conditions, such as the vehicle arriving ratio during the green light time and the delays calculation on different saturation degree. The computation equation is as follows [4]:

$$d_{ij} = d_1 \bullet PA + d_2 + d_3 \qquad (1)$$

For some single crossroad, d_{ij} represents the vehicle average delay of lane group j at phase i. d_1 represents delay value when vehicles arrival uniformly. d_2 represents incremental delay times due to that the vehicles arrive at crossroad randomly. d_3 represents delay times caused by the existing queue delay before some analysis time period. If it is on the under-saturation condition, the value of d_3 is zero. *PA* represents an adjustable parameter, it has no unit. Its value is related to the traffic condition and the green signal ratio, the traffic condition is divided into under-saturation and saturation, the value of *PA* is less than or equal to 1. It is less than 1 on under-saturation condition; it equals to 1 on saturation and oversaturated conditions. So the *PA* can be shown as follows on under-saturation condition:

$$PA = \frac{(1-p)f}{1-\frac{Green_i}{Cycle}} \qquad (2)$$

p represents ratio of vehicle arrival, f represents a verification factor.

$$d_1 = \frac{Cycle(1-\frac{Green_i}{Cycle})^2}{2\bullet(1-\min(S_{ij},1)\bullet\frac{Green_i}{Cycle})} \qquad (3)$$

Cycle represents the cycle time of traffic signal light, *Green* is green signal time in a cycle, S_{ij} is the saturation degree of lane *i* at phase *j*.

$$d_2 = 900T[(S_{ij}-1)+\sqrt{(S_{ij}-1)^2+\frac{8IS_{ij}k}{cT}}] \qquad (4)$$

In the equation (4), *T* represents analysis time period for studying the delay times; its general experience value is 0.25h. I represents the impact factor which represents influence of the upstream vehicle to the downstream vehicle, value of *I* is 1 when only consider a single crossroad. *K* represents a verification coefficient used to control crossroad. *C* represents the traffic capacity of some crossroad.

$$d_3 = \frac{1800Q(V+1)t}{cT} \qquad (5)$$

In the equation (5), *Q* represents the vehicle queue length in the analysis time interval. *V* represents the delay parameter of traffic, *t* represents the queuing time of vehicle in analysis time period. The parameters of *V* and *t* are computed according to the traffic saturation degree in the analysis time period. On under-saturation condition, the value of *t* is zero, otherwise:

$$t = \min\{T, \frac{Q}{c[1-\min(s_{ij},1)]}\} \qquad (6)$$

For the parameter *V*, the value of *V* is zero when *t* is less than *T*, otherwise:

$$v = 1-\frac{cT}{Q\bullet[1-\min(s_{ij},1)]} \qquad (7)$$

In summary, the average delay time can be described by following equation:

$$\begin{cases} d_{kij}=\frac{Cycle_k(1-\frac{Green_{ki}}{Cycle_k})^2}{2\bullet(1-\min(S_{kij},1)\bullet\frac{Green_{ki}}{Cycle_k})}\bullet PA+900T[(S_{kij}-1)+\sqrt{(S_{kij}-1)^2+\frac{8IS_{kij}k}{cT}}] \\ \quad +\frac{1800Q(V+1)t}{cT} & ;\ if\ \ s_{kij}\geq 1 \\ d_{kij}=\frac{Cycle_k(1-\frac{Green_{ki}}{Cycle_k})^2}{2\bullet(1-\min(S_{kij},1)\bullet\frac{Green_{ki}}{Cycle_k})}\bullet PA+900T[(S_{kij}-1)+\sqrt{(S_{kij}-1)^2+\frac{8IS_{kij}k}{cT}}] & if\ s_{kij}<1; \end{cases} \qquad (8)$$

In above equation, *k* represents crossroad number, *i* represents phase number, *j* represents lane group number. The constraint condition for a crossroad is that $\sum_{i=1}^{n} Green_i$ equals to cycle time. That is to say, sum of each phase time is equal to the cycle time.

2.2. Regional Model

A single crossroad model consists of several phases and lanes. From HCM2000 Model, the average delay time D_k of crossroad *k* can be obtained.

$$D_k = \sum_{i=1}^{m}\sum_{j=1}^{n} d_{ij} \qquad (9)$$

In the equation (9), parameter *i* represents the phase number, *j* represents the lane group number.

*H*CM2000 Model is only suitable for single crossroad. For a region, it consists of several crossroads, so the total travel time in a region can be described by following equation:

$$DT = \sum_{k=1}^{d} D_k + \sum_{l=1}^{L} TIME_l \quad (10)$$

In the equation (10), *DT* represents the total travel time of vehicles in the region; D_k represents the average delay time at the crossroad *k*. *k* represents the crossroads number in a region. $TIME_l$ represents the travel time on road section between adjacent crossroads; *l* represents the road section number in a region. $TIME_l$ can be described by the following equation from *BPR:*

$$TIME_l = T_{00}[1+\alpha(\frac{q_l}{pa})^{\beta}] \qquad (11)$$

In the equation (11), T_{00} represents the free travel time in a section, *q* represents the traffic flow in some section, *pa* represents the traffic capacity in some section. *α* and *β* represent adjustable parameters. So equation (10) is the evaluation model of performance index.

3. Particle Swarm Optimization Algorithms

Similar to genetic algorithm, particle swarm optimization algorithm is also based on the fitness function and the population. In the particle swarm algorithm, the position of particle represents the potential solution of the optimized object in the search space; all particles have a fitness value which is determined by the optimized function. Each particle has a speed, it determines the particle's flight direction and distance, and other particles would follow the current optimal particle to search in the solution space. Particle swarm algorithm is initialized to a group of random solutions firstly, and then finds the optimal solution through iterative processes; in every step, the particle updates itself by tracking two extreme values, one is the optimal solution which particle itself found currently, known as the individual extreme value; another extreme value is the optimal solution which the whole population found currently, known as the global extreme value. But like other intelligent algorithms, the particle swarm optimization (*PSO*) algorithm is easy to fall into local optimal solution because of premature convergence. Now there are many ways to solve this problem, our tactics is to improve inertia weight factor based on composite exponent.

3.1. Basic Particle Swarm Algorithm

As mentioned above, the particle swarm algorithm is a search model based on the position and speed. Each particle represents a candidate solution in the solution space; the quality of solution is decided by the fitness function. We suppose that there is one particle in the space named *d*, its speed is V_i＝ (V_{i1},V_{i2},V_{i3}……V_{id}), so the speed determines its position, namely, solution of fitness function. When the particle swarm optimization (*PSO*) algorithm is initialized, its position is P_i＝(P_{i1},P_{i2},P_{i3}……P_{id}). In iteration, the particle would update its own speed and position by tracking two extreme values, *i.e.*, individual extreme value B_i＝(B_{i1},B_{i2},B_{i3}…) and global extreme value G_i＝ (G_{i1},G_{i2},G_{i3}…). Updating equations of basic particle swarm algorithm are as follows:

$$\begin{cases} v_i(k)=v_i(k-1)+c_1\bullet rand()\bullet(B_i-P_i(k-1))+c_2\bullet rand()\bullet(G_i-P_i(k-1)) \\ P_i(k)=P_i(k-1)+v_i(k-1) \end{cases} \qquad (12)$$

In the equation (12), c_1 and c_2 are called learning factor, and also called accelerating factor, their role is to make the particles have the independent learning ability and have ability to learn from social swarm. Then each particle can fly near to the optimal position of itself in history and also can fly near to the optimal position of population in history. If the two factors are adjusted properly, they can accelerate the convergence speed of PSO, and also can make it jump out of local optimal solution. The value of function *rand()* is the random number between zero and one, it is used to maintain the diversity of population. In the process of initialization, we must limit the maximum speed and position of particles according to actual condition. In the equation (13), the second part represents the independent learning ability of particle; its essence is particle's learning and memorizing ability. The third part represents the group learning ability; its essence is ability that individual particle learns from the population.

3.2. Classic Particle Swarm Algorithm

References [8, 9] have shown that convergence performance of basic particle swarm algorithm is poorer, and it is easy to fall into local optimal solution. Shi and Eberhart put forward a general inertia weight factor of particle swarm optimization algorithm; its update equation of algorithm is as follows:

$$\begin{cases} v_i(k)=W\bullet v_i(k-1)+c_1\bullet rand()\bullet(B_i-P_i(k-1))+c_2\bullet rand()\bullet(G_i-P_i(k-1)) \\ P_i(k)=P_i(k-1)+v_i(k-1) \end{cases} \tag{13}$$

W is inertia weight factor, references [8, 9] have shown that W can improve the convergence performance; W has the ability of weighing the local optimal solution and the global optimal solution; it can effectively reduce the premature probability. When the value of the inertia weight factor is smaller, the particle swarm algorithm is easy to fall into local optimum, but the convergence speed is faster. When the value of the inertia weight factor is bigger, it can expand the searching scope, and it is easier to find the global optimal solution, but it would lead to slower convergence speed and much more iteration numbers. So Shi and Eberhart improved the update equation about W.

$$w = w_{max} - \frac{w_{max} - w_{min}}{N_{max}} \bullet n \tag{14}$$

In the equation (14)，w_{max} is the maximum value of inertia weight factor, w_{min} is the minimum value of inertia weight factor, N_{max} is the maximum iteration numbers, n is the current iteration numbers. The experience value of w_{max} is 0.95; the experience value of w_{min} is *0.4*.

3.3. Improved Particle Swarm Algorithm

In a word, Shi and Eberhart put forward the improving strategy; *i.e.,* inertia weight factor of particle swarm optimization algorithm is based on strategy of linear decreasing. But the above-mentioned particle swarm optimization algorithm is easy to fall into precocious phenomena when it is used for traffic timing plan. To solve the premature problem, this paper puts forward a nonlinear inertia weight factor strategy for improving particle swarm optimization algorithm. When the *PSO* is used for the optimization problem, it is easy to produce the phenomenon of "gather". That is to say, when one or a number of particles are at a so-called "optimal" position, the other particles would close up to this position. If this position is not the global optimal position, the whole iterative process would fall into a local optimal solution.

As above discussing, the change of the inertia weight factor value can affect the search ability of the algorithm and position of solution vector. A larger inertia weight factor value is helpful to improve the global search ability of the algorithm; the smaller inertia weight factor value can increase the local search ability of the algorithm. If In the initial stages of algorithm, a larger inertia weight factor is need to increase the global search ability; with the increasing of the iteration numbers, it need to strengthen the local search ability to avoid that the particles are away from the global optimal solution. Therefore, in this paper, the inertia weight factor based on the composite exponential decreasing function can meet the requirements of searching optimal solution. Let's do some simple qualitative explanation to the strategy. At the beginning stage of algorithm, the iteration numbers are less, exponent value is close to 1, therefore larger inertia weight factor value can be obtained and it is helpful for global searching. At the same time, the exponential function decreases quickly at beginning stage; so it can improve the convergence speed of the algorithm. With the increasing of the iteration numbers, the exponential value decrease more slowly, so the inertia weight factor also reduce more slowly; at the later stage of iteration, it must improve the local search ability, and the smaller inertia weight factor values are in favor of local search. So the improved strategy can meet with the requirement of stronger local search ability at the later stage of iterations. Compared with the linear declining strategy of classic particle swarm algorithm, inertia weight factor of improved algorithm is bigger at the beginning stage of iteration, so it increase the global search ability; at the later stage of the iteration, the exponential function decline slowly and tends to stable, its declining speed of function is far less than linear function, So this is more conducive to the local optimization. The improved updating equation of speed and position is as follows:

$$\begin{cases} v_i(k) = w_e \left(\frac{w_s+w_e}{w_s-w_e}\right)^{\frac{1}{1+\frac{u \cdot n}{N}}} \bullet v_i(k-1) + c_1 rand \bullet (B_i - P_i(k-1)) + c_2 rand \bullet (G_i - P_i(k-1)) \\ P_i(k) = P_i(k-1) + v_i(k-1) \end{cases} \quad (15)$$

In the equation (15), w_e represents the inertia weight factor when evolve to the maximum iteration numbers, w_s is the inertia weight factor under the initial state. We take the value 0.4 for w_e and take the value 0.95 for ws; they are derived from large number of experiment of Shi and Eberhart [8]. u is an adjustable parameter, n is the current iteration numbers, N is the maximum iteration numbers. From experimental process, different value u can lead to different experimental results, larger value u can make the exponential function values decrease too fast, so it can significantly increase the convergence speed of the algorithm, but it is bad for the convergence precision. Smaller value u is bad for the convergence speed, but it is helpful for expanding the searching scope. After many experiments, we take the value 9 for u in this paper.

4. Experimental Schemes

In this paper, we use the minimum travel time of vehicles in some traffic region as evaluation index of model performance, *i.e.,* the fitness function for the algorithm is min*(DT)*. We suppose that some traffic region includes four crossroads, traffic region structure is shown in Figure 1, the crossroad number is from *1* to *4*，the label of road section is corresponding to *A~D* respectively，the road sections are two-way and four lanes, and its lanes are divided into two types, namely, straight direction and turn-left direction. Physical property of road sections is shown in Table 1; Crossroad traffic is controlled by four phase signal, the phase sequence is shown in Figure 2. Among them, Phase 1 represents straight and turn-right at north-south direction; phase2 represents turn-left at north-south direction; phase 3 represents straight and turn-right at west-east direction; phase 4 represents turn-left at west-east direction. The sequences of phases in a cycle are from phase 1to phase 4. To demonstrate the efficiency of algorithm, every crossroad and every phase have same passing right.

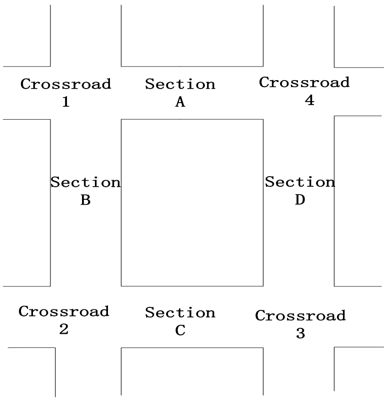

Figure 1. Road Network Structure

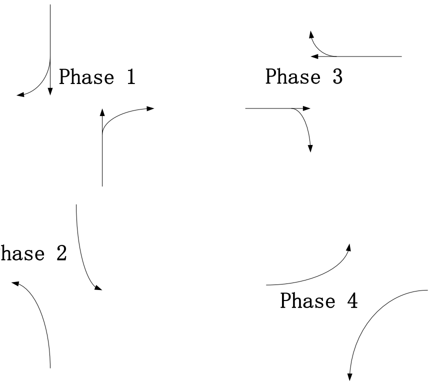

Figure 2. Phase Sequence in a Cycle

Table1. Physical Property of Road Sections

Road Name	Road length (unit: m)
Section A	800
Section B	1000
Section C	700
Section D	600

It's not necessary to optimize when the traffic flow exceeds a certain number, so we only consider under-saturation traffic flow and saturation traffic flow. That the traffic flow is less than 1200pcu/h is defined as under-saturation condition and that the traffic flow is between 1200pcu/h and 2200pcu/h is defined as saturation condition. The traffic flow of each crossroad used in experiments is shown in Table *2*.

Table2. Traffic Flow in Road Section

Road Section	Under-saturation (unit: pcu/h)	Saturation (unit: pcu/h)
Section A	500	1600
Section B	600	1700
Section C	800	1900
Section D	700	1800

In *MATLAB* simulation experiments, all parameters and conditions are as following, they are suitable for original particle swarm algorithm and improved particle swarm algorithm. Lane groups in model are divided according the phase sequences, the phase 1~4 are corresponding to lane group 1~4. The minimum green signal time is 20s and the maximum green signal time is 60s by the international standard. The maximum cycle time is 130s. The driving speed of vehicles in all road sections is 40km/h, vehicles arrive at crossroad randomly, and the numbers of arriving vehicles obey random distribution regularity. Saturation degree value S_{kij} for lane group j of crossroad k at phase i is a random number between 0 and 0.8 on under-saturation condition, and it is between 0.8 and 1.2 on saturation condition. For the simulation experiment using *MATLAB*, the classic particle swarm algorithm with general inertia weight factor is firstly used to optimize the region traffic model, and then the improved particle swarm algorithm is used. The population is initialized by random initial function; the population scale is set to 500, the iteration numbers are set to 500. According to experience, the learning factor is set to 2, the initial weight value and final value is set to 0.95 and 0.4 respectively. Due to the time length of each phase is obtained by saturation, so there is no need to set the initial value. *MATLAB* simulation tool automatically takes four bit precision for experimental data, so it results in the timing value with four bit decimal. In practical application, the results may be added 1 second firstly and then rounded. Only a bit decimal is reserved for experimental results in this paper.

5. Experimental Results and Discussion

Table 3 shows experimental results with the classical algorithm and improved algorithm on the under-saturation condition of traffic flow, the value in Table 3 is the timing plan at different phase and crossroad. The improved plan can reasonably distribute the green signal times according to different saturation degree of different phase. But the green signal times of different phase are very close by the original algorithm; there is no reasonable difference of green signal times even due to different saturation degree; this shows that the original algorithm has fallen into local optimal solution. Usually, the green signal time length of every phase is from *20s* to *60s* according to the actual road situation of my country. So the green signal times of every phase at different crossroad are nearly same by the algorithm with general inertia weight factor, it cannot distribute the effective green signal times according the different saturation degree of lane group; compared with the improved algorithm, its cycle time is longer. But the green signal times are reasonable according to traffic flow of different lane group by the algorithm with exponential inertia weight factor; this shows that the *PSO* jump out of local optimal solution too. So the algorithm with improved inertia weight factor enhanced the ability of global searching optimal solution, the effective green signal times are distributed reasonably, which can shorten the delay time and driving time in a region. Figure 3 shows the different delay times comparison of two algorithms on under-saturation condition

Table 3. The Experimental Results of Under-saturated Condition (Unit: s)

Crossroad Number	Improved Method					Original Method				
	Phase 1	Phase2	Phase 3	Phase 4	Cycle	Phase 1	Phase 2	Phase 3	Phase 4	Cycle
Crossroad 1	27.2	21.9	24.5	22.4	96.0	25.0	24.2	23.9	24.7	97.8
Crossroad 2	24.9	24.0	25.3	22.3	96.5	24.9	26.3	24.5	26.2	101.9
Crossroad 3	23.5	22.4	27.2	25.7	98.8	26.7	26.1	26.4	28.6	107.8
Crossroad 4	27.0	23.9	24.3	22.6	97.8	27.4	25.1	26.7	26.5	105.7

On the under-saturation condition, we respectively done 50 experiments for the two algorithms on the same ordinary computer, the original algorithm took 12.3s averagely and the improved algorithm took 13.8s averagely. Although the improved algorithm increases the time complexity and space complexity, it can still meet with the requirement of real time traffic control in practical application.

Table 4 shows experimental results with the classical algorithm and improved algorithm on the saturation condition of traffic flow, the value in table 4 is the timing plan at different phase and crossroad. The timing values in original algorithm are very near, it cannot demonstrate the difference due to different saturation degree; compared with the improved plan, its cycle time is longer. This is because it is easy to fall into local optimal solution. The improved algorithm enhances the ability of global searching and can make it jump out local optimal solution; it can reasonably distribute the green signal times according to different saturation degree of different phase and shorten the cycle times. Figure 4 shows the different delay times of two algorithms on saturation condition. The delay time by improved algorithm optimization is short than by original algorithm.

Table 4. The Experimental Results of Saturated Condition (Unit: s)

Crossroad Number	Improved Method					Original Method				
	Phase 1	Phase 2	Phase 3	Phase 4	Cycle	Phase 1	Phase 2	Phase 3	Phase 4	Cycle
Crossroad 1	29.0	24.3	26.7	25.8	105.8	28.6	28.6	25.6	27.3	110.1
Crossroad 2	27.6	24.6	30.2	25.5	107.9	29.4	30.9	26.8	25.6	112.7
Crossroad 3	25.8	28.7	31.9	24.9	111.3	30.1	31.0	27.2	30.3	118.6
Crossroad4	29.1	27.5	27.9	24.0	108.5	31.2	29.3	29.7	26.6	116.8

On the saturation condition of traffic flow, we respectively done 50 experiments for the two algorithms on the same ordinary computer, the original algorithm took 16.2s averagely and the improved algorithm took 17.8s averagely. Although the improved algorithm increases the time complexity and space complexity, it can still meet with the requirement of real time traffic control in practical application.

Table 5 shows the delay times computed by two algorithms on under-saturation condition and saturation condition. Figure 3 and Figure 4 are derived from data in Table 5, they shows the delay times comparison by two algorithm; they are drawn by drawing program of *MATLAB*.

Table 5. Delay Time at Crossroad (Unit: s)

Under-saturation			Saturation	
Crossroad Name	Improved Method	Original Method	Improved Method	Original Method
Crossroad 1	18.5	21.8	41.6	42.5
Crossroad 2	19.6	22.5	44.4	46.6
Crossroad 3	23.8	25.0	48.0	50.7
Crossroad 4	21.3	23.2	46.7	48.9

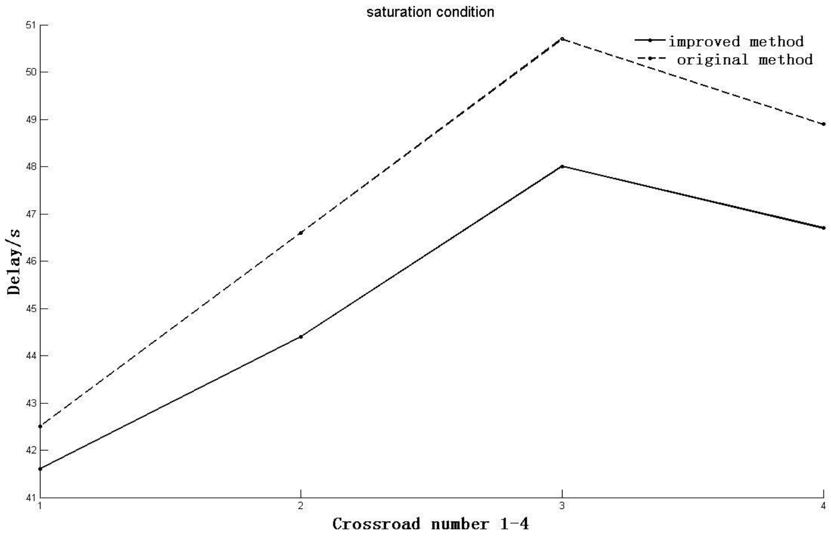

Figure 4. The Delay Time Comparison on Saturated Condition

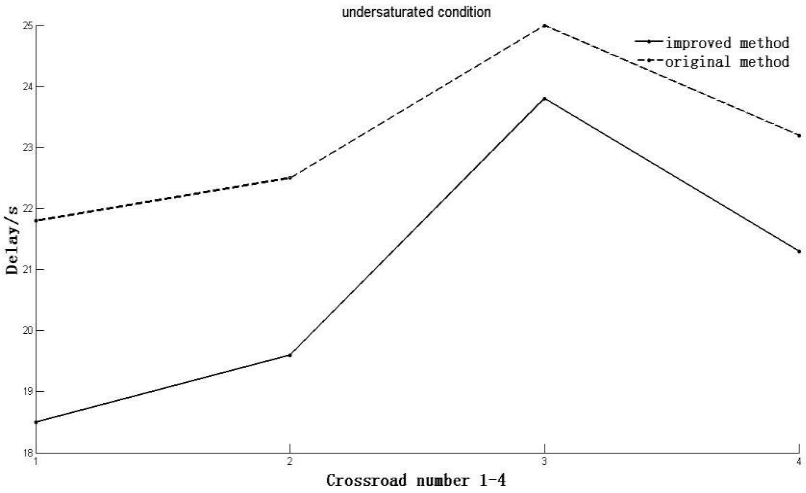

Figure 3. The Delay Time Comparison on Under-Saturated Condition

6. Conclusions

In this paper, a region traffic control model is built; the model includes four main crossroads; the index of performance evaluation is the minimum average driving time of vehicles in some region, the performance evaluation is derived from HCM2000. The timing plan of traffic control model is optimized by particle swarm algorithm; we improved the inertia weight factor because the classic particle swarm is easy to fall into local optimal solution. So we took componential inertia weight factor, it is nonlinear; it can enhance ability of global searching and local searching because of its componential declining property. The experimental results of MATLAB simulation show that the improved algorithm can reasonably distribute the green signal times according the different saturation degree and shorten the delay times and driving times in some region. Although the improved strategy increases the complexity of algorithm, it can still meet the requirement of real-time traffic control. The experimental results only demonstrate the efficiency of improved algorithm. In fact, the traffic optimization involves a variety of factors; for example, the initial green signal time closely related to historical experience data of traffic flow, phase sequence of every crossroad. So these parameters and conditions must be adjusted in practical traffic control system. The future work would focus on complex model with more crossroads.

Acknowledgements

The project was supported by the key item of Sichuan Province Education Department (NO. 13ZA0032) and Xihua university item YJD20140005.

References

[1] K. M. Ng, M. B. Reaz, M. A. M. Ali and T. G. Chang, “A brief survey on advances of control and intelligent system method for traffic-responsive control of urban networks”, Tehnički vjesnik, vol. 20, no. 3, **(2013)**, pp. 555-562.

[2] C. K. Lee, I. Yun, J. H. Choi and S. J. Ko, “Evaluation of semi-actuated signals and pedestrian push buttons using a microscopic traffic simulation model”, KSCE Journal of Civil Engineering, vol. 17, no. 7, **(2013)**, pp. 1749-1760.

[3] F. V. Webster, “Traffic signal settings”, Road Research Technical Paper, vol. 39, no. 1, **(1958)**, pp. 1-39.

[4] A. L. Cunha and J. R. Setti, “Truck Equivalence Factors for Divided, Multilane Highways in Brazil”, Procedia Social and Behavioral Sciences, **(2011)**, pp. 248-258.

[5] Z. Hui, Y. Yuzhen and L. Zhen-Long, “Urban traffic coordination control based on distributed Q-learning”, Journal of Wu Han University of Technology, vol. 31, no. 6, **(2007)**, pp. 1121- 1124.

[6] L. Lei, W. Yang and L. Yu-Gang, “System modeling and optimization model of oversaturated intersection group”, Computer Engineering and Applications, vol. 46, no. 4, **(2010)**, pp. 26-28.

[7] L. Qin and X. Jian-Min, “Traffic regional coordinated control model”, Journal of transportation engineering, vol. 12, no. 3, **(2012)**, pp. 108-112.

[8] J. Kennedy and R. E-berhart, “Particle swarm optimization”, Proceedings of IEEE international conference on neural networks, vol. 4, no. 2, **(1995)**, pp. 1942~1948.

[9] Y. Shi and R. E-berhart, “A modified particle swarm optimizer”, Evolutionary Computation Proceedings, IEEE World Congress on Computational Intelligence; the IEEE International Conference on IEEE, **(1998)**, pp. 69~73.

[10] W. Qing-Hong, Z. Ying and M. Zong-Min, “Particle swarm optimization algorithm and its application were reviewed”, Micro computer information, no, 30, **(2010)**, pp. 34-35.

[11] A. Mukhopadhyay and M. Mandalia, “A Hybrid Multi-objective Particle Swarm Optimization Approach for Non-redundant Gene Marker Selection”, Proceedings of Seventh International Conference on Bio-Inspired Computing: Theories and Applications (BIC-TA 2012). Springer India, **(2013)**, pp. 205-216.

[12] Z. Yuan-Dong, F. Zheng-Hua, “With a weighting function learning factor of particle swarm optimization (*PSO*) algorithm”, Computer application, vol. 33, no. 8, **(2013)**, pp. 2265-2268.

[13] Z. Jing, Z. Qian and L. Jun-Qiang, “The pedestrian phases for intersection delay model based on VISSIM”, Roads and trucks, no. 3, **(2012)**, pp. 49-51.

[14] N. Reza and H. Bruce. “Distribution of delay in signalized intersections: Day-to-day variability in peak-hour volumes”, Journal of Transportation Engineering, vol. 138, no. 9, **(2012)**, pp. 1123-1132.

[15] T. Serhan, C. Kubra and O. Mustafa, “Different approaches to minimum delay prediction at single-lane traffic circles in izmir Turkey”, Canadian Journal of Civil Engineering, vol. 40, no. 3, **(2013)**, pp. 274-284.

[16] M. D. Foy, R. F. Benecohal and D. E. Goldberg, “Signal timing determination using genetic algorithms Transportation Research Record 1365 TRB”, Washington D. C: National Research Council, **(1992)**, pp. 108-115.