2011 14th International IEEE Conference on
Intelligent Transportation Systems
Washington, DC, USA. October 5-7, 2011

Neural Network Based Online Traffic Signal Controller Design with Reinforcement Training

Yujie Dai, Jinzong Hu, Dongbin Zhao, IEEE Member and Fenghua Zhu

Abstract— **Traffic congestion leads to problems like delays, decreasing flow rate, and higher fuel consumption. Consequently, keeping traffic moving as efficiently as possible is not only important to economy but also important to environment. Traffic system is a large complex nonlinear stochastic system. Traditional mathematical methods have some limitations when they are applied in traffic control. Thus, computational intelligence (CI) technologies gain more and more attentions. Neural Networks (NNs) is a well developed CI technology with lots of promising applications in traffic signal control (TSC). In this paper, a neural network (NN) based signal controller is designed to control the traffic lights in an urban traffic road network. Scenarios of simulation are conducted under a microscopic traffic simulation software. Several criterions are collected. Results demonstrate that through online reinforcement training the controllers obtain better control effects than the widely used pre-time and actuated methods under various traffic conditions.**

I. Introduction

Traffic congestion caused by increasing traffic flow has become a very serious problem. With traffic demand exceeding capacity of existing surface transportation infrastructure, delays in travel time is only one of the many symptoms of congestion. It also leads to decreasing flow rate, higher fuel consumption and thus has negative economic and environmental effects. According to U.S. Department of Transportation, congestion merely on the urban road network cost American nation about $85 billion per year in longer and less reliable journey times, reduced mobility, increased vehicle operating costs, and environmental degradation [1]. Thus, reduction in road congestion would clearly benefit economy and environment, thus improve the quality of life for people.

TSC is commonly thought as the most important and effective flow control method to provide safe and expeditious travel on roads. Signal control methods have gone through pre-timed control, actuated control and intelligent control. Traffic system is a large complex nonlinear stochastic system. Computational intelligence (CI) methodologies were involved in the solution of traffic control problems. CI is study of adaptive mechanisms to enable or facilitate intelligent behavior in complex, uncertain and changing environments [2]. It facilitates solving problems that were previously difficult or impossible to be solved, providing a feasible way to get an optimal or suboptimal resolution. Besides, CI methodologies have an ability to adapt to dynamic traffic system. So lots of work can be found in publications on the subject of applying CI technologies in traffic control recent years.

NNs is a well developed and prosperous CI technology. Kinds of networks have been developed, such as: single-layer networks, multi-layer networks, self-organizing networks. It has been successfully used in many fields like pattern classification, robotics and prediction. It has also been adopted in TSC by many researchers. Earlier work could be found decades ago, like Spall and Chin used a NNs controller in their S-TRAC (System-wide Traffic-Adaptive Control) to produce optimal instantaneous (minute-to-minute) signal timings while automatically adapting to long-term (month-to-month) system changes [3]. Large number of NN-based traffic signal controllers emerged recent years. Choy and Srinivasan used a back propagation neural network to implement fuzzy control rules in their local real time signal controller which is capable of continuous online learning [4], [5]. The relationship between traffic demands and there timing plans can be modeled using NN approach. Azzam *et al.* constructed a neural network based traffic signal controller to generate real time timing plans according to the prevailing traffic conditions for each intersection [6]. The neural network took real time traffic conditions data as input and generated several traffic time plan parameters as output. NNs have been successfully used in many aspects in TSC, but some problems like how to choose structure of the NNs, how to train the NN are still very personal. Most of the time, design and training of the NNs are complex and experience needed.

In this paper, we adopted a simple three layered NNs to implement traffic signal controller for an intersection and proposed a new simple training method based on reinforcement idea. Tuning of NNs parameters is made online based on a criterion signal.

Remaining part of this paper is organized as follows: Section II and section III introduce design and training of the NN-based signal controller separately. Section IV gives a detailed introduction on the simulation configurations. The results and further discussion are in section V. Section VI concludes this paper.

This work was supported in part by National Natural Science Foundation of China (Nos. 60874043, 60921061 and 61034002).

Yujie Dai, Dongbin Zhao and Fenghua Zhu are with the State Key Laboratory of Intelligent Control and Management of Complex Systems, Institute of Automation, Chinese Academy of Sciences. No.95 Zhongguancun East Road, Haidian District, Beijing 100190, China (phone: +86-10-82619580; fax: +86-10-82619580; e-mail: daiyujie07@mails.gucas.ac.cn, dongbin.zhao@ia.ac.cn, fenghua.zhu@ia.ac.cn).

Jinzong Hu is with the Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, (email: hujz@ciomp.ac.cn)

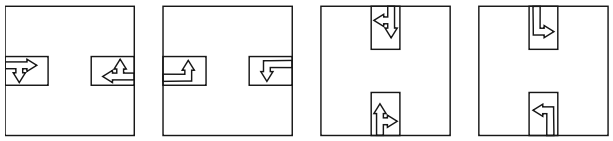

Fig. 1. A signal cycle comprises 4 phases.

II. Design of Neural Networks based Traffic Signal Controller

NNs is always used as some kind of an universal nonlinear approximator of relationship between input and output value. In this section, we will give a brief introduction on the choice of input and output variables of NNs and the mechanism of the proposed controller. Some related basic concepts in TSC are described to lay the ground of further discussion on TSC algorithms.

A. Traffic Basic Notions

Traffic lights are installed at intersections to control traffic flow avoiding collisions between competing movements. Typically there are three colored lights: red, yellow and green whose common meanings are stop, caution and go. Usually different traffic movements on an intersection are classified into several competing groups. Once a group is discharging by the lights (green), others are forbidden. Green status of these grouped movements is called phase. The time a phase lasts is its phase time. All phases of an intersection show up in a time series form a cycle. A typical signal cycle with 4 phases are shown in Fig. 1. For detail definition of these concepts please refer to [13]. TSC is to optimize the traffic flow by adjusting these parameters of the signal plan at an intersection.

The prior TSC method is pre-timed. All parameters like phase time, phase sequence and cycle time, are pre-determined and fixed. So it is also called fixed-time method. Decision of these parameters was based on analysis of historical traffic data via some statistical methods. Senior version of pre-timed method could employ different time plans for different time of day or for special recurring traffic conditions. Some off-line software was developed to calculate optimal signal settings for single intersections or networks. This method is still a prior choice for many urban cities in the world since its easy implementation and maintenance. But traffic system is a dynamic system, any pre-defined traffic signal plans do not fit various traffic conditions. So, traffic responsive idea gradually came into practice with development of sensing technology. Basic idea of traffic responsive control is that signal plan is generated based on real time traffic data collected by detectors located at road network, in contrast with fixed time plans of pre-timed control. The most widely applied traffic responsive control method may be the actuated control. An actuated controller 'responses' to real traffic by adjusting its time plan according to a set of predefined rules. Simply speaking, controller extends time of current phase for a small duration. The phase sequence is also not fixed to a certain extent. Actuated control is more adaptable to situations that traffic saturation is less than 80% and traffic randomness is relatively high. Besides, extension of the phase only depends on vehicles of current phase, without taking queuing situation of other phases into account. Therefore it cannot obtain optimal usage of resources: time and space of intersections.

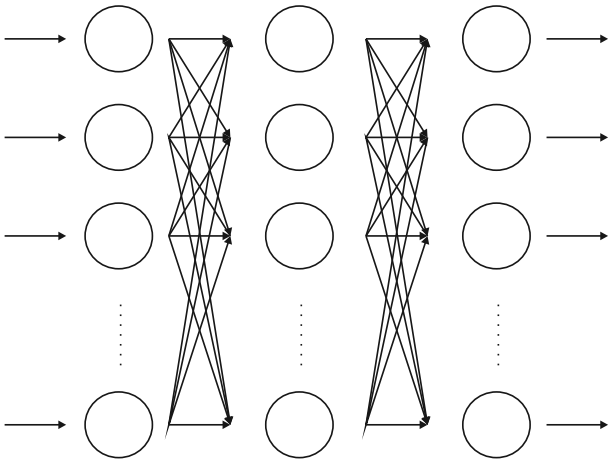

Fig. 2. Three layered feed-forward neural networks.

Our TSC method combines features of both pre-timed and actuated methods together: the phase sequence is strictly fixed like the pre-timed method and phase time is extended like the actuated method. The signal controller in this paper is formed by a neural network. Traffic data such as phase time and number of waiting vehicles of phases are taken as input of the controller. The output of the NN is action of the signal controller which means extend current phase or not.

In next subsection we will introduce the construction of the NN controller.

B. Back Propagation Neural Networks

NN imitates the function of biological neurons in brain and connections between them. By updating weights of neurons, NN learns and memorizes the training data, discovering patterns or features of them. NN can learn arbitrary mapping between any two data sets of real, discrete or vector values which may contain noise. Multi-Layer Perception (MLP), also called feed-forward networks, is a model of NN and its distinguishing feature is that it can approximate any continuous function to any arbitrary accuracy if large enough number of hidden units are used [7], [8].

A three layered sigmoid feed-forward neural network is adopted to accomplish traffic signal controller. A sketch of its structure is shown in Fig. 2. There are three layers: input layer, hidden layer and output layer. The transfer function of hidden and output layer is a Log-Sigmoid function: $y = 1/(1+\exp^{-x})$. The number of neurons in each layer is 240, 120 and 2 separately. Our proposed method and pre-timed method have same phases and sequence shown in Fig. 1. Input neurons of the NN are assigned to time of current green phase, numbers of vehicles waiting in each phase and a group of indicators to indicate the phase is green or red.

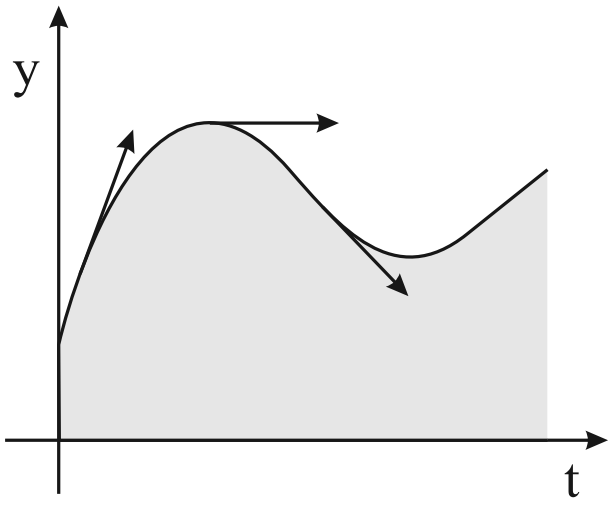

Fig. 3. Fluctuation of vehicles in an intersection.

Output of the NN is assigned the meaning of extending the current phase or not judging by their values.

Training algorithm of NN is the famous error back propagation algorithm. A revised variable step size training method is developed based on results of [9], [10]. This method speeds up the error training process and makes sure it satisfy real time online learning requirement.

When time meets the check point, the traffic data collected are input to the NN. Then the controller decides whether to extend the current green phase by a short duration or terminate it immediately simply according to the output value of NN. After the action has been taken, traffic condition will be evaluated and a suggestion for better actions will be calculated, then the NN is trained to 'learn' this better action. This learning process is done online and the reinforcement of the action is continuous all the time. The reason why we choose action reinforcement training idea and its principles will be comprehensively elaborated in next section.

III. Reinforcement Training

Basic reinforcement training idea comes from nature. It was commonly used when people train animals and when we learning things ourselves. The core idea was encouragement and punishment. This is very simple: good actions will be encouraged and bad actions will be punished. The founders of artificial intelligence (AI) [11] had once proposed that we build artificial brains based on reinforcement learning. This notion was borrowed and reinforced by a machine learning method named reinforcement learning (RL). Through continuous interaction with environment, the controller gradually learned how to respond to the environment in order to get better or best reward. In another words, better actions were reinforced in its learning process.

Now, let's take a look at what we face in the TSC problem. Real time traffic data are collected, and signal controller has to decide what to do based on these data. But the problem is the model of traffic flow is unknown most of the time, or even assuming we have got a precise model, the burden of calculation for optimal action is always unbearable and not timely. That is partly why we resort to CI technologies. They are used to map a better relationship between the conditions we have got and the actions we have to take. Just as Barto said in [12] RL is learning what to do —how to map situations to actions— so as to maximize a numerical reward signal. There is no mathematical solution of the action, so we do not know exactly what action is ought to be taken under certain condition. The idea of the reinforcement learning is that we choose actions that may lead to a better reward, and then reinforce the action according to observed return of environment.

So we proposed a reinforcement training method to train NN to obtain a better map between traffic conditions and actions. The signal controller is a three layered sigmoid feed-forward neural network. Training algorithm is the error back propagation algorithm. The main difference of our method with the RL is that the major reward given to the controller is not a delayed or cumulate one. An important distinguishing feature of RL is a delayed reward. That is because, generally speaking, an action may not affect only the immediate reward but also the subsequent rewards in a certain range. Next I will explain the reason we choose an immediate reward as major part of the final reward and the specific definition of the reward. Consider the sum of stopped vehicles on all approaches of an intersection, the number varies because vehicles enter and discharge. It was a function with respect to time, a typical process was shown in Fig. 3. The number of vehicles increases if the derivative has a positive sign and decreases if the derivative has a negative sign. The value of the derivative indicates the changing rate. So, if it keeps a smaller negative value for a while, the number of waiting vehicles will definitely decrease obviously. We can discover the derivative of stopped vehicles is an effective immediate reflection of control effect. And it is only valid in a small time range, because its effects counteracts with each other in a long time fluctuation. We also take the number of stopped vehicles into consideration in order to keep less vehicles waiting. So, final reward is some form like $r = y + \dot{y}$. Each time step, the reward is calculated and the action of last step is reinforced according to it. The reinforce principle is encouraging good action and punishing bad action. Output of the NN controller is extending current phase or not. Each time better action is encouraged by increase its value and worse action is punished by decrease its value. Their new values and traffic conditions of last step are used to train the NN to obtain a better map between traffic conditions and actions.

Our proposed TSC method can be summarized as follows: At each time step, control action is generated by the NN controller with the input of traffic condition. Then action of last time step is reinforced according to the observed reward form the traffic condition. Learning and training process is online and continuous. To verify effectiveness of this method we conducted several scenarios of simulation in a microscopic software. Pre-timed and actuated control methods are also conducted under the very same traffic

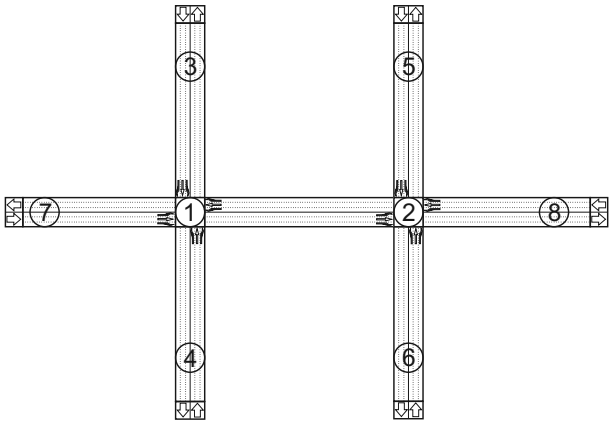

Fig. 4. Surface road network with two intersections.

conditions for comparisons. Our method outperforms them in almost all indices with a remarkable improving and has better adaptive features than them.

IV. Simulation Configurations

To verify effectiveness of our proposed method we conducted several scenarios of simulations in a microscopic software named TSIS (Traffic Software Integrated System) 5.1. Several criteria named MOEs (Measures of Effectiveness) are collected to compare with the widely used pre-timed and actuated control. Pre-timed and actuated control methods are conducted under the very same traffic conditions. Results demonstrate effectiveness and adaptability of our method. The configuration of simulation and parameters of road network built for simulation are depicted this section, results will be discussed in next section.

A. Road Structure

A road network (shown in Fig. 4) is built in TSIS. There are totally eight nodes in this network, node one and two are two intersections with traffic controller, other nodes are entry and exiting nodes used to fill vehicles in and discharge vehicles from this network. Each link connecting these nodes has three channelized lanes. Length of these links is 1000 ft, except two 1500 ft long links between node one and two. Free flow speed of the road network is $30mph$. We do not mention some road structure parameters using default values of TSIS software.

B. Traffic Scenarios

TABLE I
Traffic volume (vph) for entry nodes.

Node	3	4	5	6	7	8
ME	500	500	500	500	500	500
MN	500	500	500	500	600	600
HE	1000	1000	1000	1000	1000	1000
HN	1000	1000	1000	1000	1200	1200

Different traffic patterns are assigned in this road network. In the term of traffic volume, one medium and one relative higher volume traffic condition are designed. In the term of traffic flow a equilibrium and a non-equilibrium pattern are designed. We combine these traffic patterns together and conducted four different scenarios of traffic simulation. They are shortened as ME, MN, HE and HN for convenience. Traffic volumes of entry nodes for every scenario are listed in TABLE I. Traffic turn movements of intersections for non-equilibrium scenarios are shown in TABLE II.

TABLE II
Traffic turn movements (percentage) for intersections.

Link		7-1	4-1	2-1	3-1	1-2	6-2	8-2	5-2
MN	right	25	33	25	33	25	33	25	33
	through	50	33	50	33	50	33	50	33
	left	25	33	25	33	25	33	25	33
HN	right	30	33	30	33	30	33	30	33
	through	40	33	40	33	40	33	40	33
	left	30	33	30	33	30	33	30	33

C. Controller Parameters

Three signal controllers are tested under above-mentioned traffic condition. They are pre-timed, actuated and our reinforcement training method. Pre-timed and actuated controllers are built in TSIS. Our method is developed using C++ following the instruction of RTE (run-time extension), an interface provided by TSIS for the interactions with external applications. For more information about the TSIS software and the RTE interface, please refer to TSIS user manuals and RTE developer's guide[15], [16].

No detector is needed in pre-timed method. For actuated controller, there is a five-foot long presence detector located at the position fifty feet away from the stop line for each phase. For the proposed controller, we just count the stopped vehicle number in each phase in a certain distance using the method provided by TSIS. If the number exceeds fifty, redundancy is discarded. Although there are two intersections in this road network, we do not consider coordination between them till now. So, coordination parameters of pre-timed and actuated controller are taken default parameters or just left without configured.

Because these intersections are not very wide ones, amber time and all-red time for all signal controllers are set as two seconds. The other time parameters for each algorithm are discussed below:

1) Pre-timed: The control cycle of the pre-timed algorithm is shown in Fig. 1. There are four phases with equal time in a control cycle. Green time of each phase varies from 15 seconds to 30 seconds with five seconds as step. Result discussed in next section is the best one of these four time sets in different scenarios. Offset time between the two intersections is zero second.

2) Actuated: Max green time is sixty seconds, min green time is ten seconds and the vehicle extension (extension interval) time is five seconds. No coordination parameter.

3) Reinforcement Training: Phase sequence is the same with pre-timed, and other time parameters are the same with

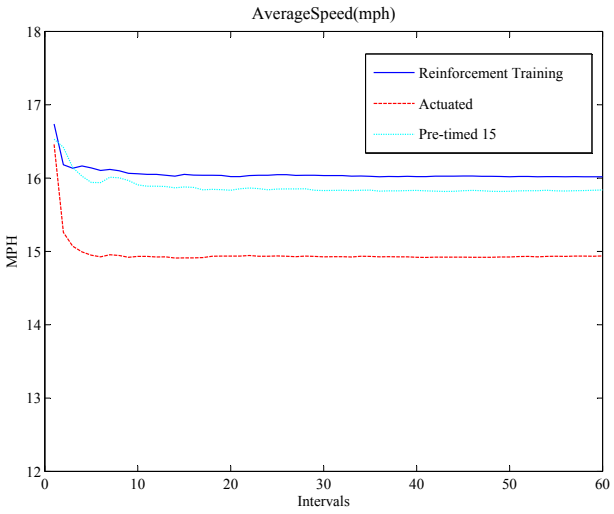

Fig. 5. Average speed of medium and equilibrium traffic condition.

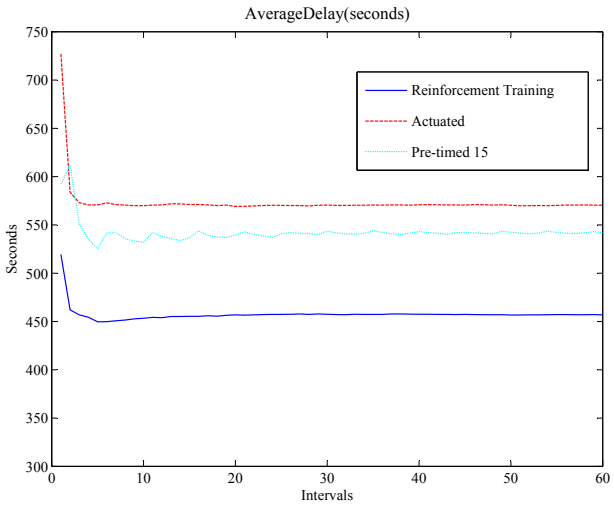

Fig. 6. Average delay of medium and non-equilibrium traffic condition.

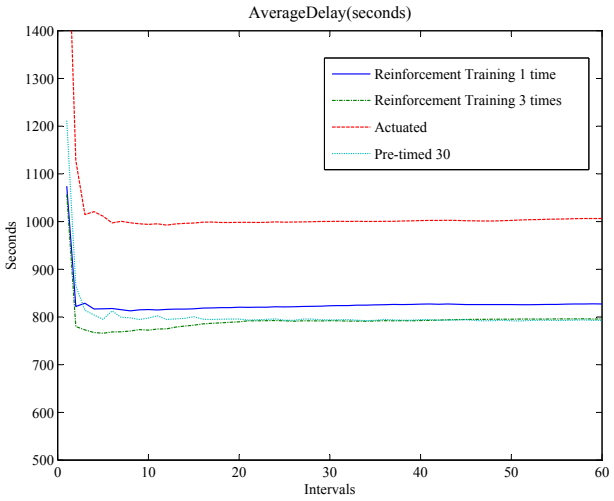

Fig. 7. Average delay of high and equilibrium traffic condition.

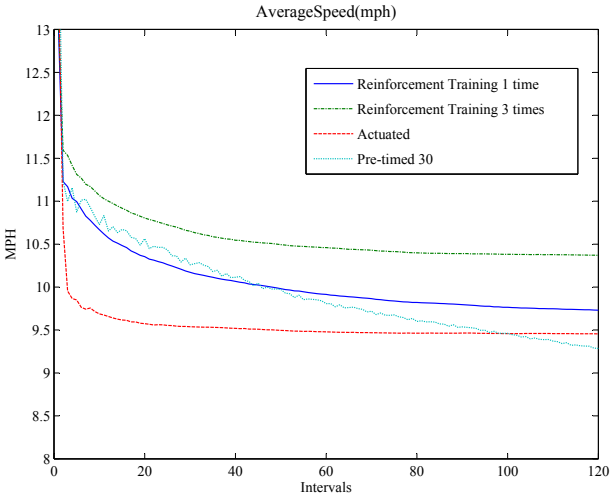

Fig. 8. Average speed of high and non-equilibrium traffic condition.

actuated controller. No coordination parameter.

D. Result Collection

After finishing configuration of TSIS and controllers, aforementioned four scenarios with different controllers are simulated on the road network. For each scenario, the same one hundred iterations (each with a random seed) are conducted. Then selected MOEs of approaches of the two intersections are recorded every minute. Then the averages of MOEs are calculated to evaluate effectiveness of different controllers. For specific definitions of these MOEs, please refer to [17].

V. Results and Discussion

Various tests have been done on the four different scenarios: medium and equilibrium traffic, medium and non-equilibrium traffic, high and equilibrium traffic and high and non-equilibrium traffic. After simulation, several MOEs of approaches of the two intersections are collected. Since paper limitation and similarity of these results, only four groups of results are chosen and posted here. They are Fig. 5, Fig. 6, Fig. 7 and Fig. 8. Selected two MOEs are average delay and average speed. This is because most of the time, people concerns travel delay time and travel speed more than other MOEs. Generally speaking, almost in all collected MOEs, our method performances better than the other two and show some good features they do not have. Our discussions mainly focus on the figures here, but do not limited to them. One thing has to be mentioned is that we conducted one hour's simulation on ME, MN, and HE traffic conditions and two hours' simulation on HN traffic condition. The reason can be seen in Fig. 8, the convergence property is not clearly shown in first hour for all methods. So we extend the simulation time to two hours.

We can discover different specific features for each method. Such as pre-timed method, we can easily found that its best time settings are different in medium and high traffic conditions. The best green time in medium traffic volume is 15 seconds (Fig. 5) but in relatively higher traffic volume, 30 seconds performs best (Fig. 7). Time set has to be changed

only when traffic volume changed let alone variation of traffic turn movements. Obviously, pre-timed method cannot adapt to traffic variations. One 'strange' phenomenon seen in these figures is that actuated controller seems worse than pre-time controller. One reason is that we choose the best one in four pre-timed sets no matter in which traffic condition, but actuated and our method only have one configuration in all conditions. We just want to use this 'unfair' comparison to demonstrate our method has better adaptive feature not to prove that pre-timed controller is better than actuated controller. Another reason was pointed in section II. That is actuated control is more adaptable to relatively high random situations, while tested situations in this paper is relatively steady ones. For actuated controller, one interesting feature can been found: it performs better in high or non-equilibrium traffic condition, this can been seen in Fig. 5 and Fig. 8. This method can quickly converge to some stable point with less vibration than pre-timed method. In these scenarios its time parameters keep the same value while different settings of pre-timed method fit different traffic conditions. This demonstrates its adaptability. For our proposed method, it can be seen in these figures here, has better performance than the best pre-timed method and better adaptability than the actuated method.

Another good feature of our proposed method is its learning ability. In our study, the simulation of every traffic scenario is repeated for several times. A learning process can be noticed in Fig. 7 and Fig. 8. We plot the results of first iteration and third iteration. A remarkable improvement is obtained through repeated training under same traffic condition. The improvements depend on the traffic condition. In severe traffic conditions improvement is obtained more slowly but obviously compared with less severe ones. The ones shown in the medium traffic scenarios are all training results of first iteration. No matter in which traffic conditions, results vary only in a small range after nearly four or five iterations.

VI. Conclusion

In this paper, TSC problem is outlined. Traffic signal control has been regarded as one of the most important traffic control methods. A new reinforcement training based online learning traffic signal controller is designed. A feed-forward neural network is adopted to accomplish the traffic signal controller. Scenarios of simulation are conducted under TSIS to verify adaptability of this method. Results of MOEs demonstrate several good features traditional methods do not have. Especially, this controller could learn through iterations with environment.

We want to enhance this method in some aspects, for example definition of the reward and a more flexible phase sequences. In order to verify the effectiveness of this method in real applications, it needs to be tested under more complex and various traffic conditions. In the future a coordinated version of traffic signal control system for large network based on this simple but effective method which has the ability to improve performance by online learning is expected.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the pertinent comments from anonymous reviewers.

References

[1] http://ostpxweb.ost.dot.gov/policy/reports/Costs of Surface Transportation Congestion.pdf, accessed in Apr. 2011.

[2] A. Engelbrecht, *Computational Intelligence: An Introduction*, 2nd edition, NY, USA: John Wiley & Sons, 2007.

[3] J. C. Spall, D. C. Chin, "Traffic-responsive signal timing for systemwide traffic control," *American Control Conference*, vol. 4, pp. 2462–2463, 1997.

[4] M. C. Choy, D. Srinivasan, and R. L. Cheu, "Neural networks for continuous online learning and control,"*IEEE Transactions on Neural Networks*, vol. 17, no. 6, pp. 1511–1531, 2006.

[5] D. Srinivasan, M. C. Choy, and R. L. Cheu, "Neural networks for real-time traffic signal control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 3, pp. 261–272, 2006.

[6] A. Azzam ul, U. M. Sadeeq, J. Ahmed, and H. Riaz ul, "Traffic responsive signal timing plan generation based on neural network," in *IEEE International Conference on Automation Science and Engineering (CASE),* pp. 833–838, 2008.

[7] L. K. Jones, "Constructive approximations for neural networks by sigmoidal functions," *Proceedings of the IEEE*, vol. 78, no. 10, pp. 1586–1589, 1990.

[8] E. K. Blum and L. K. Li, "Approximation theory and feedforward networks," *Neural Networks*, vol. 4, no. 4, pp. 511–515, 1991.

[9] C. W. Lee, "Training feedforward neural networks: An algorithm giving improved generalization," *Neural Networks*, vol. 10, no. 1, pp. 61–68, 1997.

[10] G. D. Magoulas, M. N. Vrahatis, and G. S. Androulakis, "Effective backpropagation training with variable stepsize," *Neural Networks*, vol. 10, no. 1, pp. 69–82, 1997.

[11] J. Feldman and E. Feigenbaum, *Computers and Thought*, McGraw-Hill, 1963.

[12] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction.* Massachusetts London, England: The MIT Press Cambridge, 1998.

[13] Y. J. Dai and D. B. Zhao, "A traffic signal control algorithm for isolated intersections based on adaptive dynamic programming," in *International Conference on Networking, Sensing and Control (ICNSC)*, pp. 255–260, 2010.

[14] Y. J. Dai and D. B. Zhao, and J. Q. Yi, "A comparative study of urban traffic signal control with reinforcement learning and adaptive dynamic programming," in *International Joint Conference on Neural Networks (IJCNN),* pp. 1–7, 2010.

[15] TSIS User's Guide, Version 5.1, ITT Industries, Inc., FHWA Office of Operations Research, February, 2003.

[16] CORSIM Run-Time Extension (RTE) Developer's Guide, Version 5.1, ITT Industries, Inc., FHWA Office of Operations Research, February, 2003.

[17] CORSIM User's Guide, Version 5.1, ITT Industries, Inc., FHWA Office of Operations Research, February, 2003.