Hierarchical Fuzzy Logic Control for Multiphase Traffic Intersection Using Evolutionary Algorithms

Yi Hu, Andrew Chiou and Qinglong Han
Faculty of Business and Informatics
CQUniversity Australia
Rockhampton Campus, 4702 QLD
y.hu@cqu.edu.au, a.chiou@cqu.edu.au, q.han@cqu.edu.au

***Abstract*-In this paper, a hierarchical fuzzy logic traffic controller is constructed for a real intersection of fourteen vehicle lanes and two pedestrian crossings controlled by signals with seven light phases. The hierarchical fuzzy controller has seven inputs as queue lengths of the seven light phases, and one output as green time of the selected phase. In the hierarchical fuzzy controller, there are six layers of fuzzy sub-controllers with two inputs and one output. The sub-controllers in the first five layers have identical structure that has two inputs of queue lengths and one output of combined queue length employed as one input of next layer. The sub-controller in the last layer has two inputs, combined queue length obtained from the fifth layer and queue length of the selected phase, and one output as green time of the selected phase. Using the developed fuzzy controller, the best fuzzy rule base is obtained based on real traffic data of the intersection by employing evolutionary algorithm. The performance of this controller is simulated and compared to that of a controller that is currently employing in the intersection. The results show that the developed fuzzy controller shortens more than 38% of the vehicle waiting time.**

I. INTRODUCTION

Enormous efforts have been made to develop fuzzy traffic signal control. Traffic signal control is a complex problem that has intensive characteristics, such as randomness, burst, and uncertainty. In the past decades, considerable research has been devoted to automatic or optimal signal control using fuzzy logics [1-5]. Pappis [1] and Hong [5] used fuzzy control into the urban traffic signal control. Chiu [3] developed fuzzy decision rules to adjust cycle time, phase split and offset for signal control and tested the model using a traffic flow simulation model. Kelsey and Bisset [4] simulated an isolated north-south and east-west intersection using a fuzzy logic controller and a pre-timed controller.

For major of fuzzy systems, implicit assumptions limit the application of fuzzy logic system with a few parameters. The number of fuzzy rules is directly dependant on input parameters of fuzzy controller. In order to reflect the real traffic environment, many parameters need be considered in fuzzy logic system. However, as the number of fuzzy input parameters increase, the number of fuzzy rules of the system grows exponentially [6]. Hence, an effective fuzzy system with multiple inputs needs to be developed.

Hierarchical fuzzy controllers have been developed for traffic signal control due to its effectiveness on multiple inputs parameters. Since Raju and Zhou [6] firstly introduced hierarchical fuzzy systems, hierarchical control architectures have been built in the applications of artificial intelligent in traffic signal coordinated control in recent years. Choy and Cheu [7] present a new hybrid approach to implement a cooperative, hierarchical system for a real-time traffic signal control by dividing the large system into various sub-problems. Wei et al, [8] introduce a cellular automata model of urban traffic signal control system, while the hierarchical control architecture of traffic signal is constructed. By using the hierarchical fuzzy logic architecture, number of fuzzy rules in the system is reduced thereby reducing the computational time while maintaining the systems robustness and efficiency.

In this paper, a new hierarchical fuzzy controller is constructed for a main traffic intersection which consists of 14 vehicle lanes and 2 pedestrian crossings controller by signals with seven light-phases. The hierarchical fuzzy controller has seven inputs as queue lengths of the seven light phases, and one output as green time of the selected phase. In the hierarchical fuzzy controller, there are six layers of fuzzy sub-controllers with two inputs and one output. The sub-controllers in the first five layers have an identical structure that has two inputs of queue lengths and one output of combined queue length employed as one input of next layer. The sub-controller in the last layer has two inputs, combined queue length obtained from the fifth layer and queue length of the selected phase, and one output as green time length of the selected phase. Using the developed fuzzy controller, the best fuzzy rule base is obtained based on real traffic data of the intersection by employing evolutionary algorithm. The performance of this controller is simulated and compared to that of a controller that is currently employing in the intersection. The results show that the developed fuzzy controller shortens more than 38% of the vehicle waiting time.

II. TRAFFIC SIGNAL CONTROL

The considered intersection shown in Fig. 1 has. 2 pedestrian crossings, 14 lanes, in which 7 lanes are straight lanes, 6 lanes are turns, and one lane is both a straight lane and turn. These 14 lanes and 2 pedestrian crossings are arranged in 10 groups controlled by 10 traffic lights. TABLE I lists the lane group and corresponding lights. As lane 2 is

978-1-4244-3507-4/09/$25.00 ©2009 IEEE

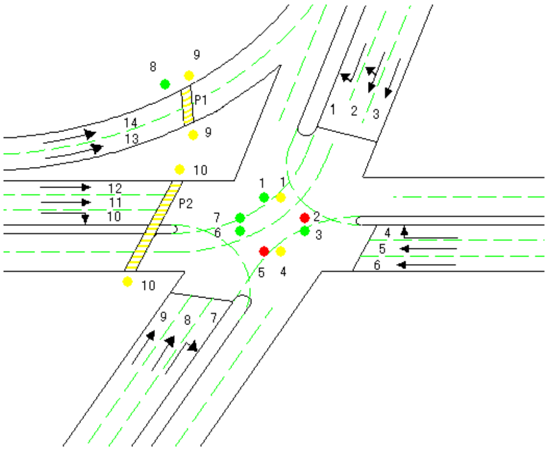

Fig. 1. Intersection layout

both a straight and turning lane, the first three lanes are controlled by light one. Lights 2, 4, 6, and 8 control turning lanes. Lights 3, 5, and 7 control straight lanes. Lights 9 and 10 control the two pedestrian crossings.

The fuzzy traffic light system is controlled in light phases. 7 light phases with maximum compatible lanes are arranged as listed in TABLE II. The 7 light phases cover all 10 sets of traffic lights. The first three phases consist only vehicle lanes and the last four phases include vehicle lanes and pedestrian crossings.

Time sequences of each light phase are arranged as late start, green, yellow, and all red as shown in Fig. 2 [9]. Late start and all red are arranged for clearing safety at the end of red light. The late start and all red time for each phase are listed in TABLE II. During later start, all lights are red except the lights that are green in both the last phase and the current phase. At the beginning of the green phase, the controller will turn on all green lights and calculate the green light time length using the fuzzy controller. Once the green light finishes, the phase that has the maximum vehicles will be selected as the next phase. If the next phase is still the current phase, the light state remains unchanged until a different phase is selected. Once a different phase from the current phase is determined, the yellow lights turn on. If a green light in the current phase is still a green light in the next phase, the green light will stay green all the way from the yellow light of the current phase to end of green light of the next phase. After 5 seconds of yellow light, all the yellow lights will turn red. After all red and late start, the green lights of the next phase will commence.

TABLE I
LINE GROUPS AND CORRESPONDING LIGHTS

Group No	Light No	Lanes	Group No	Light No	Lanes
1	1	1,2,3	6	6	10
2	2	4	7	7	11,12
3	3	5,6	8	8	13,14
4	4	7	9	9	P1
5	5	8,9	10	10	P2

TABLE II
LIGHT PHASES AND LANE GROUPS

Phase No.	Lights	Lanes	Late Start(s)	All Red(s)
1	3,7,8	5,6,11--14	2	2.5
2	6,7,8	10,11—14	2	2.5
3	1,8	1,2,3,13,14	3	2.5
4	2,3,9	4,5,6,P1	3	2.5
5	4,8,10	7,13,14,P2	0	2.5
6	4,5,9,10	2,6,P1,P2	0	3.5
7	7,8,9,10	4,10,P1	3	2.5

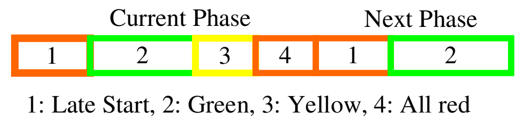

Fig. 2. Time sequences of a light phase

III. TRAFFIC AND PEDESTRIAN SIMULATION

As the best fuzzy rule base need evaluated from real statistic data using evolutionary algorithm, traffic and pedestrian simulation have to be carried out.

Vehicles facing a red signal, as well as, vehicles facing a yellow signal that cannot safely pass through the intersection, decelerate and stop according to Equation (1). This group of vehicles are simulated by the deceleration model developed in this study.

$$a(t) = \begin{cases} 0.2(v_{free} - v(t)), & \Delta x(t) > d_{sa}(t), \\ 0, & d_{sa}(t) \geq \Delta x(t) > d_{sb}(t), \\ -4.5, & \Delta x(t) \leq d_{sb}(t), \end{cases} \qquad (1)$$

in which all the distances are defined as

$$\begin{aligned} \Delta x(t) &= x_f(t) - x(t), \\ d_{sb}(t) &= 1 + \tfrac{1}{7}(v^2(t) - v_f^2(t)), \\ d_{sa}(t) &= 1.5 + 2\max(d_{sb}(t),\ 0). \end{aligned} \qquad (2)$$

In this model, movement of a vehicle is dependent on the vehicle immediately ahead. The vehicle will accelerate if the distance Δx between the two vehicles is longer than the safe acceleration distance d_{sa}; the safe acceleration distance is assumed to be twice as large as the safe brake distance or zero, which ever is the larger. The vehicle will keep its original speed if the distance is shorter than the safe acceleration distance but longer than the safe break distance d_{sb}; the vehicle will decelerate if the distance is shorter than the safe break distance. In this study, the breaking acceleration is assumed a constant value of $4.5 m/s^2$ for simplicity. After the acceleration is determined, the velocity and position of the vehicle can be calculated forward using Equations

$$v(t+\Delta t) = v(t) + a(t)\Delta t \quad (3)$$

$$x(t+\Delta t) = x(t) + v(t)\Delta t + \tfrac{1}{2}a(t)\Delta t^2 \quad (4)$$

When the vehicle velocity decreases, two adjacent vehicles become very close. The velocity of the following vehicle is mainly dependent on the velocity of the front vehicle. To mimic reality, another relation is assumed,

$$\begin{aligned} &v(t+\Delta t) = v_f(t),\ a(t) = 0, && \text{if } \Delta x \le 1.0\text{m},\ v(t) < 0.2\text{m/s}, \\ &v(t) = 0, && \text{if } v(t) < 0. \end{aligned} \quad (5)$$

Thus, all the vehicles in the queue keep a spacing of about 1 metre. In the simulation of the car flow, the front car position and velocity of the leading car is the position of the stop line and zero.

Vehicles in the front of green lights and vehicles, which can safely pass the stop line when the yellow light is on, will accelerate to speed-limit of the road. This group of vehicles are simulated using the acceleration model developed in this study. The accelerating vehicles can be divided into two groups according to their movement state. The first group is the front waiting vehicles characterized by $v(t) \le v_f(t)$. This group of vehicles accelerates one after another. The second group of vehicles characterized by $v(t) > v_f(t)$ is the later approaching vehicles coming forward with high speed behind the first group of vehicles.

Movement of the first group of vehicles is simulated by:

$$a(t) = \begin{cases} 0, & \Delta x(t) < 3.7\text{m and } v(t) < 2 \text{ m/s}, \\ 0.2(v_{free} - v(t)) - \exp(12.4 - 22.4\Delta x(t)/d_{sca}(t)), & \text{other wise} \end{cases} \quad (6)$$

Where the safe clearance d_{sca} is determined from testing data as:

$$d_{sca}(t) = 3.7 + 0.1v(t) \quad (7)$$

For the waiting or low speed cars, a safe clearance distance is needed to start accelerating. The acceleration has two terms for each case. The acceleration in Equation (6) has two parts. The second part is a deceleration term to maintain safe clearance from the front car. When the distance between two adjacent cars is equal to or greater than the safe clearance, $d_{sca}(t)$, the acceleration term is close to zero. When the distance is smaller than $d_{sca}(t)$, the term decreases exponentially as the distance decreases, and reaches $-3.2\,\text{ms}^{-2}$ when the distance is half of $d_{sca}(t)$. Thus the second part of acceleration takes effect only when the two cars are too close. The first term is acceleration to increase the speed of the car. After the acceleration of the vehicle is determined, its velocity and location is calculated using Equation (3) and (4).

The second group of vehicles are simulated by the deceleration model until $v(t) \le v_f(t)$ condition is met.

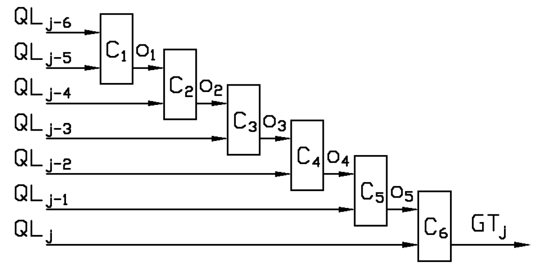

Fig. 3. Structure of hierarchical fuzzy controller

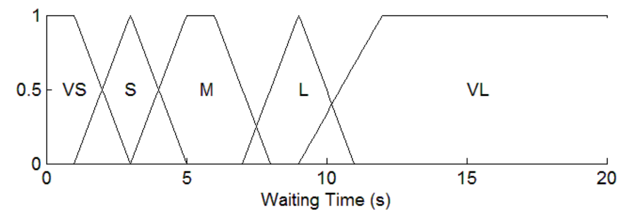

Fig. 4. Input and output membership functions

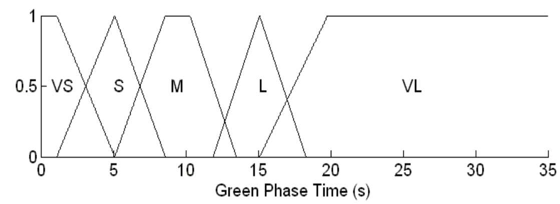

Fig. 5. Output membership functions for sub-controller C

As speed of vehicles near the intersection is below 60 km/h, it was assumed that radii of curves have no effect on vehicle movements.

Pedestrian is simulated by constant speed movement. When the pedestrian green lights are on, pedestrian will move at a constant speed of 3m/s. when pedestrian red light is on, or there is no enough time left to pass when the yellow lights turn on, the pedestrian will stop and wait in the front of lights.

IV. HIERARCHICAL FUZZY LOGIC CONTROLLER

A hierarchical fuzzy logic traffic controller is constructed to determine green time of selected phase. Traffic lanes are controlled in light phase. When green light of the current phase finishes, the phases with maximum vehicles will be selected as the next phase. The fuzzy controller will calculate green time for the selected phase when its green lights turn on.

As shown in Fig. 3, the fuzzy controller has seven inputs as queue length of the seven phases, and one output as the green time of the selected phase. The queue length of each phase is defined as the maximum queue length among vehicle lanes in the light phase. In the hierarchical fuzzy controller, there are six layers of fuzzy sub-controllers, C1 to C6. The output of last layer servers as one input of the next layer. The first five sub-controllers have an identical structure, which has two inputs as queue lengths, and one output as combined queue length. Sub-controller C6 has two inputs as queue lengths and one output of green time. Queue length of the selected phase,

QL_j, is employed as one input of the sub-controller C6, as it has direct effect on the green time of the phase, GT_j. The rest queue lengths of other phases are used as inputs of sub-controllers of up levels as shown in Fig. 3.

All inputs of the hierarchical fuzzy controller and outputs of sub-controllers C1 to C5 are set five linguistic values: very short, short, medium, long, and very long, whose membership functions are shown in Fig. 4. When vehicle number is more than 12, the queue is considered very high. The output of sub-controller C6 is also set 5 linguistic values, very short, short, medium, long, and very long, whose membership functions are shown in Fig. 5. From the road test and simulation [10], the time for 20 waiting vehicles passing through a stop line is 34.7 seconds. So, the maximum value of output variable is set to 35 seconds. Each sub-controller has 25 rules, thus there are total 150 rules for the hierarchical controller.

If the current light phase has P1 calls, the minimum green light duration is 6 seconds, if it hasP2, or P1 and P2 calls, the minimum green light duration is 10 seconds.

The fuzzy rule base for the hierarchical fuzzy controller is evaluated as the best rule base found by an evolutionary algorithm using test data.

V. Fitness Function and Evolutionary Algorithm

To evaluate the best fuzzy rule base, a fitness function is defined as the following to characterize performance of the fuzzy controller:

$$Fitness = T_{mw} = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=0}^{T}\Delta T_w^{ij} \quad . \tag{8}$$

It is the averaged waiting time over the whole simulation time T and all vehicles. The waiting time is defined as:

$$\Delta T_w = \begin{cases} \frac{v_{\text{limit}} - v(t)}{v_{\text{limit}}}\Delta t, & v(t) < 0.8 v_{\text{limit}}, \\ 0, & v(t) \geq 0.8 v_{\text{limit}}, \end{cases} \tag{9}$$

in which both the geometry delay and queuing delay of the intersection are considered.

In using the evolutionary algorithm, it is necessary to encode the fuzzy rule base as an individual in the population. By numbering the output fuzzy sets, very short, short, medium, long, and very long by integers 1, 2, 3, 4 and 5 respectively, a rule base can be encoded as a decimal number string with 150 bits.

The evolutionary algorithm employed in this paper is as follows: Given two initial individuals whose corresponding entries in the rule bases are generated randomly, one point crossover is used to generate 500 individuals. Then 4% of the individuals are randomly chosen to take mutation at one random point. At the mutation point, the corresponding element in the string is set to a random integer in the range 1 to 5. During the traffic simulation, the fitness of each individual is calculated. Based on the fitness, two new individuals are selected to generate the new population by crossover and mutation using the same procedure that is used to generate the initial population. These two individuals are chosen to be the best individual and the next best individual based on ranking of the fitness function. After repeating the procedure for 50 generations, the individual with the best fitness value is obtained as the best rule base.

VI. Simulation Results

Traffic simulations are carried out using two light controllers. The first is the pre-cycle controller that is currently being used by the Department of Main Roads at the intersection. The second is the fuzzy controller constructed above. The first controller has four phases (A, B, C, and D) as listed in TABLE III. The sequence is fixed as A, B, C and D. Phase C has three different choices depending on certain conditions. The green light duration for each phase is fixed at 20 seconds. The pre-cycle controller is non-responsive to real-time fluctuations in traffic demand. This controller can not select next phase according to the traffic volume like fuzzy controller because of the fixed phase sequence.

The real statistic vehicle data of the intersection is provided by the Department of Main Roads. The total traffic volume is shown in Fig. 6. The traffic has two peak hour periods at 9am and 5pm and trough hour at 4 am. The maximum and minimum traffic density is about 3000 and 100 vehicles/hour.

The time average queue lengths over the 14 lanes of the intersection are shown in Fig. 7. The time variations in queue length for the two controllers have the same shape as that of the traffic volume shown in Fig. 6. When traffic density is low, the queue lengths are shorter. When traffic density is high, the queue lengths are longer. The average queue length of the fuzzy controller is much shorter than that of the pre-cycle controller. At the peak hour of 9 am, these queue lengths for the two controllers are 1.5 and 2.5, respectively.

The average waiting time for each hour are shown in Fig. 8. The waiting time of the fuzzy controller is shorter than that of the pre-cycle controller through the whole day. At peak hours, the average waiting time of fuzzy controller are 21 and 23 seconds, which shortens 45%, 38% of that for the pre-cycle controller (38 and 37 seconds). During non-peak hours, the fuzzy controller has further better performance. For example, the waiting time at 1am is 6 second, which is four times shorter than that of the pre-cycle controller (32 seconds).

The average pedestrian waiting time from the two controllers are shown in Fig. 9. Similar to the results found from vehicle waiting time, the pedestrian waiting time of fuzzy controller is shorter than that of the pre-cycle During peak hours, the average waiting time of fuzzy controller is around half of that of the pre-cycle controller.

TABLE III
FOUR PRE-CYCLE LIGHT PHASES

Phase	Light No.	Lane
A	1, 8	1,2,3,13,14
B	4, 5	7,8,9
C0	2, 6	4,10
C1	2, 3	4,5,6
C2	6, 7, 8	10—14
D	7,8	11--14

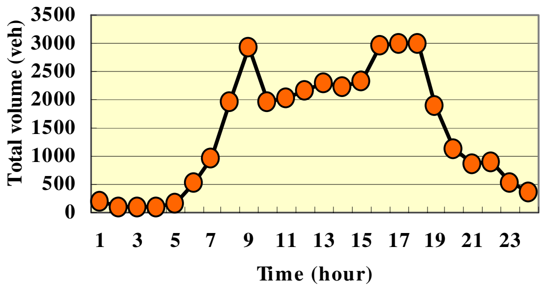

Fig. 6. Traffic volume against time

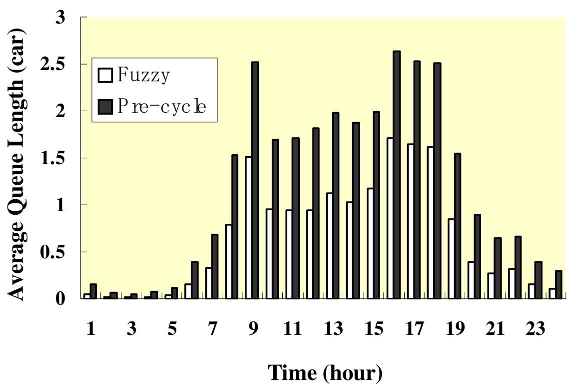

Fig. 7. Average queues for two controllers

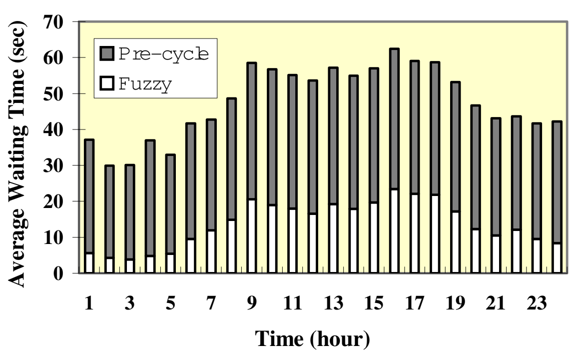

Fig. 8. Average waiting time for two controllers

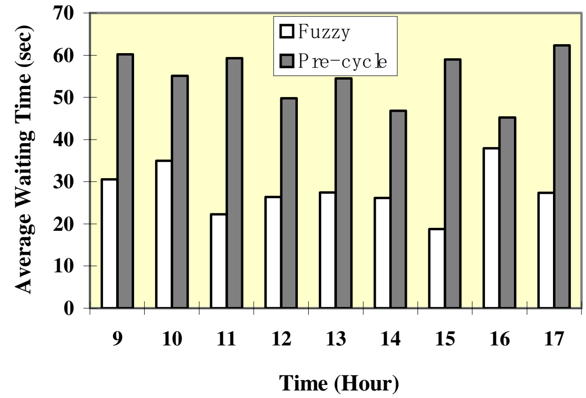

Fig. 9. Average pedestrian waiting time for two controllers

VII. CONCLUSIONS

In this paper, a hierarchical fuzzy traffic controller is constructed for a real intersection. Traffic simulation is carried out using vehicle acceleration and deceleration movement models developed in this study. The best fuzzy rule base is obtained using evolutionary algorithm from real statistic data of the intersection.

From simulation results, it is found that the developed fuzzy controller had better performance than the pre-cycle controller which is currently using in the intersection. It produces shorter average queue length than that of pre-cycle controller, and shortens more than 38% of vehicle average waiting time of the pre-cycle controller.

REFERENCES

[1] G. Eason, B. Noble, and I.N. Sn C. Pappis and E. Mamdani, "A fuzzy logic controller for a traffic junction," IEEE Trans. Syst., Man, Cybern., vol. 7, no. 10, pp. 707–717, 1997.

[2] S. Chiu, and S. Chand. "Self-organising traffic control via fuzzy logic,". In Proceedings of the 32nd conference on decision and contro,. pp 1897-1902, Dec 1993.

[3] M. Nakatsuyama, H. Nagahashi, and N. Nishizura. "Fuzzy logic controller for a traffic junction in a one way arterial road," *9th IFAC-World Congress,* pp 13-18, Budapest, Hungary, 1994.

[4] K. Bisset and R. Kelsey. "A Simulation of Environment For Fuzzy Control of Traffic Systems," 12th IFAC- World Congress, Vol 5, pp 553-556, Sydney, Australia, 18-23 July 1993.

[5] W. Hong and X-q. Mu. "A cooperative fuzzy control method for traffic lights," J. Syst. Simul., vol. 13, no. 5, 2001., pp 551–553, Chinese.

[6] G. Raju and J. Zhou. "Adaptive Hierarchical Fuzzy Controller,". IEEE Transaction on systems, Man, and Cybernetics, vol 23, no. 4, pp 973-980, August 1993.

[7] M. Choy and R. Cheu. "Cooperative, Hybrid Agent Architecture for Real-time Traffic Signal Control," *IEEE Transaction on systems, Man, and Cybernetics—Part A: System and Humans*, vol. 33, no. 5, pp 597-608, September 2003.

[8] J. Wei, A. Wang, and N. Du. "Study of Self-Organizing Control of Traffic Signals in an Urban Network Based on Cellular Automata," *IEEE Transactions on Vehicular Technology,* vol. 54, no. 2, pp 744-748. March, 2005.

[9] AUSTROADS, *Guide to Traffic Engineering Part 7: Traffic Signals* (3rd Edition). Association of Australian State road and Transport Authorities, Sydney, Australia. ISBN 0855886226 (2003).

[10] J.H Lee, and H. Lee-Kwang. “Distributed and Cooperative fuzzy Controllers for Traffic Intersections Group”, *IEEE Trans. Syst. Man, and Cybern. PartC: Applications and Reviews*, Vol.29, 1999.