

# Intelligent traffic signal controller based on type-2 fuzzy logic and NSGAI

Chen Wen\*, Zhao Hui, Li Tao and Liu Yuling

*College of Computer Science, Sichuan University, Chengdu, China*

**Abstract.** Intelligent traffic signal control (TSC) system is important for the alleviation of traffic congestion. Usually, most of the researches about TSC focused on single intersection based on type-1 fuzzy set. Compared with type-1 fuzzy logic controller (FLC), type-2 FLC can deal with more uncertainties in the road traffic control system. Therefore, a type-2 FLC optimized by NSGAI (T2-NSGAI) is designed for TSC in a complex road network, in which the intersection's traffic signal time is dynamically adjusted by its own and adjacent intersections' traffic volumes to reduce global delay time and traffic congestion. In T2-NSGAI, the expert rule set and the parameters of the fuzzy membership functions are simultaneously optimized by NSGAI to achieve less time delay and traffic congestion. In the simulations of a six-intersection traffic network with different vehicular arrival rates, it is demonstrated that T2-NSGAI has better performance compared with vehicle actuated controller based on fixed-time control (FTC), type-1 FLC, type-2 FLC and isolatedly optimized Type-2 FLC and the total delay time could be reduced by 76.3%, 65.1%, 58.3% and 35.4% respectively.

**Keywords:** Type-2 logical fuzzy, traffic signal control, NSGAI, phase sequence, multi objects optimization

## 1. Introduction

OVER the past few years, there has been a steady growth of vehicles and total miles of road leading to an enormous increase in traffic congestion. Given that major road construction in cities are both difficult and costly, one of the most effective ways to alleviate congestion is to increase the capacity of the existing infrastructure using Traffic signal control (TSC). TSC can bring a substantial reduction to traffic congestion, leading to improved conditions both for the drivers and the environment [12].

TSC in the signalized traffic intersections can be either fixed-time signal control or adaptive control [5]. Generally, the fixed-time control is simple and the time of the signal light is fixed, therefore, it can not dynamically response to real-time fluctuations of traffic

volumes. To overcome the limitations of the fixed-time control that using fixed time intervals, adaptive TSC was designed based on the real-time traffic information to make quickly adjustments to signal settings [3].

Nevertheless, controlling the traffic signals of a transportation network constitutes a significant challenge due to the complexity of the problem, especially the uncertain and dynamic behavior of the network [3]. The TSC system has many uncertainties, i.e., the traffic queue length, collected by the auto loop detectors, is non-stationary, and it varies all the time. The traffic variations are impacted by many factors, including the length of cars, the road condition, the weather, the traffic rules, etc. Therefore, new intellectual approaches are needed to deal with the complexity and uncertainty of the TSC system [10].

Fortunately, Fuzzy logic control (FLC) provides such an intellectual approach for the TSC system, which is capable of mimicking human thought and holding the expert knowledge in the form of inference rules [1, 11]. It is well known that, Type-2 fuzzy

\*Corresponding author. Chen Wen, College of Computer Science, Sichuan University, No. 24 South Section 1 Yihuan Road, Chengdu 610065, China. Tel.: +86 28 8540 4936; Fax: +86 28 8540 5568; E-mail: wenchen@scu.edu.cn.

sets can efficiently deal with large uncertainties based on its three-dimensional membership. Compared with Type-1 FLC, Type-2 FLC has better performance, but the performance is largely affected by the quality of the fuzzy rules and it is extremely difficult to select the optimal parameters for the membership functions of Type-2 FLC [7]. As an efficient and powerful stochastic parameter optimization method for multiple objects, Non-dominated Sorting-based Genetic Algorithm (NSGA-II) has shown better performance than traditional genetic algorithm (GA) and particle swarm optimization (PSO) [6]. Furthermore, NSGAI is designed for the problems of multiple objects optimization [4], which makes it more suitable for the parameter optimization problem of type-2 fuzzy logic control system.

In the paper, we proposed a type-2 fuzzy logic controller, optimized by NSGAI, for the real-time light control in the four phase intersections with turning vehicles taken into consideration. To achieve high vehicle throughput rate, both the expert rule set and the parameters of the membership functions were simultaneously optimized by NSGAI. Then the fuzzy controller was applied in a multi intersections traffic network, in which the light time was dynamically adjusted by the traffic volumes of each intersection and its adjacent neighbors.

## 2. Simulation of the traffic

In the paper, as shown in Fig. 1, the common four phase traffic situation is considered in our work, including: W-E straight phase, W-E left turning phase, N-S straight phase and N-S left turning phase. Therefore, each intersection has four-phase and they alternately emergent, which is similar with the widely used intersections in real traffic network. Assume that  $l$  and  $f$  represent the distance between two adjacent intersections and the length of one vehicle respectively, and the average speed of vehicles is  $v$  km/h, then there are at most  $l/f$  vehicles in each lane. As shown in Equation (1), the time cost of vehicles to reach the next intersection consists of three parts: the intersection passing time  $l/v$ , the delay time for red signal  $t_g$  and the delay time for congestion  $t_c$ . Apparently, the length of the signal time should be dynamically adjusted based on the average moving speed of vehicles and the neighbor intersections' queue length to reduce the delay time while raise the throughput rate. There are many sensing technologies to detect vehicles' speed data at intersec-

tions. Accordingly, the number of vehicles in each lane can be estimated based on the sensed speed.

$$D = l/v + t_g + t_c \quad (1)$$

## 3. Intelligent signal control based on type-2 FLC

### 3.1. Input data

To design a FLC system, we need to choose the input and output variables firstly. As described in Fig. 1, there are four lane directions in each intersection: E-W straight, E-W left-turning, S-N straight, S-N left-turning. Apparently, the vehicles in the direction, which has more traffic volume, have high privilege to pass. And the traffic state of the straight or left-turning adjacent intersections should also be taken into consideration. Therefore, at the end of every green light time, in each intersection the queue length of the 4 directions and the traffic state of the target intersection are chosen to be two input variables to the fuzzy controller.

#### 3.1.1. The queue length

The queue length, throughput rate, average delay time and parking times are commonly used criteria to evaluate the traffic conditions. According to [13], we suppose that the minimum successive time unit is 1s, and the vehicles' arrival time in each direction is

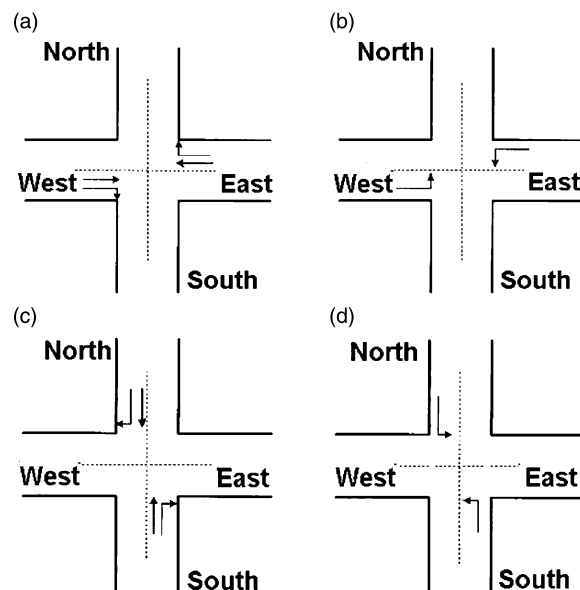


Fig. 1. The four phase of an intersection.

random, thus, in a time unit interval, the number of arrived vehicles  $q_n$  is shown in Equation (2)

$$q_i = \begin{cases} 1, & \text{if a vehicle arrives in the } i\text{th unit time} \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Let  $lq$  and  $rq$  represent the vehicle queue lengths during green light time and red light time respectively, the queue length is:

$$lq(i) = \begin{cases} \max(0, lq(i-1) + q_i - s), & \text{if } i > 1 \\ 0, & \text{if } i = 1 \end{cases} \quad (3)$$

$$rq(i) = \begin{cases} rq(i-1) + q_i, & \text{if } i > 1 \\ q_1, & \text{if } i = 1 \end{cases} \quad (4)$$

where  $i$  indicates the  $i$ -th simulation time unit, while  $s$  denotes the number of departure vehicle in one time unit during green light time.

### 3.1.2. The traffic state

As defined in Equations (5) and (6), the traffic state of the  $i$ -th intersection  $s_i$  is denoted by the flow rate  $fr_i$ , and  $fr_i$  in each direction is calculated by the arriving rate of upstream minus downstream departure flow rates for each lane in that direction of the phase. Sampling of the vehicle data is done at a pre-given time interval, which can be about 10s (Minimum green time):

$$ts_i = \frac{fr_i}{\max\{fr_i, fr_j\}}, \quad s_j \in n_i \quad (5)$$

where  $n_i$  is the intersection neighbors of  $s_i$ , and  $fr_i$  is defined as:

$$fr_i = \frac{\sum_{k=1}^{c_g/\nabla T} \sum_{i=1}^n \sum_{j=1}^l (f_{ij}^{up} - f_{ij}^{down})}{n \cdot l \cdot (Cg/\nabla T)} \quad (6)$$

where  $C_g$  represent the current green time,  $n$  is the number of links having right of way,  $l$  is the number of lanes in the links,  $\nabla T$  is the sampling period and  $f_{ij}^{up}$ ,  $f_{ij}^{down}$  is the count of vehicle in the upstream and downstream directions.

### 3.2. The fuzzifier

The fuzzifier is utilized to map a numeric vector  $x=(x_1, \dots, x_p) \in X_1 \times X_2 \times \dots \times X_p \equiv X$  into a type-2 fuzzy set  $\tilde{A}_x$ , which is an interval type-2 fuzzy set in this case. In the paper, type-2 singleton fuzzifier is employed. In a singleton fuzzification, there is only a single point on nonzero membership in the input fuzzy set.  $\tilde{A}_x$  is a type-2 fuzzy singleton if  $\mu_{\tilde{A}_x}(x) = 1/1$  for  $x = x'$  and  $\mu_{\tilde{A}_x}(x) = 1/0$  for  $x \neq x'$ .

In the paper, to fuzzify the input data and output result: queue length, traffic state and green time, four linguistic partitions are adopted: "Sparse (S)", "Low (L)", "Medium (M)" and "High (H)". Interval type-2 uncertain standard deviation Gaussian membership functions are employed, as illustrated in Fig. 2. The basic domain of green time is denoted as  $[0, 60]$ , while the domain of queue length and neighbor status are  $[0, 15]$ ,  $[0, 100]$  respectively. The upper and lower boundary standard deviation can be calculated based on the recorded flow rates and queues at different period of time on different days in a week.

Usually, the rule base of FLC is expressed in the "if-then" form, which holds the expert knowledge. The structure of fuzzy rules in a type-1 FLC and a type-2 FLC is in the same manner, but in the type-2 FLC the antecedents and the consequents are denoted by type-2 fuzzy sets. So assume that for a type-2 FLC with  $p$  inputs  $x_1 \in X_1 \dots x_p \in X_p$  and one output  $y \in Y$ , there are  $M$  rules, and the  $l$ -th rule in the type-2 FLS can be

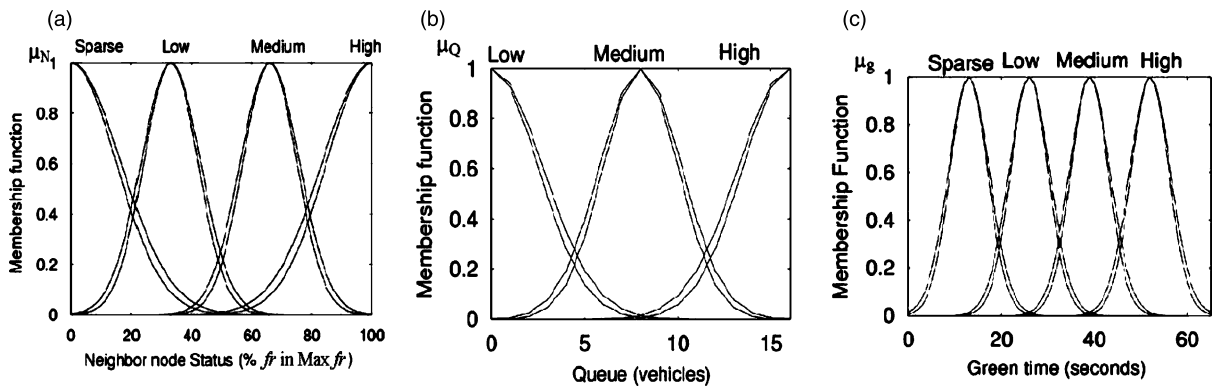


Fig. 2. Fuzzified antecedents and consequents. (a) Fuzzified neighbor status, neighbor status, (b) Fuzzified queue length, (c) Fuzzified green time.

Table 1  
Preliminary fuzzy rule base

Queue length	Neighbor target intersection state			
	S	L	M	H
L	H	H	M	M
M	H	M	M	L
H	M	M	L	L

written as:

$$R^l : \text{if } x_1 \text{ is } \tilde{F}_1^l \text{ and } x_2 \text{ is } \tilde{F}_2^l, \dots, x_p \text{ is } \tilde{F}_p^l, \quad (7)$$

$$\text{then } y^l \text{ is } \tilde{G}^l, \quad l = 1, \dots, M$$

$\tilde{F}^l$  and  $\tilde{G}^l$  represent the fuzzy linguistic partition of the input and output variable respectively. Furthermore, the corresponding membership functions (MF) of the  $i$ -th input variable and output variable are denoted by  $\mu_{\tilde{F}_i^l}(x_i)$  and  $\mu_{\tilde{G}^l}(x_i)$ .

According to common sense and the daily experience, the preliminary rule base is shown in Table 1. The rule base is optimized and validated based on a group of traffic network consisting of interconnected intersections.

### 3.3. Inference engine

The inference engine of the type-2 FLS is responsible for combining rules and mapping the input fuzzy sets to the output fuzzy sets. It is necessary to compute the join (unions) and the meet (intersections), as well as the extended sup-star compositions of type-2 relations. If  $F_1^l \times \dots \times F_p^l = A^l$ , the fuzzy rules can be re-written as follows

$$R^l : \tilde{F}_1^l \times \dots \times \tilde{F}_p^l \rightarrow \tilde{G}^l = \tilde{A}^l \rightarrow \tilde{G}^l, \quad l = 1, \dots, M \quad (8)$$

The inference result of the  $l$ -th fired rule  $R^l$  is described by the membership function  $\mu_{R^l}(x,y) = \mu_{R^l}(x_1, \dots, x_p, y)$  where  $\mu_{R^l}(x,y) = \mu_{\tilde{A}^l \rightarrow \tilde{G}^l}(x,y)$  can be written as

$$\begin{aligned} \mu_{R^l}(x, y) &= \mu_{\tilde{A}^l \rightarrow \tilde{G}^l}(x, y) \\ &= \mu_{\tilde{F}_1^l}(x_1) \cap \dots \cap \mu_{\tilde{F}_p^l}(x_p) \cap \mu_{\tilde{G}^l}(y) \quad (9) \\ &= [\cap_{i=1}^p \mu_{\tilde{F}_i^l}(x_i)] \cap \mu_{\tilde{G}^l}(y) \end{aligned}$$

In general, the  $p$ -dimensional input to  $R^l$  is given by the type-2 fuzzy set  $\tilde{A}_x$  whose membership function becomes:

$$\mu_{\tilde{A}_x} = \mu_{\tilde{x}_1}(x_1) \cap \dots \cap \mu_{\tilde{x}_p}(x_p) = \cap_{i=1}^p \mu_{\tilde{x}_i}(x_i) \quad (10)$$

where  $\tilde{x}_i$  ( $i=1 \dots p$ ) are the labels of the fuzzy sets describing the inputs. Each rule  $R^l$  determines a type-2 fuzzy set  $\tilde{x}^l = \tilde{A}_x \circ R^l$  such that:

$$\begin{aligned} \mu_{\tilde{B}^l}(y) &= \mu_{\tilde{A}_x \circ R^l} \\ &= \cup_{x \in X} [\mu_{\tilde{A}_x}(x) \cap \mu_{R^l}(x,y)], \end{aligned} \quad (11)$$

where  $y \in Y$ ,  $l = 1 \dots M$ . This dependency is the input/output relation, which holds between the type-2 fuzzy set that activates a certain rule in the inference engine and the type-2 fuzzy set at the output of that engine. In the FLS, interval type-2 fuzzy sets and intersection under product  $t$ -norm are employed, so the inference of antecedents can be addressed as

$$\bar{f}^l = [\bar{\mu}_{\tilde{x}_1^l}(x_1) \cap \bar{\mu}_{\tilde{F}_1^l}(x_1)] \cap \dots \cap [\bar{\mu}_{\tilde{x}_p^l}(x_p) \cap \bar{\mu}_{\tilde{F}_p^l}(x_p)] \quad (12)$$

$$\underline{f}^l = [\underline{\mu}_{\tilde{x}_1^l}(x_1) \cap \underline{\mu}_{\tilde{F}_1^l}(x_1)] \cap \dots \cap [\underline{\mu}_{\tilde{x}_p^l}(x_p) \cap \underline{\mu}_{\tilde{F}_p^l}(x_p)] \quad (13)$$

Then the inference result of the  $l$ -th triggered rule is simplified and shown in Equation (14):

$$\mu_{\tilde{B}^l}(y) = \mu_{\tilde{G}^l}(y) \cap F^l, \quad F^l = [\bar{f}^l, \underline{f}^l] \quad (14)$$

Therefore, the final inference result of the all triggered rules in the fuzzy rule base is shown in Equation (15).

$$\mu_{\tilde{B}}(y) = \cup_{i=1}^M \mu_{\tilde{B}^i}(y) \quad (15)$$

### 3.4. Type-reducer and defuzzifier

After the fuzzy inference process, the output from an interval type-2 fuzzy set is acquired. Type-reducer is necessary to convert interval type-2 fuzzy set into type-1 fuzzy set output, which is then converted in a numeric output through running the defuzzifier. For the case of our FLS center of sets (cos) type reduction is employed,  $Y_{cos}$ , which is expressed as

$$\begin{aligned} Y_{cos}(x) &= [y_l, y_r] \\ &= \int_{y^l} \dots \int_{y^M} \int_{f^l} \dots \int_{f^M} \frac{1 / \sum_{i=1}^M f^i y^i}{\sum_{i=1}^M f^i} \quad (16) \end{aligned}$$

In Equation (16),  $Y_{cos}$  represent interval set calculated by  $y_l$  and  $y_r$ ,  $y_i = [y_l, y_r]$ , which is the centroid of the result set  $\tilde{G}^l$ .

$$y_l = \sum_{i=1}^M f_l^i y^i / (\sum_{i=1}^M f_l^i),$$

$$y_r = \sum_{i=1}^M f_r^i y^i / (\sum_{i=1}^M f_r^i)$$
(17)

The output interval of the type-2 FLC system can be calculated by the values of  $y_l$  and  $y_r$ . After type-reduction, we get an interval set  $Y_{cos}$ , which is applied to defuzzifier, and the crisp output is obtained as:

$$f(x) = (y_l + y_r)/2$$
(18)

#### 4. Optimization process using NSGAI

In the paper, NSGAI is employed to optimize the fuzzy rule and parameters of the type-2 FLC system. NSGAI is a high efficient and powerful random search algorithm for solving multiple objects optimization problems. NSGAI was first introduced by K. Deb [4]. For NSGAI, The optimal solutions lie on the non-dominated front (Pareto optimal front) in the feature space of multiple objectives.

##### 4.1. Introduction of NSGAI

Usually, the multi-objective optimization problem can be expressed as how to choice the correct mapping of a set of  $n$  parameters to a set of  $m$  objectives. It can be described algebraically in Equation (19):

$$\begin{cases} \text{Find } x = \langle x_1, x_2 \dots x_n \rangle \\ \text{Minimum } y = \langle f_1(x), f_2(x) \dots f_m(x) \rangle \\ \text{Subject to } l_i \leq x_i \leq u_i, 1 \leq i \leq m \end{cases}$$
(19)

In Equation (19),  $x$  represents the set of decision variables, and  $y$  denotes the objective set. For an evolutionary algorithm with multiple objectives,  $f_i(x)$  is the fitness function for the  $i$ -th objective. Specifically, if any object function  $f_i(x)$  is originally designed to find a maximum value, and then it can be modified as  $-f_i(x)$ . Just as other evolutionary algorithms, NSGAI is also based on the idea of evolution of populations of possible candidate individuals, which represent possible solutions, and the evolution process includes mutation, crossover and selection. Furthermore, NSGAI introduced the concept of pareto dominate  $<$  to classify the individuals into different pareto front based on the

multi-objects values, accordingly the pareto domination is defined as:

If  $x_A, x_B$  are two possible solutions, if  $x_A < x_B$ , then:

$$\begin{cases} \forall i = 1 \dots m, f_i(x_A) \leq f_i(x_B) \wedge \\ \exists j = 1 \dots m, f_j(x_A) < f_j(x_B) \end{cases}$$
(20)

The solution  $x_A < x_B$  means for each object function,  $x_A$  is at least as good as  $x_B$ , and at least for one object,  $x_A$  is better than  $x_B$ . When  $x_A$  and  $x_B$  are not pareto dominate each other, both solutions belong to the same non-dominated front, which means if  $x_A$  is better than  $x_B$  in some objectives, then it must be worse at least in one other objective.

When a possible solution  $s$  in the last generation is not dominated by any other feasible solutions,  $s$  is referred to as a pareto optimal solution of the  $m$ -objective minimization problem in Equation (19). The set of all pareto optimal solutions forms the tradeoff surface (pareto front) in the objective space.

##### 4.2. Encoding of the parameters and rules

In the paper, the fuzzy rule set and the MF parameters are simultaneously optimized by NSGAI. For the fuzzy inference rules, as shown in Table 1, there are 12 rules. Therefore, 12 codes are defined to denote the outputs of the fuzzy rules.

For example, the first four codes represent the output of the first row in Table 1, the linguistic Sparse (S), Low (L), Medium (M) and High (H) are denoted by numeric value 1,2,3 and 4 respectively. And the rules in Table 1 is expressed by chromosome, it equals [4,4,3,3;4,3,3,2;3,3,2,2].

In the rule base, as shown in Fig. 3, each linguistic degree in the Gaussian membership function of the queue length is decided by 7 parameters:  $\sigma_{L1}, \sigma_{L2}, \sigma_{M1}, \sigma_{M2}, \sigma_{H1}, \sigma_{H2}$  and the mean value  $m$ . Consider the other input variable neighbor traffic state and the output variable green time, there are 31 parameters in all for the optimization of membership function. Finally, each individual of NSGAI is encoded as shown in Fig. 4.

##### 4.3. The multi object fitness function

The delay of multiple intersections is taken as the fitness function. Let  $D$  denotes the total delay in the last sampling period of an intersection A. And the number of the arrived vehicles is counted by the loop detectors. Suppose  $n$  represents the total number of the arrived

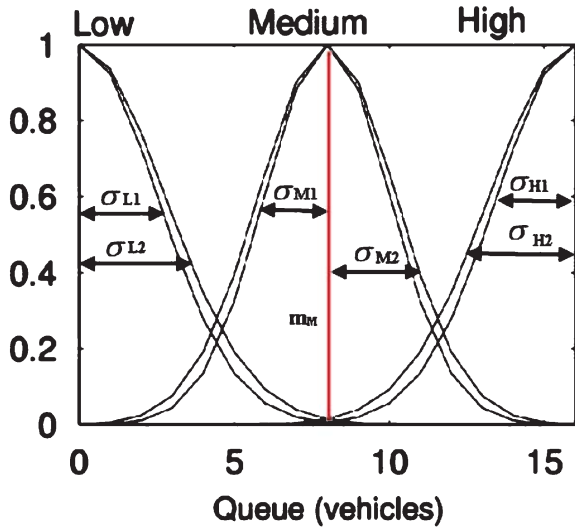


Fig. 3. Coding for the MF parameters.

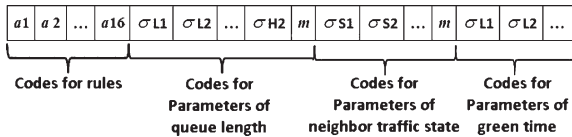


Fig. 4. Coding for the MF parameters.

vehicles at A. And the average delay of A is  $D/n$ . This value is an absolute delay because  $D$  only related to the number of the arrived vehicles. The average delay of other intersections can be calculated in a similar manner. Therefore, the multiple optimization objects of NSGAI is

$$f = \{f_1, f_2 \dots f_m\}, f_i = D/n, i = 1 \dots m \quad (21)$$

During the evolution of NSGAI, the individuals of each generation is evaluated by the multiple fitness function defined in Equation (21), furthermore, to keep the diversity of the individuals, the crowding distance is also taken into consideration when we choice individuals into the next generation. The detail of NSGAI is described in [4].

### 5. Simulation results

To evaluate the performance of the fuzzy controllers, type-2 fuzzy logic control optimized by NSGAI (T2-NSGAI), a network of six intersections is de-signed. All intersections are 4-way with four phases, as shown

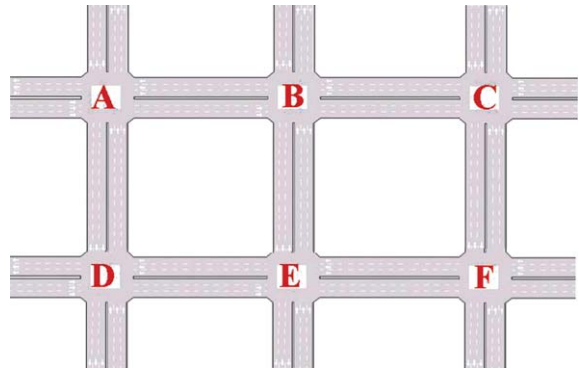


Fig. 5. The traffic network in the simulations.

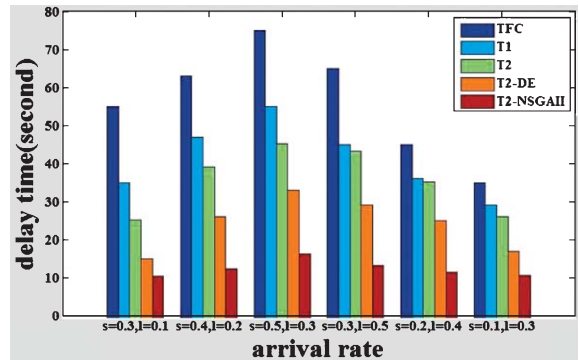


Fig. 6. The delay time of intersection A.

in Fig. 5. The proposed method and targets are tested in same traffic flow patterns and the results are compared and analyzed. The sequence of signal phasing, traffic flow information and signal plans are generated by the matrix of origin-destination of PARAMICS software. The bench mark objects include 4 kinds of commonly used methods, includes: simple fixed-time control (FTC) [2], fuzzy logic control based on type-1 (T1) [1], fuzzy logic control based on type-2 (T2) [8], fuzzy logic control based on type-2 isolatedly optimized by DE (T2-DE) [13].

The comparisons of the delay time of the intersections among different methods are shown in Figs. 6 to 11, where  $s$  represents the arrival rate of straight vehicles per second and  $l$  denotes the arrival rate of left-turning vehicles per second.

From the figures, we can see that T2-NSGAI has better performance compared with the benchmarks TFC, T1, T2, T2-DE, the average total delay time is reduced by 76.3%, 65.1%, 58.3% and 35.4% respectively. The average delay of T2-NSGAI is the smallest, while the

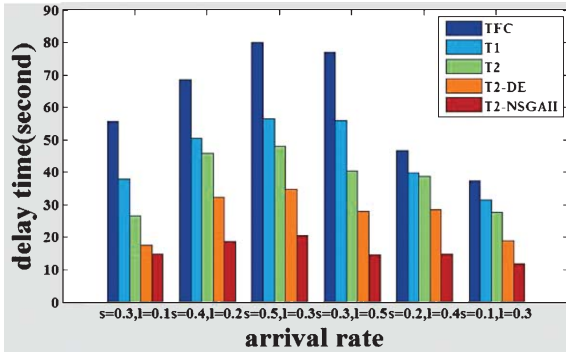


Fig. 7. The delay time of intersection B.

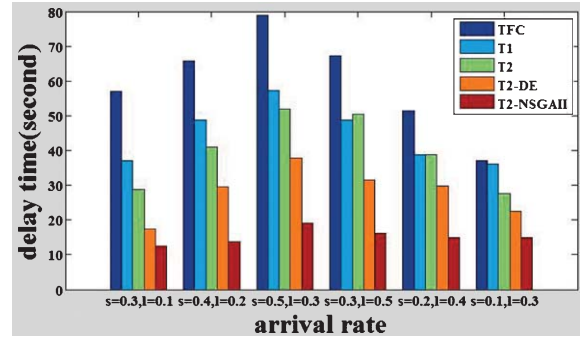


Fig. 10. The delay time of intersection E.

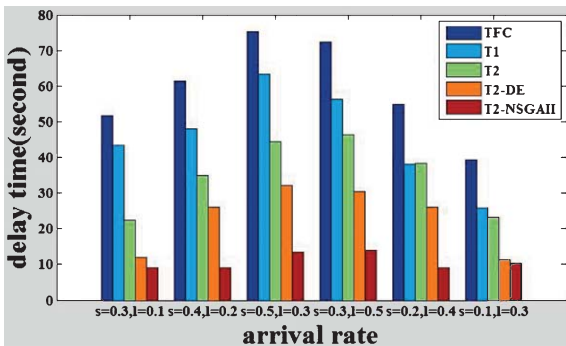


Fig. 8. The delay time of intersection C.

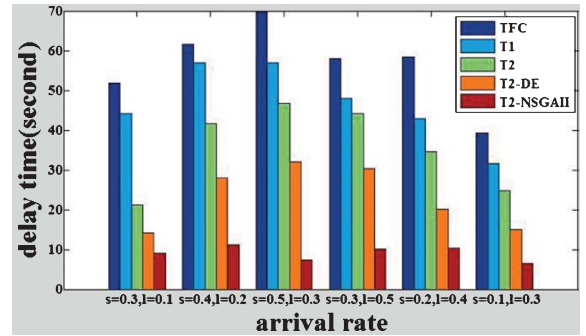


Fig. 11. The delay time of intersection F.

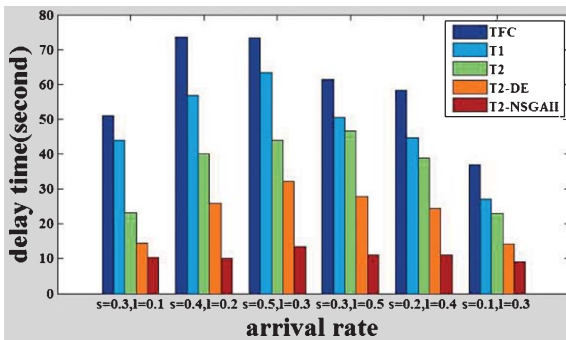


Fig. 9. The delay time of intersection D.

simple FTC method gets the worst performance. Moreover, type-2 FLC methods optimized by DE or NSGAI perform better than the T2 without optimization, which demonstrates that the optimization can facilitate the FLC systems to get suitable parameters and further improve the throughput rate of the traffic system. The figures also show that T2-NSGAI is superior to T2-DE, that's because the fuzzy rules and MF parameters of all

the adjacent intersections of T2-NSGAI are simultaneously optimized, it makes sure that the rules and the MF parameters are compatible. However, in T2-DE, the fuzzy rules, MF parameters are optimized independently, which is prone to get trapped into local optimal solutions and make the rules and parameters incompatible.

The average queue lengths of intersections are shown in Table 2. At the boundary intersections such as A, C and D, the arrived vehicles can smoothly pass through them. Meanwhile, some intersections in the central blocks begin to get traffic jam. From Table 2 we can find that, the queue length of T2-NSGAI method is the smallest and the level of crowdedness is less than others. Furthermore, the results shows that the vehicle

Table 2  
The queue lengths of intersections (s =0.4, l=0.2)

Model	A	B	C	D	E	F
FTC	20.1	31.5	18.8	21.3	33.5	20.8
T1	16.6	21.4	16.4	18.6	22.5	19.0
T2	13.7	18.9	15.1	15.9	19.4	15.8
T2-DE	10.1	15.6	14.0	13.4	16.5	13.9
T2-NSGAI	6.9	11.5	9.6	9.8	12.2	6.5

queue length of all the T2 method (T2, T2-DE, and T2-NSGAI) is less than that of T1, which demonstrates our theoretical analysis that type-2 FLC has the ability of deal with more uncertainties which exist in the traffic control system, because of its special three-dimensional membership functions holds more uncertainties. It should be noted that the communication and changing phase sequence information between intersections also facilitates the FLC system, especially when they are simultaneously optimized.

## 6. Conclusion

In this paper, an intelligent traffic signal controller based on type-2 FLC and NSGAI was proposed. As the type-2 fuzzy logic can well model nonlinear systems, especially on the condition that only a small number of data or expert knowledge is available, it has better ability in tackling uncertainties and robustness than traditional type-1 FLC. Therefore, type-2 FLC is more suitable for traffic signal control. Then NSGAI was employed to optimize both the fuzzy rules and the MF parameters, as it can well optimize multiple objects simultaneously compared with other evolutionary algorithms. The proposed method and other benchmarks were applied to a six-intersection traffic network. Different traffic scenarios were simulated and compared with each other. The results showed that T2-NSGAI got better performance than the other methods, which demonstrated T2-NSGAI's ability in the application of complicated traffic signal control.

## Acknowledgments

This work is sponsored by the National Natural Science Foundation of China (61402308).

## References

- [1] C.H. Chou and J.C. Teng, A fuzzy logic controller for traffic junction signals, *Information Sciences* **143**(1) (2002), 73–97.
- [2] D.I. Robertson, *TRANSYT: A Traffic Network Study Tool, Technical Report*, Transport Research Laboratory in the United Kingdom, 1969.
- [3] D. Zhao, Y. Dai and Z. Zhang, Computational intelligence in urban traffic signal control: A survey, *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews* **42**(4) (2012), 485–494.
- [4] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, A fast and elitist multiobjective genetic algorithm: NSGA-II, *IEEE Transactions on Evolutionary Computation* **6**(2) (2002), 182–197.
- [5] M.B. Trabia, M.S. Kaseko and M. Ande, A two-stage fuzzy logic controller for traffic signals, *Transportation Research Part C* **7** (1999), 353–367.
- [6] O. Castillo and P. Melin, A review on the design and optimization of interval type-2 fuzzy controllers, *Applied Soft Computing* **12**(4) (2012), 1267–1278.
- [7] O. Castillo, R.M. Martinez, P. Melin, F. Valdez and J. Soria, Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot, *Information Science* **192**(1) (2012), 19–38.
- [8] P.G. Balaji and D. Srinivasan, Type-2 fuzzy logic based urban traffic management, *Engineering Applications of Artificial Intelligence* **24** (2011), 12–22.
- [9] Q. Liu and J. Xu, Traffic Signal Timing Optimization for Isolated Intersections Based on Differential Evolution Bacteria Foraging Algorithm, *Proceeding of the 8th International Conference on Traffic and Transportation Studies*, 2012, pp. 210–215.
- [10] R. Evers and S. Proost, The myth of traffic-responsive signal control: Why common sense does not always make sense, *Transportation Research Part A* **77** (2015), 350–357.
- [11] S.M. Rahman and T.N. Ratrou, Review of the fuzzy logic based approach in traffic signal control: prospects in Saudi Arabia, *Journal of Transportation Systems Engineering and Information Technology* **9**(5) (2009), 58–70.
- [12] S. Timotheou, C.G. Panayiotou and M.M. Polycarpou, Towards Distributed Online Cooperative Traffic Signal Control Using the Cell Transmission Model, *Proceedings of the 16th Intelligent Transportation Systems Conference*, 2013, pp. 1737–1742.
- [13] Y.R. Bi, D. Srinivasan, X.B. Lu, Z. Sun and W.L. Zeng, Type-2 fuzzy multi-intersection traffic signal control with differential evolution optimization, *Expert Systems with Applications* **41** (2014), 7338–7349.