bidang
REKAYASA

INTELLIGENT TRAFFIC LIGHT CONTROL BASED ON INTERVAL TYPE 2 FUZZY SYSTEM

MUHAMMAD ARIA
Department of Electrical Engineering
Engineering and Computer Science Faculty
Universitas Komputer Indonesia

This paper presents the design and simulation of an intelligent traffic lights controller based on Interval Type 2 Fuzzy System to control a traffic complex intersection. To control a set of intersection we distribute controls to the controller at each intersection. The controller contains fuzzy next phase module and fuzzy green phase module. The next phase module is working on the phasing, while the green phase module belongs to the green extension level of this multi-level control system. A macroscopic simulator has been developed to simulate the situation of a traffic control intersection. And this simulation software was used to facilitate the evaluation of the proposed Type 2 Fuzzy System strategy. The software allows simulation of different traffic conditions at the intersection. The Type 2 Fuzzy Logic System is compared with Type 1 Fuzzy System and a conventional Fixed-Time Controller. The simulation results show that the Type 2 Fuzzy Logic Controller has better performance in the case of time-varying traffic patterns and heavy traffic conditions, Interval Tupe-2 Fuzzy algorithm reduced average vehicle delay 13,2 % better than Fixed Timer and 1,6 % than Type 1 Fuzzy algorithm. But Interval Type 2 Fuzzy computation time is more complex than both Fixed Timer and Type 1 Fuzzy algorithm, so Interval Type 2 Fuzzy system takes 3.6 times slower than Fixed Timer and 1.8 slower than Type 1 Fuzzy algorithm. In application, Interval Type 2 Fuzzy algorithm needs 17888 units of memory, while Fixed Timer only needs 4840 unit memory and Type 1 Fuzzy algorithm just needs 13032 units memory.

Index Terms – Fuzzy Decision Module, Fuzzy Green Phase Module, Fuzzy Next Phase Module, Interval Type 2 Fuzzy System, Karnik-Mendel algorithms

INTRODUCTION

In the real world, especially in metropolitan areas, there are many intersections and each intersection is located close to its neighborhoods. There are many conventional methods for traffic signal control but sometimes they fail to deal with complex intersections, time-varying traffic conditions efficiently. Various strategy based on fuzzy logic system have been studied to control the intersection group [1] - [6]. This paper explores the use of interval type 2 TSK Fuzzy system (T2TSK), which is well known for its powerful in handling an uncertainties [7].

The system allows communications with neighboring controllers and manages phase sequences and phase lengths adaptively according to traffic density, waiting time of vehicles and congestion.

Performance of interval type 2 TSK fuzzy then comparing between type 1 TSK Fuzzy and conventional fixed-time controller. Although there are a lot of work on type 2 Fuzzy as discussed in [8] - [10], as far as the author concerns, there is no work on development of T2 TSK Fuzzy is suitable for complex intersection traffic control. Contribution of this paper is to propose T2TSK algorithm suitable for complex intersection traffic control and compare the performance between the interval type 2 TSK Fuzzy and type 1 TSK Fuzzy.

The paper is arranged as follows. Interval type 2 TSK Fuzzy structure is provided in Section II. Fuzzy Logic Traffic Signal Controllers for Complex Intersections is presented in Section III. Design of traffic simulator are developed in Section IV. The simulation results are provided in Section V. In Section VI, we conclude with conclusion.

T2TSK STRUCTURE

Type-2 fuzzy sets were originally presented by Zadeh in 1975. The new concepts were introduced by Mendel and Liang allowing the characterization of a type-2 fuzzy set with a superior membership function and an inferior membership function; these two functions can be represented each one by a type-1 fuzzy set membership function. The interval between these two functions represent the footprint of uncertainty (FOU), which is used to characterize a type-2 fuzzy set. Type-2 fuzzy sets allow us to handle linguistic uncertainties, as typified by the adage "words can mean different things to different people".

For type-2 TSK models, there are three possible structure [11]:

1. Antecedents are type-2 fuzzy sets, and consequents are type-1 fuzzy sets. This

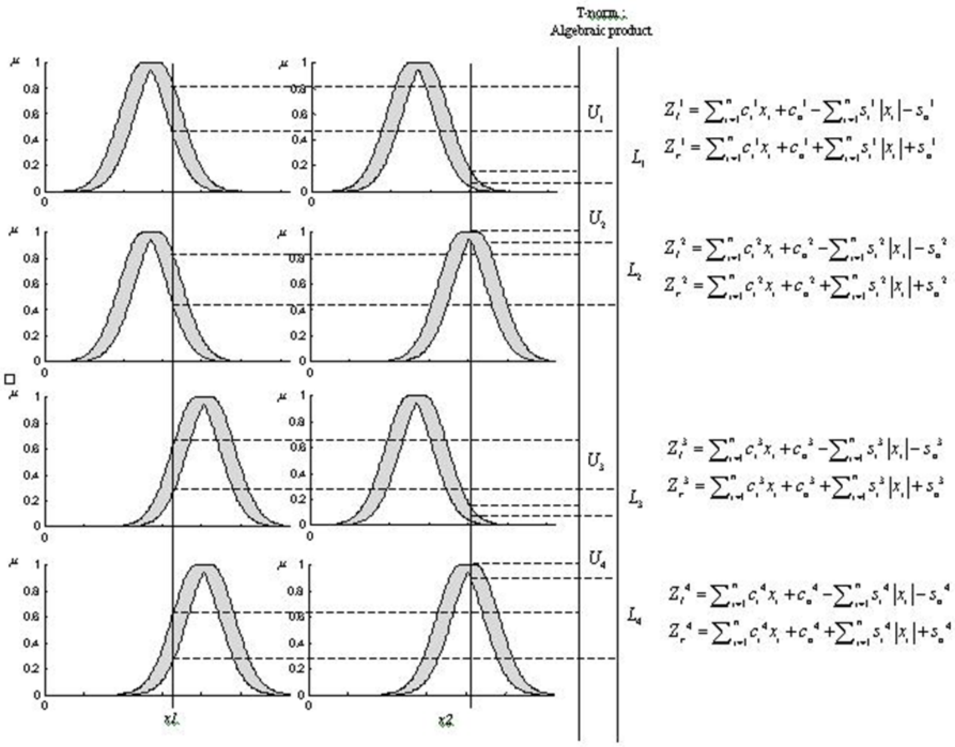

Figure 1. Interval type 2 TSK Fuzzy System Structure

is the most general case and we call it Model I.

2. Antecedents are type-2 fuzzy sets, and consequents are crisp number. This is special case or Model I and we call it model II.
3. Antecedents are type-1 Fuzzy sets and consequents are type-1 fuzzy sets. This is another special case of Model I and we call it Model III.

We use Model I to design interval type-2 TSK Fuzzy system in this paper. A schematic diagram of the proposed T2TSK structure is shown in **Figure 1.**, which is organized into *i* input variables and *m* rules.

Rule Base

In a first-order type-2 TSK Model I with a rule base of *m* rules and *n* input variables, is denoted as

$$\text{IF } x_1 \text{ is } \mu_1{}^b(x_1) \text{ AND} \ldots \text{AND } x_a \text{ is } \mu_a{}^b(x_a) \quad \text{THEN } Z \text{ is } p_1{}^b x_1 + p_2{}^b x_2 + \ldots + p_a{}^b x_a + p_0{}^b \tag{1}$$

where $b \in [0, m]$ and $a \in [0, n]$. The consequent parameter $p_1{}^b, p_2{}^b, \ldots, p_a{}^b, p_0{}^b$, which are type- sets, has interval, is denoted as

$$p_a{}^b = [c_a{}^b - s_a{}^b, c_a{}^b + s_a{}^b] \tag{2}$$

The membership grades
$\mu_1{}^b(x_1), \mu_2{}^b(x_2), \ldots, \mu_a{}^b(x_a)$
are interval sets to, which denoted as

$$\mu_a{}^b = [\underline{\mu}_a^b, \overline{\mu}_a^b] \tag{3}$$

Where $\underline{\mu}_a^b$ is lower membership function and $\overline{\mu}_a^b$ is upper membership function. These rules let us simultaneously account for uncertainty about antecedent membership functions and consequent parameter values.

Fuzzification

This process is transforming the crisp input to a type-II fuzzy variable. The primary membership functions for each antecedent are interval type-2 fuzzy systems described by Gaussian primary membership function with uncertain means, denoted as

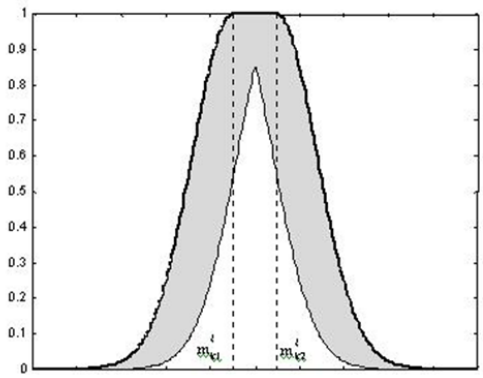

Figure 2. Gaussian interval type-2 fuzzy membership function with uncertain means

where $m_a^b \in [m_{a1}^b, m_{a2}^b]$ is the uncertain mean, with $a = (1,\ldots,n)$ is the number of antecedent, $b = (1,\ldots,m)$ is the number of rules and σ_a^b is the standard deviation.

$$\mu_a^b(x_a) = \exp\left[-\frac{1}{2}\left(\frac{x_a - m_a^b}{\sigma_a^b}\right)^2\right] \tag{4}$$

There are two kinds of type-2 sets. First is a gaussian type-2 fuzzy set, which the membership grade of every domain point is a Gaussian type-1 set contained in [0,1]. Second is an interval type-2 fuzzy set which the membership grade of every domain point is a crisp set whose domian is some interval contained in [0,1]. **Figure 2.** shows gaussian interval type-2 fuzzy membership function with uncertain means.

The upper membership function is defind as

$$\bar{\mu}_a^b(x_a) = \begin{cases} N(m_{a1}^b, \sigma_a^b, x_a), & x_a < m_{a1}^b \\ 1, & m_{a1}^b \le x_a \le m_{a2}^b \\ N(m_{a1}^b, \sigma_a^b, x_a), & x_a > m_{a2}^b \end{cases} \quad (5)$$

where

$$N(m_{a1}^b, \sigma_a^b, x_a) = \exp\left[-\frac{1}{2}\left(\frac{x_a - m_a^b}{\sigma_a^b}\right)^2\right] \quad (6)$$

And lower membership function is defind as

$$\underline{\mu}_a^b(x_a) = \begin{cases} N(m_{a2}^b, \sigma_a^b, x_a), & x_a \le \dfrac{m_{a1}^b + m_{a2}^b}{2} \\ N(m_{a2}^b, \sigma_a^b, x_a) & x_a > \dfrac{m_{a1}^b + m_{a2}^b}{2} \end{cases} \quad (7)$$

Fuzzy Inference System

Fuzzy inference mechanism applies the fuzzy reasoning on the rules in the rule base in order to derive a mathematically reasonable output or conclusion which represents the problem conditions best. Fuzzy inferences in antecedent using algebraic product, is denoted as

$$\underline{W}^b = \underline{\mu}_1^b(x_1) \times \underline{\mu}_2^b(x_2) \times \ldots \times \underline{\mu}_n^b(x_n) \quad (8)$$

and

$$\overline{W}^b = \bar{\mu}_1^b(x_1) \times \bar{\mu}_2^b(x_2) \times \ldots \times \bar{\mu}_n^b(x_n) \quad (9)$$

Figure 3. shows Fuzzy inference illustrative example of the simplified case with two input variable.

The interval value of the consequent Z^b is $Z^b = [Z_l{}^b, Z_r{}^b]$, where

$$\begin{aligned} Z_l{}^b &= \sum_{i=1}^n c_i^b x_i + c_0^b - \sum_{i=1}^n s_i^b |x_i| - s_0^b \\ Z_r{}^b &= \sum_{i=1}^n c_i^b x_i + c_0^b + \sum_{i=1}^n s_i^b |x_i| + s_0^b \end{aligned} \quad (10)$$

and $Z_l{}^b$ and $Z_r{}^b$ denote the lower and upper values of consequent output for b th rule. c_i^b denotes the center (mean) of Z^b and s_i^b denotes the spread of Z^b.

Type Reduction

The Karnik-Mendel algorithms is used for determining c_l and c_r. This process takes the type-2 output set and convert it to a type-1 set. The five steps for determining c_r[12] :

[1] Initialize θ_r^b by setting:

$$\theta_r^b = \frac{1}{2}\left[\underline{W}^b + \overline{W}^b\right] \quad b = 1, \ldots, n \quad (11)$$

and then compute :

$$c' = \frac{\sum_{b=1}^m \theta_r^b Z_r^b}{\sum_{b=1}^m \theta_r^b} \quad (12)$$

[2] Find k_r $(1 \le k_r \le N-1)$ such that

$$Z_r^{k_r} \le c' \le Z_r^{k_r+1} \quad (13)$$

[3] Set :

$$\theta_r^b = \begin{cases} \underline{W}^b & b \le k_r \\ \overline{W}^b & b \ge k_r + 1 \end{cases} \quad (14)$$

And compute :

$$c'' = \frac{\sum_{b=1}^k Z_r^b \underline{W}^b + \sum_{b=k+1}^m Z_r^b \overline{W}^b}{\sum_{b=1}^k \underline{W}^b + \sum_{b=k+1}^m \overline{W}^b} \quad (15)$$

[4] Check if $c'' = c'$. If yes, stop and set $c'' = c_r$. If no, go to step [5]

[5] Set $c' = c''$ and go to step [2]

For determining c_l, same as previouse procedur, except in step 3, set

$$\theta_l^b = \begin{cases} \underline{W}^b & b \le k_l \\ \overline{W}^b & b \ge k_l + 1 \end{cases} \quad (16)$$

so that :

$$c'' = \frac{\sum_{b=1}^k Z_l^b \underline{W}^b + \sum_{b=k+1}^m Z_l^b \overline{W}^b}{\sum_{b=1}^k \underline{W}^b + \sum_{b=k+1}^m \overline{W}^b} \quad (17)$$

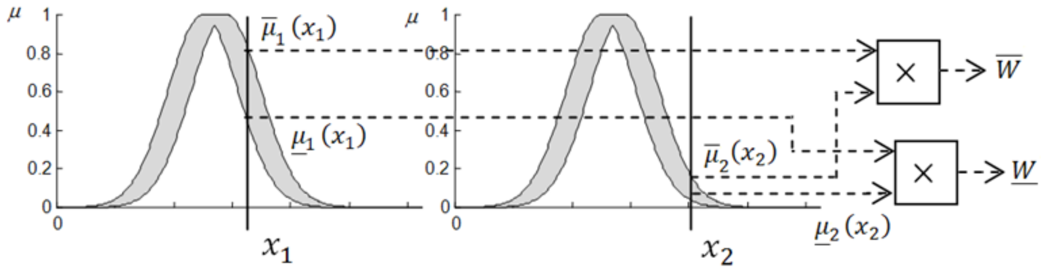

Figure 3. Illustrative example of inference mechanism using algebraic product

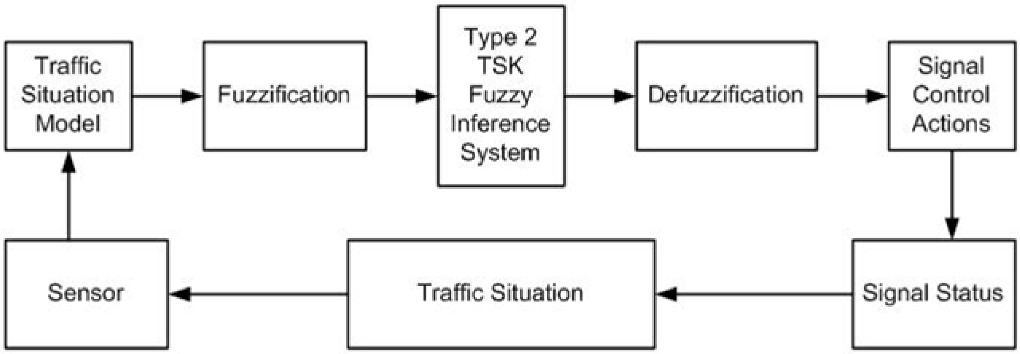

Figure 4. Fuzzy Traffic Signal Control

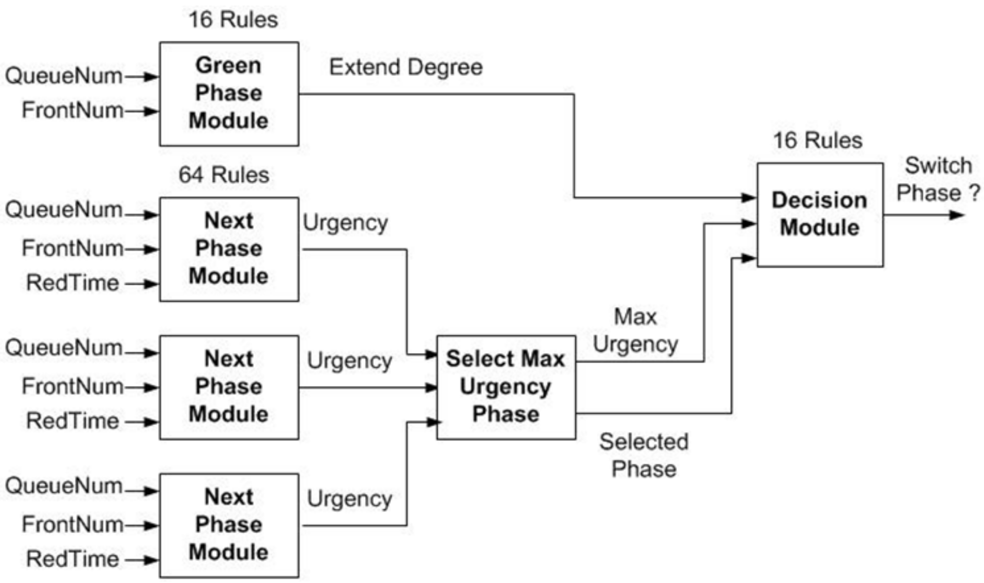

Figure 5. General Structure of logic traffic signal controllers

Defuzzification

Since the resultant type-reduced output is an interval type-1 fuzzy set, the output of fuzzy can be calculate using the average of its lower and upper bounds

$$y = \frac{c_l + c_r}{2} \quad (18)$$

FUZZY LOGIC TRAFFIC SIGNAL CONTROLLERS FOR COMPLEX INTERSECTIONS

A fuzzy logic controlled traffic lights uses sensors that count cars. This provides the controller with traffic densities in the lanes and allows a better assessment of changing traffic patterns. As the traffic distributions fluctuate, the fuzzy controller can change the signal light accordingly. The fuzzy logic controller determines whether to extend or terminate the current green phase based on a set of fuzzy rules. The flow diagram of a controller is shown if **Figure 4**.

The objective of the Fuzzy traffic light controllers is to reduce the total delay time of waiting vehicles as well as to avoid heavy traffic congestion and to synchronize the local traffic controller with its neighbors, such as controlling the outgoing vehicles into neighboring traffic controllers. The fuzzy traffic lights controller is designed with a number of useful features so for example is that if a large volume of vehicles are congested at a neighboring intersection, the number of vehicles coming into that intersection will be reduced.

Fuzzy logic traffic signal controllers contains three modules and this is as shown in **Figure 5**. It consist of a Next Phase Module, a Green Phase Module and a Decision Module. The Next Phase Module observes the condition all the other phases except the green phase. The Select Max Urgency Phase selects the most urgent phase. The Green Phase Module observes the condition of traffic flow of the green phase only. The Decision Module decides the urgency degree between the Next Phase and the Green Phase Modules. It also decides by how long to extend the green phase signal or whether to change to other phases. For example, if the Green Phase Module is more urgent than the Next Phase Module, the green signal will be extended. On the other hand, if the Next Phase Module is more urgent than the Green Phase Module, the Decision Module will change the green phase signal to another phase.

Fuzzy Logic Next Phase Module

The *NextPhase Module* has 3 inputs and 2 outputs. The 3 inputs are (1) *QueueNum* (2) *FrontNum* and (3) *RedTime*. *QueueNum* (zero, short, medium, large, very large) refers to the number of vehicles remain in a lane during a red light phase. *FrontNum* (short, medium, large, very large) quantifies the number of vehicles in the link between the affected intersection and the downstream intersections. Information regarding the quantity of vehicles in the front link, left link and the right link are all sent to the fuzzy controller. This input is important in order to avoid congested links. Another input which is considered in our design is referred to as *RedTime* (short, medium, large, very large). It calculates the number of vehicles waiting at a red light. This input is considered as to avoid the drivers waiting too long for the green signal. **Figure 6** - **Figure 8** shows the membership functions of QueueNum, FrontNum and RedTime.

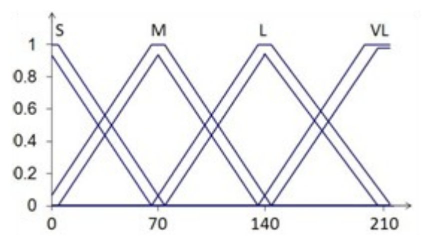

Figure 6. Membership functions for QueueNum

The Next Phase Module ouputs are *Urgency*. *Urgency* represents the worsening traffic condition of the selected phase. If the traffic of the selected phase is rather bad, the Value of *Urgency* increases. The output fuzzy variable *Urgency* has been proposed to have 5 membership functions. **Figure 9.** shows the membership functions of Urgency.

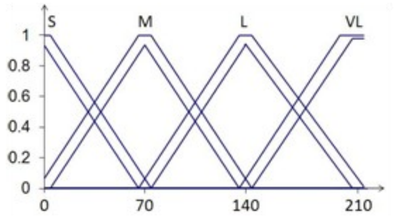

Figure 7. Membership functions for FrontNum

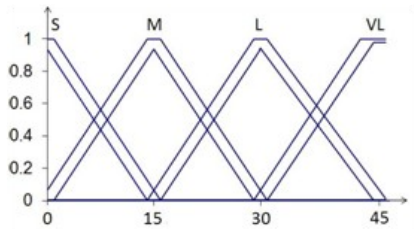

Figure 8. Membership functions for RedTime

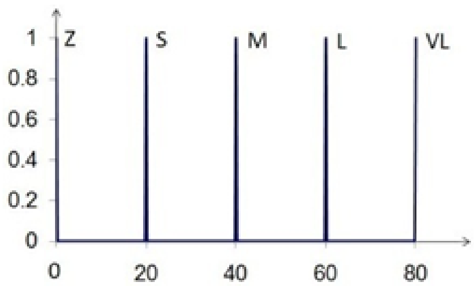

Figure 9. Membership functions for Urgency

For the Select Max Urgency, 2 outputs are considered: (1) *Urgency* and (2) *Phase*. To get the value of *Urgency* of a phase, we should consider the *Urgency* of the each lane in that phase. The *Urgency* values of these lanes are combined as the value of that phase. The phase, which has the highest *Urgency* value, will be the next phase after the green phase. *Phase* refers to the phase selected for the next phase after the green phase.

In this Module, 65 rules have been developed to relate the 3 inputs to the output, *Urgency*, as shown in **Table 1**. In this case, the labels Z means Zero, S for Short, M for Medium, L for Long and VL for Very Long

Fuzzy Logic Green Phase Module

The *GreenPhase Module* observes the traffic conditions of the green phase and it consists of 2 inputs which are (1) *QueueNum* (2) *FrontNum* and 1 output which is *Extend*. *QueueNum* is the remaining vehicles in a lane during the green signal. *FrontNum* is referred to as the number of the vehicles in the link where vehicles will go to this link during the green phase. The output of the *GreenPhase Module* is *Extend* which translates into the possibility of extending the green phase. The membership function of these inputs are the same as those in the *NextPhase Module* as shown in **Figure 6** and **Figure 7.**

The output fuzzy variable, *Extend*, consists of 5 membership functions. This membership function are the same as *Urgency* membership function as shown in **Figure 9**. In this module, 17 rules have been developed as given in **Table 2**

Table 1. Fuzzy Rules of the NextPhase Module

RULE	INPUTS			OUTPUTS
	QueueNum	FrontNum	RedTime	Urgency
1.	Z			Z
2.	S	S	S	S
3.	S	S	M	S
4.	S	S	L	M
5.	S	S	VL	L
6.	S	M	S	S
7.	S	M	M	S
8.	S	M	L	M
9.	S	M	VL	L
10.	S	L	S	S
11.	S	L	M	S
12.	S	L	L	M
13.	S	L	VL	M
14.	S	VL	S	S
15.	S	VL	M	S
16.	S	VL	L	S
17.	S	VL	VL	S
18.	M	S	S	S
19.	M	S	M	M
20.	M	S	L	L
21.	M	S	VL	VL
22.	M	M	S	S
23.	M	M	M	M
24.	M	M	L	L
25.	M	M	VL	L
26.	M	L	S	S
27.	M	L	M	M
28.	M	L	L	M
29.	M	L	VL	L
30.	M	VL	S	S
31.	M	VL	M	S
32.	M	VL	L	M
33.	M	VL	VL	M
34.	L	S	S	M
35.	L	S	M	L
36.	L	S	L	VL
37.	L	S	VL	VL
38.	L	M	S	M
39.	L	M	M	L
40.	L	M	L	L
41.	L	M	VL	VL
42.	L	L	S	S
43.	L	L	M	M
44.	L	L	L	M
45.	L	L	VL	L
46.	L	VL	S	S
47.	L	VL	M	S
48.	L	VL	L	M
49.	L	VL	VL	L
50.	VL	S	S	L
51.	VL	S	M	VL
52.	VL	S	L	VL
53.	VL	S	VL	VL
54.	VL	M	S	L
55.	VL	M	M	L
56.	VL	M	L	VL
57.	VL	M	VL	VL
58.	VL	L	S	M
59.	VL	L	M	L
60.	VL	L	L	L
61.	VL	L	VL	VL
62.	VL	VL	S	S
63.	VL	VL	M	M
64.	VL	VL	L	L
65.	VL	VL	VL	VL

Fuzzy Logic Green Phase Module

The *GreenPhase Module* observes the traffic conditions of the green phase and it consists of 2 inputs which are (1) *QueueNum* (2) *FrontNum* and 1 output which is *Extend*. *QueueNum* is the remaining vehicles in a lane during the green signal. *FrontNum* is referred to as the number of the vehicles in the link where vehicles will go to this link during the green phase. The output of the *GreenPhase Module* is *Extend*

which translates into the possibility of extending the green phase. The membership function of these inputs are the same as those in the *NextPhase Module* as shown in **Figure 6** and **Figure 7.**
The output fuzzy variable, *Extend*, consists of 5 membership functions. This membership function are the same as *Urgency* membership function as shown in **Figure 9**. In this module, 17 rules have been developed as given in **Table 2**.

Table 2. Fuzzy Rules of

RULE	INPUTS		OUTPUTS
	Queue Num	FrontNum	Urgency
1	Z		Z
2	S	S	S
3	S	M	S
4	S	L	S
5	S	VL	S
6	M	S	L
7	M	M	M
8	M	L	S
9	M	VL	S
10	L	S	VL
11	L	M	L
12	L	L	M
13	L	VL	S
14	VL	S	VL
15	VL	M	VL
16	VL	L	L
17	VL	VL	M

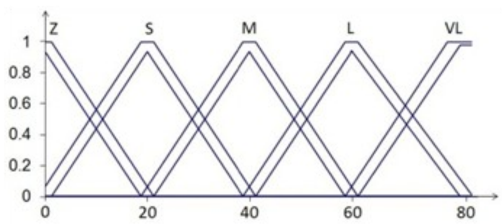

Figure 10. Membership functions for Urgency

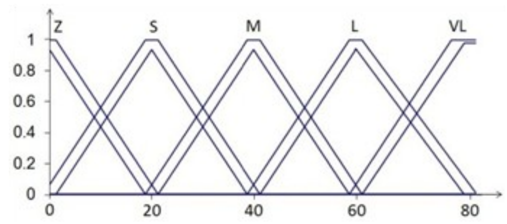

Figure 11. Membership functions for Extend

Fuzzy Logic Decision Module

The Decision Module makes the decision whether to switch to the green phase. Its inputs are the outputs from the *NextPhase* and *GreenPhase* Modules which are *Urgency* and *Extend* as described. These 2 antecedents, *Urgency* and *Extend*, are compared at every interval and the consequent determines whether to change the phase or extend the green signal. In this case we use *Yes* for ending the Green signal phase and change to a new phase, and *No* for continuing the Green signal. If *Urgency* is higher than *Extend*, it means that the traffic condition for the next phase has heavier traffic than the current green phase, and thus the output will change the phase rather than extend the green signal at the current phase. The graphical representation of *Urgency* and *Extend* membership functions is presented in **Figure 10** and **Figure 11**.
The rules of the *Decision Module* are given as in **Table 3**. In all the three modules of the fuzzy controller, the Sugeno inference method are used in the controllers.

DESIGN OF TRAFFIC SIMULATOR

A simulator for intersection group has been developed to test the developed traffic controller. The simulated intersection model consists of nine intersections. Each intersection is connected with its neighborhoods in the four directions. Each link is two way and have capacity 250 cars. **Figure 12**. shows the simulated intersection group.
The intersection group are divided into four groups input-links according to the

Table 3. Fuzzy Rules of the Decision Module

RULE	INPUTS		OUTPUTS
	EXTEND	URGENCY	DECISION
1	Z	Z	No
2	Z	S	Yes
3	Z	M	Yes
4	Z	L	Yes
5	Z	VL	Yes
6	S	Z	No
7	S	S	No
8	S	M	Yes
9	S	L	Yes
10	S	VL	Yes
11	M	Z	No
12	M	S	No
13	M	M	No
14	M	L	Yes
15	M	VL	Yes
16	L	Z	No
17	L	S	No
18	L	M	No
19	L	L	No
20	L	VL	Yes
21	VL	Z	No
22	VL	S	No
23	VL	M	No

location of them: the north input-links, the west input links, the east input links and the south input links. We give the traffic generation plan to each input-link group, so all input-links in a group have the same plan. Cars are generated according to the given plan of a input link and inserted into the link. In the development of the traffic simulator, the following assumptions are made :

i. the intersection is four way junction with traffic coming from the north, west, east and south directions
ii. when traffic from the north and south moves, traffic from the west and east stops, and viceversa
iii. only passenger cars exist and there is no obstruction to traffic flow such as cross-walks.
iv. left turning traffic and right turning traffic are 20% of the traffic of a link respectively

In this simulation we use the following headway equation:

$$h(t) = \frac{a}{h(t-1)} + b \quad (19)$$

SIMULATION RESULTS AND DISCUSSION

Performance of Type 2 Fuzzy Logic System (T2FL) is investigated using simulation studies. The developed controller was compared with Type-1 Fuzzy Logic System (T1FL) and Fixed Time Controller. Simulation was performed under 18 situations.

1. Case that the traffics of all input links are same :
   (a) 900 cars/h
   (b) 1100 cars/h
   (c) 1300 cars/h
   (d) 1500 cars/h
   (e) 1700 cars/h
   (f) 1800 cars/h

2. Case that the traffic changes every 15 minutes
   a. case of light traffic (cars/h)

Time	0 – 15	15 – 30	30 – 45	45 – 60
North	1000	1000	900	1000
West	1200	1100	1100	1000
East	800	1000	1000	1100
South	900	1000	1100	1200

   b. case of normal traffic (cars/h)

Time	0 – 15	15 – 30	30 – 45	45 – 60
North	1300	1400	1300	1300
West	1400	1400	1500	1300
East	1400	1300	1400	1300
South	1400	1300	1400	1300

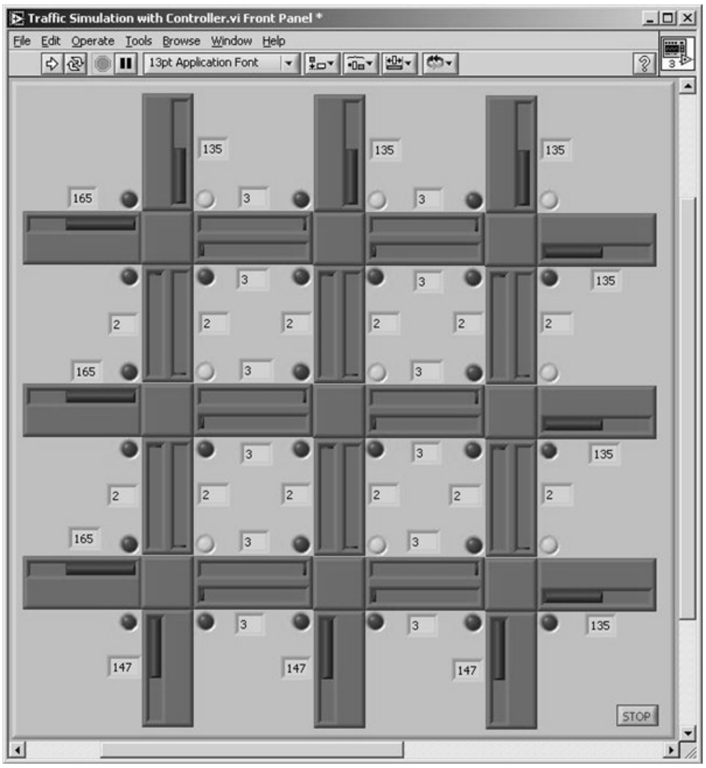

Figure 12. Simulated Intersection Group.

c. case of heavy traffic (cars/h)

Time	0 – 15	15 – 30	30 – 45	45 – 60
North	1600	2000	1600	1900
West	1600	1800	1700	1900
East	1800	1600	1900	1600
South	1600	2000	1700	1800

The average delay time of a car per intersection was collected. Average delay values are calculated by using weighted average method considering traffic volumes of approaches and it is expressed as seconds per vehicle unit for each flow.

The simulation results are summarized in and **Table 4** and **Table 5.** But T2FL computation time is more complex than both Fixed Timer and T1FL algorithm. **Table**

Table 4. Average delay time for Case 1

Case	Fixed-Time	T1FL	T2FL	T1FL Improvement than Fixed-Time	T2FL Improvement than Fixed-Time	T2FL Improvement than T1FL
1.a	54.3	45.9	50	15.60%	8.00%	-9.00%
1.b	55.9	50.4	51.1	9.80%	8.50%	-1.40%
1.c	60.5	60.1	55.1	0.60%	9.00%	8.40%
1.d	71.7	72.1	64.7	-0.60%	9.70%	10.20%
1.e	111.5	89.6	88.1	19.70%	21.00%	1.70%
1.f	137.9	106	106.4	23.20%	22.80%	-0.40%

6 show the comparison of T2FL, T1FL and fixed-time computation time. So Interval Type 2 TSK Fuzzy system takes 3.6 times slower than Fixed Timer and 1.8 slower than Type 1 TSK Fuzzy algorithm. In application, Interval Type 2 TSK Fuzzy algorithm needs 17888 units of memory, while Fixed Timer only needs 4840 unit memory and Type 1 TSK Fuzzy algorithm just needs 13032 units memory.

Table 5. Average delay time for Case 2

Case	Fixed-Time	T1FL	T2FL
2.a	57.0	47.8	48.3
2.b	61.1	56.9	53.0
2.c	110.9	72.6	74.7

Table 6. Computation time

	Average of time computation (microsecond)
Fixed-Time	9375
T1FL	18750
T2FL	33854

CONCLUSION

In this paper, we have proposed the traffic controller for complex intersection group based on interval type 2 TSK fuzzy systems and implement the simulator for performance evaluations. To control a set of intersection, we distribute controls to each controller. Each controller takes charge of controlling its traffic signal and cooperating with its neighborhood. Our approach can be easily extended to any situation. According to the simulation studies, results of the fuzzy logic controller is the same as the fixed-time controller in normal traffic flow. But the simulation shows promising results in the cases of heavy traffic and time-varying traffic with large variance. In the cases of heavy traffic and time-varying traffic, T2FL algorithm reduced average vehicle delay 13,2 % better than Fixed Timer and 1,6 % than T1FL algorithm. But T2FL computation time is more complex than both Fixed Timer and T1FL algorithm, so T2FL system takes 3.6 times slower than Fixed Timer and 1.8 slower than T1FL algorithm. In application, T2FL needs 17888 units of memory, while Fixed Timer only needs 4840 unit memory and T1FL just needs 13032 units memory.

REFERENCES

Mohammad Hossein Fazel Zarandi, Shabnam Rezapour, *A Fuzzy Signal Controller for Isolated Intersections*, Journal of Uncertain System, vol 3, No. 3, pp 174 – 182, 2009

Lin Zhang, Honglong Li, Panos D. Prevedouros, *Signal Control for Oversaturated Intersections Using Fuzzy Logic*, Transportation Research Record, Hawaii, 2004

Jee-Hyong Lee, Keon-Myung Lee, KyoungA Seoing, Chang Bum Kim and Hyung Lee-Kwang, *Traffic Control of Intersection Group Based on Fuzzy Logic*

Marzuki Khalid, See Chin Liang and Rubiyah Yusof, *Control of a Complex Traffic Juncion using Fuzzy Inference*

Tan Kok Khiang, Marzuki Khalid and Rubiyah Yusof, *Intelligent Traffic Lights Control by Fuzzy Logic*

Yetis Sazi Murat, Ergun Gedizlioglu, *A New Approach for Fuzzy Traffic Signal Control*

Jerry M. Mendel (2007), *Type-2 Fuzzy Sets and Systems : An Overview*, IEEE Computational Intelligence Magazine, vol 2, no.1, pp. 20-29

Ching-Hung Lee, Yu-Ching Lin, and Wei-Yu Lai, *Systems Identification Using Type-2 Fuzzy Neural Network (Type-2 FNN) Systems*, Proceedings IEEE International Symposium on Computational Intelligence in Robotics and Automation, 2003

Oscar Castillo and Patricia Melin, *Adaptive Noise Cancellation Using Type-2 Fuzzy Logic and Neural Networks*, Prof. of Fuzzy 2004, IEEE press, 2004

S. Coupland, M. Gongora, R.I.John, K.Wills, *A Comparative Study of Fuzzy Logic Controllers for Autonomous Robots*

Qilian Liang and Jerry M. Mendel (1999), *An Introduction to Type-2 TSK Fuzzy Logic Systems*, IEEE International Fuzzy Systems Conference Proceedings, pp III-1534 – III-1539.

Jerry M. Mendel and Hongwei Wu (2007), *New Results About the Centroid of An Interval Type-2 Fuzzy Set, Including the Centroid of a Fuzzy Granule*, Information Sciences an International Journal, pp 360 – 377.

Salman Mohagheghi (2006), *An Interval Type-II Robust Fuzzy Logic Controller for a Static Compensator in a Multimachine Power System*, International Joint Conference on Neural Networks, pp 2242 – 2244.

Hani Hagras (2007), *Type – 2 FLCs : A New Generation of Fuzzy Controllers*, IEEE Computational Intelligence Magazine, vol 2, no.1, pp. 30 – 43

Jang, J.-S.R., C.-T. Sun, dan E Mizzutami, *Neuro-Fuzzy and Soft Computing,* Prentice Hall Inc., 1997

G.M.Mendez, M.A. Hernandez (2007), *Interval Type-1 Non-Singleton Type-2 TSK Fuzzy Logic Systems Using the Hybrid Training Method RLS-BP*, IEEE Symposium on Foundations of Computational Intelligence

Jerry M. Mendel, Robert I. John and Feilong Liu (2006), *Interval Type-2 Fuzzy Logic Systems Made Simple*, IEEE Transactions on Fuzzy Systems, vol 14, no 6, pp 808 – 821.

Jerry M. Mendel, Hongwei Wu (2006), *Type-2 Fuzzistics for Symmetric Interval Type-2 Fuzzy Sets: Part 1, Forward Problems*, IEEE Transactions on Fuzzy Systems, vol 14, no 6, pp 781 – 792.

Qun Ren, Luc Baron and Marek Balazinski (2006), *Type-2 Takagi-Sugeno-Kang Fuzzy Logic Modeling using Subtractive Clustering*,

N.N. Karnik and J.M. Mendel (1998), *Introduction to Type-2 Fuzzy Logic Systems*, University of Southern California, Los Angeles

L. Li, H. Liu (2006), *Type-2 Fuzzy Logic Approach for Short Term Traffic Forecasting*, IEE Proc. Intell. Transp. Syst. Vol 153