Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

Fuzzy logic controller for traffic signal controller unit system and modeling with colored petri net

Behnam Barzegar

Department of Computer Engineering, Islamic Azad University, Nowshahr Branch, Nowshahr, Iran
behnam.barzegar@yahoo.com

Abstract

We offer an intellectual fuzzy system for traffic management in which the traffic signal at an intersection is controlled in such a way that it minimizes the stop time thereby implementing better performance. We also offer a solution controlling intersections prioritizing the paths. This solution starts with sending time variable to the control center.

Keywords: Fuzzy traffic signal, Color petri net, Fuzzy logic, Traffic light.

Introduction

Population explosion or migration of rural population is palpable with a great spur in construction activities of residential towers, extending cities and even building residential estates in a metropolitan city. In any countries, traffic managers always look for a way to ease out the traffic pressure. One of the conventional ways of reducing load on city traffic is constructing new roads and adding them to the urban and interurban road network. Most of the times, some paths are connected to each other as intersections. Thus, each intersection needs a police officer to manage traffic logically and reasonably.

But increasing number of conventional traffic signals with all time functioning (constant- time) doesn't help much solving traffic problem. We offer a method that manages traffic in crossroads using fuzzy logic. In fact, implementing proportional values of rules and logics is impossible, unless we apply fuzzy logic which has ability of recognizing and processing these values. Our recommended method includes two input parameters which involve congestion of vehicles at traffic signal and congestion of passing vehicles. For entering proportional values (very less, less, average) to suggested fuzzy system, we could use sensors such as camera and processing related picture or electromagnetic sensors. After processing input values in suggested fuzzy system and applying intellectual one, the output 'time' value in seconds or change of situation, is implemented to traffic light, and eventually vehicles are directed in their paths at optimal time.

Thus, there is a need to address the problem of managing crossroads and controlling available traffic signals which has special importance. This could be done by prioritizing paths *i.e.,* in main roads of high traffic, the flow should be quicker. In our recommended method, from available traffic signal at each intersection of main path, a signal is sent to crossroad control center. We first simulate mentioned method with CPN modeling tool to evaluate its performance by entering different values. Colored petri net (CPN) is used for modeling to predict different modes of system and probable occurrences. For better graphics and functionality of the two recommended system, we simulate them with C# programming language.

Related work

The above problem has been argued by previous workers and some activities are executed as highlighted below:

Fuzzy controller system has been suggested which created a time, according to the 2 or 3 arrival parameters and their evaluation. This created time is related to the increasing of time needed when vehicles cross the junction. Shilpa *et al.* (2008) divided a street into 3 longitudinal traffic lanes through camera sensor and image processing. A crossing chance is provided in each lane. An operation is a function performed according to phases. Khiang and Khalid *et al.* (1996) simulated traffic junction on two kinds of controller system (ordinary and fuzzy), according to cases such as waiting time, traffic density, cost etc. Barzegar *et al.* (2011) introduced the simulation of traffic light controller by Fuzzy Petri net through implemented operations.

Traffic signal

Conventional traffic signals are designed with a fixed counter (timer). By changing the conditions of traffic signals in an adjusted time, it allows vehicles to ply in their direction or stop them. Because the rules are arbitrary, the time of traffic signal may also be different from one junction to another. Keeping equal performance of these traffic signals in various traffic conditions may worsen the situation with traffic congestion. Thus, a controller system should be designed, which can understand and execute the results as those implemented by men. It means that the designed controller system must have distinguishing, processing and executing power of the brain of police officers and displays the same result. It should be intelligent in different conditions and resolve the problem. One of the functions of fuzzy logic is the control of time and traffic situation in a junction. With respect to traffic density in a lane, it is not simple to make mathematical formula to determine a specific moment for each four directions. Thus, using fuzzy logic, we can match the traffic density of each direction to conditions, such as too low and low and also consider the allotted time for each side as a traffic function of that side in relation to the traffic of other sides of a junction. For example, to control the traffic of an intersection, the following procedures are performed: if

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

the traffic density in the east-west direction was higher than it in north-south, traffic light will allot a longer crossing time in east-west direction than north-south.

Petri net

In 1962, Petri net was introduced by Carl Adam Petri, as a modeling implement of computer systems. By its subsequent development now we have Colored Petri net (CPN), Time Petri net and Hierarchical Petri net. Coloured Petri Net (CPN) is a tool by which validation of discrete-event systems are studied and modeled. CPNs are used to analyze and obtain significant and useful information from the structure and dynamic performance of the modeled system. Coloured Petri Nets mainly focus on synchronization, concurrency and asynchronous events (Barzegar *et al.,* 2011). The graphic features of CPNs specify the applicability and visualization of the modeled system. Furthermore, synchronous and asynchronous events present their prioritized relations and structural adaptive effects. Petri net characteristics are: 1) more attractiveness for its graphical presentation, 2) not allotting to specific systems, 3) being proper in modeling all systems, 4) less planning but efficient elements which caused the simple use of this tool. A customary concept of CPN is: CPN have six members, $CPN = \{P,T,C,I^-,I^+, M_0\}$, which are:

1) $P=\{p_1, p_2, \dots, pn\}$ denotes a finite and not-empty set of places, 2) $T=\{t_1, t_2, \dots, tm\}$ denotes a finite and not-empty set of transitions, $P \cap T=\varnothing$ 3) C, is a color function which is not-empty and finite set of colors to each place and also is a finite and not-empty set of states to each transition, 4) I^-, I^+ denote the backward and forward incidence functions defined by P×T, such that $I^-(p,t)$, $I^+(p,t) \in [C(t)\rightarrow C(p)MS]$, $\forall (p,t) \in P\times T^2$.

5) M_0, defined the function of P which specifies first signs such as $M_0(P) \in C(P)MS$.

There are four main components in Petri net:

1) (●) Token: specifying the existence in system. 2) (⬭) Place: temporary place for maintenance of Tokens. 3) (→) Arc: show Token directions. 4) (▯) Transition: specifying the main operation in system.

A system can be modeled just by these four simple elements.

Environmental conditions

The following principles and hypothesis are considered in the development of the proposed fuzzy intelligent controller system:

1. Main direction is west-east Street which has priority due to high traffic density.
2. The main direction has several continuous junctions. The first junction is in the beginning of the main street (in west) and the final junction is placed in the end of street (in east).
3. Sideways are north-south directions for each intersection which have less priority than west-east direction (Fig.1).

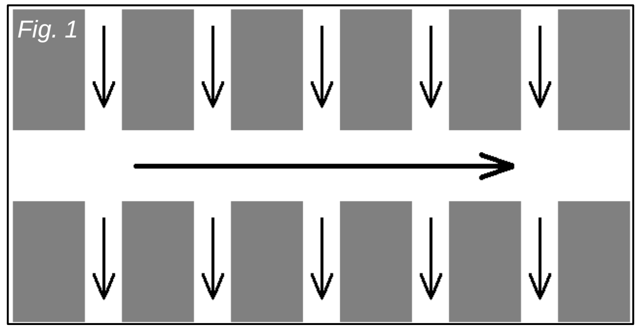


Fig. 1

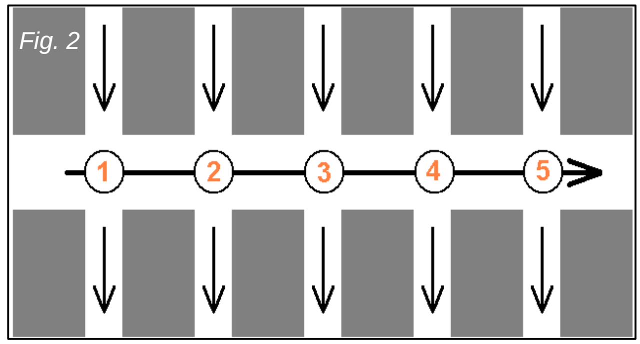


Fig. 2

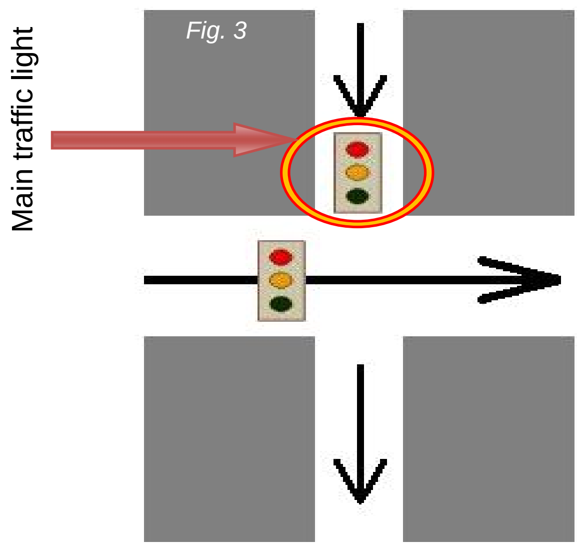


Fig. 3

4. Junction directs vehicles from north to south and west to east.
5. When vehicles cross the junction from north to south, moving by west to east direction is not allowed (Fig.2).
6. Turning to left and right is not argued.
7. Thus, vehicles are allowed to move directly.
8. Fuzzy intelligent traffic lights understand and analyze the density of vehicles in north to south and west to east directions.
9. Maximum and minimum times of a green color are 40 and 5 second respectively.
10. Main traffic signal: Those placed in each junction on the west-east direction change the traffic light depend on the Main traffic signal in north-south direction (sub traffic light) (Fig.3).
11. All main traffic lights send signals to control center or receive signal from it.

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

Table 1. Input variable: queue in red phase and crossing cars in junction. Output variable: time in per second.

Input		Output
Arrival	Queue	Time(s)
Very Little (vl)	Very Few (vf)	Very Short (vs)
Little (li)	Few (fe)	Short (sh)
Average (av)	Mediume (me)	Middle (mi)
Much (mu)	Many (ma)	Numerous (nu)

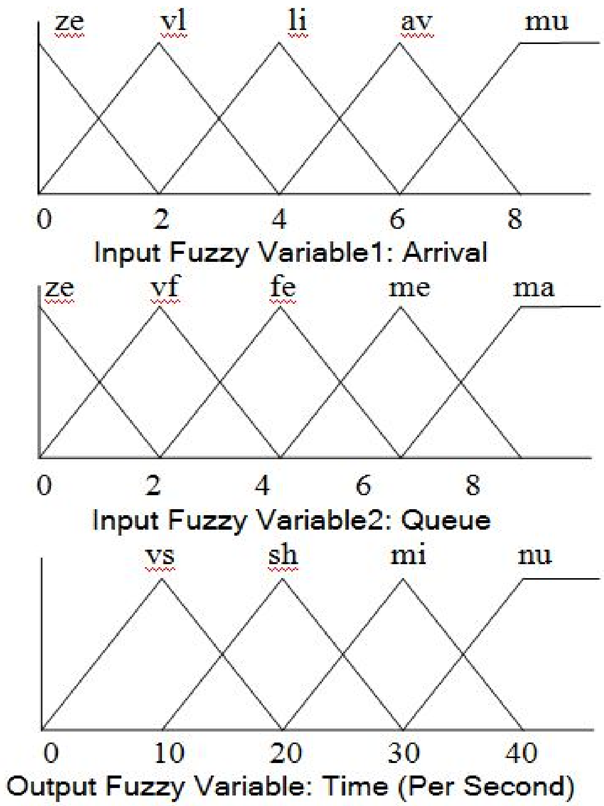
The proposed traffic signal system (First phase)

We can control traffic in a junction through utilizing the proposed traffic light system in first phase.

Input and output parameters

For this controller system, there are five membership functions for each input and output parameters. They express the relations of fuzzy values (Table 1).

Fig. 4. degree of membership for each Fuzzy variables



In Fig.4, we showed the diagram of parameters, where axis y is the degree of membership for each fuzzy variable. In the input variable diagram, axis x shows the cars density and in output variable diagram, it denotes time in per second. As you are seeing, time will increase and also will be adjusted through the growth of cars density. Two points should be mentioned:

1. These diagrams and variables can be developed for various traffic conditions and different cases.
2. In rules table, different condition will be considered where the number of vehicles reaches zero in each direction or even in two (explained below).

According to input parameters and on the basis of determined rules in Table 2, the output of table or time is according to second, or is on changeable or unchangeable traffic light conditions. Now, explain the rules table.

Table 2. Fuzzy rules in table form

Rules Table		Arrival				
		ze	vl	li	av	mu
Queue	ze	nc	nc	nc	nc	nc
	vf	cs	vs	sh	mi	nu
	fe	cs	vs	sh	mi	nu
	me	cs	vs	vs	sh	mi
	ma	cs	vs	vs	sh	mi

For example, if the density of vehicles waiting behind the red color (density of red phase) is medium (me) and the density of moving cars (density of green phase) is light (Li) , then, the green color length will be too short. And if the density of cars in the red color queue is medium (me) and crossing cars density is many, middle time is attributed.

Also in conditions where there is no vehicle in one or two lanes, correct performance of traffic light is predicted. This situation can be demonstrated with “Ze”, that means, density of cars in lane, is Zero. For example, the latest condition in a traffic light will be preserved if there is no car in a junction to move (Ze) and in a red color queue (Ze). It means that a traffic light in the latest lane remains in green color without time counting, because it is probable that the next car crosses the same lane. There is no argument about time and its counting in this condition. Because both lanes are empty, then counting and changing the condition will be irrational. Changing the traffic light is not necessary even if the density of cars moving is higher than zero and the density of cars in a red color queue corresponds to zero (Ze). But if the moving cars density corresponds to zero (Ze) and the density of cars in a red color queue is higher than zero, the traffic light will immediately change to a proper time in order to cars not be have to wait behind the red color for a green one.

These rules can be shortened as follows:

IF ((Queue is me) AND (Arrival is li)) Then (Time is vs)

IF((Queue is me) AND (Arrival is mu)) Then (Time is mi)

IF ((Queue is ze) AND (Arrival is ze)) Then (Time is nc)

IF ((Queue is me) AND (Arrival is li)) Then (Time is nc)

IF ((Queue is me) AND (Arrival is ze)) Then (Time is cs)

It is clear that these rules can extend to various circumstances and different junctions. There is no time extension, but time will be formulated on the basis of traffic density in junction and timing of flow direction will be implemented on traffic lights. Then time will be sent to the control center as a parameter in order to make the

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

desired decision and implement necessary executive operations.

Inference engine

It is supposed that values obtained by sensors, which are the values of input parameters, are integer or will be transformed to it by rounding. According to Fig.4, they will be transformed to fuzzy relative values and will be admitted as a member of defined degrees (zero, too low, low, medium, many). Now, we investigate these transformations. This will perform through a proper formulating by controller system. Formulating an algorithm or a problem is performed to execute the programming through it with computers and to mechanize the system.

After formulating operations, it is necessary to replace inputs into formula and observe the results. As mentioned before, we specified five degree of the memberships for each input which will have a function for each degree as the following: Be careful that the output of function (density of lane) will return in a variable, De.

$$\mu_1(De)=\begin{cases} 0 & x<0 \\ 0.5*(2-x) & 0<x<2 \\ 0 & \text{Other} \end{cases} \qquad \mu_2(De)=\begin{cases} 0.5*(x) & 0<x<2 \\ 0.5*(4-x) & \\ 0 & \text{Other} \end{cases}$$

$$\mu_3(De)=\begin{cases} 0.5*(x-2) & 2<x<4 \\ 0.5*(6-x) & 4<x<6 \\ 0 & \text{Other} \end{cases} \qquad \mu_4(De)=\begin{cases} 0.5*(x-4) & 4<x<6 \\ 0.5*(8-x) & 6<x<8 \\ 0 & \text{Other} \end{cases}$$

$$\mu_5(De)=\begin{cases} 0.5*(x-6) & 6<x<8 \\ 1 & x>=8 \end{cases}$$

Now we have a time which is appropriate with density of vehicles and is countable (timing).

The proposed control center system (Second phase)

Now, through the proposed system of a control center, we control traffic lights in junctions in order to prioritize some streets to each other.

Input and output parameters

If n was proposed as the number of junctions in a control center, then we will have n-1 input parameter and n-1 output parameter; because the first junction sends just one parameter to the control center and the last junction in Main Street just receives one parameter from it. The rest of junctions have both two parameters.

But this can be applied for one-way streets. In two-way streets the number of input parameters sent to the control center and output parameters received are equal with n and all junctions have both two parameters.

The input parameters of the control center are parameters which Main traffic light in each junction is timing vehicles to cross the main direction and then are sent to control centers. The output parameters of control centers are those which are calculated as LT + T.

T is the time which received from previous traffic light. LT is a fixed time which a car needs to traverse the distance of previous traffic light and current traffic light.

The sum of these times must be sent to current traffic light. The condition of current junction must change to green color according to send time for Main Street in order for traffics to cross the junction.

When the traffic density is in an average level in Main Street, the use of fuzzy intelligent traffic light in each junction leads to more appropriate operation. Even in equal densities between Main Street and cross Side Street, we can observe a good operation of fuzzy intelligent traffic light.

Simulation by CPN

First phase simulation

First, two values (parameters) were obtained from the sensors of north-south and west-east directions. Sensors will transform density of cars on the basis of exact values. Arrived parameters will change to fuzzy values (relative values) and will be admitted as a member of too few, few, medium and many degrees (Fig.5). Then two fuzzy values**,** should be analyzed and being process on the basis of rules table to produce output (Fig.6).

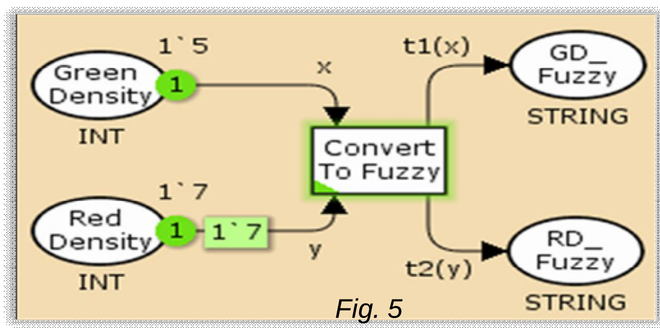


Fig. 5

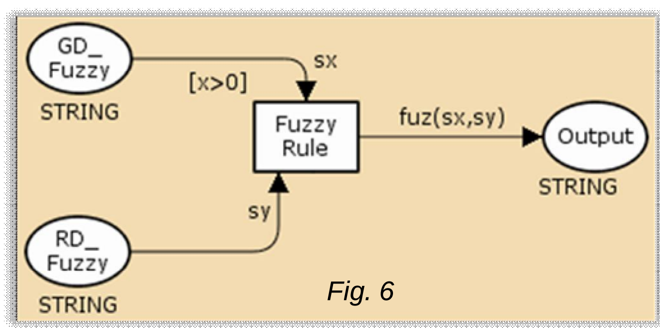


Fig. 6

Now, the output will be considered as input for different functions in order to distinguish that output is a time which must be counted or it is a condition which should be executed (Fig.7). Active function output will be implemented on traffic light.

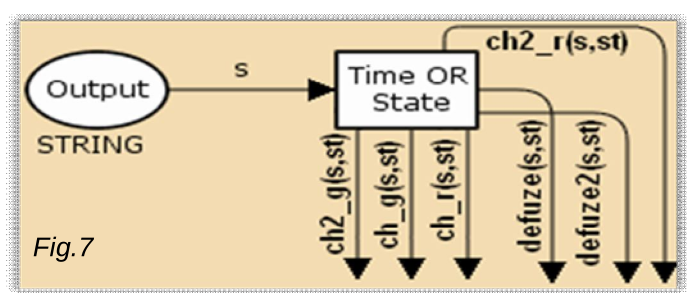


Fig.7

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

Second phase simulation

When the traffic density reaches the average or higher level at the beginning of Main Street, the first traffic light (at the beginning of Main Street) according to its intelligent operation and after evaluating traffic density, devotes a time to cars cross from junction and will change the junction state into crossing condition for Main Street.

Then it will send this devoted time as a parameter for the control center. The control center sums this received time with fixed distance time (Time needed to traverse the distance of the first and second junctions) and then it will send the obtained results to the second traffic signal (in the second junction). Thus, the second junction condition will change to green color equal with the sent time in order for the traffic to leave the Main Street (Fig.8 & 9).

Now the second traffic signal (main traffic light in second junction) sends devoted time to control center. As soon as the control center received this parameter, it gets the time devoted by the first traffic light (in first junction) and sums it with the distance fixed time (time which needed to traverse from the second junction to the third one). If the first junction condition turns to red color for Main Street, the obtained results will be sent to the third traffic light (in the third junction).

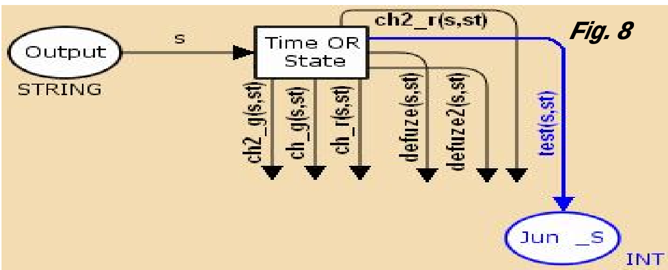


Fig. 8

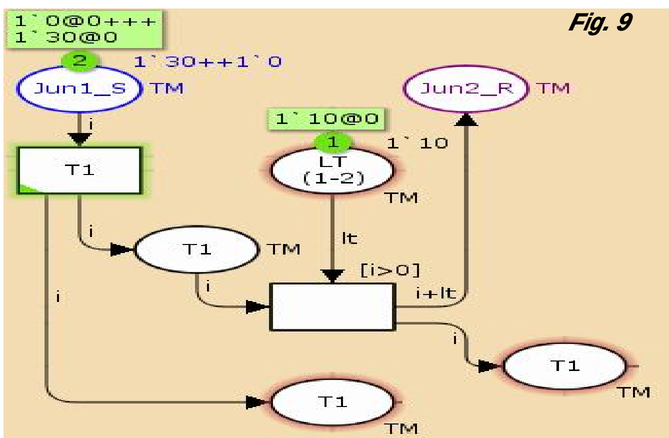


Fig. 9

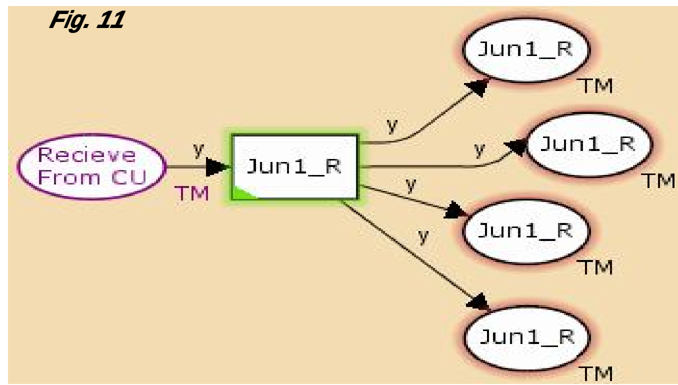


Fig. 11

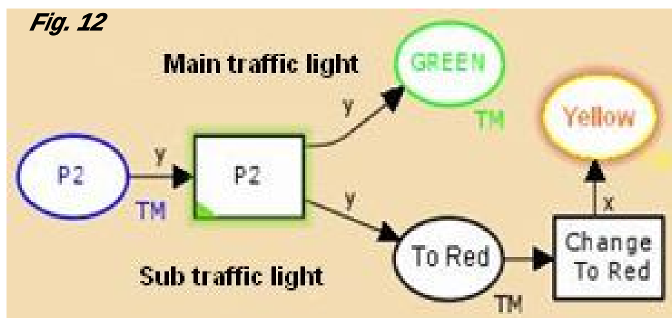


Fig. 12

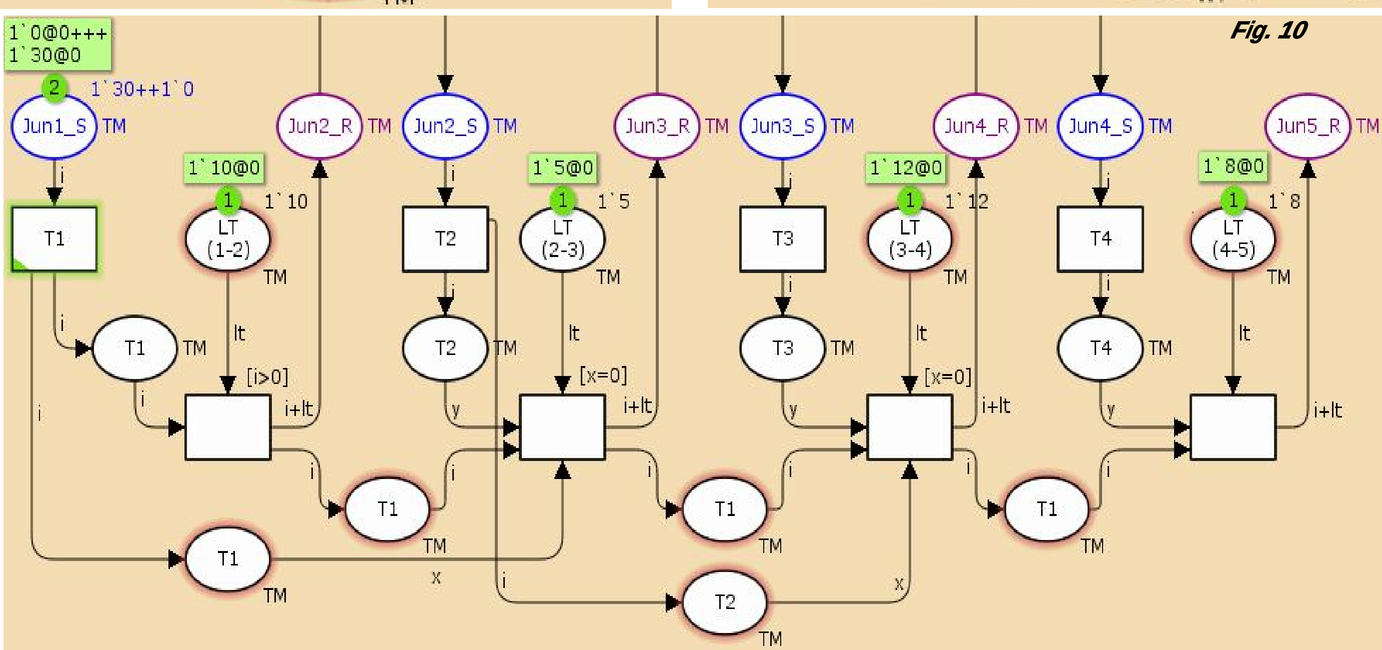


Fig. 10

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

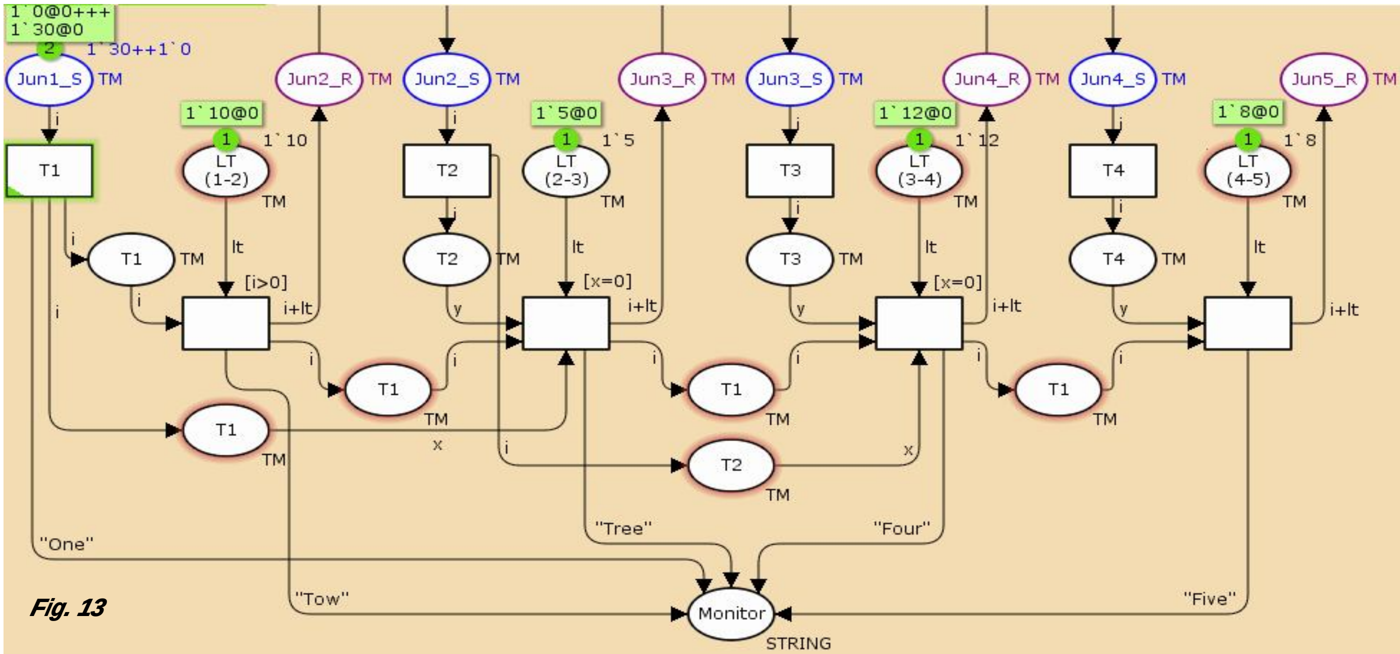


Fig. 13

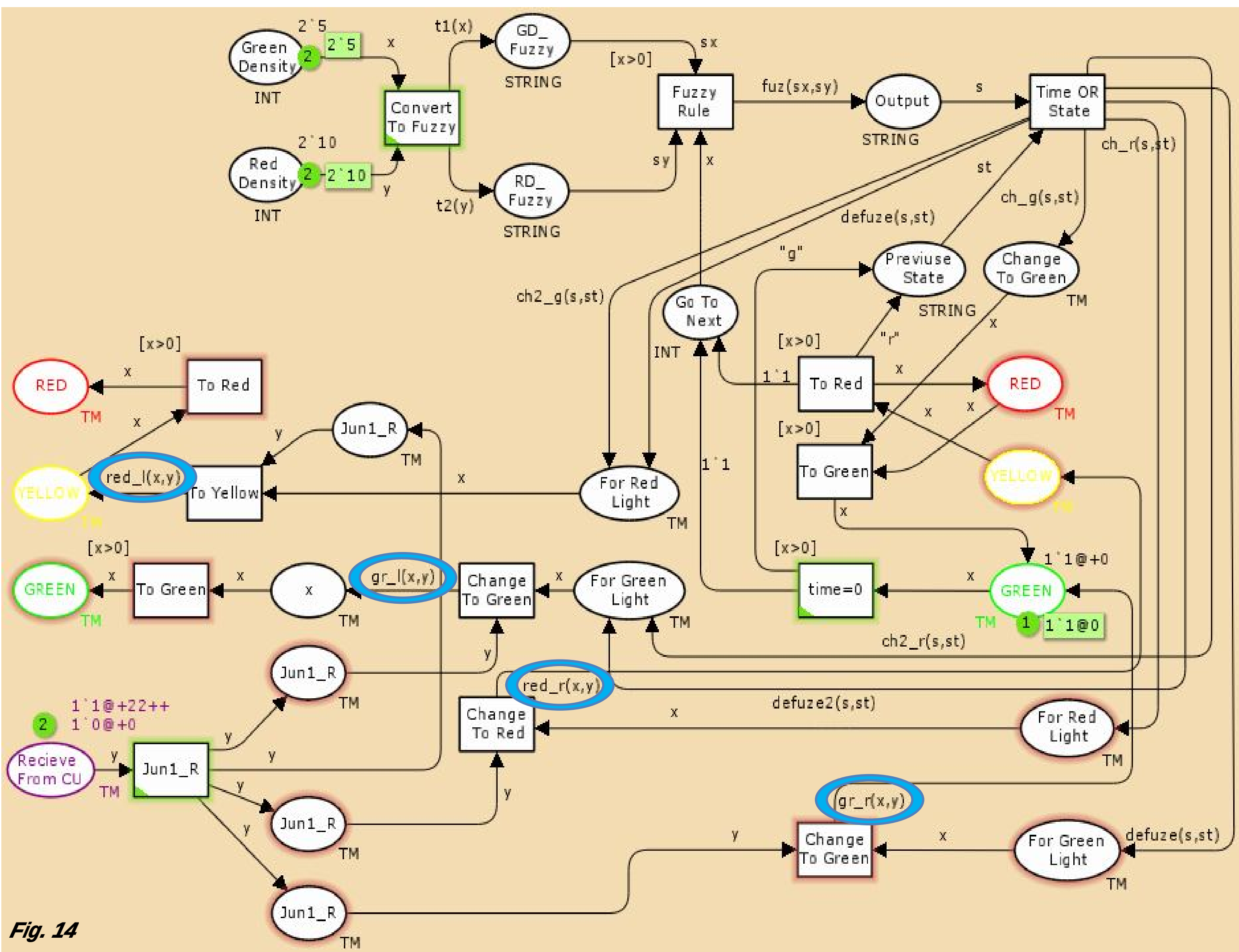


Fig. 14

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

Table 3. Functions table

Function name	Input parameter(s)	Function act
T1()	x	changes integer x to fuzzy value Sx
T2()	y	changes integer y to fuzzy value Sy
Fuzzy()	Sx, Sy	Creates output by receiving fuzzy parameters, according to rules in the fuzzy table.
Ch_g()	S, St	performs the transition of the last condition (red) to green mode on traffic lights.
Ch_r()	S, St	performs the transition of the last condition (green) to red on traffic lights.
Ch2_g()	S, St	acts in reverse of Ch-g function and implements output on other traffic lights.
Ch2_r()	S, St	executes in reverse of Ch-r function and implement output on other traffic light.
Defuze()	S, St	If the output was a time parameter, it denotes its value and will perform on traffic lights.
Defuze2()	S, St	Acts as Defuze function performance and execute on second traffic light.
Red-l ()	x, y	It will be activated through inspecting the received parameter from the control center. If the received parameter was equal to zero, usually the traffic light (the main traffic light) will change to red; otherwise, the changing into red color will be prevented.
Red-r ()	x, y	It will be activated through inspecting the received parameter from the control center. If the amount of received parameter was zero, sub traffic lights change to red in usual conditions; otherwise, on the basis of received time, it will turn to red color.
Gr-r ()	x, y	It will be activated through inspecting the received parameter from the control center. If received parameter was equal to zero, usually traffic light (sub traffic light) will change to green; otherwise the changing into green color will be prevented.
Gr-l ()	x, y	It will be activated through inspecting the received parameter from the control center. If the amount of received parameter was zero, the main traffic light change to green in usual condition; otherwise on the basis of the received time, it will turn to green color.
Test ()	S, St	Inspects the devoted time. If this devoted time was higher than the average level, Tests (S, St) will be sent to it; otherwise, it will be sent to zero value to the control center.

This process will continue until the last traffic light (in the last junction) (Fig.10). In other words, this process will continue until specified cars leave the Main Street. Through connecting all control centers to one place, we can be informed about current conditions of junctions (Fig.13).

Now, we investigate the parameter received from the control center by traffic lights (output parameter of control center). The received parameter must be sent to four independent functions. These functions are activated through inspecting the amount of input parameters and change the common operations of traffic lights (main and sub traffic light) in a junction. If there was not an activity, traffic lights will continue their common operations and perform the regular changes in its conditions. Effective functions are as the following: Red_r(x,y), Red_l(x,y), Gr_r(x,y), Gr_l(x,y) (Fig.11,12, 13, & 14)Function operation of above simulation is as the following (Table 3)

Simulation with programming language of C#

In this section, we have simulated performance of suggested systems with C# programming language with attention to sections 7 and 8.A. CPN Tool uses just four simple symbols for simulating and these simple symbols reduce power of recognizing the large systems, So We by creating and using simple, variant and related graphic pictures, exhibit real sample of traffic situations with smaller scale, to guarantee its practicability within easier understanding of recommended systems. For easy implementation, we include two suggested systems as two separated modes in one program.

First, according to Fig.15, we start to explain mode of suggested system in first phase (Intellectual fuzzy traffic light in on inter section) which is activated by clicking on" Junction" button. By clicking on "Increase" and "Decrease" buttons which are related to places number 1 and 2, number of vehicles in north- south and west- East paths could be adjusted arbitrary (Fig.16). These input values are actually received from camera sensors and process related pictures or received from electromagnetic

Fig. 15

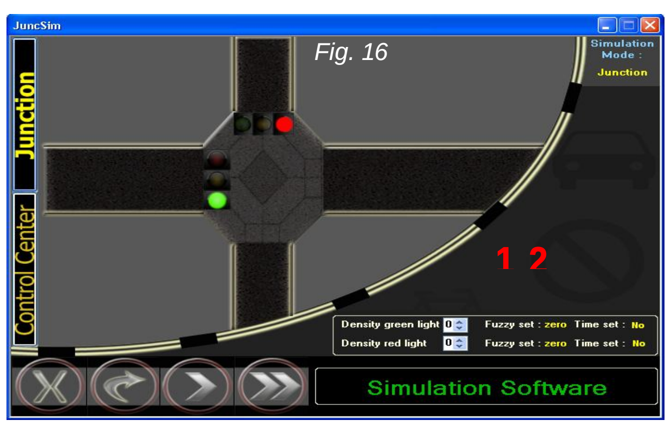


Fig. 16

Indian Journal of Science and Technology Vol. 4 No. 11 (Nov 2011) ISSN: 0974- 6846

sensors.

These input values in program by "Value_to_Fuzzy" function with mentioned code are converted to fuzzy value and became one of member of degrees of very less, less, average, high or zero.

```
mi = 0;
 if (0 < value && value < 2)
   { mt = Convert.ToInt32(0.5 * value); mi = 1; }
 else if (2 < value && value  < 4)
   { mt = Convert.ToInt32(0.5 * (value - 2)); mi = 2; }
 else if (4 < value && value  < 6)
   { mt = Convert.ToInt32(0.5 * (value - 4)); mi = 3; }
 else if (6 < value && value < 8)
   { mt = Convert.ToInt32(0.5 * (value - 6)); mi = 4; }
```

Now we should process two new fuzzy values according to rules of table. Processing input fuzzy values is done by "Rule_Function" function. You observe in below sample code that we save output in " Rule_out" variable .We discuss " Rule_out" Variable in " Fuzzy_to_Time " Function with below code until if output of " Rule_Function " function was based on second, we examin Suitable time inside "Time" Variable according to Fig.4. Function Code is as follows:

```
Random obj_rand = new Random();
 if (mi != 0)
   {
if (rule_out == "vs" || rule_out == "sh" ||
            rule_out ==   "mi" || rule_out == "nu")
        {
          switch (mi)
            {
              case 1: time = obj_rand.Next(Range); break;
              case 2: time = obj_rand.Next(Range); break;
            case 3: time = obj_rand.Next(Range); break;
              case 4: time = obj_rand.Next(Range); break;
            }
          }
        }
else
     {
       switch (rule_out)
       {
          case "vs": time = 10; break;
          case "sh": time = 20; break;
          case "mi": time = 30; break;
          case "nu": time = 40; break;
       }
     }
```

We investigate output of "Rule out" once again to determine situation. It means that maybe output of "Rule Table" function isn't be based on seconds and situation should be changed. In this case, in main path, traffic light goes to yellow mode and then timer object is called in order to after 3 seconds goes to red mode in west-East path and changes to green mode in north-south path and movement turn is saved in "Move line" variable. After determining output situation, we call move - car function in order to set movement of available cars which its code is as follows:

```
if (mode == "junction")
  {
    if (move_line=="main")
      {
        while (P < point)
         {
          if (Car_Ho_G1.Location.X <= point)
          Car_Ho_G1.Location=new Point(x+count1, y);
          if (car_number >= 2)
         {
          if (Car_Ho_G2.Location.X <= point)
          Car_Ho_G2.Location=new Point(x+count1, y);
          }
        }
        count1 = count1 + speed;
        P++;
      }
  }
```

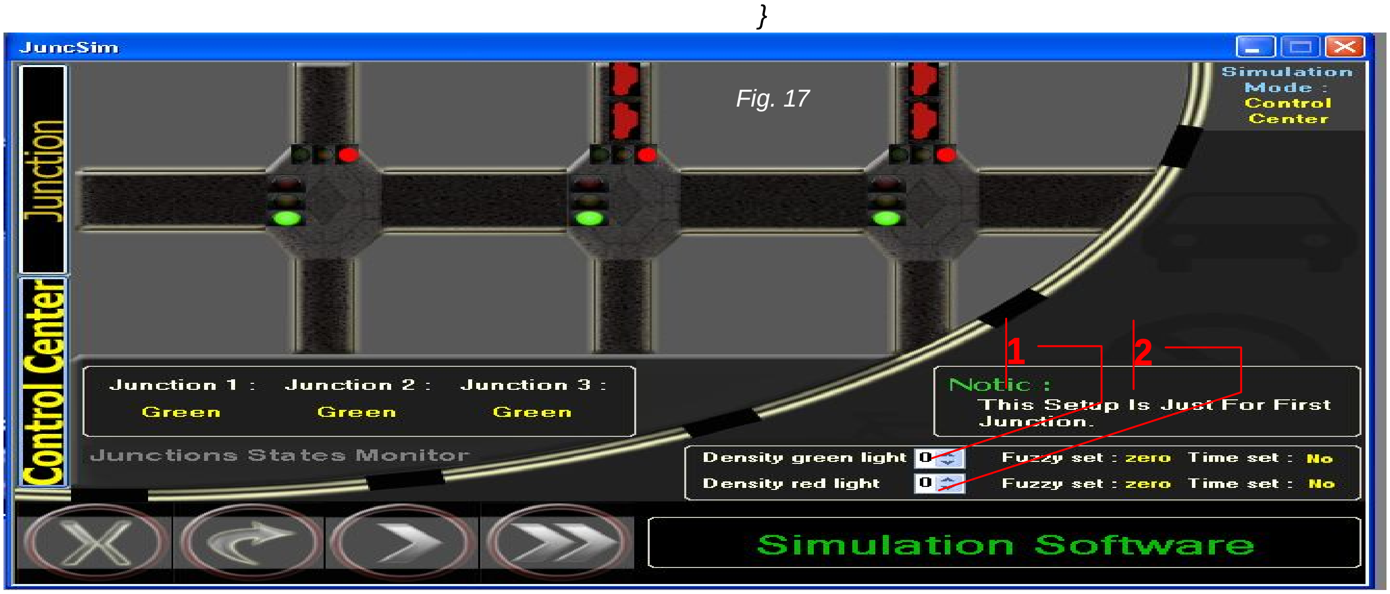


Fig. 17

If movement turn is for main path, like real world, first the cars at the head of the line are moved and then by making safe distance while moving, let next lines cars to move , and this manner is continued until last car in the last line receives to destination. If it is secondary path turn, again cars at the first line are moved and by creating safe distance in secondary path , while moving let cars in next lines to move.

Note to Two points is worthy: 1) If "Rule_Out" variable includes time, then number of vehicles which are permitted to move will be saved in "Rule_Out" variable as limited to time and if time be expired, reminded vehicles are moved in other paths based on their related time. 2) If "rule out" variable doesn't have time value, after moving available cars in current path, according to rules table, due to lack of vehicles in other path, traffic light doesn't change (Fig.17).

Now we engage to explain second phase simulation, *i.e.* control center, by using language programming C#. This simulation mode is started by clicking on “control center” button. In this mode of simulation, objects number 1 and 2 are determined number of vehicles respectively for main and secondary paths in first crossroad (Fig.16). For second and third crossroads, by default we put three vehicles in each secondary paths to complete our simulation. Performance of all intersections is based on rules table. Movement of vehicles in paths and crossroads is based on simulated mode in previous section and defined functions in it.

Note that in this simulation program, if number of vehicles in main path be four or more, first intersection sends a signal for control center which involve a time more than average time, and control center in turn, after receiving this parameter, sends a signal for next intersection of path to change situation of main path to green mode and now is not allowed to change situations to other modes. Vehicles are passed from intersections in amount of available time in "Rule_out" variable. Then, if there are vehicles in secondary path, traffic light will change its mode, and by sending a signal inform control unit too. This trend is also repeated for second and third crossroads until vehicles in main path pass from successive crossroads and arrive to final destination. If number of vehicles in main path aren't four or more, each traffic light act independently and intellectually in regard of its intersection and still sends its signal to control center continuously , but until there is no heavy traffic in main center, control center doesn't involve in traffic light function.

Conclusion

Intelligent fuzzy traffic light systems with applying human logic and argument, present better performance than conventional traffic light (constant-time).

In fact, this new generation of traffic signals are to replace the conventional traffic lights, cause constant time traffic signals, regardless to vehicles congestion changes, just count time interval which is set on them during built or installation. While intellectual fuzzy traffic signals include small processor which assigns time for passing vehicles on the basis of measuring vehicle congestion in each path and processing them, either in special situations apply fast changing or not changing situation on traffic signal.

Of course, we could avoid of infinite counting problem for one path or infinite stop problem in other path with limiting time in minimum and maximum interval. Note that due to non-constant time, there is no need to elongation time. We find usefulness and logical performance and even necessity of intelligent fuzzy traffic signal in each inter junction. But we by studying many streets in the basis of traffic congestion, mentioned subtle and important point about controlling crossroads. This performance will be suitable for streets with low or even average traffic load , but in crowded streets or in rush hours that load of traffic is high, independent function of each traffic signal could be problematic. So a control center is necessary for controlling and managing situation of crossroads, and this center should be activated just in times that an unordinary congestion is observed in main path. We first simulated mentioned systems with CPN tool. Then we verify performance of systems in different conditions by entering different input parameters and based on real sample of vehicle congestion in streets. So we simulated various modes and conditions of an intersection and a street traffic with C# programming language, while presented better and more perfect graphic. We after different test found that recommended system of intelligent fuzzy traffic signal with minimizing wasting time.

Implements faster and more reasonable change and causes justify performance. Also, control center of suggested system with prioritizing paths and defining level of traffic, causes flowing traffic and avoids locking traffic in main path. Finally, we present mentioned system hoping that reduces biological and sonic pollution and increases better and more comfortable time to live.

References

1. Barzegar, Davoudpour M, Meybodi MR, Sadeghian A and Tirandazian M (2011) Formalized learning automata with adaptive fuzzy Colored Petri net an application specific to managing traffic signals. *Scientia Iranica.* pp: 1-12.
2. Kok Khiang Tan, Marzuki Khalid and Rubiyah Yusof (1996) Intelligent traffic lights control by fuzzy logic. *Malaysian J. Compu. Sci.* pp: 29-35.
3. Shilpa Mehta (2008) Fuzzy control system for controlling traffic lights. *Hong Kong IMECS*. pp: 19-21.