VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

Review Report Soft Computing

TRAFFIC CONTROL USING FUZZY LOGIC

Reg No:	Name:
20BIT0095	Routhu sai praneeth

Winter semester 2021-22

Soft Computing – ITE1015

School of Information Technology and Engineering

Fall Semester 2021-22

- **Faculty Name:** Balakrushna Tripathy Sir
- **Group Representative Mail ID:-** routhusai.praneeth2020@vitstudent.ac.in

- **Aim:-**

To Design Traffic Signals Using Fuzzy Logic.

- **Objective:-**

Requirement for optimizing traffic control methods for better accommodating the increasing demand.

- **Abstract:-**

` Vehicular traveling is increasing throughout the world, particularly in large urban areas. By the increasing use of automobiles in cities, traffic congestion occurs. In our day-to-day life, the number of vehicles like motorcycle, car, bus, van and truck etc. are increasing very rapidly. Traffic is the chief puzzle problem which every country faces because of the enhancement in number of vehicles throughout the world, especially in large urban towns. Hence the need arises for simulating and optimizing traffic control algorithms to better accommodate this increasing demand. As a result, outcome of traffic blocking and enormous problems in metro cities. Due to traffic jamming problem some large issues are found such as pollution, wastage of time, fuel, and accidents. Traffic jamming arises terribly often time switch in the contemporary world that has an effect on the means of life and introduces such a large number of issues and challenge. Thus, there is a requirement for optimizing traffic control methods for better accommodating the increasing demand. To resolve this problem by using traffic signal timing optimization is the better solution using fuzzy logic technique an intelligent traffic signal control is introduced. The existing traffic light controllers would not give a best solution to this problem as it changes traffic lights based on constant cycle time. Fuzzy logic controllers can be used to deal with linguistic and unpredictable traffic data to control the signal timings. This research presents an application of fuzzy logic for two stage traffic light system to overcome problems like congestion, accidents, speed, and traffic irregularity.

➢ Introduction:-

Concept of the application of fuzzy logic has been very active in the field of computer research. It has been applied in various applications such as military, medical, industrial, etc., which are unique. It has abilities to compare the system and automate the environment changes. Fuzzy logic has been distinguished in the computer industry as similar logic to that of humans. It can be seen that the fuzzy logic has led to applications such as traffic management, the technique was used to develop a system for monitoring and blocking traffic to solve the problem of traffic congestion. Fuzzy logic has been widely used to develop a traffic signal controller because it allows qualitative modelling of complex systems. Traffic congestion leads to losses in productivity, fuel wastage, environment degradation, reducing the usage of roads and related crashes. Increase in the usage of private vehicles, ineffective traffic management and insufficient roads increase traffic congestion exponentially in urban areas. Out of the above major three factors, efficient, effective traffic management can be improved by using the advancement of technology. Traffic management in road intersections where the traffic signals are installed is crucial. Traffic congestion in this area can be reduced by effectively utilizing the same resource.

➢ Problem definition:

The main goal of soft computing is to develop intelligent machines and to solve nonlinear and mathematically un-modelled problems. This paper is aimed at solving identified problems of traffic congestion at 4-way intersections by developing an intelligent traffic congestion control system based on fuzzy logic model which mimics human reasoning. Fuzzy logic models are suitable for controlling intersections, especially those with heavy traffic, because it is able to emulate the control logic of traffic police officers who sometimes replaces traffic signal control when the intersection is congested.

➢ SOFTWARE Requirements:

Software used: MATLAB (Fuzzy Tool Box) Compatibility: ➢ Operating System: All operating systems.

➢ Hardware Requirements:

- ➢ Minimum RAM: 1 GB 19
- ➢ Hard Disk: 40 GB
- ➢ Processor: Intel Pentium 4 /AMD Processor.

➢ Literature Review:

REVIEW ON VARIOUS SCHEMES:

- **A Traffic Lights Control System Based on Fuzzy Logic (2014):**

This research presents an application of fuzzy logic for multi-agent based autonomous traffic lights control system using wireless sensors to overcome problems like congestion, accidents, speed, and traffic irregularity. The real time parameters such as traffic density and queue length are obtained by using image-processing techniques. Thus, On and Off timings for the green, red and or amber lights are adjusted to the actual road conditions. This paper describes a fuzzy logic signal controller for a four– way intersection suitable for mixed traffic, including a high proportion of motorcycles. The proposed agent-based approach can provide a preferred solution by minimizing the vehicles' waiting time especially the emergency vehicles using fuzzy logic control under the situations that normally occur during emergencies. In this paper, they discussed the implementation of an intelligent traffic lights control system using fuzzy logic technology. Software based on Visual Basic has been developed to simulate an isolated traffic junction. The rules and membership functions of the fuzzy logic controller can be selected and be changed and their outputs can be compared in terms of several different representations. The software is graphical in nature and runs under the Windows environment.

- **Design and Simulation of Adaptive Traffic Light Controller Using Fuzzy Logic Control Sugeno Method (2015):**

This paper presents the design of an adaptive traffic light controller using fuzzy logic control Sugeno Method. This fuzzy logic control is used to determine the length of green time at an intersection. The purpose of this paper is to design an adaptive traffic light controller with three inputs, namely the number of queues, waiting time, and the traffic flow of vehicles. The design was applied in a simulation to observe the number of queues, waiting time, and the number of vehicles passing an intersection. The simulation results show that the traffic light using fuzzy logic control performs better than using fixed time control. The number of queues and waiting time are lower, and the number of departures is higher than using the fixed time controller for a traffic light.

- **Design and Analysis of a Two Stage Traffic Light System Using Fuzzy Logic (2015):**

In this paper, a two-stage traffic light system for real-time traffic monitoring has been proposed to dynamically manage both the phase and green time of traffic lights for an isolated signalized intersection with the objective of minimizing the average vehicle delay in different traffic flow rate. There are two different modules namely traffic urgency decision module (TUDM) and extension time decision module (ETDM). In the first stage TUDM, calculates urgency for all red phases. On the bases of urgency degree, proposed system selects the red-light phase with large traffic urgency as the next phase to switch. In second stage ETDM, calculates green light time i.e., extension time of the phase which has higher urgency according to the number of vehicles. Software has been developed in MATLAB to simulate the situation of an isolated signalized intersection based on fuzzy logic. Simulation results verify the performance of our proposed two stage traffic light system using fuzzy logic.

- **Traffic system using fuzzy logic (2018):**

This paper is all about to introduce a dynamic traffic system that evaluates the real time traffic parameter like density of vehicles to design the signal time. They have implemented this using fuzzy logic for better efficiency. Depending on the real time parameters of traffic like density, they designed a system which uses the green signal depending on the number of vehicles in that particular lane. They proposed Fuzzy logic-based traffic control system in a 4-lane road. Signal improvement is one of the most cost-effective method to reduce the problem of traffic congestion. Finally, a simulation and model-based design environment for dynamic and embedded systems, integrated with MATLAB. Simulink is a data flow graphical programming language tool for modeling, simulating and analyzing multi-domain dynamic systems. It is basically a graphical block diagramming tool with customizable set of block libraries. They even displayed tbhe overall process using hardware resources like Arduino sensors, potentiometer, FIS tool box and LCD to get the proper results.

- **Effective Lane Management for Emergency Vehicles and Adaptive Signaling for Dynamic Traffic Congestion Using Fuzzy Logic (2019):**

The proposed idea is designed for effective lane management system using fuzzy logic and heuristic value. Furthermore, emergency vehicle is managed effectively based on prioritization. The simulation result shows that the proposed system significantly reduces the average waiting time of the emergency vehicles. It increases the chance of emergency vehicle reaching destination on time by effective prioritization.

➢ COMPARATIVE STUDY ON VARIOUS SUBTITLES:

1. Title:

A Traffic Lights Control System Based on Fuzzy Logic (2014).

Author:

Mojtaba Salehi , Iman Sepahvand, Mohammad Yarahmadi 2014.

Technologies Used:

Fuzzy logic technology. Software based on Visual Basic is used here to simulate an isolated traffic junction.

Advantages:

The system proposed here is very flexible. The feedback of the queue length and traffic densities can be taken from images taken from cameras. it can be used advantageously y for traffic light controlling systems. The definitions of the fuzzy sets of the antecedents are also very easily changeable. This will be highly useful in traffic control in the today congestion traffic.

Issues:

A lot of research work has to be done to verify the expected features by simulation. The performance of the fuzzy logic system is affected by the configuration of the membership functions of the input and output variables and the rule’s base.

Metrics Used:

Traffic density queue length real-time strategies, fixed-time strategies, distance velocity number of vehicles, input member functions, output member functions.

2. Title:

An Application of Fuzzy Logic Model in Solving Road Traffic Congestion.

Author:

C. Ugwu,Bale, Dennis 2014

Technologies Used:

Fuzzy Logic, Fuzzification, Defuzzification

Advantages:

The research has shown that a sensor based fuzzy logic model has the potential of eliminating the road traffic congestion to barest minimum. This was achieved by its ability to take decisions whether to extend or terminate the current green light time based on a set of fuzzy rules and Real time traffic information.

Issues:

A state can be skipped if there are no vehicle queues for the corresponding approach.

Metrics Used:

Fuzzy rules and Realtime traffic information.

3. Title:

A Design Of Fuzzy Logic Traffic Controller for Isolated Intersections with Emergency Vehicle Priority System Using MATLAB Simulation.

Author:

Mohit JhaShailja Shukla2014

Technologies Used:

Queuing theory model of multiple– input single-output. SIMULINK model and SimEvent toolbox in MATLAB. fuzzy inference system method in MATLAB.

Advantages:

It can recognize and adjudicate the fuzzy phenomenon, and control the system effectively. This system is also works well intelligently for an emergency vehicle case. It avoids the vehicles waiting in crossing, improves the intersection vehicle crossing capacity and realizes the intelligent control of traffic light.

Issues:

We have to use an "embedded MATLAB function block" which passes an emergency vehicle queue length and their waiting time using ‘C’ coding which is not much effective.

Metrics Used:

Vehicle queue length, waiting time Poisson distribution function.

4. Title:

Design and Simulation of Adaptive Traffic Light Controller Using Fuzzy Logic Control Sugeno Method (2015)

Author:

Mohit JhaShailja Shukla2014

Technologies Used:

Manual Kapasitas Jalan Indonesia (MKJI) (calculation of the cycle time) FUZZY INFERENCE SYSTEM SUGENO METHOD Additive Method Probabilistic OR Method Max Method.

Advantages:

The performance of the system is improved in results of waiting time, number of queue and number of departures. Comparison of average waiting time.

Issues:

It is totally based on time of waiting but not the density of the traffic.

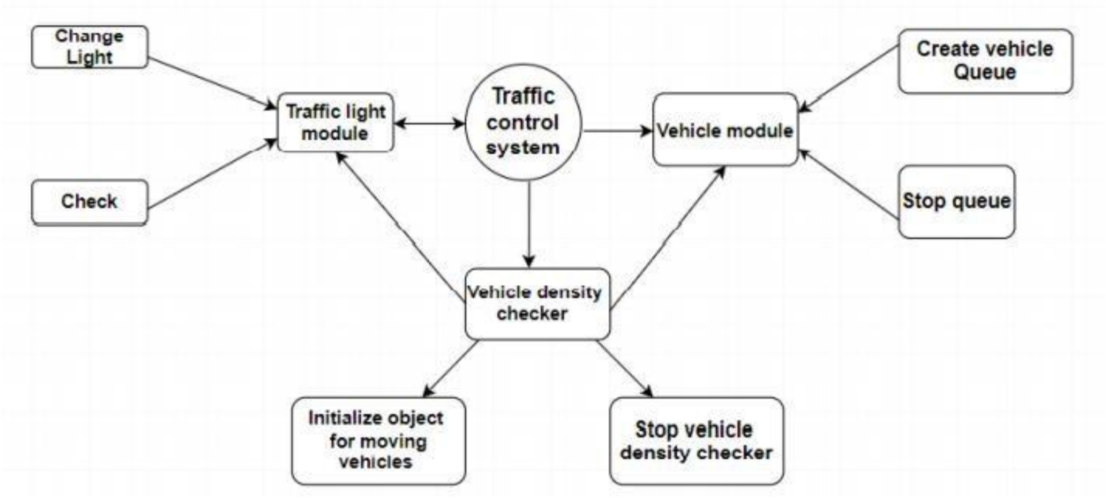
Metrics Used:

Waiting Time (WT), Number of Queue (NQ) and Number of Departure (NP).

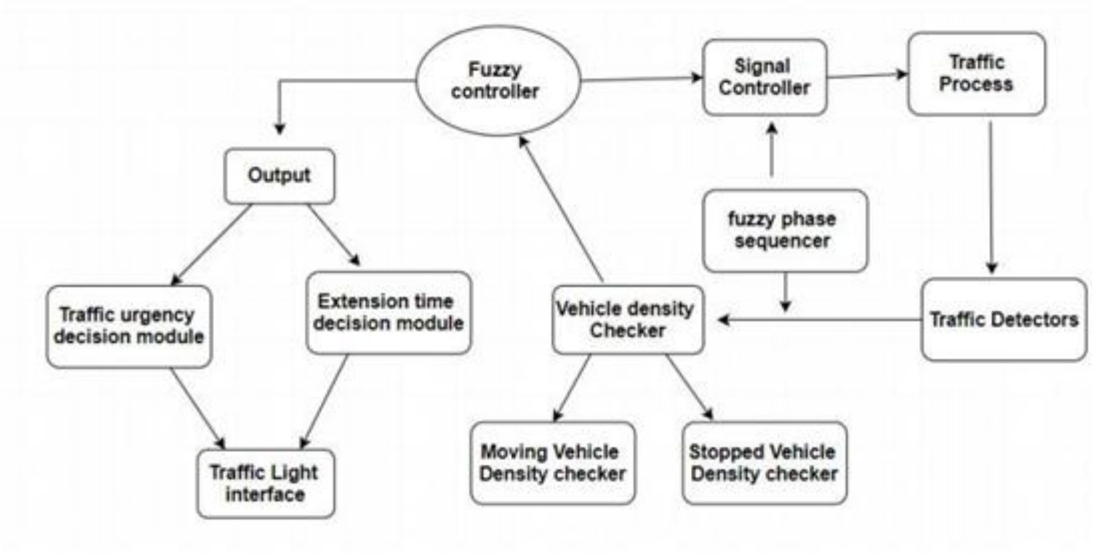
➢ Gaps in Literature:

This study demonstrates the formation of a traffic network. The selection of the Highway Engineering subway road used in the study is divided into two types: Arterial Street and Collector Road, which is the next step to integrate the use of Fuzzy Logic in robot judgment, according to the number of channels. Each type of road traffic system is VISSIM and host with complete simulation delays. The simulations of the VISSIM system will have the right to change the type of output lights and the appropriate model for each type of road to reduce traffic congestion.

➢ Work Flow diagram: -



➢ Architecture of the system:



➢ Component Modules:

- Vehicle arrival distribution module
- Vehicle delay computation module
- Vehicle density
- Traffic Urgency Decision Module
- Extension Time Decision Module
- Traffic signal controller

➢ Modules Description:

Vehicle arrival distribution module

In this First-In-First-Out (FIFO) principle is applied to the vehicles queue. The vehicles are known as customers while services time is the time for the vehicles to depart and to cross the intersection. The arrival of the vehicles in a Poisson process and the time period in order to reach the number of vehicles follows Poisson distribution. This is the input module for the system.

vehicle delay computation module

Here the performance evaluation for traffic signal control of intersection is done. If the value of average vehicle delay is small then the traffic signal control effect is better. The amount of queuing vehicles in red light phase at time) can be calculated. Here the processing of input information takes place using fuzzy logic.

Vehicle density

Vehicle density can be expressed as k = n 5280 / l

where k = vehicle density

n = no. of vehicles occupying a length l of the road

l = the length of road occupied by the vehicles

Traffic Urgency Decision Module

It calculates urgency for all red phases. On the basis of urgency degree, it selects the next red-light phase to switch. There are two input variables namely Queue, Timer and one output variable namely Urgency. The input variable Queue counts number of vehicles of current redlight phase and variable Timer count the duration of red light since the last end of the green light. The output variable Urgency is traffic urgency of red-light phases.

Extension Time Decision Module

It calculates green light time i.e., Extension time of the phase which has higher urgency can be calculated according to the number of vehicles The two input variables namely QueueLane 1, Queue-Lane-2 and one output variable Extension-Time. The input variable QueueLane1 count the number of vehicles of lane1 and input variable Queue-Lane2 count number of vehicles of other side of red-light phase which has big traffic urgency. The output variable Extension-Time is the extension time of current green light phase.

Traffic signal controller

The main objective of the traffic signal controller is to provide the sophisticated control and coordination on traffic. It changes the green light from street 1 to street 2 so the vehicle can go through the traffic and it changes light when the street is clear.

- **Language Used:-**

MATLAB Simulink is a data flow graphical programming language tool for modeling, simulating and analysing multi-domain dynamic systems.

➢ Algorithm:-

Algorithm- Fuzzy Logic

This paper aims at developing an intelligent traffic light controller using fuzzy logic technique and illustrated a control procedure using if – then rules.

In artificial intelligence (AI) systems, fuzzy logic is used to imitate human reasoning and cognition. Rather than strictly binary cases of truth, fuzzy logic includes 0 and 1 as extreme cases of truth but with various intermediate degrees of truth.

➢ Algorithm Steps:

In the Recsize.m code is used for create a rectangle box with four entries that makes the number of vehicles that enter the street. Four sides Rectangle and the Design of virtual road lanes.

In the Reccolour.m code gives the colour red and green to the number of vehicles that enter to the street and makes decision in which street that vehicles that will cross traffic. And also, this is also implemented by if then statements. Text box for indicating the vehicles number also implemented here.

In fuzzygt.fis contains the member ship functions and rules and makes the fuzzy design using mamdani. This fuzzy logic also implemented here.

In the fuzzyselect.fis contains the member ship functions and rules and makes the fuzzy design using mamdani.

In this screenshot the test.m contains the random function which will generate the number of vehicles that enter into the traffic and the sum of vehicles that crosses each lane combine all other files and gives the output.

➢ The Dataset:-

There is no dataset for this system the input is given in the code itself by giving number of streets and vehicles are enter into the street.

Sample Code:

Recsize.m

```
function recsize(X1, Y1, figure1)

% Create axes
axes1 = axes('Parent',figure1,...
'Position',[0 0 1 1]);
xlim([0 3.5])
ylim([0 3.5])
hold(axes1,'all');

% Create rectangle
annotation(figure1,'rectangle',...
[4/7 0 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);

annotation(figure1,'rectangle',...
[4/7 4/7 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);

annotation(figure1,'rectangle',...
[0 4/7 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);

annotation(figure1,'rectangle',...
[0 0 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);

hold off
```

Reccolour.m

```
function recsize(x,y,figure1)
red=[1 0 0];
green=[0 1 0];
yellow=[1 .8 0];

% drawnow
if y(1)==1
light1=green;
elseif y(1)==0.5
light1=yellow;
```

```matlab
else
light1=red;
end
if y(2)==1
light2=green;
elseif y(2)==0.5
light2=yellow;
else
light2=red;
end
if y(3)==1
light3=green;
elseif y(3)==0.5
light3=yellow;
else
light3=red;
end
if y(4)==1
light4=green;
elseif y(4)==0.5
light4=yellow;
else
light4=red;
end

annotation(figure1,'textbox',...
[0.38 0.46 0.03 0.067],...
'String',{x(1)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light1);

annotation(figure1,'textbox',...
[0.48 0.35 0.03 0.067],...
'String',{x(2)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light2);

annotation(figure1,'textbox',...
[0.58 0.46 0.03 0.067],...
'String',{x(3)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light3);

annotation(figure1,'textbox',...
[0.48 0.57 0.03 0.067],...
'String',{x(4)},...
'FitBoxToText','on','FontWeight','bold',...
'EdgeColor','none','BackgroundColor',[1 1 1],'Color',light4);
```

Fuzzygt.fis

```
[System]
Name='fuzzygreentime'
Type='mamdani'
Version=2.0
NumInputs=4
NumOutputs=1
NumRules=81
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='bisector'

[Input1]
Name='street1'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[-24 0 20]
MF2='medium':'trimf',[15 30 45]
MF3='high':'trimf',[40 60 80]
[Input2]
Name='street2'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[-24 0 20]
MF2='medium':'trimf',[15 30.2 45]
MF3='high':'trimf',[40 60 80]
[Input3]
Name='street3'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[-24 0 20]
MF2='medim':'trimf',[15 30 45]
MF3='high':'trimf',[40 60 80]
[Input4]
Name='street4'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[-24 0 20]
MF2='medium':'trimf',[15 30 45]
MF3='high':'trimf',[40 60 80]
[Output1]
Name='green-time'
Range=[0 8]
NumMFs=3
MF1='low':'trimf',[0 1 2]
MF2='medium':'trimf',[3 4 5]
MF3='high':'trimf',[6 7 8]
[Rules]
1 1 1 1, 3 (1) : 1
1 1 1 2, 3 (1) : 1
1 1 1 3, 3 (1) : 1
1 1 2 1, 3 (1) : 1
```

```
1 1 2 2, 2 (1) : 1
1 1 2 3, 3 (1) : 1
1 1 3 1, 3 (1) : 1
1 1 3 2, 3 (1) : 1
1 1 3 3, 2 (1) : 1
1 2 1 1, 3 (1) : 1
1 2 1 2, 2 (1) : 1
1 2 1 3, 3 (1) : 1
1 2 2 1, 2 (1) : 1
1 2 2 2, 2 (1) : 1
1 2 2 3, 1 (1) : 1
1 2 3 1, 3 (1) : 1
1 2 3 2, 1 (1) : 1
1 2 3 3, 1 (1) : 1
1 3 1 1, 3 (1) : 1
1 3 1 2, 3 (1) : 1
1 3 1 3, 2 (1) : 1
1 3 2 1, 3 (1) : 1
1 3 2 2, 1 (1) : 1
1 3 2 3, 1 (1) : 1
1 3 3 1, 1 (1) : 1
1 3 3 2, 1 (1) : 1
1 3 3 3, 1 (1) : 1
2 1 1 1, 3 (1) : 1
2 1 1 2, 2 (1) : 1
2 1 1 3, 3 (1) : 1
2 1 2 1, 2 (1) : 1
2 1 2 2, 1 (1) : 1
2 1 2 3, 1 (1) : 1
2 1 3 1, 2 (1) : 1
2 1 3 2, 1 (1) : 1
2 1 3 3, 1 (1) : 1
2 2 1 1, 2 (1) : 1
2 2 1 2, 2 (1) : 1
2 2 1 3, 1 (1) : 1
2 2 2 1, 2 (1) : 1
2 2 2 2, 1 (1) : 1
2 2 2 3, 1 (1) : 1
2 2 3 1, 1 (1) : 1
2 2 3 2, 1 (1) : 1
2 2 3 3, 1 (1) : 1
2 3 1 1, 1 (1) : 1
2 3 1 2, 1 (1) : 1
2 3 1 3, 1 (1) : 1
2 3 2 1, 1 (1) : 1
2 3 2 2, 1 (1) : 1
2 3 2 3, 1 (1) : 1
2 3 3 1, 1 (1) : 1
2 3 3 2, 1 (1) : 1
2 3 3 3, 1 (1) : 1
3 1 1 1, 3 (1) : 1
3 1 1 2, 2 (1) : 1
3 1 1 3, 1 (1) : 1
3 1 2 1, 2 (1) : 1
3 1 2 2, 1 (1) : 1
3 1 2 3, 1 (1) : 1
3 1 3 1, 1 (1) : 1
3 1 3 2, 1 (1) : 1
```

```
3 1 3 3, 1 (1) : 1
3 2 1 1, 2 (1) : 1
3 2 1 2, 1 (1) : 1
3 2 1 3, 1 (1) : 1
3 2 2 1, 1 (1) : 1
3 2 2 2, 1 (1) : 1
3 2 2 3, 1 (1) : 1
3 2 3 1, 1 (1) : 1
3 2 3 2, 1 (1) : 1
3 2 3 3, 1 (1) : 1
3 3 1 1, 1 (1) : 1
3 3 1 2, 1 (1) : 1
3 3 1 3, 1 (1) : 1
3 3 2 1, 1 (1) : 1
3 3 2 2, 1 (1) : 1
3 3 2 3, 1 (1) : 1
3 3 3 1, 1 (1) : 1
3 3 3 2, 1 (1) : 1
3 3 3 3, 1 (1) : 1
```

Fuzzyselection.fis

```
[System]
Name='fuzzyselect'
Type='mamdani'
Version=2.0
NumInputs=4
NumOutputs=1
NumRules=81
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='bisector'

[Input1]
Name='street1'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[0 13 22]
MF2='medium':'trimf',[10 30 42]
MF3='high':'trimf',[38 60 60]

[Input2]
Name='street2'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[0 13 22]
MF2='medium':'trimf',[10 30 42]
MF3='high':'trimf',[38 60 60]
```

```
[Input3]
Name='street3'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[0 13 22]
MF2='medim':'trimf',[10 30 42]
MF3='high':'trimf',[38 60 60]

[Input4]
Name='street4'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[0 13 22]
MF2='medium':'trimf',[10 30 42]
MF3='high':'trimf',[38 60 60]

[Output1]
Name='witch-street'
Range=[0 14]
NumMFs=5
MF1='Lstreet1':'trimf',[0 1 2]
MF2='Lstreet2':'trimf',[3 4 5]
MF3='Lstreet3':'trimf',[6 7 8]
MF4='Lstreet4':'trimf',[9 10 11]
MF5='no-differ':'trimf',[12 13 14]
[Rules]
1 1 1 1, 5 (1) : 1
1 1 1 2, 4 (1) : 1
1 1 1 3, 4 (1) : 1
1 1 2 1, 3 (1) : 1
1 1 2 2, 5 (1) : 1
1 1 2 3, 4 (1) : 1
1 1 3 1, 3 (1) : 1
1 1 3 2, 3 (1) : 1
1 1 3 3, 5 (1) : 1
1 2 1 1, 2 (1) : 1
1 2 1 2, 5 (1) : 1
1 2 1 3, 4 (1) : 1
1 2 2 1, 5 (1) : 1
1 2 2 2, 5 (1) : 1
1 2 2 3, 4 (1) : 1
1 2 3 1, 3 (1) : 1
1 2 3 2, 3 (1) : 1
1 2 3 3, 5 (1) : 1
1 3 1 1, 2 (1) : 1
1 3 1 2, 2 (1) : 1
1 3 1 3, 5 (1) : 1
1 3 2 1, 2 (1) : 1
1 3 2 2, 2 (1) : 1
1 3 2 3, 5 (1) : 1
1 3 3 1, 5 (1) : 1
1 3 3 2, 5 (1) : 1
```

```
1 3 3 3, 5 (1) : 1
2 1 1 1, 1 (1) : 1
2 1 1 2, 5 (1) : 1
2 1 1 3, 4 (1) : 1
2 1 2 1, 5 (1) : 1
2 1 2 2, 5 (1) : 1
2 1 2 3, 4 (1) : 1
2 1 3 1, 3 (1) : 1
2 1 3 2, 3 (1) : 1
2 1 3 3, 5 (1) : 1
2 2 1 1, 5 (1) : 1
2 2 1 2, 5 (1) : 1
2 2 1 3, 4 (1) : 1
2 2 2 1, 5 (1) : 1
2 2 2 2, 5 (1) : 1
2 2 2 3, 4 (1) : 1
2 2 3 1, 3 (1) : 1
2 2 3 2, 3 (1) : 1
2 2 3 3, 5 (1) : 1
2 3 1 1, 2 (1) : 1
2 3 1 2, 2 (1) : 1
2 3 1 3, 5 (1) : 1
2 3 2 1, 2 (1) : 1
2 3 2 2, 2 (1) : 1
2 3 2 3, 5 (1) : 1
2 3 3 1, 5 (1) : 1
2 3 3 2, 5 (1) : 1
2 3 3 3, 5 (1) : 1
3 1 1 1, 1 (1) : 1
3 1 1 2, 1 (1) : 1
3 1 1 3, 5 (1) : 1
3 1 2 1, 1 (1) : 1
3 1 2 2, 1 (1) : 1
3 1 2 3, 5 (1) : 1
3 1 3 1, 5 (1) : 1
3 1 3 2, 5 (1) : 1
3 1 3 3, 5 (1) : 1
3 2 1 1, 1 (1) : 1
3 2 1 2, 1 (1) : 1
3 2 1 3, 5 (1) : 1
3 2 2 1, 1 (1) : 1
3 2 2 2, 1 (1) : 1
3 2 2 3, 5 (1) : 1
3 2 3 1, 5 (1) : 1
3 2 3 2, 5 (1) : 1
3 2 3 3, 5 (1) : 1
3 3 1 1, 5 (1) : 1
3 3 1 2, 5 (1) : 1
3 3 1 3, 5 (1) : 1
3 3 2 1, 5 (1) : 1
3 3 2 2, 5 (1) : 1
3 3 2 3, 5 (1) : 1
```

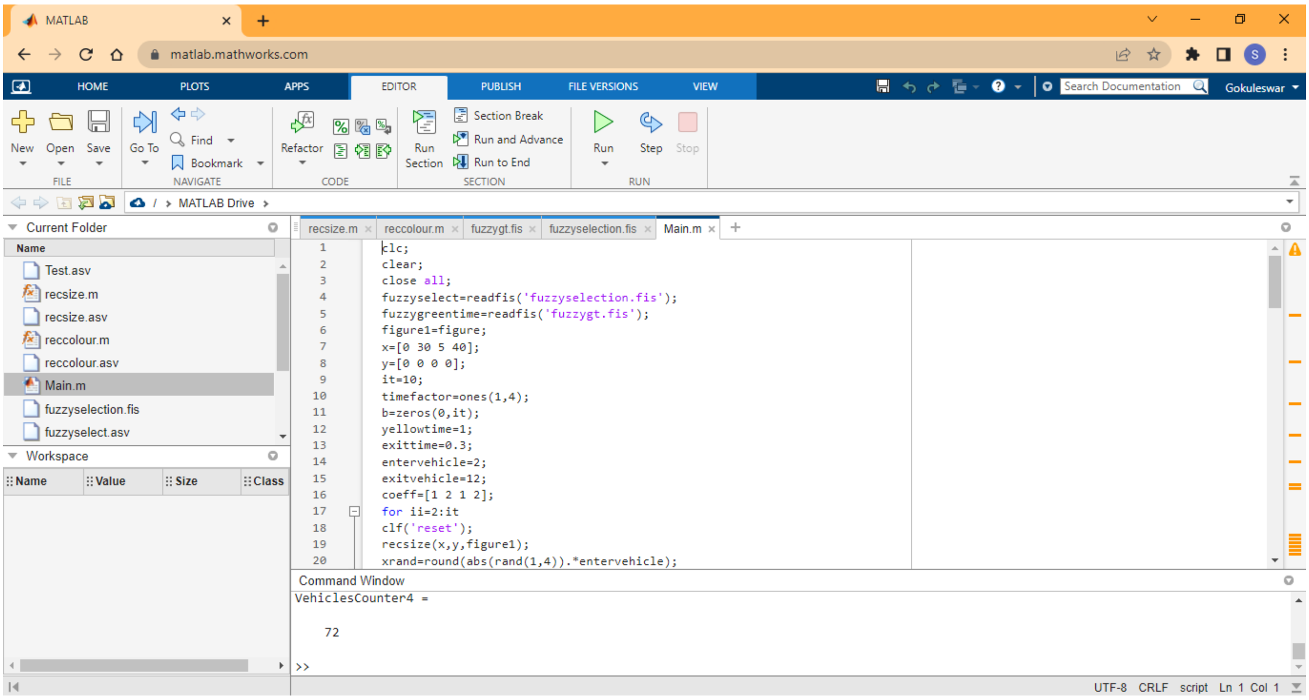
Main.m

```
clc;
clear;
close all;
fuzzyselect=readfis('fuzzyselection.fis');
fuzzygreentime=readfis('fuzzygt.fis');
figure1=figure;
x=[0 30 5 40];
y=[0 0 0 0];
it=10;
timefactor=ones(1,4);
b=zeros(0,it);
yellowtime=1;
exittime=0.3;
entervehicle=2;
exitvehicle=12;
coeff=[1 2 1 2];
for ii=2:it
clf('reset');
recsize(x,y,figure1);

%Random function is used to generate vechiles
xrand=round(abs(rand(1,4)).*entervehicle);
x=x+xrand;
if x<[60 60 60 60]
ytemp=evalfis(x,fuzzyselect);
if ytemp<=3
b(ii)=1;
elseif ytemp<=6
b(ii)=2;
elseif ytemp<9
b(ii)=3;
elseif ytemp<12
b(ii)=4;
else
[a,b(ii)]=max(x);
end
else
[a,b(ii)]=max(x);
end
for m=1:4
if timefactor(m)>=5
y=zeros(1,4);
timefactor(m)=0;
b(ii)=m;
end
end
y=zeros(1,4); y(b(ii))=1;
timefactor=timefactor+ones(1,4);
timefactor(b(ii))=0;
if b(ii)~=b(ii-1) && ii~=2
y(b(ii-1))=0.5;
y(b(ii))=0;
reccolour(x,y,figure1);
pause(yellowtime);
```

```
y(b(ii-1))=0;
y(b(ii))=1;
end
if x<[60 60 60 60]
greentimetemp=evalfis(x,fuzzygreentime);
if greentimetemp<=2
timeconstantfuzzy=.8;
elseif greentimetemp<=5
timeconstantfuzzy=.9;
else
timeconstantfuzzy=1;
end
else
timeconstantfuzzy=0.8;
end
greentime(ii)=ceil(((x(b(ii))+(.7*entervehicle))/exitvehicle)*timeconstantfuzzy);
for j=1:greentime(ii)+1
x(b(ii))=x(b(ii))-exitvehicle;
xrand=round(abs(rand(1,4)).*entervehicle.*coeff);
x=x+xrand;
if x(b(ii))<=0
x(b(ii))=0;
reccolour(x,y,figure1);
pause(exittime);
break
break
end
reccolour(x,y,figure1);
pause(exittime);
end
end

%statistics
[aa1, bb1]=find(b==1);
[aa2 bb2]=find(b==2);
[aa3 bb3]=find(b==3);
[aa4 bb4]=find(b==4);
VehiclesCounter=exitvehicle.*greentime;
VehiclesCounter1=0;
VehiclesCounter2=0;
VehiclesCounter3=0;
VehiclesCounter4=0;
for n=1:length(bb1)
VehiclesCounter1=VehiclesCounter1+VehiclesCounter(bb1(n));
end
for n=1:length(bb2)
VehiclesCounter2=VehiclesCounter2+VehiclesCounter(bb2(n));
end
for n=1:length(bb3)
VehiclesCounter3=VehiclesCounter3+VehiclesCounter(bb3(n));
end
for n=1:length(bb4)
VehiclesCounter4=VehiclesCounter4+VehiclesCounter(bb4(n));
end
GreenCounter1=sum(aa1)
VehiclesCounter1
GreenCounter2=sum(aa2)
VehiclesCounter2
```

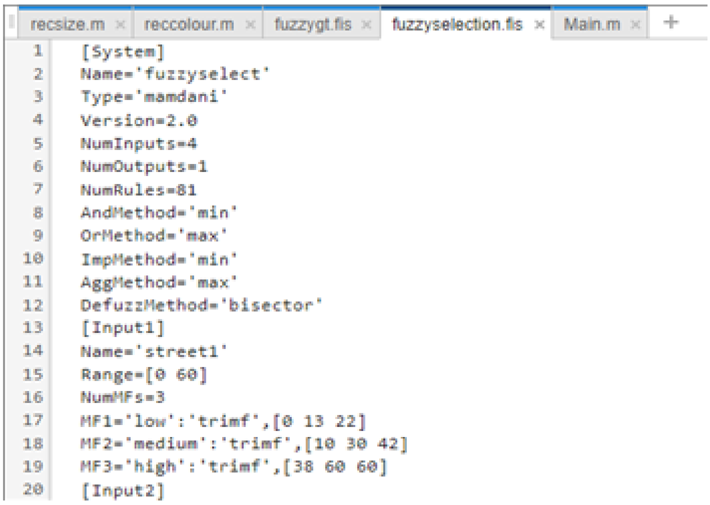
```
GreenCounter3=sum(aa3)
VehiclesCounter3
GreenCounter4=sum(aa4)
VehiclesCounter4
```

➢ Snapshots:

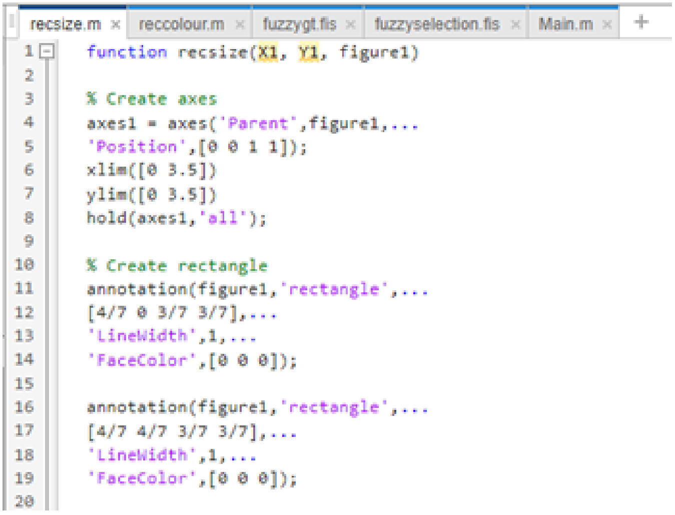


```
clc;
clear;
close all;
fuzzyselect=readfis('fuzzyselection.fis');
fuzzygreentime=readfis('fuzzygt.fis');
figure1=figure;
x=[0 30 5 40];
y=[0 0 0 0];
it=10;
timefactor=ones(1,4);
b=zeros(0,it);
yellowtime=1;
exittime=0.3;
entervehicle=2;
exitvehicle=12;
coeff=[1 2 1 2];
for ii=2:it
clf('reset');
recsize(x,y,figure1);
xrand=round(abs(rand(1,4)).*entervehicle);
```

```
VehiclesCounter4 =

    72

>>
```

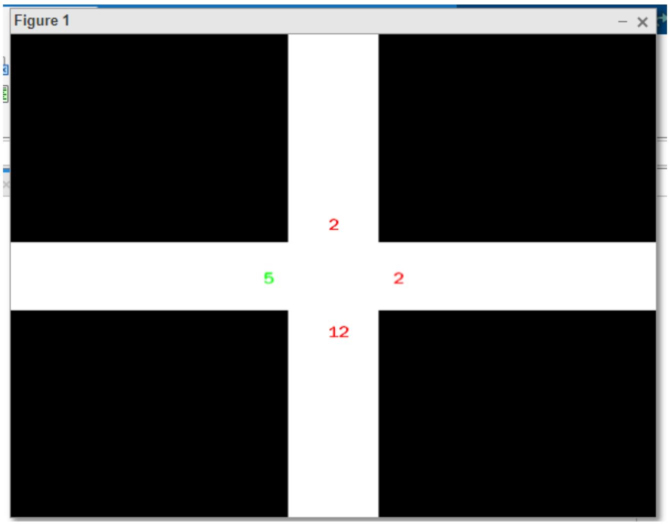


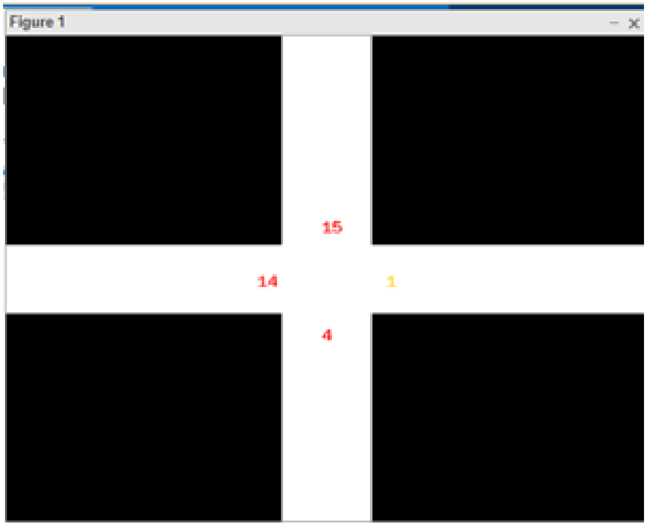
```
[System]
Name='fuzzyselect'
Type='mamdani'
Version=2.0
NumInputs=4
NumOutputs=1
NumRules=81
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='bisector'
[Input1]
Name='street1'
Range=[0 60]
NumMFs=3
MF1='low':'trimf',[0 13 22]
MF2='medium':'trimf',[10 30 42]
MF3='high':'trimf',[38 60 60]
[Input2]
```

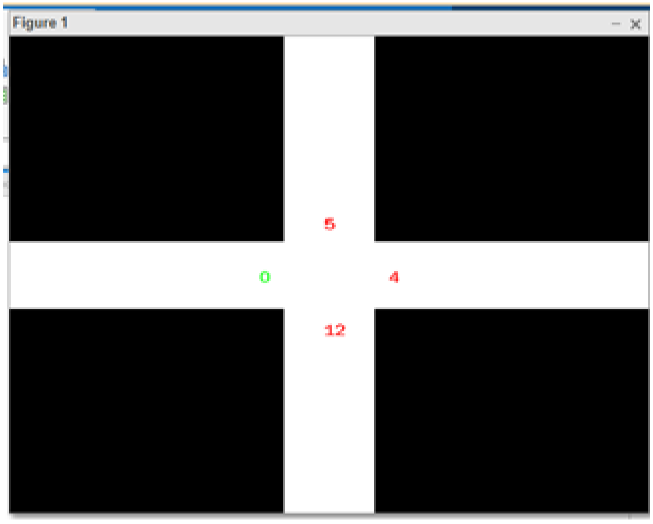


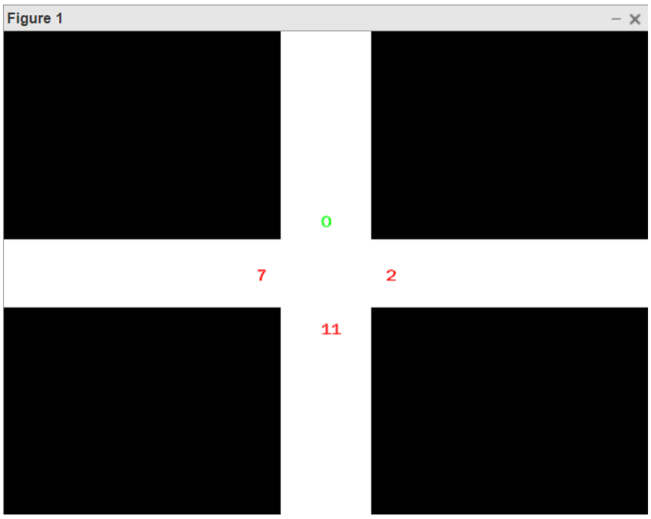
```
function recsize(X1, Y1, figure1)

% Create axes
axes1 = axes('Parent',figure1,...
'Position',[0 0 1 1]);
xlim([0 3.5])
ylim([0 3.5])
hold(axes1,'all');

% Create rectangle
annotation(figure1,'rectangle',...
[4/7 0 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);

annotation(figure1,'rectangle',...
[4/7 4/7 3/7 3/7],...
'LineWidth',1,...
'FaceColor',[0 0 0]);
```

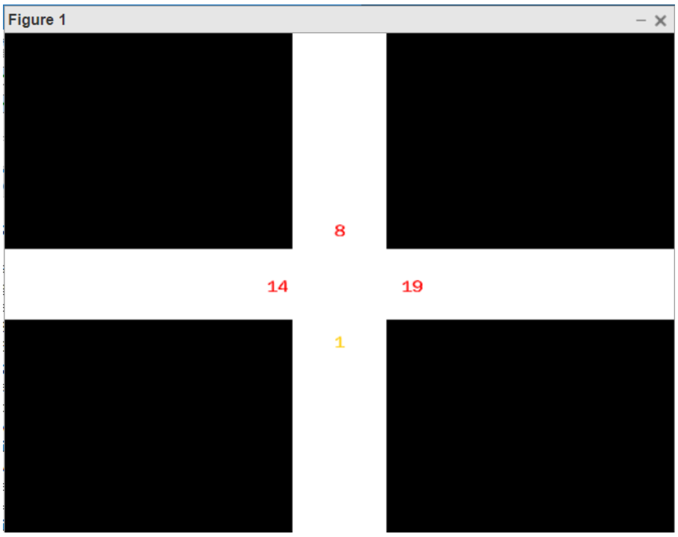
➢ Results Snapshots:

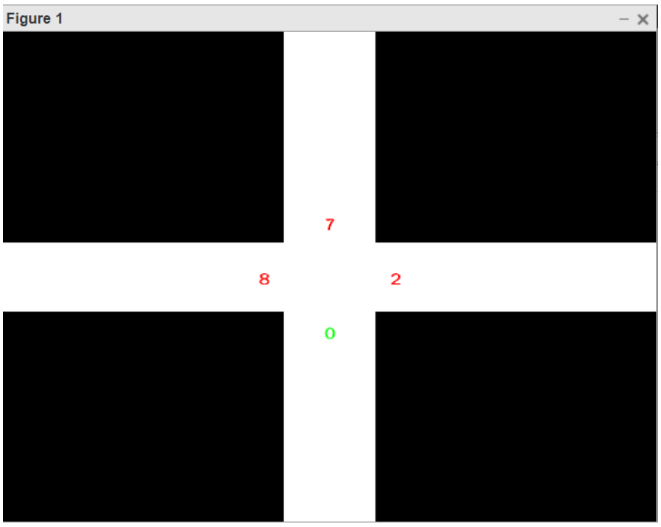


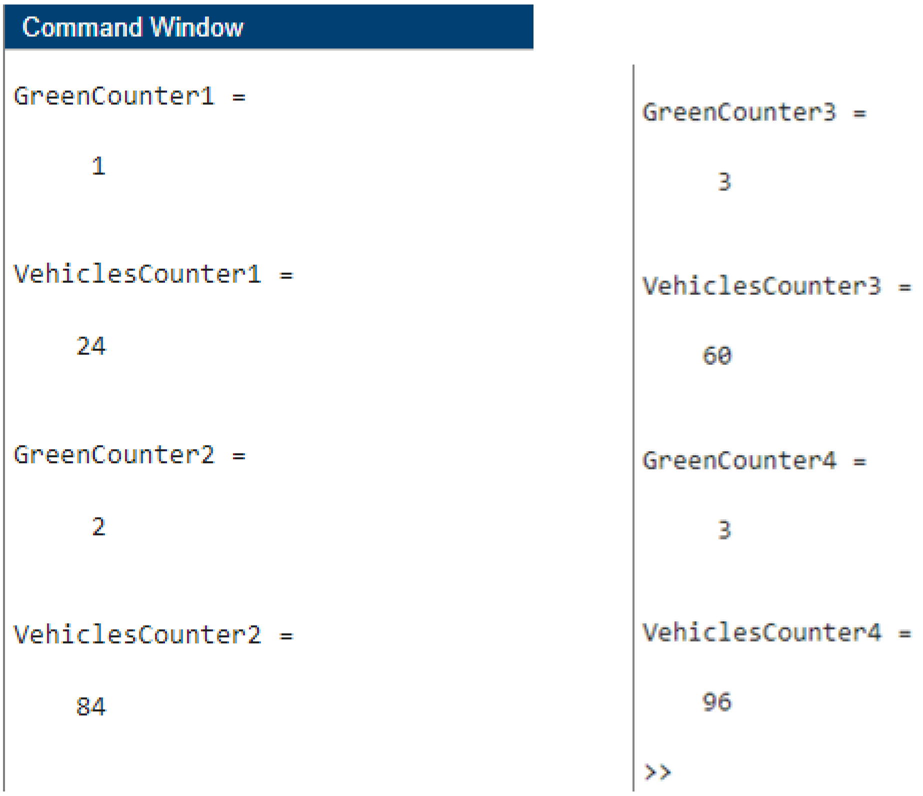












```
GreenCounter1 =

     1


VehiclesCounter1 =

    24


GreenCounter2 =

     2


VehiclesCounter2 =

    84
```

```
GreenCounter3 =

     3


VehiclesCounter3 =

    60


GreenCounter4 =

     3


VehiclesCounter4 =

    96

>>
```

➢ Result Analysis and Conclusion:

The system proposed here is very flexible. Because of the flexibility of the fuzzy logic in dealing with uncertainty, it can be used advantageously for traffic light controlling systems. The proposed FLSC and fixed time controller produces little difference in results in terms of constant traffic flow while in the case of time varying traffics, the proposed FLSC is superior to the fixed time controller. This controller gives a suitable green time to improve the traffic capacity effectively and reduced the intersection delay, which can ensure vehicles don't allow waiting too long on the road. While in the case of fixed time controller when green times finished, this will give the green time to the next phase even if the current vehicle flow is large. So, arriving vehicles must wait for the next cycle to leave. The performance of the FLSC is affected by the configuration of the membership functions of the input and output variables and the rule base. It can be observed that fuzzy logic control system provides better performance in terms of improving the safety and efficiency by reducing the waiting delay of vehicles on signals. Less traffic congestion and less waiting time at red traffic lights will reduce the fuel consumption, air pollution, sound pollution, and time and energy waste. Furthermore, with the comparative ease terms like weather conditions, environments aspects etc. can be added to the fuzzy system. A lot of research work has to be done to verify the expected features by simulation. The definitions of the fuzzy sets of the antecedents are also very easily changeable. This is a very promising application of fuzzy logic in practical areas, and will be highly useful in traffic control in the today congestion traffic.

➢ Base paper and other reference papers: -

Main Paper :-

https://www.researchgate.net/publication/292869518_Traffic_signal_control_using_fuzzy_logic

Ideology:-
https://www.researchgate.net/publication/266736638_APPLICATION_OF_FUZZY_LOGIC_IN_ACADEMIC_SETUP

➢ Refferences:

➢ [1]Jigna Ashish Patel.,(2014). Traffic Control Using Fuzzy Logic. International Journal Of Electronics, Communication & Instrumentation Engineering Research And Development (IJECIERD);ISSN(P): 2249-684x; ISSN(E): 2249-7951.

➢ [2] Ugwu, C., & Dennis, B. (2014). An Application of Fuzzy Logic Model in Solving Road Traffic Congestion'. International Journal of Engineering Research & Technology (IJERT), 3(2), 2960- 2969.

➢ [3] Prontri, S., Wuttidittachotti, P., & Thajchayapong, S. (2015, June). Traffic signal control using fuzzy logic. In 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON) (pp. 1-6). IEEE.

➢ [4] Zachariah, B., Ayuba, P., & Damuut, L. P. (2017). Optimization of traffic light controlsystem of an intersection using fuzzy inference system. Science World Journal, 12(4), 27-33.

➢ [5] Sujithra,P.,& Sujithra,T.,& Masoodhu Banu, N.M., (2019). Effective Lane Management for Emergency Vehicles and Adaptive Signaling For Dynamic Traffic Congestion Using Fuzzy Logic. International Journal of Innovative Technology and Exploring Engineering (IJITEE) 2278-3075.

➢ [6] Chandan, K., Seco, A. M., & Silva, A. B. (2017). Real-time traffic signal control for isolated intersection, using car-following logic under connected vehicle environment. Transportation research procedia, 25, 1610-1625.

➢ [7] Jha, M., & Shukla, S. (2014). Design Of Fuzzy Logic Traffic Controller For Isolated Intersections With Emergency Vehicle Priority System Using MATLAB Simulation. arXiv preprint arXiv:1405.0936.