

## **Implementation of a Fuzzy-Based Line Follower Robot using Arduino**

**Mohammad Safwan Mohamed Alias, Ya'akob Yusof\* and Nurul Muthmainnah Mohd Noor**

*Faculty of Mechanical Engineering, Universiti Teknologi MARA (UiTM), 40450 Shah Alam, Selangor, Malaysia*

### **ABSTRACT**

Applying an intelligent system means the machine is capable of making its own unique decision. Fuzzy logic is a form of logic that acknowledges other forms of true and false values. With fuzzy logic, propositions may be diagrammatic with degrees of truthfulness and falsehood. The main aims of this study are to demonstrate the method to apply fuzzy logic in a structure that uses Arduino as its brain, and apply it in a line follower mobile robot's decision making algorithm. This mobile robot consists of two front wheels and a nub caster at the back. It uses a line follower array that consists of eight infrared (IR) sensors. The system control is based on the input from IR sensors, which measures the intensity of light reflected by the track. Data were then transmitted to the microcontroller and will then be sent the correct command to the motor driver so that the trail can be followed. The performance of the line follower robot when using the fuzzy logic algorithm was compared to the line follower algorithm, which uses simple if-else commands. Analysis is primarily on the time taken to complete the track, along with the behaviour of the robot while manoeuvring. From the results, fuzzy logic is shown to provide a better performance in terms of the time taken to complete the track compared to the other set of rules. In addition, from the video recordings, fuzzys are moving smoother as compared to the non-fuzzy logics.

*Keywords:* Arduino, Fuzzy Inference System, fuzzy logic, line follower robot, Matlab

### **ARTICLE INFO**

*Article history:*

Received: 25 October 2016

Accepted: 17 March 2017

*E-mail addresses:*

[mohdsafwan\\_mohd@yahoo.com](mailto:mohdsafwan_mohd@yahoo.com)

(Mohammad Safwan Mohamed Alias),

[yaakob2241@salam.uitm.edu.my](mailto:yaakob2241@salam.uitm.edu.my) (Ya'akob Yusof),

[muthmainnah@salam.uitm.edu.my](mailto:muthmainnah@salam.uitm.edu.my)

(Nurul Muthmainnah Mohd Noor)

\*Corresponding Author

### **INTRODUCTION**

Robots are electromechanical machines that having ability to perform tasks or actions based on electronic programming. Line follower robots are one type of mobile robots that is able to detect and follow a track drawn on the floor even when the path is altered by changing the shape of the line. The path can be a visible black line on a white background, or vice versa.

The forward movement of a line follower robot involves both motor turns on and rotates simultaneously. To move to the left, the right motor is turned on and left motor is turned off, and to move to the right, right motor is turned off and left motor is turned on. It can also move faster when one motor rotates forward while the other one rotates backward. When applying the fuzzy logic, the time to make decision whether to turn left or right is much shorter and the movement will become smoother compared to the algorithms without the fuzzy logic.

This paper discusses the design of a line follower robot operation based on fuzzy logic controller and without fuzzy logic controller. The fuzzy logic algorithm was applied in a Matlab's Toolbox, Fuzzy Inference System (FIS) software using single input, two output options. The input is the difference in the readings between eight IR sensors set in front, while the outputs are the speed for the right and left motors.

## **METHOD**

The line follower robots have few major parts, namely the microcontroller, motor driver, IR sensors, actuators and its chassis or body structure.

The microcontroller is programmed to create a mechanism for move forward, turn right or turn left based on the input returning from the sensor or comparator. When using Arduino, a comparator is not needed. The outputs of the microcontroller are fed to the motor driver.

For this study, the microcontroller used is Arduino Uno. An Arduino is one of the simplest and most easy-to-use microcontrollers available. Arduino comes with an IDE that helps to burn programming code onto the microcontroller from a computer. It is open supply and extensive documentation could be easily obtained online.

A robot needs a driver for controlling and generating power to the motor. In general, the microcontroller sends instructions to the driver after processing data from sensors and the driver transmits voltage to the motor according to the inputs (Hasan et al., 2013; Pakdaman, Sanaatiyan, & Ghahroudi, 2010). The suitable motor driver for this study is IC L298 because it is be used to control two motor (Pakdaman & Sanaatiyan, 2009).

Arduino Motor Shield which is based on the L298 was used. Two DC motors could be driven, where the speed and direction of each motor could be controlled independently. The shield is TinkerKit compatible, which means that an electronic project could be created quickly by plugging TinkerKit modules to the board.

In terms of the sensor, the most suitable type is Infrared (IR) sensor. The reasons include ease of implementation, low cost, and availability (Pakdaman, Sanaatiyan, & Ghahroudi, 2010). IR sensor is used to detect white and black surfaces. White surfaces generally reflect well, while black surfaces reflect poorly.

In the designed robot, it is decided to use an IR array that consists of eight IR sensors. The configuration of IR sensors is in-line because it gives advantage at sensing, and responding to the turns in an easy course with wide turns. It is also easy to install. Its main shortcoming is its poor in sensing capability at a greater than 90 degree turn, thus the track must avoid this.

For the movement of the line follower robot, wheels system is used. The designed mechanism used 2 DC motors because DC motors can rotate and initiate both forward and backward movement, depending on the direction of the applied current. The DC motors come with a good speed ratings and generate enough torque for the robot.

Chassis is the main supporting structure of a robot, where all the components are attached to it. Therefore, it needs sufficiently large and sturdy structure to cope with the weight of the component. There are some good materials that can be used for designing robot such as wood, plastic, aluminium and brass alloy. Thus, it was decided to use the shadow chassis, in which the chassis plates and mounts are cut from ABS plastic and the hub patterns are utilised for sensors, controllers and power. All of the components and hardware for the line follower robot are assembled, as shown in Figure 1.



Figure 1. The line follower robot

The track is shown in Figure 2. A track is a path for the line follower robot to follow. The result, which is the time taken for the robot to complete the track as well as its behaviour, is used in analysing the robot performance. An example of a simple track is either a round or rectangle shape track. While for the advanced track, there should be more intersections and corners.



Figure 2. Track

### Fuzzy Logic Controller

Figure 3 and Figure 4 show the process flowchart of the line follower robot, with and without fuzzy logic, respectively. The algorithm for this line follower robot consists of a single input and two outputs. The different value of eight IR sensors is served as the input to both algorithms. For without fuzzy, it is a simple if-else algorithm which make the robot to go left if it is too much towards the right, and vice versa. It did not have crisps value consideration and sense of motor speed. For fuzzy logic, the output of the scheme, which is acquired from the fuzzification of each input and output using the associated membership functions, will control the speed of the right and left DC motors.

For the fuzzy logic programming, Matlab's Fuzzy Logic Toolbox™ has been used to design the FIS or fuzzy logic controller, before it is embedded into the microcontroller, the Arduino UNO. The toolbox provides functions that allow the user to analyse, simulate and design the systems based on fuzzy logic.

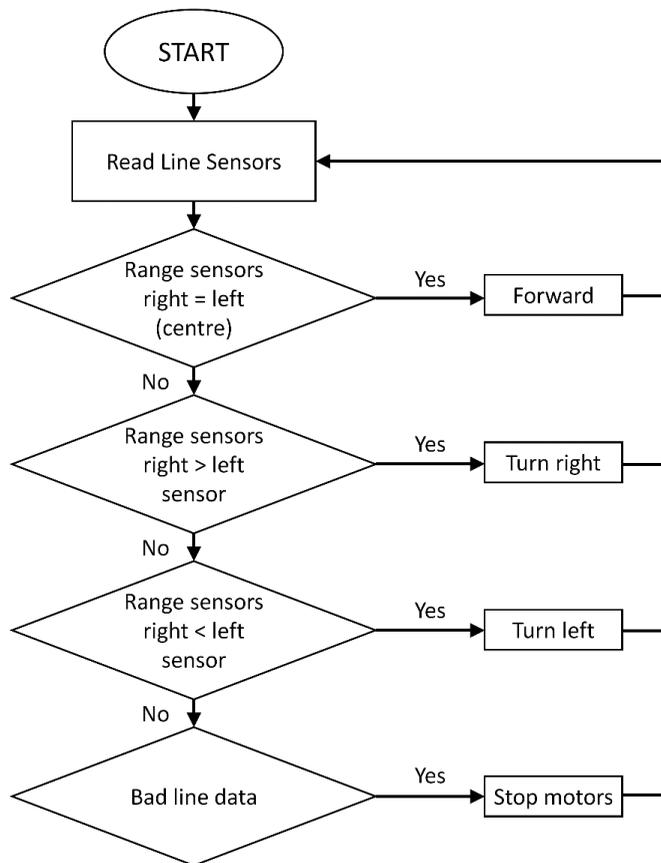


Figure 3. Flowchart for line follower robot (without fuzzy)

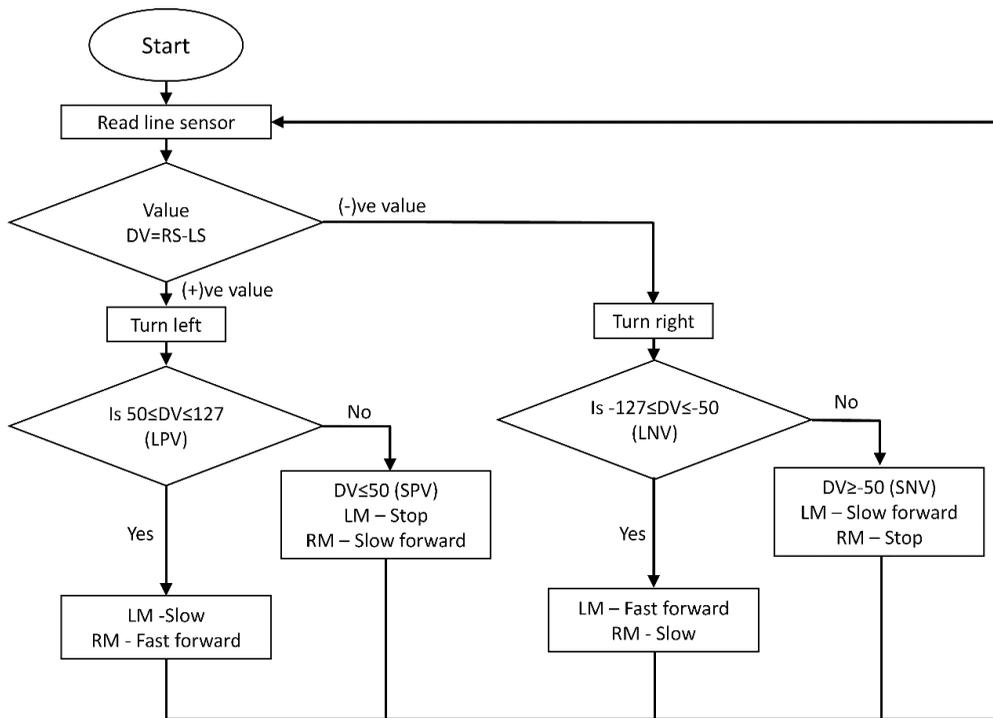


Figure 4. Flowchart for line follower robot with fuzzy logic's algorithm. The basis of this algorithm was taken from Farooq et al. (2014)

In general, a fuzzy logic controller consists of fuzzification, fuzzy rule base and defuzzification process. Fuzzification process is a process that converts crisp input value into a fuzzy value. A single input, which is the difference value between the eight IR sensors (DV), is termed by four fuzzy set. The DV ranges from -127 to +127. The terms of the fuzzy sets are presented in Table 1, while the membership functions for the input, DV, is described in Figure 3 and the following equations.

Table 1  
Fuzzy set for input of the line follower robot

Fuzzy set	Description
LPV	DV has large positive value
SPV	DV has small positive value
LNV	DV has large negative value
SNV	DV has small negative value

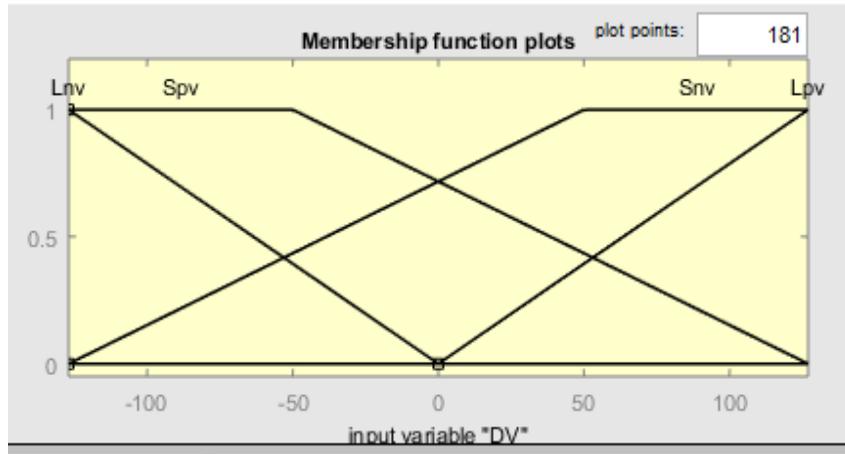


Figure 5. Membership function plot for the input (DV)

$$\mu_{DV,LPV} = \begin{cases} 0, & -127 \leq x \leq 0 \\ 0.79d_v, & 0 \leq x \leq 127 \end{cases} \quad (1)$$

$$\mu_{DV,SPV} = \begin{cases} 100, & -127 \leq x \leq -50 \\ -0.64d_v + 80.8, & -50 \leq x \leq 127 \end{cases} \quad (2)$$

$$\mu_{DV,LNV} = \begin{cases} 0.79d_v, & -127 \leq x \leq 0 \\ 0, & 0 \leq x \leq 127 \end{cases} \quad (3)$$

$$\mu_{DV,SNV} = \begin{cases} -0.64d_v + 80.8, & -127 \leq x \leq 50 \\ 0, & 50 \leq x \leq 127 \end{cases} \quad (4)$$

Over these expressions, the triangular (trimf) and trapezoidal (trapmf) membership functions were used because of their simplicity to implement and fast for computation (Farooq et al., 2014).

The output of the fuzzy controller is on the right (RM) and on the left (LM) are motor speeds, which are expressed by slow and fast fuzzy sets. Slow is the minimum (0%), while speed is the maximum (100%) speed of the DC motor. The terms of the fuzzy sets are presented in Table 2. Membership functions for both the LM and RM outputs are shown in Figure 4 and the expressions below.

Table 2  
Fuzzy set for output of a line follower robot

Fuzzy set	Description
Slow	One or both motors' speed are slow
Fast	One or both motors' speed are high

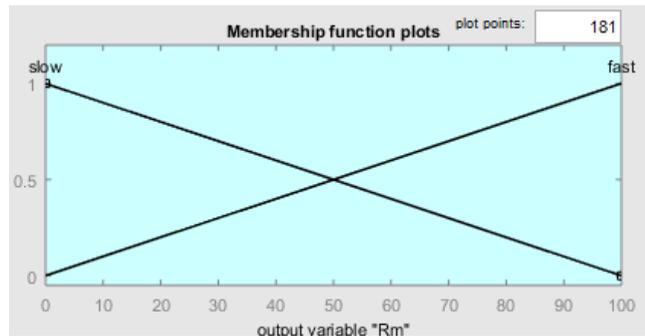


Figure 6. Membership function plot for output (RM/LM)

$$\mu_{LM,slow} = -l_m + 100, \quad 0 \leq l_m \leq 100 \quad [6]$$

In this study, the microcontroller defines the rules to control the speed of both motors to attain preferred navigation for robot. The rule-base created for the correlation between sensor values and the speed of both motors is shown in Table 3.

Table 3  
Fuzzy rule-base for the input and outputs

Output	Right Motor (RM)	Left Motor (LM)
Input		
Small negative value (SNV)	Fast	Fast
Large negative value (LNV)	Slow	Fast
Small positive value (SPV)	Fast	Fast
Large positive value (LPV)	Fast	Slow

Based on rule 1, for instance, if the sensor detects black line with small negative value, the robot will move fast forward. Rule 2 describes the situation when the robot gets diverged from the track largely in the left direction. In this situation, the robot will turn to the right to keep the robot on the track by spinning the right motor slowly and the left motor faster than the right motor.

The control output obtained from the combination of input, output membership functions, and fuzzy rules remains an imprecise or fuzzy element; and this method is known as fuzzy inference. To create fuzzy output accessible for real applications, a defuzzification method is required. The defuzzification method is supposed to convert the fuzzy output to the crisp output.

Commonly used techniques for the defuzzification method are the Mean of Maximum method, Centre of Gravity method and the Height method. The technique used in this study

is the centre of gravity method (COA). The formula for calculating the COA method can be described as:

$$Z_o = \frac{\sum_{i=1}^n Z_i \mu_{out}(Z_i)}{\sum_{i=1}^n \mu_{out}(Z_i)} \quad (7)$$

Where  $\mu_{out}(Z_i)$  is the  $i=1, 2, \dots, n$  sampled values of the gathered the output membership function and  $Z_o$  is the crisp output that described the duty cycle for PWM signal speed for the motor (Farooq et al., 2014).

### RESULTS AND DISCUSSION

The results show the performance of the line follower robot when using the fuzzy logic algorithm and the common if-else commands. Based on Figure 5, situation (a) is when the input value is large positive value, thus the output values shows RM is higher than LM. It means that the robot movement is to the left. While in situation (b), the value of LM is higher than RM. That shows, the movement robot is now in right turn.

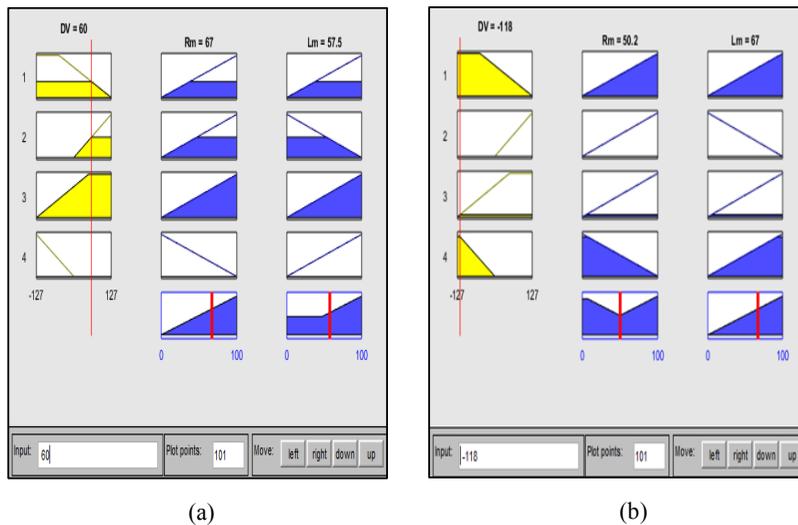


Figure 7. The rule view of the input and outputs membership function when the input was set to: (a) 60; and (b) -118 input value

Table 4  
Experimental results and observations

Criteria	Without Fuzzy Logic	With Fuzzy Logic
Average time for one complete cycle	33.8s	26.8s
Average speed for one complete cycle	4.93m/s	6.21m/s
Average time to complete five full cycles	169.1s	134.3s
Line tracking movement	Uneven, especially at corners	Smooth
Tendency to move out from track	Low	Low

The results presented are the average performance from the conducted tests. Due to the algorithm, the robot with fuzzy logic performs better than the one without fuzzy, as expected. In terms of the speed, the fuzzy's robot takes about 21% faster to complete the same track compared to the other. This is mainly due to the crisps' value considerations in the programming, as well as the speed adjustment features in the programming. It also tracks the line smoothly as compared to without fuzzy, especially when approaching corners. With the application and implementation of the fuzzy logic algorithm, it gives more advantages and benefits to the robot.

However, there are a few observations that need to be highlighted. Among other is the robot's tendency to move away far from the track, which is stated as low. This is due to the nature of the track used, which does not have sharp turns. This is because the study intends to merely demonstrate the difference between fuzzy and non-fuzzy approach. If the track consists of sharp turns, the programming will need to be improved, especially on controlling the speed of the motor.

## CONCLUSION

This paper presents an analysis of a simple line follower system, with and without fuzzy logic algorithm implemented in its brain using Arduino as its microcontroller. Both the systems were compared to illustrate which one would give a better performance in terms of the time taken to complete the track, as well as the movement of the robot. From the results, it is clear that the line follower with fuzzy logic gives a better performance compared to without fuzzy, both in terms of the time taken to complete the track and behaviour of the robot during the test.

For future experiments, it is important to improve the ability of the robot especially in taking sharp corners. In improving that ability, the algorithm will need to be improved, especially in defining the speed of the car during motion. In addition, this Matlab's FIS system is also possible to be applied in other applications such as autonomous mobile robot, autonomous vacuum cleaner and other systems.

## REFERENCES

- Farooq, U., Amar, M., Asad, M. U., Abbas, G., & Hanif, A. (2014). Fuzzy logic reasoning system for line following robot. *International Journal of Engineering and Technology*, 6(4), 244.
- Hasan, K. M., Al-Nahid, A., Reza, K. J., Khatun, S., & Basar, M. R. (2013, April). Sensor based autonomous color line follower robot with obstacle avoidance. In *Business Engineering and Industrial Applications Colloquium (BEIAC), 2013 IEEE* (pp. 598-603). IEEE.
- Pakdamani, M., & Sanaatiyan, M. M. (2009, December). Design and implementation of line follower robot. In *Computer and Electrical Engineering, 2009. ICCEE'09. Second International Conference on* (Vol. 2, pp. 585-590). IEEE.
- Pakdamani, M., Sanaatiyan, M. M., & Ghahroudi, M. R. (2010, February). A line follower robot from design to implementation: Technical issues and problems. In *Computer and Automation Engineering (ICCAE), 2010 The 2<sup>nd</sup> International Conference on* (Vol. 1, pp. 5-9). IEEE.
- Sanaatiyan, M. M. & Ghahroudi, M. R. (2010). A Line Follower Robot from design to Implementation. *Technical Issues and Problems*, 1, 5-9.

