Journal of Applied Sciences

ISSN 1812-5654

OPEN ACCESS

Journal of Applied Sciences

ISSN 1812-5654
DOI: 10.3923/jas.2016.570.579

Research Article
Navigation of an Automated Guided Vehicle Based on Sugeno Inference Engine

B.B. Abu Bakar, S.S. Mohmad and I. Adam

British Malaysian Institute, Universiti Kuala Lumpur, Bt. 8, Jalan Sungai Pusu, 53100 Gombak, Selangor, Malaysia

Abstract
This study presents the application of Sugeno inference engine to guide the navigation of an Automatic Guided Vehicle (AGV). In general, the input of the system is derived from the magnetic guided sensor. The sensor generates 8-bit data representing the current location of the AGV. The processed data is used as a command which direct the AGV navigation. The algorithm has been simulated and successfully tested with a maximum deviation of 2 cm off the target line. In addition, the method has been compared with a traditional technique which using if else method, resulting in the proposed technique outperform the traditional technique in term of speed, memory usage and efficiency. This study discusses in details the development of the technique and the code listing is also provided for further comment. The novelty of this technique is on its programmability and portability.

Key words: Automatic guided vehicle, hardware, software, fuzzy controller, 8-bit input

Received: April 03, 2016 Accepted: July 06, 2016 Published: November 15, 2016

Citation: B.B. Abu Bakar, S.S. Mohmad and I. Adam, 2016. Navigation of an automated guided vehicle based on Sugeno inference engine. J. Applied Sci., 16: 570-579.

Corresponding Author: B.B. Abu Bakar, British Malaysian Institute, Universiti Kuala Lumpur, Bt. 8, Jalan Sungai Pusu, 53100 Gombak, Selangor, Malaysia

Competing Interest: The authors have declared that no competing interest exists.

Data Availability: All relevant data are within the paper and its supporting information files.

INTRODUCTION

With the rapid developing of robotic technologies, an Automated Guided Vehicle (AGV) have been widely used to various fields of applications, particularly in those dangerous and harsh environments (Fig. 1)[1]. Automatic navigation technology is one of the key issues among all these robot researches. By utilizing distinctive kinds of sensors, the navigation approaches of mobile robots can be categorized into several types such as magnetic field, inertia, vision and sonar. As a promising technology, magnetic navigation provides the advantages like high accuracy, perfect robustness, simple installation and low maintenance cost.

The study of magnetic navigation began since 90 sec in United States and Japan. Recently, its applications in the field of intelligence transportation can be seen in many regions like United States, Netherlands and Japan[2]. Current systems nowadays are mostly based on permanent magnetic markers, which are discrete thus limits the accuracy and the speed of the navigation (Fig. 2). On the other hand, few researches have been done on wire based magnetic navigation[3]. For many years, researchers have tried to cultivate navigation algorithm to be used by mobile robots while at the same time they try to deal with the naturally inaccurate information from sensors.

In general, line tracking technique is relatively easy to understand and easy to implement[4]. The current position gained and the motors speed is adjusted accordingly. This is accomplished by using the current position (output sensor) as the pointer to the array which determine motors speed. Table 1 shows the input and corresponding output for 3-line sensor.

The system has to be designed not to consider the don't care input which in this case is 0b101 as no such case will happen. This is due to the position of the line itself that lies only at the centre as illustrated on figure below.

It can be solved simply by provide a control mechanism in the code which ignore such input. For example by using the case of the conditional switch such as in Fig. 3.

The current trend on line tracking require 2×2^n memory where, n is the number of line for example, the 3-line input sensor on Table 1 earlier would need 16 from 2×2^3.

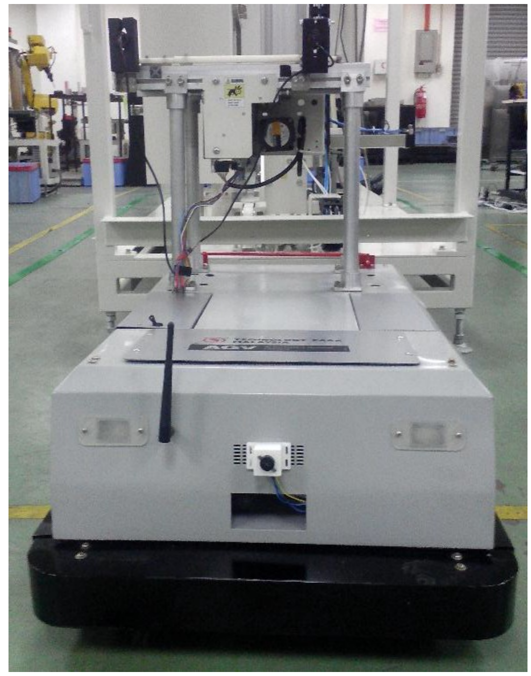


Fig. 1: Appearance of automatic guided vehicle

Fig. 2: Input of a magnetic line

Table 1: Corresponding input and output for 3-line sensor

B2	B1	B0	Motor left	Motor right
0	0	0	100% inversely to previous direction	100% inversely to previous direction
0	0	1	100%	60%
0	1	0	100%	100%
0	1	1	100%	80%
1	0	0	60%	100%
1	0	1	x	x
1	1	0	80%	100%
1	1	1	100% inversely to previous direction	100% inversely to previous direction

```
 if (copy_line!=line)
 {
     switch(line)
     {
         case 0b000:
         case 0b111:
                 if (copy_line==100 || Copy_line==110)
                         {
                         ML=100; MR=0;
                         }
                 if (Copy_line==001 || Copy_line==011)
                                 {
                                 ML=0; MR=100;
                                 }
}
switch(line)
{
        case 0b000:
        case 0b111:
            if (copy_line==100 || copy_line==110)
                {
                ML=100; MR=0;
                }
            if(copy_line==001 || Copy_line==011)
                {
                ML=0; MR=100;
                }
            break;
        case 0b001:ML=100; MR=60;
            break;
        case 0b010: ML=100; MR=60;
            break;
        case 0b011: ML=100; MR=60;
            break;
        case 0b100: ML=100; MR=60;
            break;
        case   0b110: ML=100; MR=60;
            break;
        }
copy_line=Line;
}
```

Fig. 3: Control mechanism by code

This number will gradually increase as the number of input increase. On the other hand, it will be difficult to tune the corresponding output of the motor by using the current trend.

This study present the technique where the line is converted into linear scale. For example in the previous case, the weight of the middle sensor is 0, the right is 1 and the left is -1. This is done by linearize the input using Sugeno inference engine. Thus the converted input will be as simulated in Table 2.

The value can be used in such fuzzy or classical control system to navigate the AGV. The process of tuning can be easier. An 8-bit input line sensor is altered into a weighted output voltage. Fuzzy control algorithm are utilised to control the direction and speed of the robot to achieve high-accuracy and high-speed automatic navigation. The effectiveness, robustness and applicability of this method have been verified through the experiments.

Table 2: Converted input of 3-line sensor

B2	B1	B0	Converted input
0	0	0	x
0	0	1	1
0	1	0	0
0	1	1	0.5
1	0	0	-1
1	0	1	0
1	1	0	-0.5
1	1	1	x

MATERIALS AND METHODS

Hardware and software: The AGV was designed as a rear drive and rack steering. Two drive wheel were located at the back of the vehicle and the other two in front is the independently controllable wheels. Such a configuration provided very convenient forward and backward locomotion, as well as steering in any direction just by controlling the speeds of the motors. The caster at the rear provided stable pivoting for turning. The drive motors were connected by chains to the drive shaft that accommodated two rear wheels, which ensured straight motion.

The magnetic guide sensor that sense the magnetic line is controlled by the microcontroller. The microcontroller assisted by fuzzy, determines the speed and velocity of the AGV. The information obtained then transmitted to the activator that controls the speed of the BLDC motor (Fig. 4).

Simulation by MATLAB: The fuzzy controller used in this study is designed using five basic steps. The first step is identifying the variables, input and output with its corresponding ranges for the controller, the selection of meaningful linguistic variables and the establishment of the fuzzy sets for each variables. The sugeno inference engine is selected as it is less mathematically extensive and suitable to be used by medium range microcontroller[5]. The FIS editor of the system is shown in Fig. 5.

In order to express the uncertainty of the system, in the second step, the membership functions (p(x)) for each linguistic variables are developed. The membership functions developed and its corresponding linguistic variables are represented in Fig. 6.

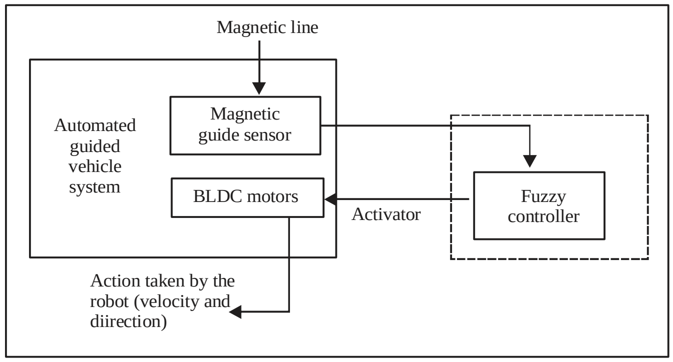


Fig. 4: Fuzzy logic controller

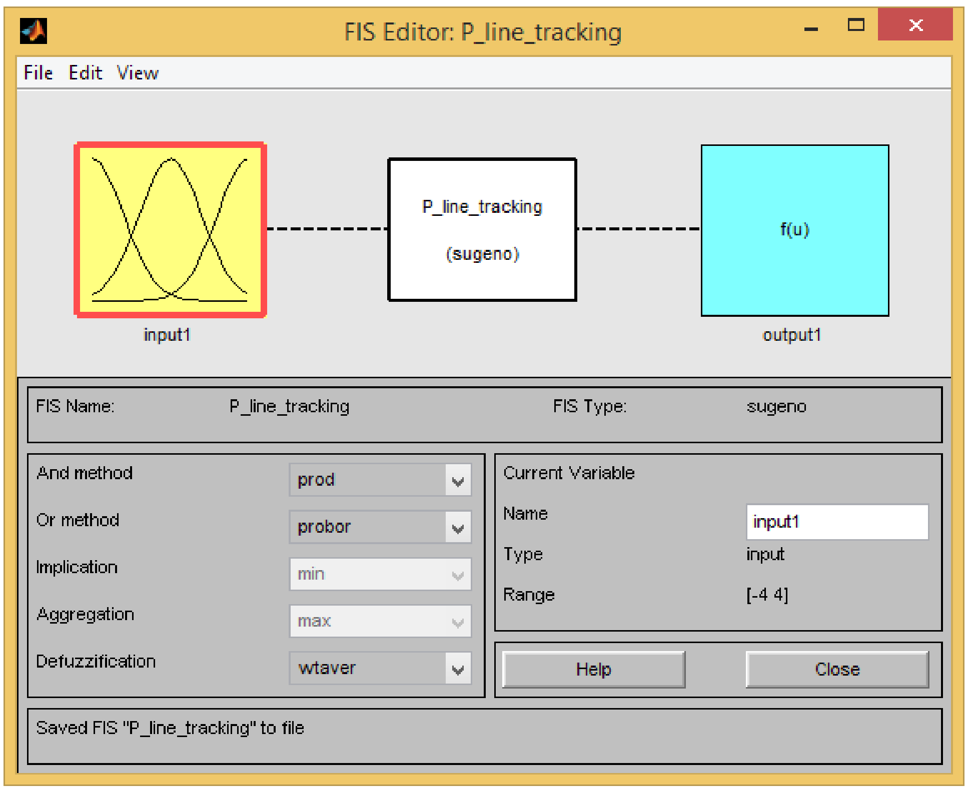


Fig. 5: Identifying the variables (input and output)

The third step was devoted to formulate inference rules using the human knowledge (regarding the control problem) obtained from experience. In this case, knowledge was obtained via common sense. The entire knowledge (rules), needed by the mobile robot to navigate were expressed in Fig. 7, containing 7 rules needed to navigate autonomously.

The fourth step corresponds to the definition of the inference engine and finally, conclusions in fuzzy form is obtained by the inference engine were converted to singular real numbers. The obtained number by the defuzzification process (5th step) represents the action to be taken by the controller.

Simulation by proteus: The circuit for ISIS proteus simulation is shown in Fig. 8. As shown, 8-slide switch is used to represent the 8-bit magnetic sensor. The input to the system is the present location of the sensor which is represented by the position of the switches. For example if the sensor is in the middle of the line the input would be 00111100_B.

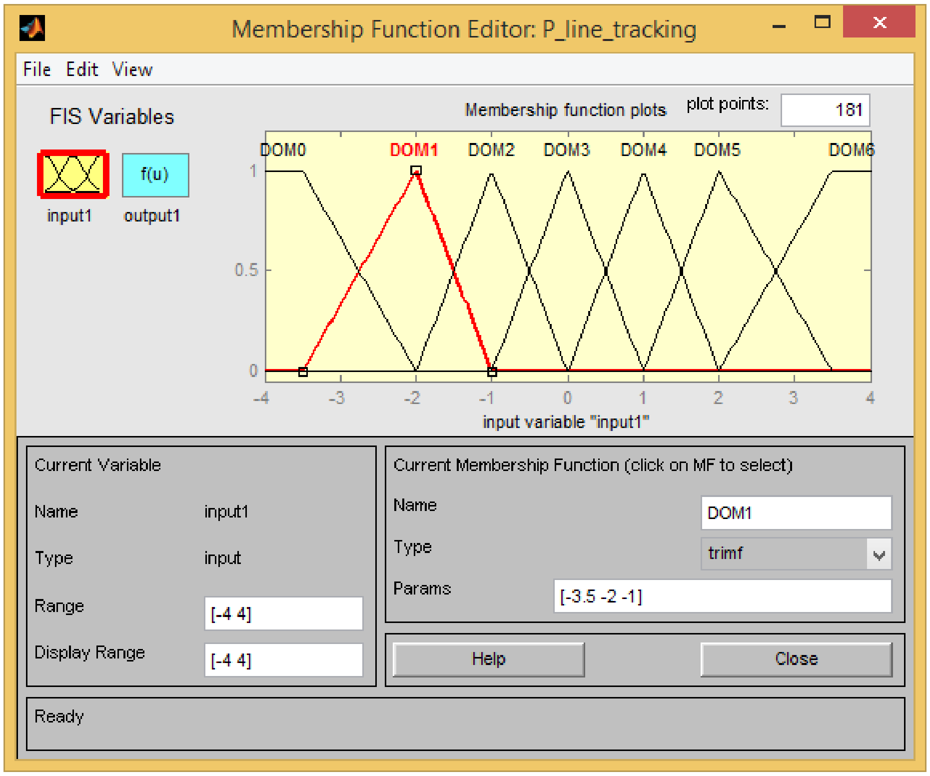


Fig. 6: Fuzzy degree of memberships as the input

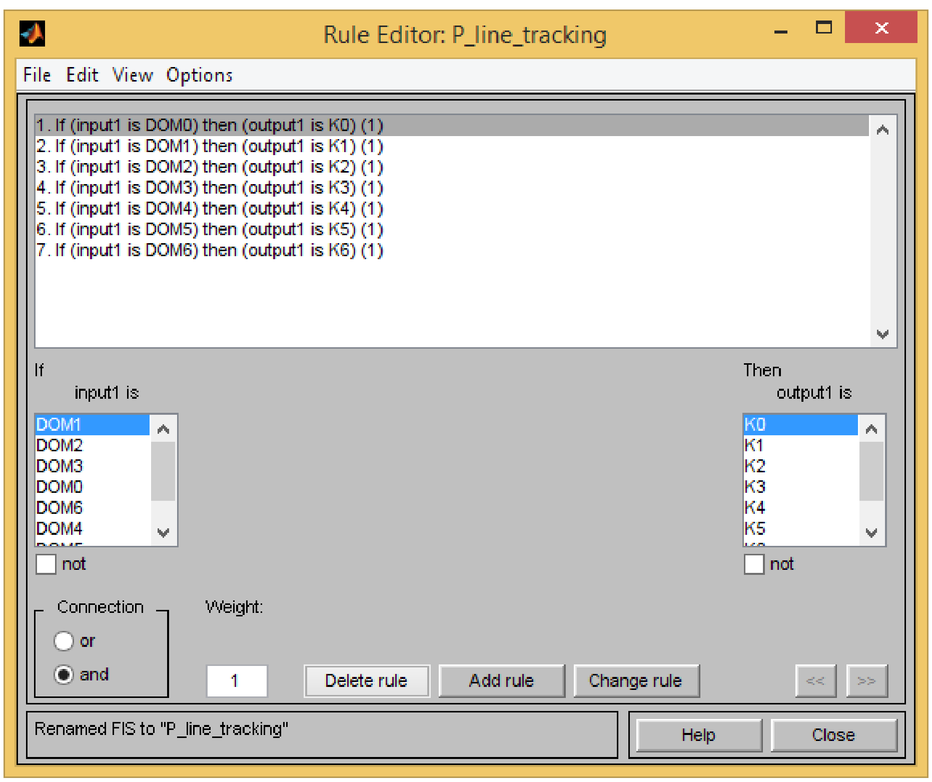


Fig. 7: Seven rules in the system

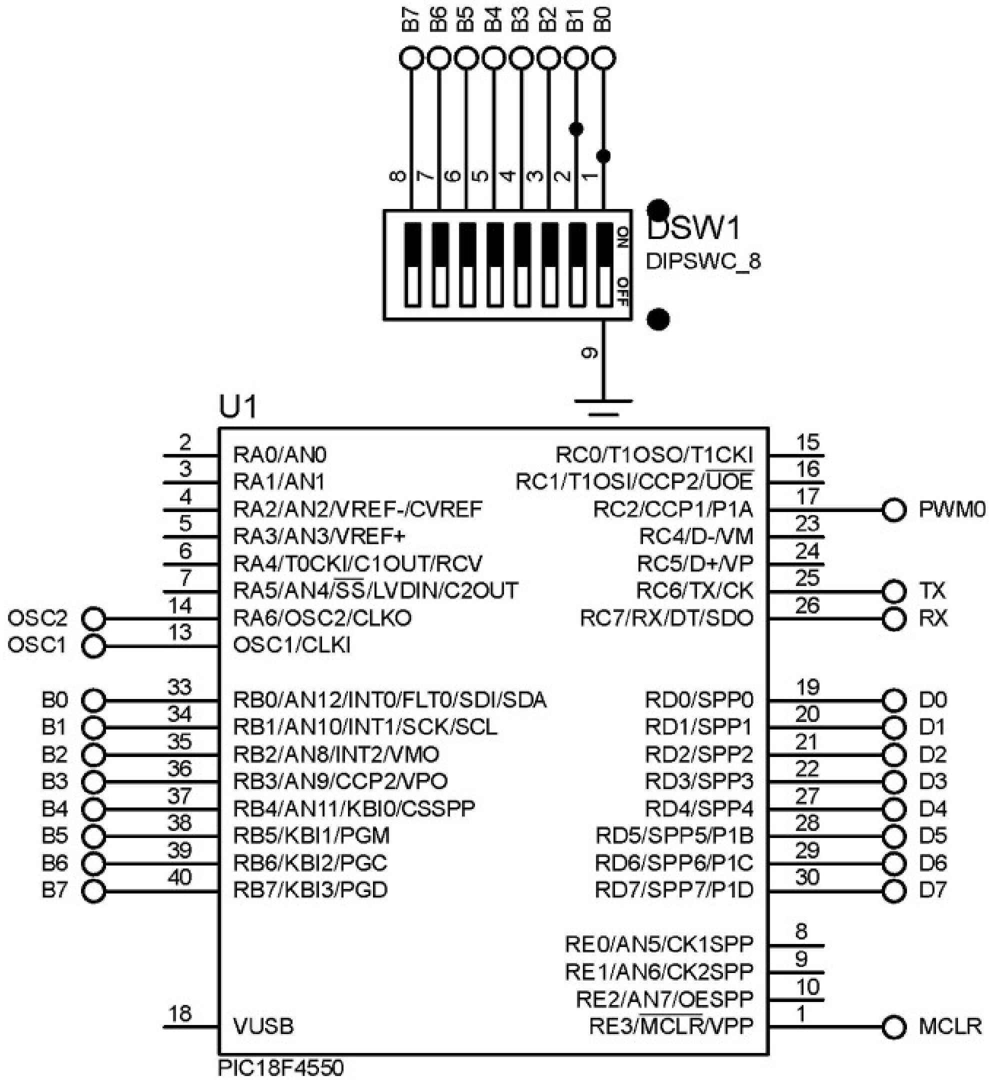


Fig. 8: Simulation by proteus

The expected result will be the corresponding converted input. It is the number from -3.5 up to 3.5. The most left position is represented by -3.5 and the most right is represented by 3.5. The microcontroller has been programmed to read the switches, process the input according to the fuzzy logic technique discussed previously and print the result through serial port onto the virtual terminal.

The input and corresponding result which is the converted input is listed in Table 3. Assuming the less significant switch is closed while the others are opened, the input to the system and the corresponding result would be 10000000_B and -3.50, respectively. This is illustrate in the 1st row of Table 3.

RESULTS AND DISCUSSION

The 8-bit input line sensor is modified into a weighted number so that the process to determine the desired output is easier (Fig. 9). This is achieved by applying the Sugeno inference engine to convert the binary like input into weighted number. The converted input hence the weighted number can be used easily in a system such fuzzy logic controller or proportional derivative controller just to name a few. In this study, the weighted number is being used as an input to another fuzzy logic controller to determine the exact speed of line following robot. Further using the weighted input the exact duty cycle thus the speed of left and right motors of the line following robot is deduced and shown in Table 4.

Table 3: List of corresponding input and converted input

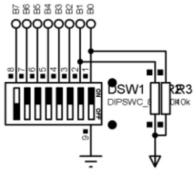
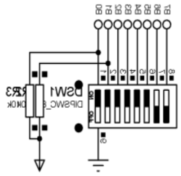
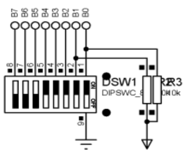
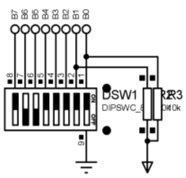
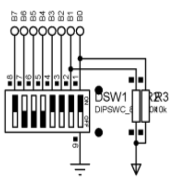
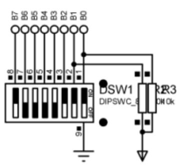
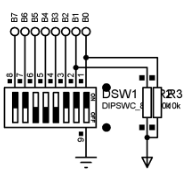
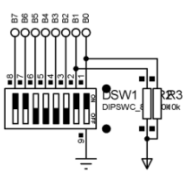
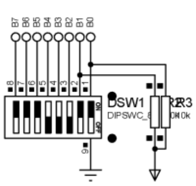
Input (sensor)	Converted output (through virtual terminal)
	Input =>1.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, converted=> -3.50
	Input =>1.00, 1.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, converted=> -3.00
	Input =>1.00, 1.00, 1.00, 0.00, 0.00, 0.00, 0.00, 0.00, converted=> -2.50
	Input =>0.00, 1.00, 1.00, 0.00, 0.00, 0.00, 0.00, 0.00, converted=> -2.00
	Input =>0.00, 1.00, 1.00, 1.00, 0.00, 0.00, 0.00, 0.00, converted=> -1.50
	Input =>0.00, 1.00, 1.00, 1.00, 1.00, 0.00, 0.00, 0.00, converted=> -1.00
	Input =>0.00, 0.00, 1.00, 1.00, 1.00, 0.00, 0.00, 0.00, converted=> -0.50
	Input =>0.00, 0.00, 1.00, 1.00, 1.00, 1.00, 0.00, 0.00, converted=> 0.00
	Input =>0.00, 0.00, 0.00, 1.00, 1.00, 1.00, 0.00, 0.00, converted=> 0.50

Table 3: Continue

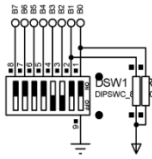
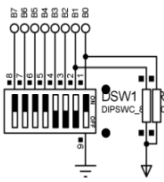
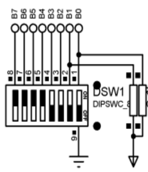
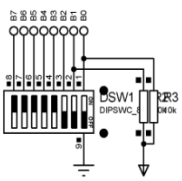
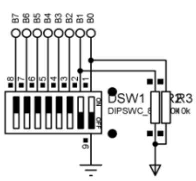
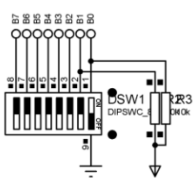
Input (sensor)	Converted output (through virtual terminal)
	Input =>0.00, 0.00, 0.00, 1.00, 1.00, 1.00, 1.00, 0.00, converted=> 1.00
	Input =>0.00, 0.00, 0.00, 0.00, 1.00, 1.00, 1.00, 0.00, converted=> 1.50
	Input =>0.00, 0.00, 0.00, 0.00, 1.00, 1.00, 1.00, 1.00, converted=> 2.00
	Input =>0.00, 0.00, 0.00, 0.00, 0.00, 1.00, 1.00, 1.00, converted=> 2.50
	Input =>0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 1.00, 1.00, converted=> 3.00
	Input =>0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 1.00, converted=> 3.50

Table 4: Duty cycle of left and right motors of the line following robot

Input	Weighted	Output	Left motor (Duty cycle) (%)	Right motor (Duty cycle) (%)
00000001	3.50	86.66	79	14
00000011	3.00	80.00	58	20
00000111	2.50	73.78	40	28
00001111	2.00	67.36	33	33
00011111	1.50	60.76	28	40
00111111	1.00	42.00	20	58
01111111	0.50	21.21	14	79
01111110	0.00	0.00	100	100
11111110	-0.50	-21.21	14	79
11111100	-1.00	-42.00	20	58
11111000	-1.50	-60.76	28	40
11110000	-2.00	-67.36	33	33
11100000	-2.50	-73.78	40	28
11000000	-3.00	-80.00	58	20
10000000	-3.50	-86.66	79	14

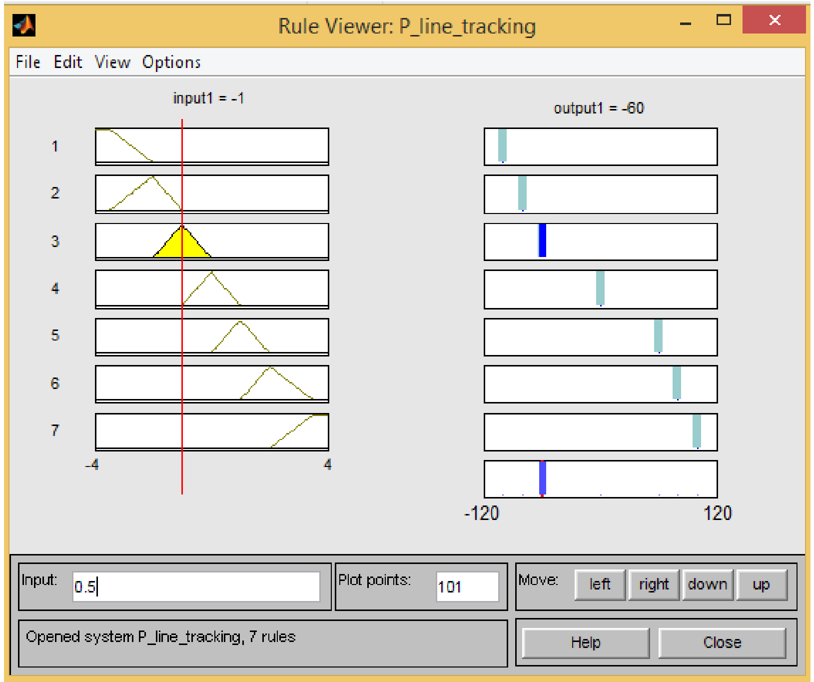


Fig. 9: Rule viewer when modified input is 0.5

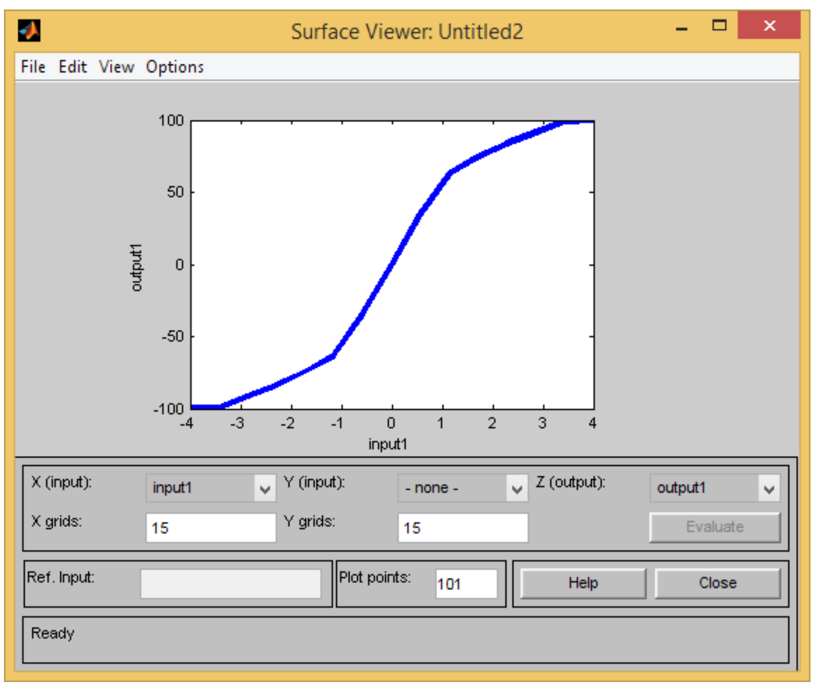


Fig. 10: Surface viewer of the system

The result in Table 4 is consistent to the result of simulation using the MATLAB package. The surface viewer of the system is simulated proves that the system is more sensitive when it detect left and right side as compared to center (Fig. 10). This in return, will control the direction of the AGV so that it will move in along the line at almost all time.

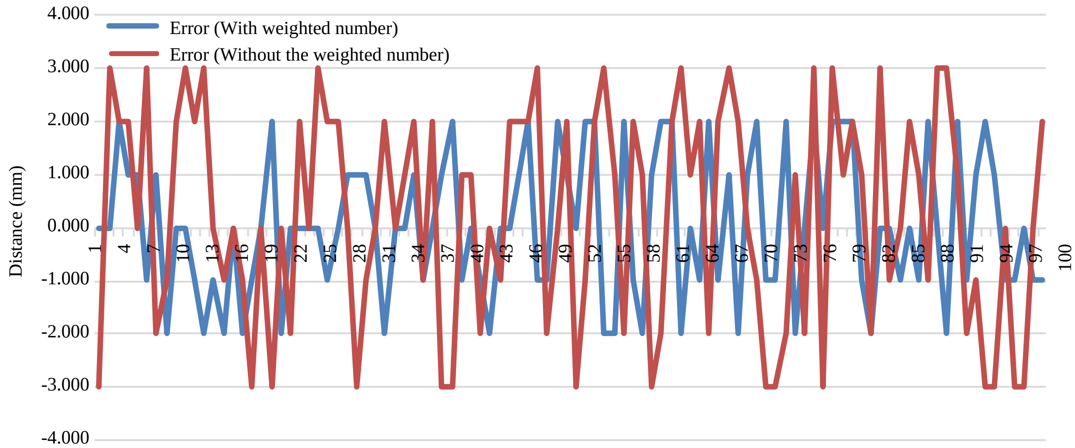


Fig. 11: Graph on system error with and without weighted number

Figure 11 shows that the two input with and without the weighted number, respectively. It is shown that the error is lesser when weighted number is used than the error without the weighted number used. The sum square error with weighted number is also smaller which is 1.351 mm as compared to 1.851 mm without the weighted number.

CONCLUSION

According to our simulation and experimental results, the designed fuzzy controller has verified its efficiency. Anyhow, the efficiency of the method can be greatly enhanced in the future. This can be done by incorporating a fuzzy set indicating the velocity (acceleration) and integrating the corresponding rules (knowledge) into the system. Our next step is to experiment other fuzzy inference techniques and to present a comparative study of the results. Also the controller will be amended in order to make possible the navigation within dynamic environments. The use of the fuzzy controller is not focus only to common static environments such as factories and warehouses, but can also be used in dynamic environments as well, where unknown moving objects are part of the environment.

ACKNOWLEDGMENT

The study reported in this article is the milestone of the UniKL-BMI research and development project titled Flexible Manufacturing System. The technical supports by colleagues and laboratory staff of UniKL-BMI are gratefully acknowledged.

REFERENCES

1. Carpi, F. and C. Pappone, 2009. Magnetic maneuvering of endoscopic capsules by means of a robotic navigation system. IEEE Trans. Biomed. Eng., 56: 1482-1490.
2. Riehle, T.H., S.M. Anderson, P.A. Lichter, N.A. Giudice and S.I. Sheikh et al., 2012. Indoor magnetic navigation for the blind. Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, August 28 September 1, 2012, San Diego, CA., pp: 1972-1975.
3. Bai, Y.L., L.J. Yang and C.H. Dong, 2011. Electromagnetic navigation smart car control system based on MC9S128. Measurement Control Technol., Vol. 11.
4. Arshad, N.M., M.F. Misnan and N.A. Razak, 2011. Single infra-red sensor technique for line-tracking autonomous mobile vehicle. Proceedings of the IEEE 7th International Colloquium on Signal Processing and its Applications, March 4-6, 2011, Penang, pp: 159-162.
5. Adam, I., A.H. Ali, A.A. Salam, Z. Zaharuddin and M.A. Dandu, 2005. Designing of fuzzy logic controller using PIC16 microchip controller. Proceedings of the UniKL-BMI International Seminar, November 9, 2005, Kuala Lumpur, Malaysia.