An Automatic Mobile Robot Obstacles Avoidance in a Static Environment by using Hybrid Technique (Fuzzy Logic and Artificial Neural Network)

Nur Hidayatul Nadihah Binti Abd Wahab,
Faculty of Electrical Engineering.
Universiti Teknologi MARA,
Shah Alam, Selangor
faida900@gmail.com

Nohaidda Binti Sariff
Faculty of Electrical Engineering,
Universiti Teknologi MARA,
Shah Alam, Selangor
Nohaiddasariff@yahoo.com

Abstract- **This project is focusing on testing the performances of the E-puck mobile robot to avoid obstacles in a static environment. Fuzzy Logic (FL) and Artificial Neural Network (ANN) have been used to improve the performances of the robot in term of time and smoothness of its path. The objective of this project is to design an autonomous mobile robot that can avoid obstacles in three different complexities of environment. Fuzzy logic is used to create rules of avoiding obstacles. The rules consist of the input from Infrared (IR) sensor and the output is from the speed of robot motor. Then the output is fed into ANN for training process to get better performance. The result shows that the objective of this project is successfully achieved. The overall performances of the E-puck robot shows that integration between these controllers capable to reduce the time taken by the robot avoiding the obstacles. Furthermore, the path is more accurate.**

Keywords: Mobile Robot, Fuzzy Logic Control, Artificial Neural Network, Obstacle Avoidance, Webots

I. INTRODUCTION

A mobile robot is an automatic machine that has the capability to move around in any given environment. In many applications, the robot is required to move without colliding with any obstacles. Therefore, the robot need to have autonomous navigation capabilities in order to complete the task. One approach to autonomous navigation is using model-based. It uses one design of the environment to generate a safe path to the target location [1]. Furthermore, there are many methods to control the mobile robot that make it intelligent in order to increase the performance especially in collision avoidance and path planning.

One of the approaches that mostly used among researchers is Fuzzy Logic (FL), because as a controller it shorten the time for engineering development in the case of highly complex system to avoid an obstacle. Artificial Neural Network (ANN) is one of the approaches to autonomous navigation that use computerized system and act as machine learning. It consists of many types of tool to learn from the past experiences similar to the human brain [2]. In ANN, the concept of MultiLayer perceptron is used to solve the nonlinear problems to get a better performance in obstacle avoidance instead of combining aids from the IR sensor which is more complex.

There are many kinds of methods that have been proposed by many researchers. Those methods, whether use single intelligent or combinations of two intelligent. Xiaochuan Wang and Simon X. Yang [3] have developed Neuro-fuzzy control system for obstacle avoidance of a nonholonomic mobile robot. They constructed eighteen fuzzy rules for the obstacle avoidance as the input and the output is the velocities of two separate driven wheels. Beside that, Gustavo Pessin, Fernando Osório, Alberto Y. Hata and Denis F. Wolf [4] give an idea to develop a multirobotic system by using Genetic Algorithm (GA) and ANN. In this project, GA has been used for planning, evolves positioning strategy. They developed a robot with a distance sensor that being controlled by a multilayer perceptron (MLP) in ANN. Its allows mobile robot to move in a dynamic environment with obstacle The results show that the ANN satisfactorily controls the mobile robot.

In addition, M. A. Jeffril and N. Sariff [5] also proposed the same concept which is used ANN and Fuzzy. They constructed fuzzy rules as the input by grouping IR sensor and the output is the speed of two wheels. Velappa Ganapathy, Soh Chin Yun, and Jefry Ng [6] have also developed mobile robot by using FL and ANN that focused on exploring the four combinations of training algorithms. They proposed the concept of Path Remembering that will make the mobile robot to come out from acute obstacles. Those projects proved that robot is able to avoid and explore in the environments based on Neural Network controller learning algorithm as well as increase the overall performance of the robot itself.

Regarding on those researches, it can summarize that the uses of Fuzzy Logic, Artificial Neural Network and other algorithms by using different method gives different performances for the desired output. Therefore, for this project, the integration of Fuzzy Logic and ANN is proposed to solve path planning obstacle avoidance problem in a static environment. In this project the fuzzy rules are constructed to use as a controller. The concept to take all the possibility for all sensors is used in constructing the fuzzy rules. Then, the ANN will be used to train the behaviour of the robot that corresponds to the IR sensors on the robot. This approach is focused on the ability of the robot to move as well as to avoid the obstacles in any different of complexity of the environment.

II. SCOPE OF WORK

This project is designed by using two main software which are Webots PRO and MATLAB. Webots PRO is used to simulate with a static environment avoiding obstacle. The performance of the mobile robot will be observed graphically in this software. In Webots PRO, C programming language is used for allowing user to control the robot. Another software is MATLAB which is functioned for constructing the Artificial Neural Network that will train the robot behavior. These two software will be interfaced together to achieve the project's objectives.

On this project E-puck robot is used. E-Puck size is 7.4cm in diameter, 4.5cm in height and weight about 150g. E-Puck has 8 infra-red sensors measuring ambient light and proximity of obstacles in a 4cm range. The environment with checkered floor in specified dimension is created in order to get the mobile robot moving freely in environment surround with obstacles. Three different environments with various obstacle locations were created to test the capability of the mobile robot. The obstacles are set to be solid square shapes, and spherical shapes. The locations of the obstacles in the each of the environments are randomly placed.

For the Fuzzy Logic (FL) controller, the fuzzy rules are constructed based on the eight IR sensors equipped at E-puck. The concept to take all the possibility for all sensors is used which considered all 256 rules to make the fuzzy rules for the E-puck. There are 4000 data in term of amount of light intensity were collected in each sensor. The collection of data is done in the tested environment which is then fed into an Artificial Neural Network (ANN) for the training process. The process of training ANN is done by using Matlab before it can be loaded into the E-puck again. Multi Layer Perceptrons is used for function approximation in ANN bringing in the Levenberg-Marquadt learning algorithm.

III. METHODOLOGY

The methods of this project are divided into four which are environment design, robot design, Fuzzy Logic (FL) design and Artificial Neural Network (ANN) design. The process sequences are as figure below.

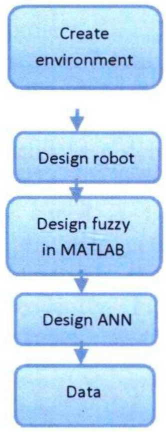


Figure 1: Process sequences of project

1. *Environment Design*

1.1 Properties of Environment

In Webots software, the new environment called world will be created in a checkered floor with basic world setting. A world, in Webots, is properties the environment. It contains a description of floor size, geometry, appearance (like color or brightness), and orientation. For this research, the properties of the environment as below:

- Checkered floor size 1 meter times 1 meter (1m × 1m) with Translation -0.5 0 -0.5, Rotation 0 1 0, 0, Scale 1 1 1, Contact Material "default".
- Viewpoint (fieldView) 0.785398 with orientation -0.964988 - 0.0991044 - 0.0357371 1.42999 position -0.184541 3.91117 0.718744 near 0.05.
- Sky colors 0.4 0.7 1 point light color 1 1 1 with intensity 1.

1.2 Creates the Obstacles

Below are the steps in creating the obstacles. Those steps are applicable in creating all shapes and obstacle within the environment.

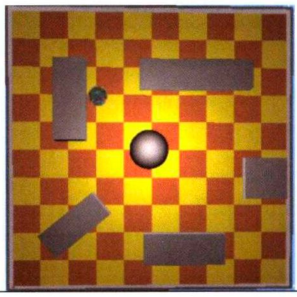
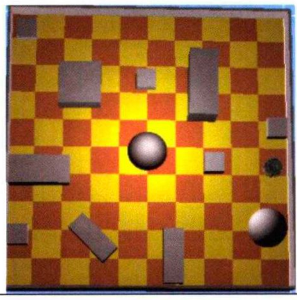
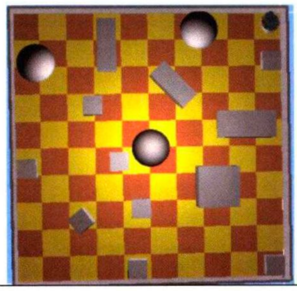
1. Select scene tree viewer.
2. Choose a Solid node.

3. Open this newly created Solid node from the + sign and type "wall" in its name field.
4. Select the children field and Insert after a Shape node.
5. Open this Shape, select its appearance field and create an Appearance.
6. Define the boundingObject field of the wall as to prevent the E-puck robot to pass through the walls.

1.3 Graphic User Interface

There are three environments have been created which are simple environment, medium environment and hard environment (refer to table 1). Simple environment consists of 5 numbers of the square and one spherical, while medium environment consists of 10 numbers of the square and 2 numbers of spherical. For hard environment, there are 12 numbers of square and 3 number of spheres.

Table 1: GUI of three environments

Complexity	GUI
Simple (testing) environment	
Medium environment	
Hard environment	

2. Robot Design

On this project E-puck robot is used. On observing the various types of robot that can be used in Webots software, the E-puck mobile robot is the one that has all the specifications needed for this project. Also from the scene tree viewer, 'PROTO (Webots)' node was selected and E-puck Differential Wheel was added. For the controller, it can be added to the 'controller' node. This selection is to enable the E-puck get into the environment with their controller choose by the user. The fuzzy logic method had been chosen to be the robot controller for this project.

For the E-puck size is 7.4cm in diameter, 4.5cm in height and weight about 150g. E-Puck has 8 infra-red sensors measuring ambient light and proximity of obstacles in a 4cm range. The speed unit of E-puck motor is 0.00628 rad/s and its maximum angular speed is 1000 units. The IR sensors are used for data collection in term of amount of light intensity. In each of the IR sensors is working independently. Figure 1 shows the structure of E-puck robot.

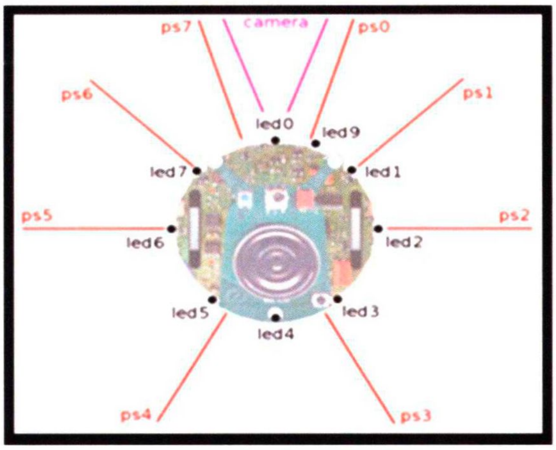


Figure 1: Sensors, LEDs and camera of E-puck

3. Fuzzy Logic Design

Mamdani FL Toolbox is used to create the FL rules. The input is eight IR sensors and the output is the modification of the speeds by the obstacle basic rules. The E-puck need to read the values of the IR sensor which represents the values of the amount of light that is between 0 to 3500.

Figure 2 represents the Mamdani systems using the FL tool box with 8 inputs, 256 rules and 5 outputs to perform the Fuzzy Inference System (FIS). Figure 3 shows the membership function of the input. The range is between 0 to 3500 and the threshold has been set to 800. In order to get the robot move follows the rules, the amount of light detection must exceed the threshold.

Table 2 shows the turning angle of every threshold that has been set up randomly for the output membership function (refer to Figure 4). The movements are forward, turn 45 degrees to the right, 45 degrees to the left, 90 degrees to the right and 90 degrees to the left. The range output values are between 0 to 1. Every threshold gives different E-puck turning angle. Table 3 shows the example of E-puck movement threshold while Figure 5 shows the list of the rules that indicate the movement of the robot. The ouput threshold is depend on the input threshold for the amount of light by detection of the IR sensor.

Table 2: The turning angle of every threshold

Turning angle (degree)	Threshold
Forward 0	0.375 < x < 0.625
Negative 90	X > 0.125
Negative 45	0.125 < x < 0.375
Positive 45	0.625 < x < 0.875

For this project, the rules have been set randomly which is 15 to 256 numbers of rules. Thus, the performance of the robot will be different that will be shown in section four (Result).

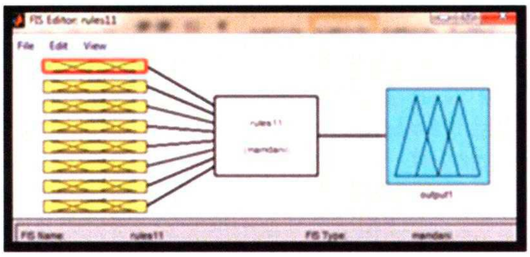


Figure 2: Mamdani Fuzzy Logic Toolbox

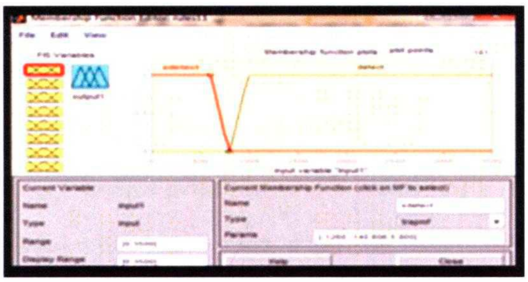


Figure 3: Membership function of the input

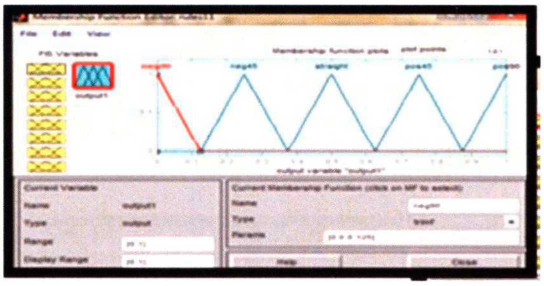


Figure 4: Membership function of the output

Figure 5: Fuzzy Logic set rules

4. Artificial Neural Network Design

ANN is introduced to make the mobile robot more intelligent. The output from FL is then being fed into the ANN for the training process. In ANN the concept of MultiLayer perceptron (Figure 6) has been used for the mobile robot system is a nonlinear system. The network consists of layers of hidden unit which enable the MLP to learn complex task.

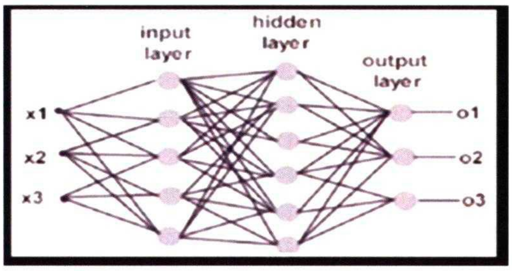


Figure 6: Structure of an MLP 3layers

For this project MLPs is used for Function Approximation. The Function Approximation process automatically uses a learning algorithm named Levenberg-Marquadt. This purpose is a task that performed by network trained to respond to input with an approximation of a desired function. There are several steps that must be followed to create the MLPs. The first step is setting the dataset division parameters. This dataset is from the output of fuzzy logic controller. The dataset is divided into training (adjustment of MLPs weight), testing (testing the trained MLP) and validation (check data periodically to avoid overhit) sets). The next step is train MLP and finding the optimal MLP parameters. The number of hidden units, number of epochs and input/output lag spaces can be adjusted. The next step is simulating the dataset and validate result. To be accepted as a suitable model, several methods need to be done to ensure a good model fit, as well as random and uncorrelated residual. The model can be accepted only if those tests are passed. They are the autocorrelation test, cross-correlation test and the histogram test.

For this project those following specifications were used for this algorithm:

- No of inputs: 8 no of input from IR sensor.
- No of outputs: one output represents all the movement.
- No of hidden layer: 5
- Division of dataset: 80% for training, 20% for testing and non for validation.

5. Testing Overall E-puck Performances

There are five testing that has been done for this project. There are:

1) E-puck performance while avoiding the obstacles in the simple (testing) environment. The FL controller will be tested before being combined with ANN.

2) E-puck performance robot in term of analysis of the trajectory will be observed in different numbers of rules in the simple (testing) environment. The different number of FL rules give the effect to the performance of the robot. The comparison has been done between 15 numbers of rules and 256 number of rules.

3) Performances of data in ANN as a valid system. The data that will be analyzed are a good model fit, as well as random and uncorrelated residual, one step prediction in training and testing, the regression line, the overall performance of the neural network controller.

4) E-puck performance in three different complexity of the environment by using hybrid performances. The robot will be observed based on their trajectory and time taken in 4000 numbers of loops.

IV. RESULT

1) E-puck performance (FL controller)

Based on the observation in the tested environment, the movement of the E - puck robot with FL controller in term of avoiding obstacle are successfully following the rules. When the IR camera detects the amount of light and reach over the threshold, it smoothly turns as its rules. The degree of E-puck movement depends on which sensor senses the obstacles. If input 1 (1) and input 2 (1), the speed of two wheels will be -300 and +300 turning 90 degrees to the left. For the example, refer to Table 3.

Table 3: Example of fuzzy rules

	ps0	ps1	ps2	ps3	ps4	ps5	ps6	ps7	Output
1	0	0	0	0	0	0	0	0	Forward
2	1	0	0	0	0	0	0	0	90° turning
3	0	1	0	0	0	0	0	0	90° turning
4	1	1	0	0	0	0	0	0	-90° turning
5	0	0	1	0	0	0	0	0	Forward
6	1	0	1	0	0	0	0	0	-90° turning
7	0	1	1	0	0	0	0	0	-90° turning
8	1	1	1	0	0	0	0	0	-90° turning
9	0	0	0	1	0	0	0	0	Forward
10	1	0	0	1	0	0	0	0	45° turning
11	0	1	0	1	0	0	0	0	45° turning
12	1	1	0	1	0	0	0	0	45° turning

Table 4 shows the threshold value (distance sensors) and speed of both wheels have been used in the fuzzy rules. It shows that when the distance between the robot and the obstacles is far away, the sensor value will be low in a range of 0.375 and 0.625 (refer Figure 7). Otherwise, the value will be high in the range of 0.125 and above and that will make the E-puck move 90 degrees to the left with +- 300 of speed to avoid the obstacle (refer to Figure 8). Then, after E-puck robot move 45 degrees to the left (refer Figure 9), the range of distance sensor is between 0.125 and 0.375 with +- 100 and then move straight with speed +- 500. Based on the figure 10 (the robot firstly meet the obstacle) to figure 11 (the turning to the left, avoiding the obstacle), it shows that E-puck can successfully avoid the obstacle follows the rules.

Table 4: E-puck movement threshold and speed of both wheels

Turning angle (degree)	Threshold	Speed of left wheel	Speed of right wheel
Forward 0	0.375 < x < 0.625	+500	+500
Negative 90	X > 0.125	-300	+300
Negative 45	0.125 < x < 0.375	-100	+100
Positive 45	0.625 < x < 0.875	-100	+100
Positive 90	X > 0.9	-300	+300

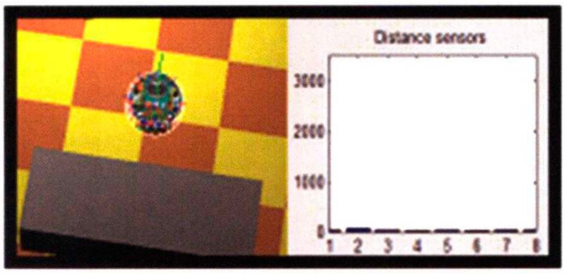


Figure 7: Webots PRO GUI

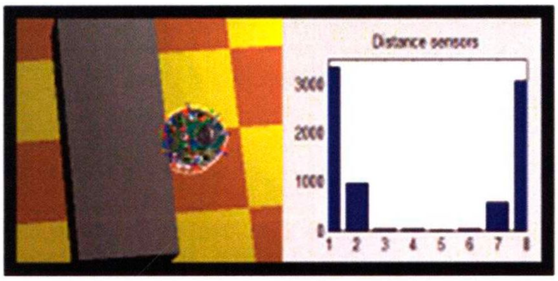


Figure 8: Webots PRO GUI

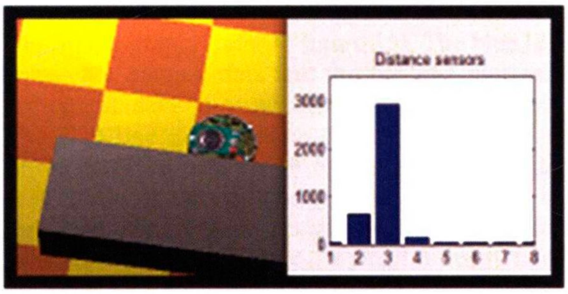


Figure 9: Webots PRO GUI

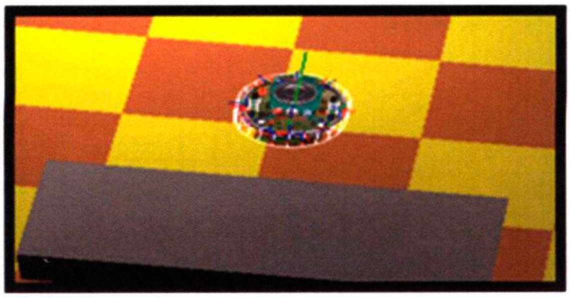


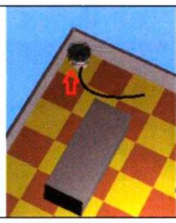
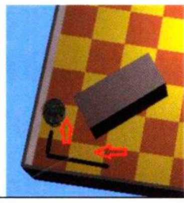
Figure 10: Webots PRO GUI

Figure 11: Webots PRO GUI

2) Performances of two different numbers of FL rules

The robot is successfully avoided the solid square and solid spherical shape for both methods, but when it's facing the edge of the wall, the fifteen rules do not work. The robot will stuck and not turning either left or right. By taking 256 rules, all the IR sensors are working independently and the possibility for all sensors is being measured in all angles of its movement. So that, it did not face any local minima problem. The analysis is in the Table 5 below.

Table 5: Comparison between different number of rules

Number of rules	Trajectory	Analysis
15 no of rules		Local minima problem
256 no of rules		Successfully avoided all obstacle

3) Performances of data in ANN

Figure 12, 13 and 14 below shows the result for residual testing (autocorrelation, cross correlation and histogram). All the data reach 95% confidence level, which shows that the data is valid.

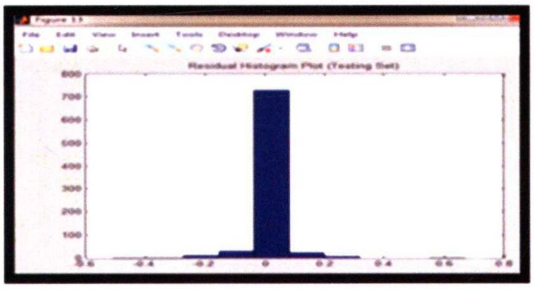


Figure 12: Histogram test

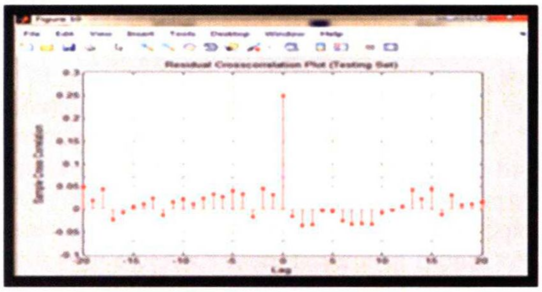


Figure 13: Crosscorrelation test

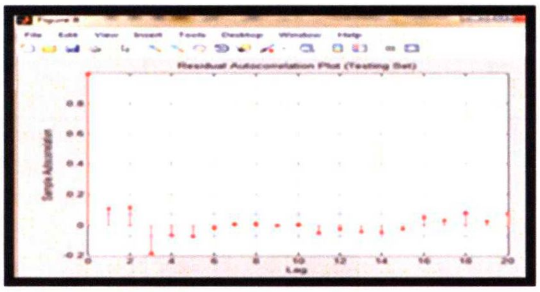


Figure 14: Autocorrelation test

From the graph, (refer to figure 15). The blue line represents training data and the red line represent the output data after go though training process. The blue line almost same to the red line which shows that the path made by the robot by using fuzzy logic is same with the path in ANN. It can be said that the data is perfect as the wave line between FL and ANN give the small differences as shown in figure 2.

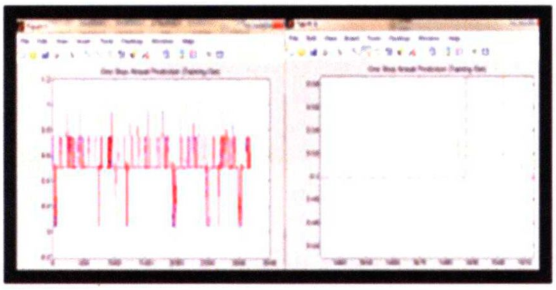


Figure 15: One step ahead prediction training set

Figure 16 shows that the actual testing data versus the output obtained from the neural network controller. The blue line represents actual training data while the red line represents output testing data. The graph shows that the blue and the red line are almost the same which means the testing result are followed as trained.

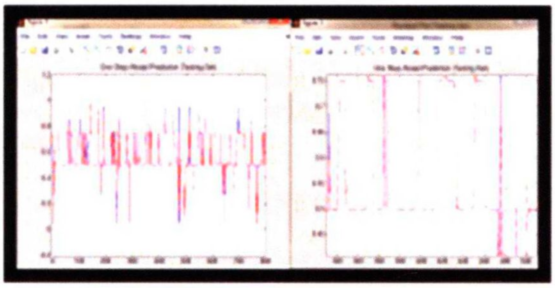


Figure 16: One step ahead prediction testing set

Figure 17 shows the regression line for all the data from training process and testing process. The graph shows that regression line has followed the dotted line that represent desired value. Based on the figure it shows that the value of R for overall is more than 95%, which is proved that result obtain for this project is acceptable.

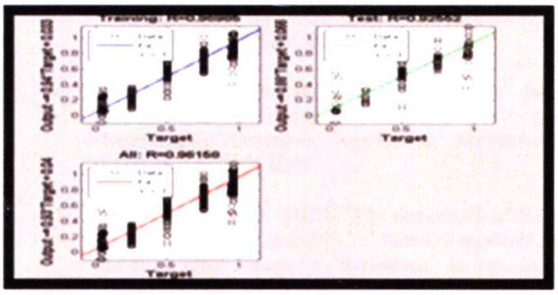


Figure 17: Regression

Figure 18 below shows the best training, performance graph that being a plot of mean square error (mse) versus epochs. The best performance training is 0.0016014 at epoch 27. The MSE is small which mean it is closer to the output match. It is used to evaluate the neural network output error between actual and the expected output. Performance is lower when epoch is high. The difference between the actual and the target is almost the same.

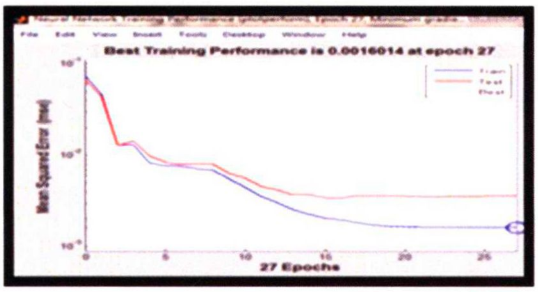
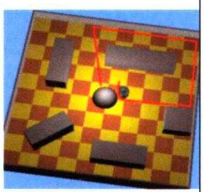
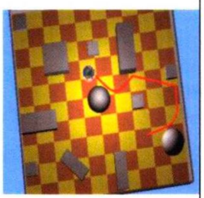
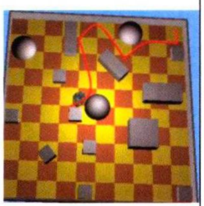


Figure 18: Best Training Performance

4) Performance of E-puck (FL+ANN)

The performances of E-puck robot show some difference between the robot with fuzzy logic controller and robot with FL+ANN controller. The E-puck robot took about almost 4 minutes to finish 4000 loops depends on their case study. The overall result shows that the time taken for the E-puck robot with FL+ANN more faster than E-puck robot without ANN. The result is shown in the table below:

Case study/Metho d		Time taken FL	Time taken FL+AN N
Simple (tested environment)		4:15:93 6	4:15:648
Medium environment		4:16:00 0	4:15:808
Hard environment		4:16:00 0	4:15:744

V. CONCLUSION

As a conclusion, this development of robot by using Fuzzy Logic (FL) and Artificial Neural Network (ANN) gives the best performances in term of efficiency, flexibility and robustness. Based on the result obtained, the objectives of this project are achieved. The E-puck robot is successfully avoiding any static obstacle in three different complexity of the environment. The time taken by the robot to finish the 4000 numbers of loops is also reduced. In addition, the concept of taking all the possibilities for all sensors which considered all 256 rules give high impact to the movement of the robot compared to only take a few rules in random sensors.

However, for future recommendation, this development can apply to the environment which has a target. By combining with the global path planning system such as A'algorithm, Genetic Algorithm and Ant Colony Optimization it can be one complete path planning for robot navigation system.

VI. REFERENCES

1) Mariam Al-Sagban and Rached Dhaouadi, "Neural-Based Navigation of a Differential-Drive Mobile Robot," in *2012 12th International Conference on Control, Automation, Robotics & Vision Guangzhou, China,* 5-7th December 2012 (ICARCV 2012).

2) R. R. Janghell, AnupamShukla, RituTiwari and Rahul Kala, "Breast Cancer Diagnosis using Artificial Neural Network Models," in *2010 International Conference on Soft Computing and Pattern Recognition on, 2010 IEEE, pp 89-94.*

3) Xiaochuan Wang and Simon X. Yang, "A Neuro-Fuzzy Approach to Obstacle Avoidance of a Nonholonomic Mobile Robot," *in 2003 Intemalional Conference on Advanced Intelligent MechatronicS (AIM 2W3, 2003 IEEEIASME.*

4) Gustavo Pessin, Fernando Osório, Alberto Y. Hata and Denis F. Wolf, "Intelligent Control and Evolutionary Strategies Applied to Multirobotic Systems, *" in 2010 IEEE.*

5) M. A. Jeffril and N. Sariff, "The integration of fuzzy logic and artificial neural network methods for mobile robot obstacle avoidance in a static environment," *in System Engineering and Technology (ICSET), 2013 IEEE 3rd International Conference on, 2013, pp. 325-330.*

6) Velappa Ganapathy, Soh Chin Yun, and Jefry Ng, "Fuzzy and Neural Controllers for Acute Obstacle Avoidance in Mobile Robot Navigation," *in 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics Suntec Convention and Exhibition Center Singapore, July 14-17,* 2009

7) T. Khelchandra, H. Jie, and S. Debnath, " Path Planning of Mobile Robot with Neuro-Fuzzy Technique," *in 2012 12th Internasional Conference on Intelligent Systems Design and Applications (ISDA).*

8) Hema,Paulraj M, Abdul Hamid, K.F.Sim and Rajkumar Palaniappan, "An Intelligent vision System for Object Localization and Obstacle Avoidance for an Indoor Service Robot," *in 2011 IEEE Student Conference on Research and Development.*