American Journal of Sciences and Engineering Research

E-ISSN -2348 – 703X, Volume 6, Issue 3, 2023

Designing a Fuzzy Controller for Wall-Following Mobile Robot

Le Ngoc Dung

Faculty of Accounting and Finance, Dong Nai Technology University, Dong Nai Province, Vietnam.

ABSTRACT: *Fuzzy control is an intelligent control method that emulates the uncertain information processing and decision-making processes of humans. This approach is highly suitable for controlling complex objects that lack a clear mathematical model. In this research, a Proportional-Derivative (PD) fuzzy controller was designed and investigated for application in controlling a wall-following mobile robot. The input signals to the fuzzy controller are the distance error between the robot and the wall, and the rate of change of the distance error. The output signals are the left wheel velocity ωL and the right wheel velocity ωR. A simulation model was constructed using Matlab software, with the objective of controlling the mobile robot to maintain a stable distance from the wall. The simulation results demonstrated the feasibility and effectiveness of employing the proposed motion control method for a two-wheeled mobile robot in wall-following scenarios.*

Keywords: *Wall-following control, Fuzzy logic controller, Mobile Robot Control*

I. INTRODUCTION

Nowadays, in the field of automatic control, fuzzy control systems are widely used due to their advantages, such as reducing the workload by eliminating the need for object models. For complex design problems, using a fuzzy controller allows for a reduction in computational complexity and product cost. Autonomous robot control is a typical example, as it has significant applications in various fields such as painting robots or welding robots. These robots can move at high speed and accuracy to perform repetitive tasks. The challenge is how to make the robot follow a specific trajectory.

In the research of Daniela Coman and authors Ching-Chang Wong and colleagues, conventional robot control involves methods for path generation and path tracking. Controlling the motion of a robot along a path is a difficult task, especially in real-time and high-speed control. To achieve good tracking performance, these authors proposed a path-following approach based on fuzzy logic [1], [2].

In the study by authors Reinhard Braunstingl, Pedro Sanz, and Jose Manuel Ezkerra, a new approach to the control of wall-following mobile robots was presented. The local path planning is based on the concept of overall perception of the mobile robot. The fuzzy controller utilizes information from these concepts to control the mobile robot to run along walls and avoid obstacles [3].

Author Cheng-Hung Chen and colleagues also proposed Wall-Following Mobile Robot Control using Fuzzy Logic Controller (FLC) with improved error search and reinforcement learning. In this study, the fuzzy logic controller was improved by the error search capability through the FLC_R-IDS algorithm, which uses adaptive parameters to adjust the control parameters. Additionally, reinforcement learning was employed to guide the robot's behavior [4].

In this research, the author proposed the design of a PD fuzzy controller that only requires one feedback parameter, which is the distance error. The fuzzy controller utilizes this error to make appropriate decisions for the speed of the left and right wheels of the mobile robot, enabling efficient movement while maintaining a

 Received-18-06-2023, Accepted- 28-06-2023

distance from the wall. Successfully implementing the control of a two-wheel mobile robot using a fuzzy controller is highly significant in terms of scientific research and practical applications.

II. MOBILE ROBOT CONTROL SYSTEM

A. Mobile Robot physical model [5]

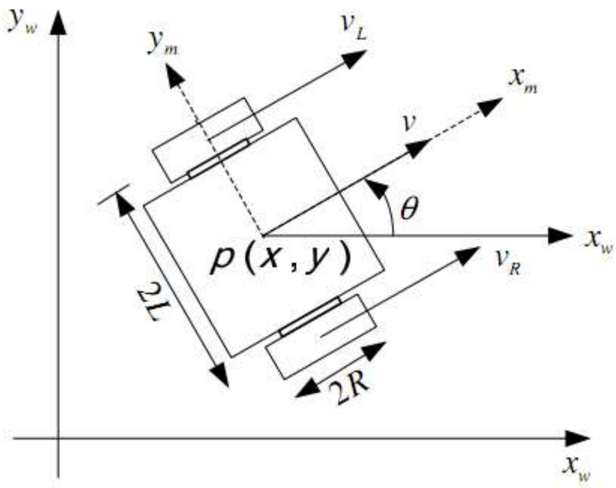

Figure 1: Sketch of a two-wheeled mobile robot

Notes: X_w,Y_w : global coordinate system

x_m, y_m : the local coordinate system mounted on the robot

p : robot body center

R : wheel radius

L : distance from center of the robot to center of the wheel

(x, y) : position of center p

θ Anti-clockwise angle between x_m and X_w

B. Descriptive Mathematical Model of Mobile Robot

The kinematic equation of the two-wheeled mobile robot [2]:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix} \quad (1)$$

Notes:

$\dot{x}, \dot{y}$ are the speeds of the robot along the X_w and Y_w axis;

v is the linear velocity of the robot;

$\dot{\theta} = \omega$ is the angular velocity of the robot.

$\dot{\theta}$ >0 when the robot rotates counterclockwise;

$\dot{\theta}$ <0 when the robot rotates clockwise.

The linear velocity v and the angular velocity ω are described as follows:

$$\begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2L} & \frac{1}{2L} \end{bmatrix} \begin{bmatrix} v_L \\ v_R \end{bmatrix} \quad (2)$$

Notes:

$v_L = Kc.R\omega_L$ is the speed of the left wheel of the mobile robot

$v_R = Kc.R\omega_R$ is the velocity of the right wheel of the mobile robot

ω_L : Mobile robot left wheel angular velocity

ω_R : Mobile robot right angular velocity

From (1) and (2) we deduce: $$\begin{bmatrix} v_L \\ v_R \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & L \\ -\cos\theta & -\sin\theta & L \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} \tag{3}$$

This is also the mathematical model used to model mobile robots

C. Control target

Control the mobile robot to follow the wall with a given distance d between the robot and the wall, control the mobile robot so that d remains constant during the robot's motion. At the same time, the mobile robot can follow walls of different shapes such as circles, parabolic shapes.

D. Mobile robot fuzzy controller design

- Approach plan:

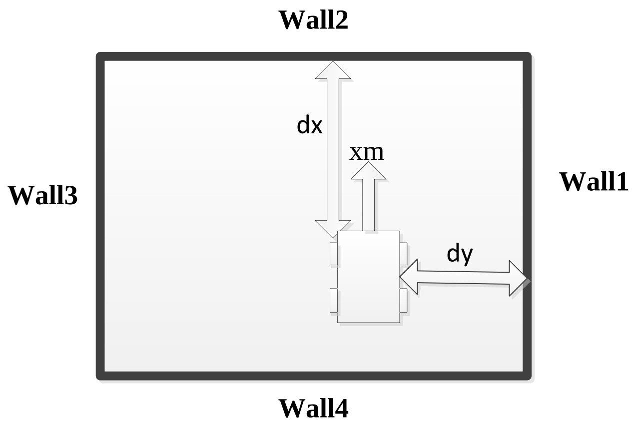

Figure 2: Approach of mobile wall-clinging robot

As mentioned above, we are equipped with an ultrasonic sensor along the xm axis of the robot. This sensor is responsible for measuring the distance dx (the distance from the robot head to the obstacle that is the wall in front of it). The robot's ym-axis ultrasonic sensor is responsible for measuring the distance dy (the distance from the right side of the robot to the obstacle that is the wall), which is also the distance the robot joystick follows. Thus, the fuzzy controller proposed to build the robot control is essentially a PD fuzzy set that only needs 1 feedback parameter, which is the error of the distance. The fuzzy controller relies on this error to make decisions that are suitable for the control goal, specifically:

- The controller input consists of 2 variables:

+ Error of Ed position is described by fuzzy set {NB, ZE, PB}

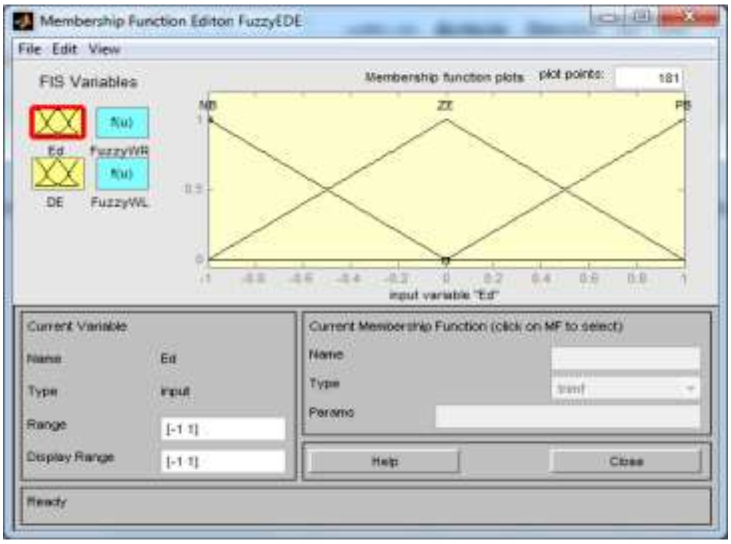

Figure 3: Function of position error Ed of fuzzy set PD

Here we normalize the fuzzy set to the range [-1 1] by multiplying the distance error value with **Kd = 1/Edmax**

$$Ed = \begin{cases} -Ed\max & if\ Ed < -Edmax \\ Ed & if\ Edmax \geq Ed \geq -Edmax \\ Edmax & if\ Ed > Edmax \end{cases}$$

+ The rate of variation of DE error is described by fuzzy set {NB, ZE, PB}

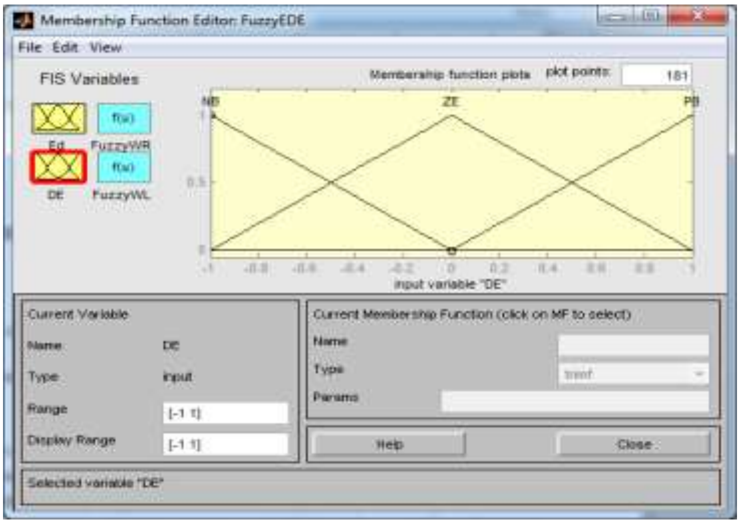

Figure 4: Function of rate variation of error DE of fuzzy set PD

Similarly, we normalize the fuzzy set to the range [-1 1] by multiplying the angular error value with **KDE = 1/ DEmax**

$$\mathrm{DE} = \begin{cases} -\mathrm{DEmax}\ if\ \mathrm{DE} < -\mathrm{DEmax} \\ \mathrm{DE} & if\ \mathrm{DE}max \geq \mathrm{DE} \geq -\mathrm{DE}max \\ \mathrm{DE}max & if\ \mathrm{DE} > \mathrm{DE}max \end{cases}$$

- The output consists of 2 variables:

The output of the fuzzy controller is left wheel speed ω_L and right wheel speed ω_R are Singleton lines described by 3 bars {NB, ZE,PB}={-1,0,1}. Similar to inputs when designing, they are also normalized to the range [-1 1]. The output value will be multiplied by the Kout value. Kout is the maximum speed value that we control 2 wheels of the Robot.

- Constructing a composition law:

Suppose to control the robot to stick to wall 1 with a distance d as shown in Figure 5

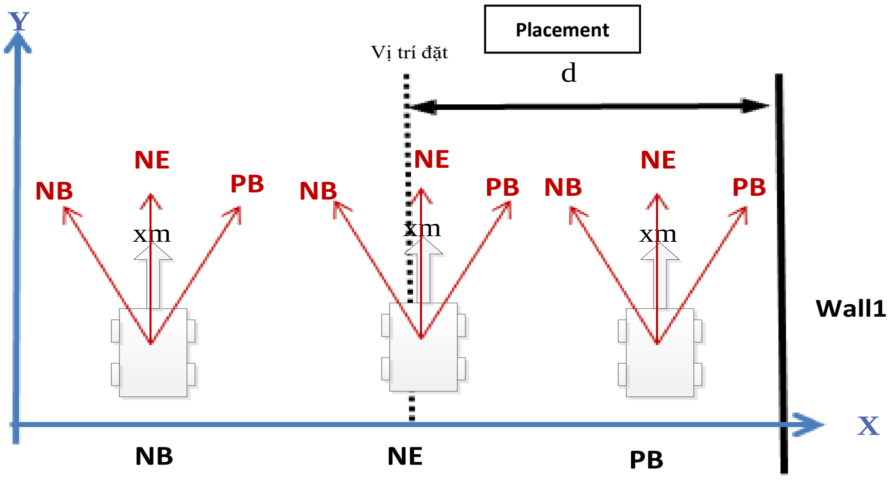

Figure 5: Robot position analysis diagram to design fuzzy set rule PD

Redefine error Distance **Ed = Robot position - Wall position + d**

- **Figure on the left {Ed=NB, DE=NB}**: the robot is to the left of the reference position and tends to move away from the reference position (in the direction NB derivative of the Ed function). Therefore, a strong control law is needed to make the robot rotate quickly to return to the reference position. Therefore, we control the right wheel to turn backward (NB) and the left wheel to turn forward (PB) **(ωR= NB, ωL=PB), (rule 1).**
- **Figure on the left {Ed=NB, DE =ZE}**: The robot is to the left of the reference position and tends to move parallel to the reference position (in the ZE direction, derivative of Ed=0→ the error does not change → the robot goes straight). Therefore, we need a control law that makes the robot return in the direction that makes the robot close to the reference position (rotation must be slower than rule 1). Control the right wheel to stay stationary (ZE) and the left wheel to rotate to (PB) **(ωR= ZE, ωL=PB), (rule 2).**
- **Left figure {Ed=NB, DE=PB}**: the robot is to the left of the reference position and tends to move closer to the reference position (in the direction of PB → direction of the positive Ed derivative). Therefore, we need a control law to make the robot go straight back to the reference position quickly. Control 2 wheels to move quickly forward **(ωR= PB, ωL=PB), (rule 3).**
- **The middle figure {Ed=ZE, DE=NB}**: the robot is moving on the reference position but tends to move to the left (in the NB direction, which makes the DE derivative error negative). Therefore, we need a control law to make the robot return to the reference position (right rotation). Controls the right wheel to stay stationary (ZE) and the left wheel to rotate to (PB) **(ωR= ZE, ωL=PB), (rule 4).**
- **The middle figure {Ed=ZE, DE=ZE }:** The robot is moving at the right reference position and in the same direction of the reference trajectory (the direction turns the Ed zero → error Ed=ZE constant). At this time, the robot only needs to run straight until it touches the wall in front and make a corner in the direction of the wall in front **(ωR= PB, ωL=PB), (rule 5).**
- **Middle figure {Ed=ZE, DE=PB}**: The robot is moving on the reference position but tends to move to the right (in the direction PB the direction makes the derivative of Ed positive). Therefore, we need a control law to make the robot return to the reference position (left rotation). Controls the right wheel to go forward (PB) and the left wheel to stay stationary (ZE) **(ωR= PB, ωL=ZE), (Rule 6).**
- **Figure on the right {Ed=PB, DE=NB}**: the robot is to the right of the reference position and tends to move closer to the reference position (in the NB direction, which makes the derivative of Ed negative). Therefore, we need a control law to make the robot go straight back to the reference position quickly. Control 2 wheels to move quickly forward **(ωR= PB, ωL=PB), (Rule 7).**
- **Figure on the right {Ed=PB, DE=ZE}**: the robot is to the right of the reference position and tends to move parallel to the reference position (in the ZE direction which makes the Ed derivative equal to zero error

Ed). unchanged). Therefore, we need a control law to make the robot return in the direction that makes the robot close to the reference position Ed=ZE (left turn is slower than rule 7). Controls the right wheel to rotate to (PB) and the left wheel to stay stationary (ZE) **(ωR= PB, ωL=ZE), (Rule 8).**

- **Figure on the right {Ed=PB, DE=PB}**: the robot is to the right of the reference position and tends to move away from the reference position (in the direction of PB the direction makes the derivative of Ed positive). So we need a strong control law to make the robot turn left quickly back near the reference position. Therefore, we control the right wheel to turn forward (PB) and the left wheel to rotate backward (NB) **(ωR= PB, ωL=NB), (law 9).**

From the above analysis, we have the fuzzy control rule according to Table 1

Table 1: Fuzzy control rule of PD controller

DE / Ed	NB	ZE	PB	Right/Left wheel speed
NB	NB PB	ZE PB	PB PB	ωR ωL
ZE	ZE PB	PB PB	PB ZE	ωR ωL
PB	PB PB	PB ZE	PB NB	ωR ωL

- PD fuzzy controller for Mobile robot as shown in Figure 6

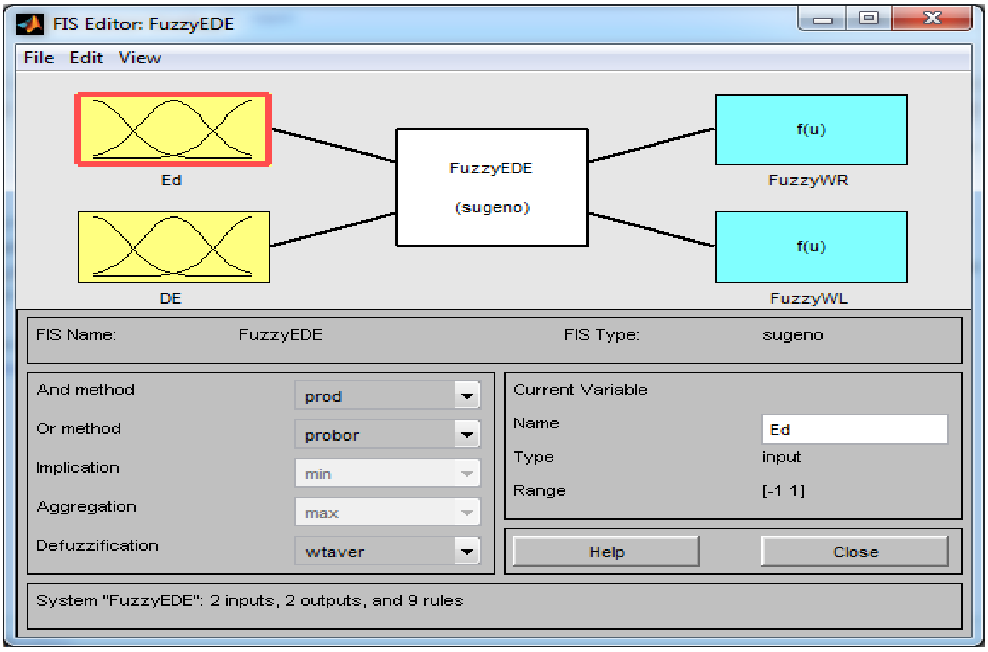

Figure 6: PD fuzzy controller for Mobile robot

- Simulation diagram of Mobile robot built on Matlab Simulink software as shown in Figure 7

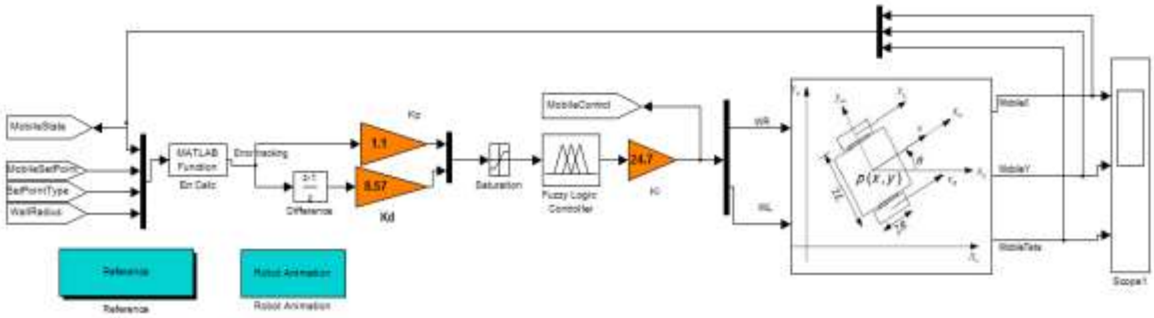

Figure 7. Simulation diagram of Mobile robot using PD controller

III. SIMULATION RESULTS

The fuzzy controller PD keeps the Mobile robot following the trajectories of a straight line, a rectangle and a circle with the distance between the wall and the robot is d = 0.5 (given).

+ The PD fuzzy set controls the robot to follow a straight wall as shown in Figure 8.

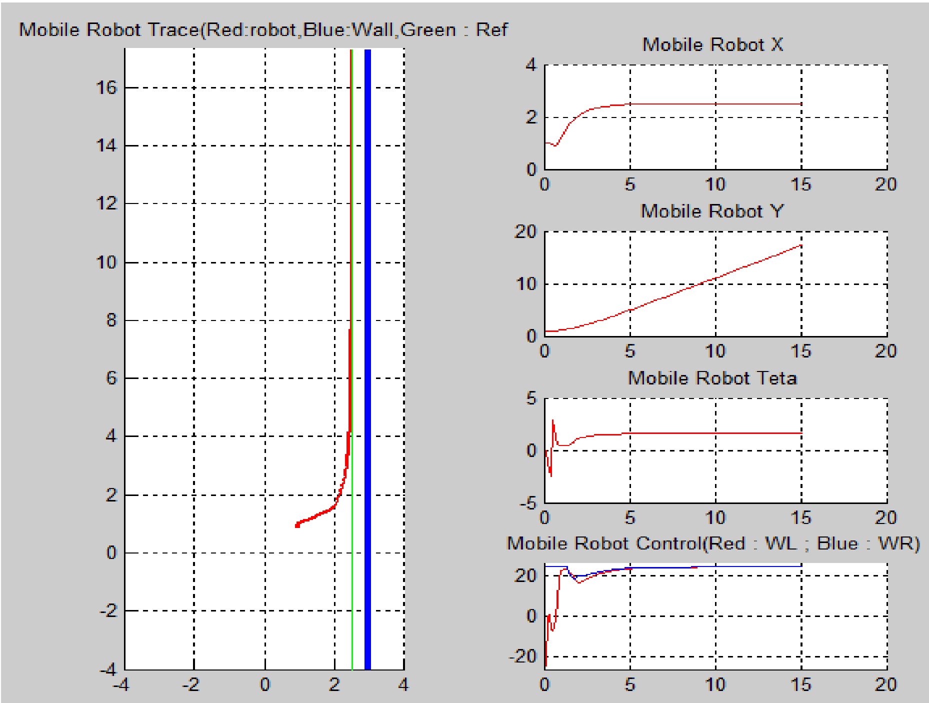

Figure 8. Mobile Robot following a straight wall results

+ The PD fuzzy set controls the robot to follow 4 straight walls forming a rectangle as shown in Figure 9

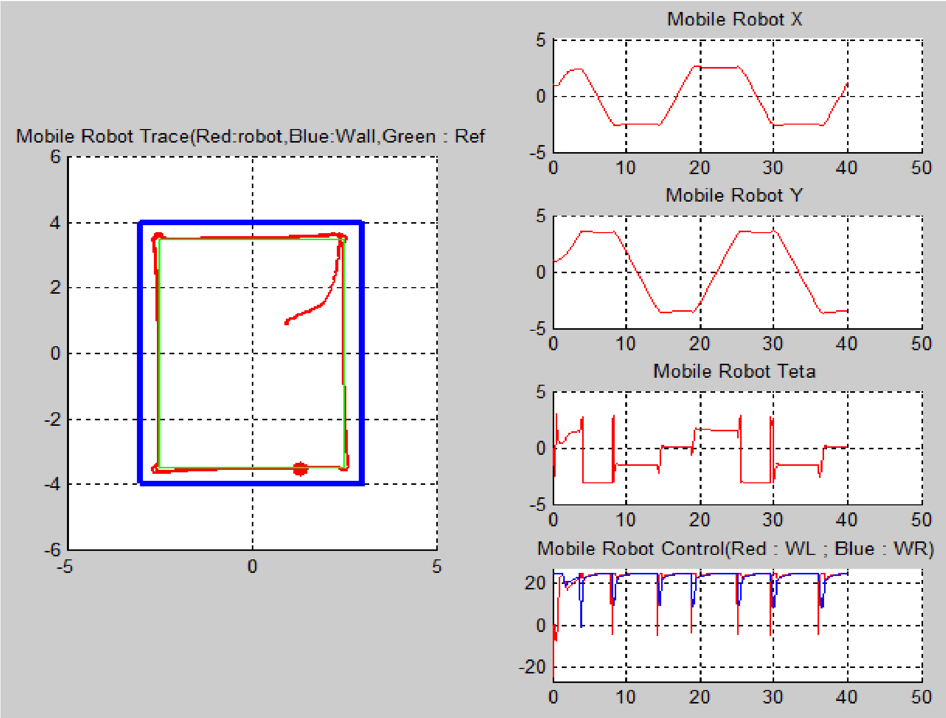

Figure 9. Mobile Robot follows to a rectangular wall

+ The fuzzy set controls the robot to follow the wall with radius R as shown in Figure 10

Figure 10. Mobile Robot clings to a circular wall

IV. CONCLUSION

The author has successfully researched and designed a Sugeno fuzzy set with input and ED (distance error) and DE (displacement error variable speed), determining the direction of motion of the robot through speed. DE error (as analyzed in the design of fuzzy rules). Therefore, we avoid having to determine the direction of the trajectory so that the robot can move along the trajectory without relying on sensors to determine the direction angle relative to the obstacle. Simultaneously, the simulation results show that the Mobile robot works stably along the wall with straight lines, rectangles and circles. However, there are a few problems in the implementation process:

The research and construction of the control fuzzy set for Mobile robot has been through experience and unsuccessful attempts, so it takes a long time but the results are not really optimal.
Failure to determine the error in the direction of motion can also cause the robot to overshoot its trajectory and in case of poor control the robot will lose control.
I propose more studies to determine the error of motion direction of the robot and the application of optimization algorithms such as PSO and GA to optimize the parameters for better fuzzy controllers.

V. REFERENCES

[1]. Ching-Chang Wong, Hoi-Yi Wang, Shih-An Li, and Chi-Tai Cheng. Fuzzy Logic Controller For Wheeled Mobile Robots. International Journal of Engineering Research & Technology, Vol. 2 Issue 7, July - 2013.
[2]. Daniela coman. Mobile Robot Path Tracking using Fuzzy Logic. Proceedings of the 13th WSEAS International Conference on SYSTEMS.
[3]. Reinhard Braunstingl - Pedro Sanz - Jose Manuel Ezkerra, Fuzzy Logic Wall Following of a Mobile Robot Based on the Concept of General Perception. International conference on advanced robotics, sant feliu de guixols, Spain, pp.367-376, sept., 1995.
[4]. Cheng-Hung Chen, Shiou-Yun Jeng and Cheng-Jian Lin. Mobile Robot Wall-Following Control Using Fuzzy Logic Controller with Improved Differential Searchand Reinforcement Learning. Mathematics 2020, 8, 1254; doi:10.3390/math8081254.
[5]. Ching-Chang Wong, Hou-Yi Wang, and Shih-An Li. PSO-based Motion Fuzzy Controller Design for Mobile Robots. International Journal of Fuzzy Systems, Vol. 10, No. 1, March 2008.