Fuzzy-Pid Logic Control On Using Rplidar A2 At The Quadruped Robot To Apply The Hexagonal Trackway

Diono[1], Muhammad Jaka Wimbang Wicaksono[2], Vivin Octowinandi[3] , Muhammad Raihan[4]

{diono@polibatam.ac.id[1], jakawimbang@polibatam.ac.id [2], vivin@polibatam.ac.id[3], emailraihan763@gmail.com[4]}

Batam State Polytechnic, Electrical Engineering Department, Indonesia[1,2,3,4]

Abstract. This paper is shown for development at the KRI (Indonesian Robot Contest) competition in the division of the Indonesian Fire Fighting Robot Contest organized by the Ministry of Research and Technology Higher Education (KEMENRISTEKDIKTI) presenting a wall design that follows the behaviour of a four-legged robot based on a Fuzzy-PID controller. The PID controller is proposed here because of its ability to control many cases of non- linear systems, and Fuzzy is added to it for auto-tuning. In this case, we propose a PID controller for the movement stability of a Quadruped robot while following a hexagonal wall. In this paper, the fuzzy-PID controller is used to control the legs of the robot by adjusting the swing angle value when moving forward to maintain the distance between the robot and the wall. The experimental results are verified by applying the proposed control method on a prototype quadruped robot.

Keywords: Fuzzy PID, Quadruped Robot, Wall Following, legged locomotion, Hexagonal Wall

1 Introduction

Mobile robots can be broadly classified into three categories: wheeled, tracked, and legged. Walking robots are the second choice of wheeled robots. Most of the land on earth is not flat. Walking robots are slower than wheeled robots, but walking robots have the important advantage of moving over uneven terrain, whereas wheeled robots are closely associated with flatter surfaces. The quadruped robot has four legs controlled by 12 servo motors. According to the nature of stability, the gait of a quadruped robot can be classified into static and dynamic. Static gait focuses on stability, while dynamic gait focuses on fast and efficient things [1]. The quadruped robot moves based on the motion kinematics applied to each of its legs, which are governed by servo motors [2].

At the KRI (Indonesian Robot Contest), the extinguishing and rescue robot division has

ICAE 2023, November 07, Batam, Indonesia

DOI 10.4108/eai.7-11-2023.2342950

challenges that change every year, from the challenges that are passed to the walls that are not straight. This challenge makes the robot able to adapt. Therefore, this paper shows that the robot can pass through the hexagonal wall and minimize the collision between the robot and the wall.

The robot will pass through hexagonal-shaped obstacles and must maintain its position with the given setpoint to complete the obstacles. The sensor used is Rplidar A2 [3]. The sensor will be used as input, and the output on the robot is in the form of robot action. For the robot to run correctly, a fuzzy-PID control system is used, where fuzzy itself has a vague meaning, meaning a value that can be true or false in the equation. Conversely, fuzzy logic allows a problem formulation to be solved easily with an accurate solution. The PID (Proportional Integral Derivative) control system is a controller to determine the accuracy of an instrumentation system with feedback characteristics on the system [7].

The PID controller is proposed here because of its ability to control many cases of non-linear systems. The fuzzy system is used to perform auto-tuning of KP, KI, and KD values in the PID system so that these values can change depending on the rules made in the Fuzzy system.

2 Research Methodology

2.1 Hardware & Software

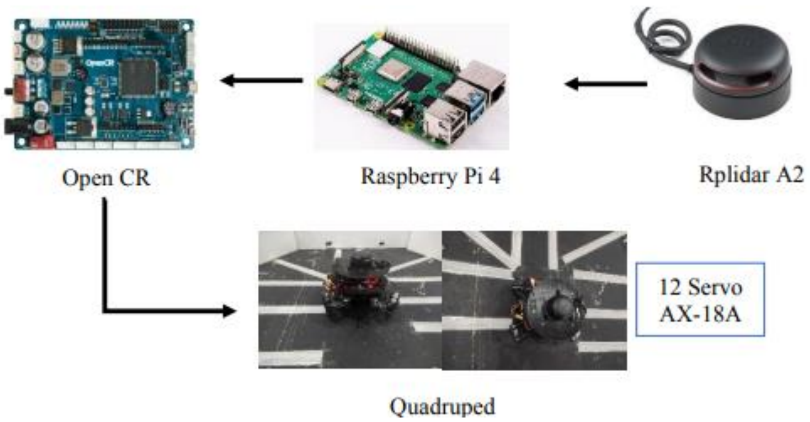

Fig. 1. Hardware

The hardware in this fuzzy-pid system uses a Raspberry PI 4 main control where the Rplidar will be communicated using USB to the Raspberry Pi. The Raspberry Pi sends communication using USB to OpenCR. OpenCR will forward to the AX-18A servo, which will command the legs to move

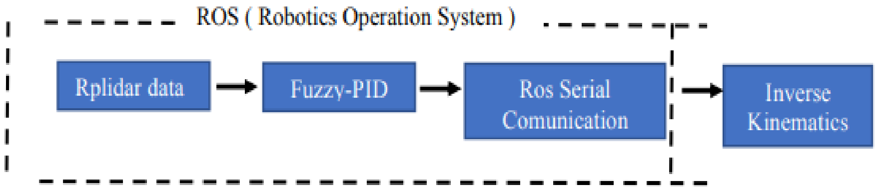

Fig. 2. Software

The system used on the robot is ROS (Robotics Operation System). This system helps communicate data and serial communication between Raspberry Pi and OpenCR in Raspberry Pi 4. There are main programs, programming methods used, programming against Rplidar A2, and ROS serial communication. In OpenCR, there is an inverse kinematic calculation that is used.

2.2 Inverse Kinematics

The six-legged robot leg structure uses an insect leg consisting of 3 joints and three bone parts, namely the coxa, femur, and tibia. Inverse Kinematics is used to find the angle of the coxa, femur, and tibia joints so that the tip of the robot leg can reach the desired end-point position. The steps to find the angle values of the coxa (θc), femur (θf), and tibia (θt) are as follows[8].

1. calculate the angle θc :

$$\tan \boldsymbol{\theta_c} = \frac{y}{x} \quad (1)$$

$$\boldsymbol{\theta_c} = tan^{-1}\left(\frac{y}{x}\right) \quad (2)$$

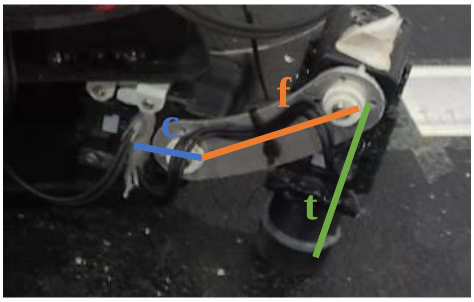

Fig. 3. Structure of Robot legs

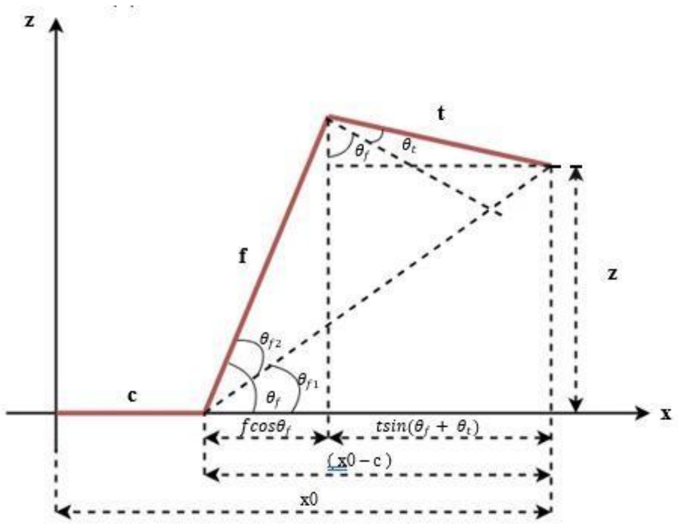

Fig. 4. Viewpoints of joints in the XZ plane [8]

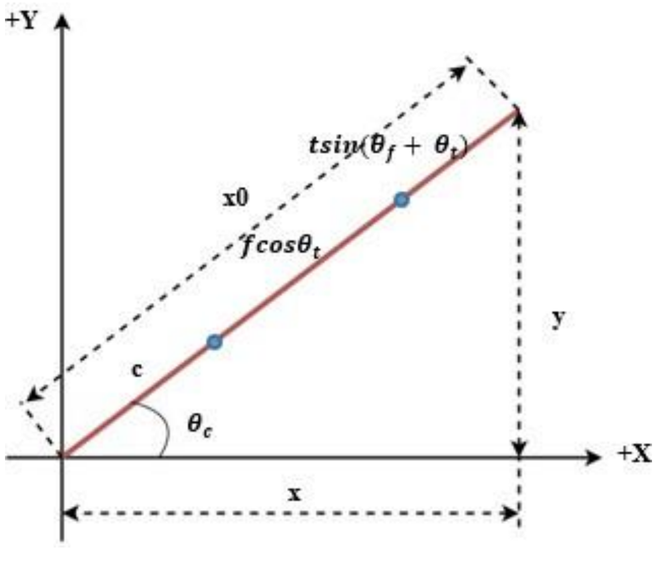

Fig. 5. Viewpoints of joints in the XY plane[8]

2. Calculating the magnitude of θ_f : To calculate the angle of θ_f, the values of x0, a, angles θ_{f1}, and θ_{f2} (in Figure 4) are required. To find the values of θ_f, θ_{f1}, θ_{f2} the calculation process is carried out in equations (3) to (7)[8].

$$x0 = \sqrt{x^2 + y^2} \quad (3)$$

$$f1 = tan^{-1} \left(\frac{z}{(x0-c)}\right) \quad (4)$$

$$a = \sqrt{z^2 + (x0 - c)^2} \quad (5)$$

$$\theta_{f2} = cos^{-1} \left(\frac{f^2+a^2-t^2}{2.a.f}\right) \quad (6)$$

$$\theta_f = \theta_{f1} + \theta_{f2} \quad (7)$$

3. Calculating the amount of θ_t: to calculate the angle θ_t, the analysis in Figure 4 is performed. This calculation process is shown in (8).

$$\theta_t = cos^{-1} \left(\frac{f^2+t^2-a^2}{2.f.t}\right) - 90^\circ \quad (8)$$

$$(syarat : a <= f + t)$$

2.3 Fuzzy-PID

The PID controller is a combination of three types of controllers, namely proportional controllers, integral controllers, and derivative controllers. Combining the three types of controllers to improve system performance, where each controller will complement and cover each other's weaknesses and strengths.

$$PID = k_P e(t) + k_i \int_0^t e(t)\, dt + k_d \frac{d_e(t)}{dt}$$

Fuzzy means vague, hazy or unclear. Fuzzy is a term for declaring a group/set that can be distinguished from other sets based on the degree of membership with unclear (vague) boundaries[5]. Unlike classical sets, which distinguish membership of a set into two, a set of members or non-members[6].

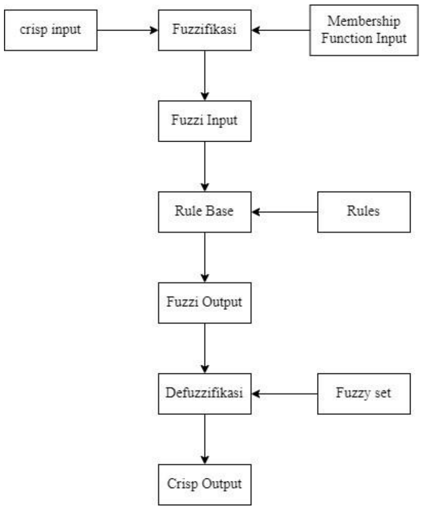

Fig. 6. Fuzzy Control Chart.

2.3.1 Fuzzy Working Principle

There are several stages in fuzzy processing, namely:

1. Fuzzification
   This fuzzification is used as input membership function parameters in the form of errors (e) and error changes (de). The degree of membership is between 0 and 1.
2. Evaluate Rule
   Rule evaluation is a decision-making process (inference) based on the rules set on a rule base to link between input and output fuzzy variables.
3. Defuzzification
   The input of the defuzzification process is a fuzzy set obtained from the composition of the fuzzy rules, while the resulting output is a number in the domain of the fuzzy set.

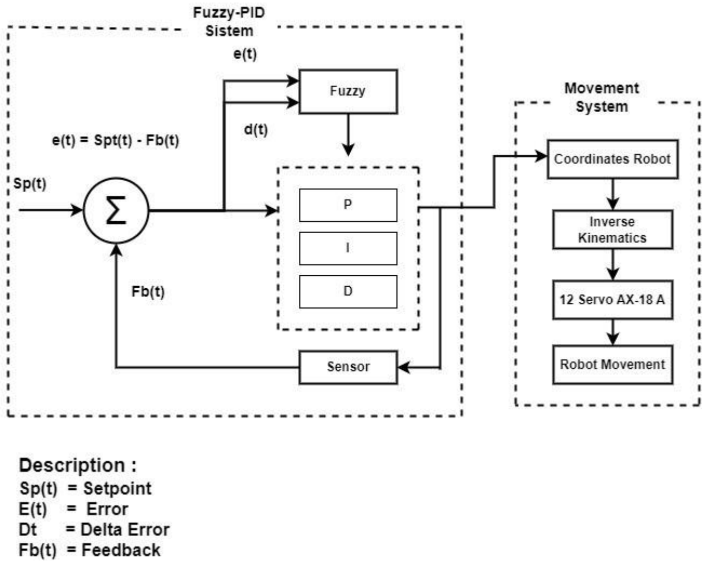

Fig. 7. Block Diagram.

In this research, robot position control will be made using the Fuzzy PID control system. In the Fuzzy PID process, the input given is an error value (e) and a change in error value (de) from a predetermined setpoint from the PID process and will be processed into the Fuzzy system. The output value results from the fuzzy will be mapped first so that the PID control will understand the value the Fuzzy gives. PID will process the value. The result of the PID output value will be a reference for the movement of the robot and the speed of the robot itself[7]

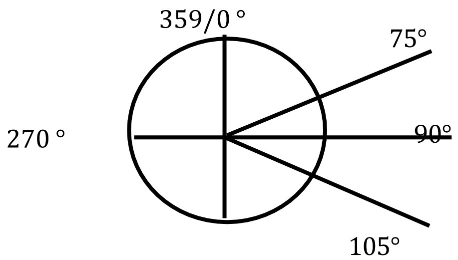

Fig 8. Lidar Degree.

In the picture above, using an angle of 80 degrees to 100 degrees on the Rplidar A2 sensor, the reading will be concluded or get the average value of 20 degrees, where later the output from the sensor is only 1 piece of data, the data will be calculated right into the block diagram shown.The number of variables created is 3 variables used as initialization of each Fuzzy membership. A = { Low, Medium, High }

Description ::

a) Low = Near
b) Medium = Middle
c) High = Far

Moreover, the value range of each membership is as follows:

1. Error (e)

Low = { 10 – 16 } , medium { 13 – 19 }, high { 16 – 22 }

1. Error Change (de)

Low = { 0 – 1,5 }, medium { 0,75 – 2,25 }, hign { 1,5 – 3 }

The following is the membership function for each of these inputs : Error :

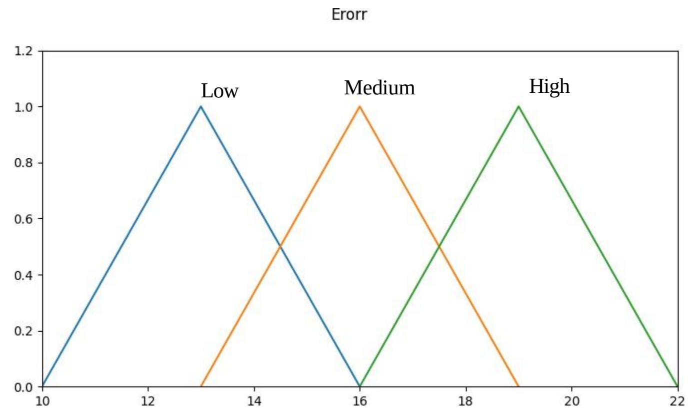

Fig. 9. Membership Error**.**

Delta Erorr (de) :

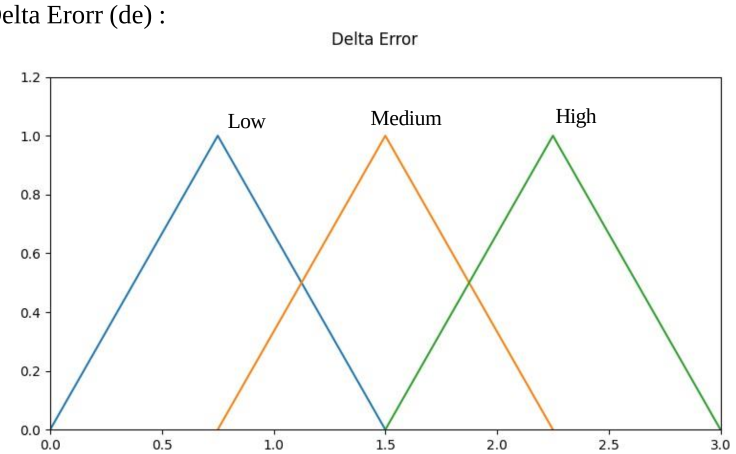

Fig. 10. Membership Delta Error**.**

Table 1. Rule.

	Error			
Delta Error	L	L	M	H
	L	L	M	H
	M	M	M	H
	H	L	H	L

In Figure 9, the membership error is determined by the field factor of the KRI competition in the division of firefighting robots and victim rescue. The field aisle at the competition is at most 45 cm. Therefore, three membership errors are made: low, medium, and high, with an error value determined by trial and error. In Figure 10 the m,membership delta error is obtained from experimental experiments carried out, and the value

3 Result And Discussion

Fig. 11. Testing Field**.**

The field image above is a test field where the field has a flat surface and is also hexagonal, and the robot must be able to pass through these obstacles using this Fuzzy PID. Before applying the fuzzy PID method to a quadruped robot, some things must be done, namely paying attention to the level of accuracy in reading the Rplidar A2 sensor at the constant robot position. The table below shows the Rplidar A2 sensor readings at 80-100 degrees:

Table 2. Validation of Lidar Distance.

No	Reading Distance (cm)	Distance read in the terminal	%Erorr
1	14	14	0%
2	16	16	0%
3	18	18	0%

4	20	20	0%
5	22	22	0%
6	24	24	0%
7	25	25	0%
8	26	26	0%
9	28	28	0%
10	30	30	0%

The results from the table above show that the accuracy obtained has an error rate of 0%, and then the robot is tested using the PID method to get the best constant value.

Table 3. Look up Constant PID.

Setpoint	Kp	Ki	Kd	Erorr
38	0,57	0,1	0,011	4,10%
	0,57	1	0,011	5,65%
	0,1	0,1	0,011	3,46%
	0,1	0,1	0,1	4,46%
	0.7	0.1	0.01	2,74%
	0,8	0,1	0,1	3,25%
	0,9	0,1	0,1	3,47%
	1	0,1	0,1	3,00%
	1	0,1	0,01	2,09%
	1,5	0,3	0,01	4,20%

The results from the table above show the best constant values at Kp = 1, Ki = 0.1 and Kd = 0.01 with an error percentage of 2.09%, and these error values will be entered into the fuzzy rule so that tuning the PID constant automatically.

Table 4. Fuzzy PID.

No	Setpoint	Average Error	The average value of the error obtained	Results
1	28	1,45%	0.116 m/s	Succeed

2	30	1,76%	0.130 m/s	Succeed
3	32	2,17%	0,130 m/s	Succeed
4	34	1,65%	0,130 m/s	Succeed
5	36	1,74%	0,130 m/s	Succeed
6	38	1,19%	0.130 m/s	Succeed
7	40	2,86%	0.116 m/s	Succeed
8	42	2,01%	0.130 m/s	Succeed
9	44	1,22%	0,130 m/s	Succeed
10	46	3,65%	0,154 m/s	Succeed

The test from Table 3 above is to place the robot close to the setpoint value. From the results of 10 trials, there were 10 successful attempts. The robot's speed is not always the same as the speed of the other setpoint, where the indicator of success is that the robot can pass through a hexagonal wall without hitting a wall.

$$P \frac{10}{10} = 100\ \%$$

$$P = 100\%$$

This can be seen from the 10 trials carried out, which had a success percentage of 100%.

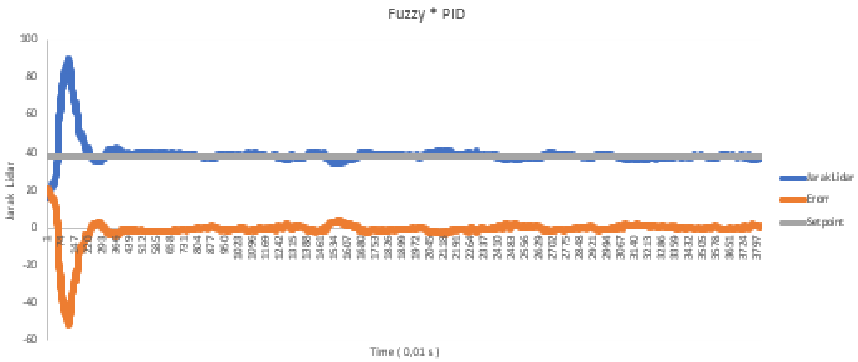

Fig. 12. Fuzzy PID Graph**.**

The picture above is a graph of Lidar Distance Against Walls. It can be seen that there are three lines of different colours, namely orange color represents the error value, the blue color represents the Lidar Distance value, the grey color represents the setpoint on the robot, lidar distance represents the robot's position against the wall, where the value of the lidar distance.

follows the value of the given setpoint, namely setpoint 38, from this data there is overshoot data where overshoot data occurs because the robot's position is far from the setpoint value.

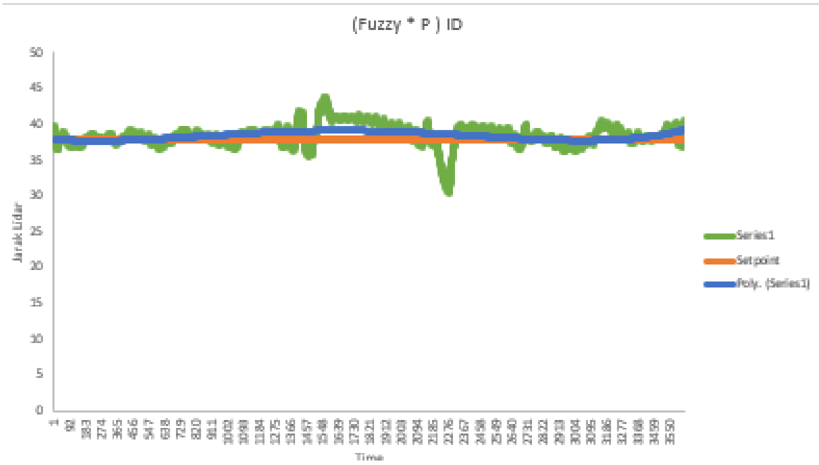

Fig. 13. Constant P Fuzzy Graph**.**

The image above is a graph of Lidar Distance Against Walls. It can be seen that there are three lines of different colours. Namely, the orange color represents an error value, and the green color represents the Lidar Distance value. In contrast, the blue color represents a polynomial filter to make it easier to analyze the graph above. Lidar distance represents the robot's position against the wall. Multiplying the fuzzy value only by the constant KP can reduce errors and increase the speed of system response. However, if the kp is enlarged, it will cause a significant overshoot, even the instability seen in Figure 13 above.

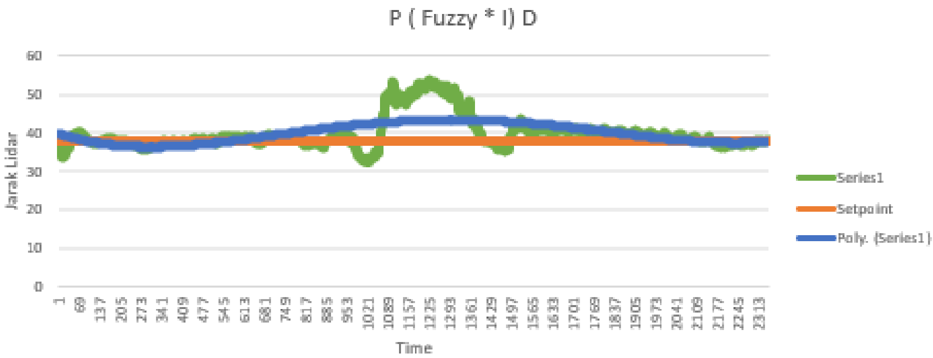

Fig 14. Constant I Fuzzy Graph**.**

The image above is a graph of Lidar Distance Against Walls. It can be seen that there are three lines of different colours. Namely, the orange color represents an error value, and the green color represents the Lidar Distance value. In contrast, the blue color represents a polynomial filter to make it easier to analyze the graph above. Lidar distance represents the position of the robot against the wall. Multiplying the fuzzy value only by the constant KP can reduce errors and increase the speed of system response. However, if the kp is enlarged, it will cause a significant overshoot, even the instability seen in Figure 14 above.

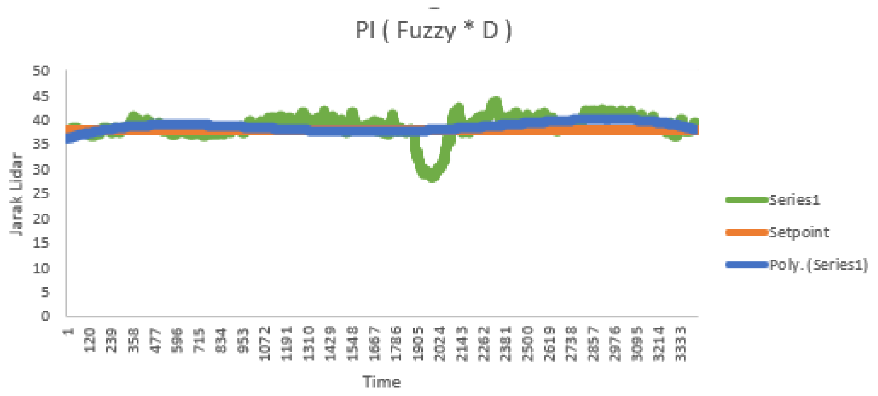

Fig. 15. Constant D Fuzzzy Graph.

The image above is a graph of Lidar Distance Against Walls. It can be seen that there are three lines of different colours. Namely, the orange color represents an error value, and the green color represents the Lidar Distance value. In contrast, the blue color represents a polynomial filter to make it easier to analyze the graph above. Lidar distance represents the position of the robot against the wall. The KD constant increases, causing a long adjustment time due to the less high KP and KI values.

4 Conclusion

In all experiments between fuzzy PID or fuzzy multiplied by KP, Fuzzy multiplied by KI, and Fuzzy multiplied by KD, it can be concluded that all of the above experiments can pass hexagonal obstacles without the robot colliding with the wall, experiments using Fuzzy-PID, namely trying the value of Fuzzy multiplied by all PID constants produced better results than Fuzzy multiplied by one of the PID constants.

Acknowledgements Bismillahirohmanirrohim, Praise and gratitude the author goes to Allah subhanahu wata'ala because thanks to His grace, the author can complete the Final Project Report with the title “Fuzzy PID Logic control on Quadruped Robot Using Rplidar A2 Sensor for Hexagonal Movement Application”. Not forgetting shalawat and greetings, the author conveys to the Prophet Muhammad sallallaahu 'alaihi wasallam because thanks to him, we came out of the age of ignorance and can feel the beauty of an era full of helpful knowledge. I thank my parents and my family, who have encouraged me to work on this book and final assignment; I also thank my supervisor Diono, S.Tr.T., M.Sc, who has guided me.

References

[1] M. Liu, F. Xu, K. Jia, Q. Yang, and C. Tang, "A Stable Walking Strategy of Quadruped Robot Based on Foot Trajectory Planning," *Proc. - 2016 3rd Int. Conf. Inf.Sci. Control Eng. ICISCE 2016*, pp. 799–803, 2016, doi: 10.1109/ICISCE.2016.175.

[2] R. C. Prayogo, A. Triwiyatno, and Sumardi, "Quadruped Robot with Stabilization Algorithm on Uneven Floor using 6 DOF IMU based Inverse Kinematic," *Proc. - 2018 5th Int. Conf. Inf. Technol. Comput. Electr. Eng. ICITACEE 2018*, pp. 39–44, 2018, doi: 10.1109/ICITACEE.2018.8576969.

[3] A. N. Catapang and M. Ramos, "Obstacle detection using a 2D LIDAR system for an Autonomous Vehicle," *Proc. - 6th IEEE Int. Conf. Control Syst. Comput. Eng. ICCSCE 2016*, no. November, pp. 441–445, 2017, doi: 10.1109/ICCSCE.2016.7893614.

[4] S. Sendari, M. S. Hadi, A. N. Handayani, Y. R. Wahyudi, and H. I. Lin, "Implementation of PD (Proportional Derivative) Control System on Six-Legged Wall Follower Robot," *2018 Int. Autom. Control Conf. CACS 2018*, no. cm, pp. 1–6, 2019, doi: 10.1109/CACS.2018.8606776.

[5] C.-F. J. Zhi-Yuan Yang, "Hexapod robot wall-following control using a fuzzy controller," 2014. [Online]. Available: http://www.cyberbotics.com/

[6] C. F. Juang, Y. H. Chen, and Y. H. Jhan, "Wall-following control of a hexapod robot using a data-driven fuzzy controller learned through differential evolution," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, 2015, doi: 10.1109/TIE.2014.2319213.

[7] D. Ghorpade, A. D. Thakare, and S. Doiphode, "Obstacle Detection and Avoidance Algorithm for Autonomous Mobile Robot using 2D LiDAR," *2017 Int. Conf. Comput. Commun. Control Autom. ICCUBEA 2017*, 2018, doi: 10.1109/ICCUBEA.2017.8463846.

[8] Indra Agustian Kurniawan, Feriyonika, Sabar Pramono, "Inverse dan Body Kinematics pada Robot Hexapod, " 2018