



POSITION CONTROL OF VTOL SYSTEM USING ANFIS VIA HARDWARE IN THE LOOP



Hari Maghfiroh*, Chico Hermanu, Vernanda Sitorini Zul Hizmi

Department of Electrical Engineering, Faculty of Engineering, Universitas Sebelas Maret, Indonesia

Abstract

Electric motors have been widely applied in various equipment. One application is found in Unmanned Aerial Vehicles (UAVs). An electric motor speed control system that can balance the aircraft's position is one of the mandatory features that must be owned by the aircraft. The position balancer control also supports the Vertical Take-Off Landing (VTOL) system. This study's VTOL position control system uses Hardware-in-the-loop (HIL) method with MATLAB Simulink and Arduino. ANFIS (Adaptive Neuro-Fuzzy Inferences System) is used as a position control algorithm. The controller performance is compared with conventional PID and FLC (Fuzzy Logic Controller). The system is tested as an initial position variation and loading test. The experiment shows that HIL can help fast prototyping by faster changes in the controller algorithms and is easy to program. The result is varied in each experiment. In the ISE (Integral Square of Error) point of view, ANFIS is better than PID by 100 % and has a very small difference from FLC in the initial position test. ANFIS is better by 95.44% and 4.56% compared with PID and FLC in the loading test, respectively.

This is an open access article under the [CC BY-NC](#) license



Keywords:

ANFIS;
HIL;
Position control;
VTOL;

Article History:

Received: October 28, 2020
Revised: December 15, 2020
Accepted: December 30, 2020
Published: July 30, 2021

Corresponding Author:

Hari Maghfiroh
Department of Electrical Engineering, Faculty of Engineering, Universitas Sebelas Maret, Indonesia
Email: hari.maghfiroh@staff.uns.ac.id

INTRODUCTION

Electric motors have been widely applied in production, medical, auto industry, instrumentation, automotive, to aerospace [1]. A simple example of electric motors in the aerospace field is Unmanned Aerial Vehicles (UAVs). An electric motor speed control system that can balance the aircraft's position is one of the mandatory features that must be owned by the aircraft.

The airplane balancer system on certain airplane models must be active since the aircraft will take off. Like the helicopter or UAV models popular today, the drones, which usually are quadrotor type, will balance the entire fuselage when taking off. This type seeks to achieve a stable and precise point by balancing the forces produced by the four rotors [2]. The use of drones is rife in the fields of agrarian [3], military [4], industry, and film [5].

One type of drone with the rapid development is Vertical Take-Off Landing (VTOL) [6]. The position balancer control also supports the VTOL system and has been developed by several methods. The method currently used in VTOL is the Proportional Integral Derivative (PID) method [7]. Several studies have emerged to try other control systems or combine several control systems with improving control performance. Jacob and Kumar proposed Sliding Mode Control (SMC) to balance the VTOL System. They conclude that SMC performs better than PID [8].

However, the SMC algorithm cannot adapt to system changes well. The other researchers propose a Fuzzy Logic Controller (FLC) [9] or Adaptive Network-based Fuzzy Inference System (ANFIS) [10][11]. The control system can be implemented and handle changes in system conditions even though the plant model is unknown [12]. Research and development

methods for VTOL will continue to develop along with aircraft types or models.

Aydoğdu and Çunkaş study the realization of fuzzy logic controlled brushless dc motor drives using MATLAB/Simulink. The control algorithm of fuzzy logic and PID are compared [13]. The dynamic characteristics of the BLDC motor) as well as the current and voltage of the inverter components were observed and analyzed using the developed model.

Rabah et al. have study about design of a fuzzy-PID controller for quadcopter trajectory-tracking. The proposed controller is compared with the PD and FLC to differentiate which one has better performance using MATLAB. The simulation results show that the Fuzzy-PID controller responds better on different paths than the other two controllers [14]. Burhanuddin and Malik study fuzzy control UAV based MATLAB on Arduino flight controller. Fuzzy controllers are designed to be tested in Hardware in Loop (HIL). The experimental results and the validation of the controller application function were considered satisfactory. It was concluded that it was possible to stabilize the quadcopter with a fuzzy logic controller [15]. Mahfouz et al. compare PID and ANFIS in controlling the roll, pitch, yaw, and altitude of Quad-Rotor Helicopters. From the simulation result, they conclude that ANFIS has better performance than PID. On the other hand, PID is the easier one [16].

Currently, the researchers are quite helped by software that can display the project design results without having to spend money to do the project in a tangible form. For example, one of the software used to research is MATLAB with Simulink. Inc. Simulink is a graphical programming language that can display data flow for modeling, simulation, and analysis of a dynamic system before being transferred to hardware. The condition is one of the advantages of Simulink for novice programmers because programmers do not need to write the syntax of C, C ++, or HDL code, but the syntax components are presented in block form [17]. In addition, Simulink is also able to deploy or integrate systems on hardware in real-time.

Based on the references that have been explained, this research will focus on the VTOL position control system by comparing the PID, FLC, and ANFIS methods using the Simulink interface, which is able to integrate with embedded systems without the need to reprogram embedded systems used. This method is also known as Hardware in the Loop (HIL). The use of HIL using MATLAB has been done by [18, 19, 20]. All conclude that HIL can be used for fast prototyping.

The contribution of this research is to compare the performance of PID, FLC, and ANFIS in the position control system of VTOL via HIL. The result of this experiment can be used to determine which controller that base suited in VTOL position control. It also gives information about the effectiveness of the HIL method in prototyping.

This paper is organized as follows. Section 2 presents material and method which explain the theoretical review of the VTOL System and the design of PID, FLC, and ANFIS for position control of VTOL. Then, in section 3, the simulation and testing results are discussed. Finally, the conclusion is in section 4.

MATERIAL AND METHOD

Material

The material in this study is the Vertical Take-Off Landing (VTOL) System. VTOL system refers to aircraft that can take off, float, and land vertically. Typical VTOL-based helicopter designs or multirotor designs combine four or more propellers that create lift and thrust for aircraft. Multirotor VTOLs can be equipped with various loads, including high-resolution cameras, multispectral sensors, and environmental monitoring sensors such as CO₂ and radiation detectors [21].

Figure 1 shows the design of the single-axis VTOL system used in this research, whereas Figure 2 shows its hardware implementation. As can be seen in Figure 2, this model is a quarter model of a VTOL or quadrotor that only uses one propeller. The position control is doing by balancing the left and right bar of Figure 2. This system consists of a Li-Po 3S battery, Arduino Uno, MPU sensor, ESC (Electronic Speed Controller), and BLDC motor as power supply, main controller, position sensor, motor driver, and motor propeller, respectively. The BLDC motor is used since it light and has the high speed and efficiency [22].

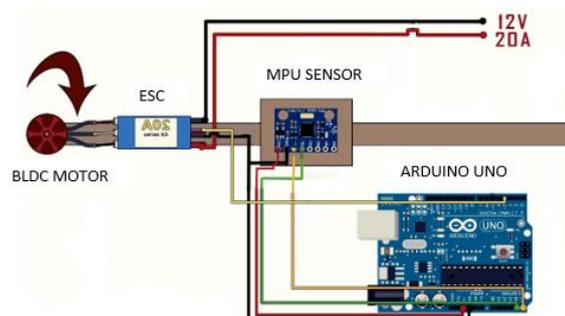


Figure 1. VTOL Schematic

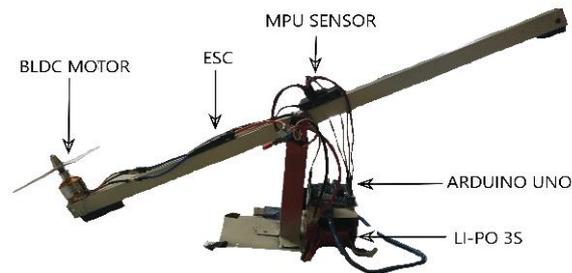


Figure 2. Hardware Implementation

Method

The control system used in this study is a type of automatic regulating system, which is a feedback control system that maintains the position of the system regarding any interference.

Proportional Integral Derivative (PID)

PID is a control system commonly used in industry because it is easy to use and simple [23][24]. The design of a PID begins by using the System Identification Toolbox to get the system model or known as Black Box modelling [25]. The reference data is in a time database, so Time-Domain Signal is selected in the data import option. Furthermore, the input data x is a workspace containing the PWM control value, and the output data y is a workspace containing the angular value of the sensor reading. There are 784 data taken for ± 18 seconds with a 0.0238 sample time. Next, experiment to find the transfer function estimation using second-order, with the number of poles and zero of 2 and 0, respectively. The transfer function uses the same discrete time as the sample time. Figure 3 shows the Simulink block diagram of PID control with the plant transfer function.

The PID block on Simulink provides an automatic tuning feature and will display the system conditions before and after tuning along with the tuning parameters of the PID. If the tuning is still unsuitable, the PID value can be changed on the Tuning Tools tab to get the desired results. The tuning process is complete and gets a K_p parameter value of -1.274, a K_i value of -0.195, and a K_d value of 0.288.

After updating the PID controller block, the system is simulated for 70 seconds and gets the graph output in Figure 4. The system can be controlled using the PID with a rise time of ± 3 seconds, overshoot 12.3% in the range of the 5 seconds, and reaches a steady state at 30 seconds.

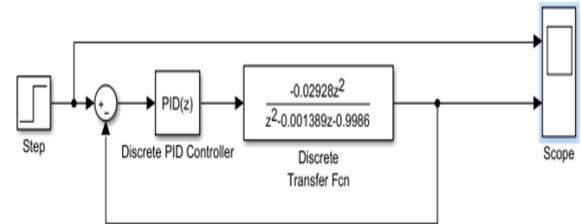


Figure 3. PID Controller System

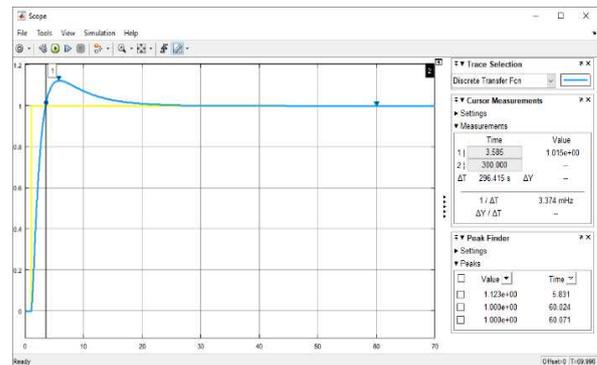


Figure 4. PID Tuning in MATLAB/Simulink

Fuzzy Logic Controller (FLC)

FLC design starts with using the Fuzzy Logic Designer Toolbox in MATLAB. FLC will use two inputs, namely error and derivative error, and one output: control value. Each block has a membership function to classify the values to be controlled. This input error and derivative error block use a range of values from -30 to 30. The value is chosen because the sensor's reading angle is $\pm 26^\circ$, and the desired angle is 0° .

Figure 5 is the Fuzzy Logic Designer Toolbox in the MATLAB environment, while the membership function of input and output is shown in Figure 6. NB, NS, Z, PS, PB are negative big, negative small, zero, positive small, and positive big, respectively.

After initiating the membership function in each block, the next step is to design the rule base FLC, Table 1, in the rule editor. Next, the rules are entered into the rule editor by inputting the error value, error derivative, and output value. The value inputted from the first line decreases, then the second line decreases, and soon. The order in which the rules are entered does not affect anything. Then the Fuzzy Inference System (FIS) will be exported to the MATLAB workspace for later recall and use in the FLC block in Simulink.

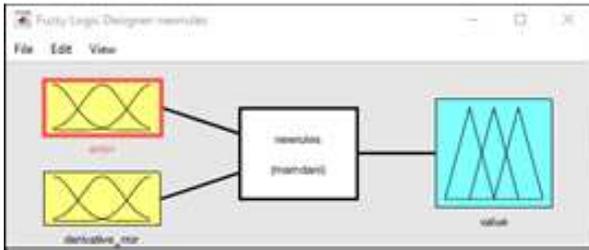


Figure 5. Fuzzy Logic Designer Toolbox

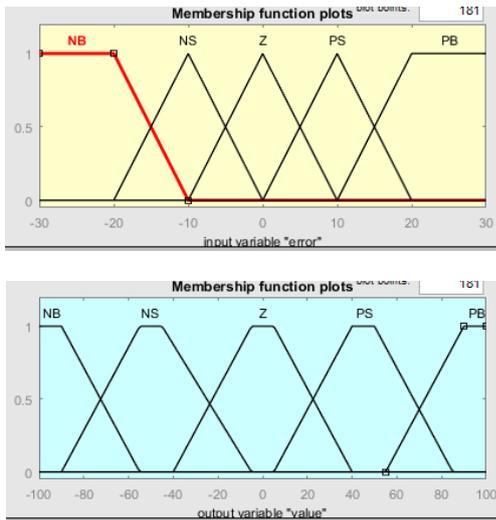


Figure 6. Membership Functions Input and Output of FLC

Table 1. Rule Base FLC

de/ e	NB	NS	Z	PS	PB
NB	PB	PB	PB	PS	Z
NS	PB	PB	PS	Z	NS
Z	PB	PS	Z	NS	NB
PS	PS	Z	NS	NB	NB
PB	Z	NS	NB	NB	NB

Adaptive Network-based Fuzzy Inference System (ANFIS)

The difference between FLC and ANFIS is in how to generate fuzzy membership and rules. In FLC, it is manually defined, whereas ANFIS design it using Neural Network (NN). In the MATLAB environment, the design of ANFIS can be done using Neuro-Fuzzy Designer Toolbox. The data to be trained is loaded into the workspace, with input-1 being error data, input-2 is a derivative error, and output is the control value. The toolbox will generate FIS and training data to get the appropriate rules from the input and output of the reference data. Figure 7 shows the results of the training in the neuro-fuzzy designer toolbox.



Figure 7. Training Result in the ANFIS Toolbox

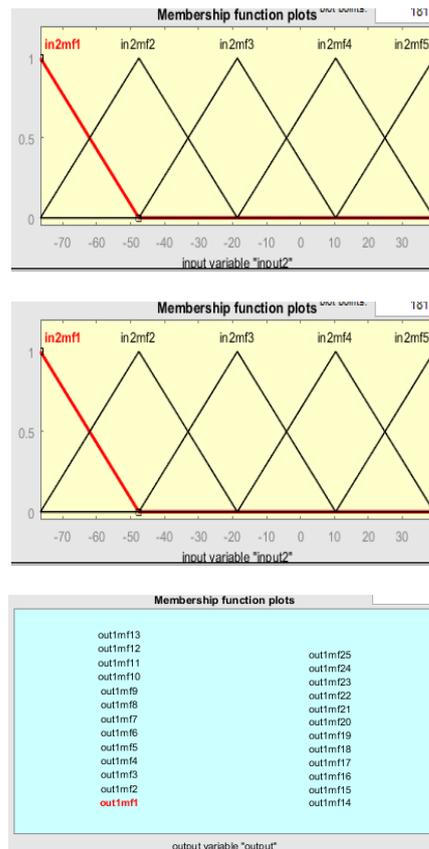


Figure 8. Membership Functions for Input 1 and Input 2, and Output of ANFIS

The results of testing errors on ANFIS were 7.5844. ANFIS is created with each of the five triangular membership functions at constant input and output. ANFIS also creates rules with names and automatic parameters from reference data. The membership functions are shown in Figure 8, whereas the rules in the rule editor are presented in Table 2. Then the FIS will be exported to the MATLAB workspace for later recall and use in the FLC block in Simulink.

Table 2. Rule Base ANFIS

de/ e	in1mf1	in1mf2	in1mf3	in1mf4	in1mf5
in2mf1	out1mf1	out1mf6	out1mf11	out1mf16	out1mf21
in2mf2	out1mf2	out1mf7	out1mf12	out1mf17	out1mf22
in2mf3	out1mf3	out1mf8	out1mf13	out1mf18	out1mf23
in2mf4	out1mf4	out1mf9	out1mf14	out1mf19	out1mf24
in2mf5	out1mf5	out1mf10	out1mf15	out1mf20	out1mf25

RESULTS AND DISCUSSION

The Hardware in the Loop (HIL) concept is used in this experiment because it gives an easy way to control implementation and monitoring. MATLAB Simulink is used as an interface and a virtual processor in which controller algorithms are implemented. The output of this controller is sent to the Arduino using serial communication, and sensor reading from the system is feedback to Simulink via Arduino.

The Simulink system starts when the COM23 receives serial sensor reading data from Arduino, then the data is converted with a data type converter block into a double data type. The difference between the reference value and the sensor reading is an error that will enter the control system. In Figure 9, there are two control

blocks in the blue area, namely the PID Controller and Fuzzy Logic Controller. For PID Controller programming, it only has one input which is the error value. Whereas the Fuzzy Logic Controller has two inputs, namely error value and derivative error. A manual switch is used to change the control method between the PID and FLC methods. In the Fuzzy Logic Controller block, the previously created FIS file will be called from the workspace. In the Fuzzy Logic Controller block, the FLC and ANFIS methods can be changed according to the FIS name that has been created. The control value is limited by the saturation block ± 150 . It is done to limit the outlier value and adjust the programming on Arduino. The control value will be converted into single data and bytes sent back to Arduino via COM23 serial. Display and scope can be added anywhere to monitor data every step. Then Figure 10 is a position control VTOL with interface Simulink in real-time mode simulation. There are two tests done in this research which is different initial position and loading test.

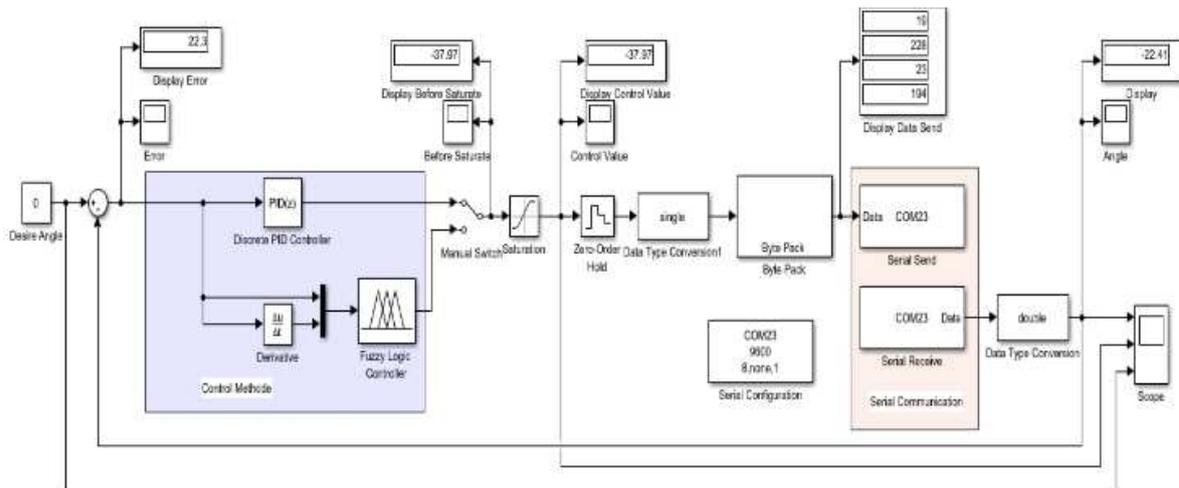


Figure 9. Interface on Simulink

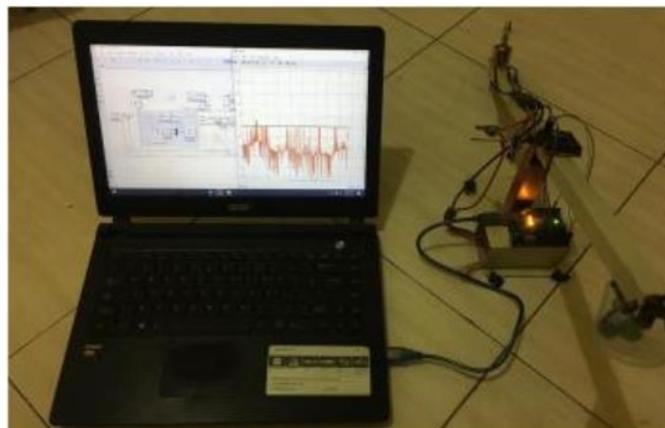


Figure 10. Position Control VTOL with Simulink Interface

Different Initial Position

This scheme tests the performance of a VTOL system when one of its propeller systems is at the upper or lower of its horizontal line or x-axis line. There are three different initial positions tested which are -15°, 0°, and 15°, which depict lower position, horizon, and upper positions, respectively.

1) *Initial position -15°*

Figure 11 is the result of an initial position of -15°. PID, FLC, and ANFIS are in the yellow, blue, and red graphs, respectively. After the system is turned on, it oscillates around its center point, trying to stabilize the position at its horizon. As a result, FLC and ANFIS are stable after the 5s event not in the horizon line (0°). On the other hand, the PID response swings up and becomes stable in the upper of the horizon line after the 20s.

The comparison of the controller's performance with the initial position of -15° is presented in Table 3. From the quantitative data from Table 3, PID is superior in terms of undershoot and ISE (Integral Square of Error) with the smallest value. FLC is superior in terms of rising time. In comparison, ANFIS is best for settling time and overshoot.

2) *Initial position 0°*

Figure 12 is an initial position test result at 0°. After the system works from its horizon position, it varies up and down from the initial position, trying to be stable. ANFIS gives the response nearest to the equilibrium or horizon line. FLC oscillate under the horizon line and stable after the 20s under the horizon. Whereas the PID response is not stable until the experiment time is over.

The comparison of the controller's performance with the initial position of 0° is presented in Table 4. At this test, PID has the worst performance compared with the two others. At the same time, FLC is still superior in rising time. The last, ANFIS, has the best performance in terms of settling time and ISE. Whereas both overshoot and undershoot of it are the same as the FLC response.

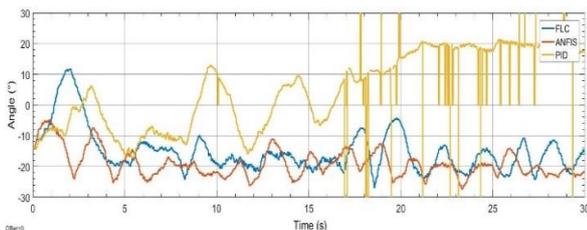


Figure 11. Initial Position Test Result at -15°

Table 3. Comparison of Controller Performance with Initial Position -15°

Parameters	PID	FLC	ANFIS
Rise Time (s)	3.21	0.96	1.81
Settling Time (s)	17.31	5.84	5.56
Overshoot (°)	21	11	0
Undershoot (°)	15	25	25
ISE	4.965*10 ³	8.367*10 ³	1.210*10 ⁴

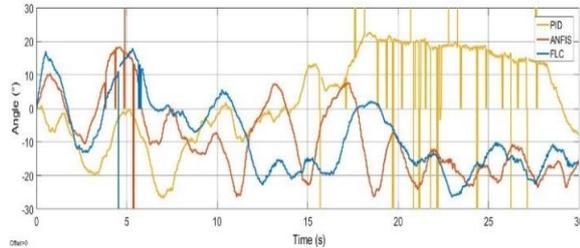


Figure 12. Initial Position Test Result at 0°

Table 4. Comparison of Controller Performance with Initial Position 0°

Parameters	PID	FLC	ANFIS
Rise Time (s)	1.56	0.81	1.16
Settling Time (s)	16.01	14.11	10.35
Overshoot (°)	21	18	18
Undershoot (°)	25	25	25
ISE	2.546*10 ¹²	5.298*10 ³	4.137*10 ³

3) *Initial position 15°*

Figure 13 shows the results when the initial position is 15°. The responses are different from the two previous results. It is seen that, after starting from the initial position, all control method gives responses which go below the horizon line. All of them reach stable around -20° after 5s.

Table 5 resumes the performance parameters value of this test. It is clearly seen from the data that FLC is superior in terms of rising time and ISE. At the same time, PID has the lowest settling time and undershoots compared with others. In contrast, ANFIS has the smallest overshoot.

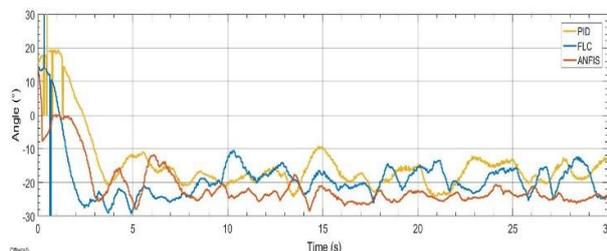


Figure 13. Initial Position Test Result at 15°

Table 5. Comparison of Controller Performance with Initial Position 15°

Parameters	PID	FLC	ANFIS
Rise Time (s)	0.96	0.51	0.56
Settling Time (s)	6.96	11.31	7.26
Overshoot (°)	19	15	0
Undershoot (°)	24	30	28
ISE	1.008*10 ⁴⁷	1.100*10 ⁴	1.944*10 ⁴

Loading Test

This kind of test is to know the response of the system when a load is added. First, the test is done by setting the propeller at equilibrium or horizon, then several tests by varying the load of five marbles (27.5 grams), six marbles (33 grams), and seven marbles (38.5 grams) are done. In this test, the load is added before the system start, and the initial position of the system is at the horizon line (0°).

1) Load 27.5 grams

Figure 14 shows 27.5 grams of load test results. All the response graphs are in a negative value, or the propeller is below the horizon line. This condition means that the propeller can raise the load. However, it cannot stabilize the load yet. The graph shows that after 5s, all the controller is stable below the horizon. PID is the fastest to reach stable conditions with the lowest rise time and settling time. It is also superior in undershoot and ISE with minimum value. At the same time, FLC and ANFIS have the same performance in overshoot and undershoot, while ANFIS has a lower ISE than FLC. The performance parameters value of this test is resumed in Table 6.

2) Load 33 grams

Figure 15 shows 33 grams load test results. It shows that the response in this test is varied between each controller. PID is oscillated around the horizon line and could not reach a stable condition until the test was ended. FLC is faster to reach a stable condition. However, its condition is below the horizon line. On the other side, ANFIS also oscillates around the horizon line.

Table 7 resumes the performance value of this test. PID has the fastest rise time with the highest ISE. FLC is superior in terms of settling time and overshoot. The last method, ANFIS, is the best in undershoot and ISE but has the worst settling time.

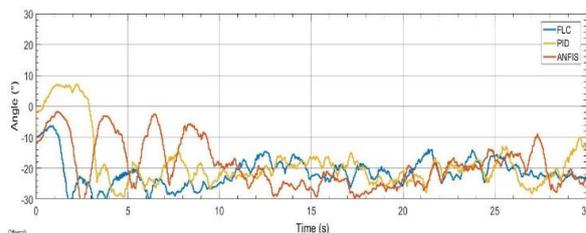


Figure 14. 27.5 grams Load Test Result

Table 6. Comparison of Controller Performance with 27.5 grams Load

Parameters	PID	FLC	ANFIS
Rise Time (s)	0.36	0.56	1.16
Settling Time (s)	3.26	10.91	10.86
Overshoot (°)	7	0	0
Undershoot (°)	30	31	31
ISE	1.226×10^4	1.452×10^4	1.229×10^4

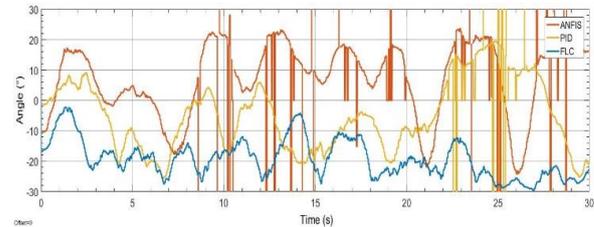


Figure 15. 33 grams Load Test Result

Table 7. Comparison of Controller Performance with 33 grams Load

Parameters	PID	FLC	ANFIS
Rise Time (s)	0.41	1.26	0.71
Settling Time (s)	7.46	3.01	8.56
Overshoot (°)	20	0	22
Undershoot (°)	25	25	15
ISE	7.090×10^{16}	1.230×10^4	4.539×10^3

3) Load 38.5 grams

Figure 16 shows the result of the 38.5 grams load test. The graph shows that all of the control methods give the same graph pattern over the horizon at this load. This means that the load is heavy, and the propeller is going up on the horizon. It also can be seen that all the responses are stable after 2s. FLC has higher oscillating in the middle of the test, whereas two others remain stable until the test ends. Table 8 shows the quantitative value of the performance index. It is seen that PID is superior in terms of overshoot. In comparison, ANFIS is better in settling time and ISE, while FLC has the highest settling time.

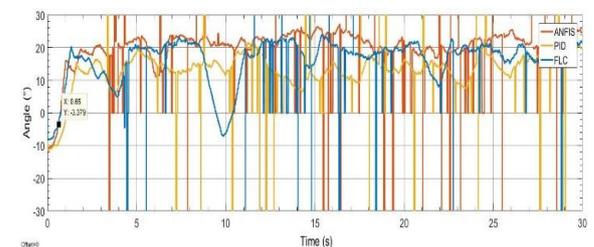


Figure 16. 38.5 grams Load Test Result

Table 7. Comparison of Controller Performance with 38.5 grams Load

Parameters	PID	FLC	ANFIS
Rise Time (s)	1.11	0.71	0.71
Settling Time (s)	2.95	4.71	2.7
Overshoot (°)	20	21	23
Undershoot (°)	0	5	0
ISE	1.351×10^{20}	1.404×10^{19}	6.168×10^{18}

Comparison of Controller Performance

The result of all the tests is compared by calculating the average value for each characteristic. Table 9 is the average value of the controller performance comparison table with the initial position -15°, 0°, 15° (Table 3, Table 4, and Table 5). Table 11 is the average value of the controller performance comparison table with a load of 27.5 grams, 33 grams, and 38.5 grams (Table 6, Table 7, and Table 8).

In this analysis, PID is used as the base for comparison. Therefore, the PID parameters value is considered as 100%. FLC and ANFIS performances are compared to PID to determine the best one in some parameters value. Table 10 and Table 12 are percentage comparisons when PID is the base of comparison for Table 9 and Table 11, respectively.

Table 10 inform that in the initial position test, PID has the smallest value of undershooting. FLC is superior in terms of rising time and ISE. At the same time, ANFIS is dominant in settling time, and overshoot. Table 12 resumes the result of the loading test. It

informs that at the loading test, PID has the shortest rise time and settling time. In contrast, FLC is better in overshoot. The last method, ANFIS, is superior in terms of undershoot and ISE.

The result is varied in each experiment. However, the overall result shows that ANFIS has better ISE compared to PID and FLC. Furthermore, it proves that ANFIS is better in position control of the VTOL system than PID and FLC. Since this experiment uses the HIL method, the tuning of PID can be done using MATLAB autotuning, whereas the ANFIS training data is received from PID. On the other side, FLC is manually tuned to choose the best membership and fuzzy rules. The HIL method offers the advantage which is easy to program when the controller algorithm is changed. This method can be used for rapid prototyping because the response of the controller algorithm can be evaluated faster.

Table 8. Comparison of Initial Position Testing Controller Performance

Parameters	PID	FLC	ANFIS
Rise Time (s)	1.91	0.76	1.17
Settling Time (s)	13.42	10.42	7.72
Overshoot (°)	20.33	14.67	6
Undershoot (°)	21.3	26.67	26
ISE	3.360*10 ⁴⁶	8.221*10 ³	1.189*10 ⁴

Table 9. Percentage of Comparison of Initial Position Testing Controller Performance

Parameters	PID (%)	FLC (%)	ANFIS (%)
Rise Time (s)	100	39.79	61.60
Settling Time (s)	100	77.61	57.52
Overshoot (°)	100	72.13	29.51
Undershoot (°)	100	125	121.88
ISE	100	2.45*10⁻⁴¹	3.54*10 ⁻⁴¹

Table 10. Comparison of Load Testing Controller Performance

Parameters	PID	FLC	ANFIS
Rise Time (s)	0.62	0.84	0.86
Settling Time (s)	4.56	6.21	7.37
Overshoot (°)	15.67	7	15
Undershoot (°)	18.33	20.33	15.33
ISE	4.51*10 ¹⁹	4.68*10 ¹⁸	2.06*10 ¹⁸

Table 11. Percentage of Comparison of Load Testing Controller Performance

Parameters	PID (%)	FLC (%)	ANFIS (%)
Rise Time (s)	100	134.57	137.23
Settling Time (s)	100	136.28	161.81
Overshoot (°)	100	44.68	95.74
Undershoot (°)	100	110.91	83.64
ISE	100	10.39	4.56

CONCLUSION

Position control of the VTOL system using ANFIS with the HIL method has been carried out. The controller performance is compared with conventional PID and FLC. The system is tested as an initial position variation and loading test. MATLAB Simulink is used with Arduino to make a HIL. The experiment shows that HIL can help fast prototyping with faster changes in the controller algorithms and is easy to program. The result is varied in each experiment. From the ISE point of view, ANFIS is better than PID by 100% in the initial position test and has a very small difference with FLC. ANFIS is better by 95.44% and 4.56% than PID and FLC, respectively, in the loading test.

ACKNOWLEDGMENT

We thank the Department of Electrical Engineering, Universitas Sebelas Maret, which supports this research.

REFERENCES

- [1] K. Ishihara and J. Morimoto, "An optimal control strategy for hybrid actuator systems: Application to an artificial muscle with electric motor assist," *Neural Network*, vol. 99, pp. 92-100, 2018, doi: 10.1016/j.neunet.2017.12.010
- [2] P. Castillo, R. Lozano and A. E. Dzul, *Modelling and Control of Mini-Flying Machines*, Compiegne, France: Springer, 2005
- [3] C. Rokhmana, "The Potential of UAV-based Remote Sensing for Supporting Precision Agriculture in Indonesia," *Procedia Environmental Sciences*, vol. 24, pp. 245-253, 2015, doi: 10.1016/j.proenv.2015.03.032
- [4] J. Maddalon et al., *Perspectives on Unmanned Aircraft Classification for Civil Airworthiness Standards NASA STI Programs*, Virginia: NASA Langley Research Center, 2013
- [5] R. K. Lea, R. Allen and S. L. Merry, "A comparative study of control techniques for an underwater flight vehicle," *International Journal of Systems Science*, vol. 30, no. 9, pp. 947- 964, 1999, doi: 10.10180/002077299291831
- [6] NN, "Unmanned Aircraft Systems Roadmap 2005-2030," *Department of Defense. Office of the Secretary of Defense*, United States, 2005
- [7] A. Adriansyah, S. H. M. Amin, A. Minarso, and E. Ihsanto, "Improvement of Quadrotor Performance with Flight Control System using Particle Swarm Proportional-Integral-Derivative (PS-PID)," *Jurnal Teknologi*, vol. 79, no. 6, pp. 121-128, 2017, doi: 10.11113/jt.v79.10680
- [8] J. Jacob and S. S. Kumar, "Sliding Mode Control of VTOL System", in *IOP Conference Series: Material Science and Engineering*, vol. 561, ID: 012069, 2019, doi: 10.1088/1757-899X/561/1/012069
- [9] Z. Omar and C. Bil, "Genetic Fuzzy Control for Autonomous Ducted-Fan VTOL UAV," in *26th International Congress of The Aeronautical Sciences (ICAS 2008)*, Alaska, USA, 2008
- [10] S. Rezazadeh, M. A. Ardestani and P. S. Sadeghi, "Optimal Attitude Control of a Quadrotor UAV Using Adaptive Neuro-Fuzzy Inference System (ANFIS)," *3rd International Conference on Control, Instrumentation, and Automation (ICCIA 2013)*, Tehran, Iran, 2013
- [11] H. Housny, E.A. Chater, and H. El Fadhil, "Multi Closed-loop Adaptive Neuro-Fuzzy Inference System for Quadrotor Position Control", *Advances in Science, Technology and Engineering Systems Journal*, vol. 5, no. 5, pp. 526-535, 2020, doi: 10.25046/aj050565
- [12] H. Maghfiroh, "Optimasi Sistem Kendalil PID dengan Double Tuning Dalam Implementasi Pengendalian Kecepatan Motor DC Berbasis PLC," *Undergraduate Thesis*, Universitas Gajah Mada, Yogyakarta, 2013
- [13] O. Aydoğdu and M. Çunkaş, "Realization of Fuzzy Logic Controlled Brushless DC Motor Drives Using Matlab/Simulink," *Mathematical and Computational Applications*, vol. 15, no. 2, pp. pp. 218-229, 2010, doi: 10.3390/mc15020218
- [14] M. Rabah, A. Rohan, Y.-J. Han and S. H. Kim, "Design of Fuzzy-PID Controller for Quadcopter Trajectory-Tracking," *International Journal of Fuzzy Logic and Intelligent Systems*, vol. 18, no. 3, pp. pp. 204-213, 2018, doi: 10.5391/IJFIS.2018.18.3.204
- [15] K. L. Pham, J. Leuchter, R. Bystřický, V. T. Pham and N. N. Pham, "Design and Simulation System for Quadrotor UAVs," *2021 International Conference on Military Technologies (ICMT)*, 2021, pp. 1-5, doi: 10.1109/ICMT52455.2021.9502775
- [16] M. Mahfouz, M. Ashry, and G. Elnashar, "Design and Control of Quadrotor Helicopters Based on Adaptive Neuro-Fuzzy Inference System", *International Journal of Engineering Research & Technology (JERT)*, vol. 2, no. 12, 2013

- [17] "MathWorks - Simulink," The MathWorks Inc., 2020. [Online]. Available: <https://in.mathworks.com/products/simulink.html>. [Accessed 27 Juni 2020].
- [18] I. Tejado, J. Serrano, E. Pérez, D. Torres and B. M. Vinagre, "Low-cost Hardware-in-the-loop Testbed of a Mobile Robot to Support Learning in Automatic Control and Robotics," *IFAC-Elsevier Ltd.*, no. 49-6, pp. 242-247, 2016
- [19] S. P. Biswas, M. K. Hosain and M. W. Rahman, "Real-time arduino based simulator enabled hardware-in-the-loop electric DC machine drive system," *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, 2017, pp. 823-826, doi: 10.1109/R10-HTC.2017.8289082
- [20] M. I. Lizarraga, V. Dobrokhodov, G. H. Elkaim, R. Curry and I. Kaminer, "Simulink Based Hardware-in-the-Loop Simulator for Rapid Prototyping of UAV Control Algorithms," *AIAA Infotech@Aerospace Conference*, Seattle, Washington, 2009
- [21] L. Plummer, "WIRED - What is VTOL? A beginner's guide to vertical take-off and landing technology," *Condé Nast Britain*, 28 April 2017. [Online]. Available: <https://www.wired.co.uk/article/vtol-vertical-take-off-landing-explained>. [Accessed 29 Juni 2020]
- [22] R. I. Zhilgotov and V. Y. Frolov, "Development of the sensorless control system BLDC motor," *2017 IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering (EIConRus)*, 2017, pp. 1109-1111, doi: 10.1109/EIConRus.2017.7910749
- [23] H. Maghfiroh, A. Ataka, O. Wahyunggoro and A. I. Cahyadi, "Optimal energy control of DC Motor speed control: Comparative study," *2013 International Conference on Computer, Control, Informatics and Its Applications (IC3INA)*, 2013, pp. 89-93, doi: 10.1109/IC3INA.2013.6819154
- [24] M. A. Shamseldin and A. A. EL-Samahy, "Speed control of BLDC motor by using PID control and self-tuning fuzzy PID controller," *15th International Workshop on Research and Education in Mechatronics (REM)*, 2014, pp. 1-9, doi: 10.1109/REM.2014.6920443
- [25] H. Maghfiroh, I. Iftadi, A. Sujono, "Speed Control of Induction Motor using LQG," *Journal of Robotic and Control (JRC)*, vol. 2, no. 6, pp. 565-570, Nov 2021, doi: 10.18196/jrc.26138