Temperature and water level control system in water thermal mixing process using adaptive fuzzy PID controller

D Tunjung¹, P Prajitno¹ and D Handoko¹

¹Department of Physics, Faculty of Mathematics and Natural Science (FMIPA), Universitas Indonesia, Depok, Indonesia

E-mail: prawito@sci.ui.ac.id

Abstract. Thermal Mixing Process, which is important in various industries, is a multiple-input multiple-output process (MIMO). It works by regulating hot-water and cold-water flows to control the temperature and level of the mixture. The Adaptive Fuzzy PID Control (AFPIDC) algorithm is a combination of two types of controller, has a simple PID basis with added Fuzzy aspects to speed up control. The AFPIDC algorithm is applied to the simulation of water thermal mixing process and is done with MATLAB/SIMULINK program. The fuzzy algorithm uses two inputs (errors & changes of errors) and has three outputs (changes of PID constants). The system is tested by simulating setpoint changes and adding disturbance. The testing result shows that AFPIDC controllers performs better than PID in controlling temperature and level. In temperature control, the PID settling time is 830 seconds, AFPIDC is 328 seconds and the PID overshoot is 6.3% and AFPIDC is 0%. In level control, the settling time of PID is 3221 seconds, while AFPIDC is 235 seconds, with PID overshoot is 10.5%, while AFPIDC 0%. From testing the system with leakage disturbance, the temperature controller needs time to regain stability at PID 780 seconds, AFPIDC 250 seconds. Meanwhile the level controlling stabilizes at PID 4510 seconds, and AFPIDC at 225 seconds.

1. Introduction

One of the most common and important process in many industries is the thermal mixing process. Thermal Mixing Process works by mixing the same ingredients with a known temperature to reach the desired mixture temperature. 90% of loops in industries still use PID because it is simple and easy to implement even though the performance is not as good as intelligent control. Fuzzy controllers can solve complex problems in the process to be controlled by solving the input and output logic which are described by linguistic variables. However, determining the fuzzy membership function and fuzzy sets is quite difficult compared to determining the gain in the PID system [1]. Therefore, the Adaptive Fuzzy PID Controller (AFPIDC) was created, which is a combination of PID and fuzzy controllers to get the benefits of each controller.

There are some studies have been conducted on this AFPIDC topic. Such as real-time level control in water storage tanks which results in faster performance than conventional PID controllers. Besides, concentration control in chemical reactors by manipulating flow rate values by using a self-tuning Fuzzy PID Controller results in faster control than PID controllers [2]. The AFPIDC study is also implemented in many other systems such as control of chlorine flow [3] and levels in boilers [4]. The goals of this research are to design a AFPIDC controller for water thermal mixing process and compare the performance with conventional (PID) controller.
2. Method

2.1. Water Thermal Mixing Process

The main objective of the process carried out in this simulation is to control the temperature ($T$) and level ($h$) in the mixed tank. This is done by adjusting the flow of water that enters from the hot water tank and the cold-water tank. This process is a type of multiple inputs - multiple-output (MIMO) and is generally known as the thermal mixing process. The flow values of hot water ($wh$) and cold water ($wc$) are used to control the temperature and water level in the mixed tank, while the values of hot water ($Th$) and cold water ($Tc$) can be considered as a system disturbance [5]. The system works continuously so that there is a flow of water coming out of the mixture tank ($w$). Therefore, $wh$ and $wc$ must be maintained to reach the desired temperature and level.

2.2. Mathematical Model of the Process

The water level at the mixing tank ($H$) will be affected by the volume of water in the mixing tank which can be determined by using the mass balance. The change in mass in the mixed tank will be the same as the sum of all the values for the flow of water in and out of the tank. In the system, it is assumed that there is no heat loss due to absorption by the container or due to radiation outside the system. So that the change in mixed water temperature ($T$) can be determined using energy equilibrium. To simulate the level response to the controller, it is necessary to make a transfer function using the differential equations that have been obtained. The transfer function is determined by carrying out the Laplace transformation of the differential function so that the $H$ function becomes the Laplace form $H'$ as well as the variable and in the Laplace form $w'_h$, $w'_c$, $w'$.

\[
\frac{H'(s)}{w'_h(s)} = \frac{0.00796}{s} \quad (1) \quad \frac{T'(s)}{w'_h(s)} = \frac{203.125}{250s + 1} \quad (2)
\]
\[
\frac{H'(s)}{w'_c(s)} = \frac{0.00796}{s} \quad (3) \quad \frac{T'(s)}{w'_c(s)} = \frac{-203.125}{250s + 1} \quad (4)
\]

The control valves used in the equation have the same characteristic with the previous experiment [6], which have time constant value of 0.018 seconds and have the transfer function stated by equation (5) and (6).

\[
\frac{M_s(s)}{G_c(s)} = \frac{0.0176}{0.018s + 1} \quad (5) \quad \frac{M_s(s)}{G_c(s)} = \frac{1.76}{0.018s + 1} \quad (6)
\]

Decoupler control is used in the MIMO system to reduce the interaction between the two loops in the system, so that when a set point changes in one loop, the control variable loop has little or no effect on the other loop. In system thermal mixing, the value and is the input received by the system, the
The flow of hot water, and a flow of cold water. While the value and is the output generated by the system, the water mixture temperature \( T(s) \), and water mixture level \( H(s) \)

\[
Y_1 = (T_{12} G_{p12} + G_{p11}) U_{11} + (T_{12} G_{p11} + G_{p12}) U_{22}
\]

\[
Y_2 = (T_{21} G_{p22} + G_{p21}) U_{11} + (T_{21} G_{p21} + G_{p22}) U_{22}
\]

With giving zero value on the influence of the other loop \((U_{12} & U_{21})\) and input the \( G_{p12} \) dan \( G_{p21} \) value to both equations, we got the following transfer functions for the decoupler

\[
T_{12} = -\frac{G_{p12}}{G_{p21}} = -1
\]

\[
T_{21} = -\frac{G_{p21}}{G_{p22}} = 1
\]

2.3. Controller Design
Controlling Adaptive Fuzzy PID Controller (AFPIDC) works by changing the value of the PID constants used in the controller using fuzzy logic. The search for PID constants in this simulation uses the PID tuner function in MATLAB Simulink so that the PID parameter values have a pretty good performance.

| Table 1. PID Controller Parameters. |
|-----------------|----------|----------|
| Controller      | Temperature PID | Level PID |
| P               | 0.1808          | 0.1466    |
| I               | 0.00136         | 0.00011   |
| D               | 0                | 3.90516   |
| N (filter)      | 100              | 0.02291   |

The fuzzy logic controller used in this controller has two inputs and three outputs. The input used is error and derivative (delta) error. The resulting output is the change in parameter values P, I, and D (\( \Delta Kp, \Delta Ki, \Delta Kd \)). These three outputs are then used to change the PID parameter values that have been written on the SIMULINK simulation value. The fuzzy logic used is made using the Fuzzy Logic Designer tool available on MATLAB. This simulation uses the Fuzzy Inference System (FIS) with Mamdani type, And Method with MIN type, Or Method with MAX type, implication with MIN type, Aggregation with MAX type, and Defuzzification with Centroid type. All input variables (error and delta error) and output (dKp, dKi, dKd) used in Fuzzy have 7 membership functions. The seven membership functions are Negative Big (NB), Negative Medium (NM), Negative Small (NS), Zero (Z), Positive Small (PS), Positive, Medium (PM), and Positive Big (PB).

| Table 2. Fuzzy Rules of AFPIDC Controller. |
|-----------------|----------|----------|
| \( E, dE \) \( (Kp, Ki, Kd) \) | \( NB \) | \( NM \) | \( NS \) | \( Z \) | \( PS \) | \( PM \) | \( PB \) |
| \( NB \)         | PB, NB, PS | PB, NB, NS | 'M, NM, NB | PM, NM, NB | PS, NS, NB | Z, Z, NM | Z, Z, PB |
| \( NM \)         | PB, NB, PS | PB, NB, NS | 'M, NM, NB | PS, NS, NM | PS, NS, NM | Z, Z, NS | Z, Z, Z  |
| \( NS \)         | PM, NB, Z  | PM, NM, NS | 'M, NS, NM | PS, NS, NM | Z, Z, NS | NS, PS, NS | NS, Z, Z  |
| \( Z \)          | PM, NM, Z  | PM, NM, NS | PS, NS, NS | Z, Z, NS | NS, PS, NS | NM, PM, NS | NM, PB, Z |
| \( PS \)         | PS, NS, Z  | PS, NS, Z  | Z, Z, Z    | NS, PS, Z | NS, PS, Z | NM, PM, Z  | NM, PB, Z |
| \( PM \)         | PS, Z, PB  | Z, Z, NS   | NS, PS, NS | NM, PS, PS | NM, PM, PS | NM, PB, PS | NB, PB, PB |
| \( PB \)         | Z, Z, PB   | Z, Z, PM   | NM, PS, PM | NM, PM, PM | NM, PM, PM | NB, PB, PS | NB, PB, PB |
The value range throughout the membership function of input and output fuzzy selected have identical values, from -6 to 6. To perform tuning at the input and output, given the scaling on each side. There are three scaling constants used, namely Ke (scaling error), Kde (scaling delta error), and Ku (scaling output). The greater the Ke value will give bigger overshoot and smaller steady-state error, the bigger Kde value will produce a faster settling time, and the bigger Ku value will enlarge the oscillation in the controller. Fuzzy rules used in AFPIDC take references that have been proven to control certain parameters [7]. Each output (dKp, dKi, and dKd) has a relationship with the error value and delta error. However, AND logic is used to combine multiple outputs at one input combination value. Therefore, a total of 49 fuzzy rules were obtained for all input and output combinations. For more details, fuzzy rules can be seen in table 2. Figure 2 shows the detail of complete diagram for both Temperature and Level AFPIDC Controller.

![Figure 2. Detailed Diagram of AFPIDC Controller for (a) Temperature Control and (b) Level Control.](image)

3. Result and Discussion

3.1. AFPIDC Tuning

The tuning of the AFPIDC level and temperature controllers were carried out on the scaling constants Ke, Kde, and Ku by trial-and-error method. In the AFPIDC temperature controller, the greater the Ke value, the smaller the oscillations that appear, but the slower the process response becomes stable. Likewise, at the Kde value, the greater the Kde, the smaller the oscillation but the slower the process response. For Ku values, the greater the Ku used, the smaller the overshoot that appears, and the slower the process response. Therefore, the three values of Ke, Kde, and Ku which have the smallest oscillation but have the fastest time are taken, namely the values of Ke: 3, Kde: 180, and Ku: 0.01. For AFPIDC Level controller, the three Ke, Kde, and Ku values which have the smallest oscillation but have the fastest time are taken, namely the Ke values: 10, Kde: 582, and Ku: 0.0017.

3.2. Setpoint Test

In testing the system by performing a set point change, system performance is seen from the rise time (second), settling time (second), percent overshoot, and steady-state error. The test was carried out with many changes in the setpoint, namely the difference between small, medium, and large set points in the up and down directions, respectively. Tests were carried out on PID controllers for level and temperature and AFPIDC controllers for level and temperature. Each setpoint change will be calculated the value of the four parameters, and the average is taken. The first test is the setpoint change of temperature.
Figure 3. PID and AFPIDC Controller Curve in (a) Temperature Response and (b) Level Response

AFPIDC controller have better rise time, settling time, and percent overshoot which is smaller than the PID controller for temperature. Besides, the AFPIDC controller also does not have the same steady-state error value as the PID controller. In controlling the water temperature, the AFPIDC controller has a faster response and smaller overshoot because there is a Fuzzy Logic controller.

Table 3. Temperature Response by PID and AFPIDC Controller Performance Analysis.

| Setpoint        | (30-40 °C) | (40-60 °C) | (60-85 °C) | (85-60 °C) | (60-40 °C) | (40-30 °C) |
|-----------------|------------|------------|------------|------------|------------|------------|
| Controller      | PID        | AFPIDC     | PID        | AFPIDC     | PID        | AFPIDC     |
| Rise Time       | 240s       | 236s       | 225s       | 225s       | 241s       | 231s       |
| Settling Time   | 827s       | 335s       | 833s       | 324s       | 829s       | 332s       |
| Overshoot       | 6.4%       | 0%         | 0%         | 0%         | 0%         | 0%         |
| Error           | 0%         | 0%         | 0%         | 0%         | 0%         | 0%         |

Controlling Fuzzy logic is made AFPIDC can adjust depending on the needs of the value of the error and change of error. This can be seen more clearly from the fuzzy rule table which can manipulate PID constants depending on the error situation and the error changes in real-time.

Table 4. Level Response by PID and AFPIDC Controller Performance Analysis.

| Setpoint        | (0,1-0,2) m | (0,2-0,4) m | (0,4-0,6) m | (0,6-0,4) m | (0,4-0,2) m | (0,2-0,1) m |
|-----------------|-------------|-------------|-------------|-------------|-------------|-------------|
| Controller      | PID         | AFPIDC     | PID         | AFPIDC     | PID         | AFPIDC     |
| Rise Time       | 407s        | 118s        | 425s        | 177s        | 407s        | 111s        |
| Settling Time   | 3194s       | 167s        | 3208s       | 157s        | 3239s       | 150s        |
| Overshoot       | 10.5%       | 0%          | 10.5%       | 1.4%        | 10.5%       | 1.8%        |
| Error           | 0%          | 1.05%       | 0%          | 0.52%       | 0%          | 0.26%       |

It can be seen on the level response curve graph that the AFPIDC controller is better than the PID controller for controlling the water level. This advantage can be seen from the value of the rise time, settling time, and percent overshoot which is smaller than the PID controller for temperature. Also, in contrast to the PID controller, AFPIDC has a small but insignificant steady-state error value. This steady-state error value may arise because of the scaling constant whose order is far different from the order of the PID constant. In controlling the water level, the AFPIDC controller also has a faster
response and less overshoot because of the Fuzzy Logic controller. Controlling Fuzzy logic is also made to suit the needs AFPIDC depending on the value of the error and change of error. This can be seen more clearly from the fuzzy rule table which can manipulate PID constants depending on the error situation and the error changes in real-time. Table 5 contains the averaged performance of both PID and AFPIDC in controlling both Temperature and Level.

| Table 5. Averaged Controller Performance Analysis |
|--------------------------------------------------|
| Controller | Temperature | Level |
|            | PID | AFPIDC | PID | AFPIDC |
| Rise Time  | 239.83 s | 227 s | 412.16 s | 118 s |
| Settling Time | 830 s | 328 s | 3221 s | 235 s |
| Overshoot | 6.3% | 0% | 10.5% | 1.8% |
| ESS | 0% | 0% | 0% | 1.2% |

3.3. Disturbance Test
The simulated disturbance is the leakage of hot and cold-water flow values at a certain period. Leaks are provided by reducing the simulated flow of water from the control valve block. After a leak occurs for a certain period, the leak will disappear so that the flow will return to normal. The system response will be seen when compensating to stabilize the output temperature and level. The performance of the process against this disturbance is seen from how long it takes the system to stabilize after being disturbed. Performance controllers AFPIDC for level and temperature requires a time that is faster (225 seconds and 250 seconds respectively) to return stable than the PID controller for level and temperature (4510 seconds and 780 seconds respectively). Therefore, the AFPIDC controller has better performance to deal with system problems.

4. Conclusion
After carrying out the entire process in this research, from starting to build system modeling, building, and tuning the controller, and testing the controller, we obtained the following conclusions. Firstly, the test results indicate that the controller AFPIDC had a response that quickly, look at the value of rise time, and settling time, which is smaller than a PID controller. The second conclusion is the system test results show that the AFPIDC controller has a very small percent overshoot compared to the PID controller. The AFPIDC controller also has a steady-state error value. However, the steady-state error value is quite small and not significant. The AFPIDC controller has better interference resistance than the PID controller. It can be seen from the time it takes AFPIDC to return to a stable state faster than the PID controller. In the AFPIDC controller, fuzzy tuning by changing the scaling input - fuzzy output is very important to get the desired performance.

Acknowledgement
This research study is supported by research grants of Publikasi Terindeks Internasional (PUTI) Saintekes Universitas Indonesia, Grant No. NKB-4898/UN2.RST/HKP.05.00/2020.

References
[1] Fini A, Gogani M and Pourgholi M 2015 Fuzzy gain scheduling of PID controller implemented on real time level control 4th Iran Jt. Congr. Fuzzy Intell. Syst. CFIS 1–5
[2] Paulusova J, Orlicky L and Dubravska M 2013 Self-tuning fuzzy PID controller Proc. 2013 Int. Conf. Process Control 304–308
[3] Mazlan Z and Ibrahim R 2012 Development and implementation of Adaptive Fuzzy PID Controller (AFPIDC) for flow control application Int. Conf. Intell. Adv. Syst. A Conf. World Eng. Sci. Technol. Congr. - Conf. Proc. 1 1–6.
[4] Hulinjing, Zhangke and Liutao 2012 Study on the boiler drum water level based on Fuzzy
adaptive control Proc. 2012 24th Chinese Control Decis. Conf. 1659–1663

[5] Seborg D, Edgar T, Melichamp D and Doyle F 2011 Process Dynamics and Controls, Third. John Wiley & Sons, Inc.

[6] Ahmad B and Prajitno P 2020 Design of neural network and PLC-based water flow controller Journal of Physics: Conference Series 1528(012065) 1-7

[7] Yue W and Liu Y 2009 Boiler drum level controlled by fuzzy self-adapting PID Asia-Pacific Conf. Comput. Intell. Ind. Appl. 2 381–384