Vol. 3(2), pp. 1-19, December 2024
eISSN: 2976-4009

ASEAN Journal of Process Control

*Research Article*

# Metaheuristic Inspired Algorithm for Supervisory Fuzzy Controller for Plant wide

**Khalid Abdirashid Sulaiman [1], Pui Vun Chai [1*], Mohd Fauzi Zanil [2]**

[1] Department of Chemical and Petroluem Engineering, Faculty of Engineering, Technology and Built Environment, 56000, Cheras, Kuala Lumpur, Malaysia

[2] School of Engineering, Asia Pacific University of Technology and Innovation, Kuala Lumpur, Malaysia

*Corresponding Author: chaipv@ucsiuniversity.edu.my

**Abstract:** Plantwide faces challenges to obtain controllers that are fit to correct errors from actual and desired inputs. This makes industries such as methanol production industry to have nonlinear process in which the process conditions such as temperature, pressure, and level are interacting each other and causing to reduce product quality, equipment damage, explosion, and flooding. Therefore, the goal of this research is to investigate tuning algorithm performance for multiple loop control system by developing metaheuristic tuning algorithm performance for fuzzy logic controller and by developing dynamic process models for plantwide application. However, plantwide process dynamic is simulated using UniSim software for controlling temperature of cooler, pressure and level of flash drum. Mathematical modelling is performed to derive transfer function of the respective process while taking the values of variables from the process simulated in UniSim. Fuzzy Logic Controller (FLC) is designed as there are three inputs and outputs as each one have gaussian type membership function in which the final rules were defined. Since manually configuration control system is time consuming and abundant trial and errors to get the optimal parameters, Particle Swarm Optimization (PSO) is implemented to use as metaheuristic algorithm to optimise control performance by finding the best particle to update the best global through successive iterations and swarm size. Thus, it is found that control performance increased as the PSO used to optimize control system for Proportional-Integral-Gain (PID) and FLC. The result got shows significant decrease of overshoot and undershot and responding faster with less rise time and settling time after using PSO to optimise conventional controller that combined with FLC. For example, pressure is controlled quickly to reach stability within 1 second after optimised FLC and PID as it has rise time of $2.8704 \times 10^{-11}$ sec for the PSO-PID and $3.3242 \times 10^{-12}$ sec for Optimised FLC.

**Keywords:** Fuzzy Logic Controller, Particle Swarm Optimization, Plantwide Dynamic Process, Interacting Process, Dynamic Process Model

## 1. Introduction

Plantwide process include methanol production process is required to monitor process conditions such as pressure, temperature, and level. Usually, most plantwide processes contain varieties of equipments that combined such as cooler, heater, flash drum, distillation column, and fixed bed reactor. Therefore, complexity of plantwide process dynamic control increases as there are different process conditions such as temperature, pressure, and level that are interacting each other and leading to nonlinear process control system [1,2]. Nonlinearity characteristics indicates difficulty to control the plantwide process and requirement [3–5] to design controllers that can perform excellent to control process conditions fast, reduce overshoot or undershoot, and cancelling the disturbance. Plantwide process selected in this research is methanol production for specific equipments as they are flash drum and cooler to design a controller that can control perfectly pressure and level in flash drum also temperature in cooler. However, low quality controllers can cause damage to industrial equipments, and product quality [6] to avoid this selected plantwide process is simulated dynamically in UniSim software to assess performance of the controllers to further optimize and improve their parameters by using Particle Swarm Optimization (PSO) then optimized conventional controllers connected to Fuzzy Logic Controller (FLC) to improve controller performance.

Fuzzy Logic Controller (FLC) is really important when it comes to nonlinear process control as FLC is more flexible and robustness [6]. FLC can able to provide desirable output response for the process that related to Multiple input and multiple output variables which it can reduce settling time, rise time, and can make overshoot and undershoot to become zero. The fuzzy logic controller tipping consists of three main components: the fuzzifier, which transforms input values into fuzzy linguistic variables; the inference engine, which combines membership function degrees with input variables to produce appropriate output fuzzy sets with corresponding membership degrees; and the defuzzifier, which offers the last control action that can be applied to the system [2].

Nevertheless, in 1995 Drs. Eberhart and Kennedy created Particle Swarm Optimization (PSO), a population-based stochastic optimization technique that was motivated by the social behaviour of fish schools and flocks of birds by finding best particle and updating global best through iterations [7]. PSO requires less effort to implement and at the same time it has a large potential to get optimal values for different problems within short time. Therefore, PID parameters optimized by using PSO to minimize cost function as the PID requires more time to tune its parameters manually so, using PSO is best option to improve controller parameters. After using, PSO on FLC it is found that overshoot and undershoot are reduced also the rise time and settling time are decreased compared to manual PID conventional controller.

## 2. Materials and Methods

### *2.1. Plantwide Dynamic Simulation*

Program that is used for simulating plantwide is Honeywell UniSim. So, methanol process will be simulated in UniSim for Flash drum and Cooler equipment while using Peng Robinson fluid package. Therefore, in order to simulate the process, the components to be selected are: Methanol, Water, Hydrogen, Carbon dioxide, Carbon monoxide [8]. Each component has specific percentage fraction so methanol has 0.1, water has 0.8, hydrogen has 0.033, carbon dioxide has 0.033, and carbon monoxide has 0.033.

#### *2.1.1 Dynamic simulation for Level*

Level parameter is important to maintain at certain percentage of vessel height because maintaining plantwide level avoids flooding caused by the flowrate. Proportional (P) controller is selected to control level of the flash drum while the set point that selected is 60%.

#### *2.1.2 Dynamic simulation for Pressure*

UniSim dynamic mode will provide a sight from response of controller from the setpoint through a computer plantwide simulation. Therefore, Proportional-Integral-Derivative (PID) controller is selected to be pressure controller of flash drum in order to match with desirable set point as it is 260 kPa and all the parameters of this type of controller were defined.

#### *2.1.3 Dynamic simulation for Temperature*

In industries it is crucial to control the temperature as choosing low performance temperature controller can cause overheating and overcooling that might reduce product quality. However, to design a suitable temperature Proportional-Integral-Derivative (PID) controller is chosen and all three parameters were defined to initiate dynamic simulation as the set point to achieve is 25 °C.

### *2.2. Mathematical Modelling*

Moreover, in order to do derivation for getting transfer function of the control system several steps are taken which are firstly to do mass or energy balance of the respective equipment or respective process, after that differential equation is served to make the equation in explicit form and able to take inverse Laplace transform for both sides of the equation. After the inverse Laplace transform is used on the equation, the equation is arranged to standard form to get the transfer function for $k_{p1}$ and $k_{p2}$ as there are three transfer functions derived as shown in Eq (1), Eq (2), and Eq (3) for each pressure process, level process, and temperature process of flash drum and cooler respectively. However, $k_{p1}$ and $k_{p2}$ are defined for their equivalent fractions, those fractions have values corresponding to UniSim simulation design, so their values is referred to process that simulated in UniSim then these values replaced to $k_{p1}$and $k_{p2}$.

$$P(s) = G_P(s) = \frac{k_{p1}}{k_{p2}+1} Q_i(s) - \frac{k_{p1}}{k_{p2}+1} H(s) \tag{1}$$

$$H(s) = G_H(s) = \frac{k_{p1}}{k_{p2}+1} Q_i(s) - \frac{k_{p1}}{k_{p2}+1} P(s) \tag{2}$$

$$T'(s) = G_T(s) = \frac{k_{p1}}{k_{p2}+1} T_i'(s) - \frac{k_{p1}}{k_{p2}+1} Q'(s) \tag{3}$$

### *2.3. Block Flow Diagram Modelling in Simulink*

Blocks for pressure, temperature and level were modelled, since there are interactions from flash drum process to control the level and pressure also interactions from cooler to control the temperature. In order to control level of flash drum outlet bottom valve has a direct effect to manipulate the flash drum level but, the top outlet flowrate and cooling coil changes have indirect effect to manipulate the flash drum level. In addition to that, to control pressure of flash drum, outlet top valve has a direct effect to manipulate the flash drum pressure but, the bottom outlet flowrate valve and the cooling coil have indirect effect to manipulate the flash drum pressure. Moreover, to control process temperature heat leaving from the cooling coil has a direct effect to manipulate the process temperature output in the cooler, but the bottom outlet flowrate valve and top outlet flowrate valve changes in the flash drum have indirect effect to manipulate the output process temperature in the cooler.

**Table 1.** Conventional controller parameters.

| Parameters | Level controller | Pressure controller | Temperature controller |
|---|---|---|---|
| Kp | 800 | 14 | 0.011 |
| Ki | - | 1.52 | 0.9 |
| Kd | - | 0.378 | 0.08 |

### *2.4. Fuzzy Logic Controller (FLC) Design*

Fuzzy logic controller tipping contains three main parts which are fuzzifier and its function is to convert input values into fuzzy linguistic variables, and the second part is inference engine it allows to combine the membership function degrees with the input variables in order to achieve suitable output fuzzy sets with their associated membership degrees, also the third part of the fuzzy logic controller is defuzzifier, it provides the final control action that is possible to apply the system [9]. In this paper it is used Mamdani based inference and defuzzification of the centre of the gravity [10]. However, inputs that selected are three as they are temperature, pressure and level also outputs that selected are three and they are cooling coil, valve 101, and valve 100. Each of these inputs and outputs contain three membership functions and the type of membership used is gaussian, the number of rules that created manually is 21.

**Table 2.** Inputs and their membership functions for manual Fuzzy Logic Controller.

| Type of MF used | Input membership function of Temperature | | Input membership function of Pressure | | Input membership function of Level | |
|---|---|---|---|---|---|---|
| | Fuzzy variable | Parameters value | Fuzzy variable | Parameters value | Fuzzy variable | Parameters value |
| Gaussian | Cold | [304 -962.1] | Low | [353.7 -118.9] | Negative | [197 -341.7] |
| Gaussian | Average | [253 -61.4] | Normal | [545.7 891] | Ok | [299.6 362] |
| Gaussian | Hot | [343.9 848.1] | High | [379.7 1944] | Positive | [139 981] |

**Table 3.** Outputs and their membership function for manual Fuzzy Logic Controller.

| Type of MF used | Output membership function of cooling coil (Q-100) | | Output membership function of valve_101 | | Output membership function of valve_100 | |
|---|---|---|---|---|---|---|
| | Fuzzy variable | Parameters value | Fuzzy variable | Parameters value | Fuzzy variable | Parameters value |
| Gaussian | Decrease | [14.82 -3.049] | Close | [11.87 27.28] | Close | [11.47 111] |
| Gaussian | Maintain | [16.03 49.33] | No change | [13.22 65.5] | Un change | [13.26 125.4] |
| Gaussian | Increase | [18.9 108] | Open | [11.87 102.7] | Open | [12.57 142] |

**Table 4.** Tabulated fuzzy rules.

| | | Cooling coil (Q-100) | Valve 101 | Valve 100 |
|---|---|---|---|---|
| **Temperature** | Cold | Decrease | Close | Close |
| | Average | Maintain | No change | Un-change |
| | Hot | Increase | Open | Open |
| **Pressure** | Low | Decrease | No change | Un-change |
| | Normal | Decrease | Open | Open |
| | High | Maintain | Close | Close |
| **Level** | Negative | Maintain | Open | Open |
| | Ok | Increase | Close | Close |
| | Positive | Increase | No change | Un-change |

### *2.5. Particle Swarm Optimization Implementation*

In order to perform particle swarm optimization Initialization of algorithmic parameter is required in this stage, the number of decision variables, their range, swarm size, inertia weight factor, damping inertia coefficient, constants for cognitive and social, and the number of iterations are defined as shown in Table 5. Since the number of decision variables are seven, the objective function of each particle was evaluated, and fitness formula selected is Root Mean Square Error (RMSE) as shown in Eq (4) to minimize cost function [11] during PSO iterations and optimize PID. However, the present fitness value of each particle is compared to its prior best value Particle best (Pbest), if the current value is lower than the previous Pbest value, then the current value will be used as the Pbest value and Global best (Gbest) is updated through updating velocity and position of each particle. Finally, the algorithm examines for stopping criteria by using Eq (5) and Eq (6), iteration is terminated as the maximum number of iterations reached is 100 and swarm size is 30. Finally, after PID is optimized it is connected to FLC model in order to improve controller performance [6].

$$RMSE = \sqrt{\frac{\int (Predicted_i - Actual_i)^2}{T}} \tag{4}$$

*Where,*

*$Predicted_i$ is the value for the ith observation.*

*$Actual_i$ is the value for the ith observation.*

*T is the measuring time.*

$$v_i^{k+1} = w * v_i^k + c_1 * rand_1 * \left(p_{best_i} - x_i\right) + c_2 * rand_2 * \left(g_{best_i} - x_i\right) \tag{5}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{6}$$

*Where,*

*$w$ = inertia weight factor*

*$c_1, c_2$ = cognitive constant and social constant respectively*

*$v_i^k$ = velocity of particle i in $k^{th}$ iteration*

*$x_i^k$ = position of particle i in $k^{th}$ iteration*

*$rand_1$, $rand_2$ = random number between 0 and 1*

**Table 5.** PSO parameters.

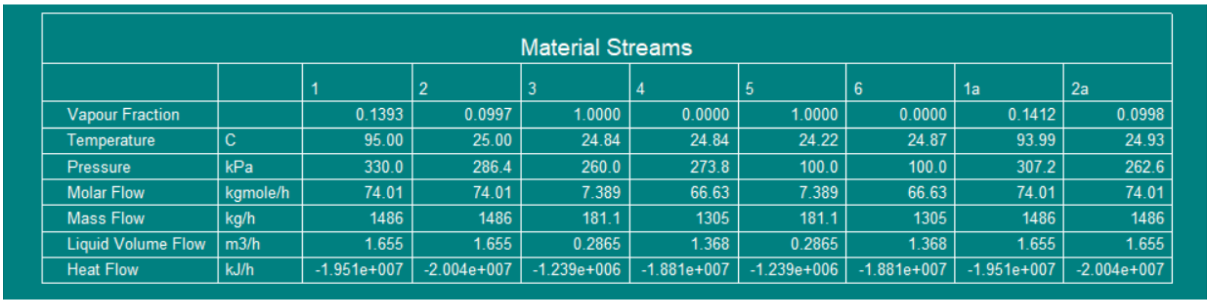
| **Parameters of PSO** | **Value** |
|---|---|
| Inertia weight factor, w | 1 |
| Damping inertia coefficient, wdamp | 0.99 |
| Cognitive constant, c1 | 1 |
| Social constant, c2 | 1 |
| Swarm size | 60 |
| Iteration number | 100 |
| Lower bound boundary, Lmin | -0.0001 |
| Upper bound boundary, Lmax | 0.0001 |

# 3. Results and Discussion

## *3.1. Plantwide Dynamic Simulation*

Methanol plantwide is simulated in UniSim to define process dynamic control. Flash drum and cooler are the two equipments that chosen to design a suitable control system for pressure, level, and temperature. Therefore, fluid package used is Peng Robinson as the components added are Methanol, Water, Carbon dioxide, and Carbon monoxide. Methanol has a percentage fraction of 0.1, water has 0.8, hydrogen has 0.033, carbon dioxide has 0.033, and carbon monoxide has 0.033 [8]. where the cooler's inlet and exit pressures are equal to 3.3 bar and 3.1 bar, respectively, and its temperature input is 95 °C with a constant flowrate of 1349 kg/h. The liquid and vapor exits of the flash drum have the same temperature and pressure as the feed stream, with flowrates of 878.5 kg/h and 121.5 kg/h, respectively. The feed temperature and pressure are 25 °C and 3 bar, respectively, with a flowrate of 1000 kg/h.

**Table 6**. Material streams of methanol plantwide process flow diagram.



Material Streams

| | | 1 | 2 | 3 | 4 | 5 | 6 | 1a | 2a |
|---|---|---|---|---|---|---|---|---|---|
| Vapour Fraction | | 0.1393 | 0.0997 | 1.0000 | 0.0000 | 1.0000 | 0.0000 | 0.1412 | 0.0998 |
| Temperature | C | 95.00 | 25.00 | 24.84 | 24.84 | 24.22 | 24.87 | 93.99 | 24.93 |
| Pressure | kPa | 330.0 | 286.4 | 260.0 | 273.8 | 100.0 | 100.0 | 307.2 | 262.6 |
| Molar Flow | kgmole/h | 74.01 | 74.01 | 7.389 | 66.63 | 7.389 | 66.63 | 74.01 | 74.01 |
| Mass Flow | kg/h | 1486 | 1486 | 181.1 | 1305 | 181.1 | 1305 | 1486 | 1486 |
| Liquid Volume Flow | m3/h | 1.655 | 1.655 | 0.2865 | 1.368 | 0.2865 | 1.368 | 1.655 | 1.655 |
| Heat Flow | kJ/h | -1.951e+007 | -2.004e+007 | -1.239e+006 | -1.881e+007 | -1.239e+006 | -1.881e+007 | -1.951e+007 | -2.004e+007 |

### *3.1.1. Dynamic simulation for Level*

P controller is chosen which contains only gain parameter in order to provide suitable value for the parameter level controller was installed on flash drum as the rage percentage of the controller is from 0% to 100%. So, process variable is selected to be level of flash drum as the output object selected is VLV-100. Process gain value that inserted is 800 in order to make the controller active to respond level process changes. Its analysed gain controller by observing overshoot and undershoot of the level process curve by comparing the setpoint as the setpoint is 60% of flash drum level. As a result, it is found that the controller can change the flash drum's level percentage by adjusting the valve opening at 29.78% from the step input of 60%. Despite this, the controller has overshot because it takes just 6 minutes to achieve a stable state, as seen in figure 2. Lastly, the blue colour curve represents the "output object," the red colour curve represents the "set point," and the green colour curve indicates the "process variable."

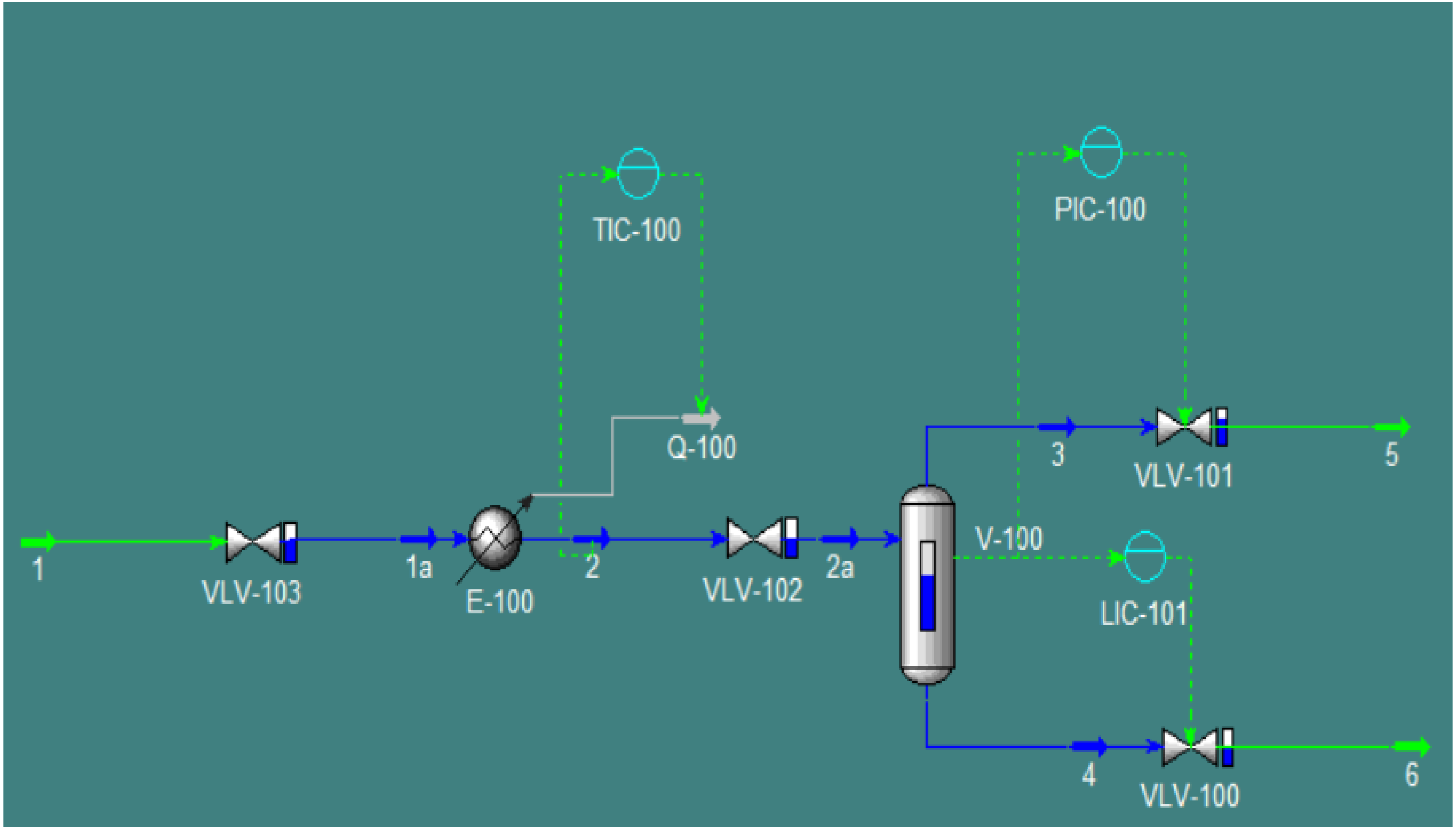


**Figure 1.** Process flow diagram of methanol plantwide process dynamic simulation.

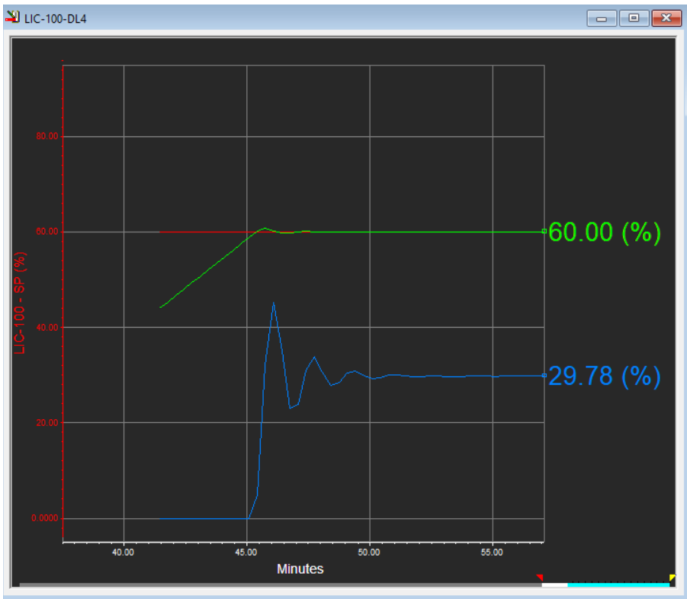


**Figure 2.** Strip chart of the level controller.

### *3.1.2. Dynamic Simulation for Pressure*

Since many industries rely on the type of control being applied to maintain pressure at a desired set point, plantwide pressure becomes extremely vital to give appropriate control. To analyze the behavior of pressure control, a simulation of the flash drum dynamics mode is performed. Since the output object is the VLV-101 and the process variable is the flash drum pressure, a PID controller is utilized for pressure control. The action is configured to be direct, meaning that the valve opening will rise in response to an increase in the vessel pressure. The pressure range chosen is between 2.3 bar and 3 bar as the setpoint is 260 kPa. PID tuning settings are given with Td equaling 0.378 sec, Ti equaling 1.52 sec, and Kc equaling 14. Controller response shows that the time taken to become stable is in one minute as shown in figure 3. The overshoot is minimal for the pressure control that leads the controller to take less time interval to become stable to change the valve opening.

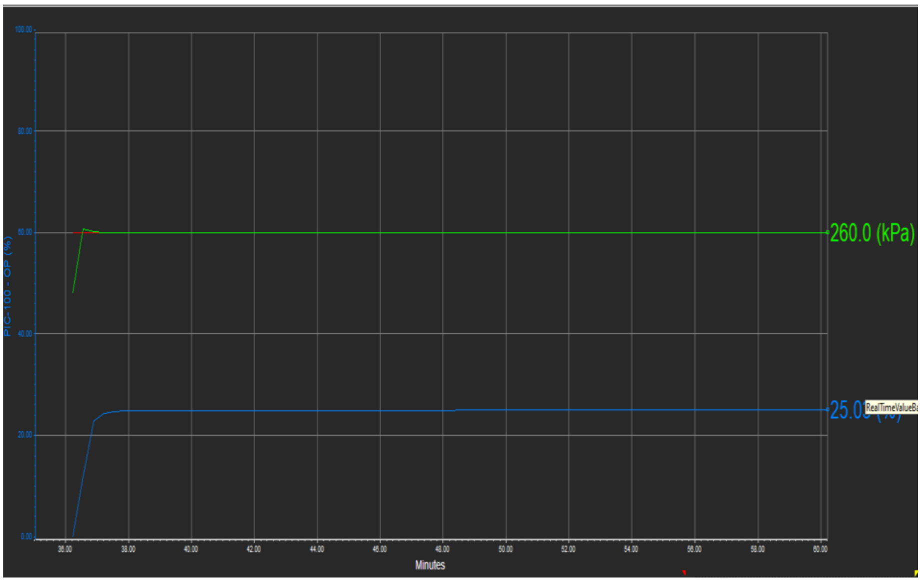


**Figure 3.** Strip chart of pressure controller.

### *3.1.3. Dynamic simulation for Temperature*

First, in order to minimize errors from output response with set point temperature, a temperature controller needs to be designed to be able to handle or respond to slight variations. The temperature has a preset process range of 15 to 25 °C as the PID controller installed on the cooler. An Auto mode controller will result in a Direct action. Additionally, 0.011, 0.9, and 0.08 seconds are the values for Kc, Ti, and Td, respectively, in PID tuning. A strip chart for the process variable, set point, and output object was generated in order to analyze the transient response. Referring to figure 4, the process variable curve indicates that it has overshoot to match the setpoint, and the process variable represented in green colour equalized to 25 °C, while the valve opening percentage is 3.11%. The setpoint is at 25 °C, where the process variable manipulated by the temperature PID controller.

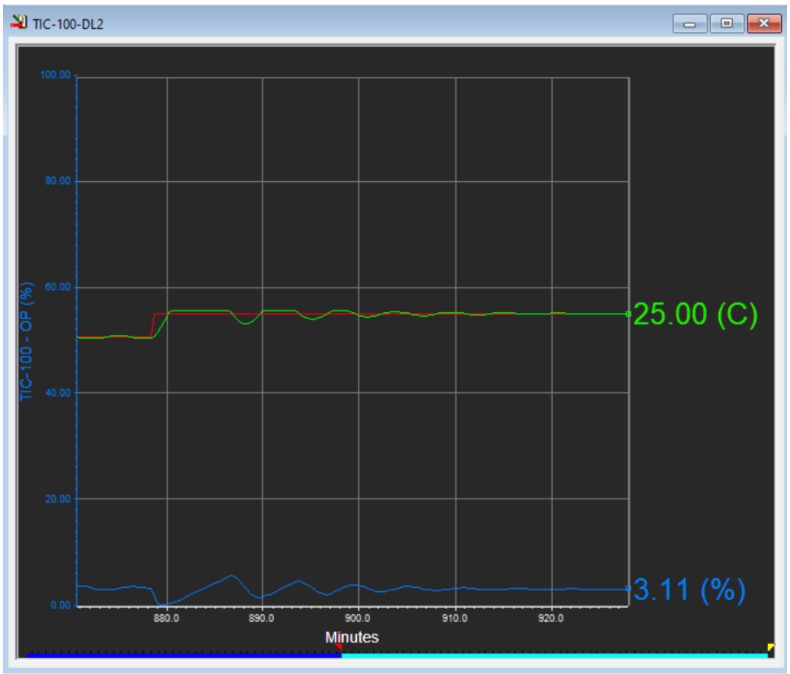


**Figure 4.** Strip chart of temperature controller.

### *3.2. Mathematical Modelling*

Transfer function of the process can be derived from mathematical model of the flash drum and cooler by using differential equation and inverse Laplace transform. As the flash drum contains pressure and level where the cooler contains temperature. So, it is used mass and energy balance for modelling the equipments, by referring Eq (7) and Eq (8) flash drum and cooler mathematical model involve on mass balance and energy balance as shown in figure 5. Transfer function can give opportunity to model targeted control system in Simulink to further improve control performance of the respective process. By referring to Eq (9), Eq (10), and Eq (11) they represent transfer function derived for pressure, level, and temperature where the respective $k_{p1}$ and $k_{p2}$ values taken from the simulated process in UniSim.

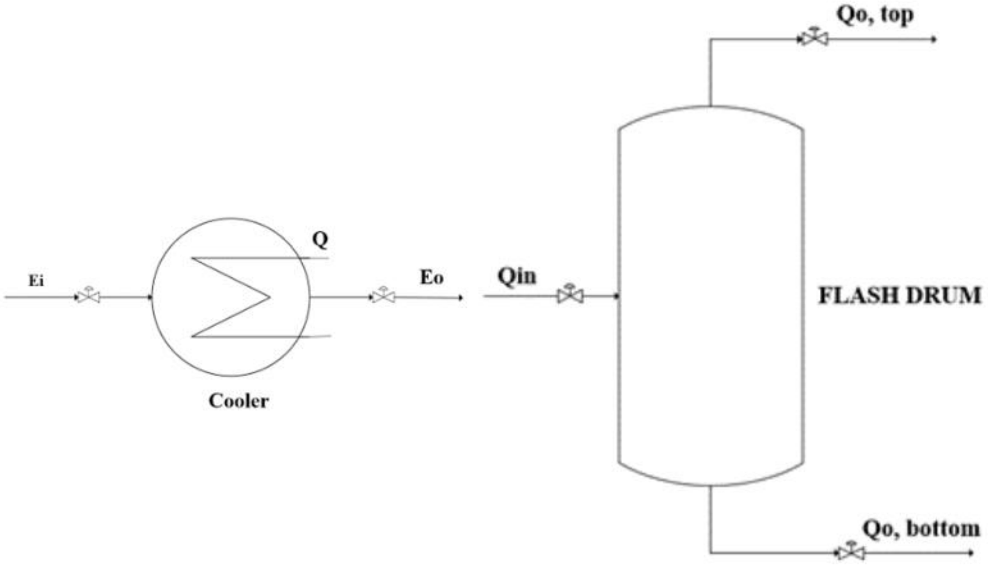


**Figure 5.** Flash drum and cooler process flow diagram for pressure, level, and temperature processes.

$$Q_T = Q_{in} - Q_{o,top} - Q_{o,bottom} \tag{7}$$

Where $Q_T$ is for the total mass flowrate, $Q_{in}$ is inlet mass flowrate, $Q_{o,top}$ is vapor outlet mass flowrate, and $Q_{o,bottom}$ is liquid outlet mass flowrate.

$$E_T = E_i - E_o - \mathrm{Q} \tag{8}$$

Where $E_T$ is for the total energy flowrate, $E_i$ is inlet energy flowrate, $E_o$ is outlet energy flowrate, and Q is the heat leaving from the cooling coil.

$$P(s) = \frac{R_1}{\frac{VR_1}{RT}s+1} Q_i(s) - \frac{1}{\frac{VR_1}{RT}s+1} H(s) \tag{9}$$

Where the equivalent value for $T$ (K), V (m$^3$), $R_1$ (m$^3$/min), and R ( $\frac{kJ}{kmol-K}$ ) is provided the Table shown below:

**Table 7.** Pressure transfer function parameters.

| **Temperature (T), K** | **Volume (V), m$^3$** | **Valve resistance ($R_1$), m$^3$/min** | **Universal gas constant, $\frac{kJ}{kmol-K}$** |
|---|---|---|---|
| 296.34 | 5 | 6.164 | 8.31434 |

$$H(s) = \frac{R_1}{R_1As+1} Q_i(s) - \frac{1}{R_1As+1} P(s) \tag{10}$$

Where the value for A (m$^2$), and $R_1$ (m$^3$/min) is inserted in Table 7:

**Table 8.** Level transfer function parameters.

| **Area (A), m$^2$** | **Valve resistance ($R_1$), m$^3$/min** |
|---|---|
| 2.058 | 2.072 |

$$T'(s) = \frac{1}{\frac{\rho V}{w}s+1}T_i'(s) - \frac{\frac{1}{wc}}{\frac{\rho V}{w}s+1}Q'(s) \tag{11}$$

Where the value for $w$ (kg/h), V (m$^3$), and c (kJ/kmol-°C) is provided in Table 8:

**Table 9.** Temperature transfer function parameters.

| **Mass flowrate ($w$), kg/h** | **Volume (V), m$^3$** | **Specific heat capacity (c), kJ/kmol-°C** |
|---|---|---|
| 1593 | 0.1 | 104.1 |

### *3.3. Block Flow Diagram Modelling in Simulink*

Since the process is nonlinear it is required to model the control system first then start optimization of control performance as shown figure 6. However, following system modelling, its critical to analyze the control system's behavior by producing a result. Hence, by referring to figure 7, for the level, pressure, and temperature output curves, scope is utilized to display the outcome of the control system of set point of pressure, level, and temperature as well as process output signals for pressure, level, and temperature. Since the set point to attain is 60%, the level output is controlled with 0.0142% and 60.1256% of overshoot and undershoot, respectively, when using a typical P controller. Table 6 indicates the settling time and rise time of 1.0125 seconds and 0.0062 seconds respectively. It takes longer to reach stability when the temperature setpoint is 25 °C and the response exhibits significant undershoot of 317.7037% and zero percentage overshoot, with settling times of 6.6309 sec and rising times of 2.1263 sec. Additionally, the pressure output response displays zero percentage of overshoot and undershoot, with settling and rise times of 1.002 seconds and 9.0840 x $10^{-5}$, respectively, indicating that the pressure set point is 260 kPa.

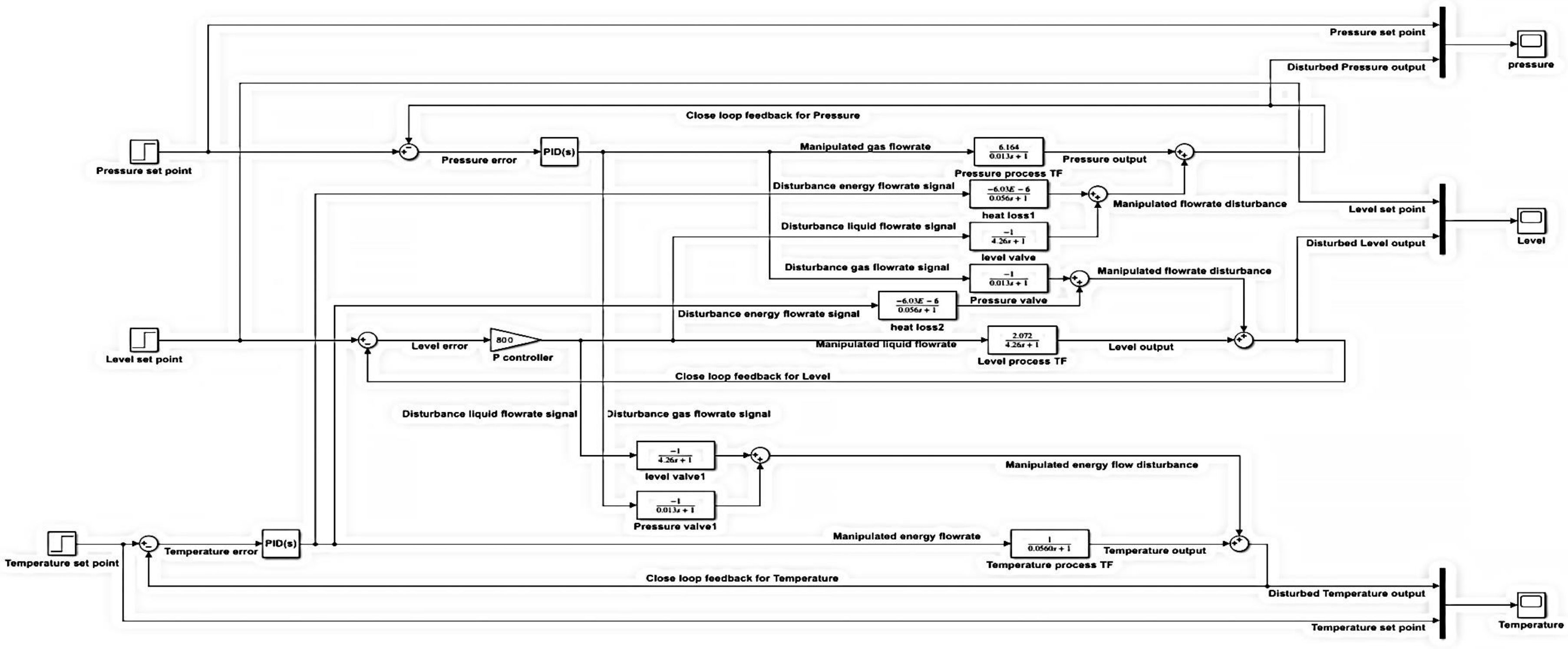


**Figure 6.** Simulink block flow diagram for pressure, temperature, and level processes.

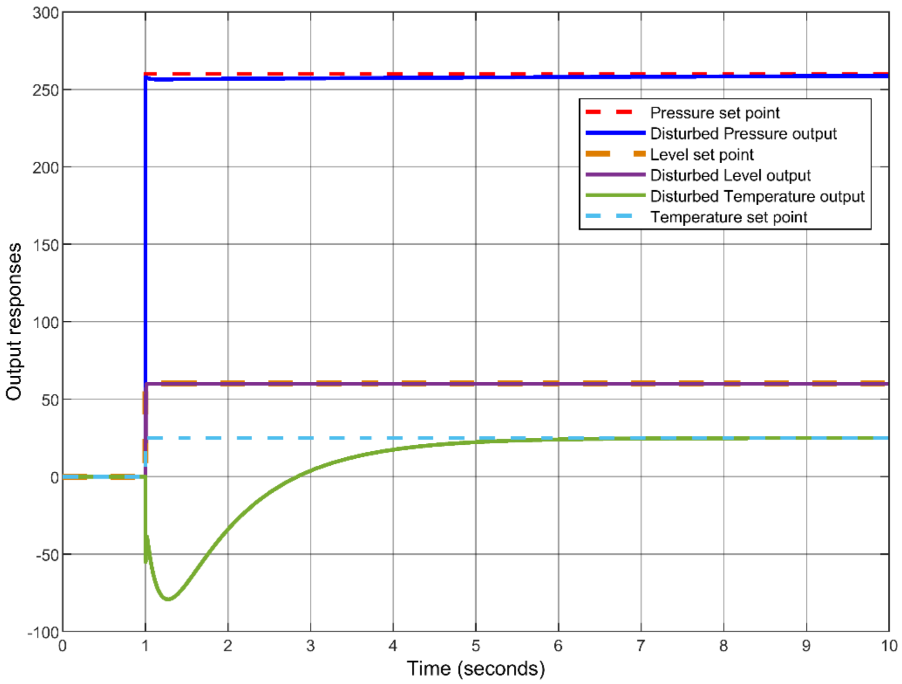


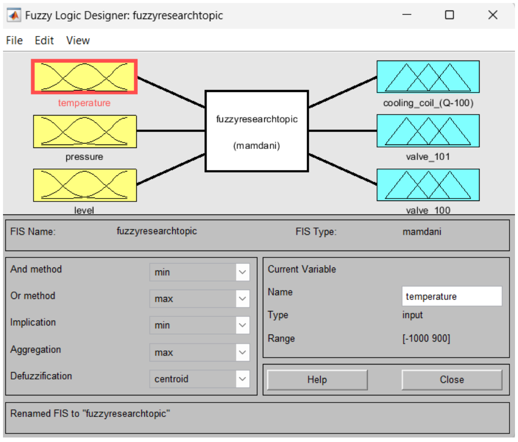
**Figure 7.** Set points and output responses for pressure, temperature, and level from conventional controller.

**Table 10.** Performance characteristics of conventional controllers.

| Process | Controller type | Set point | Overshoot (%) | Undershoot (%) | Settling time (s) | Rise time (s) |
|---|---|---|---|---|---|---|
| Pressure | PID | 260 kPa | 0 | 0 | 1.002 | $9.0840 \times 10^{-5}$ |
| Temperature | PID | 25 °C | 0 | 317.7037 | 6.6309 | 2.1263 |
| Level | P | 60% | 0.0142 | 60.1256 | 1.0125 | 0.0062 |

### *3.4. Fuzzy Logic Controller (FLC) Design*

Fuzzy logic controller contains three main parts which are fuzzifier, inference engine, and defuzzifier [2,9]. Fig. 8. illustrates how the design of a fuzzy logic controller is designed to produce the best possible graph output and to seamlessly optimize it using particle swarm optimization. As a result, the named inputs are level, pressure, and temperature, while the named outputs are the cooling coil, valves 101, and valve 100 as the type of membership functions used mentioned previously in the methodology part. To analyse FLC output responses, its created a graph that combines pressure controller, level controller, and temperature controller curves as shown in figure 9. Referring to Table 7 the temperature controller has been found to have the highest overshoot of any controller, at 217.1960%. The level controller has the highest undershoot of any controller, at 79.5807%, and the pressure controller has the lowest overshoot of any level or temperature controller, at 14.2033%. However, in contrast to other controllers, the pressure controller controls output response quickly, with a rise time of 0.0192 sec and a settling time of 6.9142 sec. Level controller requires more time to reach 60% of the flash drum level, with settling and rise times of 9.5573 sec and 0.0192 sec respectively. However, the temperature controller and pressure controller have the same rise and settling times of 0.0192 and 9.6569 seconds respectively.



FIS Name: fuzzyresearchtopic
FIS Type: mamdani
And method: min
Or method: max
Implication: min
Aggregation: max
Defuzzification: centroid
Current Variable
Name: temperature
Type: input
Range: [-1000 900]
Renamed FIS to "fuzzyresearchtopic"

**Figure 8.** Inputs and outputs of fuzzy logic controller.

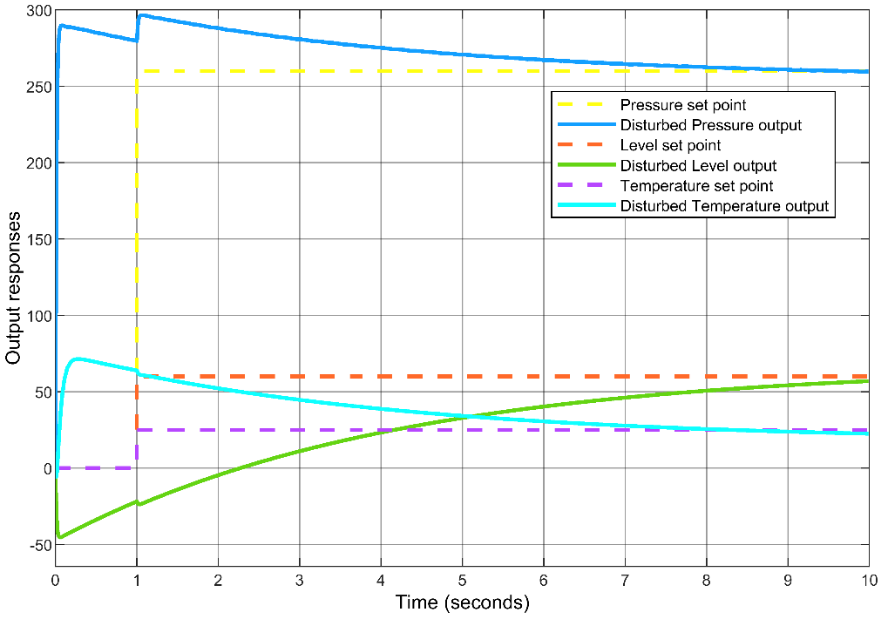


**Figure 9.** Set points and output responses generated from manual Fuzzy Logic Controller.

**Table 11.** Performance characteristics of manual Fuzzy Logic Controller.

| Process | Controller type | Set point | Overshoot (%) | Undershoot (%) | Settling time (s) | Rise time (s) |
|---|---|---|---|---|---|---|
| Pressure | FLC | 260 kPa | 14.2033 | 0 | 6.9142 | 0.0192 |
| Temperature | FLC | 25 °C | 217.1960 | 0 | 9.6569 | 0.0192 |
| Level | FLC | 60% | 0 | 79.5807 | 9.5573 | 5.5240 |

### *3.5. Particle Swarm Optimization*

This technique outperforms the conventional approach (such as the gradient method) in capturing better results [7]. However, parameters of optimization were defined to start optimization of control system and they are inertia weight factor as it is 1, damping inertia coefficient is 0.99, both cognitive and social constants equal to 1, swarm size and iteration number are 60 and 100 respectively, where the lower and upper bounds equal to -0.0001 and 0.0001 respectively [12] as the objective function formula is based on RMSE [11]. As these figures: 10, 11, and 12, showed the output responses gained after applied different techniques therefore its found that the optimum result is found when it is used Optimized FLC followed by PSO-PID techniques due to having improved performance of control system. Table 8 summarized all performance characteristics of different control approaches for example pressure can be controlled better when it is optimized FLC and PID by PSO as the settling time of both is 1 second to reach stability where the rise time for Optimized FLC is lower than the rise time of PSO-PID as the result showed 3.3242 x $10^{-12}$ sec rise time of Optimized FLC and 2.8704 x $10^{-11}$ sec rise time of PSO-PID. For the overshoot and undershoot its observed that both of them significantly decreased for temperature and level after the FLC and PID/P conventional controllers optimized by PSO as the undershoot of temperature is reduced from 317.7037 to 0 by Optimized FLC as shown in Table 8 where its overshoot reduced from 217.1960 to 0.0142.

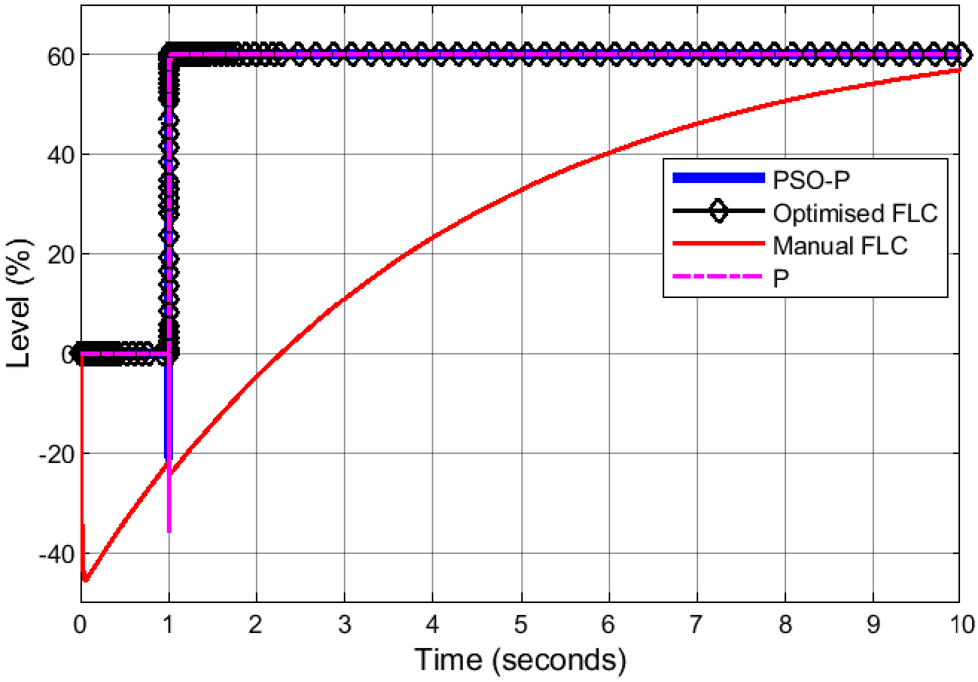


**Figure 10.** Comparison graph output curve response for level controller by using different control strategy.

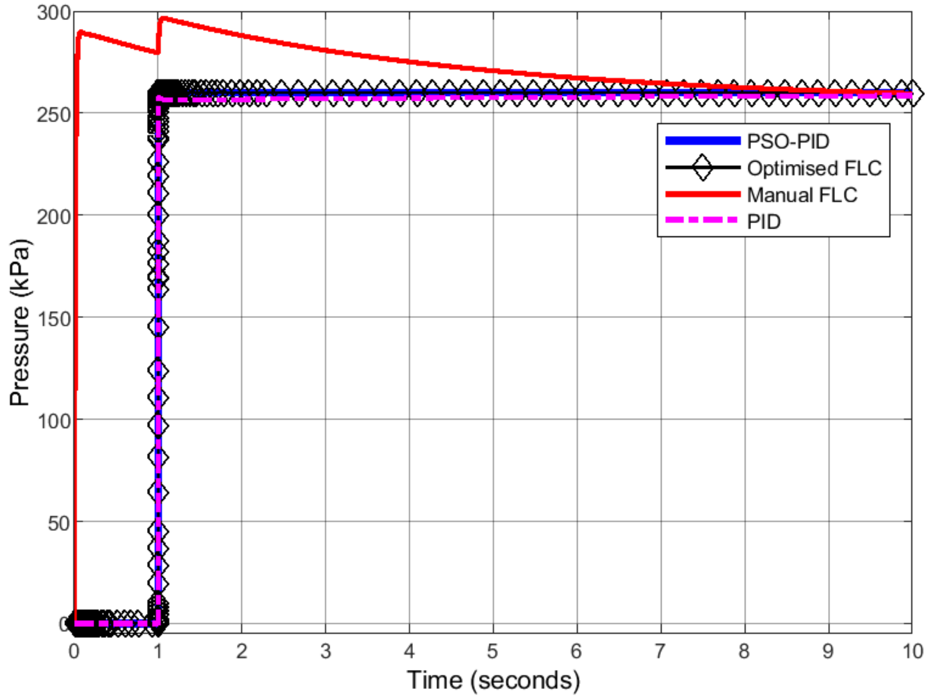


**Figure 11.** Comparison graph output curve response for pressure controller by using different control strategy.

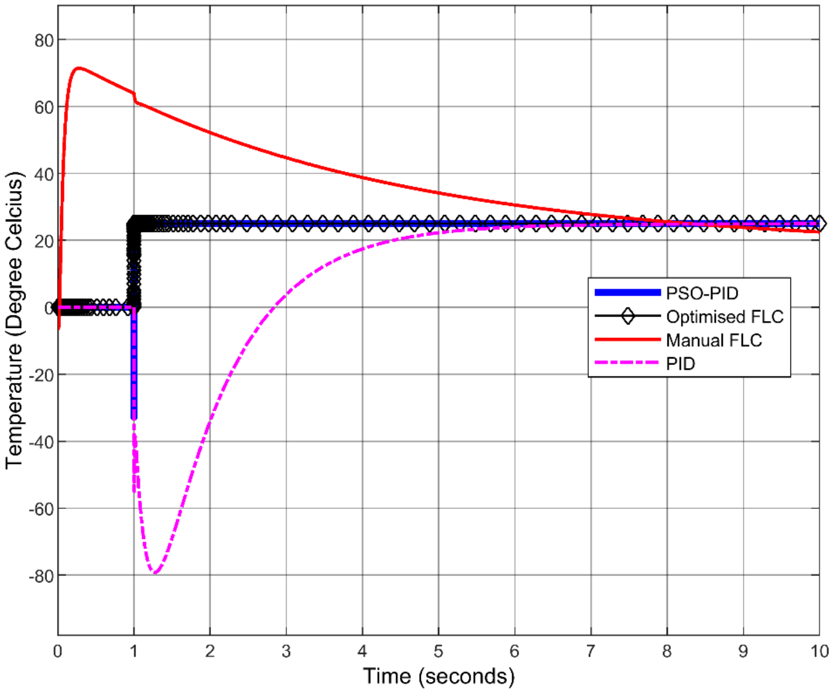


**Figure 12.** Comparison graph output curve response for temperature controller by using different control strategy.

**Table 12.** Comparison controller performance for different control strategies.

| Process | Set point | Controller type | Overshoot (%) | Undershoot (%) | Settling time (s) | Rise time (s) |
|---|---|---|---|---|---|---|
| Pressure | 260 kPa | PID | 0 | 0 | 1.002 | $9.0840 \times 10^{-5}$ |
| | | Manual FLC | 14.2033 | 0 | 6.9142 | 0.0192 |
| | | PSO-PID | $7.0042 \times 10^{-5}$ | 0 | 1.0000 | $2.8704 \times 10^{-11}$ |
| | | Optimised FLC | 0 | 0.0061 | 1.0000 | $3.3242 \times 10^{-12}$ |
| Temperature | 25 °C | PID | 0 | 317.7037 | 6.6309 | 2.1263 |
| | | Manual FLC | 217.1960 | 0 | 9.6569 | 0.0192 |
| | | PSO-PID | 0.0235 | 130.3897 | 1.0000 | $3.0641 \times 10^{-10}$ |
| | | Optimised FLC | 0.0077 | 0 | 1.0000 | $3.5709 \times 10^{-12}$ |
| Level | 60% | P | 0.0142 | 60.1256 | 1.0125 | 0.0062 |
| | | Manual FLC | 0 | 79.5807 | 9.5573 | 5.5240 |
| | | PSO-P | 0.0050 | 35.7333 | 1.0000 | $2.7906 \times 10^{-10}$ |
| | | Optimised FLC | 0.0131 | $1.5073 \times 10^{-8}$ | 1.0000 | $4.1460 \times 10^{-12}$ |

## 4. Conclusions

Control system performance improved after applied Particle Swarm Optimization (PSO) on Fuzzy Logic Controller (FLC) and Proportional Integral Derivative (PID) controller. The best settling time for all processes, according to optimized FLC and PSO-PID, is 1 second to reach steady state. Additionally, optimized FLC demonstrated that all process conditions required a lower rise time, indicating that it responds to changes in pressure, temperature, and level more quickly and robust. To control plant-wide process conditions by using Optimized FLC, the pressure, level, and temperature controllers have respective rise times of $3.3242 \times 10^{-12}$ sec, $4.1460 \times 10^{-12}$ sec, and $3.5709 \times 10^{-12}$ sec which are the lowest rise times among other controllers in this research. Lastly, it is recommended to increase the number of iterations for particle swarm optimization to optimize the traditional controller and further improve control performance.

**Acknowledgments:** First of all, I would like to express my grateful appreciation to my supervisors, Dr. Chai Pui Vun and Mr. Mohd Fauzi Bin Zanil for their excellent guidance and mentorship. Thanks for their endless encouragement throughout these few months, and their wise advises and suggestions to improve the outcome of the study. This dissertation would not been possible to complete without their input and guidance.

## References

1. C.-J. Wu, *Genetic Tuning of PID Controllers Using a Neural Network Model: A Seesaw Example*, Journal of Intelligent and Robotic Systems/1999.
2. H. Shahid, S. Murawwat, I. Ahmed, S. Naseer, R. Fiaz, A. Afzaal et al., *Design of a Fuzzy Logic Based Controller for Fluid Level Application*, World Journal of Engineering and Technology 04 (2016), pp. 469–476.
3. *Disturbance-Observer-Based Control and Related Methods - An Overview*. Institute of Electrical and Electronics Engineers Inc., 2016.
4. P. Leitão, *Agent-based distributed manufacturing control: A state-of-the-art survey*, Eng Appl Artif Intell 22 (2009), pp. 979–991.
5. L.-L. Xie and L. Guo, *How much Uncertainty can be Dealt with by Feedback?*, IEEE TRANSACTIONS ON AUTOMATIC CONTROL/2000.
6. M.I. Solihin, C.Y. Chuan and W. Astuti, Optimization of fuzzy logic controller parameters using modern meta-heuristic algorithm for gantry crane system (GCS), in Materials Today: Proceedings, 29 (2019), pp. 168–172.
7. *Metaheuristic algorithms for PID controller parameters tuning: review, approaches and open problems*. Elsevier Ltd, 2022.

**8.** P. Borisut and A. Nuchitprasittichai, *Process Configuration Studies of Methanol Production via Carbon Dioxide Hydrogenation: Process Simulation-Based Optimization Using Artificial Neural Networks*, .

**9.** Y. Bai and D. Wang, *2 Fundamentals of Fuzzy Logic Control-Fuzzy Sets, Fuzzy Rules and Defuzzifications*, .

**10.** G. García-Gutiérrez, D. Arcos-Aviles, E. V. Carrera, F. Guinjoan, E. Motoasca, P. Ayala et al., *Fuzzy logic controller parameter optimization using metaheuristic cuckoo search algorithm for a magnetic levitation system*, Applied Sciences (Switzerland) 9 (2019), .

**11.** Z. Bingül and O. Karahan, *A Fuzzy Logic Controller tuned with PSO for 2 DOF robot trajectory control*, Expert Syst Appl 38 (2011), pp. 1017–1031.

**12.** W. Der Chang and C.Y. Chen, *PID controller design for MIMO processes using improved particle swarm optimization*, Circuits Syst Signal Process 33 (2014), pp. 1473–1490.