DC Motor Speed Control Using Hybrid PID-Fuzzy with ITAE Polynomial Initiation

Hari Wibawa[1], Oyas Wahyunggoro[2], Adha Imam Cahyadi[3]

***Abstract*—DC motors are widely applied in industrial sector, especiallyprocesses of automation and robotics. Given its role in the sector, DC motor operation needs to be optimized. One of optimization steps is controlling speed using several control methods, for example conventional PID methods, PID Ziegler Nichols, PID based on ITAE polynomials, and Hybrid PID-Fuzzy. From these methods, Hybrid PID-Fuzzy was chosen as a method to be proposed in this paper because it can anticipate shortcomings of PID controllers and fuzzy controllers so as to produce system responses that are fast and adaptive to errors. This paper aimed to design a Hybrid PID-Fuzzy system based on ITAE polynomials (Hybrid-ITAE), to analyze its performance parameters values, and tp compare Hybrid-ITAE performance with conventional PID method. Working parameters being reviewed include overshoot, rise time, settling time, and ITAE. First of all, JGA25-370 DC motor was modeled in a form of a third order transfer function equation. Based on the transfer function, PID parameters were calculated using PID Output Feedback and ITAE polynomial methods. The best ITAE polynomial PID controllers were then be combined with a Fuzzy Logic Controller to form a Hybrid-ITAE system. Simulation and experimental stages were carried out in two conditions, namely no load and loaded. Simulation and experimental results showed that Hybrid-ITAE (*l* = 0.85) was the best controller for no-load simulation conditions. For loaded simulation Hybrid-ITAE (*l* = 1) was a better controller. In no-loads experiment, the best controller was Hybrid PID-Ziegler Nichols, while for loaded condition the best controller was Hybrid PID-Ziegler Nichols.**

***Keywords*—PID, Hybrid PID-Fuzzy, ITAE Polynomials, DC Motors.**

I. INTRODUCTION

DC motor is a type of electric motor widely applied as an actuator in various control system types, especially in fields of industry, automation, and robotics [1]. This is because the DC motor has a variety of speed controls that are adaptive and have high torque output.

DC motor speed can be controlled through various specific methods. A method commonly used in the industrial field is a PID control tuned by the Ziegler Nichols method [2]. The Ziegler Nichols method has good interference rejection, but it produces large overshoot and control signals that can cause saturation in the system [3]. In addition, the Ziegler Nichols method is not very adaptive to error changes. To overcome this adaptive problem, the concept of Fuzzy Logic Controller (FLC) was introduced. FLC is able to overcome complex and nonlinear systems. However, FLC has a slower response compared to PID control [4]. Thus, Hybrid PID-Fuzzy method is expected to overcome two aforementioned limitations, both the limitations of PID and FLC controls.

Based on these problems, this paper proposes the Hybrid PID-Fuzzy method as a control system to optimize process of controlling DC motor speed. This method is expected to produce rise time, minimum overshoot, and fairly good and stable output response in transient conditions. DC motor used in this paper is a JGA25-370 DC motor with a rated speed of 1,000 rpm. The motor is modeled in a form of a third order transfer function equation. PID Output Feedback method is used to obtain closed loop system matric formed by PID and DC motor transfer function. Characteristic equation from closed loop system matric was reviewd with Integrated of Time Absolute Error (ITAE) polynomial to determin PID parameter. In controller side, PID value was multiplied by output of fuzzy logic control. To find out this method's performance, its performance level was measured based on ITAE value, overshoot percentage, rise time, and settling time.

[1] *Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada, Jln. Grafika 2 Yogyakarta, INDONESIA (Tlp.: +62(274)552305; e-mail: hari.wibawa@mail.ugm.ac.id)*

[2,3] *Lecturer, Department of Electrical and Information Engineering, Faculty of Engineering, Universitas Gadjah Mada,, Jln. Grafika 2 Yogyakarta, INDONESIA (Tlp.: +62(274)552305; e-mail: oyas@ugm.ac.id, adha.imam@ugm.ac.id)*

II. THEORETICAL BACKGROUND

A. PID Controller

PID controller consists of three main components, namely proportional control, integral control, and derivative control. Proportional control, symbolized as *Kp*, is useful for reducing errors and accelerating time response. Integral control, symbolized as *Ki*, is used to eliminate steady state errors. Derivative control, symbolized as *Kd*, reduces overshoot and settling time. The PID controller is stated in (1) and (2).

$$u(t) = K_p.e(t) + K_p.\frac{1}{T_i}\int_0^t e(t)dt + K_p.T_d.\frac{de(t)}{dt}. \quad (1)$$

$$u(t) = K_p.e(t) + K_i\int_0^t e(t)dt + K_d.\frac{de(t)}{dt}. \quad (2)$$

Controlling signal variable is *u(t)*, and *e(t)* is error between reference value and output value signal, while *Kp*, *Ki*, dan *Kd* are PID parameters that must be determined accordingly to produce an optimal transient response. There are several methods that can be used to tune or to determine these three parameters, including Ziegler Nichols based PID and ITAE polynomial based PID.

B. Ziegler Nichols PID

Ziegler Nichols PID uses critical reinforcement parameter (K_{cr}) and critical period (P_{cr}) to calculate *Kp*, *Ti*, dan *Td* value. Critical reinforcement (K_{cr}) is a gain that causes system

 ISSN 2550-0554 (Online)

response to oscillate with a fixed periode, while critical period (P_{cr}) is period value of the oscillation signal [2].

In Ziegler Nichols method, the parameters *Ti* and *Td* are arranged so that the value of *Ti* = 0 and *Td* = ∞. Then, *Kp* value is enlarged from 0 to critical value K_{cr}, that is when output signal oscillates continuously. IfK_{cr} and P_{cr} values have been obtained, then PID parameters can be determined according to Table I.

TABLE I
ZIEGLER NICHOLS TUNING METHOD

Controller	K_p	T_i	T_d
P	$0.5K_{cr}$		
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

C. PID Output Feedback Method

An open loop system space state equation is expressed in (3) and (4).

$$\dot{x} = Ax + Bu \tag{3}$$

$$y = Cx. \tag{4}$$

Each size of state vector **x**, input vector **u**, and output vector **y** are $n \times 1$, $m \times 1$, $l \times 1$. Meanwhile, each size of matrix **A**, **B**, and **C** are $n{\times}n$, $n{\times}1$, and $1{\times}n$.

In PID Output Feedback Method, equation PID (2) is reduced by reference value then is fed back to (4). In this way, state equation of closed loop system space can be expressed in (5) and (6).

$$A_C = \begin{bmatrix} \dot{x}_l \\ \dot{x} \end{bmatrix} = \begin{bmatrix} 0 & C \\ (I + BK_dC)^{-1}(BK_i) & (I + BK_dC)^{-1}(A + BK_pC) \end{bmatrix} \tag{5}$$

$$C_C = y = \begin{bmatrix} 0 & C \end{bmatrix} \begin{bmatrix} \dot{x}_l \\ \dot{x} \end{bmatrix}. \tag{6}$$

Eigenvalue of closed loop characteristic equation is a system's pole that determines transient response of the system. If it is known that the closed loop matrix **H** is (*n*+1) x (*n*+1), eigenvalue is specified in (7).

$$|sI_{n+1} - H| = \prod_{i=1}^{n+l}(s - \lambda_i) \tag{7}$$

The **H** matrix corresponds to closed loop matrix equation (*X*) so $\mathbf{A_C} = \mathbf{H}$.

Reference [5] divides the **H** matrix into two parts, namely H1 (above) and H2 (bottom). The **H2** matrix contains a PID parameter.

PID parameters in this method are calculated using the PID Output Feedback [5] and ITAE polynomials [6] methods. Equation (8) states formulation of PID Output Feedback method.

$$BKM = V \tag{8}$$

with

$$K = [Ki \vdots\ Kp\ \vdots Kd] \tag{9}$$

$$M = \begin{bmatrix} \hat{I} \\ \hat{C} \\ CH_2 \end{bmatrix} \tag{10}$$

$$V = H_2 - \hat{A}. \tag{11}$$

Dimensions of *K*, *M*, and *V* are h *m*x3l, 3lx(*n*+l), and *n*x(*nl*), with *m* and *l* are numbers of *u* input and *y* output in a system. Matrix $\hat{I} = [\,I_m \vdots 0\,]$, $\hat{C} = [\,0 \vdots C\,]$ and $\hat{A} = [\,0 \vdots A\,]$, with each of their dimension are *m*x(*n*+*l*), *l*x(*n*x*l*) and *n*x(*n*+*l*). In SISO system, $m = l = 1$, and I_m is the identity matrix *m*x*m*, if the matrix is I_1, then $I_1 = 1$. So, for SISO system, $\hat{I} = [\,I_1 \vdots 0\,] = [1\ 0 \cdots 0 \vdots 0]$ and the dimension is 1*x*(*n*+1).

Equation (8) determines equation *Kp*, *Ki*, and *Kd*, while characteristic equation of **H** matrix is adjusted to ITAE polynomial.

D. ITAE Polynomials

ITAE is a working index showing accumulated errors when transient conditions are approaching a steady state. ITAE value increases as it approaches steady state [7]. ITAE is mathematically formulated in (12).

$$ITAE = \int_0^\infty t|e(t)|dt. \tag{12}$$

The ITAE parameter can be used to minimize overshoot and dampen oscillation. This parameter application in equation of servomotor system transfer function characteristics has been investigated [8]. In the study, formulated ITAE polynomials could be applied to minimize overshoot in the servomotor system. In general, the ITAE equation can be written in (13).

$$\begin{aligned} &s^{n+1} + (C_1 + \xi)(l\omega_n)s^n + (C_2 + \xi c_1)(l\omega_n)^2 s^{n-1} \\ &+(C_3 + \xi c_2)(l\omega_n)^3 s^{n-2} + \cdots + (1 \\ &\qquad + \xi C_{n-1})(l\omega_n)^n s + \xi (l\omega_n)^{n+1}. \end{aligned} \tag{13}$$

Equation (13) is known as a binomial equation with ω_n as an undamped natural frequency. Coefficients of C_1, C_2, …, *Cn* at (13) can be written in Table II.

TABLE II
COEFFICIENT OF ITAE POLYNOMIALS EQUATION

Order	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8
1	1							
2	0.4							
3	1.75	2.15						
4	2.1	3.4	2.7					
5	2.8	5	5.5	3.4				
6	3.25	6.6	8.6	7.45	3.95			
7	4.475	10.42	15.08	15.54	10.64	4.58		
8	5.2	12.8	21.6	25.75	22.2	13.3	5.15	

Value of ω_n is assumed to be equivalent to ω_n in the open loop system, so applies that

$$a_n = {\omega_n}^n \tag{14}$$

Value of a_n is a coefficient of characteristic equation in system transfer function.

E. Fuzzy Control

FLC system is a controlling system based on a set of rules representing designer's general knowledge of how controlling process can be applied. These rules are expressed in form of IF-THEN logic rules. Therefore, FLC is also called as a knowledge-based system.

The FLC system is divided into four important parts, namely Fuzzifizier, Fuzzy Rule Base, Fuzzy Inference Engine, and Defuzzifier. In an FLC, fuzzifiers are tasked with converting input values in the real number domain (crisp) to fuzzy set domains which can be interpreted by the Fuzzy Inference Engine. The conversion process takes membership function for each input. After that, Fuzzy Inference Engine uses a set of rules in Rule Base to carry out decision making and conclusions. If conclusions have been taken and are intended to be applied in the system, then conclusions are converted by Defuzzifier to real estate domain.

F. Performance Parameters

Performance parameters from system response include the following things [2], [9].

- Delay time (*td*) refers to the time required by sytem response to reach half or first peak value or half or peak time.
- Peak time (*tp*) is amount of time to reach peak value of first overshoot.
- Rise time (*tr*) is time from 10% value to 90% final value.
- Overshoot (%*OS*) is percentage of systems response wave exceeding steady state value or its final value.
- Settling time (*ts*) is time to reach steady state and remain to be around steady state value with error tolerance of 2% or 5%.

III. Method

This paper uses a JGA25-370 DC motor equipped with an encoder, Arduino Uno microcontroller, L-298N H-bridge driver module, three 3.7 volt batteries, Arduino IDE software, MATLAB 2016b software, and an artificial PLX-DAQ device by Parralax Inc. The research flow chart is shown in Fig. 1.

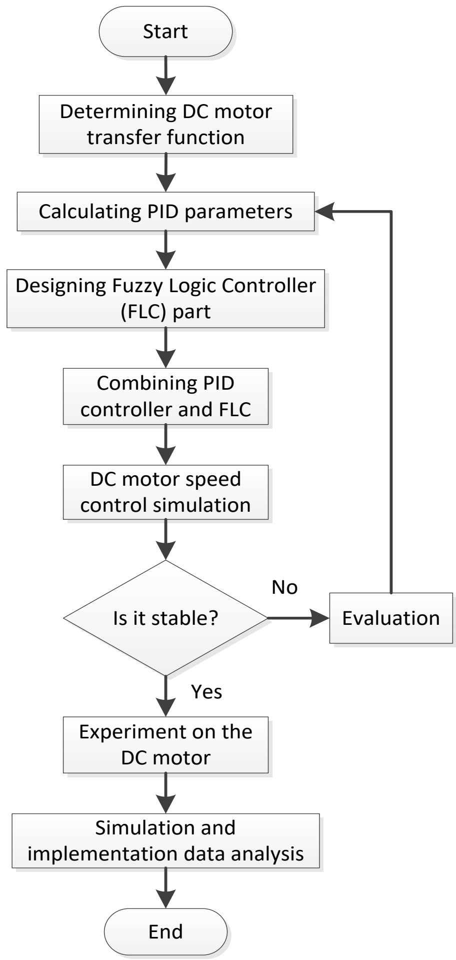

Fig. 1 Research flow chart.

A. Determination of DC Motor Transfer Function

Equation of DC motor transfer function was obtained through a process of input and output data pairs modeling using Systems Identification Toolbox in MATLAB 2016b software. Reference speed or input speed was set at 500 rps, while output speed data was recorded through PLX-DAQ software. The output and input data pairs was then as input model for System Identification Toolbox.

B. Calculating Kp, Ki, and Kd Parameters

PID parameters were calculated using PID Output Feedback and ITAE Polynomials methods. In addition, PID parameters were also calculated using Ziegler Nichols method.

Equation of three-order transfer function of open loop system is generally stated in (15).

$$G_p(s) = \frac{Y(s)}{U(s)} = \frac{k(b_0 s^3 + b_1 s^2 + b_2 s + b_1)}{s^3 + a_1 s^2 + a_2 s + a_3} \quad (15)$$

Transfer function equation (14) can be stated in a form of state space as follows.

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -a_3 & -a_2 & -a_1 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 0 \\ k \end{bmatrix}; \quad C = [\, c_3 \quad c_2 \quad c_1 \,].$$

Matrices of $\hat{I}$, $\hat{A}$, $\hat{C}$, H, H_2 and CH_2 can be written as follows.

$$\hat{I} = [\, I_3 \,\vdots\, 0 \,] = [1\ 0\ 0\ 0]$$

$$\hat{C} = [\, 0 \,\vdots\, C \,] = [0 \quad c_3 \quad c_2 \quad c_1]$$

$$\hat{A} = [0 \,\vdots\, A] = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -a_3 & -a_2 & -a_1 \end{bmatrix}$$

$$H = \begin{bmatrix} H_1 \\ \cdots \\ H_2 \end{bmatrix} = \begin{bmatrix} 0 & c_3 & c_2 & c_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -h_4 & -h_3 & -h_2 & -h_1 \end{bmatrix}$$

$$H_2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -h_4 & -h_3 & -h_2 & -h_1 \end{bmatrix}$$

$$CH_2 = [\, c_3 \quad c_2 \quad c_1] \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -h_4 & -h_3 & -h_2 & -h_1 \end{bmatrix}.$$

Matrices of *M*, *V*, *BKM* in form of (9), (10), (11) are

$$M = \begin{bmatrix} \widehat{I} \\ \widehat{C} \\ CH_2 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_3 & c_2 & c_1 \\ -c_1h_4 & -c_1h_3 & (c_3 - c_1h_2) & (c_2 - c_1h_1) \end{bmatrix}$$

$$V = H_2 - \widehat{A} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -h_4 & (a_3 - h_3) & (a_2 - h_2) & (a_1 - h_1) \end{bmatrix}$$

$$\mathrm{BKM} =$$

$$\begin{bmatrix} 0 \\ 0 \\ k \end{bmatrix} [\, K_i \;\; K_p \;\; K_d \,] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_3 & c_2 & c_1 \\ -c_1h_4 & -c_1h_3 & (c_3 - c_1h_2) & (c_2 - c_1h_1) \end{bmatrix}$$

$$\mathrm{BKM} =$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ kK_i - kK_dc_1h_4 & kK_pc_3 - kK_dc_1h_3 & kK_pc_2 - kK_d(c_3 - c_1h_2) & kK_pc_1 - kK_d(c_2 - c_1h_1) \end{bmatrix}.$$

By adjusting right and left segments of *BKM* = *V* equation, the following four equations are obtained.

$$kK_i - kK_dc_1h_4 = -h_4$$

$$kK_pc_3 - kK_dc_1h_3 = a_3 - h_3$$

$$kK_pc_2 - kK_d(c_3 - c_1h_2) = a_2 - h_2$$

$$kK_pc_1 - kK_d(c_2 - c_1h_1) = a_1 - h_1.$$

Each parameter can then be specified in (16).

$$K_p = \frac{(a_2 - h_2)c_1h_3 + (h_3 - a_3)(c_3 - c_1h_2)}{k(c_2c_1h_3 - c_3(c_3 - c_1h_2))}$$

$$K_i = \frac{(kK_dc_1h_4 - h_4)}{k} \tag{16}$$

$$K_d = \frac{kK_pc_2 + h_2 - a_2}{k(c_3 - c_1h_2)}.$$

Therefore, state space form of (6) can be written as follows.

$$A = \begin{bmatrix} 0 & 1 & 1 \\ 0 & 0 & 0 \\ -18.86 & -15.44 & -4.805 \end{bmatrix} ; \; B = \begin{bmatrix} 0 \\ 0 \\ 19.25 \end{bmatrix};$$

$$C = [\, 1 \;\; 0 \;\; 0 \,].$$

With $c_1 = c_2 = 0$ and $c_3 = 1$, PID parameter can be rewritten in (17).

$$K_p = \left|\frac{(h_3 - a_3)}{k}\right|$$

$$K_i = \left|\frac{(-h_4)}{k}\right| \tag{17}$$

$$K_d = \left|\frac{h_2 - a_2}{k}\right|.$$

Characteristic quations of closed loop order system matrix and three-order system ITAE polynomials based characteristic equation can be rewritten as follows.

$$s^4 + h_1s^3 + (h_2 + c_1h_4)s^2 + (h_3 + c_2h_4)s + c_3h_4. \tag{18}$$

ITAE Polynomial Attenuation ratio of 0,7:

$$s^4 + 2.45(l\omega_n)s^3 + 3.375(l\omega_n)^2s^2 + 2.505(l\omega_n)^3s + 0.7(l\omega_n)^4. \tag{19}$$

ITAE Polynomial Attenuation ratio of 0,9:

$$s^4 + 2.65(l\omega_n)s^3 + 3.725(l\omega_n)^2s^2 + 2.935(l\omega_n)^3s + 0.9(l\omega_n)^4. \tag{20}$$

By adjusting (18) and (19), value of *h* parameter with attenuation ratio of 0.7 can be obtained as follows.

$$h_1 = 2.45(l\omega_n)$$

$$h_2 = 3.375(l\omega_n)^2$$

$$h_3 = 2.505(l\omega_n)^3$$

$$h_4 = 0.7(l\omega_n)^4.$$

Referring to (14), ω_n was obtained.

$$\omega_n = \sqrt[3]{a_3} = \sqrt[3]{18.86}.$$

C. Designing Fuzzy Logic Controller (FLC) Parts

1) Fuzzification: Fuzzification procces was administered using two outputs, namely error $e(t)$ and error changes $\Delta e(t)$. These two inputs are expressed in two linguistic variables and both are expressed with a same membership function as shown in Fig. 2. The membership function has an interval of -1 to 1 and has three linguistic values, i.e., Negative (N), Zero (Z), and Positive (P).

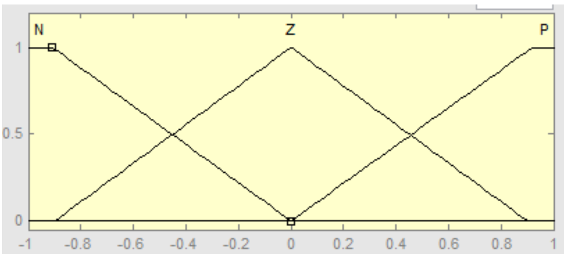

Fig. 2 Membership function for error $e(t)$ and error changes $\Delta e(t)$.

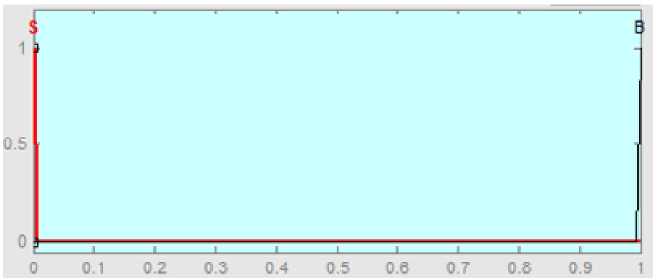

Fig. 3 Membership function for *h* variable.

2) Fuzzy Rule Base and Fuzzy Inference Engine: FLC system output is an output variable symbolized as *h*. This variable is defined in two linguistic variables, namely Small (S) and Big (B) in intervals of 0.1 to 1. Unlike input variable which has a triangular membership function, membership function of h variable has a singleton form. The h variable membership function form is shown in Fig. 3, while rule base rules for h variable is written in Table III.

TABLE III
RULE BASE FOR *H* OUTPUT VARIABLE

Error Changes	**Error**		
	N	***Z***	***P***
N	S	B	S
Z	S	B	S
P	S	B	S

N = Negative; *Z* = Zero; *P* = Positive; *S* = Small; *B* = Big

Based on rules in Table II, new PID parameters are formulated in (21).

$$Kp = h.K_{pm};\ Ki = h^2.K_{im};\ Kd = Kd. \quad (21)$$

Fuzzy Inference Engine is a FLC component utilizing fuzzy logic principles to combine several IF-THEN rules on the rule base. Types of interference engine in this paper is Minimize Inference Engine. This type applies individual rule base inference method by combining IF-THEN rules through union logic operators, implications based on Mamdani minimum operations, logic min for all t-norm operators, and max logic for all s-norm operators.

3) Defuzzification: Defuzzifier is a part of FLC which functions to map fuzzy outputs sets to a set of real numbers. These defuzzification outputs become multiplier factors for PID parameters so the values can be updated according to error and error changes at that time.

The utilized defuzzigier is center average defuzzification. Center average defuzzification method is mathematically stated in (22).

$$y^* = \frac{\sum_{l=1}^{M} y^{-1} w_l}{\sum_{l=1}^{M} w_l}. \quad (22)$$

The center average defuzzification method is graphically stated in Fig. 4.

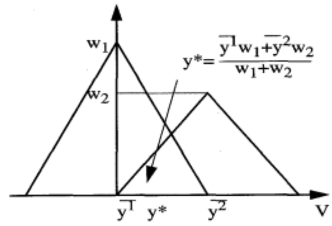

Fig. 4 Center average defuzzification.

D. Combining PID and FLC

At this stage, PID and FLC controllers were combined to form PID-Fuzzy Hybrid system. Combined PID controller with FLC was a conventional PID controller, either Ziegler Nichols method based PID or ITAE polynomials based PID which had optimal performance.

E. DC Motor Speed Control Simulation

Hybrid PID-Fuzzy control system simulation was carried out using MATLAB-Simulink. Diagram block of Simulink simulation is shown in Fig. 5. Simulation results were reviewed from system responses and parameters values of overshoot, rise

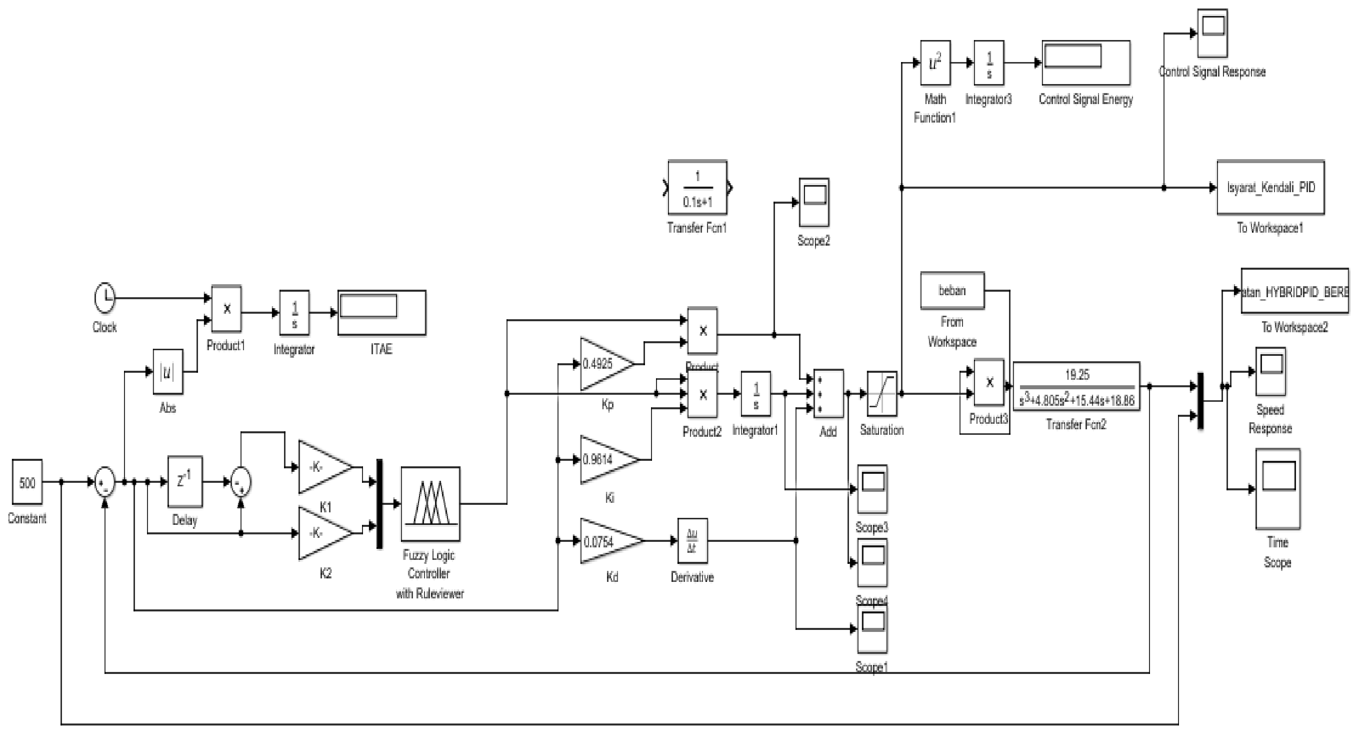

Fig. 5 Simulink simulation block diagram.

time, settling time, and ITAE. Simulation stages were carried out in two conditions, i.e., without load and with load.

F. DC Motor Speed Control Experiment

In experimental stages, designed algorithms and programs were uploaded to Arduino Uno board to apply and test DC motor speed speed control mechanism. The observed parameters were similar to simulation process, i.e., overshoot, settling time, rise time, and ITAE values. Experimental stages were carried out without load and with load. The utilized load was a permanent magnet attached to DC motor shaft disc.

G. Analysis of Simulation and Experiment Data

Analysis process compared performance data from Hybrid PID-Fuzzy controller, ITAE polynomials based OID, and Ziegler Nichols method based PID, both in simulation and experimental stages.

IV. Simulations and Experiments

A. Determination of DC Motor Transfer Function

Based on consideration of FPE and MSE values as well as ease of control system implementation, therefore third order transfer function equation was selected as DC motor system model. Transfer function equation is stated in (23).

$$G_p(s) = \frac{19{,}25}{s^3+4.805s^2+15.44\,s+18.86}. \tag{23}$$

B. PID Parameters Calculation

Selected l variable values were of 1, 0.95, 0.90, 0.85, and 0.80. Then h parameter was substituted into (8) so that PID parameter value for each l value was obtained. Parameter values of ITAE polynomials based PID with attenuation ratio of 0.7 are written in Table IV.

TABLE IV
PARAMATER VALUES OF ITAE POLYNOMIALS BASED PID

l	K_p	K_i	K_d
1	3.4340	1.8255	0.4402
0.95	3.0840	1.4869	0.3190
0.9	2.7689	1.1977	0.2041
0.85	2.4870	0.9529	0.0954
0.8	2.2363	0.7477	-0.0070

On the other hand, ITAE polynomials based PID parameters with attenuation value of 0.9 are written in Table V. Attenuation ratio value was carried out because simulation result showed that systems response oscillated when Simulink saturation block value was altered from 500 rps to 1,500 rps.

TABLE V
PARAMATER VALUES OF ITAE POLYNOMIALS BASED PID

l	K_p	K_i	K_d
1	1.8958	2.3471	0.5690
0.95	1.4857	1.9117	0.4353
0.9	1.1165	1.5399	0.3085
0.85	0.7862	1.2252	0.1885
0.8	0.4925	0.9614	0.0754

PID parameters according to Ziegler Nichols method are stated in Table VI. The PID value was obtained from K_{cr} critical reinforcement value of 2.87 and P_{cr} critical period of 1.5988 seconds.

TABLE VI
ZIEGLER NICHOLS PID PARAMATER VALUES

Controller	K_p	K_i	K_d
PID-ZN	1.7220	1.3766	0.3441

C. Calculation of Performance Parameters Comparison

After performance parameters data were collected for PID-ZN, PID-ITAE, and Hybrid-ITAE controllers, performance parameters were compared with performance comparison values formulated in (24) and (25).

$$H(i) = \frac{\%OS}{Max(\%OS)} + 0.1 \times \frac{Tr}{Max(Tr)} + \frac{Ts}{Max(Ts)} + \frac{ITAE}{Max(ITAE)} \tag{24}$$

$$N(i) = \frac{10\times(Max(H(i))-H(i))}{Max(H(i))-Min(H(i))} \tag{25}$$

with $H(i)$ is parameters comparison, while $N(i)$ is performance comparison of the-i controller with i expressing order of controller type in a table. Tr is rise time, Ts is settling time, and $\%OS$ is overshoot percentage.

D. Simulation in Without Load Condition

Data on performance parameters without load simulation results are shown in Table VII. Fig. 6 shows a graph of speed simulation results for PID-ZN and PID-ITAE controllers.

TABLE VII
RESULTS OF DC MOTOR SPEED PERFORMANCE PARAMETER SIMULATION IN NO-LOAD CONDITION

Controller	$\%OS$	Rise Time	Ts	ITAE
PID-ZN	7.80	0.56	3.92	370.70
PID-ITAE (l = 1)	15.59	0.51	2.98	247.70
PID-ITAE (l = 0.95)	11.21	0.58	3.20	230.90
PID-ITAE (l = 0.90)	6.33	0.69	3.42	225.20
PID-ITAE (l = 0.85)	2.04	0.85	3.44	247.00
PID-ITAE (l = 0.80)	0.75	1.12	3.07	354.60
HYBRID-ITAE (l = 0.85)	1.32	1.00	2.75	303.90
HYBRID-ITAE (l = 0.80)	0.23	1.46	3.26	454.00

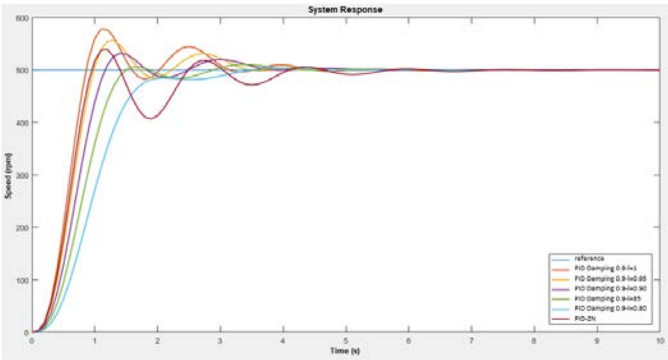

Fig. 6 Graph of DC motor speed control simulation results for PID-ZN and PID-ITAE controllers, without load condition.

Based on Table VII, PID-ITAE (l = 0.85) and PID-ITAE (l = 0.80) are two controllers with the best performance compared to other controllers. Those two PID controllers were then combined with FLC to form Hybrid PID-Fuzzy system or simply be denoted as Hybrid ITAE.

Data of Hybrid-ITAE system simulation results are shown in the last two lines in Table VII. In addition, this simulation results are also showed in a form of a graph in Fig. 7.

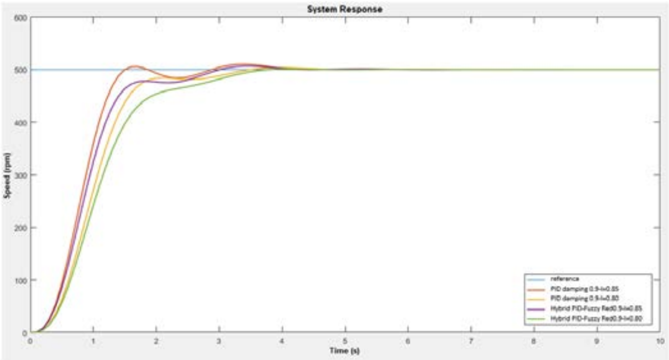

Fig. 7 Graph of DC motor speed control simulation results for PID-ITAE (*l* = 1), PID-ITAE (*l* = 0.95), Hybrid PID-ITAE (*l* = 0.85), and Hybrid PID-ITAE (*l* = 0.80), controllers without load condition.

Simulation data shows that FLC application on PID-ITAE (*l* = 0.85) reduces the %*OS* value to 1.32. In addition, the system stabilized faster because the settling time was reached at 2.75 seconds. On the other hand, ITAE parameters actually grew larger, i.e., 303.90. This shows that Hybrid-ITAE system (*l* = 0.85) has an error in higher steady state conditions than its PID version. Nevertheless, Hybrid-ITAE (*l* = 0.85) can be said to have a better performance than PID version. Hybrid-ITAE system simulation (*l* = 0.80) tends to produce worse performance. Even though the overshoot value is reduced to 0.23 seconds, the settling time and ITAE parameters actually increase. This shows that Hybrid ITAE system (*l* = 0.80) has a longer steady state and has a greater error in steady state. This condition is caused by FLC output which tends to increase *Kp* and *Ki* values. When these two parameters increase, the overshoot becomes smaller, but the settling time gets greater.

Based on that simulation results consideration and calculations in (24) and (25), Hybrid-ITAE (*l* = 0.85) can be considered as a controller with the best performance for simulation without load with *N(i)* value of 10.00, followed by PID-ITAE (*l* = 0.80) with *N(i)* of 8.68, PID-ITAE (*l* = 0.85) with *N(i)* of 8.00, and Hybrid-ITAE (*l* = 0.80) with *N(i)* of 7.12.

E. Simulation of Loaded Condition

Data of loaded condition simulation results are shown in Table VIII. Simulation results graph for PID-ZN and PID-ITAE controllers are shown in Fig. 8. In this simulation, the load was given from 0 to 5 seconds. The load was then released at 5.1 seconds until the simulation was completed in 10 seconds.

Table VIII shows that PID-ITAE (*l* = 1) and PID-ITAE (*l* = 0.95) are the best PID controllers for loaded condition simulation. These two controllers were combined with FLC to form Hybrid-ITAE system.

Data of Hybrid-ITAE system simulation results are shown in the last two lines in Table VIII. In addition, this simulation results are also showed in a form of a graph in Fig. 9.

Under load conditions, Hybrid-ITAE (*l* = 1) has a %*OS* value of 9.16. This value is smaller than the PID version, which is 9.51. This shows that FLC combination has reduced the %*OS*. In addition, the ITAE value was also reduced to 546.20. This indicates a smaller error in steady state. On the other hand, the settling time and rise time parameters tend to increase slightly. When compared with Hybrid-ITAE controllers (*l* = 0.95), Hybrid-ITAE control system (*l* = 1) has better performance.

TABLE VIII
SIMULATION RESULTS OF DC MOTOR SPEED PERFORMANCE PARAMATER SIMULATION IN LOADED CONDITIONS

Controller	%OS	Rise Time	*Ts*	ITAE
PID-ZN	10.18	0.68	8.15	908.00
PID-ITAE (*l* = 1)	9.51	0.58	7.45	554.30
PID-ITAE (*l* = 0.95)	10.46	0.69	7.56	627.50
PID-ITAE (*l* = 0.90)	12.19	0.84	7.39	763.70
PID-ITAE (*l* = 0.85)	14.42	1.14	7.61	1,003
PID-ITAE (*l* = 0.80)	17.10	1.90	7.79	1,408
HYBRID-ITAE (*l* = 0.85)	9.16	0.65	7.46	546.20
HYBRID-ITAE (*l* = 0.80)	10.48	0.79	7.61	649.20

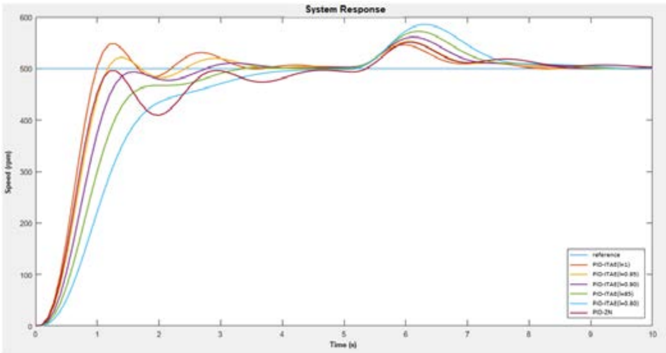

Fig. 8 Graph of DC motor speed control simulation results for PID-ZN and PID-ITAE controllers, in loaded conditions.

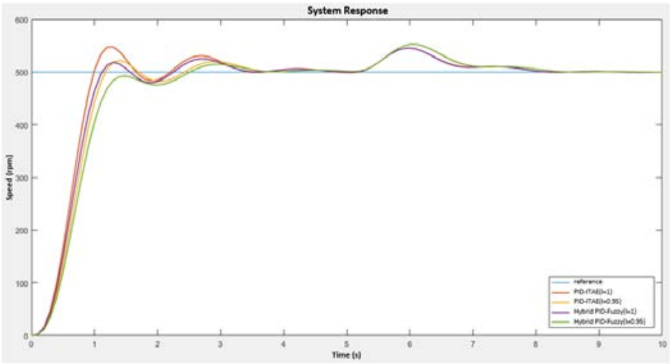

Fig. 9 Simulation results graph of DC motor speed control for PID-ITAE controllers (*l* = 1), PID-ITAE (*l* = 0.95), Hybrid PID-ITAE (*l* = 1), and Hybrid PID-ITAE (*l* = 0.95), in loaded conditions.

Based on these considerations and calculations on (24) and (25), Hybrid-ITAE controllers (*l* = 1) can be considered as the best controllers for loaded conditions with a value of *N(i)* of 10.00, followed by PID-ITAE (*l* = 1) with *N(i)* of 9.82, PID-ITAE (*l* = 0.95) with *N(i)* of 8.75, and *Hybrid*-ITAE (*l* = 0.95) controller with *N(i)* of 8.51.

F. Experiment Without Conditions

Experiment Without Conditions results data is shown in Table IX. Graph of experiment results for PID-ZN and PID-ITAE controllers are shown in Fig. 10.

TABLE IX
EXPERIMENT RESULTS OF DC MOTOR SPEED PERFORMANCE PARAMETERS UNDER NO-LOAD CONDITIONS

Controller	%OS	*Rise Time*	*Ts*	ITAE
PID-ZN	3.70	0.17	12.91	284.60
PID-ITAE (l = 1)	3.70	0.17	12.86	329.20
PID-ITAE (l = 0.95)	5.19	0.17	>13.00	317.80
PID-ITAE (l = 0.90)	3.70	0.17	>13.00	303.80
PID-ITAE (l = 0.85)	3.70	0.17	12.92	299.20
PID-ITAE (l = 0.80)	3.70	0.18	12.82	334.30
HYBRID PID-ZN	3.08	0.17	12.17	330.40
HYBRID-ITAE (l = 0.85)	3.08	0.14	12.07	546.20
HYBRID-ITAE (l = 0.80)	3.08	0.14	>13	649.20

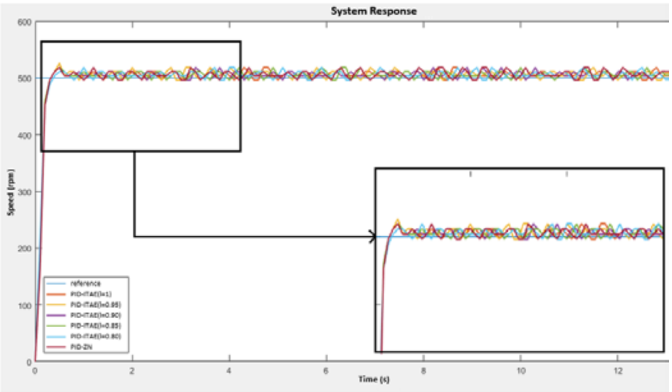

Fig. 10 Experiment results graph of DC motor speed control for PID-ZN and PID-ITAE controllers, under no load conditions.

Data of Hybrid-ITAE system simulation results are shown in the last two lines in Table IX. In addition, this experiment results are also shown in a form of graph in Fig. 11.

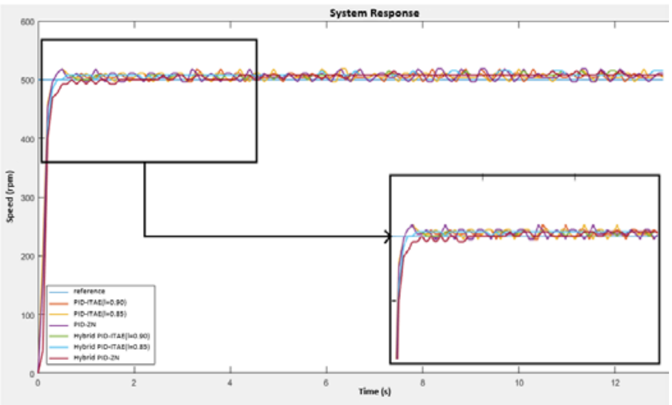

Fig. 11 Graph of DC motor speed control experiment results for PID-ZN, PID-ITAE (l = 0.90), PID-ITAE (l = 0.85), Hybrid PID-ZN, Hybrid PID-ITAE (l = 0.90), and Hybrid PID-ITAE (l = 0.85) controllers, under no load condition.

Test results show that those three Hybrid-ITAE systems have same overshoot, which is 3.08. These three overshoot values are smaller than PID version. This shows that Hybrid-ITAE system improves overshoot value for those three controllers. In Hybrid PID-ZN system, rise time and settling time values also decrease. Values of each parameter are of 0.17 seconds and 12.17 seconds. Combination with FLC makes PID-ZN controller system reach its final value and stable condition faster. On the other hand, value of ITAE Hybrid PID-ZN system is actually greater than the PID version, which is 330.40. Several performance Parameters improvement effects can also be seen in Hybrid-ITAE (l = 0.90). Overshoot value decreased to 3.08, rise time got faster which was equal to 0.14 seconds, while settling time value was measured at 12.07 seconds. It showed that there was a settling time improvement compared to the PID verson because the previous settling time parameter value could not be measured. Although the previous three parameters had improved, ITAE parameter actually grew larger, i.e., to 379.10. Combination result of FLC with PID-ITAE (l = 0.85) tend to add value to ITAE and settling time. The improved performance parameters were only overshoot and rise time. When compared its PID version, the PID looked predominant.

Based on experiment results discussion and calculation on (24) and (25), Hybrid PID-ZN can be considered as a relatively better controller system compard to other controller for no load cases with *N(i)* of 10.00, followed by PID-ZN with *N(i)* of 8.32, and Hybrid-ITAE (l = 0.90) with *N(i)* of 8.02.

G. *Experiments of Loaded Conditions*

Experiment results data on loaded conditions are shown in Table X. Graph of experiment results for PID-ZN and PID-ITAE controllers are shown in Fig. 12. In this simulation, load was given for 9 seconds, then it was released to trigger interference. Measurement ended at 13 seconds.

TABLE X
EXPERIMENT RESULTS OF DC MOTOR SPEED PERFORMANCE PARAMATER ON LOADED CONDITIONS

Controller	%OS	*Rise Time*	*Ts*	ITAE
PID-ZN	4.54	0.16	>13.00	484.00
PID-ITAE (l = 1)	4.54	0.17	12.97	475.20
PID-ITAE (l = 0.95)	4.54	0.17	12.87	421.50
PID-ITAE (l = 0.90)	4.54	0.17	12.97	328.90
PID-ITAE (l = 0.85)	3.07	0.17	>13.00	393.10
PID-ITAE (l = 0.80)	3.07	0.17	12.77	283.70
HYBRID PID-ZN	1.54	0.16	0.39	92.59
HYBRID-ITAE (l = 0.85)	1.54	0.17	0.77	168.40
HYBRID-ITAE (l = 0.80)	1.54	0.16	0.39	92.59

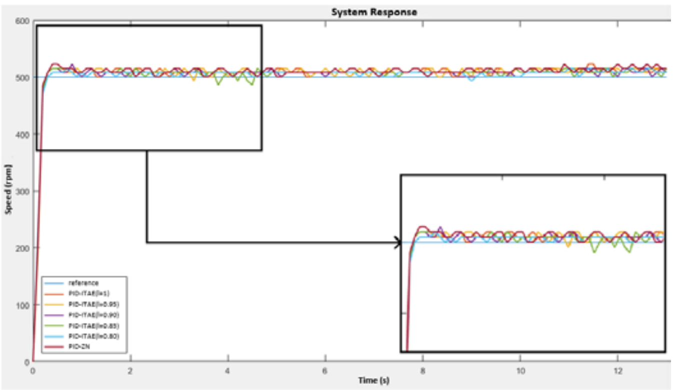

Fig. 12 Graph of DC motor speed control experiment results for PID-ZN and PID-ITAE controllers, under loaded condition.

Data of Hybrid-ITAE system simulation results are shown in the last three lines in Table X. In addition, this experiment results are also shown in a form of graph in Fig. 13.

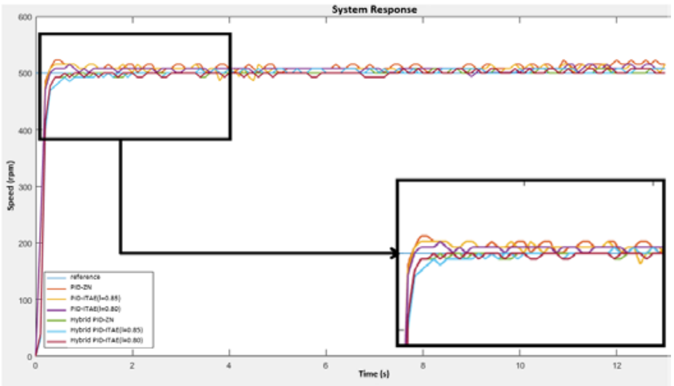

Fig. 13 Graph of DC motor speed control experiment results for PID-ZN, PID-ITAE (l = 0.85), PID-ITAE (l = 0.80), Hybrid PID-ZN, Hybrid PID-ITAE (l = 0.85), and Hybrid PID-ITAE (l = 0.80), controllers under loaded conditions.

Hybrid-ITAE system experiment results showed that overshoot value was reduced to 1.54, both in Hybrid-ITAE PID-ZN, Hybrid-ITAE (l = 0.85), and Hybrid-ITAE (l = 0.80). In addition, Hybrid-ITAE controllers (l = 0.80) experienced a rise time improvement. The controller was faster to reach final value because it had a rise time of 0.16 seconds. Whereas Hybrid-ITAE (l = 0.85) had the same rise time as the PID version. In addition, settling time for the three controllers had also been improved. Hybrid PID-ZN settling time value was 0.39 seconds. This value is the same as Hybrid-ITAE settling time value (l = 0.80) and was faster than Hybrid-ITAE settling time value (l = 0.85). Hybrid PID-ZN controller also had a much smaller ITAE value compared to Hybrid-ITAE (l = 0.80) and Hybrid-ITAE (l = 0.85), which was equal to 92.59. This shows that Hybrid PID-ZN has errors in its steady state conditions, which are smaller than Hybrid-ITAE (l = 0.80) and Hybrid-ITAE (l = 0.85). Compared to the PID version, these two Hybrid-ITAE systems have a far better performance. Based on (24) and (25), Hybrid PID-ZN controllers have *N(i)* of 10.00 and Hybrid-ITAE controllers (l = 0.85) have *N(i)* of 9.22. Whereas Hybrid-ITAE controllers (l = 0.80) have *N(i)* of 9.72.

Based on the discussion, the best performing controller for experiments with load conditions was Hybrid PID-ZN, followed by Hybrid-ITAE controllers (l = 0.80) and Hybrid-ITAE (l = 0.85).

V. Conclusion

Results of simulation and experiments show that Hybrid PID-Fuzzy has a predominant performance compared to Ziegler Nichols method based PID and ITAE polynomials based PID, both for no-load and with load condition.

Referensi

[1] C.T. Kilian, *Modern Control Technology: Components And Systems*, 2nd ed. Singapore: Delmar Publisher, 2006.

[2] K. Ogata, *Modern Control Engineering*, 5th ed. Upper Saddle River, USA: Prentice Hall, 2010.

[3] J.C. Basilio and S.R. Matos, “Design of PI and PID Controllers with Transient Performance Specification,” *IEEE Transactions on Education*, Vol. 45, No. 4, pp. 364–370, 2002.

[4] L. Ljung, *System Identification Toolbox User’s Guide R2015, How to Contact MathWorks*. US: The MathWork, Inc., 2015.

[5] P.N. Paraskevopoulos, “On the Design of PID Output Feedback Controllers for Linear Multivariable Systems,” *IEEE Transactions on Industrial Electronics and Control Instrumentation*, Vol. IECI-27, No. 1, pp. 16–18, 1980.

[6] A.A. Khan and N. Rapal, “Fuzzy PID Controller: Design, Tuning and Comparison with Conventional PID Controller,” *2006 IEEE International Conference on Engineering of Intelligent Systems*, 2006, No. 1, pp. 1–6.

[7] H. Christanto, “Optimasi Kendali Kecepatan Motor DC pada Keadaan Transien Menggunakan Pengendali PID Berbasis Polinomial ITAE”, Thesis, Universitas Gadjah Mada, Yogyakarta, Indonesia, 2014.

[8] D. Graham and R.C. Lathrop, “The Synthesis of Optimum Transient Response: Criteria and Standard Forms,” *Transactions of the American Institute of Electrical Engineers, Part II: Applications and Industry*, Vol. 72, No. 5, pp. 273–288, 1953.

[9] N.S. Nise, *Control Systems Engineering*, 7th ed. Hoboken, USA: Wiley, 2015.