

## GUI Based Control System Analysis Using PID Controller for Education

Ashwaq Abdulameer\*, Marizan Sulaiman, M.S.M. Aras, Dawood Saleem

Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka (UTeM),  
Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.

\*Corresponding author, e-mail: shahrieel@utem.edu.my

### Abstract

PID control strategy should be understood as a huge part in the education oriented on process control. Application of suitable GUI windows software can contribute in the increase of education quality and providing a better understanding of PID control through as it provides a user friendly environment which is suitable and comfortable for teaching, learning and training application. This paper present the PID control system analysis by explaining the PID controller three-term parameters, PID control types and structure, and PID tuning approach using Ziegler-Nichols and manual tuning method (in both s-domain and z-domain) with the help of simulation and Graphical User Interface GUI windows based on MATLAB. This software package is targeted for engineering students and practicing engineers.

**Keywords:** control education, PID parameters tuning, GUI/MATLAB, ZN method, discrete PID.

Copyright © 2016 Institute of Advanced Engineering and Science. All rights reserved.

### 1. Introduction

The Proportional Integral Derivative (PID) controller is a three term controller and one of the earlier control strategies introduced in the beginning of the previous century. PID controller has become a standard controller in industrial settings because of it is flexibility, relative simplicity, and satisfactory performance. The aim of using feedback control system with PID controller is to drive a process or plant by keeping the output response of the system as closes as possible to the desired response. It is typically used to control variables like temperature in thermal systems, position or velocity in mechanical systems, and voltage, current or frequency in electrical systems [1, 2]. Figure 1 shows the block diagram of the feedback control system. There are several methods for tuning the PID controller parameters to obtain satisfactory control system response. Ziegler-Nichols tuning method is the most popular method among all controller tuning methods. Ziegler-Nichols tuning for the PID controller algorithm involves calculation of the three individual constant parameters [3]. The transfer function of the PID controller (in s-domain) is generally written in parallel form as in Equation (1) [4] or in time constant form as given by Equation (2) [5].

$$C(s) = K_p + K_i \frac{1}{s} + K_d s \quad (1)$$

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (2)$$

Where  $K_p$  is the proportional gain,  $K_i$  is the integral gain ( $K_i=K_p/T_i$ ),  $K_d$  is the derivative gain ( $K_d=K_p*T_d$ ),  $T_i$  is integral time constant and  $T_d$  is the derivative time constant [6].

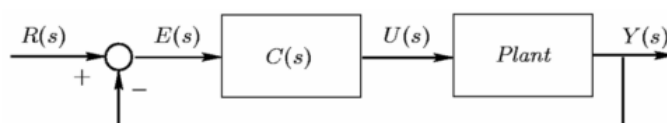


Figure 1. Feedback Control System

Technology development in the past few decades has led to replacement of analog controllers by digital controllers whose inputs and outputs are defined at discrete time instance [7]. A digital control system uses digital hardware as the heart of the controller, usually in the form of a programmed digital computer. The digital controllers are either in the form of digital circuits, digital computers, or microprocessors [8].

There are many benefits of using digital controller such as better reliability, flexibility, and expandability, as well as potentially lower prices [9]. The most common procedure for calculating a discrete PID controller starts by designing an analog PID controller and then applying Euler's transformation to get the discrete PID controller [10, 11]. Euler's method (backward difference) is applied to implement the PID controller in z-domain for numerical integration, where  $\frac{z-1}{Tz}$ , which gives the discrete transfer function of the PID controller as in Equation 3 or Equation 4 [12].

$$C(z) = K_p + \frac{K_i T z}{z-1} + \frac{K_d(z-1)}{Tz} \quad (3)$$

$$C(z) = \frac{g_0 z^2 + g_1 z + g_2}{Tz(z-1)} \quad (4)$$

Where,

$$g_0 = K_p T + K_i T^2 + K_d$$

$$g_1 = -K_p T - 2K_d$$

$$g_2 = K_d$$

$T$  = sampling time

There are many algorithms for numerical integration and numerical differentiation, hence, Equation (3) is not unique for a digital PID controller. Nevertheless, Equation (3) is the most preferred transfer function for many commercial PID controllers available. By using this equation, it is easy to obtain a P or PI controller by setting an appropriate gain equal to zero [13]. In general, PID controller parameters tuning can be very difficult, inaccurate and take a lot of time when applying for the high order system. Also, the digitalization of such controlled system can be complicated. Because of that, GUI windows can be used for simplifying the tuning of the PID controller and the digitalization of the controlled system with less time and more accuracy.

## 2. Structure of the PID Controller

It is important to select the best PID control structure to match the desired closed loop performance. The selection process is explained in the following subsections:

### 2.1. Proportional P

Proportional P is a pure gain of value  $K_p$ . This control structure is used to obtain satisfactory transient and steady state responses simply by setting a proportional gain in the system, with no dynamic compensation required [14]. There would be no PID controllers without P gain. Proportional controller depends on the process error signal. The  $K_p$  gain represents the ratio of output response to the error signal, and it multiplies the present error of the system with a constant value  $K_p$  [15]. Proportional gain increases the system undamped natural frequency, which means that system speed will increase and also reduce the system damping ratio, leads to increase system overshoot. In the end proportional gain can reduce the system error but never to eliminate it [16, 17]. A system with proportional control will usually have a steady-state offset in response to a constant reference input or to a constant disturbance input which gets smaller as the proportional gain increases. However, there is an upper limit on the size of the proportional gain, otherwise the system becomes unstable. In the end, this controller can be used only when the system is tolerable to a constant steady state error [13].

### 2.2. Proportional Integral PI

PI control structure provides both past and present controls. Since the past sense of control is present, this system will require an optimization scheme which will allow quick

response with minimum overshoot and ringing [14]. The integral term in the controller responds to the past error by eliminating the area under the error curve. The integral control adds a pole at the origin which consequently causes the steady error to become zero. The integral response will continually increase over time unless the error is zero for that reason it usually called automatic reset [15]. When the integral control gain  $K_i$  increases, the system damping ratio decreases, which means extra overshoot, and may lead the system to become unstable [16], [17]. This controller structure is used if the zero steady state error is essential to control requirement. At the same time, the PI controller becomes much slower compared to the P control or PID control. The loss of speed will result in zero error control. In this controller, there are only two parameters to determine in the design process to meet certain steady state design criteria which are  $K_p$  and  $K_i$  [18].

### 2.3. Proportional Integral Derivative (PID)

PID control structure provides past, present, and future controls. This controller is used in a system which needs to be improved in both transient response and the steady state response. The derivative term in the controller is proportional to the rate of change of error signal as an input, not on the error itself. The idea behind derivative control is to control the error before the error develops. Most practical control systems use very small derivative gain, because this gain makes the response very sensitive to the noise in the process variable signal. Thus, if the control loop is slow or the sensor feedback signal is noisy, the derivative response will cause the control system to become unstable [15]. The derivative controller does not affect the system undamped natural frequency, so the system speed remains the same, but derivative control increases the system damping ratio, which leads to reduction of system response overshoot [16, 17]. In the end, the PID structure adds two zeros and one pole at the origin. The additional pole limits the high-frequency gain [19-20], [25].

## 3. PID Tuning Methods

This paper introduces Ziegler-Nichols method as a base method for tuning the PID controller parameters and manual tuning method in both (s-domain) and (z-domain) for more improvement to the controlled system response specifications. Control tuning methods provide the controller parameters in the form of formulae or algorithms. They ensure that the obtained control system to be stable and meet given objectives.

### 3.1. Ziegler-Nichols Step Response Method

The first design method introduced by Ziegler-Nichols is based on a registration of the open loop step response of the stable system. In this method, the unit-step response curve should look as S-shaped, which is characterized by two parameters, which are delay time  $L$  and time constant  $T$ . The parameters are determined from a unit step response of the process, as shown in Figure 2 [21].

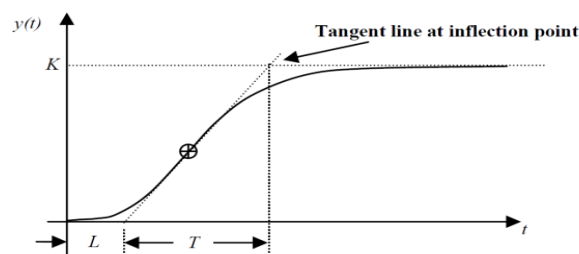


Figure 2. Open Loop System Step Response

The point where the slope of the step response has its maximum value is first determined, and the tangent at this point is drawn. The intersection between the tangent and the time axes gives the  $L$  parameter, while the intersection between the tangent and line  $y(t)=K$

gives the  $T$  parameter [21]. PID parameters can be given directly as a function of  $a=KL/T$  and  $L$ , as shown in Table 1 [22].

Table 1. Ziegler-Nichols Tuning Formula for Step Response Method

Controller	Kp	Ti	Td
P	$1/a$	-	-
PI	$0.9/a$	$3L$	-
PID	$1.2/a$	$2L$	$L/2$

### 3.2. Ziegler-Nichols Frequency Method.

The second design method is based on increase  $K_p$  from 0 to a critical value  $K_{cr}$  at which the output first exhibits sustained oscillations. To do that, first set  $T_i=\infty$  and  $T_d=0$ , then the critical gain  $K_{cr}$  and the corresponding period  $P_{cr}$  can be determined by using Routh Criterion method. The Ziegler-Nichols formula used to set the values of the PID parameters is as shown in Table 2 [23]. This method can be applied to the unstable system to make it stable [23].

Table 2. Ziegler-Nichols Tuning Formula for Frequency Method

Controller	Kp	Ti	Td
P	$0.5K_{cr}$	-	-
PI	$0.45K_{cr}$	$P_{cr}/1.2$	-
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

### 3.3. Manual Tuning Method (s-domain)

The individual effects of PID controller three terms on the closed-loop performance are summarized in Table 3 [17]. This table can be used for manually dependent in tuning the PID controller parameters [17].

Table 3. Effects of Independent P, I and D on the System Response

Controller	Rise time $T_r$	Overshoot $OS\%$	Settling time $T_s$	Steady-state error $ess$
Increase P gain [ $K_p$ ]	Decrease	Increase	Small increase	Decrease
Increase I gain [ $K_i$ ]	Small decrease	Increase	Increase	Eliminate
Increase D gain [ $K_d$ ]	Small increase	Decrease	Decrease	Minor change

This table is useful for manual tuning of PID controller, beginning with the values of parameters obtained from Ziegler-Nichols methods, by observing the effect of those parameters on the unit step response of the system. Manual tuning method is a simple method but it requires a lot of experience, and it takes time to get the required response [17].

### 3.4. Manual Tuning Method (z-domain)

The discrete-time closed-loop PID controller is the most popular controller. It has faster time response and rise time than the continuous-time PID controller that depends on the selected sampling time. However, PID controller tuning in z-domain follows the same rule as used in manual tuning in s-domain, but with one more constant to set (sampling time). At the same time, there is critical point that the designer should pay attention to, which is the choice of the sampling period. These samples should have enough frequency in order not to miss system dynamics [9]. To apply this method, firstly, the plant transfer function must change from s-domain to z-domain by using the zero-order-hold (ZOH) method as in Equation 5, and using PID control formula as in Equation (3) & Equation (4) [9].

$$Gp(z) = (1 - z^{-1})Z \left\{ \mathcal{L}^{-1} \left[ \frac{G(s)}{s} \right] \right\} \quad (5)$$

**4. Theoretical Calculation of PID Parameters and Control System Simulation**

To explain more about the mentioned PID tuning methods, a theoretical calculation is required. For that, consider the third order system response by a transfer function of [24].

$$G_p(s) = \frac{1}{(3s+1)(2s+1)(s+1)}$$

The MATLAB Simulink models shown in Figure 3 are used for the control system simulation in s-domain and z-domain.

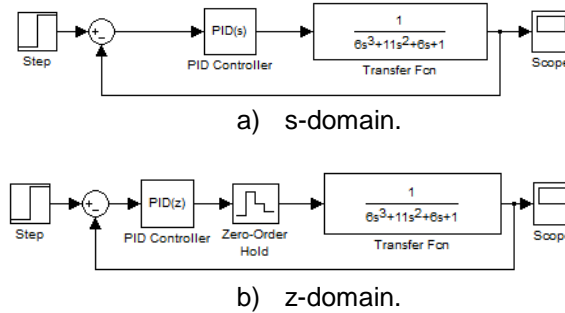


Figure 3. SIMULINK/MATLAB model

**5.1. Ziegler-Nichols Step Response Theoretical Calculation**

To apply the first method of Ziegler-Nichols calculation, step response of the plant needs to be plot as shown in Figure 4. The system response shown in this figure gives  $L=1.44$ ,  $T=7.41$ , and  $K=1$ .

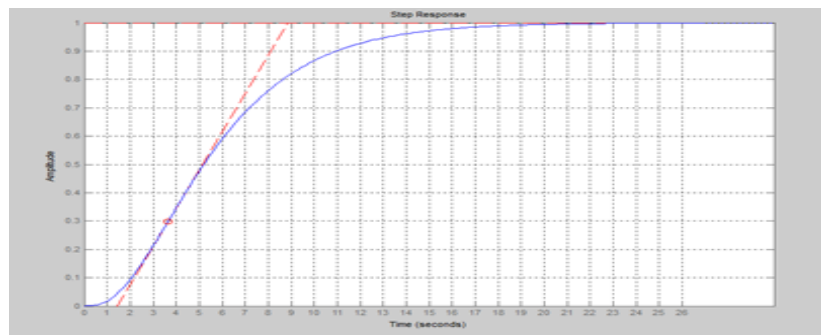


Figure 4. Step Response of Plant Transfer Function

Table 4. PID Parameters using Ziegler-Nichols Tuning Formula for Step Response Method

Controller	Kp	Ti	Td
P	5.1458	-	-
PI	4.6312	4.32	-
PID	6.175	2.88	0.72

Table 5. Response Specification for PID Parameters (Ziegler-Nichols Step Response Method)

Controller	Tr	Overshoot%	Ts
P	1.64	56.6	33.2
PI	1.7	73	97.1
PID	1.45	50.9	20.5

Ziegler-Nichols step response formula in Table 1 can yield PID controller parameters according to each structure type. The final result of this calculation method is given in Table 4, and the response of the controlled system is shown in Figure 5. The control system response specifications are shown in Table 5.

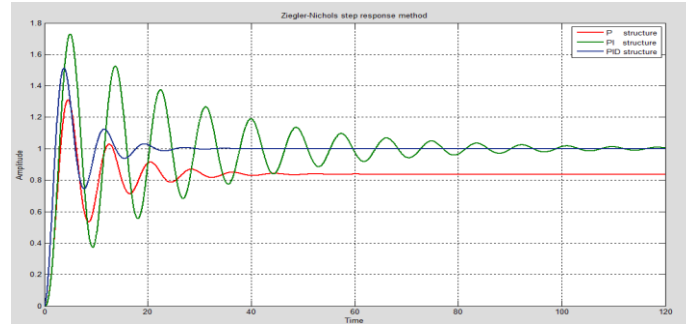


Figure 5. Ziegler-Nichols Step Response Method Result Using MATLAB Simulink

## 5.2. Ziegler-Nichols Frequency Theoretical Calculation

To apply the second method of Ziegler-Nichols calculation, the critical gain  $K_{cr}$  in needs to be determined by using Routh Criteria array for the closed-loop system (the plant with just  $K_p$  gain) characteristic equation:

$$6s^3 + 11s^2 + 6s + 1 + K_p = 0$$

The Routh array is derived as follows:

$$\begin{array}{c|cc} s^3 & 6 & 6 \\ s^2 & 11 & (1+K_p) \\ s^1 & 6 - \frac{6}{11}(1+K_p) & 0 \\ s^0 & (1+K_p) & \end{array}$$

Then, from Routh array, the  $K_p$  results as follows:

$$(1+K_p)=0 \quad K_p=-1$$

$$6 - \frac{6}{11}(1+K_p)=0 \quad K_p=10$$

The value of  $K_{cr}=10$  for Routh stability will then applied in the characteristic equation to find the frequency of oscillation, as follows:

$$\begin{aligned} 6s^3 + 11s^2 + 6s + 11 &= 0 \\ 6(jw)^3 + 11(jw)^2 + 6(jw) + 11 &= 0 \\ 11(1 - w^2) + 6jw(1 - w^2) &= 0 \\ w^2 = 1 \quad \text{then} \quad w &= \sqrt{1} \\ P_{cr} = \frac{2\pi}{w} = \frac{2\pi}{\sqrt{1}} &= 6.283 \end{aligned}$$

The final result of this calculation method is given in Table 6, and the response of the control system is shown in Figure 6. The control system response specifications are shown in Table 7.

Table 6. PID Parameters Using Zeigler-Nichols Tuning Formula for Frequency Method

Controller	Kp	Ti	Td
P	5	-	-
PI	4.5	5.236	-
PID	6	3.1415	0.78

Table 7. Response Specification for PID Parameters (Ziegler-Nichols Frequency Method)

Controller	Tr	Overshoot%	Ts
P	1.67	55.4	33.1
PI	1.77	63.5	62.6
PID	1.48	44.3	16.7

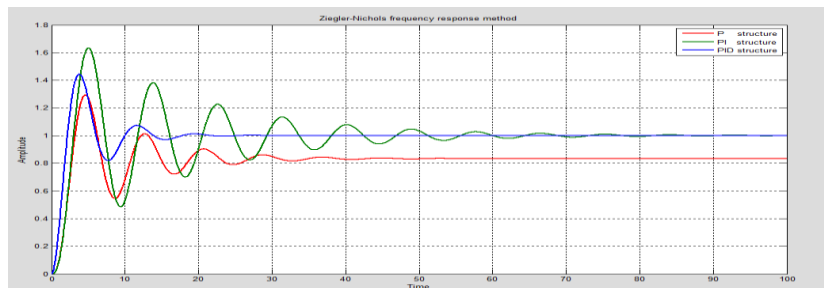


Figure 6. Ziegler-Nichols Frequency Method Result Using MATLAB Simulink.

### 5.3. Manual PID Tuning in s-domain Theoretical Calculation

It is clear from the last result that, the system overshoot was high, and the system was slow to get the final steady state. Table 3 was used for manual tuning method to set the controller parameters by trial and error to get the satisfactory result for the system. The manual tuning results for the PID controller parameters are shown in Table 8. Since this PID controller is used for undergraduate teaching, only manual tuning is discussed. Here, the increase of  $K_p$  will increase response speed, and decrease of  $T_i$  and  $T_d$  leading to reduction of response overshoot and settling time. The controlled system response is shown in Figure 7.

Table 8. PID Parameters and Response Specification for Manual Tuning Method (s-domain)

Controller	Kp	Ti	Td
PID	6.5	1	7
	Tr	Overshoot%	Ts
PID	1.4	21.1	8.09

### 5.4. PID Tuning in z-domain Theoretical Calculation

The system response can be faster ( $T_r$  and  $T_s$ ) just by tuning the sampling time in the z-domain. For the same previous system, with sampling time equal to  $T=0.1s$ , the plant and controller transfer function become:

$$Gp(z) = (1 - z^{-1})Z \left\{ \mathcal{L}^{-1} \left[ \frac{1}{s(3s+1)(2s+1)(s+1)} \right] \right\}$$

$$Gp(z) = \frac{2.654 \times 10^{-5} z^2 + 1.014 \times 10^{-5} z + 2.421 \times 10^{-5}}{z^3 - 2.823 z^2 + 2.656 z - 0.8325}$$

$$g_0 = 6.5 * 0.1 + 1 * (0.1)^2 + 7 = 7.66$$

$$g_1 = -6.5(0.1) - 2 * 7 = -14.65$$

$$g_2 = 7$$

$$C(z) = \frac{7.66z^2 - 14.65z + 7}{0.1z(z-1)}$$

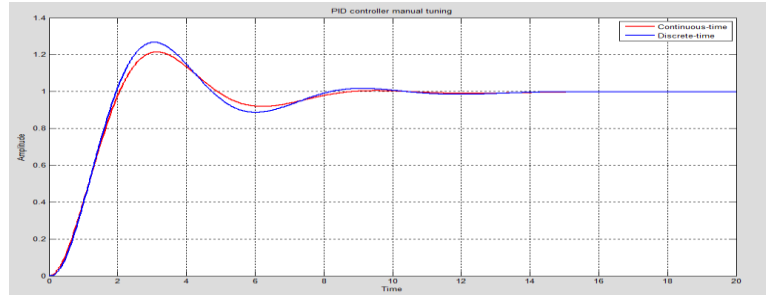


Figure 7. Manual Tuning Response Result Using MATLAB Simulink

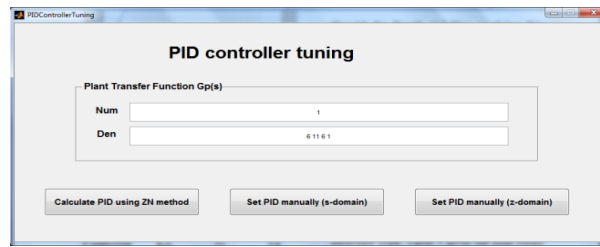


Figure 8. The Main GUI Window of the *PIDControllerTuning* Program

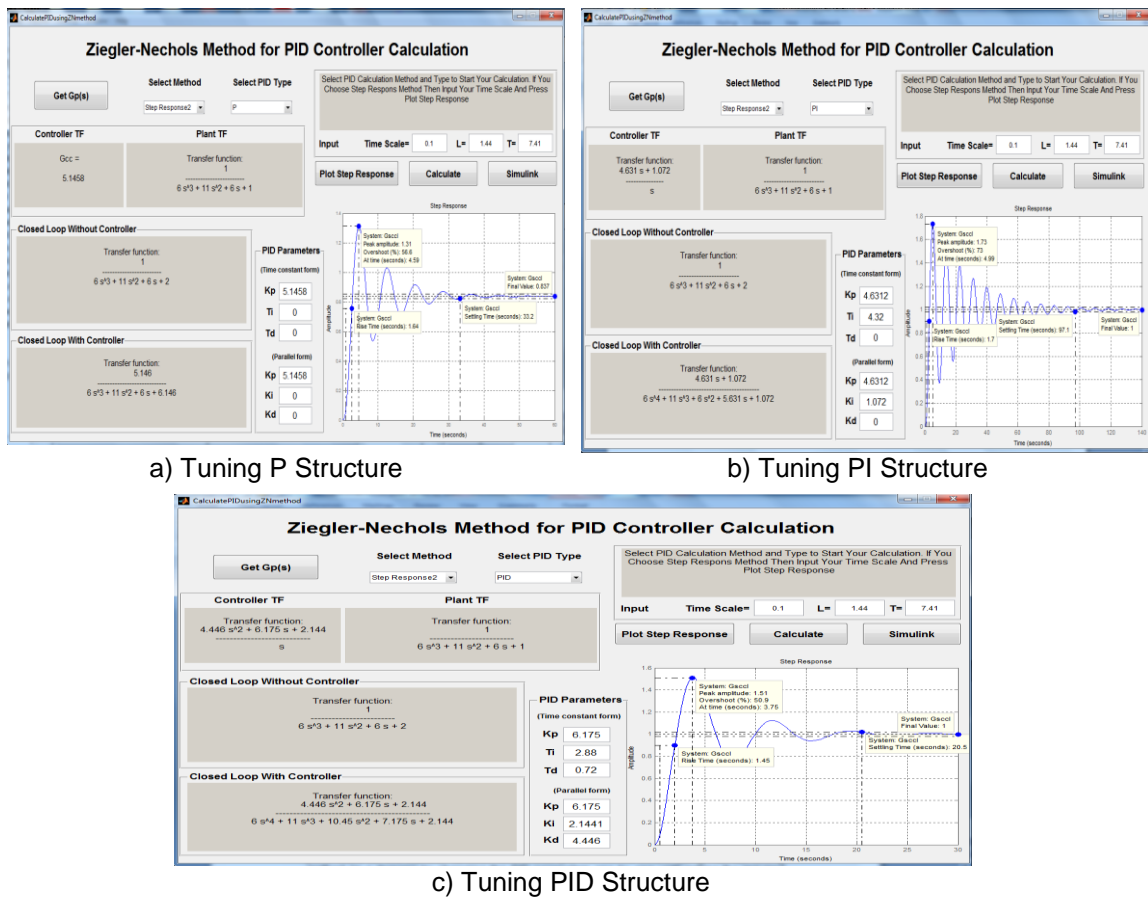


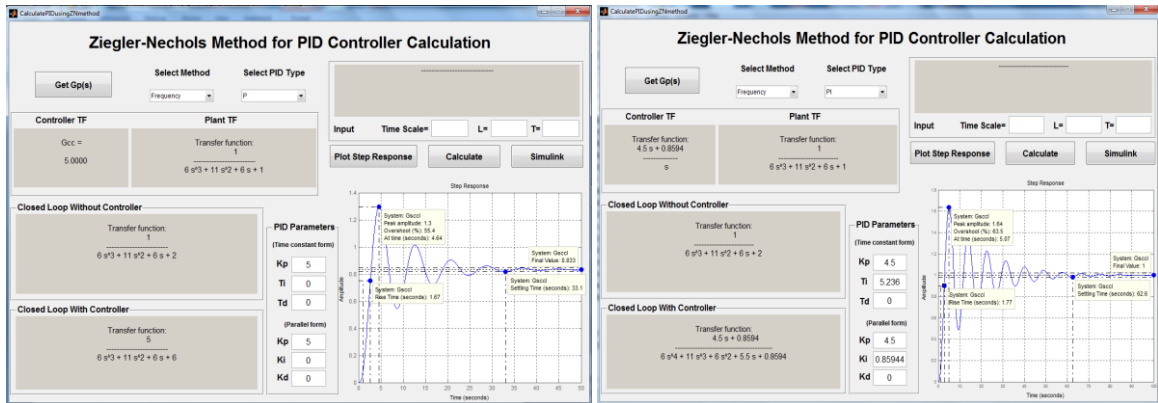
Figure 9. Ziegler-Nichols Step Response Method Result Using GUI/MATLAB Window



In the end, the system response specification will become as shown in Table 9. The controlled system response is shown in Figure 7. This figure shows that the control system response in discrete time is faster than the control system response in continuous time; at the same time showing more overshoot. System response in discrete time can be controlled by changing the PID controller parameters and sampling time.

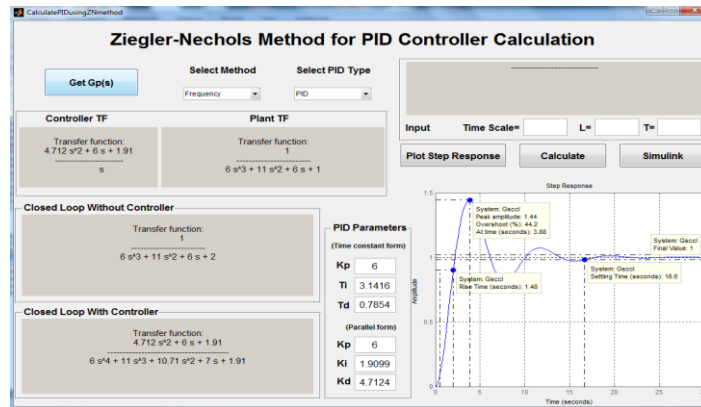
Table 9. PID Parameters and Response Specification for Manual Tuning Method (z-domain)

Controller	Kp	Ki	Kd
PID	6.5	1	7
	Tr	Overshoot%	Ts
PID	1.3	26.3	7.83



a) Tuning P Structure

b) Tuning PI Structure



c) Tuning PID structure

Figure 10. Ziegler-Nichols Frequency Response Method Result Using GUI/MATLAB Window

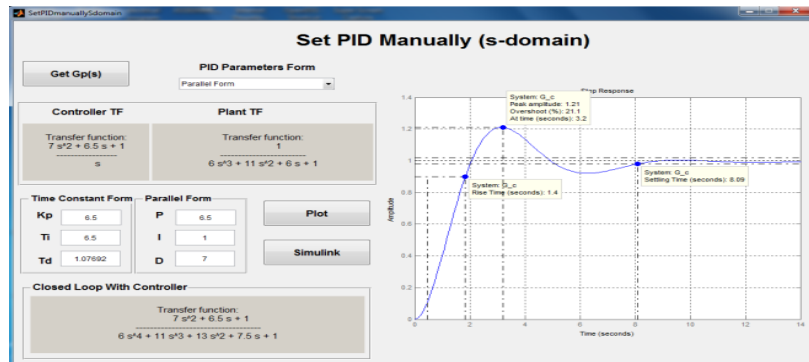


Figure 11. Manual PID Tuning (s-domain)

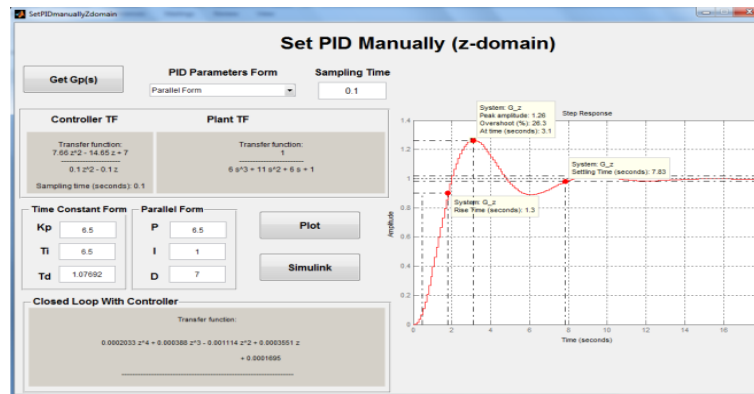


Figure 12. Manual PID Tuning (z-domain)

## 6. Results of GUI Based MATLAB Windows

The previous calculation can be easily done by using the suggested GUI/MATLAB windows for any system order. This paper introduces theoretical calculation example for third order plant to show the GUI windows ability in the calculation of the PID parameters and controlled system response analysis, by using third order transfer function as mentioned earlier as the input to the main GUI window and sub-windows of the PID Controller Tuning GUI package, as shown in Figure 8.

Figure 9, 10, 11, and 12 show the same result as in the theoretical calculation mentioned earlier in this paper, but with more advantages such as reduction in time required for the calculation and more accurate result with higher order plant. Also, it provides a good area for analyzing control system response specifications. Manual tuning GUI window shown in Figure 11 can provide a comfortable environment for tuning the PID controller by easing the change of PID parameters and analyzing the response specifications until getting satisfactory result. Moreover, the GUI window as shown in Figure 12 gives a huge flexibility to the designer for setting the PID controller in z-domain with less time. In the end, the GUI windows as shown in Figure 11 and 12 can be used for implementing and analyzing any control system with PID controller parameters resulting from other tuning methods.

## 7. Conclusion

The GUI based windows presented in this paper can be effectively implemented in engineering education courses related to control system engineering. These GUI windows can reduce the time necessary for the design of PID controller and solve more complicated tasks in process control design. Moreover, Simulink based MATLAB can be used to verify the design for the PID controller. By using this GUI based approach, teaching of control system engineering courses for the undergraduate creates a more user friendly environment in the classroom, that benefitting students at large.

## Acknowledgements

The authors offer full appreciation to Universiti Teknikal Malaysia Melaka and Ministry of Higher Education, Malaysia for providing the research grant FRGS/2/2014/TK03/FKE/01/F00238.

## References

- [1] Ang KH, Chong G, Li Y. PID control system analysis, design, and technology. *IEEE Transactions on Control Systems Technology*. 2005; 13(4): 559-576.
- [2] Ahmad MA, Azuma SI, Sugie T. Performance analysis of model-free PID tuning of MIMO systems based on simultaneous perturbation stochastic approximation. *ELSEVIER*. 2014; 41(14): 6361-6363.
- [3] Visioli A. *Advances in Industrial Control-Practical PID Control*. 2006.
- [4] Ibrahim O, Amuda SAY, Mohammed OO, Kareem GA. Performance Evaluation of Three PID

- Controller Tuning Algorithm on a Process Plant. *International Journal of Electrical and Computer Engineering (IJECE)*. 2015; 5(5): 1075-1082.
- [5] Copeland B. The Design of PID Controllers using Ziegler-Nichols Tuning. *Retrieved*. 2008: 1–4.
- [6] Palm WJ. Modeling, Analysis, and Control of Dynamic System. UAS: John Wiley & Sons, Inc. 2000.
- [7] A El-sharif I, O Hareb F, R Zerek A. *Design of discrete-time PID controller*. International Conference on Control, Engineering & Information Technology (CEIT'14). 2014: 110–115.
- [8] Santana MS, Stubberud AR. Discrete-Time Equivalent to Continuous-Time Systems. *Control Systems, Robotics, And Automation II*.
- [9] Fadali M, Visioli A. Digital control engineering: analysis and design. USA: Elsevier Inc. 2013.
- [10] Peng H, Chiu GTC, Tsao TC. *Design of Discrete Time Controller-Input/Output Approaches*. In: Digital Control of Physical Systems. 2008: 1–18.
- [11] Herjolfsson G, Hauksdottir AS. *Direct computation of optimal discrete-time PID controllers*. Proceedings of the 2004 American Control Conference. 2004; 1(3): 46–51.
- [12] Richard C, Dore RH. Modern control systems. USA: Pearson Education, Inc., Pearson Prentice-Hall, Inc. 2008.
- [13] Charles L, Phillips RD. Feedback control systems. New Jersey: Prentice -Hall, Inc. 2002.
- [14] Gene F, Franklin JD. Feedback control of dynamic systems. New Jersey: Prentice-Hall, Inc. 2002.
- [15] Owen F. Designing and tuning PID controllers. In: Frank Owen. *Editors*. Control Systems Engineering A Practical Approach. California. 2012: 4-8.
- [16] Bucek VJ. Control Systems Continuous and Discrete. New Jersey: Prentice-Hall, Inc. 1989.
- [17] Simulink M, Li S, Jiang Q. Study on pid parameters tuning method based on iz ic ic ic. 2011; (2): 408–411.
- [18] Chen Y, Ma Y, Yun W. Application of Improved Genetic Algorithm in PID Controller Parameters Optimization. *TELKOMNIKA Indonesian Journal of Electrical Engineering*. 2013; 11(3): 1524-1530.
- [19] Michael A, Johunson MH. PID Control: New Identification and Design Method. London: Spring-Verlag London Limited. 2005.
- [20] Xie W, Duan J. The Design and Simulation of Fuzzy PID Parameter Self-tuning Controller. *TELKOMNIKA Indonesian Jurnal of Electrical Engineering*. 2015; 14(2): 293-297.
- [21] Astrom K, Hagglund T. PID Controllers: Theory, Design, and Tuning. *Instrument Society of America, USA*. 1995.
- [22] Dingyu Xue, YangQuan Chen, DPA. Linear Feedback Control. In: Linear Feedback Control. Society for Industrial and Applied Mathematics. Philadelphia. 2007: 183–235.
- [23] Ogata K. *Modern Control Engineering*. 2012.
- [24] Bequette BW. Process Control: Modeling, Design, and Simulation, Prentice Hall PTR. 2002.
- [25] Omar M, Ebrahim MA, Ghany AA, Bendary F. Tuning of PID Controller for Load Frequency Control Problem via Harmony Search algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*. 2016; 1(2): 255-263.