# Comparison Between Optimally-tuned PID with Self-tuning PID for Steam Temperature Regulation.

Mazidah Tajjudin, Mohd Hezri Fazalul Rahiman,
Norlela Ishak, Ramli Adnan.
Faculty of Electrical Engineering,
Universiti Teknologi MARA,
Shah Alam, Malaysia.
e-mail: mazidah@salam.uitm.edu.my

Hashimah Ismail
Faculty of Engineering,
Universiti Selangor,
Bestari Jaya, Selangor.
hashimah@unisel.edu.my

***Abstract*— Performance of conventional PID controller was compared against the self-tuning PID with pole-placement method for steam temperature regulation. The comparisons were done by MATLAB simulation to evaluate the performance of the controllers in terms of set point regulation, response towards set point changes and disturbance rejection. The PID was optimally-tuned using GA with model reference input that will specify the desired closed-loop trajectory during controller optimization. The self-tuning PID control on the other hand was synthesized using pole-placement with type-A PID structure. Generally, both controllers' gave satisfactory response in every aspect but the IAE index shows that the self-tuning PID was superior.**

***Keywords-steam distillation process; self-tuning PID; optimal tuning PID; steam temperature control.***

## I. INTRODUCTION

PID controller has been successfully applied in industrial applications. It has been reported that more than 97% of the controllers in process control industries are of PID type [1]. The ability of PID control mode to compensate most practical industrial processes has led to their wide acceptance such as in pulp and papers industries. Proper tuning of the controller parameters will ensure optimal performance over specified operating range. Some prominent tuning methods are the Ziegler-Nichols, kappa-tau, pole-placement and design based on gain and phase margin specifications in frequency domain. Summary of some renowned PID tuning techniques can be found in [2,3].

Ziegler-Nichols tuning method is based on the system's open-loop step response where the process was approximated by a first-order plus dead-time (FOPDT) model. This technique requires a perfect estimation especially on the process dead-time because this quantity will determine the value of $T_i$ and $T_d$. Another method is based on the closed-loop response under proportional control. Here, the gain is increased until the closed-loop system becomes critically stable and the ultimate period and gain are acquired. This method is not easy to apply on working plant because the system might be brought to unstable state.

Pole-placement method on the other hand is mostly applied for a low-order system. A common approach is to adopt a second-order model and then specify a desired damping ratio and natural frequency for the system. An additional pole is assigned so that the closed-loop design specifications are fulfilled.

Recently, more systematic approaches have been innovated to resolve this matter mainly by applying optimization techniques such as Genetic Algorithm, Particle Swarm Optimization, Bacterial Foraging Optimization and stochastic algorithms[4-6]. Genetic algorithms are capable of locating optimal regions in complex domains and avoiding the difficulties associated with the gradient descent method. A comparative study was done in [7] to evaluate the PID performance optimized using GA, PSO and Ziegler-Nichols on a bus suspension system. The solution provided by PID-GA was better than Ziegler-Nichols by 89%.

The ability to adapt with process variations such as load changes and disturbances has made the self-tuning PID an excellent alternative to the conventional one; regardless of its tuning methods. Moreover, it is also applicable for a non-minimum phase system [8]. Explicit self-tuning control (STC) use the information from model parameters that must be updated recursively in order to synthesized a new controller parameters based on specified design requirements. In some self-tuning controller, the recursive process estimation was not necessary. This type of controller is referred to as implicit self-tuning controller.

Explicit STCs apply *certainty equivalence principle* where model uncertainties during parameter estimation were not considered. It is assumed that these values correspond to their actual values. Theoretical details of the principal can be found in prominent textbooks of adaptive control [9,10]. Figure 1 shows a block diagram of an explicit self-tuning control structure.

978-1-4577-1967-7/12/$26.00 ©2011 IEEE

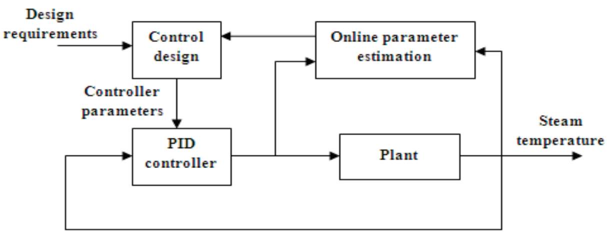


Figure 1 Explicit self-tuning control structure

This study implements an explicit STC to regulate steam temperature of a steam distillation plant for essential oil extraction process. Steam distillation is among the most popular method for essential oil extraction. This method applies hot steam to extract the essential oil contain in the raw materials. Essential oil is volatile in nature and easily evaporated during high temperature. The hot steam can somehow jeopardize the oil quality itself. Regulated steam temperature during extraction process is thus becomes necessary in order to preserve the oil quality.

## II. STEAM DISTILLATION PROCESS MODEL

This study was conducted on a pilot-scale steam distillation plant. It consists of a stainless steel column and a vertically mounted steel condenser. Figure 2 shows the simplified schematic diagram of the steam distillation plant under study. During operation, saturated steam will be generated by boiling the water inside the distillation column. The water was heated up by a 1.5kW coil-type heater. Two RTDs were installed; RTD1 was immersed in the water to monitor water temperature while RTD2 was installed 30cm from steam outlet to monitor the column temperature. This variable will be mentioned as steam temperature throughout this paper.

During closed-loop operation, the steam temperature will be measured by RTD2. The resistance to voltage conversion was done by a signal converter that produced an output within 1V to 5V for temperature range of 0°C to 120°C. Control signal will manipulate the heater power by providing a dc voltage from 0V to 5V to a power controller.

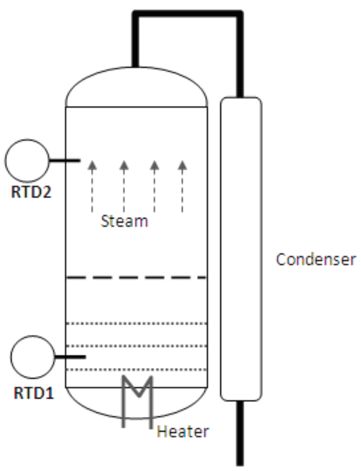


Figure 2 Steam distillation process

### A. *Second-order Plus Dead-time Model (SOPDT)*

For simulation study, the plant will be represented by a second-order model. The model was identified within linear operating temperature of 80°C to 100°C. The process was fitted to a second-order system using System Identification Toolbox and fine-tuned to minimize the residual from experimental data. The best fit model was given by equation (1) and the comparison between model output and experimental data was given in figure 3.

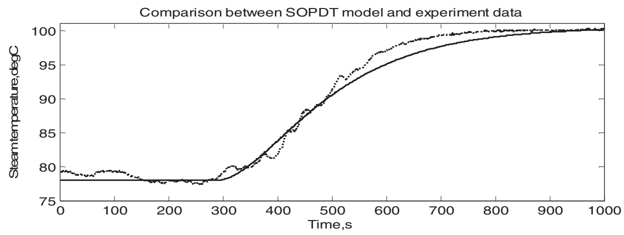


Figure 3 Open-loop response from SOPDT model and experimental data.

$$G(s) = \frac{0.000366}{(s+0.011)(s+0.0074)} \tag{1}$$

The process gain is 4.5°C/V. This model will be used to represent the actual process during simulation. Comparison between experimental data and the predicted model output gives RMSE of 0.042°C.

### B. *ARX Parametric Model*

For PID tuning and self-tuning implementation, a parametric model structure namely autoregressive with exogeneous input (ARX) was chosen to model the dynamics of the steam temperature. Its structure can be written as:

$$y(t) = q^{-nk}\frac{B(q^{-1})}{A(q^{-1})}u(t) + \frac{1}{A(q^{-1})}e(t) \tag{2}$$

where polynomials $A(q{-}1)$, $B(q{-}1)$ are given by:

$$\begin{aligned} A(q^{-1}) &= 1 + a_1 q^{-1} + \cdots + a_{na} q^{-na} \\ B(q^{-1}) &= b_0 + b_1 q^{-1} + \cdots + b_{nb} q^{-nb} \end{aligned} \tag{3}$$

where na and nb is the model order.

Several input output data sets with different magnitude level were used during the modeling to ensure model robustness. Summary of the data are listed in Table 1 and the data are shown in figure 4.

TABLE I
SUMMARY OF INPUT OUTPUT DATA FOR MODEL IDENTIFICATION

| Dataset | PRBS band | PRBS magnitude | Output range |
|---|---|---|---|
| Z0 | 0.2 | [0 5] | [89 99] |
| Z1 | 0.01 | [1 4] | [84 100] |
| Z2 | 0.01 | [2 4] | [88 100] |

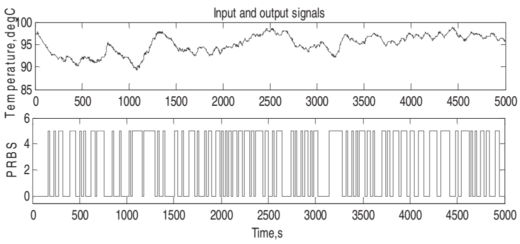

Fig. 4a Dataset Z0

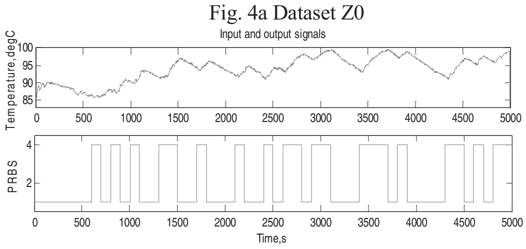

Fig.4b Dataset Z1

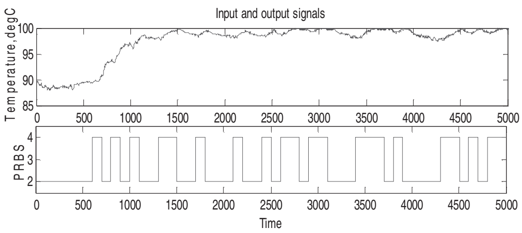

Fig.4c Dataset Z2

Steam temperature is a slow process and significant dead-time caused by transport delay was expected. Time delay between input and output signal was determined using *delayest* command in MATLAB where it was found out to be 27 unit sample.

Modeling was then performed using MATLAB System Identification Toolbox by specifying a second-order ARX with 27 unit delay. Dataset Z0 was divided equally into half and was used for estimation and validation. The estimation resolved in a polynomial given in equation 4. Validation with the second-half of the data shows good agreement with $\%R^2$ of 91.26%. Comparison between experimental data and prediction data from the estimated model was given in Figure 5a.

$$A(q) = 1 - 1{,}988q^{-1} + 0{,}988q^{-2}$$
$$B(q) = q^{-27}(0{,}006521 - 0{,}00637q^{-1}) \qquad (4)$$

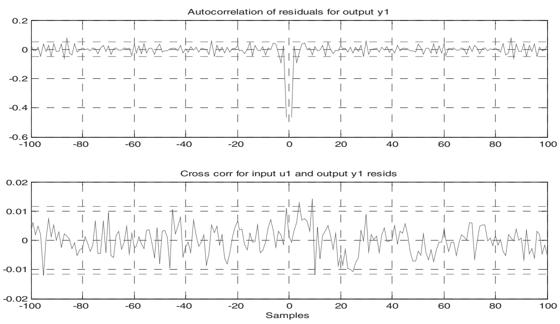

Fig.5 Auto-correlation and cross-correlation of Z0

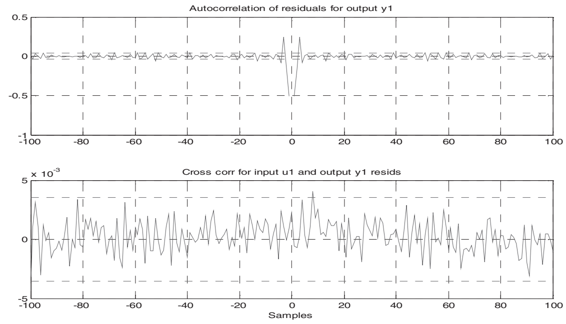

Fig. 6 Auto-correlation and cross-correlation of Z1

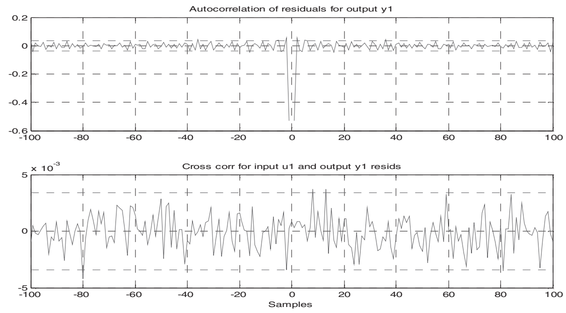

Fig. 7 Auto-correlation and cross-correlation of Z2

To improve the model robustness, it was validated against new data sets; Z1 and Z2. The validation produced $\%R^2$ of 95.37% and 97.11% respectively with acceptable residual analysis. Outcome from the validations are shown in figures 6 and 7.

## III. PID Controller Optimization

A simple PID controller represented in s-domain is given by Equation 4.

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \qquad (5)$$

where $K_p$ is the proportional gain, $K_i$ is the integral coefficient and $K_d$ is the derivative coefficient. The PID controller improves both the transient response as well as the steady-state error of the system. The derivative part improves the transient response by adding a zero to the open-loop plant transfer function. The integrator eliminates error by increasing the system type with additional pole at the origin.

Feedback control system design using PID controller has been adopted in this study because it is simple and robust when applied within specified operating range. To ensure good performance of the controller, suitable values for each parameter namely Kp, Ki and Kd must be tuned perfectly. This study applied GA optimization with a reference model that will specify the desired trajectory for the closed-loop system. Figure 8 illustrates the structure of PID optimization with reference model.

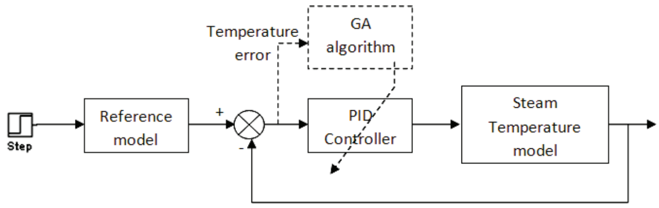


Fig. 8 PID optimization with reference model

This structure was proven to provide faster convergence and produced more consistent results compared to the conventional solution of minimizing the loss function. With the given constraints and requirements, the GA algorithm managed to find a possible solution with Kp = 8.268, Ki = 0.00017, and Kd = 61.18.

## IV. SELF-TUNING PID POLE-PLACEMENT DESIGN

For explicit self-tuning control, the plant parameters need to be updated at each sampling interval. A second-order ARX structure had been identified as the most suitable model structure to represent this process as discussed in section II.

### *A. Parameter Estimation*

The ARX model in regression form can be written as

$$y(k) = \theta^T(k)\varphi(k-1) + e(k) \tag{6}$$

where

$$\theta^T(k) = [a_1, a_2, \ldots a_n, b_1, b_2, \ldots b_n]$$

is the parameter vector and

$$\varphi^T(k-1) = [-y(k-1), -y(k-2), \ldots, -y(k-na), u(k-1), u(k-2), \ldots, u(k-nb)]$$

is the regression vector. The non-measurable random component e(k) is assumed to be zero for simplicity. The quality of the regression model is evaluated by the prediction error given by

$$\hat{e}(k) = y(k) - \hat{y}(k) \tag{7}$$

where is the predicted output and is the measured output. Parameter vector is therefore can be determined by minimizing the loss function given in equation (8). Recursive least square algorithm is the simplest and was widely used as can be referred in [11]and [12] for more details.

$$J_k(\theta) = \sum_{i=0}^{k} e^2(i) \tag{8}$$

### *B. Control Design*

The control structure based on pole-assignment method is shown in Figure 9.

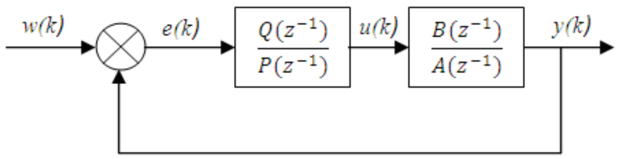


Figure 9 Self-tuning PID pole-placements

Based on the block diagram, the process transfer function is described by

$$G_p(z) = \frac{Y(z)}{U(z)} = \frac{B(z^{-1})}{A(z^{-1})} \tag{9}$$

where $A(z^{-1})$ and $B(z^{-1})$ are polynomials in the form

$$A(z^{-1}) = 1 + a_1 z^{-1} + a_2 z^{-2}$$
$$B(z^{-1}) = b_1 z^{-1} + b_2 z^{-2}$$

The transfer function of a controller is

$$G_R(z) = \frac{U(z)}{E(z)} = \frac{Q(z^{-1})}{P(z^{-1})} \tag{10}$$

Where E(z) = W(z) – Y(z) or the closed-loop system error and

$$P(z^{-1}) = (1 - z^{-1}) + (1 + \Upsilon z^{-1})$$
$$Q(z^{-1}) = q_0 + q_1 z^{-1} + q_2 z^{-2}$$

By substituting $P(z^{-1})$ and $Q(z^{-1})$ into equation (10) and transform into time-sequence equation, the controller output becomes

$$u(k) = q_0 e(k) + q_1 e(k-1) + q_2 e(k-2) + (1 - \Upsilon)u(k-1) + \Upsilon u(k-2) \tag{11}$$

$q_0$, $q_1$, $q_2$, and $\Upsilon$ are the controller parameters need to be determined according to closed-loop design specifications. Equation (11) is a general form of incremental PID. The closed-loop transfer function then becomes

$$G(z) = \frac{Y(z)}{W(z)} = \frac{B(z^{-1})Q(z^{-1})}{A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})} \tag{12}$$

with characteristics polynomial of

$$A(z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1}) = D(z^{-1}) \tag{13}$$

where $D(z^{-1})$ is the desired characteristics polynomial in the form

$$D(z^{-1}) = 1 + d_1 z^{-1} + d_2 z^{-2} + d_3 z^{-3} + d_4 z^{-4} \tag{14}$$

For easier determination of system overshoot and response speed, the following characteristics polynomial is preferred [13].

$$D(z^{-1}) = (z - \alpha)^2 [z - (\alpha + j\omega)][z - (\alpha - j\omega)] \tag{15}$$

The characteristics polynomial in equation (15) has double real poles given by α and a pair of complex conjugate poles where $\alpha^2 + \omega^2 < 1$. Parameter α influenced the speed of the closed-loop system while ω determined the desired overshoot. Equation (13) and (15) give a set of four linear algebraic equations as follows:

$$\begin{gathered} q_0 b_1 + \gamma = c + 1 - a_1 \\ q_0 b_2 + q_1 b_1 + \gamma(a_1 - 1) = d + a_1 - a_2 \\ q_1 b_2 + q_2 b_1 + \gamma(a_2 - a_1) = f + a_2 \\ q_2 b_2 - \gamma a_2 = g \end{gathered} \tag{16}$$

where

$$\begin{aligned} c &= -4\alpha; & d &= 6\alpha^2 + \omega^2 \\ f &= -2\alpha(2\alpha^2 + \omega^2); & g &= \alpha^2(\alpha^2 + \omega^2) \end{aligned} \tag{17}$$

Solving equation (16) will determine the controller parameters.

## V. SIMULATION RESULTS

Comparison between an optimally-tuned PID by GA and self-tuning PID with pole-placement design method was done through simulation. In the simulation, the steam distillation process was represented by an SOPDT model estimated in section II. The comparison will look into the efficiency of both control scheme in performing set point regulation and its robustness against output disturbance.

The optimal PID tuned using GA produced the following parameters: Kp = 8.268, Ki = 0.00017, and Kd = 61.18. This controller will be compared against the self-tuning PID with desired closed-loop poles of α = 0.9 and ω = 0.1. These poles locations will produced less overshoot at fastest dynamics.

### *A) Set point Regulation*

The magnitude of the reference steam temperature was set to 20°C. Both controllers managed to arrive at the desired set point. Some overshoot can be observed in the PID response while smoother trajectory was attained by the self-tuning control. The GA-PID control signal was not so dynamic, which cause the oscillation in the output. The self-tuning on the other hand varies frequently to sustain the specified design requirements. Controlled output and control signal during set point regulation were shown in figure 10a and 10b respectively.

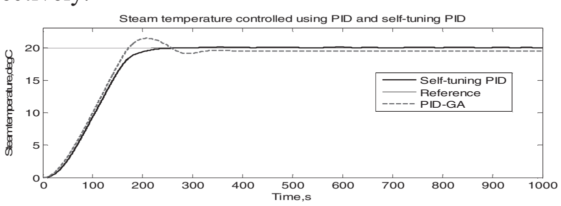


Figure 10a Set point regulation output

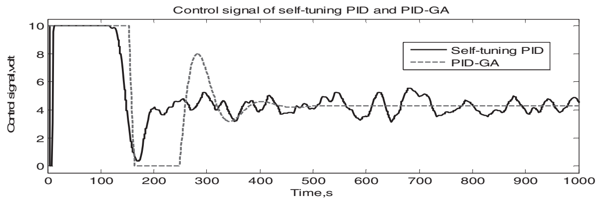


Figure 10b Control signal during set point regulation

The same behavior can be observed when the reference temperature was increased right after the initial set point had been reached. PID controlled output oscillated for a while before settled down at a reference temperature. Corresponding signals for this evaluation was shown in figure 11a and b.

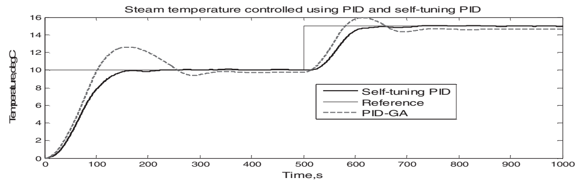


Figure 11a Set point change output

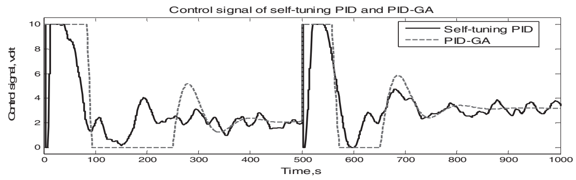


Figure 11b Control signal during set point change

### *B) Disturbance Rejection*

Regulated steam temperature throughout the distillation process is crucial in producing high quality product. A disturbance might occur during the process such as increasing water flow inside the tank that will lower down the temperature. This condition was simulated by injecting a drop in steam temperature at t = 500s. The temperature was lessening by 5°C for 50s.

Both controllers responded to the temperature change and returned the steam temperature to its reference value within 180s. The steam temperature response was given in figure 12a.

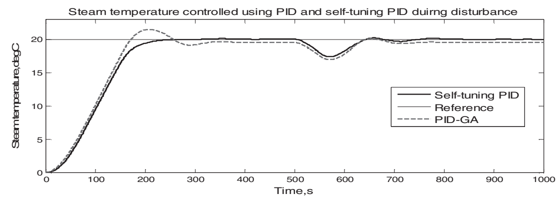


Figure 12a Disturbance rejection output response

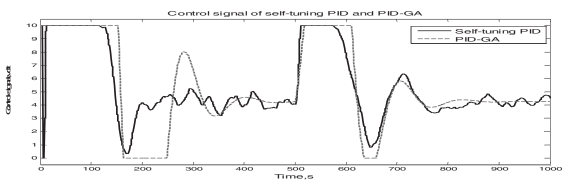


Figure 12b Control signal during disturbance

Table II summarized the Integral of Absolute Error (IAE) for each evaluation. In overall, the STPID control gave more precise regulation by producing less IAE compared to the PID.

TABLE II. IAE COMPARISON BETWEEN STPID AND PID

| | STPID | PID | Deviation (%) |
|---|---|---|---|
| Set point regulation | 2280 | 2381 | 4.43 |
| Set point change | 1277 | 1373 | 7.52 |
| Output disturbance | 2488 | 2547 | 2.37 |

## VI. CONCLUSIONS

A comparison between conventional PID and self-tuning PID was done in evaluating the efficiencies of both control schemes in regulating the steam temperature of a steam distillation process. The PID was tuned using GA optimization technique with model reference to specify the desired closed-loop trajectory. The self-tuning PID on the other hand was synthesized using pole-placement design method. Generally, both controllers produced satisfactory results but IAE index shows that the self-tuning performance is better in set point regulation and disturbance rejection.

## ACKNOWLEDGMENT

This research is supported by UiTM Excellence Fund 600-RMI-ST-DANA-5/3/DST(43/2010) and Ministry of Higher Education fund (600-RMI-ST-FRGS 5/3/Fst (85/2010)). The authors would also like to thank the Faculty of Electrical Engineering, UiTM for providing the facilities to conduct this research and for the financial supports throughout the process.

## REFERENCES

[1] K. Astrom and T. Hagglund, "Revisiting the Ziegler-Nichols step response method for PID control," *Journal of Process Control*, vol. 14, Sep. 2004, pp. 635-650.

[2] A. O Dwyer, "A summary of PI and PID controller tuning rules for processes with time delay . Part 1 : PI controller tuning rules," *IFAC Workshop on Digital Control*, Terrassa,Spain: 2000, pp. 175-180.

[3] A. O Dwyer, "PI and PID controller tuning rules for time delay processes : a summary . Part 2 : PID controller tuning rules," *Proceedings of the Irish Signals and Systems Conference*, National University of Ireland, Galway: , pp. 339-346.

[4] A. Altınten, F. Ketevanlio, S. Erdo, H. Hapo, and M. Alpbaz, "Self-tuning PID control of jacketed batch polystyrene reactor using genetic algorithm," *Chemical Engineering Journal*, vol. 138, 2008, pp. 490-497.

[5] M. Nasri, H. Nezamabadi-pour, and M. Maghfoori, "A PSO-Based Optimum Design of PID Controller for a Linear Brushless DC Motor," *World Academy of Science, Engineering and Technology*, vol. 26, 2007, pp. 211-215.

[6] D.H. Kim and J.H. Cho, "Robust Tuning of PID Controller Using Bacterial-Foraging-Based Optimization," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 9, 2005, p. 669.

[7] A. Karthikraja, "Stochastic Algorithm for PID Tuning of Bus Suspension System," *International Conference on Control, Automation, Communication and Energy Conservation*, 2009, pp. 1-6.

[8] A. Kirecci, I. Eker, and L.C. Dulger, "Self-tuning control as conventional method," *Electrical Engineering*, vol. 85, 2003, pp. 101-107.

[9] K.J. Astrom and B. Wittenmark, *Adaptive Control*, Addison-Wesley, 1995.

[10] V. Bobal, J. Bohm, J. Fessl, and J. Machacek, *Digital Self-tuning Controllers: Algorithms, Implementation and Applications*, Springer, 2005.

[11] K.J. Astrom and B. Wittenmark, "Self-tuning controllers based on pole-zero placement," *IEEE Proceedings*, 1980.

[12] Y. Zhao, E.G. Collins, and D.A. Cartes, "Self-tuning adaptive control for an industrial weigh belt feeder," *ISA Transactions*, 2003, pp. 437-450.

[13] V. Bobal and M. Kubalcik, "Self-tuning controller for temperature control of a thermo-analyzer," 1994, pp. 2-3.