Australian Journal of Basic and Applied Sciences, 7(9): 44-50, 2013
ISSN 1991-8178

Optimizing the Process Parameters of GMV Controller by PSO Tuning Method

[1]S.F. Sulaiman, [1]H.R. Abdul Rahim, [2]S.H. Johari, [1,3]K. Osman, [3]A.F. Zainal Abidin, and [3]M.F. Rahmat

[1]Faculty of Electronics and Computer Engineering, [2]Faculty of Engineering Technology, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
[3]Faculty of Electrical Engineering, Universiti Teknologi Malaysia, 81310 UTM Skudai, Johor, Malaysia.

Abstract: System modeling is very important to develop a mathematical model that describes the dynamics of a system. This work proposes a modeling and designing a Generalized Minimum Variance (GMV) controller using self-tuning and Particle Swarm Optimization (PSO) tuning method for a hot air blower system. Parametric approach using AutoRegressive with Exogenous input (ARX) model structure is used to estimate the approximated model plant. The approximated plant model is then being estimated using System Identification approach. The results based on simulation using MATLAB shows that the GMV controller using PSO tuning method offers a reasonable tracking performances of the system's output.

Key words: Hot air blower system, system modeling and identification, ARX model, GMV controller, tuning method.

INTRODUCTION

In control system engineering, the ability to accurately control the system that involves the temperature of flowing air is vital to numerous design efforts (Philips, C.L. and Harbor R.D.). This work was conducted due to this problem. Since the system to be controlled was non-linear and has significant time delay, this work focused on the objective to maintain the process temperature at a given value. Several steps are considered while doing this work; identify a process, obtain the mathematical model of the system, analyze and estimate the parameters using System Identification approach, design appropriate controllers for controlling the system and implement it to the system by simulation, and lastly make analysis and justification based on the results obtained.

A mathematical modeling process was provided a very useful method in this work since it was used in identifying a process, representing the dynamic, and describing the behavior of a physical system. A mathematical model of a physical system can be obtained using two approaches; analytical approach (physics law) and experimental approach (System Identification) (Ishak, N. *et al.*, 2010). Previous work done by (Rahmat, M.F. *et al.*, 2005) found that the main problem of applying a physical law is, if a physical law that governing the behavior of the system is not completely defined, then formulating a mathematical model may be impossible. Thus, an experimental approach using System Identification was considered in this work. In this work, a mathematical model of the temperature response for the system is developed based on the measured input and output data set obtained from Real Laboratory Process which can be obtained from MATLAB demos. System Identification Toolbox which is available in MATLAB is then used to estimate the parameters and approximate the system models according to the mathematical models obtained. Basically, System Identification approach offers two techniques in describing a mathematical model, which are parametric and non-parametric method. In this work, parametric approach using AutoRegressive with Exogenous input (ARX) model structure is chosen to estimate and validate the approximated system model. In order to ensure the validity of the ARX model, Model Validation Criterion was used to decide whether the ARX model obtained should be accepted or rejected. Once the model have been identified and validated, appropriate controllers were designed to improve the output performance of the system. A Generalized Minimum Variance (GMV) controller with two different tuning methods; self-tuning and Particle Swarm Optimization (PSO) tuning are proposed in this work. The tracking performances of the system by simulation using different tuning method in order to maintain the process temperature at a given value will then be carried out, analyzed, and justified.

MATERIALS AND METHODS

PT326 Process Trainer:

In this work, the PT326 process trainer is employed as a hot air blower system to be modeled. The process of PT326 process trainer works as follows (Process Trainer PT326 Manual Handbook): The air from atmosphere is fanned through a tube. It was then heated at the inlet as it passes over a heater grid before being releases into

Corresponding Author: S.F. Sulaiman, Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, 76100 Durian Tunggal, Melaka, Malaysia.
E-mail: sitifatimahsulaiman@utem.edu.my.

the atmosphere through a tube. Here, adjusting the electrical power supplied to the heater grid will affect the temperature of the air flowing in the tube. For instance, a voltage varying from 0 to +10 Volts produces an air temperature changes from 30 to 60˚C (Process Trainer PT326 Manual Handbook). The flowing air temperature is measured by a thermistor at the outlet and the system generally introduces a significant time delay due to the spatial separation between the thermistor and the heater coil. Thus, the power over the heating device (Watt) is considered as the input to the system, while the outlet air temperature (˚C) as the output to the system.

Model Identification and Estimation:

A mathematical model of the system is developed based on the measured input and output data set obtained from Real Laboratory Process which can be obtained from MATLAB demos. In this work, 1000 measurements of collected input and output data from Real Laboratory Process of PT326 was sampled at the sampling interval is 0.08 seconds. The input to the system was generated as Pseudo Random Binary Sequence (PRBS). PRBS is preferable to be used as an input signal to the system because of the advantage of easy to generate and introduce into a system. Besides, the signal intensity is low with energy spreading over a wide range of frequency makes PRBS as a good choice for force function (Rahmat, M.F. *et al.*, 2005). Fig. 1 shows a plot of measured input and output data of the system in time domain response.

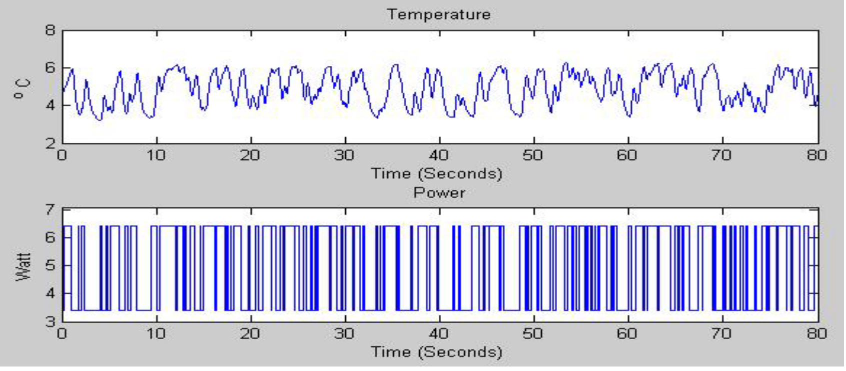

Fig. 1: The plot of measured input and output data of PT326.

In System Identification, the measured input and output data obtained must be divided into two sets of data; the first data set for estimation, while the second data set for validation purpose. In this work, the first 1-500 samples of data were used for estimation and the remaining for validation purpose. To estimate a suitable model structure to approximate the model of the PT326 process trainer, System Identification Toolbox in MATLAB environment is employed. There are a few model structures which are commonly used in real world application and these structures also available in MATLAB System Identification Toolbox: AutoRegressive with Exogenous input (ARX), AutoRegressive Moving Average with Exogenous input (ARMAX), Output Error (OE), and Box Jenkins (BJ). The ARX model structure is chosen since it is the simplest model incorporating the stimulus signal. ARX with the order of $n_a = 2, n_b = 2,$ and $n_k = 3$ (ARX223) were selected in this work, and the discrete-time transfer function as obtained from MATLAB System Identification Toolbox can be represented as:

Assuming the noise is zero,

$$A_o(z^{-1}) = 1 - 1.278z^{-1} + 0.3973z^{-2} \tag{1}$$

$$B_o(z^{-1}) = 0.06518z^{-3} + 0.04497z^{-4} \tag{2}$$

With time delay, $d = 3$, simplifying Eq. (1) and (2) produces (3),

$$\frac{B_o(z^{-1})}{A_o(z^{-1})} = \frac{0.06518z^{-3}(1+0.68994z^{-1})}{1-1.278z^{-1}+0.3973z^{-2}} \tag{3}$$

Thus, the zero polynomials based on Eq. (3) is,

$$B_c(z^{-1}) = 1 + 0.68994z^{-1} \tag{4}$$

Factorizing Eq. (4) gives the location of zero at $z = -0.68994$. That was mean the model obtained is a minimum phase model, with the single zero lie inside the unit circle. The zero and poles plot of the model is shown in Fig. 2.

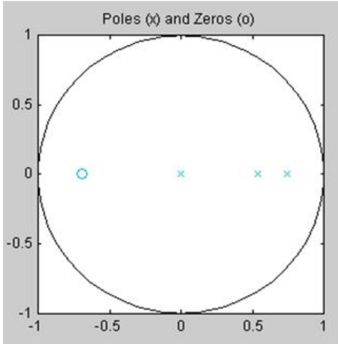

Fig. 2: The plot of poles and zeros.

The model validation is considered as a final stage of the System Identification approach. As described earlier in a beginning of Section I, the second set of data (501-1000 samples) will be used for validation purpose. In this work, the model validation is to verify the identified model represents the process under consideration adequately; to check the validity between the measured and desired data under a validation requirement. Akaike's Model Validity Criterion is used since it is very popular method for validating a parametric model such as ARX and ARMAX model structure. The mathematical model obtained is validated based on its Best Fit, Loss Function, and Akaike's Final Prediction Error (FPE). A model is acceptable if the Best Fit is more than 80% (>80%). The term *fit* means the closeness between the measured and simulated model output, and it can be calculated using Eq. (5):

$$Fit = 100\left[1 - \frac{norm\ (\hat{y}-y)}{norm\ \ (y-\bar{y})}\right]\% \tag{5}$$

where y: true value, $\hat{y}$: approximate value, and $\bar{y}$: mean value

A model is acceptable if the Loss Function and Akaike's FPE is as smallest as possible. The values of Loss Function and Akaike's FPE can be calculated using Eq. (6) and (7):

$$V = \frac{e^2(k)}{N} = \frac{e^T(k).e(k)}{N} \tag{6}$$

where; $e(k)$: error vector

$$FPE = V\frac{\left(1+\frac{d}{N}\right)}{\left(1-\frac{d}{N}\right)} \tag{7}$$

where; V: loss function, d: no. of approximated parameter, and N: no. of sample

Using System Identification Toolbox, the best fit of the output model is 89.18% as depicted in Fig. 3. From the plot, a measured value is indicated by a black curve and the simulated model output is indicated by a blue curve. The model plant is acceptable since the percentage of the best fit is greater than 80%. The Loss Function and Akaike's FPE of the ARX223 model is considered small with the value 0.00170053 and 0.00172774, as recorded in Table 1. Thus, the approximated model of ARX223 is acceptable since all those three criteria's of Model Validation Criterion are satisfied.

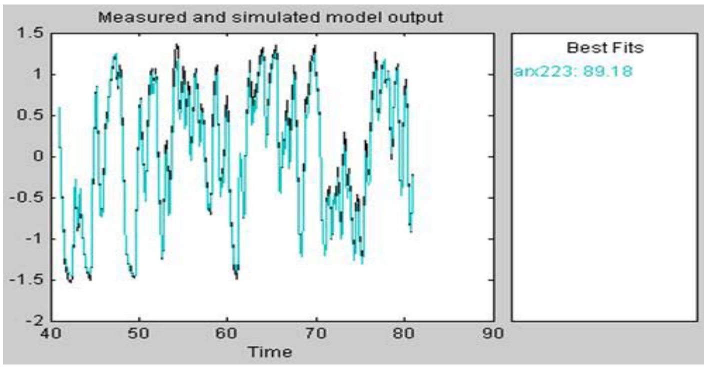

Fig. 3: The measured and simulated model output (Best Fit).

Table 1: Akaike's Model Validity Criterion Value Based on ARX223 Model Structure.

ARX223	
Best Fit	89.18%
Loss Function	0.00170053
Akaike's FPE	0.00172774

Results:
Controller Design:

Several studies are currently kept on tackling this issue on designing a suitable controller for improving the output performance of the system considered. (Harsono, E., 2009) designed a Proportional (P) and Proportional-Integral (PI) Controller, and has implemented both controllers to the simulation and real-time process. (Azlan, M.F.I., 2010) designed a Proportional-Integral-Derivative (PID) controller to control the system and he proposed Ziegler Nichols tuning method for tuning those PID parameters. An intelligent tuning method for PID controller using Radial Basis Function Neural Network (RBFNN) tuning method was presented by (Alsofyani, I.M.A., 2010). In this work, a Generalized Minimum Variance (GMV) controller using self-tuning and Particle Swarm Optimization (PSO) tuning method was proposed. GMV controller is an extension of Minimum Variance Control (MVC). This method is introduced in order to accommodate servo control and to overcome disadvantages introduced by MVC, where in MVC there are some drawbacks that the designer must consider when applying it: the performance of MVC is affected by time delay, *k*, MVC ignores the amount of control effort required, and many more.

In GMV controller, a pseudo output as in Eq. (9) is introduced:

$$\emptyset(t+k) = Py(t+k) - Rw(t) + Qu(t) \tag{8}$$

where; $y(t+k)$: output, $w(t)$: setpoint or servo signal, $u(t)$: input, and $P, R, Q\ value$: set by designer

In GMV controller, the identity is given by Eq. (9) below:

$$AE + z^{-k}G = PC \tag{9}$$

where; $E = 1 + e_1 z^{-1} + \cdots + e_{n_e} z^{-n_e}$ and $G = g_0 + g_1 z^{-1} + \cdots + g_{n_g} z^{-n_g}$
For a unique solution, the followings are ensured:
$n_e = k - 1$
$n_g = \max(n_a - 1, n_p + n_c - k)$

The GMV controller law used:

$$u(t) = \frac{H}{F} w(t) - \frac{G}{F} y(t) \tag{10}$$

where; $F = BE + QC$ and $H = RC$

Fig. 4 illustrates the general structure of a GMV controller block diagram.

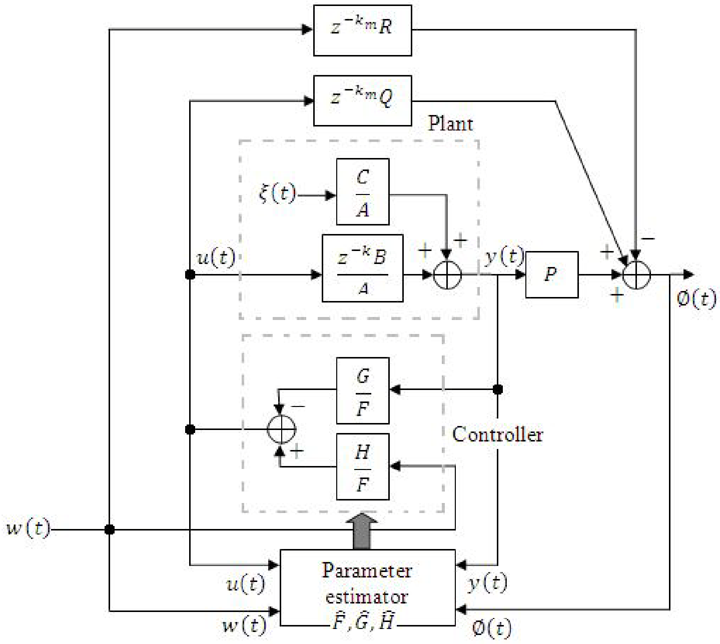

Fig. 4: The general structure of GMV controller block diagram $(k_m \leq k)$.

Therefore, the steps in designing self-tuning GMV controller in this work can be summarized as below:

Step 1:
Based on Eq. (10), equation as below is determined:

$$\emptyset(t) = Py(t) - Rw(t-k) + Qu(t-k) \tag{11}$$

Step 2:
$\hat{F}, \hat{G}$, and $\hat{H}$ are estimated using RLS algorithm:

$$\emptyset_{RLS}(t) = \hat{G}y(t-k) + \hat{F}u(t-k) - \hat{H}w(t-k) + \xi(t) \tag{12}$$

Replacing $k = 3$, Eq. (12) can be written in regression form as below:

$$\emptyset_{RLS}(t) = [y(t-3) \quad y(t-4) \quad u(t-3) \quad u(t-4) \quad u(t-5) \quad u(t-6) \quad -w(t-3)] \begin{bmatrix} \hat{g}_0 \\ \hat{g}_1 \\ \hat{f}_0 \\ \hat{f}_1 \\ \hat{f}_2 \\ \hat{f}_3 \\ \hat{h}_0 \end{bmatrix} + \xi(t)$$

Step 3:
The GMV controller law is calculated and applied to the system:

$$u(t) = \frac{\hat{H}}{\hat{F}}w(t) - \frac{\hat{G}}{\hat{F}}y(t) \tag{13}$$

Step 4:
The algorithm is repeated for the next iteration or sampling time.

i) Self-Tuning GMV Controller:

As mentioned earlier in Eq. (9), the value of parameters P, Q and R was set by the designer. Thus, in this work, the value of P and Q is assumed to be 1, while R value to be 0, as depicted in Table 2.

Table 2: The Value of P, Q and R (Set by the User).

Weighting Factors		
P	Q	R
1	1	0

ii) PSO-Tuning GMV Controller:

PSO tuning method is used in GMV controller to tune the value of weighting factors P, Q and R in the pseudo output. Table 3 is the recorded values of P, Q and R tuned by PSO.

Table 3: The Value of P, Q and R (Tuned by PSO).

Weighting Factors		
P	Q	R
0.705	-5	-4.72

Analysis And Discussion:

In this section, the proposed controllers are implemented by simulation using MATLAB software and the corresponding results are presented. A unit step input is considered as a desired temperature of the process, while the outputs from the controllers designed are the actual or measured values. A temperature of 40°C with a step change of 5 seconds is designed as a desired temperature of PT3276 process trainer. The aim of this work is to design appropriate controllers that can track or follow the setpoint (desired temperature) based on the approximated model plant that obtains using System Identification approach. The proposed controllers were developed and their performances are discussed and analyzed. Fig. 5 shows the output performances of the GMV controller. From a simulation result obtained, responses that uses PSO tuning method gives the best stability results compared to the self-tuning responses that uses a default value to tune the parameters P, Q and R. This is because; GMV controller that uses PSO tuning method also has the fast response (T_r) and allowable value of percentage overshoot, which is only 2.6% of overshoot.

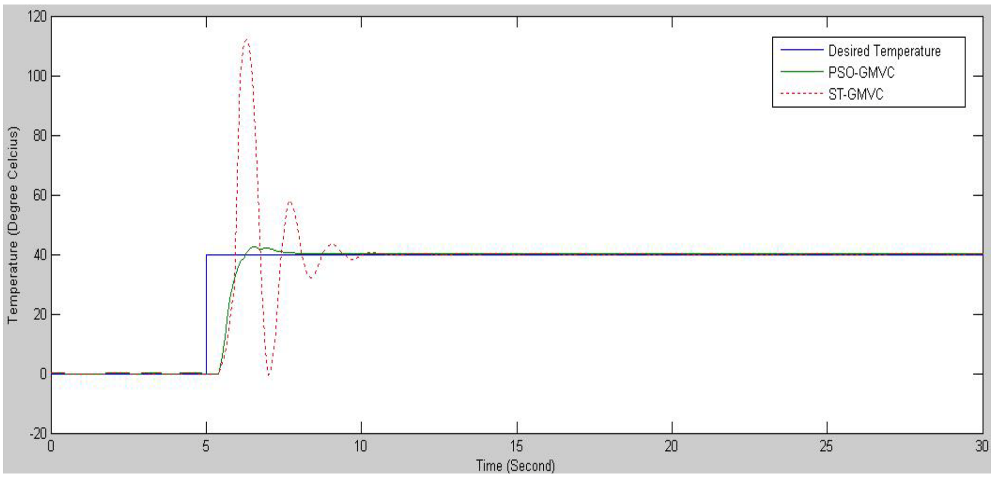

Fig. 5: Output responses of the GMV controller.

The summary of the performances of the controllers designed is shown in Table 4 quantitatively. Based on Table 4, clearly described that each controller has their own advantage and disadvantage. A GMV controller using PSO tuning method provide a good result in term of rising time (T_r) and settling time (T_s). However, both are good in eliminating error, improve the speed of the system response and make the system's output follow the reference signal at steady-state.

Table 4: Performances of the GMV Controller.

Response Characteristic	Generalized Minimum Variance (GMV) Controller	
	Self-tuning	PSO-tuning
Percent Overshoot ($\%OS$)	180%	2.6%
Peak Time (T_p)	6.6s	6.7s
Settling Time (T_s)	4.8s	3.8s
Rise Time (T_r)	0.4s	0.6s
Percent Steady-State Error ($\%e_{ss}$)	0%	0%

Results:

In this work, a GMV controller with self-tuning and PSO tuning method are designed to control one system, namely hot air blower system (PT326 process trainer), which is non-linear and has a significant time delay. The control objective is; to accurately control the system that involves the temperature of flowing air, where the controllers designed must be able to maintain the process temperature of the system at a given value. From a simulation result obtained, it can be concluded that a GMV controller using PSO tuning method obviously has improved the performance of the Self-Tuning GMV controller in term of rise time (T_r) and settling time (T_s). Thus, clearly seen that in terms of transient response of the controlled system, a GMV controller using PSO tuning method offers better results; fast rise time (T_r), setting time (T_s), and also its ability to eliminate percent error $(\%e_{ss})$. Based on observation from this work, objective of this work is successfully achieved. For further work, effort can be devoted by implementing both controllers in real-time process, so that the results obtained from experiment can be validated with theoretical or simulation.

ACKNOWLEDGEMENT

This work was supported by a grant (Project No. PJP/2012/FKEKK(1C)/S01008) from Short Term Research Grant Scheme of Universiti Teknikal Malaysia Melaka. The authors are indebted to Faculty of Electronics and Computer Engineering (UTeM), Faculty of Engineering Technology and Faculty of Electrical Engineering (UTM) for their help in conducting this work.

REFERENCES

Philips, C.L. and R.D. Harbor, Feedback Control Systems. Prentice Hall, 4th Edition.

Ishak, N., S.K. Harun, H. Ismail, N. Abdul Razak and R. Adnan, 2010. Model Identification and Real-Time Linear Position Control of Servomotor. In the Proceedings of the IEEE: Control and System Graduate Research Colloquium, pp: 71-74.

Rahmat, M.F., Y.K. Hoe, S. Usman and N. Abdul Wahab, 2005. Modelling of PT326 Hot Air Blower Trainer Kit Using PRBS Signal and Cross-Correlation Technique. Jurnal Teknologi: Universiti Teknologi Malaysia: 22.

Azlan, M.F.I., 2010. System Identification and Estimation of a Hot Air Blower System Using Parametric Method from B. Eng. thesis, Universiti Teknologi Malaysia: 45-48.

Process Trainer PT326 Manual Handbook. Feedback Instruments Ltd.

Alsofyani, I.M.A., 2010. A Model-Based Neural Network Controller for a Hot Air Blower System from M. Eng. thesis, Universiti Teknologi Malaysia, 42-43: 53-55.

Selamat, H., 2011. Introduction to adaptive and self-tuning control (Chapter 5), Universiti Teknologi Malaysia: Lecture Module, pp: 128-158.

Yu, C.C., 1999. Auto-Tuning of PID Controllers: Relay Feedback Approach. Springer-Verlag, London.

Pirabakaran, K. and V.M. Becerra, 2001. Automatic Tuning of PID Controllers Using Model Reference Adaptive Control Techniques. In the Proceedings of the IEEE Industrial Electronics Society, pp: 736-740.

Mohammad Ali, A.R., 2009. Optimization of controller parameters for a couple tank system using metamodeling technique, M. Eng. thesis, Universiti Teknologi Malaysia: 33.

Poli, R., 2007. An Analysis of Publications on Particle Swarm Optimisation Applications., pp: 1-10.

Kennedy, J. and Eberhart, R.C., 1995. Particle Swarm Optimization. In the Proceedings of the IEEE International Joint Conference on Neural Networks, pp: 1942-1948.