SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 16, No. 2, June 2019, 149-160

UDC: 681.515 **DOI: https://doi.org/10.2298/SJEE1902149K**

Direct and Indirect Self-Tuning Generalized Minimum Variance Control

Hayder Jasim Kareem[1], Ali Abdulrazzak Jasim[1], Mohammed Abase Yousif[1], Thamer Ahmed Abdullah[1]

Abstract: Theoretically, several self-tuning control (STC) algorithms have been developed and many simulation results have proved their feasibility in the past years, but applications of STC are hardly seen. This paper proposes direct and indirect STC plans that can be applied to a chemical process system with a time delay auto-regressive and exogenous model of a varied time constant. The plan controller is a combination of a generalized minimum variance control (GMVC) strategy and an identification algorithm (simplified parameter) called recursive least squares to estimate controller parameters for the direct method and plan parameters for the indirect method. The experimental results show that GMVC is able to track the desired input or set point.

Keywords: Self-Tuning, Time delay, Set point.

1 Introduction

Self-tuning control (STC) is a type of adaptive control that was developed by Astrom and Wittenmark [1 – 3]. The basic idea of STC is to construct an algorithm that automatically modifies the parameters until they reach a specific requirement [4, 5]. This is done by the addition of an adjustment mechanism, which monitors the system (in a controlled setting) or the signal (in a processing setting) and regulates the coefficient of a corresponding controller or signal processor to maintain the required performance [6]. The main idea of the STC system is described in the following [7].

Unknown or time-varying plant parameters are estimated using a parameter estimator that is based on the control signal (input $u(t)$) and the plant's signal (output $y(t)$).

The estimated parameters are used in the design of controller parameters via the controller algorithm design module. The controller parameters are adjusted according to the estimated plant parameters. Fig. 1 below represents the general structure of STC.

[1]Department of Electrical Techniques, Al- Mussaib Technical Institute, Al-Furat Al-Awast Technical University, 51009, Babylon, Iraq; E-mails: harwar80@gmail.com

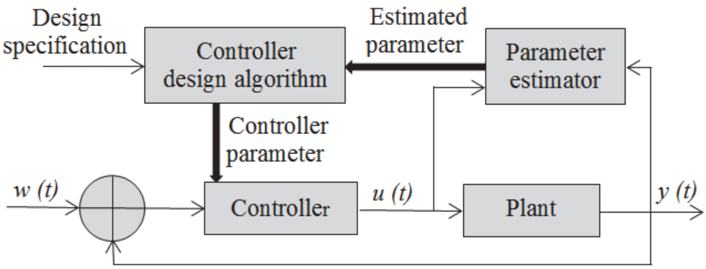

Fig. 1 – *General structure of an STC [8].*

STC can be divided into direct and indirect schemes. The direct STC technique is a function used to minimize output variance, hence the name minimum-variance-control (MVC) [8–10]. Another aspect of direct STC is that it is based on adaptive pole-assignment algorithms introduced by Astrom et al. [11, 12]. The poles of a closed-loop system are assigned depending on the performance requirement [13, 14]. As for the indirect scheme, the parameters of the process are specified directly and then the regulator parameters are computed according to the process model and the selected control law [15]. However, the regulator parameters are identified and directly updated for an indirect scheme, so the indirect process requires a shorter computation time than the direct method [3, 16, 17].

Mourad and Samira applied direct STC to a class of great-scale systems [18]. The schemes are described by linear time-invariant patterns with unknown and possibly slowly varying parameters. The use of the generalized minimum variance control (GMVC) technique in conjunction with the recursive least squares (RLS) identification algorithm displays sufficient flexibility [19]. In addition, Allidina et al. [20] proposed direct STC via the pole-assignment technique. However, this method needs more online computation time than GMVC STC [21, 22]. The unknown parameters of the regulator are estimated online using the recursive estimation method [23]. Comparisons of many different identification and parameter estimation methods have been described in [24]. The main contributions of this paper are organized as follows.

First, a GMVC with RLS for both direct and indirect schemes is used to describe a process system. Second, the parameters of both of these schemes are presented through Matlab (MathWorks 2016b) simulation [25]. Finally, the conclusion is given.

2 Controller Design Procedure

This paper is divided into indirect and direct STC schemes. Details of the direct method are presented in the first part followed by detailing of the indirect scheme in the second part. A general block diagram for process systems used in this study is shown in Fig. 1. The design criterion contains both the magnitude of input signals and the output error variance, adding the ability to expand non-minimum phase plants. The system is described by the following transfer function:

$$G(s) = \mathrm{e}^{-T_D s} \frac{1}{\tau s + 1}. \tag{1}$$

where T_D is the time delay of the process and τ is a time constant. It is assumed that the value of τ varies based on the sampling time below:

$$\tau(t) = \begin{cases} 0.5, & \text{for } t \le 10\,\text{s}, \\ 2, & \text{for } t \ge 10\,\text{s}. \end{cases}$$

The performance of the two controllers must be compared via computer simulation. The transfer function describes in (1) then leads to the following discrete time Z-transfer function form:

$$\frac{y(s)}{u(s)} = \frac{1}{\tau s + 1} \mathrm{e}^{-T_D s}.$$

From the table of Z transforms [26]:

$$\begin{gathered} \frac{1}{\tau s + 1} e^{-KTs} \quad \Rightarrow \quad \frac{(1 - \mathrm{e}^{-(T/\tau)}) z^{-(K+1)}}{1 - \mathrm{e}^{-(T/\tau)}\, z^{-1}} = \frac{y(t)}{u(t)}, \\ G(z) = \frac{y(t)}{u(t)}. \end{gathered} \tag{2}$$

For $T_D = 0.2$ and $T_S = 0.1$, then $K = 2$.

2.1 Direct schemes

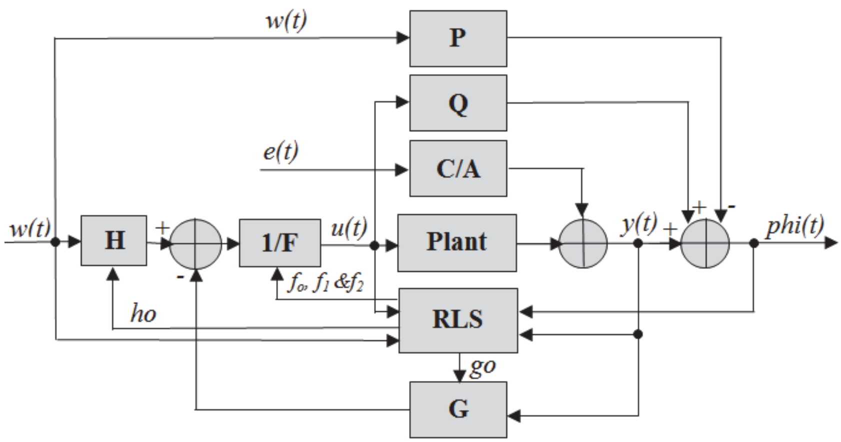

Fig. 2 – *Direct GMVC block diagram.*

In direct self-tuning GMVC, the parameters of the controller are estimated using RLS. A schematic representation of the direct adaptive model reference control system is shown in Fig. 2 [27].

The controller structures or parameters can be determined based on the discrete-time transfer function of the plant in reference to (2) and the block diagram above. When taking $P = Q = R = 1$, the unique solution of the remaining controller parameters can be obtained [28]. **Table 1** summarizes the information obtained so far.

Table 1

Values for GMVC and plant parameters.

<table>
<tr><th>Parameters</th><th>$t \leq 10$</th><th>$T > 10$</th><th>Note</th><th>Define</th></tr>
<tr><td colspan="5">GMVC Controller</td></tr>
<tr><td>K_m</td><td colspan="2">1</td><td rowspan="5">User defined</td><td rowspan="10">Controller parameters</td></tr>
<tr><td>K_{tf}</td><td colspan="2">1</td></tr>
<tr><td>P_o</td><td colspan="2">1</td></tr>
<tr><td>q_o</td><td colspan="2">1</td></tr>
<tr><td>r_o</td><td colspan="2">1</td></tr>
<tr><td>h_o</td><td>1.000</td><td>1.000</td><td rowspan="5">Estimated by RLS algorithm</td></tr>
<tr><td>f_o</td><td>1.181</td><td>0.049</td></tr>
<tr><td>f_1</td><td>0.148</td><td>0.0472</td></tr>
<tr><td>f_2</td><td>1.211</td><td>1.159</td></tr>
<tr><td>g_o</td><td>0.671</td><td>0.904</td></tr>
<tr><td colspan="5">Plant</td></tr>
<tr><td>a_o</td><td>1</td><td>1</td><td rowspan="5">Problem defined</td><td rowspan="5">Estimator parameters</td></tr>
<tr><td>A_1</td><td>–0.8187</td><td>–0.951</td></tr>
<tr><td>b_o</td><td>0.1813</td><td>0.0482</td></tr>
<tr><td>c_o</td><td colspan="2">1</td></tr>
<tr><td>K</td><td colspan="2">1</td></tr>
</table>

According to this information and based on the self-tuning GMVC algorithm in **Table 1**, the parameters of F, G, and H are estimated directly using the following steps:

Step 1:

$$\phi(t) = Py(t) - Rw(t-k) + Qu(t-k) \tag{3}$$

Step 2:

$\hat{G}$, $\hat{H}$, and $\hat{F}$ are estimated using the RLS algorithm based on the equation an adapted from the case study as follows:

$$\phi(t) = \hat{G}y(t-k) + \hat{F}u(t-k) + \hat{H}w(t-k) + \xi(t), \tag{4}$$

$$F = f_0 + f_1 Z^{-1} + f_2 Z^{-2}, \tag{5}$$

$$\phi(t) = g_0(t-3) + f_o u(t-3) + f_1 u(t-4) + f_2 u(t-5) - h_0 w(t-3). \tag{6}$$

In this step, we apply RLS in order to estimate the controller parameters from (6):

$$\phi_{RLS}(t) = \left[y(t-3) \quad u(t-3) \quad u(t-4) \quad u(t-5) \quad -w(t-3) \right] \begin{bmatrix} \hat{g}_o \\ \hat{f}_o \\ \hat{f}_1 \\ \hat{f}_2 \\ \hat{h}_o \end{bmatrix}. \tag{7}$$

Step 3:

The GMV controller is calculated using:

$$u(t) = -\frac{\hat{G}}{\hat{F}} y(t) + \frac{\hat{H}}{\hat{F}} w(t). \tag{8}$$

Step 4:

The algorithm is repeated until the maximum number of iterations is reached.

2.2 Indirect scheme

In indirect schemes, the Z-transfer function described earlier in (2) needs to be rearranged form plant model as shown by the equation below, where the noise value C is initially set as 1.

$$y(t) = \frac{B}{A} u(t) + \frac{C}{A} \varepsilon(t), \tag{9}$$

$$= \frac{bz^{-2}}{1 - az^{-1}} u(t) + \frac{1}{1 - az^{-1}} \varepsilon(t). \tag{10}$$

Therefore, by comparing (8) and (9), we get the plant coefficient as follows:

$$E = 1 + e_1 z^{-1} \quad \text{and} \quad G = g_o, \tag{11}$$

$$AE + Z^{-k} G = PC. \tag{12}$$

where the PC value is set to 1 in this design and the unique solution of the controller coefficient is obtained as in **Table 2**. The values of a and b depend on the time delay of the process T_D, which is set as 0.1 s.

In the light of the information above and based on the self-tuning GMVC algorithm in **Table 2**, this value will be used to set our plant parameters in Matlab Simulink.

Table 2
The parameter values a and b.

Parameter	Predicted values	
	$\tau = 0.5$ s	$\tau = 2$ s
A	0.1813	0.9512
B	0.8187	0.0480

For the indirect GMVC model, the value of F can be obtained based on the equation below:

$$F = EB + \lambda C\,, \tag{13}$$

$$\therefore = (1 + az^{-1})b + \lambda\,, \tag{14}$$

where $\lambda = 1$ is set in our design. Lastly, the parameter H needs to be fed into the design and can be found by means of the formula below:

$$H = RC\,. \tag{15}$$

With all the parameters that have been determined above, the block diagram for the indirect scheme is as shown in Fig. 3 below.

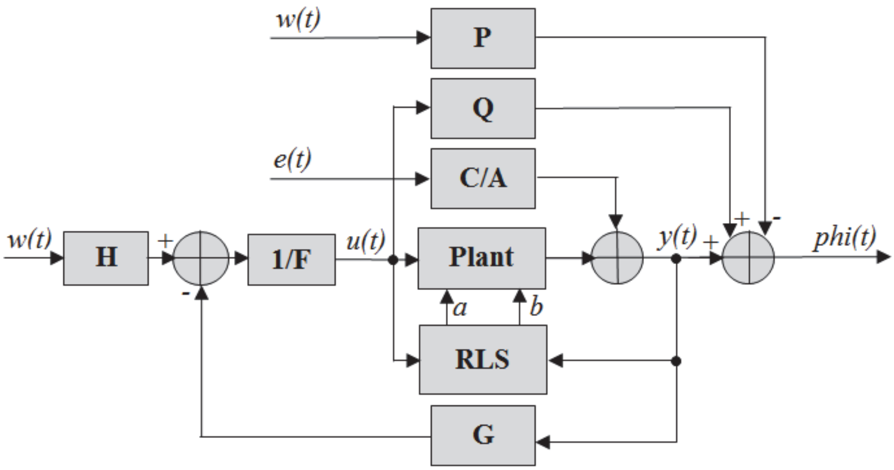

Fig. 3 – *Block diagram of indirect self-tuning GMVC.*

3 Simulation Results and Discussion

From the simulation, GMVC is introduced into the system to follow the set-point input and can estimate any change in the parameters and self-tune the control system. **Table 3** lists all the information used in the simulation.

Table 3
Parameters in Matlab Simulink.

Sampling time	0.1 s
Run time	100 s
Noise power amplitude	0.0001
Input power amplitude	10
Input	Periodic unit step
Input period	10 s
Input duty cycle	90%

3.1 Direct self-tuning GMVC

Fig. 4 shows the complete architecture of the direct GMVC model, which consists of:

1. A parameter estimator that directly estimates the controller coefficients as shown in Fig. 5. It is found that the controller coefficients change with changes in the set-point value and are also affected by changes in the plant parameters;
2. A controller which produces the desired control signal according to the set-point and the calculated controller coefficients from the parameter estimator.

The direct GMVC output response is shown in Fig. 6. It has successfully tracked a change in the set-point and is able to compensate for any change that happens in the system as shown at time $t = 10$ s, when the time constant of the system has been changed.

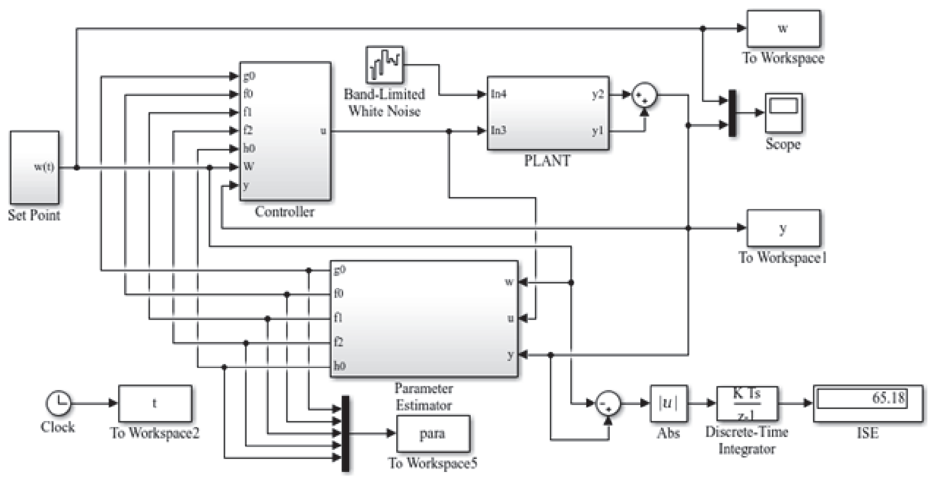

Fig. 4 – *Direct GMVC model to control time variant system.*

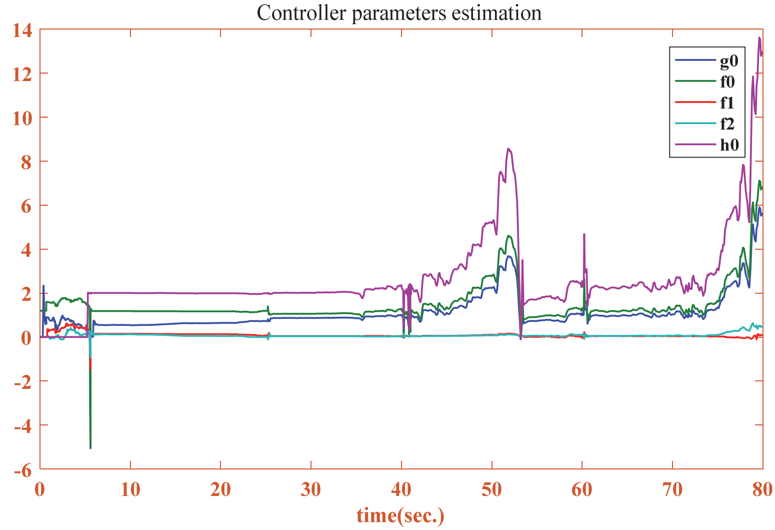

Fig. 5 – *The estimated coefficients of the controller in direct GMVC.*

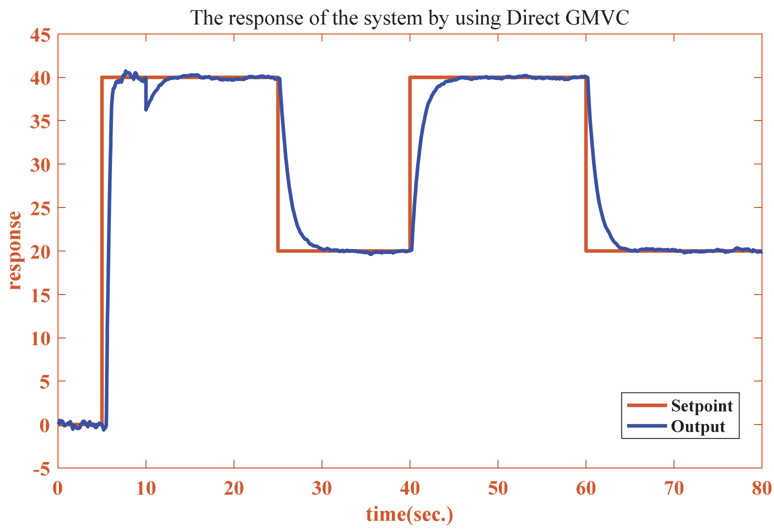

Fig. 6 – *System response for the direct GMVC control.*

In direct GMVC, the parameters estimated using RLS are h_o, f_o, f_1, and g_o, as shown in **Table 1**. These estimated values are fed into the controller to perform the correction, so the output ($y(t)$) is able to trace the set-point ($w(t)$).

3.2 Indirect self-tuning GMVC

In the indirect method, the parameters are estimated and obtained as previously stated in **Table 2** and fed into the controller to perform the correction, so the output ($y(t)$) is able to trace the set-point ($w(t)$).

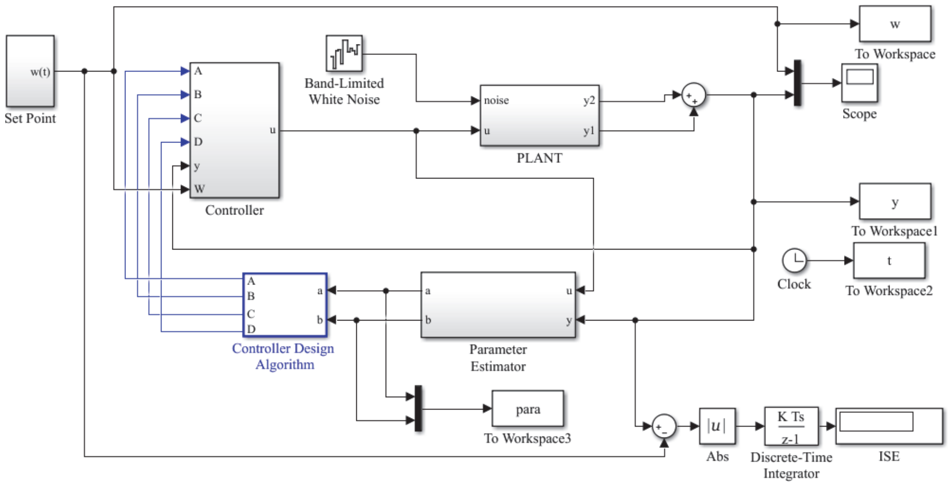

Fig. 7 – *Indirect GMVC model to control time variant system.*

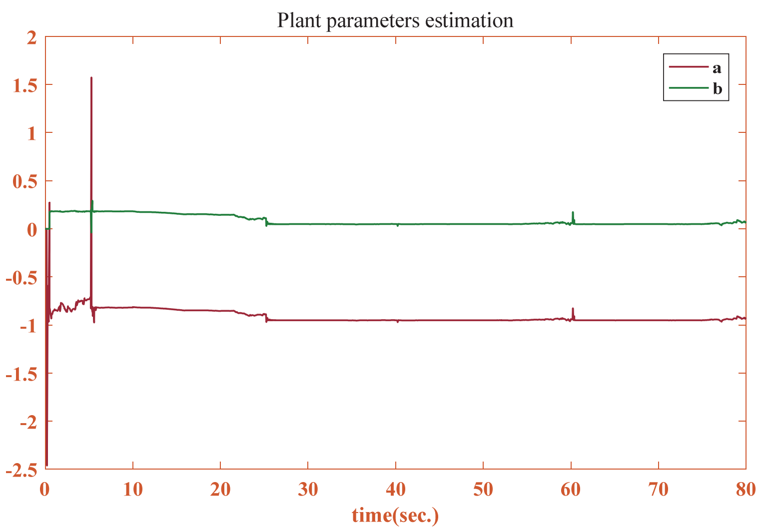

Fig. 8 – *The estimated parameters of the plant.*

Fig. 7 shows the complete architecture of the indirect GMVC model. It basically consists of:

1. A parameter estimator that estimates the parameters of a plant as shown in Fig. 8. The estimated parameters represent coefficients of the equivalent discrete transfer function. This allows the controller to track any change in the plant parameters;
2. A design algorithm that calculates the coefficients of the controller according to the estimated values of plant parameters.

The indirect GMVC algorithm response is shown in Fig. 9. It can track the change in the set-point and also shows the change in the system parameters at the time of 10 s. Indirect GMVC begins adapting itself to the new values of the plant. Finally, it can compensate for any parameter variations occurring in the control system.

The results show the output of the indirect GMVC applied in the process plant. For the first 10 s, the result of $\tau = 0.5$ is represented. After 10 s, the output represents a result of $\tau = 2$. The result shows that the indirect scheme takes longer to trace the set-point.

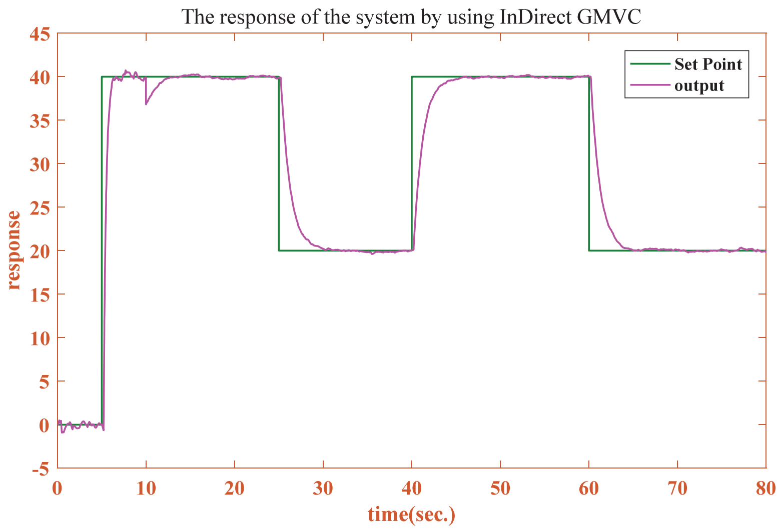

Fig. 9 – *System response after adding indirect GMVC control.*

4 Conclusion

This paper has explained the implementation of a direct and indirect generalized minimum variance algorithm to control a time variant system under different operating conditions. The results show that both schemes are able to perform their tasks successfully. Further study is suggested to explore the use of other controllers to solve such problems. The development of the direct and indirect GMVC models leads to improvements in the system response. The integral absolute error between the output and the set-point is reduced to a very low value compared to the open loop output as shown in **Table 4**. The optimal values of the estimated controller parameters for each model are obtained by using RLS. The integral absolute error of indirect GMVC is lower than that of direct GMVC. However, designing direct GMVC reduces the amount of control effort required as it combines the parameter estimation and controller design algorithm in one operation.

Table 4
Integral absolute error for the system output.

Control algorithm	Integral absolute error
Open loop	2347
Indirect GMVC	109.3
Direct GMVC	123.4

5 References

[1] K. J. Astrom, B. Wittenmark: Adaptive Control, 2nd Edition, Dover Publications, Inc., New York, USA, 2008.

[2] J. Mendes, L. Osorio, R. Araujo: Self-Tuning PID Controllers in Pursuit of Plug and Play Capacity, Control Engineering Practice, Vol. 69, December 2017, pp. 73 – 84.

[3] W. S. Black , P. Haghi, K. B. Ariyur: Adaptive Systems: History, Techniques, Problems, and Perspectives, Systems, Vol. 2, No. 4, November 2014, pp. 606 – 660.

[4] V. Bobal, J. Bohm, J. Fessl, J. Machacek: Digital Self-Tuning Controllers: Algorithms, Implementation and Applications, Springer, Germany, 2005.

[5] S. Oltean, A. Morar: Direct Digital Self-Tuning Controllers with Minimum Variance. Practical Implementation for an DC Motor, Acta Electrotehnica, Vol. 47, No. 3, 2006, pp. 168 – 173.

[6] S.- E. Oltean: Adaptive Control of Static VAR Compensator, The 6th Interdisciplinarity in Engineering International Conference (INTER-ENG 2012), Tirgu Mures, Romania, October 2012, pp. 220 – 225.

[7] T. Gang: Multivariable Adaptive Control: A Survey, Automatica, Vol. 50, No. 11, November 2014, pp. 2737 – 2764.

[8] K. Salahshoor, B. Lotfi: Adoptive Control of Well Drilling Systems, Iranian Journal of Oil & Gas Science and Technology, Vol. 2, No. 4, September 2013, pp. 26 – 38.

[9] J. Rodriguez-Resendiz, J. M. Gutierrez-Villalobos, D. Duarte-Correa, J. D. Mendiola-Santibanez, I. M. Santillan-Mendez: Design and Implementation of an Adjustable Speed Drive for Motion Control Applications, Journal of Applied Research and Technology, Vol. 10, No. 2, April 2012, pp. 180 – 194.

[10] K. J. Astrom, B. Wittenmark: On Self-Tuning Regulators, Automatica, Vol. 9, No. 2, March 1973, pp. 185 – 199.

[11] D. I. Wilson: Advanced Control Using MATLAB: Or Stabilising the Unstabilisable, 3rd Edition, Auckland University of Technology, Auckland, New Zealand, 2016.

[12] K. J. Astrom, B. Wittenmark: Self-Tuning Controllers Based on Pole-Zero Placement, IEE Proceedings D - Control Theory and Applications, Vol. 127, No. 3, May 1980, pp. 120 – 130.

[13] R. Jain, N. Sivakumaran, T. K. Radhakrishnan: Design of Self Tuning Fuzzy Controllers for Nonlinear Systems, Expert Systems with Applications, Vol. 38, No. 4, April 2011, pp. 4466 – 4476.

[14] K. A. Mohideen, G. Saravanakumar, K. Valarmathi, D. Devaraj, T. K. Radhakrishnan: Real-Coded Genetic Algorithm for System Identification and Tuning of a Modified Model Reference Adaptive Controller for a Hybrid Tank System, Applied Mathematical Modelling, Vol. 37, No. 6, March 2013, pp. 3829 – 3847.

[15] Y. S. Mohamed, B. M. Hasaneen, A. A. Elbaset, A. E. Hussein: Recursive Least Square Algorithm for Estimating Parameters of an Induction Motor, Journal of Engineering Sciences, Assiut University, Vol. 39, No. 1, January 2011, pp. 87 – 98.

[16] M. A. Khamis: Design and Simulation of Self Tuning Controller for DC Servo Motor, Diyala Journal of Engineering Sciences, Vol. 6, No. 4, December 2013, pp. 107 – 119.

[17] W. Lucia, G. Franze: Stabilization and Reference Tracking for Constrained Switching Systems: A Predictive Control Approach, International Journal of Adaptive Control and Signal Processing, Vol. 31, No. 12, December 2017, pp. 1871 – 1884.

[18] M. Elloumi, S. Kamoun: Adaptive Control Scheme for Large-Scale Interconnected Systems Described by Hammerstein Models, Asian Journal of Control, Vol. 19, No. 3, May 2017, pp. 1075 – 1088.

[19] J. Davalos-Pinto, E. Acero-Coila: Design of an Adaptive Controller with Online Identification Applied to a Pressure Tank, International Journal of Applied Science and Technology, Vol. 4, No. 2, March 2014, pp. 112 – 118.

[20] A. Y. Allidina, B. Khosrowshahi, F. M. Hughes: Implicit Pole-Assignment Self-Tuning Algorithm, International Journal of Control, Vol. 51, No. 1, 1990, pp. 51 – 70.

[21] A. Patete, K. Furuta, M. Tomizuka: Self-Tuning Control Based on Generalized Minimum Variance Criterion for Auto-Regressive Models, Automatica, Vol. 44, No. 8, August 2008, pp. 1970 – 1975.

[22] A. T. Hammid, M. Hojabri, M. H. Bin Sulaiman, A. N. Abdalla, A. A. Kadhim: Load Frequency Control for Hydropower Plants Using PID Controller, Journal of Telecommunication, Electronic and Computer Engineering, Vol. 8, No. 10, 2016, pp. 47 – 51.

[23] V. Bobal, P. Chalupa, M. Kubalčik, P. Dostal: Identification and Self-Tuning Control of Time-Delay Systems, WSEAS Transactions on Systems, Vol. 11, No. 10, October 2012, pp. 596 – 606.

[24] Advanced Textbooks in Control and Signal Processing, Edited by M. J. Grimble, M. A. Johnson, Springer, London, UK, 2013.

[25] R. V. Dukkipati: Analysis and Design of Control Systems Using MATLAB, 2nd Edition, New Age International, New Delhi, India, 2009.

[26] Table of Laplace and Z Transforms, Available at:

http://lpsa.swarthmore.edu/LaplaceZTable/LaplaceZFuncTable.html

[27] K. Gyorgy, L. David: Comparison Between Model Reference Discrete Time Indirect and Direct Adaptive Controls, Procedia Manufacturing, Vol. 22, 2018, pp. 444 – 454.

[28] Z. Y. Hitit, B. Ozyurt, S. Ertunc: The Application of System Identification and Advanced Process Control to Improve Fermentation Process of Baker's Yeast, IntechOpen – Yeast-Industrial Applications, November 2017, pp. 153 – 177.