Dokuz Eylül Üniversitesi-Mühendislik Fakültesi
Fen ve Mühendislik Dergisi
Cilt 19, Sayı 57, Eylül 2017

Dokuz Eylul University-Faculty of Engineering
Journal of Science and Engineering
Volume 19, Issue 57, September 2017

DOI: 10.21205/deufmd.2017195773

Performance of PSO Based Classical and Intelligent Controllers for Water Level Control of a Steam Generator

Emre ÖZKOP*[1], İsmail Hakkı ALTAŞ[1]

[1]Karadeniz Teknik Üniversitesi, Mühendislik Fakültesi, Elektrik-Elektronik Mühendisliği Bölümü, 61080, Trabzon

(Alınış / Received: 12.12.2016, Kabul / Accepted: 08.08.2017, Online Yayınlanma / Published Online: 20.09.2017)

Anahtar Kelimeler
Control,
Fuzzy Logic,
Particle Swarm,
Optimization,
MIMO

Abstract: In this paper, different controller techniques based on particle swarm optimization (PSO) algorithm are proposed to control the water level of a steam generator with multiple input-multiple output (MIMO) characteristics. The techniques employed are classical proportional+integral+derivative (PID) control, fuzzy logic control (FLC) and fuzzy tuned proportional-integral control (FTPIC). Gains of PID controller and parameters of FLC (the core and the boundaries of triangular membership functions in input and output spaces) are optimized by the PSO. Validations of the proposed PSO based PID control (PSO-PID), PSO based fuzzy logic control (PSO-FLC) and PSO based fuzzy tuned PI control (PSO-FTPIC) techniques are done with numerical simulation in using MATLAB. The simulation results show that the PSO-PID provides better performance for controlling the water level of a steam generator compared to the others.

Buhar Generatörünün Su Seviyesi Denetimi için PSO Temelli Klasik ve Akıllı Denetleyicilerin Performansı

Keywords
Denetim,
Bulanık Mantık,
Parçacık Sürü,
Optimizasyon,
ÇGÇÇ

Özet: Bu çalışmada çok giriş-çok çıkış (ÇGÇÇ) özelliğine sahip buhar generatörünün su seviyesi denetimi için parçacık sürü optimizasyonu (PSO) algoritmasına dayanan farklı kontrol teknikleri önerilmektedir. Bu teknikler, klasik oransal-integral-türevsel (PID) denetim, bulanık mantık denetim (BMD) ve bulanık ayarlı oransal-integral denetimdir. PID denetleyicilerin kazançları ve BMD'nin parametreleri (giriş ve çıkıştaki üçgen üyelik fonksiyonların merkezleri ve sınırları) PSO tarafından en uygun hale getirilmektedir. Önerilen PSO temelli PID denetim (PSO-PID), PSO temelli bulanık mantık denetim (PSO-BMD) ve PSO temelli bulanık ayarlı PI denetim (PSO-BAPI) tekniklerinin gerçeklemesi, MATLAB kullanılarak sayısal benzetim ile doğrulanmaktadır. Benzetim sonuçları, buhar generatörünün su seviyesi denetim için PSO-PID tekniğini diğerlerine göre daha iyi performans sergilediğini göstermektedir.

*Sorumlu yazar: eozkop@ktu.edu.tr

1. Introduction

Steam generation system is a complicated industrial process with disturbance, uncertainty and nonlinearity [1]. The steam system is a part of process of generating electric power or heating building. Accurate modelling and controlling of the steam generation system are vital and important scopes to increase the efficiency and performance in the power plants, especially while fuel costs keep rising [2]. There are many studies about modeling and control issues for the steam system in the literature [1].

The system model was designed in terms of experiments, nonlinear distributed parameter equations, artificial intelligence, neural network, neurofuzzy, stochastic fuzzy, etc [3, 4]. In the steam generation systems, main control objectives are correct air to fuel ratio, water level in the drum and steam pressure to ensure reliable, stable and efficient operation in any circumstances such as sudden load changes and disturbances.

There are studies about control of a steam generation system in the literature and each control method has both its pros and cons and superiority to other controllers depending on application aspects and cases. Model predictive control based on nonlinear, distributed and state-space approaches applied in a steam generation [5]. Pole placement control was employed in high order steam generation model and a cascade control topology with predictive aspect was used for system variables. In addition to these, multistage approach with PI controller, sliding mode control, predictive control, H2/H∞ control, combination PID controller and fuzzy logic control and also PID-controller with parameter optimization procedure were implemented [3, 6-10].

Demands on controllers are to ensure fast response, less or zero overshoot, zero steady-state error, high stability margin, robustness and provide an increase in productivity by improving quality, and reducing maintenance requirements [11]. For instance, PID controller is preferred in the most of process control applications since it works efficiently in various areas of industry and FLC utilizes quantitative and qualitative information, to trade off potentially conflicting objectives, to provide a flexible control structure, and to deal with nonlinear input/output relationships.

In this paper, the PID and FLC techniques are employed to control the water level of a steam generator. Parameters of classical (PI and PID) intelligent (FLC) controllers are optimized by PSO since the PSO algorithm can produce a higher quality solution with short computing time, accuracy, less memory size, robustness against nonlinearities, simplicity and flexibility than the other stochastic methods. This paper presents a PSO based PID controller (PSO-PID), PSO based fuzzy logic controller (PSO-FLC) and PSO based fuzzy tuned PI control (PSO-FTPIC) techniques to control the water level of a steam generator.

2. System Structure

2.1. Steam generator model

The main system structure is based on the steam generator at Abbott Power Plant in Champaign, IL shown in Figure 1. The system has multiple input-multiple output (MIMO) characteristics consisting of four inputs (fuel, air, water flow, and steam demand) and four outputs (pressure, oxygen, steam flow and level in the drum) and also there are a dual fuel (oil/gas) fired unit for heating and generating electric power [12, 13].

In the steam generation system, as fuel (u_1) is burned with air to generate heat, the water evaporates by heat creating steam. The heated steam is extracted from the upper part of the drum where water and steam are enclosed. Steam can be used in a turbine for heating buildings or to drive the generator for electricity production.

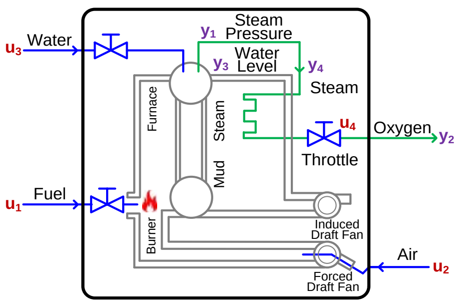

Figure 1. Basic steam generator model

The multivariable system model consisting of essential features of the actual boiler dynamics, including nonlinearities, nonminimum phase behavior, and instabilities can be described by following equations:

The parameters used in (1) to (8) are given in Table 1. The variables n_i are colored noise sequences generated by first-order models driven by zero mean, unit variance white noise.

Table 1. Parameters of the nonlinear equations of steam generator model.

$c_{11} = -0.00478$	$c_{31} = 0.00533176$	$c_{70} = -0.1048569$
$c_{12} = 0.280$	$c_{32} = 0.0251950$	$c_{71} = 0.15479$
$c_{13} = 0.01348$	$c_{33} = 0.7317058$	$c_{72} = 0.4954961$
$c_{14} = 0.02493$	$c_{41} = 0.04$	$c_{73} = -0.20797$
$c_{21} = 0.1540357$	$c_{42} = 0.0299886$	$c_{74} = 1.2720$
$c_{22} = 103.5462$	$c_{43} = 0.018088$	$c_{75} = -324212.7805$
$c_{23} = 107.4835$	$c_{51} = 14.214$	$c_{76} = -99556.24778$
$c_{24} = 1.95150$	$c_{61} = 1.00$	$c_{77} = 0.0011850$
$c_{25} = 29.04$	$c_{81} = 0.85663$	$c_{78} = -1704.50476$
$c_{26} = 1.824$	$c_{82} = -0.18128$	$c_{79} = -103.7351$
$\tau_1 = 2, \tau_2 = 2, \tau_3 = 3, \tau_4 = 3, \tau_5 = 4, \tau_6 = 10, \tau_7 = 2$		

$$\dot{x}_1(t) = c_{11}x_4(t)x_1^{9/8}(t) + c_{12}u_1(t-\tau_1) - c_{13}u_3(t-\tau_3) + c_{14} \tag{1}$$

$$\dot{x}_2(t) = -c_{21}x_2 + \frac{c_{22}u_2(t-\tau_2) - c_{23}u_1(t-\tau_1) - c_{24}u_1(t-\tau_1)x_2(t)}{c_{25}u_2(t-\tau_2) + c_{26}u_1(t-\tau_1)} \tag{2}$$

$$\dot{x}_3(t) = c_{31}x_1(t) - c_{32}x_4(t)x_1(t) - c_{33}u_3(t-\tau_3) \tag{3}$$

$$\dot{x}_4(t) = -c_{41}x_4(t) + c_{42}u_1(t-\tau_1) + c_{43} + u_4(t) + n_5 \tag{4}$$

$$y_1(t) = c_{51}x_1(t-\tau_4) + n_1(t) \tag{5}$$

$$y_2(t) = c_{61}x_2(t-\tau_5) + n_2(t) \tag{6}$$

$$\begin{aligned} y_3(t) = {} & c_{70}x_1(t-\tau_6) + c_{71}x_3(t-\tau_6) + c_{72}x_4(t-\tau_6) + c_{73}u_3(t-\tau_3-\tau_6) \\ & + c_{74}u_1(t-\tau_1-\tau_6) + \frac{[c_{75}x_1(t-\tau_6) + c_{76}][1 - c_{77}x_3(t-\tau_6)]}{x_3(t-\tau_6)[x_1(t-\tau_6) + c_{78}]} \\ & + c_{79} + n_3(t) \end{aligned} \tag{7}$$

$$y_4(t) = [c_{81}x_4(t-\tau_7) + c_{82}]x_1(t-\tau_7) + n_4(t) \tag{8}$$

where x_1 is drum pressure state (kgf/cm^2); y_1 is measured drum pressure (psi); y_2 and x_2 are measured excess oxygen level and its state, respectively (%); x_3 is system fluid's density (kg/m^3); y_3 is drum water level ($in.$); y_4 is steam flow rate (kg/s); u_1, u_2, u_3 are fuel, air, and feed water flow inputs, which take values between 0 and 1; x_4 is exogenous variable related to the steam demand.

The linearized model can be defined by

$$\dot{x} = Ax + Bu \qquad\qquad y = Cx + Du$$

where

$$A = \begin{bmatrix} -0.005509 & 0 & 0 & -0.1588 \\ 0 & -0.2062 & 0 & 0 \\ -0.01216 & 0 & 0 & -0.5672 \\ 0 & 0 & 0 & -0.040 \end{bmatrix} \quad B = \begin{bmatrix} 0.2800 & 0 & -0.01348 & 0 \\ -9.375 & 7.658 & 0 & 0 \\ 0 & 0 & 0.7317 & 0 \\ 0.02999 & 0 & 0 & 0.040 \end{bmatrix} \tag{9}$$

$$C = \begin{bmatrix} 14.21 & 0 & 0 & 0 \\ 0 & 1.0 & 0 & 0 \\ 0.3221 & 0 & 0.1434 & 11.16 \\ 0.4133 & 0 & 0 & 19.28 \end{bmatrix} \qquad D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1.272 & 0 & -0.2080 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{10}$$

2.2. Control model

A steam generation plant can be worked properly under control requirements are provided. These requirements can be defined as maintaining of steam pressure, water in the drum and mixture of fuel and air in the camber at desired levels and standards. Overheating of drum components or flooding of steam lines can be prevented by the water level control in the drum.

In this paper, the control structure is designed to manage the water level in the steam generation plant. The system control is tested by applying three different types of controller structures. First a classical PID control, then a classical fuzzy logic control (FLC) and then a fuzzy tuned PI control (FTPIC) is applied. The performances of these controllers are compared for better utilization.

The parameters of the PID controller (K_P, K_I and K_D) are optimized by PSO to improve the response of the controller in this study. The general block diagram of system with PSO based PID controller (PSO-PID) is shown in Figure 2.

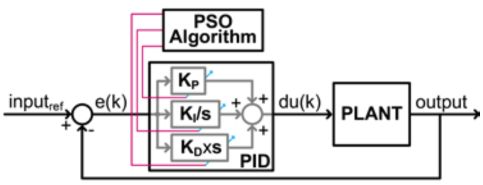

Figure 2. Structure of PSO based PID controller (PSO-PID)

The FLC consists of fuzzification, rule base and defuzzification parts. All membership functions for both input and output spaces are triangular types since triangle-shaped fuzzy membership functions are modeled easily due to their linearity and they require less time and memory in control algorithms [14].

The FLC has two inputs named as error (e) and error deviation (de) and one output (du). Five triangular type membership functions called positive big (PB), positive small (PS), zero (ZZ), negative small (NS), and negative big (NB) are used in input and output spaces of the FLC. The membership functions for two inputs (e and de) and an output (du) are shown in Figure 3. Classical particle swarm optimization with inertia weighting approach (CPSO-IWA) is used to optimize limits of the membership functions.

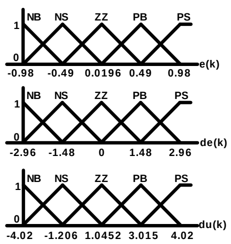

Figure 3. The membership functions of the PSO-FLC

These membership functions are employed to convert the input signals (e and de) to fuzzy subsets in the fuzzification stage. The fuzzified values of the inputs are applied in the rule table given in Table 2 to get the fuzzy number.

Table 2. Fuzzy logic rules decision table for FLC.

		de				
		NB	**NS**	**ZZ**	**PS**	**PB**
	NB	NB	NB	NS	NS	ZZ
	NS	NB	NS	NS	ZZ	PS
	ZZ	NS	NS	ZZ	PS	PS
	PS	NS	ZZ	PS	PS	PB
	PB	ZZ	PS	PS	PB	PB

There are no generally accepted methods or standards for a rule table design [15]. Therefore, techniques based on the direct knowledge from experts and the knowledge from numerical data are preferred in literature [15, 16]. In this paper, a symmetrical rule table constituted by system response approach given in [17] is used. The resultant united fuzzy subsets are converted to the crisp values at defuzzification stage. The general block diagram of system with PSO based fuzzy logic controller (PSO-FLC) is shown in Figure 4.

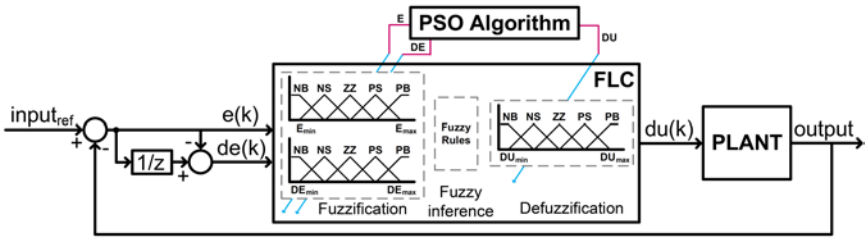

Figure 4. Structure of PSO based fuzzy logic controller (PSO-FLC)

Third controller type is the PSO based fuzzy tuned PI control (PSO-FTPIC). The classical PI controller gains (K_P and K_I) are simultaneously tuned by the FLC within determined limits ($K_{P(min)}$, $K_{P(max)}$, $K_{I(min)}$, $K_{I(max)}$) as the system operates. The limits of PI controller gains are settled by PSO algorithm. The membership functions of inputs (e and de) and an output (du) for the controller gains are shown in Figure 5.

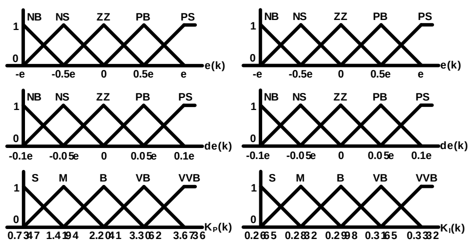

Figure 5. The membership functions of the PSO-FTPIC

The fuzzy subsets very very big (VVB), very big (VB), big (B), medium (M) and small (S) are used for the output space of the FLC. The related rule table is same as in PSO-FLC algorithm given in Table 2. The general block diagram of system with PSO-FTPIC is shown in Figure 6.

2.3. Performance indices

A measurement of control system performance is important to improve the system responses. A quantitative measure in terms of performance indices can be realized and used to compare and evaluate the system's performances. The index must be minimized to develop the system performance, such as minimizing the steady state error, rise time, maximum overshoot and settling time.

Performance indices are very useful to analyze and design control systems. The most common performance indices are the integral of the square of the error (ISE), integral of absolute magnitude of the error (IAE) and integral of time multiplied by absolute error (ITAE) defined as:

$$ISE = \int_0^T e^2(t)dt\,, \quad IAE = \int_0^T |e(t)|dt\,, \quad ITAE = \int_0^T t|e(t)|dt \tag{11}$$

where $e(t)$ is the error signal in the time domain.

Performances of control strategies are observed in systems with different orders and characteristics. To illustrate the effectiveness of the techniques, simulation models are developed in Matlab/Simulink/Simpower Software Environment.

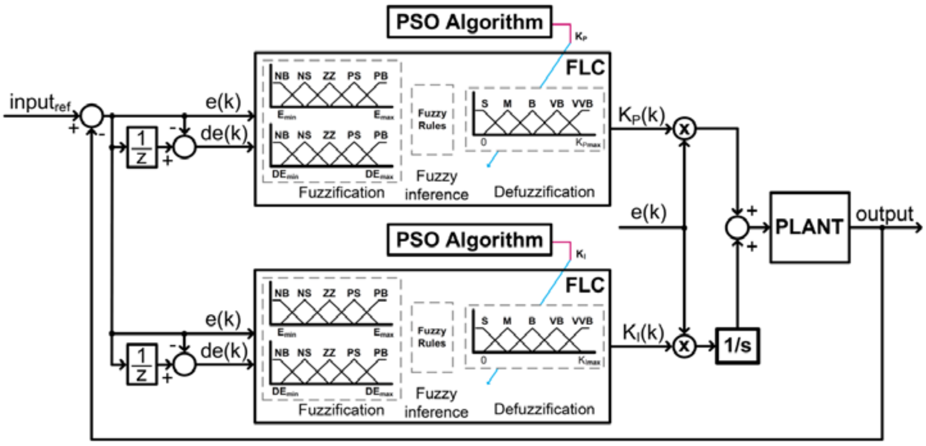

Figure 6. Structure of PSO based fuzzy tuned PI control (PSO-FTPIC)

2.4. Particle swarm optimization

Particle swarm optimization (PSO) is one of computational intelligence-based techniques, which can be used to figure out the approximate solutions to engineering optimization problems such that success rate of PSO based on the social behavior of bird flocking and fish schooling is not majorly affected by problem features such as size and nonlinearity level [18].

In this paper, the CPSO-IWA is used to optimize the parameters of controllers. In the CPSO-IWA, the swarm consists of particles, each of which based on components such that the velocity of the related component can be defined as given in Equations (12) and (13) for the ith component of jth particle [19]. The CPSO-IWA initialization parameters are given in Table 3.

$$\begin{aligned} v_{j,i}(iter) = {} & (w_{max} - (\frac{w_{max} - w_{min}}{N_m}) \times iter) \times v_{j,i}(iter-1) \\ & + c_1R_1\left[P_{j,i}(iter-1) - x_{j,i}(iter-1)\right] + c_2R_2[P_j^*(iter-1) \\ & - x_{j,i}(iter-1)] \end{aligned} \tag{12}$$

$$x_{j,i}(iter) = x_{j,i}(iter-1) + v_{j,i}(iter) \tag{13}$$

where w_{max} and w_{min} are maximum and minimum inertia weights, N_m is maximum number of iteration cycles, $v_{j,i}$ velocity, c_1 and c_2 are social and cognitive rate constants, R_1 and R_2 are uniformly distributed random numbers in [0,1], $P_{j,i}$ is local best position, P_j^* is global best position and $x_{j,i}$ is position of ith component of jth particle.

Table 3. The CPSO-IWA parameters.

Parameter	**Value**
Swarm size (S)	20
Maximum number of iteration cycles (N_m)	20
Maximum inertia weight (w_{max})	0.9
Minimum inertia weight (w_{min})	0.4
Social rate (c_1)	2
Cognitive rate (c_2)	2

There are different strategies to set inertia weight such as fixed inertia weight, fuzzy adaptive, linearly decreasing, linearly increasing, non-linear, chaotic, etc [20]. In this study, the inertia weight is linearly decreased from $w_{max} = 0.9$ to $w_{min} = 0.4$. The swarm size can be determined according to complexity of problems so that there is a suggestion to choose swarm size between 20 and 50 in the most studies [21, 22].

Moreover, it is assumed that more efficient results can be obtained by selection of a larger swarm size for higher dimensional problems [23]. On the other hand, increase in swarm size affects positively the performance of the algorithm, but a larger swarm size requires more iterations so that more computational load and cost occur. The values $c_1 = 2$ and $c_2 = 2$ used in this study are widely accepted settings used in most of problems in literature [20]. Swarm topologies vary in literature and prominent topologies are shown in Figure 7 [24]. Global best topology is employed in this study. The integral of the square of the error (ISE) is used as an objective function.

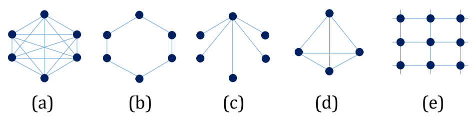

Figure 7. Swarm topologies (a) Global best, (b) Ring, (c) Wheel, (d) Pyramid, (e) Von Neumann

The all controller parameters optimized by particle swarm optimization algorithm are tabulated in Table 4.

Table 4. The parameters of all optimized controllers.

<table>
<tr><th colspan="4">PSO-PID</th></tr>
<tr><th>K_P</th><th colspan="2">K_I</th><th>K_D</th></tr>
<tr><td>3.1254</td><td colspan="2">0.2406</td><td>0.084</td></tr>
<tr><th colspan="4">PSO-FLC</th></tr>
<tr><th>E_{max} $(-E_{min})$</th><th>DE_{max} $(-DE_{min})$</th><th colspan="2">DU_{max} $(-DU_{min})$</th></tr>
<tr><td>0.98</td><td>2.96</td><td colspan="2">4.02</td></tr>
<tr><th colspan="4">PSO-FTPIC</th></tr>
<tr><th>E_{max} $(-E_{min})$</th><th>DE_{max} $(-DE_{min})$</th><th>K_{Pmax}</th><th>K_{Imax}</th></tr>
<tr><td>e(k) (adaptive)</td><td>$0.1 \times$ e(k) (adaptive)</td><td>3.6736</td><td>0.3332</td></tr>
</table>

3. Simulation Results

The steam generator system is simulated by using three control strategies for comparison and validation purposes. Fuel rate and water level as a reference variations in time are shown in Figures 8 and 9, respectively. The fuel rate rises up from 0.27 pu to 1 pu at t = 1500s. Next, after 1500s, it falls to 0.27pu (Figure 8). The water level is zero level between 0s and 1500s and then it has 1 inch value at the time interval t = [1500, 3000]. Finally, at time t = 3000s, it decreases to 0 inch (Figure 9).

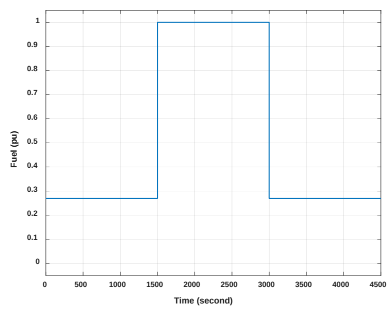

Figure 8. Steam generator fuel rate variation

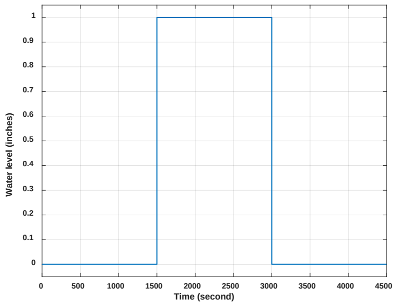

Figure 9. Steam generator water level reference

The controlled system output (water level) related to the controller techniques are shown in Figures 10 and 11. The maximum overshoots, settling times for 10%band, rise and peak times and peak values are given in Table 5. The effectiveness of controller scenarios is also tested by the performance indices (ISE, IAE and ITAE) and tabulated in Table 6.

The water level overshoots of the PSO-PID, the PSO-FLC and the PSO-FTPIC are 87%, 177% and 113%, respectively. The PSO-PID controller has better performance than the others in overshoots, settling times, peak values and times. As for the rise times of the water level output, the PSO-FLC was found to be 0.70 s, as the PSO-FTPIC and the PSO-PID were found to be 0.86 and 0.88 s, respectively.

The PSO-PID has exhibited lower ISE, IAE and ITAE values compared with the PSO-FLC and PSO-FTPIC methods as given in Table 6, that the PSO-PID control strategy performs much better than the PSO-FLC and PSO-FTPIC strategies.

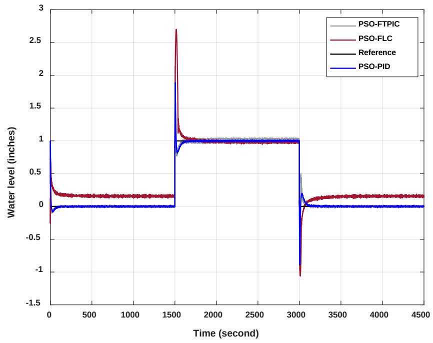

Figure 10. Steam generator water level with all controllers

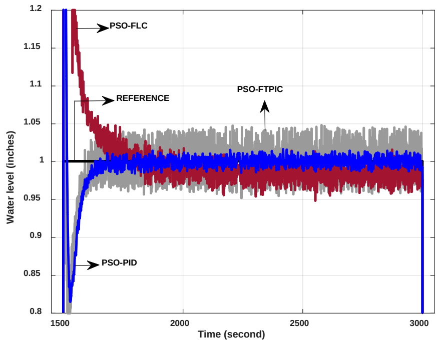

Figure 11. Zoomed view of water level output with all controllers for 20% band

Table 5. System performances for the controllers

	PSO-PID	PSO-FLC	PSO-FTPIC
Overshoot (%)	87	177	113
Settling time for 10% band (s)	787	795	1236
Rise time (s)	0.88	0.70	0.86
Peak value (inches)	1.89	2.70	2.14
Peak time (s)	225	336	799

Table 6. System controller performance indexes comparison.

Controller Type	ISE	IAE	ITAE
PSO-PID	13.02	58.06	119019
PSO-FLC	164.82	564.86	1188874
PSO-FTPIC	16.44	75.63	162752

4. Conclusion

Performance analysis and comparison of PSO optimized control algorithms used to control water level in a steam generator are studied in this paper. A PSO based PID controller (PSO-PID), a PSO based fuzzy logic controller (PSO-FLC) and a PSO based fuzzy tuned PI control (PSO-FTPIC) are proposed in this work and applied to control a steam generator system having multiple input-multiple output (MIMO) characteristics. The digital simulation results show that the PSO-PID control strategy performed better than the other control strategies.

References

[1] Liu, X.J., Lara-Rosano, F., Chan, C.W. 2003. Neurofuzzy Network Modelling and Control of Steam Pressure in 300 MW Steam-Boiler System, Engineering Applications of Artificial Intelligence, Vol. 16, No. 5-6, pp. 431-440. DOI: 10.1016/ j.engappai.2003.08.006

[2] Klauco, M., Kvasnica, M. 2017. Control of A Boiler-Turbine Unit Using MPC-based Reference Governors, Applied Thermal Engineering, Vol. 110, pp. 1437-1447. DOI: 10.1016/j.applthermaleng.2016.09.041

[3] Liu, X.J., Chan, C.W. 2006. Neuro-Fuzzy Generalized Predictive Control of Boiler Steam Temperature, IEEE Transactions on Energy Conversion, Vol. 21, No. 4, pp. 900-908. DOI: 10.1109/TEC.2005. 853758

[4] Ku, C.C. 2015. Robust Controller Design for Nonlinear Uncertain Stochastic Drum-Boiler System. 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD), 15-17 August, Zhangjiajie, China, 2378-2383.

[5] Niva, L., Yli-Korpela, A. 2012. Control of A Benchmark Boiler Process Model with DMC and QDMC, IFAC Proceedings, Vol. 45, No. 21, pp. 190-195. DOI: 10.3182/20120902-4-FR-2032.00035

[6] Lawrynczuk, M. 2017. Nonlinear Predictive Control of A Boiler-Turbine Unit: A State-Space Approach with Successive On-Line Model Linearisation and Quadratic Optimisation, ISA Transactions, Vol. 67, pp. 476-495. DOI: 10.1016/ j.isatra.2017.01.016

[7] Wei, L., Fang, F. 2017. H_∞-LQR-Based Coordinated Control for Large Coal-Fired Boiler-Turbine Generation Units. IEEE Transactions on Industrial Electronics, Vol. 64, No. 6, pp. 5212-5221. DOI: 10.1109/TIE. 2016.2622233

[8] Moradi, H., Abbasi, M.H., Moradian, H. 2016. Improving The Performance of a Nonlinear Boiler-Turbine Unit Via Bifurcation Control of External Disturbances: A Comparison Between Sliding Mode and Feedback Linearization Control Approaches, Nonlinear Dynamics, Vol. 85, No. 1, pp. 229-243. DOI: 10.1007/s11071-016-2680-x

[9] Zhang F., Wu, X., Shen, J. 2017. Extended State Observer Based Fuzzy Model Predictive Control for Ultra-Supercritical Boiler-Turbine Unit, Applied Thermal Engineering, Vol. 118, pp. 90-100. DOI: 10.1016/ j.applthermaleng.2017.01.111

[10] Alamoodi, N., Daoutidis, P. 2017. Nonlinear Control of Coal-Fired Steam Power Plants, Control Engineering Practice, Vol. 60, pp. 63-75. DOI: 0.1016/j.conengprac.2016. 12.005

[11] González, J. A., Barreiro, A., Dormido, S., Baños, A. 2017. Nonlinear Adaptive Sliding Mode Control With Fast Non-Overshooting Responses And Chattering Avoidance, Journal of the Franklin Institute, Vol. 354,

No. 7, pp. 2788-2815. DOI: 10.1016/ j.jfranklin.2017.01.025

[12] Ait Sahed, O., Kara, K., Benyoucef, A., Hadjili, M.L. 2016. An Efficient Artificial Bee Colony Algorithm with Application to Nonlinear Predictive Control, International Journal of General Systems, Vol. 45, No. 4, pp. 393-417. DOI: 10.1080/03081079. 2015.1086344

[13] Jamali, B., Jazayeri-Rad, H. 2010. Application of Adaptive Local Linear Model Tree for Nonlinear Identification of Heat Recovery Steam Generator System Based on Experimental Data. 2010 Fourth UKSim European Symposium on Computer Modeling and Simulation, 17-19 November, Pisa, Italy, 16-20.

[14] Ozkop E., Sharaf A., Altas I.H. 2016. An Adaptive Fuzzy PI Controlled Bus Quantity Enhancer for Wave Energy Systems, Turkish Journal of Electrical Engineering and Computer Sciences, Vol. 24, pp. 2454-2468. DOI: 10.3906/elk-1312-198

[15] Castro, P.A.D., Camargo, H.A. 2004. Learning and Optimization of Fuzzy Rule Base by Means of Self-Adaptive Genetic Algorithm, IEEE International Conference on Fuzzy Systems; 25-29 July, Budapest, Hungary. New York, NY, USA

[16] Li, H.X., Gatland, H.B. 1995. New Methodology for Designing A Fuzzy Logic Controller, IEEE T Syst Man Cy, Vol. 25, pp. 505-512

[17] Altas, I.H., Sharaf, A.M. 2007. A Generalized Direct Approach for Designing Fuzzy Logic Controllers in MATLAB/ Simulink GUI Environment, International Journal of Information Technology and Intelligent Computing, Vol. 1, No. 4, pp. 1-27

[18] Liu ,J., Ma, D., Ma, T.B, Zhang, W. 2017. Ecosystem Particle Swarm Optimization, Soft Computing, Vol. 21, No. 7, pp. 1667-1691. DOI: 10.1007/s00500-016-2111-4

[19] Yang, Q.H., Tian, J.P., Si, W. 2017. An Improved Particle Swarm Optimization Based on Difference Equation Analysis, Journal of Difference Equations and Applications, Vol. 23, No. 1-2, pp. 135-152. DOI: 10.1080/10236198. 2016.1199691

[20] Cui, H.Q., Shu, M.L., Song, M., Wang, Y.L. 2017. Parameter Selection and Performance Comparison of Particle Swarm Optimization in Sensor Networks Localization, Sensors, Vol. 17, No. 3, pp. 1-18. DOI: 10.3390/ s17030487

[21] Bouallègue, S., Haggège, J., Ayadi, M., Benrejeb, M. 2012. PID-Type Fuzzy Logic Controller Tuning Based on Particle Swarm Optimization, Engineering Applications of Artificial Intelligence, Vol. 25, No. 3, pp. 484-493. DOI: 10.1016/j.engappai.2011. 09.018

[22] Khare, A., Rangnekar, S. 2013. A Review of Particle Swarm Optimization and Its Applications in Solar Photovoltaic System, Applied Soft Computing, Vol. 13, No. 5, pp. 2997-3006. DOI: 10.1016/j.asoc. 2012.11.033

[23] Coban, R. 2011. A Fuzzy Controller Design for Nuclear Research Reactors Using The Particle Swarm Optimization Algorithm, Nuclear Engineering and Design, Vol. 241, No. 5, pp. 1899-1908. DOI: 10.1016/ j.nucengdes.2011.01.045

[24] Del Valle, Y., Venayagamoorthy, G.K., Mohagheghi, S., Hernandez, J.C., Harley, R.G. 2008. Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems, IEEE Transactions on Evolutionary Computation, Vol. 12, No. 2, pp. 171-195. DOI: 10.1109/ TEVC.2007.896686