A NEW BIOGEOGRAPHY-BASED OPTIMIZATION APPROACH BASED ON SHANNON-WIENER DIVERSITY INDEX TO PID TUNING IN MULTIVARIABLE SYSTEM

Marsil de Athayde Costa e Silva, marsil@ymail.com
Undergraduate course of Mechatronics Engineering

Camila da Costa Silveira, Camila.Silveira@br.bosch.com
Leandro dos Santos Coelho, leandro.coelho@pucpr.br
Industrial and Systems Engineering Graduate Program, PPGEPS
Pontifical Catholic University of Parana, PUCPR
Imaculada Conceição, 1155, Zip code 80215-901, Curitiba, Parana, Brazil

Abstract. *Proportional-integral-derivative (PID) control is the most popular control architecture used in industrial problems. Many techniques have been proposed to tune the gains for the PID controller. Over the last few years, as an alternative to the conventional mathematical approaches, modern metaheuristics, such as evolutionary computation and swarm intelligence paradigms, have been given much attention by many researchers due to their ability to find good solutions in PID tuning. As a modern metaheuristic method, Biogeography-based optimization (BBO) is a generalization of biogeography to evolutionary algorithm inspired on the mathematical model of organism distribution in biological systems. BBO is an evolutionary process that achieves information sharing by biogeography-based migration operators. This paper proposes a modification for the BBO using a diversity index, called Shannon-wiener index (SW-BBO), for tune the gains of the PID controller in a multivariable system. Results show that the proposed SW-BBO approach is efficient to obtain high quality solutions in PID tuning.*

Keywords*: PID control, biogeography-based optimization, Shannon-Wiener index, multivariable systems.*

1. INTRODUCTION

One of the most popular controllers in industrial processes is the proportional-integral-derivative (PID) controller. This control strategy offers a simple and effective solution for many real problems. About 90% of the control problems are solved by using some type of PID controller (Levine, 1996). After its creation, around 1910, and the Ziegler-Nichols tuning methods (Ziegler and Nichols, 1942) the popularity of this kind of controller has grown. This is mainly because PID controllers have structure simplicity and meaning of the corresponding three parameters, which can be easily understood by process operators. Moreover, PID controllers have the advantage of good stability and high reliability.

The use of evolutionary algorithms to tune gains of PID controllers has demonstrated ability of finding a set of good solutions (Iruthayarajan and Baskar, 2009). The evolutionary computation paradigms such as genetic algorithm (Altinten et al., 2008), differential evolution (Lianghong et al., 2008), evolution strategies (Iruthayarajan and Baskar, 2010), and evolutionary programming (Jiang and Ma, 2006) are able to find a reasonable solutions for problems in which classical methods based on gradient information cannot be applied or do not show good performance. Examples in control systems are presented in Fleming and Purshouse (2002). A recent approach called Biogeography-based Optimization (BBO) has shown promising results in solving of complex optimization problems (Simon, 2008). Biogeography is the science that studies the distribution of species in an ecosystem and how species arise or become extinct. The main contribution of this paper is validate a new of BBO approach that uses a diversity measurement to increase the capability of scape from local optima. In this work, the classical BBO and the proposed BBO based on diversity measurement are used to find the gains of a multivariable PID controller for a 2x2 industrial-scale polymerization reactor.

The remainder of this paper is organized as follows: in section 2 are presented the basic concepts of PID control. Section 3 describes the BBO algorithm and the proposed approach. The formulation of the problem is detailed in section 4. Sections 5 and 6 present the results and conclusion, respectively.

2. PID CONTROL FOR MULTIVARIABLE SYSTEMS

Consider the multivariable system (Multiple Inputs Multiple Outputs, MIMO) shown in Figure 1, where $R(t)$ is the set of reference signals, $Y(t)$ is the set of outputs and $U(t)$ is the set of control signals. The error, $e(t)$, is the difference between the output and the input signals. The control signals are calculated based on the error. The standard PID controller is described by equation (1).

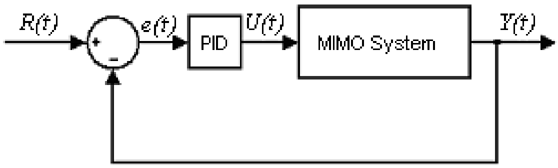

Figure 1. MIMO system with the PID control.

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{de(t)}{dt} \tag{1}$$

where K_p, K_i and K_d are the proportional, integral, and derivative gains of the PID, respectively, and t is the time. Also, the integral and the derivative gains can be written as a function of the proportional gain: $K_i=K_p/t_i$ and $K_d=K_pt_d$, where t_i and t_d are the integral and derivative time. The Laplace transform can be applied to the controller to give the following transfer function:

$$G(s) = K_p\left(1+\frac{1}{t_i s}+t_d s\right) \tag{2}$$

where $G(s)$ is the transfer function of the controller and the error is the input and the output is the control signal. Nevertheless, the derivative term of the controller can amplify some noisy signal and also causes a sudden elevation of the control signal when the set point changes. So a filter is applied to the derivative term of the controller to avoid these problems, thus the transfer function of the controller becomes the following:

$$G(s) = K_p\left(1+\frac{1}{t_i s}+\frac{t_d s}{\frac{t_d s}{N}+1}\right) \tag{3}$$

where N is the filtering constant, normally used as a number between 4 and 20.

For an nxn multivariable system $H(s)$, equation (4), the controller becomes an nxn matrix as given by (5).

$$H(s) = \begin{bmatrix} h_{11}(s) & \cdots & h_{n1}(s) \\ \vdots & \ddots & \vdots \\ h_{1n}(s) & \cdots & h_{nn}(s) \end{bmatrix} \tag{4}$$

$$G(s) = \begin{bmatrix} g_{11}(s) & \cdots & g_{n1}(s) \\ \vdots & \ddots & \vdots \\ g_{1n}(s) & \cdots & g_{nn}(s) \end{bmatrix} \tag{5}$$

where

$$g_{ij}(s) = K_{p_{ij}}\left(1+\frac{1}{t_{i_{ij}} s}+\frac{t_{d_{ij}} s}{\frac{t_{d_{ij}} s}{N}+1}\right) \tag{6}$$

In order to measure the performance of the controller four main kinds of performance criteria are usually considered: the integrated squared error (ISE), the integrated absolute error (IAE), the integrated time-weighted absolute error (ITAE) and the integrated time-weighted squared error (ITSE). These criteria are defined by the following equations:

$$ISE = \int_0^\infty \left(e_1(t)^2 + e_2(t)^2 + \ldots + e_n(t)^2\right)dt \tag{7}$$

$$IAE = \int_0^{\infty} \left(|e_1(t)| + |e_2(t)| + \ldots + |e_n(t)| \right) dt \tag{8}$$

$$ITAE = \int_0^{\infty} \left(t|e_1(t)| + t|e_2(t)| + \ldots + t|e_n(t)| \right) dt \tag{9}$$

$$ITSE = \int_0^{\infty} \left(te_1(t)^2 + te_2(t)^2 + \ldots + te_n(t)^2 \right) dt \tag{10}$$

where e_i is the error of the *i*-th output related to the *i*-th input.

3. BIOGEOGRAPHY-BASED OPTIMIZATION

The BBO algorithm, proposed by Simon (2008), uses the concepts and models of biogeography. Furthermore, BBO approaches have demonstrated ability to solve and good convergence properties on various benchmark functions and engineering optimization problems (Rarick et al., 2009; Kumar et al., 2009; Kundra et al., 2009; Simon et al., 2009; Bhattacharya and Chattopadhyay, 2010; Gong et al., 2010).

These models of biogeography describe how species migrate from a habitat to another one and how species arise or become extinct. Each solution used in the algorithm is considered as a habitat and has a habitat suitability index (HSI) that measure the suitability of the habitat. This index is related to aspects as, for example, rainfall, fauna and flora diversities, topography, and environment temperature. These aspects are also called suitability index variables (SIV).

A good habitat has a high HSI, while a poor habitat has a low HSI. This means that good habitats have more good aspects than the poor ones. Habitats with high HSI have a high immigration rate due to their good aspects, whereas poor habitats have a low immigration rate but a high emigration rate unlike good ones. The migration rates are direct related to the number of species in a habitat. So, a habitat with many species has a high emigration rate, because it is almost saturated, while habitats with few species have high immigration rate because do not have good conditions to live in. This migration process increases the diversity of the habitat and the miscegenation and contributes to the species information sharing and the mutation probability. Figure 2 represents emigration and immigration as a function of the number of species. In the Figure 2, *I* and *E* represent the maximum rates of immigration and emigration, respectively, and *S* denotes the number of species.

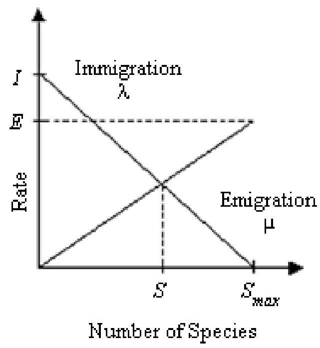

Figure 2. Emigration and immigration rates.

These concepts inspired the proposition of BBO. In the algorithm the solutions are treated as habitats and their good aspects are shared based on the migration rates. The basic algorithm of BBO is described in the following lines.

Step 1: Initialize the parameters used in the algorithm: $S_{\max}$ maximum number of species, E emigration rate, I the immigration rate, and $m_{\max}$ the maximum mutation rate.

Step 2: Calculate the probability for each value of the number of species as follows:

$$P_j = \frac{1}{S_{\max}} \tag{11}$$

where $j = 1,...,S_{\max}$, and P is the probability for the *j*-th habitat.

Step 3: Generate an initial random set of habitats according to the constraints of the problem.

Step 4: Start the loop:

(4.i) Generate the immigration and emigration rates:

$$\lambda_j = \frac{I(1-j)}{S_{\max}} \tag{12}$$

$$\mu_j = E\frac{j}{S_{\max}} \tag{13}$$

where λ_j and μ_j are the immigration and the emigration rates for the *j*-th habitat.

(4.ii) Calculate the derivative probability:

$$\begin{cases} \dot{P}_s = -(\lambda_s + \mu_s)P_s + \mu_{s+1}P_{s+1} & s = 0 \\ \dot{P}_s = -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} + \mu_{s+1}P_{s+1} & 1 \le s < S_{\max} \\ \dot{P}_s = -(\lambda_s + \mu_s)P_s + \lambda_{s-1}P_{s-1} & s = S_{\max} \end{cases} \tag{14}$$

(4. iii) Update the probability:

$$P_j = P_j + \dot{P}_j\, dt \tag{15}$$

$$P_j = \frac{P_j}{\sum_{i=0}^{S_{\max}} Pi} \tag{16}$$

where *dt* is the derivative step.

(4.iv) Use the immigration and emigration rates to modify each habitat and probabilistically mutate the individuals.

(4.v) Evaluate the habitats to make sure that the constraints of the problem are satisfied.

(4.vi) Calculate the fitness of each habitat and return to the beginning of the loop until a stopping criterion is achieved.

3.1 BBO based on Shannon-Wiener (SW-BBO)

The proposed SW-BBO approach uses a diversity index widely used to compute biodiversity in an ecosystem. The index is called Shannon-wiener index (SWI) and is calculated as follows:

$$H = -\sum_{i=1}^{S} p_i log(p_i) \tag{17}$$

where H is the diversity measure, p_i is the relative abundance of the specie i , and S is the number of species.

This index is used to calculate the mutation for each habitat. To calculate the relative abundance of species a simple method is used: divide the search space into S (number of species) sub-divisions and then count the number of species in each habitat. For example: suppose a problem with 10 variables lying between 0 and 1, we want to divide the search space into 4 levels (this is the number of species and is a user defined parameter), so each variable, depending on its value, will be a certain specie: specie 1 if it is between 0 and 0.25, specie 2 if it is between 0.25 and 0.5 and so on. The number of variables in each level is the species count of each kind of specie. Table 1 shows a habitat generated randomly with its species counts.

Table 1. Example of habitat.

Value	Specie type
0.82	4
0.91	4
0.13	1
0.92	4
0.63	3
0.09	1
0.28	2
0.55	3
0.96	4
0.96	4

Based on the data of Table 1, the Shannon-wiener index is calculated as follows:

$$H = -\left[p_1 log(p_1) + p_2 log(p_2) + p_3 log(p_3) + p_4 log(p_4)\right] \tag{18}$$

$$H = -\frac{1}{10}\left[2log\left(\frac{2}{10}\right) + log\left(\frac{1}{10}\right) + 2log\left(\frac{2}{10}\right) + 5log\left(\frac{5}{10}\right)\right] \tag{19}$$

$$H = 1.2206 \tag{20}$$

The SWI in the proposed approach is used to calculate the mutation for each habitat (solution) as follows:

$$m_i = \left(1 - \frac{H_i}{4}\right) \tag{21}$$

where m_i is the mutation probability of the i-th habitat and H_i is the SWI for the i-th habitat. Note that the value 4 is used to fit the SWI between 0 and 1, but a value near to 4 or near to 0 are rarely achieved.

4. FORMULATION OF THE OPTIMIZATION PROBLEM

The problem is to find a configuration of the gains of the PID controllers that minimizes the objective function. The system to be controlled is an industrial-scale polymerization reactor. The time scales are in hours, so the process dynamic response is very slow. The two controlled variables are two measurements representing the reactor condition, and the two manipulated variables are the references of two reactors feed flow loops with load disturbance as the purge flow of the reactor (Chien, 1999). The system dynamics is modeled by equation (22) given by

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{22.89e^{-0.2s}}{4.572s+1} & \frac{-11.64e^{-0.4s}}{1.807s+1} \\ \frac{4.689e^{-0.2s}}{2.174s+1} & \frac{5.80e^{-0.4s}}{1.801s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{-4.243e^{-0.4s}}{3.445s+1} \\ \frac{-0.601e^{-0.4s}}{1.982s+1} \end{bmatrix} \left[d(s)\right] \tag{22}$$

where $y_1(s)$ and $y_2(s)$ are the outputs, $u_1(s)$ and $u_2(s)$ are the inputs and $d(s)$ is the disturbance signal. The controller used in this work is a diagonal matrix of transfer functions, as shown in equation (23) given by:

$$G(s) = \begin{bmatrix} g_{11}(s) & 0 & \cdots & 0 \\ 0 & g_{22}(s) & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & g_{nn}(s) \end{bmatrix} \tag{23}$$

where each term is a controller with the same structure of (6). As the system is a 2x2 system then the controller becomes:

$$G(s) = \begin{bmatrix} g_{11}(s) & 0 \\ 0 & g_{22}(s) \end{bmatrix} \tag{24}$$

The goal is to find the configuration of the gains of the two controllers that minimizes the objective function. In this paper the ITAE performance index, equation (9), is used in the objective function to be minimized. The error signal is defined as the difference between the input and the output, so there are two errors: one for the first input (related to the first output) and another for the second input. Then the objective function becomes as follows:

$$f = ITAE(e_{11}) + ITAE(e_{12}) + ITAE(e_{21}) + ITAE(e_{22}) \tag{25}$$

where e_{ij} is the error signal of the *i*-th output related to the *j*-th input.

Also a penalty function is used to avoid infeasible solutions. Infeasible solutions are those which do not achieve the reference or makes the system to be unstable. The penalty function is described by the following (Coello, 1999):

$$p(X) = (Ct)^{\alpha} \sum_{i=1}^{m} \left| \phi_i(X) \right|^{\beta} \tag{26}$$

where C, α and β are user defined constants, t is the current iteration of the algorithm, $\phi_j(X)$ is the violation of the *i*-th constraint and $p(X)$ is the penalty value for the solution *X*. In this case, the solution *X* is an array containing the gains for the controller given by equation (27),

$$X = [k_{p1}, k_{p2}, t_{i_1}, t_{i_2}, t_{d_1}, t_{d_2}] \tag{27}$$

So the objective function becomes:

$$F(X) = f + p(X) \tag{28}$$

where $e_i(X)$ is the error of the *i*-th output related to the *i*-th input when using the gains of X.

5. SIMULATION RESULTS

Tests were carried out using Matlab® and Simulink®. In order to avoid the issues caused by randomness, 20 runs for each optimization algorithm were made using different initial populations. The only one stopping criteria used was the number of generations that was equal to 20. The other parameters were adjusted to: populations size P=20, number of generations G_{max}=20, maximum mutation rate m_{max}=0.7, and emigration and immigration rates E=I=1. Note that the parameters are the same for both algorithms

Table 2 shows the statistical comparison between the solutions found by the algorithms. In Figures 3-6 are the responses of the system with the best configuration found by both BBO and SW-BBO methods of the controller gains. Tables 4 and 5 present the measurements of settling time (time for the response enter in a band of 2% of the final response), rising time (time for the response achieve, for the first time, the set point), and the maximum overshoot (the maximum value of the signal).

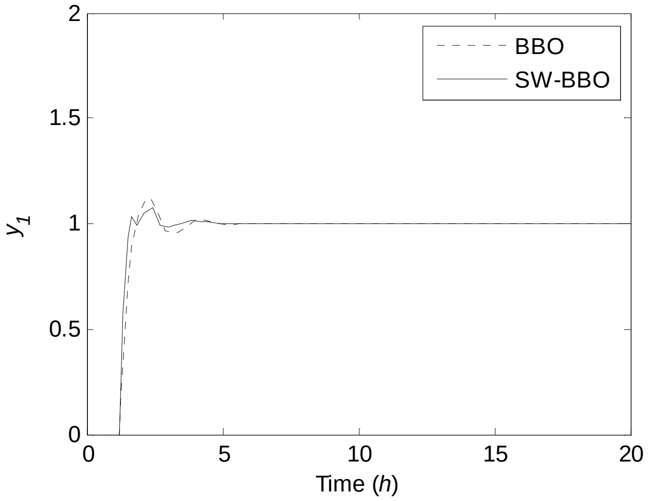

Figure 3. Response of y_1 to a step in u_1.

Figure 3 shows that the responses of y_1, with a step input applied in u_1, are very similar for both techniques, but that using SW-BBO is faster than the other using BBO. However, the regulatory response (Fig. 4) was better when using BBO for tune the controller, because the overshoot for this case was smaller and the settling time was almost the same for both cases.

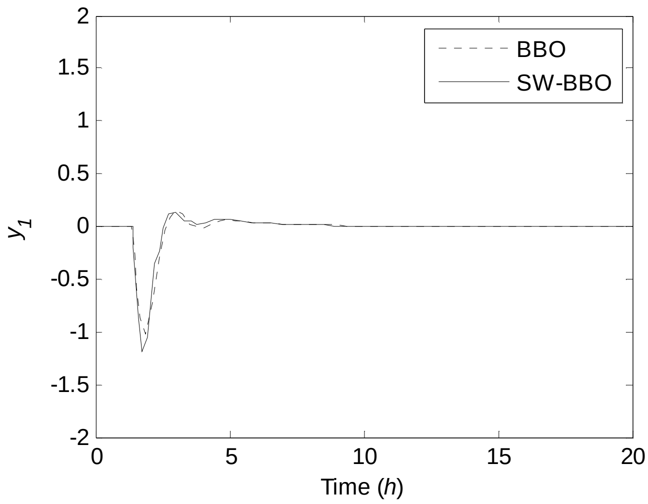

Figure 4. Response of y_1 to a step in u_2.

Figures 5 and 6 show that, for the servo response and the regulatory response (when input u_2 changes), the best case was that using the SW-BBO algorithm to tune the gains of the PID controller. Table 3 presents the best gains found by both optimization algorithms.

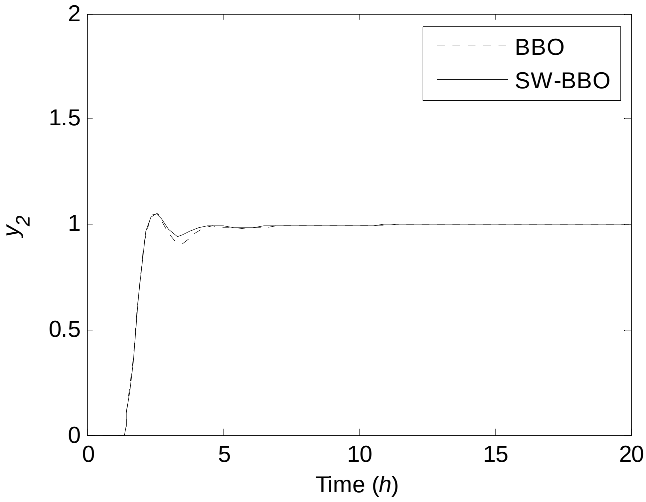

Figure 5. Response of y_2 to a step in u_2.

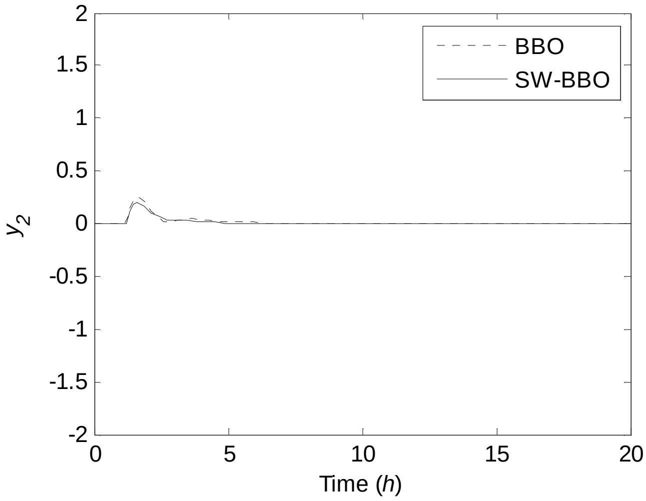

Figure 6. Response of y_2 to a step in u_1.

Table 2. Comparison in terms of objective function (20 runs).

Method	Best	Worst	Mean	Standard deviation
BBO	36.81	4706263.45	235501.07	1052308.35
SW-BBO	33.34	173.68	53.77	29.61

The worst solution of BBO is a controller that makes the system unstable, wherefore the value of the objective function is too large.

Table 3. Best configurations of PID gains.

Method	k_{p_1}	k_{p_2}	t_{i_1}	t_{i_1}	t_{d_1}	t_{d_2}
BBO	0.142	0.119	1.666	0.933	0.500	0.255
SW-BBO	0.216	0.097	1.836	0.727	0.342	0.278

Tables 4 and 5 evaluate the responses and appoint the best result. In those tables t_s is the settling time, t_r is the rising time and m_o is the maximum overshoot. Times are in hours and the maximum overshoot is the maximum absolute value of the output. Note that when it is the regulatory case, the settling time becomes large because it is the time to stay in a band of 2% of the final response, and the final response for the regulatory case is zero, so this time is the time to return to the initial state.

Table 4. Measurements when setpoint 1 changes.

	Output	t_s	t_r	m_o
BBO	y_1	2.3818	0.6542	1.1211
	y_2	19.9794	-	0.2568
SW-BBO	Output	t_s	t_r	m_o
	y_1	2.1125	0.1469	1.0768
	y_2	19.6386	-	0.2053

Table 5. Measurements when setpoint 2 changes.

	Output	t_s	t_r	m_o
BBO	y_1	18.9856	-	1.0184
	y_2	2.3821	0.6560	1.0476
SW-BBO	Output	t_s	t_r	m_o
	y_1	18.9205	-	1.1943
	y_2	2.5437	0.7042	1.0495

6. CONCLUSION

This paper has presented a comparison between two evolutionary optimization algorithms, the classical BBO and a new proposed SW-BBO approach, in PID tuning for multivariable system. Simulation results clearly show that for the reactor problem, SW-BBO demonstrates better performance than the standard BBO in PID tuning. However, these optimization algorithms were used for off-line PID tuning. In future works they can be adapted for on-line tuning of PID controllers in processes with slow dynamics.

7. ACKNOWLEDGEMENTS

This work was supported by the National Council of Scientific and Technologic Development of Brazil — CNPq — under Grants 303963/2009-3/PQ and ‘Fundação Araucária’ under Grant 14/2008-416/09-15149. The first author, also, would like to thanks the Pontifical Catholic University of Parana — PUCPR for the financial support provided through the Institutional Program for Scientific Initiation Scholarships — PIBIC.

8. REFERENCES

Altınten, A., Ketevanlioğlu, F., Erdoğan, S., Hapoğlu, H. and Alpbaz, M. (2008). Self-tuning PID control of jacketed batch polystyrene reactor using genetic algorithm, *Chemical Engineering Journal*, vol. 138, no. 1-3, pp. 490-497.

Bhattacharya, A. and Chattopadhyay, P.K. (2010). Solving complex economic load dispatch problems using biogeography-based optimization, *Expert Systems with Applications*, vol. 37, no. 5, pp. 3605-3615.

Chien, I.L., Huang, H.P. and Yang, J.C. (1999). A simple multiloop tuning method for PID controllers with no proportional kick, *Industrial & Engineering Chemistry Research*, vol. 38, no. 4, pp. 1456-1468.

Coello, C.A.C. (1999). A survey of constraint handling techniques used with evolutionary algorithms. *Technical Report Lania-RI-99-04*, Xalapa, Veracruz, México.

Fleming, P.J. and Purshouse, R.C. (2002). Evolutionary algorithms in control systems engineering: a survey, *Control Engineering Practice*, vol. 10, no. 11, pp. 1223-1241.

Gong, W., Cai, Z., Ling, C. X. and Li, H. (2010). A real-coded biogeography-based optimization with mutation, *Applied Mathematics and Computation*, vol. 216, no. 9, pp. 2749-2758.

Iruthayarajan, M.W. and Baskar, S. (2009). Evolutionary algorithms based design of multivariable PID controller, *Expert Systems with Applications*, vol. 36, no. 5, pp. 9159-9167.

Iruthayarajan, M.W. and Baskar, S. (2010). Covariance matrix adaptation evolution strategy based design of centralized PID controller, *Expert Systems with Applications*, vol. 37, no. 8, pp. 5775-5781.

Jiang C., Ma, Y. and Wang, C. (2006). PID controller parameters optimization of hydro-turbine governing systems using deterministic-chaotic-mutation evolutionary programming (DCMEP), *Energy Conversion and* Management, vol. 47, no. 9-10, pp. 1222-1230.

Levine, W.S. (1996). *The control handbook*. Piscataway, NJ, USA: CRC Press/IEEE Press.

Lianghong, W., Yaonan, W., Shaowu, S. and Wen, T. (2008). Design of PID controller with incomplete derivation based on differential evolution algorithm, *Journal of Systems Engineering and Electronics*, vol. 19, no. 3, pp. 578-583.

Kumar, S., Bhalla, P. and Singh, A. (2009). Fuzzy rule base generation from numerical data using biogeography-based optimization. *Institution of Engineers Journal of Electronics and Telecomm Engineering*, vol. 90, no. 1, pp. 8-13.

Kundra, H., Kauer, A. and Panchal, V. (2009). An integrated approach to biogeography based optimization with case based reasoning for retrieving groundwater possibility, *Proceedings of 8th Annual Asian Conference and Exhibition on Geospatial Information, Technology and Applications*, Singapore.

Rarick, R., Simon, D., Villaseca F.E. and Vyakaranam, B. (2009). Biogeography-based optimization and the solution of the power flow problem, *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, San Antonio, TX, USA, pp. 1029-1034.

Simon, D. (2008). Biogeography-based optimization, *IEEE Transactions on Evolutionary Computation*, vol. 12, no. 6, pp. 702-713.

Simon, D., Ergezer, M. and Du, D. (2009). Population distributions in biogeography-based optimization algorithms with elitism, *Proceedings of IEEE Conference on Systems, Man, and Cybernetics*, San Antonio, TX, USA, pp. 1017-1022.

Ziegler, J.G. and Nichols, N.B. (1942). Optimum settings for automatic controllers, *Transactions of the ASME*, vol. 64, no. 8, pp. 759-768.

9. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.