ISSN 2229-5518

Automatic PID Controller Parameter Tuning Using Bees Algorithm

Moslem Amirinejad, Mahdiyeh Eslami, Ali Noori

Abstract— Despite numerous advancements in process control methodologies,Proportional–Integral–Derivative (PID) controller is still the mostefficient and widely used feedback control strategy. This is due to itssimplicity and satisfactory control performance.This paper presents an efficient and fast tuning method based on a bees algorithm (BA) structure to find the optimal parameters of the PID controller so that the desired system specifications are satisfied.To demonstratethe effectiveness of presented method, the step responses of closed loop systemwere compared with that of the existing methods in the literature. Simulation results indicatethat the performance of the PID controlled system can be significantly improved by the BA-based method.

Index Terms— Bees algorithm, PID controller, PID tuning, parameter optimization, Ziegler and Nichols.

1 Introduction

PROPPORTIONAL–integral–derivative (PID) control has been widelyapplied in industry—more than 90% of the applied controllersare PID controllers [1–6]. PID controller wasintroduced in 1910 and its use and popularity had grown particularlyafter the Ziegler–Nichols empirical tuning rules in 1942 [2, 7].

The developmentin artificial intelligence and digital technology have resultedin many intelligent control schemes such as fuzzy logic control [8, 9], neural network control [10] and adaptive control [11, 12]. But no other technique could replace PID algorithmand as mentioned more than 90% of industrial controllers are still based on PIDcontrol.

In the absence of the derivativeaction, proportional–integral (PI) control is also broadly deployed,since in many cases the derivative action cannot significantlyenhance the performance or may not be appropriate for thenoisy environment. Another special form of PID controlwithout the integral action, proportional–derivative (PD) controlis also applied. Unlike the previous two cases, however,PD control cannot achieve zero steady-state error subject to loaddisturbances, which limits its applications [1–4].

Due to the prevailing applications of PI/PD/PID control, researchon tuning PI/PD/PID controllers has been of much interest inthe past decades [1–7, 13].

The optimally combined three terms functioning of PID controllercan provide treatment for both the transient and steady stateresponses. In fact, optimal control performance can only beachieved after identifying the finest set of three gains, that is, proportionalgain (K_p), integral gain (K_i) and derivative gain (K_d). Manyapproaches have been reported in literature for tuning parametersof PID controller. The conventional PID tuning techniques includeZ–N, Cohen Coon, and relay feedback methods [7, 14]. The modern techniques are basedon artificial intelligence techniques such as neural network, fuzzylogic and evolutionary computation; these are the most recenttechniques [15].

Recently, many attempts have been made by several researchersto tune the PID controller parameters using various EAs, suchas genetic algorithm (GA), covariance matrix adaptation evolutionstrategy (CMAES), particle swarm optimization (PSO), differentialevolution (DE), tribes algorithm (TA), ant colony optimization(ACO), and discrete binary particle swarm optimization (DBPSO) [16- 26].

AI-based evolutionary computational techniques can determinethe most optimal sets of controller gains based on a given objectivefunction in an iterative manner from thousands of possible alternatesolutions that best fit the designer's requirements. But theperformance of different methods may significantly vary in differentapplications.As well known that both exploration and exploitation are necessary for the optimization algorithms, such as GA, PSO, and ACO and so on. In these optimization algorithms, the exploration refers to the ability to investigate the various unknown regions in the solution space to discover the global optimum. While, the exploitation refers to the ability to apply the knowledge of the previous good solutions to find better solutions [27]. In practice, the exploration and exploitation contradict with each other, and in order to achieve good optimization performance, the two abilities should be well balanced.

In this study, the bees algorithm is applied on overall system to obtain the design objectives by adjusting the controller parameters at each iteration, repetitively until the desired closed-loop system performance is achieved. The performance of the closed-loop system can defined in terms of rise time, overshoot, settling time and steady state error.In general, the system with fast rise and settling time under no steady-state error and almost zero overshoot is desired. Hence, in this study to provide a desired performance, the integratedsquared error (ISE), settling time and overshoot is minimized by using BA. The merits of the proposed controller are illustrated by considering the third order and forth order systems.The superior

- *Moslem Amirinejad, amirinejadmoslem@yahoo.com*
- *Mahdiyeh Eslami, m_eslami@srbiau.ac.ir*
- *Ali Noori, dr_noori_cardio@yahoo.com*
  Department of Electrical Engineering Science and Research Branch, Islamic Azad University, Kerman, Iran.

performance of the BA is due to its ability to simultaneously refine a local search, while still searching globally.
The rest of paper is organized as follows. Section 2 explains the PID controller. Section 3 presents the optimization algorithm. Section 4, shows simulation results and finally Section 5 concludes the paper.

2 PID Controller

A PID controller is a combination of a proportional, anintegral and a derivative controller, integrating the mainfeatures of all three. Fig. 1 demonstrates a simplified blockdiagram of a plant controlled by a PID. The output of a PIDcontroller, which is the processed error signal, can bepresented as:

$$u(t) = K_p e(t) + K_i \int_0^{\infty} e(t)dt + K_d e(t) \qquad (1)$$

where K_p, K_i and K_d are the proportional, integral and derivative gains, respectively.

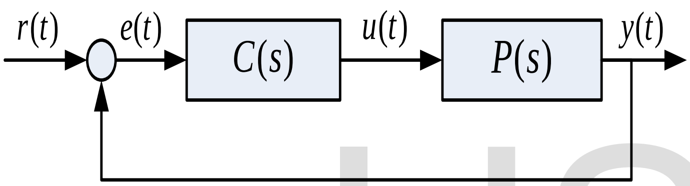

Figure 1. A plant controlled by a PID controller

In general, the objective of PID controllers like any othercontroller is to provide stability as well as reference trackingand disturbance rejection, which are all design criteriarelated to steady domain of response. Different indices havebeen suggested to evaluate the performance of a controllerbased on the above objectives. The most common ones arethe integrated absolute error (IAE), integrated squared error(ISE), integrated time squared error (ITSE), and integrated time absolute error (ITAE). These indices are normallycalculated under step testing input in the time domain as:

$$IAE = \int_0^{\infty} \left| r(t) - y(t) \right| dt = \int_0^{\infty} \left| e(t) \right| dt$$

$$ISE = \int_0^{\infty} e^2(t)dt$$

$$ITSE = \int_0^{\infty} te^2(t)dt \qquad (2)$$

$$ITAE = \int_0^{\infty} t\left| e(t) \right| dt$$

Obviously as they all represent the concept of error;minimization of these indices is desired.
For the transient domain of response, maximum overshoot (OS), settling time (t_s) and rise time (t_r) arenormally considered significant where the benefit of fastersystems, necessitates minimum possible values for them.For tuning PID controllers that is finding the optimumgains for the best performance, one or a weightedcombination of these criteria is employed. While weights andnumber of indices are diversely reported in the literature, it isgenerally accepted that time weighted indices are moreappropriate as the errors occurring later in the transientresponse are penalized heavily. In this paper, selection of anyof these criteria has been constrained by benchmarkproblems, though ISE index is calculated and reportedindependently to make comparisons more sensible.

3 Bees algorithm

BA is an optimization algorithm inspired by the natural foraging behavior of honey bees to find the optimal solution. Figure 2 shows the pseudo-code for the algorithm in its simplest form. The algorithm requires a number of parameters to be set, namely: Number of scout bees (n), number of sites selectedout of n visited sites (m), number of best sites out of m selected sites (e), number of bees recruited for beste sites (nep), number of bees recruited for the other (m-e) selected sites (nsp), initial size of patches (ngh) which includes site and its neighborhood and stopping criterion. The algorithm starts with the n scout bees being placed randomly in the search space. The fitnesses of the sites visited by the scout bees are evaluated in step 2.

1. Initialise the solution population.
2. Evaluate the fitness of the population.
3. While (stopping criterion is not met)
//Forming new population.
4. Select sites for neighbourhood search.
5. Recruit bees for selected sites (more bees for the best e sites) and evaluate fitnesses.
6. Select the fittest bee from each site.
7. Assign remaining bees to search randomly and evaluate their fitnesses.
8. End While

Fig 2. Pseudo code

In step 4, bees that have the highest fitnesses are chosen as “selected bees” and sites visited by them are chosen for neighborhood search. Then, in steps 5 and 6, the algorithm conducts searches in the neighborhood of the selected sites, assigning more bees to search near to the best e sites. The bees can be chosen directly according to the fitnesses associated with the sites they are visiting. Alternatively, the fitness values are used to determine the probability of the bees being selected. Searches in the neighborhood of the best e sites which represent more promising solutions are made more detailed by recruiting more bees to follow them than the other selected bees. Together with scouting, this differential recruitment is a key operation of the BA.
However, in step 6, for each patch only the bee with the highest fitness will be selected to form the next bee population. In nature, there is no such a restriction. This restriction is

introduced here to reduce the number of points to be explored. In step 7, the remaining bees in the population are assigned randomly around the search space scouting for new potential solutions. These steps are repeated until a stopping criterion is met. At the end of each iteration, the colony will have two parts to its new population representatives from each selected patch and other scout bees assigned to conduct random searches [28].

4 Simulation Results

The proposed PID tuning based on a BA is schematically shown in Fig. 3. The major objective of the BA program is to determine the optimal values of the PID controllerparameters to improve the transient response of the system at time.

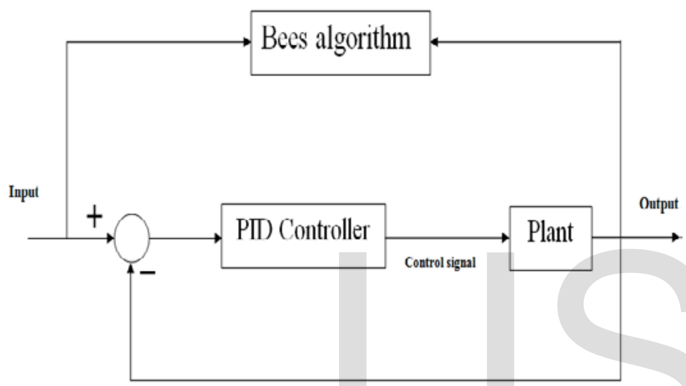

Fig 3. The proposed PID tuning based on a BA

During the optimization process, the reference input and closed loop response of the process is used by the BA. Using the changed closed loop control performance accordingto the adjusted controller parameters at the each generation, the tuning algorithmsearches the optimal parameters for the PID controller to satisfy the desired system specifications.To illustrate the effectiveness of the presented method, we compared the closedloop response to a step change of a number of simulated systems. For PID controller problem, two different processes with different order are considered as the following [29]:

$$G_1(s) = \frac{4.228}{(s+0.5)(s^2+1.64s+8.456)}$$
$$G_2(s) = \frac{27}{(s+1)(s+1)^3} \quad (3)$$

In control system applications, the chosen performance criterion is often a weightedcombination of various performance characteristics such as rise time, settling time, overshoot,and integral of the square of the error. The desired system response should haveminimal settling time with a small or no overshoot in the step response of the closed loopsystem. Therefore, the objective function f is defined using the performance indicesintegral of the square of the error (ISE), the response overshoot (OS) and the 5 percentsettling time (t_s).

$$f = 10(ISE) + 3(t_s) + OS \quad (4)$$

The searchdomain for PID gains which are the design variables here is[0.1, 5]. To implement the algorithm, D=3 is assigned torepresent three design variables K_p, K_i and K_d as PID gains. The indices employed in Eq. 4 are computed based on a model-basedresponse analysis of the processes using MATLAB version 2009a.Table 1 shows the BA parameters.

Table 1
Parameters used in the BA

Number of scout bees, n	50
Number of sites selected for neighborhood search, m	20
Number of best "elite" sites out of m selected sites, e	10
Number of bees recruited for best e sites, nep	8
Number of bees recruited for the other (m-e) selected sites, nsp	8
Number of iterations, R	100

The results obtained by BA and the other availablemethods in the literature are summarized in Table 2. Figures 4 and 5 show the open loop step response for G1 and G2 respectively. Inaddition, the step response of both systems G1 and G2 usingPID controllers tuned by Z-N [29], MGA [29] and BA aredemonstrated in Figures 6 and 7. Clearly, BA has outperformed thebest available solutions obtained by MGA and for bothprocesses G1 and G2; an improvement of about 13% isachieved. In addition, it is noticeable that the optimal gainsobtained by BA are not in the neighborhood of the onesreported by MGA. This proves that BA has been wellequipped not to trap in local optima though the optimizationproblem is not constrained and the problem space is convex.
Furthermore,having a look on the amounts of ISE, OS and t_s, it isobserved that except for the case of overshoot in G2, BA has reduced them independently which is of importance fromthe designing point of view.To complete the analysis ontuned PID controllers, it is necessary to discuss amplitude ofcontrol signals. The obtained results confirm that the ordersof amplitude of signals are limited and quite the same. As nofurther information about the physical aspects of thecontrolled processes is available, it is not possible to evaluateany boundaries for the maximum allowable amplitude ofcontrol signals in any of test problems. Finally, it is notable
that the step response of G2 tuned by BA exhibits anundershoot of about 42%. Although undershoot is generallyconsidered a minor parameter, it might be important in acertain plant. Anyway, the problem formulation does notinclude undershoot in this paper.

ISSN 2229-5518

TABLE 2. THE RESULTS OBTAINED BY BA AND THE OTHER AVAILABLE METHODS IN THE LITERATURE

		Z-N	MGA	BA	Open loop
G_1	K_p	2.19	1.637	2.766	-
	K_i	2.126	0.965	1.263	-
	K_d	0.565	0.388	2.415	-
	$t_s(\pm 2\%)$	6.6156	5.9708	4.8072	7.8812
	$OS(\%)$	16.4262	3.3811	1.9066	0
	ISE	5.6877	7.1448	4.4906	12.9354
	F	93.1497	92.7411	61.2341	152.9975
G_2	K_p	3.072	1.772	2.141	-
	K_i	2.272	1.06	1.248	-
	K_d	1.038	0.773	1.145	-
	$t_s(\pm 2\%)$	5.1551	1.8561	3.3306	5.1284
	$OS(\%)$	32.5301	0.1432	0.5994	0
	ISE	6.7537	7.3121	6.2444	14.3867
	F	115.5324	78.8320	73.0352	159.2525

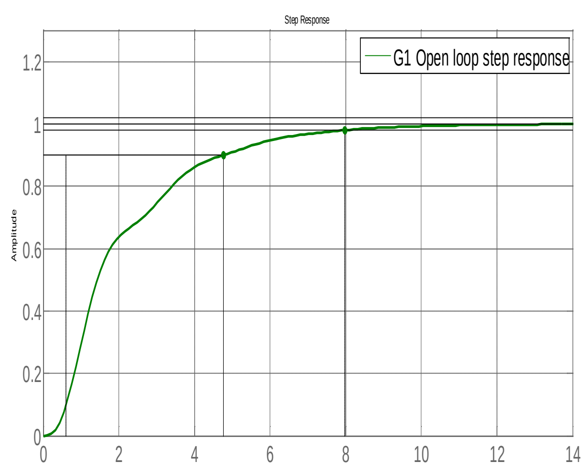

Figure 4. G1 open loop step response

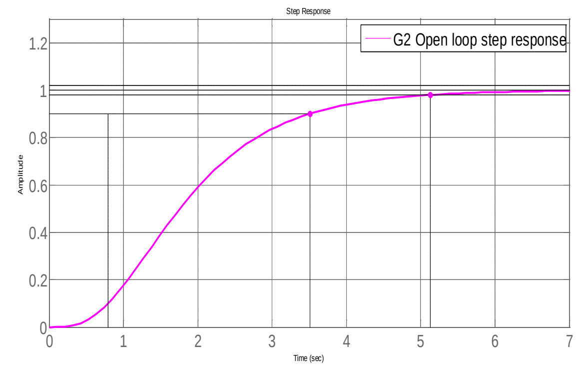

Figure 5. G5 open loop step response

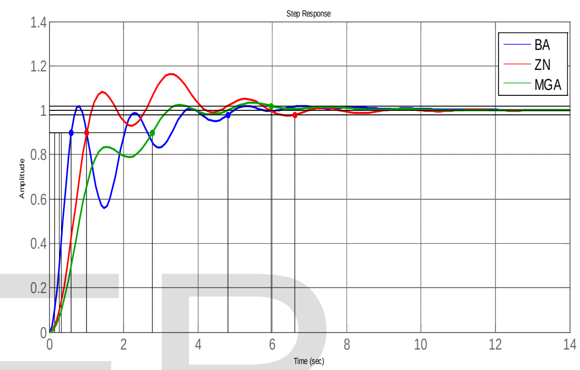

Figure 6. Step response of processes G1having PID controllers tuned by Z-N, MGA and BA

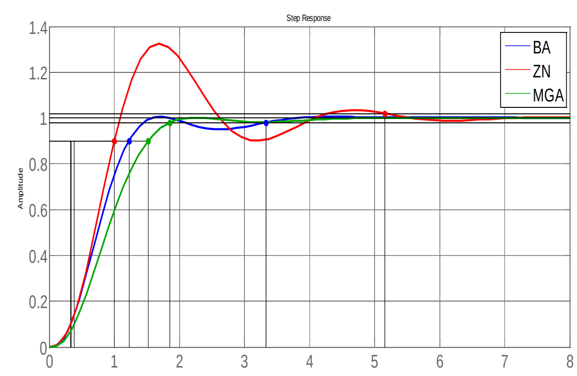

Figure 7. Step response of processes G2having PID controllers tuned by Z-N, MGA and BA

5 CONCLUSION

Bees algorithm was employed totune PID controllers for plants of high order. The method optimized PID gains as design variablesin both single- and multi-objective approaches. The objectivefunctions taken from literature were important performanceindices of ITSE, ISE and IAE as well as overshoot andsettling time. Results clearly expressed that the utilized-method has been successful in comparison to geneticalgorithm and Z-N techniques; and can beconsidered as a powerful tuning scheme for controllers.

REFERENCES

[1] O'Dwyer A. Handbook of PI and PID controller tuning rules. 2nd ed. London:Imperial College Press; 2006.

[2] Astrom KJ, Hagglund T. Advanced PID control. Research Triangle Park (NC):ISA-The Instrumentation, Systems, and Automation Society; 2005.

[3] Visioli A. Practical PID control. London: Springer; 2006.

[4] Kano M, Ogawa M. The state of art in advanced process control in Japan.In: IFAC symposium ADCHEM. Istanbul (Turkey); 2009.

[5] Johnson MA, Moradi MH, Crowe J. PID control: new identification and designmethods. Berlin: Springer; 2005.

[6] Seborg D, Edgar T, Mellichamp D. Process dynamics and control. 2nd ed. NewYork (USA): John Wiley; 2004

[7] Ziegler, J. G., & Nichols, N. B. (1942). Optimum settings for automatic controllers. Transactions on ASME, 64(8), 759–768.

[8] Goshal, S. P. (2004). Optimization of PID gains by particle swarm optimization infuzzy based automatic generation control. Electric Power System Research, 72,203–212.

[9] Lee, C. C. (1990). Fuzzy logic in control systems: Fuzzy logic controller – Part I and II.IEEE Transactions on Systems, Man and Cybernetics, 20(2), 404–435.

[10] Fukuda, T., & Shibata, T. (1992). Theory and application of neural networks forindustrial control systems. IEEE Transactions on Industrial Electronics, 39(6),472–489.

[11] Astrom, K. J., &Wittenmark, B. (1995). Adaptive controls (2nd ed.). Addison Wesley.

[12] Zuo, W. (1995). Multivariable adaptive control for a space station using geneticalgorithm. IEEE Proceedings Control Theory and Applications, 142(2), 81–87.

[13] Marsh, P. (1998). Turn on tune in-Where can the PID controllers go next. NewElectron, 31(4), 31–32.

[14] Cohen, G. H., & Coon, G. A. (1953). Theoretical considerations of retarded control. Transactions on ASME, 75, 827–834.

[15] Astrom, K. J., &Hagglund, T. (1995). PID controllers: Theory, design and tuning (2nded.). Instrument Society of America.

[16] Bingul, Z. (2004). A new PID tuning technique using differential evolution forunstable and integrating processes with time delay. In ICONIP, proceedingslecture notes in computer science (Vol. 3316, pp. 254–260).

[17] Chang, W. D. (2007). A multi-crossover genetic approach to multivariable PIDcontrollers tuning. Expert Systems with Applications, 33, 620–626.

[18] Chang, W. D. (2009). PID control for chaotic synchronization using particle swarmoptimization. Chaos, Solitons and Fractals, 39(2), 910–917.

[19] Coelho, L. S., &Bernert, D. L. A. (2009). PID control design for chaoticsynchronization using tribes optimization approach. Chaos, Solitons andFractals, 42(1), 634–640.

[20] Duan, H., Wang, D., & Yu, X. (2006). Novel approach to nonlinear PID parameteroptimization using ant colony optimization algorithm. Journal of BionicEngineering, 3(2), 73–78.

[21] Gaing, Z. L. (2004). A particle swarm optimization approach for optimum design ofPID controller in AVR system. IEEE Transactions on Energy Conversion, 19(2),384–391.

[22] Mukherjee, V., &Goshal, S. P. (2007). particle swarm optimized fuzzy PID controllerfor AVR system. Electric Power System Research, 77(12), 1689–1698.

[23] Zhang, J., Zhuang, J., Du, H., & Wang, S. (2009). Self-organizing genetic algorithmbased tuning of PID controllers. Information Sciences, 179(7), 1007–1018.

[24] Willjuice, I. M., &Baskar, S. (2009). Evolutionary algorithm based design ofmultivariable PID controller. Expert Systems with Applications, 36, 9159–9167.

[25] Willjuice, I. M., &Baskar, S. (2010). Covariance matrix adaptation evolution strategybased design of centralized PID controller. Expert Systems with Applications, 37,5775–5781.

[26] M. Menhas, L. Wang, M.Fei, H. Pan. Comparative performance analysis of various binary coded PSO algorithmsin multivariable PID controller design. Expert Systems with Applications 39 (2012) 4390–4401

[27] I.C. Trelea, The particle swarm optimization algorithm: convergence analysis and parameter selection, Information Processing Letters 85 (2003) 317–325.

[28] D.T. Pham, A. Ghanbarzadeh, E. Koc¸, S. Otri, S. Rahim, M. Zaidi, The bees algorithm–a novel tool for complex optimisation problems, in: Intelligent Production Machines and Systems (2006) 454–459.

[29] A. Bagis, "Determination of the PID controller parameters bymodified genetic algorithm for improved peformance," J. Inform. Sci.Tech., vol. 23, pp. 1469-1480, 2007.