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 most efficient and widely used feedback control strategy. This is due to its simplicity 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 demonstrate the effectiveness of presented method, the step responses of closed loop systems were compared with that of the existing methods in the literature. Simulation results indicate that 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 widely applied in industry—more than 90% of the applied controllers are PID controllers [1–6]. PID controller was introduced in 1910 and its use and popularity had grown particularly after the Ziegler–Nichols empirical tuning rules in 1942 [2, 7].

The development of artificial intelligence and digital technology have resulted in 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 algorithm as mentioned more than 90% of industrial controllers are still based on PID control.

In the absence of the derivative action, proportional–integral (PI) control is also broadly deployed, since in many cases the derivative action cannot significantly enhance the performance or may not be appropriate for the noisy environment. Another special form of PID control without the integral action, proportional–derivative (PD) control is also applied. Unlike the previous two cases, however, PD control cannot achieve zero steady-state error subject to load disturbances, which limits its applications [1–4].

Due to the prevailing applications of PI/PD/PID control, researchers have tuned PI/PD/PID controllers has been of much interest in the past decades [1–7, 13].

The optimally combined three terms functioning of PID controller can provide treatment for both the transient and steady state responses. In fact, optimal control performance can only be achieved after identifying the finest set of three gains, that is, proportional gain ($K_p$), integral gain ($K_i$) and derivative gain ($K_d$). Many approaches have been reported in literature for tuning parameters of PID controller. The conventional PID tuning techniques include Z-N, Cohen Coon, and relay feedback methods [7, 14]. The modern techniques are based on artificial intelligence techniques such as neural network, fuzzy logic and evolutionary computation; these are the most recent techniques [15].

Recently, many attempts have been made by several researchers to tune the PID controller parameters using various EAs, such as genetic algorithm (GA), covariance matrix adaptation evolution strategy (CMAES), particle swarm optimization (PSO), differential evolution (DE), tribes algorithm (TA), ant colony optimization (ACO), and discrete binary particle swarm optimization (DBPSO) [16–26].

Al-based evolutionary computational techniques can determine the most optimal sets of controller gains based on a given objective function in an iterative manner from thousands of possible alternates to find better solutions. 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 be 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 integrated squared 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...
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 be represented as:

\[ u(t) = K_p e(t) + K_i \int_0^t e(t) \, dt + K_d \frac{d}{dt} e(t) \]  

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](http://www.ijser.org)

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 are the integrated absolute error (IAE), integrated squared error (ISE), integrated time squared error (ITSE), and integrated time absolute error (ITAE). These indices are normally calculated under step testing input in the time domain as:

\[
    IAE = \int_0^\infty |r(t) - y(t)| \, dt = \int_0^\infty |e(t)| \, dt
\]

\[
    ISE = \int_0^\infty e^2(t) \, dt
\]

\[
    ITSE = \int_0^\infty te^2(t) \, dt
\]

\[
    ITAE = \int_0^\infty |e(t)| \, dt
\]

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 \( (t_r) \) are normally considered significant where the benefit of fastersystems, necessitates minimum possible values for them. For tuning PID controllers that is finding the optimum gains for the best performance, one or a weighted combination of these criteria is employed. While weights and number of indices are diversely reported in the literature, it is generally accepted that time weighted indices are more appropriate as the errors occurring later in the transient response are penalized heavily. In this paper, selection of any of these criteria has been constrained by benchmark problems, though ISE index is calculated and reported independently 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 selected out of n visited sites (m), number of best sites out of m selected sites (e), number of bees recruited for best 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 controller parameters to improve the transient response of the system at time.

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 according to the adjusted controller parameters at each generation, the tuning algorithm searches the optimal parameters for the PID controller to satisfy the desired system specifications. To illustrate the effectiveness of the presented method, we compared the closed loop 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]:

\[
G_1(s) = \frac{4.228}{(s + 0.5)(s^2 + 1.64s + 8.456)}
\]

\[
G_2(s) = \frac{27}{(s + 1)(s + 1)}
\]  

In control system applications, the chosen performance criterion is often a weighted combination of various performance characteristics such as rise time, settling time, overshoot, and integral of the square of the error. The desired system response should have minimal settling time with a small or no overshoot in the step response of the closed loop system. Therefore, the objective function \( f \) is defined using the performance indices integral of the square of the error (ISE), the response overshoot (OS) and the 5 percent settling time (\( t_s \)).

\[
f = 10(\text{ISE}) + 3(t_s) + \text{OS}
\]  

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

<table>
<thead>
<tr>
<th>Parameters used in the BA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of scout bees, ( n )</td>
</tr>
<tr>
<td>Number of sites selected for neighborhood search, ( m )</td>
</tr>
<tr>
<td>Number of best “elite” sites out of ( m ) selected sites, ( e )</td>
</tr>
<tr>
<td>Number of bees recruited for best ( e ) sites, ( n_{ep} )</td>
</tr>
<tr>
<td>Number of bees recruited for the other ( (m-e) ) selected sites, ( n_{sp} )</td>
</tr>
<tr>
<td>Number of iterations, ( R )</td>
</tr>
</tbody>
</table>

The results obtained by BA and the other available methods in the literature are summarized in Table 2. Figures 4 and 5 show the open loop step response for \( G_1 \) and \( G_2 \) respectively. In addition, the step response of both systems \( G_1 \) and \( G_2 \) using PID controllers tuned by Z-N [29], MGA [29] and BA are demonstrated in Figures 6 and 7. Clearly, BA has outperformed the best available solutions obtained by MGA and for both processes \( G_1 \) and \( G_2 \); an improvement of about 13% is achieved. In addition, it is noticeable that the optimal gains obtained by BA are not in the neighborhood of the ones reported by MGA. This proves that BA has been well-equipped not to trap in local optima though the optimization problem is not constrained and the problem space is convex.

Furthermore, having a look on the amounts of ISE, OS and \( t_s \), it is observed that except for the case of overshoot in \( G_2 \), BA has reduced them independently which is of importance from the designing point of view. To complete the analysis on tuned PID controllers, it is necessary to discuss amplitude of control signals. The obtained results confirm that the order of magnitude of signals are limited and quite the same. As no further information about the physical aspects of the controlled processes is available, it is not possible to evaluate any boundaries for the maximum allowable amplitude of control signals in any of the test problems. Finally, it is notable that the step response of \( G_2 \) tuned by BA exhibits an undershoot of about 42%. Although undershoot is generally considered a minor parameter, it might be important in certain plant. Anyway, the problem formulation does not include undershoot in this paper.
Table 2. The results obtained by BA and the other available methods in the literature

<table>
<thead>
<tr>
<th></th>
<th>Z-N</th>
<th>MGA</th>
<th>BA</th>
<th>Open loop</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>2.19</td>
<td>1.637</td>
<td>2.766</td>
<td>-</td>
</tr>
<tr>
<td>$K_i$</td>
<td>2.126</td>
<td>0.965</td>
<td>1.263</td>
<td>-</td>
</tr>
<tr>
<td>$K_d$</td>
<td>0.565</td>
<td>0.388</td>
<td>2.415</td>
<td>-</td>
</tr>
<tr>
<td>$t_s (\pm 2%)$</td>
<td>6.6156</td>
<td>5.9708</td>
<td>4.8072</td>
<td>7.8812</td>
</tr>
<tr>
<td>OS(%)</td>
<td>16.4262</td>
<td>3.3811</td>
<td>1.9066</td>
<td>0</td>
</tr>
<tr>
<td>ISE</td>
<td>5.6877</td>
<td>7.1448</td>
<td>4.4906</td>
<td>12.9354</td>
</tr>
<tr>
<td>F</td>
<td>93.1497</td>
<td>92.7411</td>
<td>61.2341</td>
<td>152.9975</td>
</tr>
</tbody>
</table>

$G_1$

<table>
<thead>
<tr>
<th></th>
<th>Z-N</th>
<th>MGA</th>
<th>BA</th>
<th>Open loop</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K_p$</td>
<td>3.072</td>
<td>1.772</td>
<td>2.141</td>
<td>-</td>
</tr>
<tr>
<td>$K_i$</td>
<td>2.272</td>
<td>1.06</td>
<td>1.248</td>
<td>-</td>
</tr>
<tr>
<td>$K_d$</td>
<td>1.038</td>
<td>0.773</td>
<td>1.145</td>
<td>-</td>
</tr>
<tr>
<td>$t_s (\pm 2%)$</td>
<td>5.1551</td>
<td>1.8561</td>
<td>3.3306</td>
<td>5.1284</td>
</tr>
<tr>
<td>OS(%)</td>
<td>32.5301</td>
<td>0.1432</td>
<td>0.5994</td>
<td>0</td>
</tr>
<tr>
<td>ISE</td>
<td>6.7537</td>
<td>7.3121</td>
<td>6.2444</td>
<td>14.3867</td>
</tr>
<tr>
<td>F</td>
<td>115.5324</td>
<td>78.8320</td>
<td>73.0352</td>
<td>159.2525</td>
</tr>
</tbody>
</table>

5 Conclusion

Bees algorithm was employed to tune PID controllers for plants of high order. The method optimized PID gains as design variables in both single- and multi-objective approaches. The objective functions taken from literature were important performance indices of ITSE, ISE and IAE as well as overshoot.
and settling time. Results clearly expressed that the utilized method has been successful in comparison to genetic algorithm and Z-N techniques; and can be considered as a powerful tuning scheme for controllers.

REFERENCES


