Available online at www.sciencedirect.com

SciVerse ScienceDirect

Journal of the Franklin Institute 348 (2011) 2795–2812

Journal of The Franklin Institute

www.elsevier.com/locate/jfranklin

Tabu search algorithm based PID controller tuning for desired system specifications

Aytekin Bagis

Department of Electrical and Electronic Engineering, Erciyes University, Kayseri 38039, Turkey

Received 22 April 2008; received in revised form 8 August 2011; accepted 3 September 2011
Available online 12 September 2011

Abstract

This paper presents a tuning approach based on a tabu search algorithm (TSA) to obtain the optimal proportional-integral-derivative (PID) controller parameters in order to achieve a desired transient response. TSA is used to determine the main parameters of the PID controller. The performance of the PID controlled system is examined by considering the characteristics of the step response of the plant. Simulation results demonstrate that the tabu algorithm based approach is one of the useful methods for PID controller tuning, and using by the presented method, performance of the controlled system can be significantly improved according to the given control specifications.

1. Introduction

One of the best-known industrial process controllers is the proportional-integral-derivative (PID) controller because of its important and impressive properties such as fast and efficient control action, simple but functional structure, ease of application, versatility for different type control problems, and robust performance [1–5]. In the design of a PID controller for a process, three parameters have to be specified: proportional gain (*Kp*), integral gain (*Ki*), and derivative gain (*Kd*). The performance of the controller directly depends on these parameters.

There are many approaches and methods to determine PID controller parameters for single input single output systems. Especially, Ziegler–Nichols (ZN) method [6], the Cohen–Coon method [7], the methods based on the use of some performance criterions such as integral square error (ISE), integral of absolute error (IAE), or integral time absolute error (ITAE)

E-mail address: bagis@erciyes.edu.tr

0016-0032/$32.00 © 2011 The Franklin Institute. Published by Elsevier Ltd. All rights reserved.
doi:10.1016/j.jfranklin.2011.09.001

[8,9], internal-model-control (IMC) based method [10,11], and gain-phase margin method [12] are considerably famous and well-known approaches. Some rules for tuning PID controllers based on the transient response characteristics of a given plant element are proposed by Ziegler and Nichols [6]. For automatic tuning of optimum PID controllers, Zhuang and Atherton present the tuning rules based on minimizing an appropriate performance criterion [8]. Ho et al. derived the simple formulas to tune the PID controller for meet gain and phase margins specifications [12]. Wang et al. disclosed another approach based on the model reduction method for designing PID controllers [13]. Lequin et al. [14] gave a model-free technique for the optimization of the parameters of a controller. For industrial control systems, Liu and Daley presented several PID controller design schemes [15]. Schei proposed a method for the automatic tuning of PID controllers based on the estimation of a discrete parametric transfer function model [16]. In Ref. [17], using dimensional analysis and numerical optimization techniques, an optimal method for tuning PID controllers for first order plus time delay systems was presented. Piazzi and Visioli presented a noncausal approach for PID control [18]. The method consisted of using an analytic stable input–output inversion procedure to determine a suitable command signal applied to the closed loop control system, instead of the typical step signal, when the process output is required to assume a new value. In the PID controller tuning method reported by Ramasamy and Sundaramoorthy used the impulse response of the plant for the calculation of PID parameters [19]. Xiong et al. presented equivalent transfer function method for PI/PID controller design of multivariable processes [20]. On the other hand, Coelho proposed a tuning method for determining the parameters of PID controller for an automatic regulator voltage system employing a chaotic optimization approach [21]. In Ref. [22], a direct synthesis method based PID controller design procedure for integrating processes with time delay is presented by Rao et al. Another design procedure based on the use of some tuning rules for PID controllers is described by Eriksson et al. [23].

In addition to the traditional PID tuning techniques, several new methods such as genetic algorithms (GAs) [24–30], tabu search algorithm (TSA) [31,32], differential evolution algorithm (DEA) [33,34], particle swarm optimization (PSO) [35–38], ant colony optimization algorithm [39], artificial bee colony (ABC) algorithm [40,41], evolutionary algorithms (EAs) [42], artificial neural networks (ANNs) [43], and fuzzy systems [44–49] have been developed recently to tune the parameters of the PID controllers. Chang proposed a multi-crossover genetic approach to multivariable PID controllers tuning [25]. A genetic algorithm based multiobjective PID control method for a linear brushless dc motor with modelling uncertainties is reported in [26]. In the study given in [27], a methodology for the optimal PID controller design using the modified GA was proposed to improve the transient stability of AC–DC transmission systems. Another approach based on the use of modified GA was presented by Bagis [28] to find the optimal parameters of the PID controller so that the desired system specifications are satisfied. Karaboga and Kalınlı have presented a TSA based PID controller design method for a second order process [31]. In Ref. [32], for the optimum PID controller parameters, an adaptive tabu search procedure based on the additional use of the back-tracking and the adaptive search radius mechanisms is presented by Puangdownreong and Sujitjorn. In the study, the inequality constraints for the performance specifications are used, and the sum of absolute errors between the system input and output is minimized. To achieve the desired closed loop system response, the results of a DEA based PID tuning study are discussed by Bagis and Savaşçıhabeş [34]. Gaing used a PSO approach for PID design in AVR system [35]. Nasri et al. presented a PSO based optimum design method for speed control of a linear brushless dc motor [37]. In order to tune the PID controller parameters, a comparison of the ABC algorithm, harmony search, and the bees algorithms is presented by Karaboga and Akay [40]. Zhao et al. proposed a rule-based scheme for gain scheduling of PID controllers for process control [44]. The proposed gain scheduling scheme in the work uses fuzzy rules and reasoning to determine the PID controller parameters. A hybrid method for parameter tuning of PID controllers was presented by Wu and Huang [45]. The approach uses different rulebase for different plants, and it has the heavy computation burden of the GA. In the fuzzy-genetic approach given by Bandyopathyay et al. [46], for autotuning a PID controller, a fuzzy inference mechanism based on the Takagi–Sugeno model, and the rulebase derived with the help of a genetic algorithm. Lee and Teng used a fuzzy neural network to develop a formula for designing the PID controller [47]. In the study, PID controller satisfies the criteria of minimum integrated absolute error (IAE) and maximum of sensitivity. In Ref. [48], Bagis and Karaboga presented an evolutionary algorithm based on a systematic neighbourhood structure to obtain the optimum fuzzy PD control of spillway gates of dams. In the other study, a PSO tuning method for the design of fuzzy PID controllers for multivariable systems was given by Ko and Wu [49].

The block diagram of a simplified control system is shown in Fig. 1 [28,34]. In practice the output of a PID controller is given by Eq. (1) [1–5].

$$u(t) = Kp\left\{e(t) + \frac{1}{Ti}\int_0^t e(t)dt + Td\frac{de(t)}{dt}\right\} \tag{1}$$

The transfer function of a PID controller is

$$Gpid(s) = \frac{U(s)}{E(s)} = Kp + \frac{Ki}{s} + Kds = Kp\left(1 + \frac{1}{Tis} + Tds\right) \tag{2}$$

where Kp=proportional gain, Ti=integral time, and Td=derivative time.

At the same time, the discrete-time equivalent expression for a PID controller is given as

$$u(k) = Kpe(k) + KiTs\sum_{i=1}^{n} e(i) + \frac{Kd}{Ts}\Delta e(k) \tag{3}$$

Here, $u(k)$ is the control signal, $e(k)$ is the error between the reference and the process output, Ts is the sampling period for the controller, and $\Delta e(k) = e(k) - e(k-1)$.

The main task of the controller tuning is to succeed high and desirable performance characteristics using the approach of determining the PID controller parameters Kp, Ti, and Td. In order to obtain a desired system response, controller parameters must be optimally adjusted as fast as possible. For this aim, an appropriate approach is to use of time response characteristics of the closed loop system with an optimized PID controller. In this paper, an efficient parameter tuning method is proposed to find the optimal PID controller parameters. The presented method is based on a TSA structure. In this study differently from Ref. [31], there are some important procedures or processes such as the

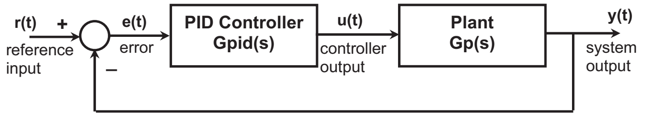

Fig. 1. Block diagram of a basic control system.

structure of the objective or cost function considered for desired system specifications, description of the PID parameters, structure of the tabu list used, detailed investigations to be presented for different processes with different orders. To demonstrate the effectiveness of presented method, the step responses of closed loop system were compared with that of the existing methods in the literature [28,34,44]. In the comparison process, important characteristics of the closed loop systems with the PID controller tuned by different approaches were also given to emphasize the effectiveness of the represented method.

The paper is organized as follows. Section 2 presents the basic definitions of the tabu search algorithm. PID controller tuning by using TSA is given by Section 3. The main parameters of the algorithm, objective function considered by the optimization process, transfer functions of the plants with different order under the control, and simulation results are also given in this section comparatively. Finally, in Section 4, conclusions are presented.

2. Tabu search algorithm

TSA proposed by Glover is an intelligent optimization technique based on the use of neighbours of the solutions [50–59]. Basically, the algorithm has two main properties: a tabu list creation and its application, and a neighbourhood mechanism for the solutions. The search procedure of the algorithm starts from any initial solution and iteratively improves to obtain a better solution via this neighbourhood process. To hinder confining the algorithm in a restricted search area, the main idea of the algorithm is based on the systematically prohibition of the production of some solutions. The aim of this operation is to avoid coming back to recently tested solutions. This critical task is performed by the tabu list acting as a short-term memory. New possible solutions are searched on the neighbour solutions of the current solution. The neighbour solutions obtained by considering the tabu list are compared with the best solution. The process of generating neighbours is repeated and the tabu list is updated in each iteration.

The outline of the TSA used in this paper is given below [50,51,53–56,59]:

```
Initial a solution;
While predetermined stopping criteria not satisfied;
{
Obtain the neighbour solutions;
Evaluate the neighbour solutions;
Choose the best admissible solution;
Test the tabu conditions;
Perform the aspiration criterion;
Update tabu list;
}
```

The basic elements of tabu search are briefly stated and defined as follows [50–59]:

- *Neighbourhood:* The basic element of the tabu algorithm is a neighbourhood mechanism. The neighbourhood of a solution contains the all of the possible different solutions obtained by a move. The move is produced using the current solution vector. Any change in this vector is a neighbour solution of the current solution. The number of the neighbour solutions is based on the number of the possible moves. The characteristics of the moves are changed by the problem type and the description of the solution vector.

- *Tabu List:* To prevent researching again of the previously tested solutions, the moves called as "tabu" of some elements of the solution vector are restricted by the algorithm. Knowledge about the elements is recorded in the tabu list. For an element indicated by the tabu list, the neighbour solutions are not produce. This list is initialized empty. In later iterations of the search, it is updated by considering the tabu conditions.
- *Tabu Conditions:* One of the most important factors of the algorithm is the tabu conditions or restrictions. These conditions are tested for all of the solution vector elements at each iteration. After this process, tabu list is updated. Therefore, number of the neighbour solutions is directly affected by tabu conditions. By carefully defining these conditions, the regeneration of a solution previously obtained is avoided, and the search performance of the algorithm can be improved. The tabu conditions are usually based on two important factors: frequency memory and recency memory [50,51,55,56,59]. The knowledge of how often the same solutions have been made in the past is preserved by frequency memory. On the other hand, the aim of the recency memory is to save the knowledge about the solutions changed in the recent past [51]. In this work, the tabu conditions are used given below: If the element k of solution vector does not satisfy the conditions (i) recency $(k) > (r \times M)$, (ii) frequency $(k) < (f \times \text{avgfreq})$, then it is accepted as tabu and not used to create a neighbour. Here, r and f are recency and frequency factors, respectively, M is the number of elements in the solution vector, and avgfreq is the average frequency value.
- *Aspiration Criterion:* This criterion is used to keep the possible good solutions, even if it is an element of the tabu list. Different aspiration criterions can be defined according to problem type or number of the variables optimized. Hence, the algorithm is avoided from the search operation in restricted or previously tested region. In general, if all available moves are classified as tabu, then a least tabu move (the solution is changed with less recently and frequently) is selected for new solution. Alternatively, to prevent

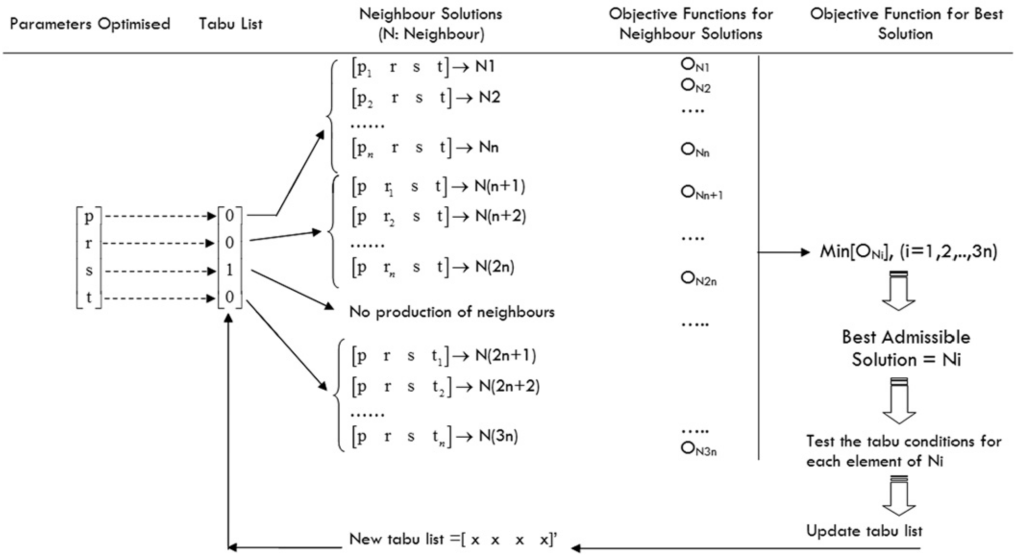

Fig. 2. Neighbour solutions obtained by using tabu list for a system with four parameters.

restricting the process at an unproductive research region, when the all moves are labelled as tabu, all moves or some of them can be freed occasionally.

- *Stopping Criterion*: These are the conditions under which the search process will terminate. In this study, maximum number of iterations reached is selected as stopping criterion of the search process.

For a sample system with four parameters (p, r, s, t), a general illustration about the tabu algorithm is given in Fig. 2. In here, p, r, s, and t are the numerical values that will be attained the neighbours. In this figure, the neighbour solutions of the parameters optimized are obtained by considering the tabu list. According to this list, since the parameter of s is tabu, its neighbours are not obtained. In this case, number of the possible solutions is reduced. This number is directly affected by coding way of the parameters. In this study, in similar to the parameters (p, r, s, t) given in Fig. 2, a definition form with nine parameters ($a1$,$a2$,$a3$,$b1$,$b2$,$b3$,$c1$,$c2$,$c3$) was used as the parameter matrix given below.

3. PID tuning using TSA

3.1. Definition of the PID tuning problem

The proposed PID tuning based on a TSA is schematically shown in Fig. 3. The objective of the TSA programme is to determine the optimal values of the PID controller parameters by minimization of an objective function. The system specifications and closed loop response of the process is used by the TSA during the optimization procedure. Using the changed closed loop control performance according to the adjusted controller parameters at the each generation, the tuning algorithm searches the optimal parameters for the PID controller to satisfy the desired system specifications.

In order to have a good closed loop time response, the following performance function needs to be considered during the design of a PID controller:

$$J = \int_0^{\infty} (y_{step}(t)\text{-}y_{step}^d(t))^2 dt = \int_0^{\infty} E^2(t)dt \tag{4}$$

This function is known as the performance indices integral of the square of the error, ISE [2,4]. In this function, $y_{step}^d(t)$ is the desired step response that may be produced by the transfer function of the process, and $y_{step}(t)$ is the step response of the system with the PID

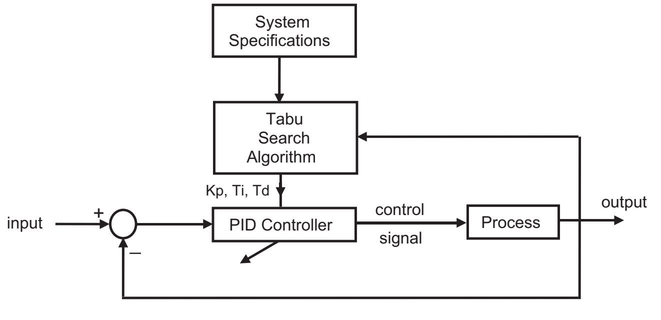

Fig. 3. PID tuning scheme with TSA.

controller. Statistical considerations show that the performance function J is the one of most appropriate methods to measure of the performance of a system. A system is considered as an optimum control system when the system parameters are adjusted so that the index reaches an extremum value, commonly a minimum value. But, a solution based on the minimization of only Eq. (4) may not give good control results for the system. For this case, some other performance functions should be taken into account.

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 O to be minimized in this work can be defined using the performance indices integral of the square of the error (ISE), the response overshoot (OS) and the 5 percent settling time ts [28,34].

$$O = \alpha(ISE) + \beta ts + \delta(OS) \tag{5}$$

Here, the variables of α, β, and δ are the improvement factors. In time domain, the objective function can be formed by different performance specifications such as ISE, IAE [2,4], rise time, settling time, overshoot, and steady state error. Number of these factors and structure of the function can be determined according the problem type. In addition to these factors, response of the system can be significantly affected at desirable direction by using the adjustment parameters such as α, β, and δ [28]. The values of these parameters are selected by using trial-and-error method. As a general approach, to achieve a minimum performance characteristic (minimum overshoot, minimum settling time, etc.), the adjustment parameter of this characteristic must be increased. In this study, similarly to the values given in Refs. [28] and [34], the values of the parameters α, β, δ were selected as 10, 3, and 1, respectively. In addition to these, system stability is also ensured by the objective function. Namely, time domain response of the stable systems approximates to a finite value. In the objective function optimized by tabu algorithm, all of the time domain chracteristics (ISE, settling time, and overshoot) are minimized. The parameters caused to an instable system are eliminated by the algorithm.

In this study, in order to achieve more effective search, each controller parameter was characterized by the weighted average value of three numerical values $[a;b;c]=[(a1,a2,a3);(b1,b2,b3);(c1,c2,c3)]$ (Fig. 4). These numerical values represent a triangular search area with a weight point of $W(a)=W((a1,a2,a3))$, $W(b)=W((b1,b2,b3))$, or $W(c)=W((c1,c2,c3))$ that describes a normalized PID controller parameter. Thus the definitions of the $Kp=W((a1,a2,a3))$, $Ki=W((b1,b2,b3))$, and $Kd=W((c1,c2,c3))$ can be used. When any numerical value is changed, the weight point and the controller parameter defined by this weight point are also changed. This systematic definition of the parameters contributes that large areas of the solution space are searched and solutions do not get stuck in local optima. As to this definition, 9 numerical values are used for three controller parameters. This means that the number of the parameters to be optimized by the tabu algorithm is 9 ($a1,a2,a3$, $b1,b2,b3$, $c1,c2,c3$) (Fig. 4). This value of 9 is also equal to the size of the tabu list.

The neighbourhood structure used in this paper is given in Fig. 5 [28,59]. In this structure, for a solution with n bits, the number of the neighbour solutions is n, as depicted in Fig. 5. To obtain the neighbour solutions for the best solution, encoding and decoding operations are applied. In encoding and decoding procedures, standard binary coding operation is used for binary string to real value conversion. For this purpose, an adjustment coefficient defined by term of $\{(\mathrm{Vub}-\mathrm{Vlb})/(2^{\mathrm{bits}}-1)\}$ is used [60]. In here, Vub and Vlb are upper and lower bounds of the parameter search interval, "bits" is the bit

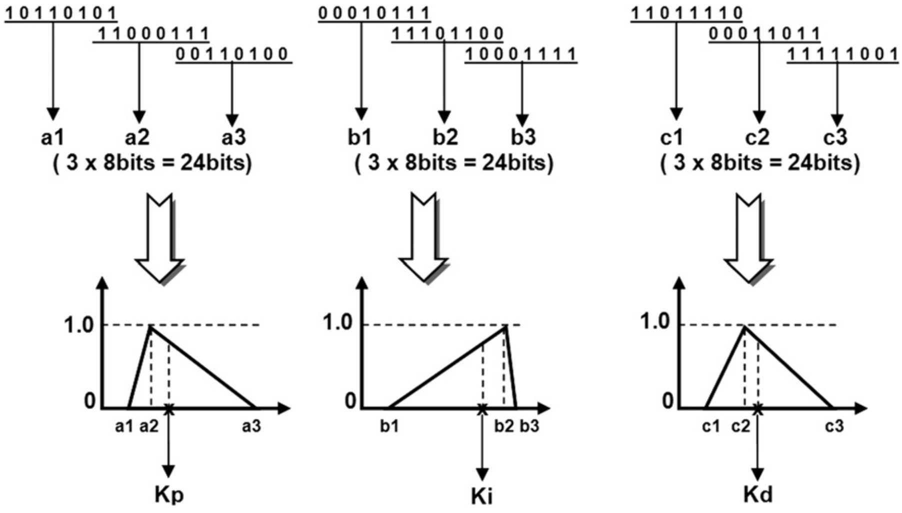

Fig. 4. Description of the PID controller parameters.

	Real-Coded	Binary-Coded (8 bits)	*Decoding*
Best Solution :	3.5882 → *Encoding* →	1 0 1 1 0 1 1 1	
Neighbour Solutions*		0 0 1 1 0 1 1 1	1.0784
		1 1 1 1 0 1 1 1	4.8431
		1 0 0 1 0 1 1 1	2.9608
		1 0 1 0 0 1 1 1	3.2745
		1 0 1 1 1 1 1 1	3.7451
		1 0 1 1 0 0 1 1	3.5098
		1 0 1 1 0 1 0 1	3.5490
		1 0 1 1 0 1 1 0	3.5686

* Each of the possible neighbour solutions was obtained by changing only one bit in the solution vector.

Fig. 5. Neighbourhood structure for a solution.

number for each variable. For instance, if the values of the Vub, Vlb, and bits are 5, 0, and 8, respectively, adjustment coefficient will be value of the 0.01960784. In this case, as shown in Fig. 5, the result of the encoding process for 3.5882 is [10110111] because of the operation of $\{(3.5882/0.01960784) = 182.998 \cong 183 \rightarrow$ *base 10 to base 2 conversion* $\rightarrow$ [10110111]}. Likewise, in the decoding process for the first neighbour solution, the value of the 1.0784 is obtained as a result of the operation of {[00110111] $\rightarrow$ *base 2 to base 10 conversion* $\rightarrow 55 \rightarrow 55 \times 0.01960784 = 1.0784\}$.

The number of elements in a solution vector was represented by 24 bits for each parameter (Fig. 4). Therefore, total number of bits required for defining a PID controller parameter set is [three parameters $\times$ bit number $= 72$]. The parameter space for *Kp*, *Ki*, and *Kd* was in the range between 0 and 5. The value of the algorithm parameters *r*, *f*, and the iteration number were selected as 0.2, 2.0, and 30, respectively. The control parameter values for the TSA are given in Table 1. TSA uses an alternative aspiration criterion to decide whether a solution is freed or not. If all solutions are tabu, then the all-possible moves of the solutions are freed at one point for a new evaluation. If there is no solution better than the best solution found so far, then the other solution with minimum objective function is determined as the next solution. Studies carried out have shown us that, unfortunately, optimum solution cannot be always guarantee in the population based heuristic optimization algorithms. Likewise, some important choices such as optimum control parameters of the algorithm used, optimum population size and/or iteration number are generally determined according to problem type, experience, and knowledge of the users. For this study, TSA was run many times in different iteration numbers, and, it is shown that, the improvement of the solutions was completed within the first 30 iterations.

To illustrate the effectiveness of the presented method, we compared the closed loop response to a step change of a number of simulated systems [28,34,44]. These are methods of the Ziegler–Nichols (ZN), Kitamori's (KT), fuzzy-PID, differential evolution algorithm (DEA), and modified genetic algorithm (MGA). For PID tuning problem, three different processes existing in the literature are considered as given in Eq. (6) [28,29,34,44,56]. For time delay definition in the process G_1, the first order padé approximation ($e^{-\theta s}$» $[1-(\theta s/2)]/[1+(\theta s/2)]$) is used [1, 3, 4, 34, 61].

$$G_1(s) = \frac{e^{-0.5s}}{(s+1)^2} \tag{6a}$$

$$G_2(s) = \frac{4.228}{(s+0.5)(s^2+1.64s+8.456)} \tag{6b}$$

$$G_3(s) = \frac{27}{(s+1)(s+3)^3} \tag{6c}$$

3.2. Simulation results

The simulation results of the processes with different order given in Eq. (6) are shown in Table 2. The time responses for the ZN, MGA, DEA, and TSA based PID controllers are also plotted in Fig. 6. The parameters of the ZN PID controller are obtained by using

Table 1
The values of the control parameters for TSA.

Tabu Search Algorithm	
Recency factor (r)	0.2
Frequency factor (f)	2.0
Iteration number	30
Tabu conditions	(1) recency (k) > ($r \times M$), (2) frequency (k) < ($f \times$ avgfreq)
Number of the variables used for description of the PID parameters	9 ($a1$, $a2$, $a3$ for Kp; $b1$, $b2$, $b3$ for Ki; $c1$, $c2$, $c3$ for Kd)
Tabu list size	9
Parameter space for PID tuning	$Kp \in [0,5]$, $Ki \in [0,5]$, $Kd \in [0,5]$
Bit number for each variable	8
Bit number for each parameter of the PID	24
Number of neighbourhood per parameter	24

$Kp=0.6Ku$, $Ti=0.5Tu$, and $Td=0.125Tu$. In here, Ku and Tu are the gain and the period of oscillation at the stability limit under P-control, respectively. To observe the disturbance rejection behaviours of the processes, impulse responses for different PID controlled systems are given in Fig. 7. Fig. 8 shows the PID parameters determined by the TSA based tuning approach for controlling the fourth-order process ($G_3(s)$). It is noted that this figure contains an impressive detail about the TSA. To obtain the optimum PID parameters, the number of the iterations used by the tabu algorithm is merely 30. In terms of this important property, TSA is too faster than the MGA and DEA provided the good performance characteristics for step response [28,34]. Surely, the main reason for this situation is that the neighbourhood mechanism of the algorithm investigates a large possible solution region during iterations. In the literature to be considered, iteration times and/or time consumed was not selected as a comparison characteristic. Thus, the values of the characteristics mentioned were not stated in the literature compared. However, to denote the computational performance of the tabu algorithm, the computation times per

Table 2
Simulation Results.

Process	Ziegler–Nichols PID controller [44]		Kitamori's PID controller [44][a]		Fuzzy PID controller [44]	
$G_1(s)$	$Kp=2.808$ $Ti=1.64$ $Td=0.41$	$OS=32\%$ $ts=4.16$ IAE=1.37 ISE=0.871	$Kp=2.212$ $Ti=2.039$ $Td=0.519$	$OS=6.8\%$ $ts=2.37$ IAE=1.04 ISE=0.805	$OS=6.0\%$ $ts=3.09$ IAE=1.18 ISE=0.772	
$G_2(s)$	$Kp=2.19$ $Ti=1.03$ $Td=0.258$	$OS=17\%$ $ts=5.45$ IAE=0.99 ISE=0.526	– – –		$OS=6.1\%$ $ts=5.01$ IAE=1.01 ISE=0.533	
$G_3(s)$	$Kp=3.072$ $Ti=1.352$ $Td=0.338$	$OS=32.8\%$ $ts=3.722$ IAE=1.13 ISE=0.628	$Kp=2.357$ $Ti=1.649$ $Td=0.414$	$OS=10.9\%$ $ts=2.3$ IAE=0.833 ISE=0.596	$OS=1.9\%$ $ts=2.632$ IAE=0.811 ISE=0.537	
Process	MGA based PID Controller [28]		DEA based PID controller [34]		TSA based PID controller	
$G_1(s)$	$Kp=2.391$ $Ti=2.23$ $Td=0.61$	$OS=0.05\%$ $ts=1.34$ IAE=0.935 ISE=0.732	$Kp=2.38$ $Ti=2.125$ $Td=0.6294$	$OS=0.262\%$ $ts=1.311$ IAE=0.912 ISE=0.725	$Kp=2.4118$ $Ti=2.1923$ $Td=0.6315$	$OS=0.06\%$ $ts=1.283$ IAE=0.912 ISE=0.722
$G_2(s)$	$Kp=1.637$ $Ti=1.697$ $Td=0.237$	$OS=3.4\%$ $ts=2.89$ IAE=1.189 ISE=0.664	$Kp=2.71$ $Ti=2.1594$ $Td=0.9018$	$OS=1.982\%$ $ts=3.383$ IAE=0.929 ISE=0.401	$Kp=2.765$ $Ti=2.1907$ $Td=0.877$	$OS=1.922\%$ $ts=3.386$ IAE=0.918 ISE=0.399
$G_3(s)$	$Kp=1.772$ $Ti=1.670$ $Td=0.436$	$OS=0.16\%$ $ts=1.65$ IAE=0.95 ISE=0.68	$Kp=2.185$ $Ti=1.6886$ $Td=0.546$	$OS=1.141\%$ $ts=1.257$ IAE=0.816 ISE=0.563	$Kp=2.027$ $Ti=1.814$ $Td=0.4744$	$OS=0\%$ $ts=1.4298$ IAE=0.895 ISE=0.618

[a]The PID parameters of the Kitamori's controller are not available for the process **$G_2(s)$**. OS is the percent maximum overshoot, ts is the 5 percent settling time, and IAE, ISE are the integral of the absolute error ($IAE=\int_0^\infty |E(t)|dt$) and the integral of the squared error, respectively.

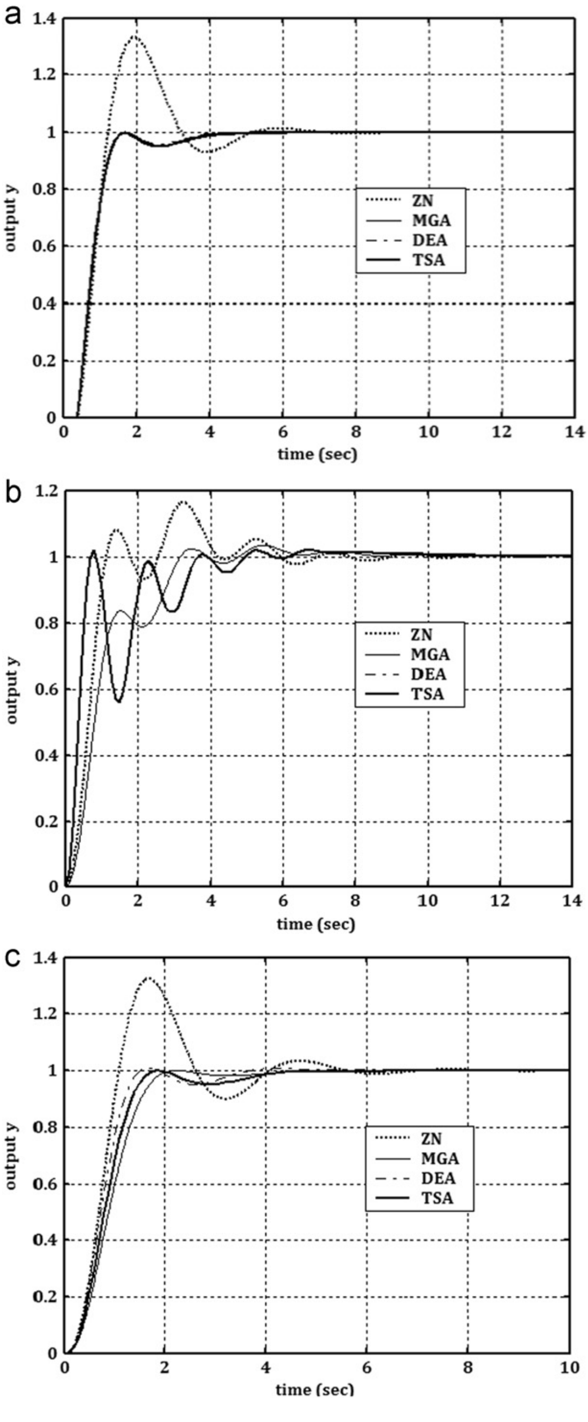

Fig. 6. Comparison of step responses of the controlled (a) second-order process, (b) third-order process, and (c) fourth-order process.

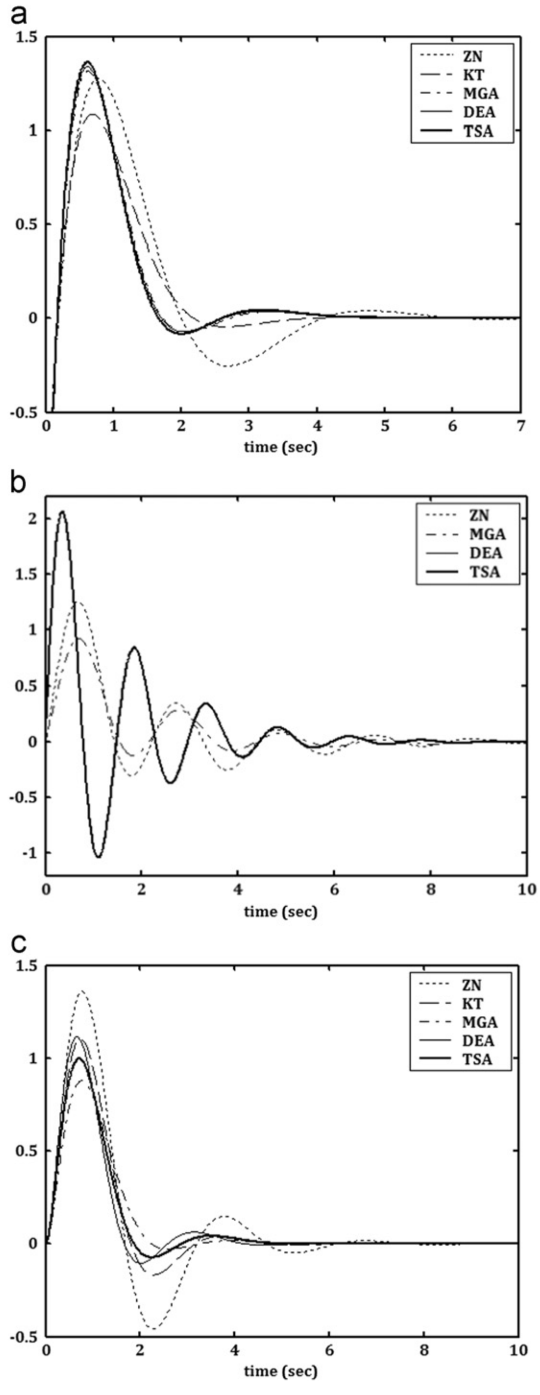

Fig. 7. Comparison of impulse responses of the controlled (a) second-order process, (b) third-order process, and (c) fourth-order process.

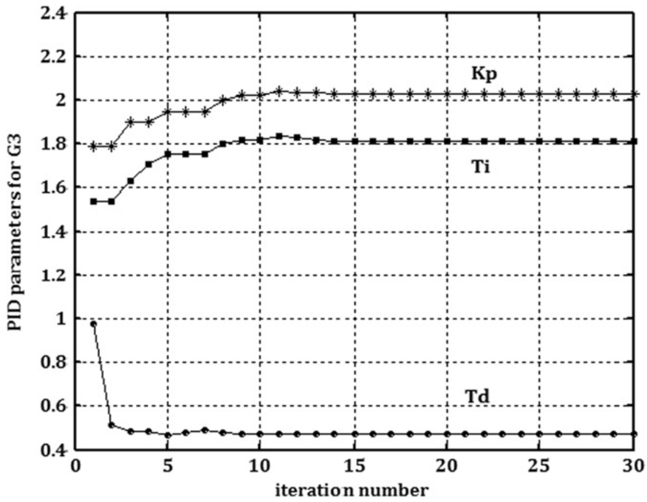

Fig. 8. PID parameters determined by the TSA based tuning approach for fourth-order process.

evaluation (or iteration) of the algorithm are given as a note. For the random initial solutions, the average values of these times of the systems with G_1, G_2, and G_3 are 1.43 s, 1.45 s, and 1.48 s, respectively. In the simulation, Matlab programming package [3,4,61,62] and Mobile Dual Core Intel Pentium M, 2000 MHz computer were used.

In this study, the performance specifications of the step response in time domain were considered by the algorithm. Performance improvements for such a problem were demonstrated by minimization of the settling time of the system's step response, and eliminate of the undesirable overshoots and value of the ISE. For PID tuning considerations, measurements of the robustness were not utilized in the objective function employed by the algorithm. However, some important measurements of the robustness for the PID controlled plants were also presented in this section. These are gain margin (*GM*), phase margin (*PM*), maximum sensitivity (*Ms*), and maximum complementary sensitivity (*Mp*) given in the literature [3,5,36,47]. The sensitivity functions adopted for the calculations are given in Eq. (7). PID controller parameters and robustness measurements of different PID controlled processes are listed in Table 3.

$$Ms = \max_{0 \le w < \infty} |S(jw)| = \max_{0 \le w < \infty} \left| \frac{1}{1 + G(jw)G_{pid}(jw)} \right| \tag{7a}$$

$$\mathrm{Mp} = \max_{0 \le w < \infty} |T(jw)| = \max_{0 \le w < \infty} \left| \frac{G(jw)G_{pid}(jw)}{1 + G(jw)G_{pid}(jw)} \right| \tag{7b}$$

As clearly seen from the Table 2, the TSA, MGA, and DEA based values lead to significant improvement in the performance of the PID controlled system. In the design of a PID control system, there is a trade off generally between the fast dynamic response of the system and its overshoot characteristics. However, using the presented approach, both the settling time and the overshoot can be simultaneously reduced. This is a very important property for a fast and efficient PID controller tuning operation. This fact is also verified by the objective functions of the systems. In the MGA based controller, the values of the objective function O for the plants of G_1, G_2, and G_3 are obtained as 11.39, 18.71, and

11.91, respectively [34]. When the DEA based PID parameters are used, these values are 11.44, 16.132, and 10.55, respectively [34]. On the other hand, the values of the objective function O for TSA based systems are calculated as 11.1282, 16.0679, and 10.4777, respectively.

For process G_1, the values of the PID parameters obtained using MGA, DEA, and TSA and step response characteristics of the PID controlled systems are very close to each other (Table 2). Thus, dynamic behaviours of the systems in time domain are almost identical as shown in Fig. 6a. On the other hand, step response specifications of the TSA based control system are slightly improved compared to the MGA and DEA based systems. In another study using TSA given by Ref. [31], the PID parameters only for process G_1 were obtained as $Kp=2.194$, $Ki=1.209$, and $Kd=1.465$. And, step response specifications for these parameters were given as $OS=5.5\%$, $ts=1.8$ s, IAE=0.995, and ISE=0.741 [31]. It is evident that the results of our study given in Table 2 are more successful than the other study for desired system specifications in here. As clearly shown from Table 2 and Fig. 6b, in particular, there is a remarkable similarity between the DEA and TSA based PID controlled systems for process G_2. Step response specifications for these two systems are quite good as compared to the other systems. In the meantime, high oscillation number and/or high variation rate in the peak values of the step response are also observed as the undesirable behaviours (Fig. 6b). From Table 2 and Fig. 6c, for process G_3, the performance of TSA based PID controlled system is quite satisfactory according to objective function considered. In this study, most considerable improvement has been achieved by applying TSA in the maximum overshoots (Table 2, Fig. 6). For the plants of G_1, G_2, and G_3 as the best values, they are found that the percent maximum overshoots are 0.06, 1.922, and 0.0, respectively. In addition to this, the values of the ISE parameter obtained as 0.722, 0.399, and 0.618 for the same processes by TSA are also quite attractive and preferred values. Consequently, we can candidly say that the performance improvement aimed by the objective function used in this study is achieved by using TSA. In here, it would be appropriate to note that the most obvious advantage of TSA is the fast convergence behaviour within short iteration numbers according as the quality of initial solution.

It would be seen from Table 2 and Fig. 6 that the presented method results are better and more desirable than the Z–N tuning method. On the other hand, it can be seen from Fig. 7 that the swing in the reference signal is clearly suppressed with the TSA based controller. As shown in Fig. 7b, there is harmoniousness between the impulse response and oscillatory step response of the TSA based system. In here, it is possible to note the undesirable high overshoots of the impulse response as a negative property that is in need of improvement. Nevertheless, the PID controlled system optimized by the TSA, at least as fast as the other systems in damping of the undesirable oscillations.

It is well known fact that the sensitivity functions of $S(jw)$ and $T(jw)$ must be small in a robust system [3,5]. To improve the disturbance rejection and stability margin, these functions must be small over a wide frequency range. However, on the other hand, the sum of the sensitivity function and the complementary sensitivity function is always unity. That is, we have the relationship $S(jw)+T(jw)=1$, for all w. Therefore, it impossible to make both $S(jw)$ and $T(jw)$ large or both small at the same frequency. It is observed from the Table 3 that the TSA based controller provides good robustness characteristics in general. Especially, for process G_2, as a natural result of the step response with high oscillations, maximum sensitivity function Ms and complementary sensitivity function Mp of the presented method are no smaller than the others. For the processes G_1 and G_3, the values of the *Mp* and *Ms* are close to or smaller than the other systems' values. On the other hand, the values of the gain and phase margins are reasonable large in general. It is known that a smaller *Mp* result in a more robust system with a smaller overshoot. Similarly, a small *Ms* means that the controlled system has a good disturbance rejection performance. Although considerations of the robustness are not contemplated as a main aim in this study, the satisfying robustness characteristics are obtained using the TSA based PID controller. However, it is an obvious fact that the robustness performance of the system needs to be improved.

There are two important objectives of this paper: The first aim is to correctly determine the parameters of the controller ensuring the desired system behaviour according to an objective function. The other one is to investigate the effectiveness of the TSA for different systems given in the literature. These objectives are also the main contribution of this study. Thus, the simultaneously achievement of the desired system specifications was intended. Instead of the design definitions smaller than some certain limits, the specifications as small as possible were preferred. Using a different description approach for PID parameters, the obtainment of the solution diversity and the improvement of the search efficiency was aimed.

Consequently, we can clearly say that TSA based parameter tuning approach can provide an important and effective tuning method in design of the PID controllers. Using a formative and/or improvable initial solution, and an efficient neighbourhood mechanism for the possible solutions, this method can be successfully employed for improved of the transient response in a PID controlled system. Furthermore, it is possible that better robust stability properties can be achieved with properly included frequency specifications into the algorithm. We believe that this study will provide further insights on the effectiveness of the tabu algorithm.

Table 3
PID parameters [28,34,44] and robustness measurements of different processes.

Process	**Controller**	**PID parameters**			**Measurements of robustness**			
		Kp	***Ti***	***Td***	***Ms***	***Mp***	***GM***	***PM* (deg.)**
G_1(s)	ZN	2.808	1.640	0.410	1.9039	1.4017	2.9993	41.5727
	Kitamori	2.212	2.039	0.519	1.5811	1.0000	3.5601	60.6627
	MGA	2.391	2.230	0.610	1.6762	1.0000	3.0265	64.3476
	DEA	2.38	2.125	0.6294	1.6855	1.0000	2.9880	64.7666
	TSA	2.4118	2.1923	0.6315	1.7030	1.0000	2.9405	64.6991
G_2(s)	ZN	2.190	1.030	0.258	2.0683	1.3270	2.8434	71.3214
	Kitamori	–	–	–	–	–	–	–
	MGA	1.637	1.697	0.237	1.6678	1.0000	3.8965	84.0372
	DEA	2.71	2.1594	0.9018	2.256	1.8884	Inf	31.2356
	TSA	2.765	2.1907	0.877	2.2732	1.9048	Inf	30.975
G_3(s)	ZN	3.072	1.352	0.338	2.0364	1.5427	3.2885	38.6822
	Kitamori	2.357	1.649	0.414	1.5938	1.0385	4.6345	57.7871
	MGA	1.772	1.670	0.436	1.4156	1.0000	6.2002	68.4615
	DEA	2.185	1.6886	0.546	1.5105	1.0000	4.9358	69.3921
	TSA	2.027	1.814	0.4744	1.4807	1.0000	5.4231	69.1870

4. Conclusion

In this paper, the results of a tabu search algorithm based PID controller tuning approach are presented to obtain the desired closed loop system specifications. The method has been tested on various processes given in the literature by using the step response characteristics of the systems. Simulation results showed that using by the presented tuning approach, the gain parameters of a PID controller could be optimally obtained to improvement in the performance of the control system. Thus, tabu algorithm based PID controller can successfully manage the different processes and provide better transient response and satisfactory robustness measurements under the predetermined control conditions.

References

[1] B.C. Kuo, Automatic Control Systems, 6th ed., Prentice Hall, 1991.
[2] K.J. Åström, T. Hägglund, PID Controllers: Theory, Design, and Tuning, 2nd ed., Instrument Society of America, New York, USA, 1995.
[3] K. Ogata, Modern Control Engineering, Prentice Hall, 1997.
[4] R.C. Dorf, R.H. Bishop, in: Modern Control Systems, 8th ed., Addison-Wesley Longman, Inc, 1998.
[5] R.S. Burns, in: Advanced Control Engineering, Butterworth-Heinemann, 2001.
[6] J.G. Ziegler, N.B. Nichols, Optimum settings for automatic controllers, Transactions of ASME 64 (1942) 759–768.
[7] G.H. Cohen, G.A. Coon, Theoretical consideration of retarded control, Transactions of the ASME 75 (1953) 827–834.
[8] M. Zhuang, D.P. Atherton, Automatic tuning of optimum PID controllers, IEE Proceedings on Control and Applications 140 (3) (1993) 216–224.
[9] D.W. Pessen, A new look at PID-controller tuning, Transactions of the ASME, Journal of Dynamical Systems Measures and Control 116 (1994) 553–557.
[10] D.E. Rivera, M. Morari, S. Skogestad, Internal model control. 4. PID controller design, Industrial and Engineering Chemistry Process Design and Development 25 (1986) 252–265.
[11] M. Morari, E. Zafiriou, in: Robust Process Control, Prentice-Hall, Englewood Cliffs NJ, 1989.
[12] W.K. Ho, C.C. Hang, L.S. Cao, Tuning of PID controllers based on gain and phase margin specifications, Automatica 31 (3) (1995) 497–502.
[13] Q.G. Wang, T.H. Lee, H.W. Fung, Q. Bi, Y. Zhang, PID tuning for improved performance, IEEE Transactions on Control Systems Technology 7 (4) (1999) 457–465.
[14] O. Lequin, M. Gevers, M. Mossberg, E. Bosmans, L. Triest, Iterative feedback tuning of PID controllers: comparison with classical tuning rules, Control Engineering Practice 11 (2003) 1023–1033.
[15] G.P. Liu, S. Daley, Optimal-tuning PID. control for industrial systems, Control Engineering Practice 9 (2001) 1185–1194.
[16] T.S. Schei, Automatic tuning of PID controllers based on transfer function estimation, Automatica 30 (12) (1994) 1983–1989.
[17] S. Tavakoli, M. Tavakoli, Optimal tuning of PID controllers for first order plus time delay models using dimensional analysis, in: Proceedings of the Fourth International Conference on Control and Automation (ICCA'03), Montreal, Canada, 2003.
[18] A. Piazzi, A. Visioli, A noncausal approach for PID control, Journal of Process Control 16 (2006) 831–843.
[19] M. Ramasamy, S. Sundaramoorthy, PID controller tuning for desired closed-loop responses for SISO systems using impulse response, Computers and Chemical Engineering 32 (2008) 1773–1788.
[20] Q. Xiong, W.J. Cai, M.J. He, Equivalent transfer function method for PI/PID controller design of MIMO processes, Journal of Process Control 17 (2007) 665–673.
[21] L. Coelho, Tuning of PID controller for an automatic regulator voltage system using chaotic optimization approach, Chaos, Solitons and Fractals 39 (2009) 1504–1514.
[22] A.S. Rao, V.S.R. Rao, M. Chidambaram, Direct synthesis-based controller design for integrating processes with time delay, Journal of the Franklin Institute 346 (2009) 38–56.

[23] L. Eriksson, T. Oksanen, K. Mikkola, PID controller tuning rules for integrating processes with varying time-delays, Journal of the Franklin Institute 346 (2009) 470–487.
[24] P. Wang, D.P. Kwok, Optimal design of PID process controllers based on genetic algorithms, Control Engineering Practice 2 (4) (1994) 641–648.
[25] W.D. Chang, A multi-crossover genetic approach to multivariable PID controllers tuning, Expert Systems with Applications 33 (2007) 620–626.
[26] C.L. Lin, H.Y. Jan, N.C. Shieh, GA based multiobjective PID control for a linear brushless dc motor, IEEE/ASME Transactions on Mechatronics 8 (1) (2003) 56–65.
[27] Y.P. Wang, N.R. Watson, H.H. Chong, Modified genetic algorithm approach to design of an optimal PID controller for AC-DC transmission systems, Electrical Power and Energy Systems 24 (2002) 59–69.
[28] A. Bagis, Determination of the PID controller parameters by modified genetic algorithm for improved performance, Journal of Information Science and Engineering 23 (2007) 1469–1480.
[29] A. Bağış, A. Savaşçıhabeş, Determination of the PID controller parameters by using binary and real coded genetic algorithm, in: Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications (ASYU 2008), Isparta, Turkey, June 2008, (in Turkish).
[30] Z. Jinhua, Z. Jian, D. Haifeng, W. Sun'an, Self-organizing genetic algorithm based tuning of PID controllers, Information Sciences 179 (2009) 1007–1018.
[31] D. Karaboğa, A. Kalınlı, Tuning PID controller parameters using tabu search algorithm, in: Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, (1996), 134–136.
[32] D. Puangdownreong, S. Sujitjorn, Obtaining an optimum PID controller via adaptive tabu search, in: Proceedings of the International Conference on Adaptive and Natural Computing Algorithms (ICANNGA 2007), Part II, LNCS 4432, (2007), 747–755.
[33] W. Lianghong, W. Yaonan, Z. Shaowu, T. Wen, Design of PID controller with incomplete derivation based on differential evolution algorithm, Journal of Systems Engineering and Electronics 19 (3) (2008) 578–583.
[34] A. Bağış, A. Savaşçıhabeş, PID tuning by using differential evolution algorithm for desired closed loop system response, in: Proceedings of the International Symposium on Innovations in Intelligent Systems and Applications (INISTA 2010), Kayseri & Cappadocia, Turkey, June 2010, 170–174.
[35] Z.L. Gaing, A particle swarm optimization approach for optimum design of PID controller in AVR system, IEEE Transaction on Energy Conversion 19 (2004) 384–391.
[36] J. Zhao, T. Li, J. Qian, Application of particle swarm optimization algorithm on robust PID controller tuning, Lecture Notes in Computer Sciences 3612 (2005) 948–957.
[37] M. Nasri, H. Nezamabadi-poor, M. Maghfoori, A PSO based optimum design of PID controller for a linear brushless dc motor, Proceedings of the World Academy of Sciences, Engineering and Technology 20 (2007) 211–215.
[38] M. Zamani, M.K. Ghartemani, N. Sadati, M. Parniani, Design of a fractional order PID controller for an AVR using particle swarm optimization, Control Engineering Practice 17 (2009) 1380–1387.
[39] D. Hai-bin, W. Dao-bo, Y. Xiu-fen, Novel approach to nonlinear PID parameter optimization using ant colony optimization algorithm, Journal of Bionic Engineering 3 (2006) 073–078.
[40] D. Karaboga, B. Akay, Proportional-integral-derivative controller design by using artificial bee colony, harmony search, and the bees algorithms, Journal of Systems and Control Engineering 224 (7) (2010) 869–883.
[41] H. Gozde, M.C. Taplamacioglu, Comparative performance analysis of artificial bee colony algorithm for automatic voltage regulator (AVR) system, Journal of the Franklin Institute (2011). doi:10.1016/j.jfranklin.2011.05.012.
[42] M.W. Iruthayarajan, S. Baskar, Evolutionary algorithms based design of multivariable PID controller, Expert Systems with Applications 36 (2009) 9159–9167.
[43] P.C. Chen, J.K. Mills, Synthesis of neural networks and PID control for performance improvement of industrial robots, Journal of Intelligent and Robotic Systems 20 (2–4) (1997) 157–180.
[44] Z.Y. Zhao, M. Tomizuka, S. Isaka, Fuzzy gain scheduling of PID controllers, IEEE Transactions on Systems, Man, and Cybernetics 23 (5) (1993) 1392–1398.
[45] C.J. Wu, C.H. Huang, A hybrid method for parameter tuning of PID controllers, Journal of the Franklin Institute 334B (4) (1997) 547–562.
[46] R. Bandyopadhyay, U.K. Chakraborty, D. Patranabis, Autotuning a PID controller: a fuzzy-genetic approach, Journal of Systems Architecture 47 (2001) 663–673.
[47] C.H. Lee, C.C. Teng, Calculation of PID controller parameters by using a fuzzy neural network, ISA Transactions 42 (2003) 391–400.

[48] A. Bagis, D. Karaboga, Evolutionary algorithm based fuzzy PD control of spillway gates of dams, Journal of Franklin Institute 344 (2007) 1039–1055.
[49] C.N. Ko, C.J. Wu, A PSO tuning method for design of fuzzy PID controllers, Journal of Vibration and Control 14 (3) (2008) 375–395.
[50] F. Glover, Tabu search—part I, ORSA Journal on Computing 1 (3) (1989) 190–206.
[51] F. Glover, M. Laguna, Tabu Search, Kluwer Academic Publishers, USA, 1997.
[52] M.A. Abido, Y.L. Abdel-Magid, A tabu search based approach to power system stability enhancement via excitation and static phase shifter control, Electric Power Systems Research 52 (1999) 133–143.
[53] D.T. Pham, D. Karaboga, Intelligent Optimisation Techniques: Genetic Algorithms, Tabu Search, Simulated Annealing and Neural Networks, Springer-Verlag, 2000.
[54] H. Youssef, S.M. Sait, H. Adiche, Evolutionary algorithms, simulated annealing and tabu search: a comparative study, Engineering Applications of Artificial Intelligence 14 (2001) 167–181.
[55] A. Bagis, Determining fuzzy membership functions with tabu search—an application to control, Fuzzy Sets and Systems 139 (1) (2003) 209–225.
[56] A. Bagis, Performance comparison of genetic and tabu search algorithms for system identification, Lecture Notes in Artificial Intelligence, LNAI-I 4251 (2006) 94–101.
[57] M. Srinivas, G.P. Rangaiah, A study of differential evolution and tabu search for benchmark, phase equilibrium and phase stability problems, Computers and Chemical Engineering 31 (2007) 760–772.
[58] A.E. Fallahi, C. Prins, R.W. Calvo, A memetic algorithm and a tabu search for the multi-compartment vehicle routing problem, Computers & Operations Research 35 (2008) 1725–1741.
[59] A. Bagis, Fuzzy rule base design using tabu search algorithm for nonlinear system modeling, ISA Transactions 47 (1) (2008) 32–44.
[60] R.E. King, in: Computational Intelligence in Control Engineering, Marcel Dekker, Inc., NY, 1999.
[61] MathWorks, Control System Toolbox for Use with Matlab-Computation, Visualization, Programming, Using the Control System Toolbox-Version 1, The MathWorks, Inc., (2000).
[62] MathWorks, Control System Toolbox. Available online at: 〈http://www.mathworks.com/help /toolbox/control/〉 (accessed on June 28, 2011).