

RESEARCH ARTICLE

Parameter effect analysis of particle swarm optimization algorithm in PID controller design

Mustafa Sinasi Ayas^{a*}  and Erdinc Sahin^b 

^aDepartment of Electrical and Electronics Engineering, Karadeniz Technical University, Turkey

^bDepartment of Energy Systems Engineering, Karadeniz Technical University, Turkey
msayas@ktu.edu.tr, esahin@ktu.edu.tr

ARTICLE INFO

Article History:

Received 02 August 2018

Accepted 18 January 2019

Available 09 April 2019

Keywords:

PID controller

PSO algorithm

Controller parameter tuning

Error-based objective functions

SOPDT model

AMS Classification 2010:

90C26; 93C05; 78M50; 49N05

ABSTRACT

PID controller has still been widely-used in industrial control applications because of its advantages such as functionality, simplicity, applicability, and easy of use. To obtain desired system response in these industrial control applications, parameters of the PID controller should be well tuned by using conventional tuning methods such as Ziegler-Nichols, Cohen-Coon, and Astrom-Hagglund or by means of meta-heuristic optimization algorithms which consider a fitness function including various parameters such as overshoot, settling time, or steady-state error during the optimization process. Particle swarm optimization (PSO) algorithm is often used to tune parameters of PID controller, and studies explaining the parameter tuning process of the PID controller are available in the literature. In this study, effects of PSO algorithm parameters, i.e. inertia weight, acceleration factors, and population size, on parameter tuning process of a PID controller for a second-order process plus delay-time (SOPDT) model are analyzed. To demonstrate these effects, control of a SOPDT model is performed by the tuned controller and system response, transient response characteristics, steady-state error, and error-based performance metrics obtained from system response are provided.



1. Introduction

Meta-heuristic optimization algorithms have attracted attention in control system area, especially in controller design process. Because these algorithms have been agreed as an alternative method of solving deterministic optimization problems or stochastic programming whose solution is not feasible in most cases even though optimality is proven. Genetic algorithm (GA) [1, 2], ant colony optimization (ACO) [3], cuckoo search algorithm (CSA) [4], flower pollination algorithm (FPA) [5], differential evolution [6], artificial bee colony algorithm (ABC) [8], and particle swarm optimization (PSO) [7, 9–11] are some of meta-heuristic optimization algorithms used in controller design process in control system area.

PID controller has been preferred in industrial control applications to obtain desired transient and steady-state response of the closed loop system since 1940s. Either PID controller or its combinations have been included in almost 90% of the industrial control applications [12] because of its functionality, simplicity, applicability, and easy of use [13]. A PID controller has three different parameters required to optimally tune in order to obtain desired system response. Even though conventional tuning methods such as Ziegler-Nichols [14], Cohen-Coon, and Astrom-Hagglund [15] methods are available in the literature, meta-heuristic optimization algorithms considering a predefined fitness function have become popular in the tuning process.

*Corresponding Author

Although several meta-heuristic optimization algorithms are available, PSO algorithm is commonly used in many control applications [16]. PSO algorithm was used to design an optimum PID controller for an automatic voltage regulator (AVR) system [9] by defining a performance criterion W which contains overshoot, rise time, settling time, and steady-state error obtained from system response. In another study, to show the effectiveness of PSO algorithm in control system area, Solihin et. al designed a PID controller for a DC motor model and compared the results with Ziegler-Nichols tuning method [11]. The authors used various fitness functions to see the effect of fitness function in the optimization process. Zhao et. al [17] designed PID controllers for both a first-order process plus delay-time (FOPDT) model and a second-order process plus delay-time (SOPDT) model by proposing a novel fitness function which provides less overshoot and control input. Speed control of a DC motor was performed using PID controller whose parameters were tuned by PSO algorithm and the performance of the PID controller were compared to another PID controller tuned by differential evolution (DE) algorithm [18]. A comparative study for PID controller design with different algorithms was published in [19] where GA, DE, and PSO algorithms were used to tune the parameters of the PID controller.

Even though other studies related to PID controller design based on PSO algorithm are available in the literature, we have not come across with any study analyzing effects of PSO algorithm parameters, yet. Therefore, variable parameters of PSO algorithm such as inertia weight, acceleration factors, and population size are taken into account and effects of these variables on PID controller parameter optimization process are analyzed in this study. During the optimization process, a SOPDT model is used since most of the high order processes can be modeled by either a FOPDT model or a SOPDT model [20]. Both visual and numerical results are provided in the paper to see the effects of each variable on change of fitness value, transient response characteristics, steady-state error, and error-based performance metrics.

The rest of this paper is organized as follows: Section 2 presents a brief background about PID controller and PSO algorithm, and introduces optimization process of the PID controller. Simulation results are given in Section 3 where the effects of inertia weight, acceleration factors, and population size of PSO algorithm are analyzed

separately. In addition, the obtained system responses are provided in this section. Concluding remarks are made in Section 4.

2. PID controller and optimization

2.1. PID controller

PID controller and its combinations have been still preferred in many control applications to improve system dynamic response in addition to steady-state error, although being over 100 years old. Because it has critical advantages such as functionality, simplicity, applicability, and easy of use [13]. A PID controller has three different terms: proportional, integral, and derivative. Each term has a gain called with the same name. In other words, proportional, integral, and derivative terms have the gains K_P , K_I , and K_D , respectively. In this structure, the integral term increases system type by adding a pole at the origin, whereas the derivative term adds a finite zero to the open loop transfer function. Therefore, both the steady-state error and transient response of the closed-loop system improve thanks to integral and derivative terms, respectively.

Block diagram representation of a parallel form PID controller is demonstrated in Figure 1 where $E(s)$ and $U(s)$ are the error and control signals in Laplace domain, respectively. The general transfer function of a parallel form PID controller in Laplace domain is:

$$U(s) = (K_p + \frac{K_i}{s} + K_d s)E(s) \quad (1)$$

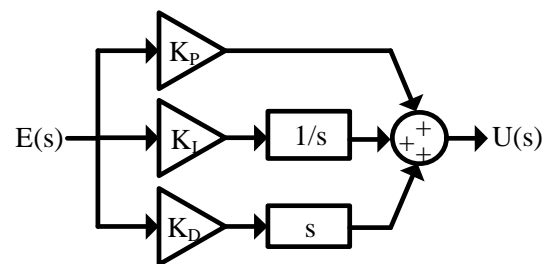


Figure 1. Block diagram of a PID controller.

2.2. Particle swarm optimization

Particle swarm optimization developed by Kennedy and Eberhart [21] is a stochastic evolutionary optimization algorithm based on simulating the movements of a swarm like fish schooling and bird flocking. In order to model the movements of the swarm, position and velocity update equations of the particles are used. The equations of the velocity and position are given below, respectively.

$$V_i^{k+1} = w^k V_i^k + c_1 r_1 (P_{best}^k - X_i^k) + c_2 r_2 (G_{best}^k - X_i^k) \quad (2)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (3)$$

where k is the iteration number, i is the number of the particle, w is the inertia weight that directly effect the velocity, c_1 and c_2 are the acceleration factors called cognition and social constants, respectively, r_1 and r_2 are the random numbers between 0 and 1, P_{best} is the best local solution, G_{best} is the best global solution, and V_i and X_i are the velocity and position of the particle i , respectively.

When considered the velocity and position equations, it can be inferred that population size, inertia weight w , and acceleration factors c_1 and c_2 effect the result of the algorithm. In general, c_1 and c_2 are set to 2 and the inertia weight w which balances the global and local search is linearly decreased from about 0.9 to 0.4 [9].

The implementation of the PSO algorithm is described as follows:

Step 1. Initialize the particles with random velocities and positions.

Step 2. Evaluate and compare fitness values of the particles in the population and obtain the local best value (P_{best}) of the population for current iteration, keep the P_{best} value in memory.

Step 3. Compare the P_{best} value to global best (G_{best}) value, which is initially assigned to P_{best} value, and assign global best (G_{best}) value to the position of the particle with the best fitness function value.

Step 4. Update the velocities of the particles by using Eq. 2

Step 5. Move each particle to their new position by using Eq. 3

Step 6. Increase iteration number, go to step 2 and repeat the steps until the stopping criterion is met.

2.3. Optimization of the controller

Three parameters of the PID controller, i.e. K_P , K_I , and K_D , were tuned by using PSO algorithm to find minimum fitness function value in Matlab/Simulink platform. As the fitness function, integral of time-weighted absolute error (ITAE)

which is as error based fitness function was utilized. ITAE fitness function was preferred because it produces smaller overshoot and oscillations than the other error-based fitness function such as integral of absolute error (IAE) and integral of squared error (ISE) [22, 23]. Although ITAE value was used as the fitness function, both IAE and ISE values were calculated in simulations as error-based performance metrics in addition to ITAE value. The equations of the mentioned metrics are:

$$ITAE = \int_0^t t |e(t)| dt \quad (4)$$

$$IAE = \int_0^t |e(t)| dt \quad (5)$$

$$ISE = \int_0^t e(t)^2 dt \quad (6)$$

where $e(t)$ is the error signal between reference and actual signals.

A general schematic representation of the whole system used in the parameter optimization process of the PID controller is shown in Figure 2. In this process, which was performed with a sampling time T_s of 0.001s, a classical PID controller without filter was utilized. K_P , K_I , and K_D parameters of the PID controller were optimized by PSO algorithm. The optimization process started with initializing the particles with random velocities and positions. The number of the particles were equal to population size and each particle contains three parameters called \hat{K}_P , \hat{K}_I , and \hat{K}_D representing the parameters of the PID controller. As the second step, ITAE values of the particles were evaluated and compared to obtain local best value (P_{best}) of the population for current iteration and kept it in memory. Then, the P_{best} value of the population were compared to global best (G_{best}) value of the population, which is initially assigned to P_{best} value, and the position of the particle with the minimum ITAE fitness function value was assigned as the global best (G_{best}) value. The process went forward with updating the velocities of the particles and obtaining new positions for them. By using these new positions, calculating P_{best} and G_{best} values were repeated during 50 iterations. Since the algorithm starts with randomly assigned velocities and positions, the optimization process was performed 10 times for each analyses explained in detail below, i.e. effect of inertia weight, acceleration parameters,

and population size. The optimization process might be performed more than 10 trials. However, it requires much more time to analyze effects of these three different parameters. Moreover, when the result figures containing fitness value (ITAE) vs. iteration, i.e. Figures 3, 5, and 7, was considered, it was seen that the fitness values generally aggregate to a common fitness value. Hence, we decided that 10 trials were sufficient to rely on the algorithm.

Mean, standard deviation, median, maximum and minimum values of obtained fitness values were considered for the 10 different trials of each analysis and the details are given in the next section.

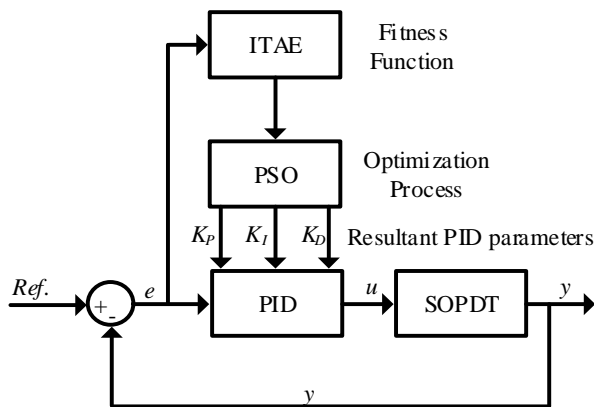


Figure 2. Schematic representation of the parameter optimization process of the PID controller.

3. Simulation results

To analyze effects of PSO algorithm parameters in PID controller design, simulation of a SOPDT model was performed in Matlab/Simulink platform. The transfer function of the SOPDT model used in the study is:

$$G(s) = \frac{0.3172}{s^2 + 1.007s + 0.9515} e^{-0.27s} \quad (7)$$

Unity step reference input was provided to the closed loop system given in Figure 2, and the best controller parameters, i.e. K_P , K_I , and K_D values, were searched considering minimum ITAE fitness function value during 10 seconds simulation.

First of all, effect of inertia weight w was considered in this section. Then effect of acceleration factors c_1 and c_2 were analyzed. Finally, effect of population size used in the PSO algorithm was investigated.

3.1. Effect of inertia weight

The inertia weight w constructs a relation between the past and current velocities of the swarms as seen in Eq. 2. Therefore, it affects the flying abilities of swarms to either in a wide-range or in a narrow-range. Global and local searching abilities are intended by wide and narrow-range flying abilities. Therefore, larger or smaller inertia weight determines the trade-off between global and local searching abilities. Smaller inertia weight is suggested to fine-tune in a smaller search space, whereas larger inertia weight is asked to global exploration in a larger search space [24]. In general, the inertia weight w is linearly decreased from about 0.9 to 0.4 depending on the maximum and current iteration numbers [9]. Eq. 8 is used to calculate the inertia weight

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{iter_{max}} iter_{current} \quad (8)$$

where $iter_{max}$ and $iter_{current}$ represent the maximum and current iteration numbers, respectively. ω_{max} and ω_{min} are set to 0.9 and 0.4, respectively. To see the effect of the inertia value on a SOPDT, various w values were used in optimization process. 11 different w values from 0.05 to 1.0 in addition to linearly decreased w value calculated by using Eq. 8 were utilized in the parameter optimization process of the PID controller. Since the PSO algorithm starts with random initial positions, the algorithm performed 10 times. All trials were performed with stationary acceleration factors c_1 and c_2 , population size, and maximum iteration number which are set to 2, 2, 10, and 50, respectively. The change of ITAE fitness values during the optimization process with different w values are shown in Figure 3. In order to make a fair comparison and provide a more possible visual figures, both the scales of x-axis and y-axis were set to the same possible values in each sub-figures. Therefore, some subfigures may not contain the 10 different trial lines as in Figures 3(a), (b), and (c) which consist of 9 different trial lines. From the figures, the minimum fitness function can not be observed clearly. Therefore, in addition to visual results, numerical results belonging to optimization process are provided in Table 1, where FV, Mean, SD, Median, Max, Min stand for the best fitness value, the mean of the fitness values, the standard deviation of the fitness values, the median of the fitness values, maximum of the fitness values and minimum of the fitness values, respectively. It can be concluded from the table that reaching to optimal solution, i. e. global

best position, is more possible when $w = 0.2$ than the other inertia weights, when considered the standard deviation of the fitness values. Furthermore, the maximum ITAE fitness value obtained with $w = 0.2$ is less than the other inertia weights. As a result of the numerical results, the minimum fitness value is 0.4477 and obtained when $w = 0.2$ for this SOPDT model during 10 seconds.

The best PID controller parameters for each w value are given in Table 2. The obtained system response using these controller parameters are shown in Figure 4, where a zoomed region is also demonstrated to see the transient response characteristics more clearly. From the figure, it can be concluded that satisfactory steady-state error is obtained for each w value. However, transient response obtained for each w is required a further thought. Therefore, the transient response characteristics, i.e. maximum overshoot (MO), settling time (Ts), rise time (Tr), and peak time (Tp), are also provided in Table 2 in addition to steady-state error, and error-based performance metrics, i.e. IAE, ISE, and ITAE. The minimum steady-state error and ITAE performance metric was obtained when $w = 0.2$. However, less overshoot and settling time occurred when $w = 0.6$ and $w = 0.5$, respectively. As a result, since the fitness function of the PSO algorithm is ITAE fitness value, the best controller parameters were obtained when $w = 0.2$.

Table 1. Effect of inertia weight on fitness value.

w	FV	Mean	SD	Median	Max	Min
0.05	0.4532	0.5672	0.2456	0.4946	1.2562	0.4532
0.1	0.4484	0.5775	0.2673	0.5007	1.3333	0.4484
0.2	0.4477	0.5082	0.1399	0.4593	0.9032	0.4477
0.3	0.4484	0.5627	0.2231	0.4644	1.1247	0.4484
0.4	0.4480	0.5420	0.2110	0.4629	1.1247	0.4480
0.5	0.4500	0.5394	0.2064	0.4887	1.1247	0.4500
0.6	0.4517	0.5382	0.2066	0.4797	1.1247	0.4517
0.7	0.4502	0.5541	0.2072	0.4890	1.1263	0.4502
0.8	0.4587	0.5651	0.2202	0.4915	1.1857	0.4587
0.9	0.4673	0.5947	0.2117	0.5300	1.1857	0.4673
1.0	0.5045	0.6177	0.1884	0.5612	1.1316	0.5045
LD	0.4488	0.5367	0.2071	0.4732	1.1247	0.4488

3.2. Effect of acceleration parameters

Effect of acceleration factors c_1 and c_2 on ITAE fitness value and both transient and steady-state response of the SOPDT are considered in this section. c_1 and c_2 constants weight the acceleration ratios of the particles towards the local (P_{best}) and global (G_{best}) best positions, respectively. High acceleration constant values may cause inconsistent movement of the particle, i.e. the particles may suddenly converge to the best global position or past it towards other local best positions,

whereas low values may allow the particle to go around far away from the global best position. Therefore, the acceleration factors c_1 and c_2 are set to 2.0 as a result of trial and error method in most studies.

In this study, (c_1, c_2) acceleration constant pair were set to (0, 4), (1, 3), (2, 2), (3, 1), and (4, 0) by keeping inertia weight w and population size stationary at 0.2 and 10, respectively. w was set to 0.2, since minimum ITAE fitness value had been obtained at that weight in the previous subsection.

The change of ITAE fitness values during the optimization process with different (c_1, c_2) pairs are given in Figure 5 where the scales of each subfigures are set to the same possible values to make a fair comparison and provide a more possible visual figures. Numerical results of the optimization process are also provided in Table 3, where FV, Mean, SD, Median, Max, Min stand for the best fitness value, the mean of the fitness values, the standard deviation of the fitness values, the median of the fitness values, maximum of the fitness values and minimum of the fitness values, respectively. The minimum fitness value is 0.4477 and obtained with $(c_1 = 2.0, c_2 = 2.0)$ for this system during 10 seconds. As a result of the numerical result, c_2 constant, which is related to the global (G_{best}) best position, has more effect on the fitness value as expected. Because the minimum fitness value is obtained with $(c_1 = 2.0, c_2 = 2.0)$ and the increase of the fitness value becomes more when the weight of c_2 constant is decreased.

The best PID controller parameters for each acceleration constant pair are given in Table 4. The obtained system response using these controller parameters are demonstrated in Figure 6, where a zoomed region is also given to see the transient response characteristics more clearly. When considered the numerical results given in Table 4, it can be concluded that the best controller parameters were obtained with $(c_1 = 2.0, c_2 = 2.0)$ pair according to defined fitness value, although the best transient response obtained with $(c_1 = 0, c_2 = 4.0)$.

Table 3. Effect of acceleration parameters on fitness value.

C1	C2	FV	Mean	SD	Median	Max	Min
0	4	0.4492	0.5447	0.2049	0.4890	1.1247	0.4492
1	3	0.4493	0.5011	0.1022	0.4785	0.7873	0.4493
2	2	0.4477	0.5082	0.1399	0.4593	0.9032	0.4477
3	1	0.4507	0.7244	0.2060	0.7580	1.1466	0.4507
4	0	0.5513	2.1092	1.2775	1.5627	4.8274	0.5513

Table 2. Effect of inertia weight on system dynamic response, steady-state error and performance metrics.

Controller Parameters				Transient Response Characteristics				Error-based Metrics			
w	Kp	Ki	Kd	MO (%)	Ts	Tr	Tp	Ess	IAE	ISE	ITAE
0.05	12.5363	3.5191	7.7843	2.2481	2.5800	0.6900	1.5900	2.4e-06	0.8743	0.6988	0.4532
0.1	13.3055	3.5799	8.2060	2.2787	2.5600	0.6600	1.5200	3.7e-06	0.8604	0.6849	0.4484
0.2	13.6308	3.6244	8.3257	2.8301	2.5500	0.6400	1.5000	9.02e-07	0.8552	0.6791	0.4477
0.3	13.7343	3.6662	8.2867	3.6717	2.5500	0.6300	1.4900	3.1e-06	0.8539	0.6770	0.4484
0.4	13.3769	3.6077	8.1810	2.8409	2.5600	0.6600	1.5200	1.9e-06	0.8590	0.6833	0.4480
0.5	14.1622	3.6696	8.6112	3.0230	2.5100	0.6300	1.4500	3.7e-05	0.8485	0.6706	0.4500
0.6	12.6971	3.5296	7.8798	2.1768	2.5800	0.6900	1.5800	3.6e-06	0.8712	0.6958	0.4517
0.7	14.0695	3.6762	8.5284	3.2739	2.5200	0.6300	1.4600	5.0e-05	0.8495	0.6719	0.4502
0.8	14.1778	3.7780	8.2815	5.8517	2.5700	0.6100	1.4700	5.0e-05	0.8523	0.6695	0.4587
0.9	14.7846	3.8085	8.5959	6.0705	2.5300	0.5900	1.4300	1.4e-04	0.8467	0.6607	0.4673
1.0	16.5954	3.8457	10	4.0038	2.8600	0.5500	1.3100	3.0e-04	0.8385	0.6394	0.5045
LD	13.8335	3.6525	8.4119	3.1014	2.5400	0.6400	1.4800	3.3e-05	0.8524	0.6757	0.4488

Table 4. Effect of acceleration parameters on system dynamic response, steady-state error and performance metrics.

Controller Parameters					Transient Response Characteristics				Error-based Metrics			
C1	C2	Kp	Ki	Kd	MO (%)	Ts	Tr	Tp	Ess	IAE	ISE	ITAE
0	4	13.9362	3.6169	8.5811	2.1622	2.5200	0.6400	1.4700	5.6e-06	0.8519	0.6746	0.4492
1	3	13.1337	3.6020	8.0138	3.1200	2.5800	0.6700	1.5400	5.2e-06	0.8631	0.6874	0.4493
2	2	13.6308	3.6244	8.3257	2.8301	2.5500	0.6400	1.5000	9.0e-07	0.8552	0.6791	0.4477
3	1	13.0244	3.6075	7.9132	3.4741	2.5900	0.6700	1.5500	2.4e-06	0.8654	0.6892	0.4507
4	0	11.5784	3.7322	6.6511	8.6182	2.2400	0.6900	1.7100	2.1790e-04	0.9294	0.7175	0.5513

3.3. Effect of population size

In an optimization process, if the population size is small, the algorithm require less computational effort. On the other hand, the probability of premature convergence increases [25]. Therefore, researches have a common idea of that the algorithms provide poor solutions when the population size is small [26] and require more computational effort when it is large [27].

In this section, the effect of population size parameter on the PSO algorithm is considered by keeping inertia weight w and acceleration constant pair stationary at 0.2 and ($c_1 = 2.0, c_2 = 2.0$), respectively. As in the previous subsection, the PSO algorithm is performed for 10 times for all population sizes and the results are recorded.

Figure 7 shows the change of ITAE fitness values during the optimization process with different population sizes. It can be concluded from the figure that the standard deviation of the obtained fitness value is the smallest with the largest population size. In other words, the probability of converging global best position is higher than the others. This result can be supported by Table 5, where FV, Mean, SD, Median, Max, Min stand for the best fitness value, the mean of the fitness values, the standard deviation of the fitness values, the median of the fitness values, maximum of

the fitness values and minimum of the fitness values, respectively. In addition to standard deviation, the maximum value of ITAE value is getting smaller while increasing the population size. On the other hand, the required computation time increases as the population size increase.

The obtained best PID controller parameters for each population size are given in Table 6. The obtained system response using these controller parameters are demonstrated in Figure 8, where a zoomed region is also provided. The transient response characteristics obtained from system response is also given in Table 6. That the best controller parameters were obtained with the largest population size is concluded from the numerical results in the table. In addition to ITAE fitness value, all transient response characteristics except maximum overshoot is better when the population size is equal to 50.

Table 5. Effect of population size on fitness value.

Size	FV	Mean	SD	Median	Max	Min	Time(s)
10	0.4477	0.5082	0.1399	0.4593	0.9032	0.4477	1023.1
25	0.4480	0.4538	0.0130	0.4483	0.4887	0.4480	2521.9
50	0.4477	0.4482	5.8e-04	0.4480	0.4494	0.4477	5375.5

4. Conclusion

In this paper, the effects of inertia weight, acceleration factors, and population size of the PSO algorithm during a PID controller design process for

Table 6. Effect of population size on system dynamic response, steady-state error and performance metrics.

Pop. Size	Controller Parameters			Transient Response Characteristics				Error-based Metrics			
	K _p	K _i	K _d	MO (%)	T _s	T _r	T _p	Ess	IAE	ISE	ITAE
10	13.6308	3.6244	8.3257	2.8301	2.5500	0.6400	1.5000	9.0e-7	0.8552	0.6791	0.4477
25	13.3667	3.6044	8.1821	2.7809	2.5600	0.6600	1.5200	6.7e-7	0.8591	0.6835	0.4480
50	13.6532	3.6306	8.3249	2.9459	2.5500	0.6400	1.4900	4.8e-7	0.8548	0.6787	0.4477

a SOPDT model were separately analyzed. For this SOPDT model, first the effect of the inertia weight was observed and the best PID controller parameters providing minimum ITAE fitness value were obtained with the inertia weight $w = 0.2$. Then, the effect of the acceleration factors were analyzed and it is concluded that the importance of social constant c_2 related to global best solution is higher than the cognition constant c_1 . However, the best solution of the optimization process was obtained with ($c_1 = 2.0, c_2 = 2.0$) for this system. At last, the effect of the population size were considered by keeping the inertia weight w and the acceleration constant pair stationary at 0.2 and ($c_1 = 2.0, c_2 = 2.0$), respectively. It was observed that the probability of converging the global best solution increases as the population size increases. However, the required computation time increases.

References

- [1] Krohling, R. A., & Rey, J. P. (2001). Design of optimal disturbance rejection pid controllers using genetic algorithms. *IEEE Transactions on Evolutionary Computation*, 5(1), 78–82. .
- [2] Wang, P., & Kwok, D. (1994). Optimal design of pid process controllers based on genetic algorithms. *Control Engineering Practice*, 2(4), 641–648.
- [3] Chiha, I., Liouane, N., & Borne, P. (2012). Tuning pid controller using multiobjective ant colony optimization. *Applied Computational Intelligence and Soft Computing*, 2012, 11.
- [4] Ayas, M. S., Altas, I. H., & Sahin, E. (2018). Fractional order based trajectory tracking control of an ankle rehabilitation robot. *Transactions of the Institute of Measurement and Control*, 40(2), 550–564.
- [5] Sahin, E., & Altas, I. H. (2018). Optimized fractional order control of a cascaded synchronous buck–boost converter for a wave-uc hybrid energy system. *Electrical Engineering*, 100(2), 653–665.
- [6] Altinoz, O. T., (2018). Multiobjective PID controller design for active suspension system: scalarization approach. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 8(2), 183–194.
- [7] Erguzel, T. T., (2015). A hybrid PSO-PID approach for trajectory tracking application of a liquid level control process. *An International Journal of Optimization and Control: Theories & Applications (IJOCTA)*, 5(2), 63–73.
- [8] Rajasekhar, A., Jatoth, R. K., & Abraham, A. (2014). Design of intelligent pid/pi λ d μ speed controller for chopper fed dc motor drive using opposition based artificial bee colony algorithm. *Engineering Applications of Artificial Intelligence*, 29, 13–32.
- [9] Gaing, Z.-L. (2004). A particle swarm optimization approach for optimum design of pid controller in avr system. *IEEE transactions on energy conversion*, 19(2), 384–391.
- [10] Zamani, M., Karimi-Ghartemani, M., Sadati, N., & Parniani, M. (2009). Design of a fractional order pid controller for an avr using particle swarm optimization. *Control Engineering Practice*, 17(12), 1380–1387.
- [11] Solihin, M. I., Tack, L. F., & Kean, M. L. (2011). Tuning of pid controller using particle swarm optimization (pso). *International Journal on Advanced Science, Engineering and Information Technology*, 1(4), 458–461.
- [12] Basu, A., Mohanty, S., & Sharma, R. (2016). Designing of the pid and fopid controllers using conventional tuning techniques. *International Conference on Inventive Computation Technologies (ICICT)*, 2, 1–6.
- [13] Ang, K. H., Chong, G., & Li, Y. (2005). Pid control system analysis, design, and technology. *IEEE transactions on control systems technology*, 13(4), 559–576.
- [14] Ziegler, J. G. & Nichols, N. B. (1942). Optimum settings for automatic controllers. *trans. ASME*, 64(11).
- [15] Hägglund, T. & Åström, K. J. (2002). Revisiting the ziegler-nichols tuning rules for pi control. *Asian Journal of Control*, 4(4), 364–380.
- [16] Zhang, Y., Wang, S., & Ji, G. (2015). A comprehensive survey on particle swarm optimization algorithm and its applications. *Mathematical Problems in Engineering*, 2015, 1–39.

- [17] Zhao, J., Li, T., & Qian, J. (2005). Application of particle swarm optimization algorithm on robust pid controller tuning. *International Conference on Natural Computation*, Springer, 948–957.
- [18] Idir, A., Kidouche, M., Bensafia, Y., Khettab, & Tadjer, K., S. (2018). Speed control of DC motor using PID and FOPID controllers based on differential evolution and PSO. *International Journal of Intelligent Engineering and Systems*, 11(4), 241–249.
- [19] Pano, V. & Ouyang, P. R. (2014). Comparative study of ga, pso, and de for tuning position domain pid controller. *2014 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 1254–1259.
- [20] Sun, L., Li, D., Zhong, Q.-C., & Lee, K. Y. (2016). Control of a class of industrial processes with time delay based on a modified uncertainty and disturbance estimator. *IEEE Transactions on Industrial Electronics*, 63(11), 7018–7028.
- [21] Kennedy, J. & Eberhart, R. (1995). Pso optimization. *Proc. IEEE Int. Conf. Neural Networks*, 4, 1941–1948.
- [22] Nagrath, I. (2006). *Control systems engineering*. New Age International.
- [23] Schultz, W. & Rideout, V. (1961). Control system performance measures: Past, present, and future. *IRE Transactions on Automatic Control*, 1, 22–35.
- [24] Shi, Y. & Eberhart, R. C.. (1998), Parameter selection in particle swarm optimization. *International conference on evolutionary programming*, Springer, 591–600.
- [25] Mallipeddi, R. & Suganthan, P. N. (2008). Empirical study on the effect of population size on differential evolution algorithm. *IEEE World Congress on Computational Intelligence Evolutionary*, 3663–3670.
- [26] Piszcz, A. & Soule, T. (2006). Genetic programming: Optimal population sizes for varying complexity problems. *Proceedings of the 8th annual conference on Genetic and evolutionary computation*, 953–954.
- [27] Koumoussis, V. K. & Katsaras, C. P. (2006). A saw-tooth genetic algorithm combining the effects of variable population size and reinitialization to enhance performance. *IEEE Transactions on Evolutionary Computation*, 10(1), 19–28.

Mustafa Sinasi Ayas received BSc (2009) degree in Electrical and Electronics Engineering from Hacettepe University, Turkey. He obtained his MSc (2012) and PhD (2017) degrees from Karadeniz Technical University, Turkey. He is working as an assistant professor at the Department of Electrical and Electronics Engineering, Karadeniz Technical University, Turkey. His research interests includes control systems, parallel robotics, and metaheuristic optimization algorithms.

Erdinc Sahin is at Department of Energy Systems Engineering, Karadeniz Technical University as an assistant professor. He received his B.S. degree in Electrical and Electronics Engineering Department from Eskişehir Osmangazi University, M. S. degree in Control and Automation Department from Istanbul Technical University, and PhD degree in Electrical and Electronics Engineering Department from Karadeniz Technical University. His research interests include renewable energy systems, fractional order system modeling and controller design, power electronics and drive systems, and metaheuristic algorithms.



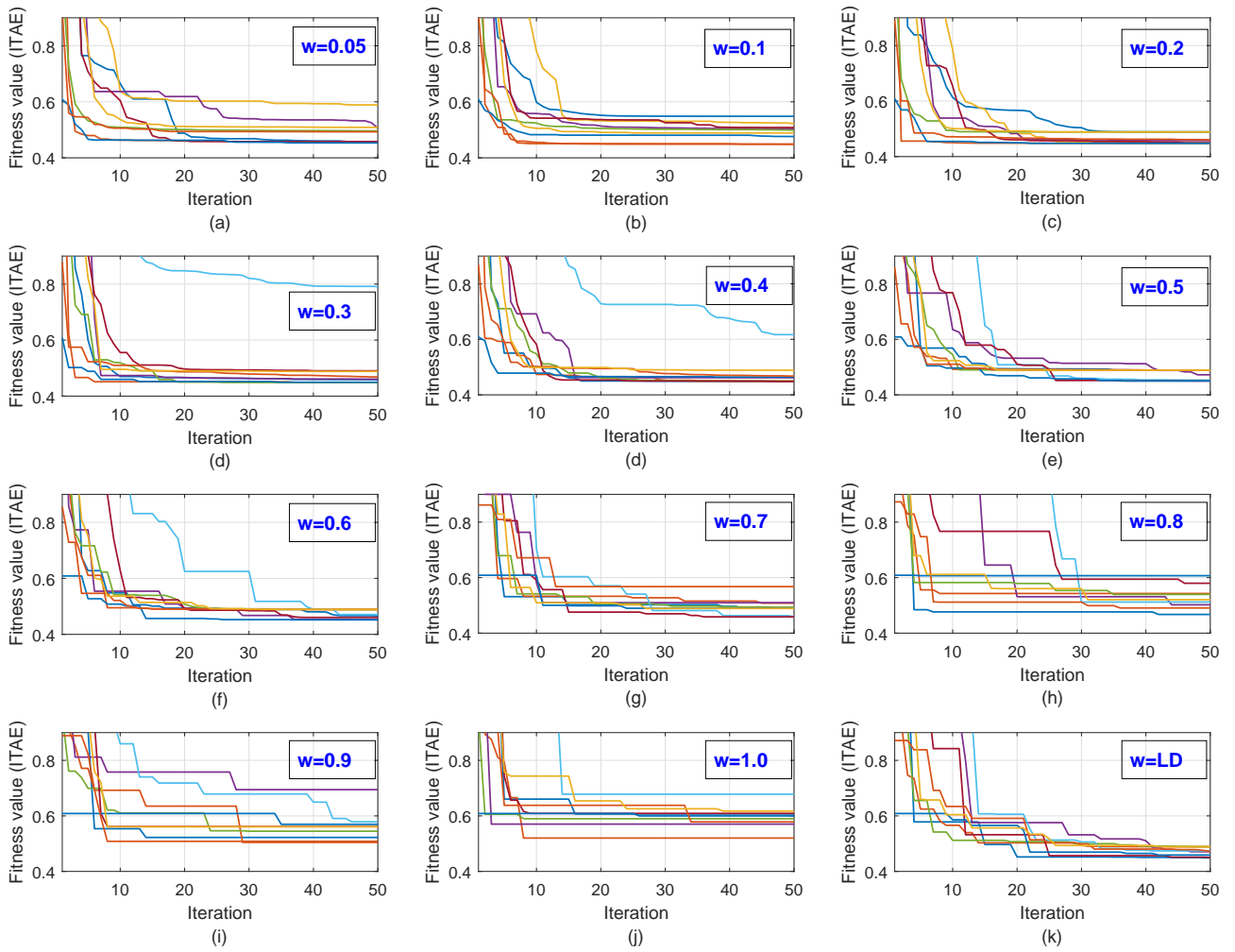


Figure 3. Fitness values obtained with different inertia weight for 10 different trial: Inertia weight is 0.05, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0 and LD (linearly decreased from 0.9 to 0.4) in (a), (b), (c), (d), (e), (f), (g), (h), (i), (j) and (k), respectively.

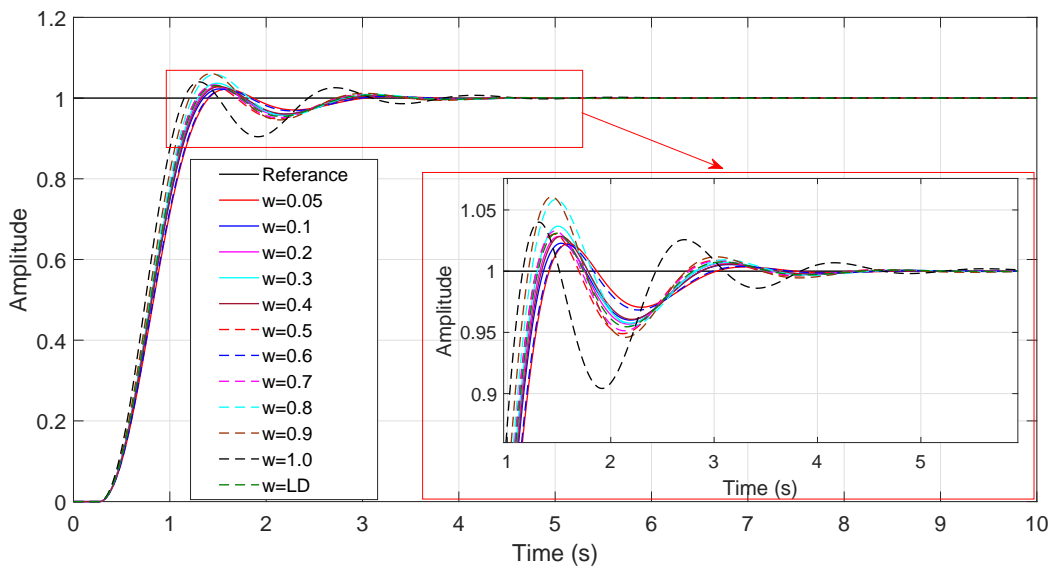


Figure 4. Obtained system responses with different inertia weights.

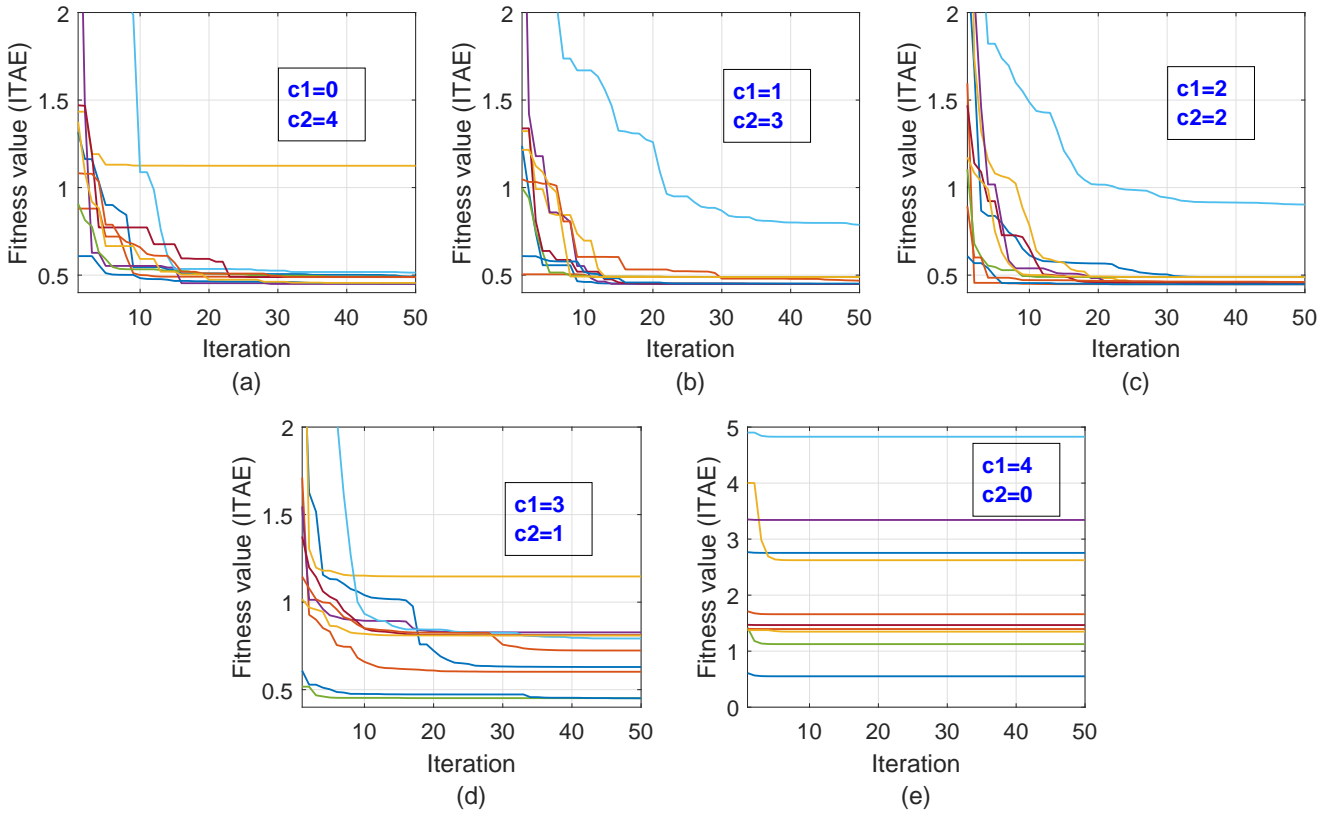


Figure 5. Fitness values obtained with different acceleration parameters for 10 different trial: (a) $c_1=0, c_2=4$; (b) $c_1=1, c_2=3$; (c) $c_1=2, c_2=2$; (d) $c_1=3, c_2=1$; (e) $c_1=4, c_2=0$.

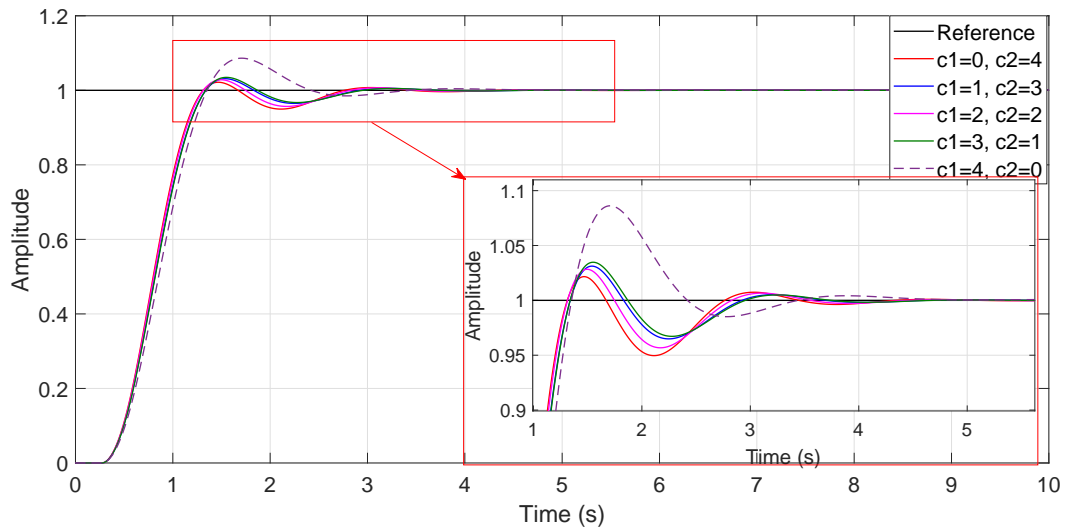


Figure 6. Obtained system responses with different acceleration parameters.

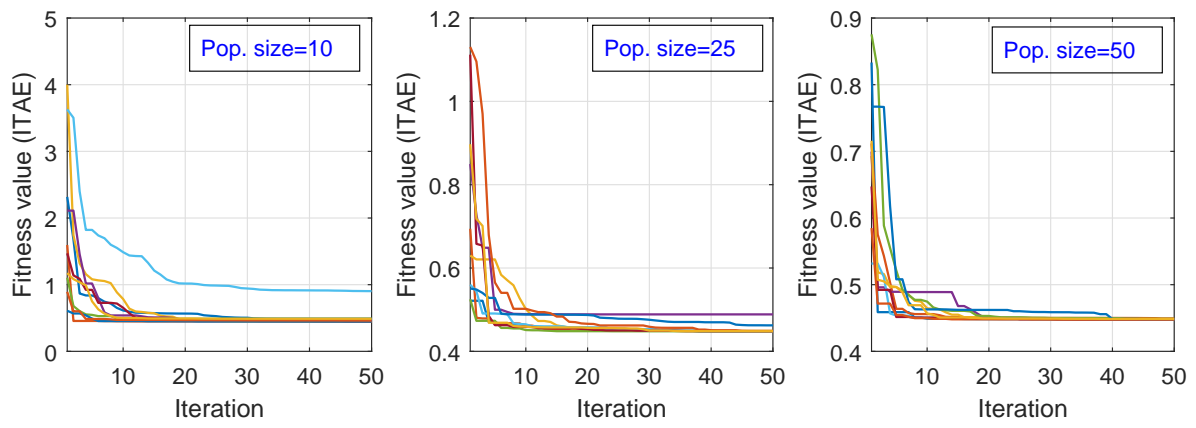


Figure 7. Fitness values obtained with different population size for 10 different trial. Population size is 10, 25, and 50 in (a), (b), and (c), respectively.

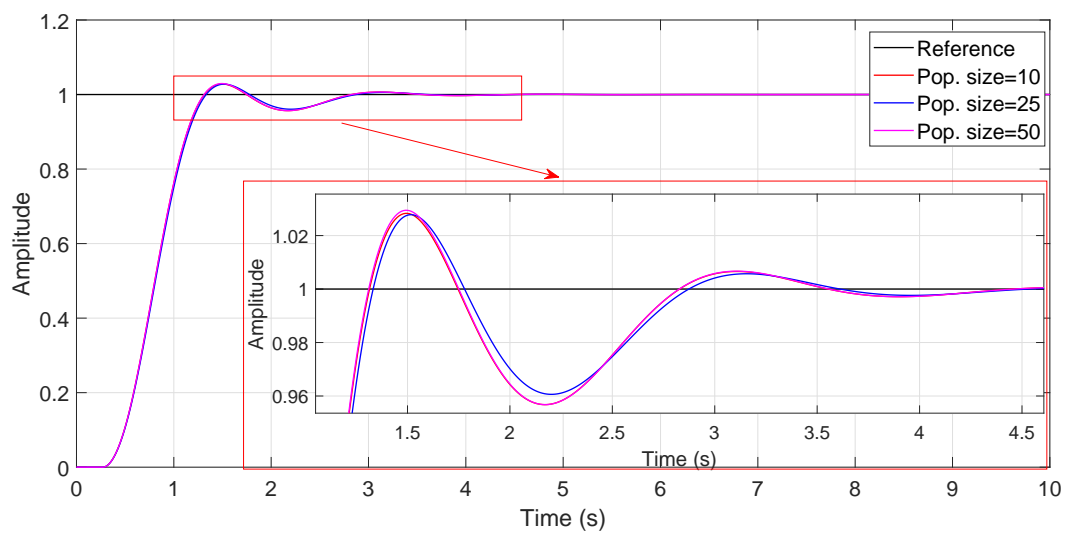


Figure 8. Obtained system responses with different population sizes.