UDC 621
DOI: 10.15587/1729-4061.2021.225383

OPTIMAL PARAMETER VALUES OF PID CONTROLLER FOR DC MOTOR BASED ON MODIFIED PARTICLE SWARM OPTIMIZATION WITH ADAPTIVE INERTIA WEIGHT

Mohammed Obaid Mustafa
PhD
Department of Electrical Engineering
College of Engineering
University of Mosul
Al Majmoaa str., Mosul, Iraq, 41002
E-mail: mohammed.obaid@uomosul.edu.iq

A significant problem in the control field is the adjustment of PID controller parameters. Because of its high nonlinearity property, control of the DC motor system is difficult and mathematically repetitive. The particle swarm optimization PSO solution is a great optimization technique and a promising approach to address the problem of optimum PID controller results. In this paper, a modified particle swarm optimization PSO method with four inertia weight functions is suggested to find the global optimum parameters of the PID controller for speed and position control of the DC motor. Benchmark studies of inertia weight functions are described. Two scenarios have been suggested in order to modify PSO including the first scenario called M1-PSO and the second scenario called M2-PSO, as well as classical PSO algorithms. For the first scenario, the modification of the PSO was done based on changing the four inertia weight functions, social and personal acceleration coefficient, while in the second scenario, the four inertia weight functions have been changed but the social and personal acceleration coefficient stayed constant during the algorithm implementation. The comparison between the presented scenarios and traditional PID was carried out and satisfied simulation results have shown that the first scenario has rapid search speeds, and very effective and fast implementation compared to the second scenario and classical PSO and even improved PSO technique. Moreover, the proposed approach has a fast searching speed compared to classical PSO. However, it has been found that the classical PSO algorithm has a premature, inaccurate and local convergence process when solving complex optimization issues. The presented algorithm is proposed to increase the search speed of the original PSO

Keywords: tuning of PID, particle swarm optimization, DC motor, inertia weight functions

Received date 11.01.2021
Accepted date 03.02.2021
Published date 26.02.2021

1. Introduction

In all engineering and industrial systems, electric drives are typically an essential feature, such as pumps, fans, mills, conveyor belts, elevators, and many others. Indeed, the DC motors are widely used in the mechanisms that require high accuracy control of speed such as biomedical and robotics [1].

The controller of the DC motor may be of any kind, such as Proportional-Integral-Derivative (PID) [2], neural network (NN) [3], fuzzy logic (FL) [4]. Speed control of the DC motor by conventional PID controllers is well known and their objective is a good performance that includes minimization of the steady-state error, minimization of overshoot, minimization of rise time and minimization of settling time. PID controllers are a very efficient solution to obtain the desired output from the plant in a steady state as well for dynamic response [5]. PID controllers are the process industry's most used controllers. It has been recognized that more than 95 % of the control loops are of the PID type of process control [2]. Many different methods have been developed for tuning the PID types of controllers over the past decades. The main task of the PID controller tuning rules focuses on calculating the step response or the process ultimate point such as Ziegler-Nichols, CohenCoon,

Chien-Hrones-Reswick. The scope of our research work was to find tuning rules for PID controllers used for the control of the DC motor. Minorsky [5, 6] first suggested the PID controller in 1922 and applied first for industrial applications in 1939. Due to its simple use, low cost, easy maintaining and easy realization, the PID controller has played the most important role as the heart of control engineering practice in the close loop control systems.

Traditional PID techniques suffer from complex, time-consuming implementation and often generate significant overshooting non-customized performance measure and inadequate process information. To hold basic properties intact, the control system must be configured to face all system disturbances [2]. Incorrect tuning of the PID parameters may lead to slow recovery, poor robustness and poor results and the worst case scenario will be the failure of the system. The main message brought by this paper is to demonstrates how the information from the process step response may be successfully used for finding certain PID parameters, which satisfy a relatively demanding magnitude optimum criterion and for a wide variety of process models [1, 2].

Therefore, the proposed approach finds the optimum values for the PID controller based on modified PSO in order to improve the performance of the control system, reduce the steady-state error percentage to zero, and increase the speed of the system's response.

2. Literature review and problem statement

Many researchers have studied various new control techniques in order to improve the system performance and tuning PID controllers. Several methods have been proposed to determine the values of the PID controller such as Ziegler and Nichols, Cohen and Coon [7, 8]. They have suggested experimental PID tuning techniques relying on trial and error and based on the system response curve [8, 9]. However, when the system is complicated, there can be problems in tuning the PID controller. such as high order, time delay, variable load torque, wide operating periods, non-minimum phase and non-linear processes [8, 9]. For example [7–9], the Ziegler and Nichols method may provide a high order system with big overshoots, highly oscillatory and longer settling time. To solve these challenges and difficulties, various approaches have been proposed to find optimum PID parameters such as [10] meta-heuristic algorithms [11], differential evolution (DE) [12], Flower Pollination Algorithm (FPA) [13], genetic algorithms (GN) [14], Levenberg-Marquardt Algorithm (LMA) [15], Grey Wolf Optimization Algorithm (GWO) [16], Jaya optimization algorithm (JOA) [17], PSO [18, 19], Improved Sine Cosine Algorithm (ISCA) [20]. These are optimization methods that have been introduced to tune the controller parameters for speed control of the DC motor. The common objective of these studies was to use an optimally developed controller using an optimization algorithm to minimize the steady-state error, overshoot, rise time and settling time, thus, to increase the speed control efficiency of the DC motor.

The papers [16, 17] have shown that the presented method suffers from memory capacity and computational cost, local minimum, parameter selection difficulties, and increased calculation time. Furthermore, in [20, 21] are poor although and they are wealthy in terms of exploration, which creates an imbalance between exploration and exploitation. The work presented in [14] is an iterative search approach based on natural genetic mechanism. However, GA is quite fussy; it involves selection, copy, crossover and mutation scenarios and so on. The implementation time became a problem [14], for instance, GA crossover effects typically differ significantly during a run. The members of the population are generally randomized at the start, so that crossover can have major effects, bringing a chromosome into the problem space at a relatively large distance. But there are several popular traits between GA and PSO. First, they are flexible technology for optimization. Second, they've all got a powerful universal property independent of any gradient. However, PSO is a lot simpler than GA, and its service is easier, without any option, copy, crossover [22]. PSO does not suffer from some of GA's difficulties. To solve the same kind of problems as genetic algorithms (GA), particle swarm optimization can be used. Changes in genetic populations contribute to the loss of the problem's prior understanding. The particle swarm system has memory, which the genetic algorithm does not have. In PSO, individuals that move past optima are tugged to return towards them; all particles maintain knowledge of successful solutions. Moreover, the drawbacks of the PSO method are that in high-dimensional space it is simple to collapse into global optimization and has a low convergence speed in the iterative analysis.

Indeed, the classic PSO [23] is not the best method to solve nonlinear engineering problems and it is slow and converges to local optima in some cases. Therefore, to improve the PSO performance, different variants of the algorithm were developed and modified [24, 25]. In 1998, the inertia weight, which plays a vital tool in the PSO process was introduced [25]. The success of PSO depends on values of inertia weight [24]. Therefore, this paper introduces a new modified particle swarm optimization PSO method based on four types of inertia weight functions, which are suggested to find the optimum parameters of the PID controller for speed and position control of the DC motor. Also, the comparison study between the presented schemes, classical PSO algorithms and traditional PID is being carried out to find the better scheme to apply modified PSO for DC motor control.

3. The aim and objectives of the study

The aim of the study is to modify and improve the PSO algorithm by adaptive inertia weight functions Wi, social and personal acceleration coefficient (C1 and C2) in order to find optimum PID parameters for speed and position control of the DC motor.

To accomplish the aim, the following objectives have been set and undertaken:

– to study the comparison between classical PSO and presented modified PSO based on inertia weight functions, social and personal acceleration coefficient;

– to perform time response of the DC motor control based on the modified PSO algorithms;

– to choose better inertia weight functions for the development and modified PSO as part of the PID controller;

– to implement the structure and operating algorithm of modified PSO for the tasks on control over nonlinear objects.

4. Materials and methods

4. 1. Model of DC motor

The DC motor is widely used in many industrial applications such as the control of a pendulum arm and robotics. The electrical equation for the DC motor is presented in equation (1) [18].

$$R_a i_a(t) + L_a \frac{di_a(t)}{dt} + v_b(t) = v_s(t) \tag{1}$$

and

$$v_b(t) = k_b\,\omega_a(t), \tag{2}$$

where R_a is armature resistance in ohm, L_a is armature inductance in henry, $i_a(t)$ is the armature current in Ampere, $v_b(t)$ is the back emf voltage in volt, $v_s(t)$ is the voltage source in volt, k_b is the back emf constant in volt/(rad/sec) and $\omega_a(t)$ is the speed of the motor in rad/sec. The torque equation can be discretion as:

$$T_m - T_{\omega 1} - T_\omega - T_L = 0, \tag{3}$$

where T_m is the electromagnetic torque in N·m, $T_{\omega 1}$ is the torque due to rotational acceleration of the rotor, T_ω is the torque produced from the velocity of the rotor and T_L is the torque of mechanical load.

k_T is the torque constant and it depends on the flux density of the magnets, the reluctance of the iron core and the number of the turn in the armature winding and $T_{\omega 1}$ and T_ω can be written as:

$$T_{\omega 1} = j\frac{d\omega_a(t)}{dt}, \tag{4}$$

$$T_\omega = B\omega_a(t), \tag{5}$$

where j is the moment of inertia of the rotor and the equivalent mechanical load in kg·m²/rad and B is the damping and friction coefficient associated with the mechanical rotational system of the motor in N·m/(rad/sec). Substituting equation (4), (5) in equation (3), we can get the following equation:

$$k_T i_a(t) - j\frac{d\omega_a(t)}{dt} - B\omega_a(t) - T_L = 0. \quad (6)$$

From equation (1) and (6), we can find the transfer function $\left(T\cdot F_1 = \omega_a(s)/v_s(s)\right)$ and $T\cdot F_2 = \omega_a(s)/v_s(s)$ for the DC motor is described in equations (7) and (8):

$$T.F_1 = \frac{k_T}{jL_a S^2 + (L_a B + jR_a)S + (k_b k_T + BR_a)}, \quad (7)$$

$$T.F_1 = \frac{k_T}{S\left(jL_a S^2 + (L_a B + jR_a)S + (k_b k_T + BR_a)\right)}, \quad (8)$$

where $\theta(s)$ is the angular displacement of the rotor.

4. 2. PID controller

One of the most popular techniques of controlling in industrial processes is the PID controller. The PID controller is characterized by its simplicity of execution, good robustness and high performance. The essential principle of the standard PID controller is that the control quantity (t) is the linear combination of proportional-integral-derivative of the difference (t). The description of the PID controller law is explained in the following equation.

$$u(t) = K_p e(t) + K_i \int e(t)\,\mathrm{d}t + K_d \frac{de(t)}{dt}, \quad (9)$$

where $e(t)$ is the system error (difference between the reference input and the system output), $u(t)$ the control variable, K_p, K_i and K_d are the proportional, integral and derivative gain, respectively. Each parameter of the PID controller adds some useful features to the output response of the system.

4. 3. Particle swarm optimization

Particle swarm optimization (PSO) algorithms are widely used in the design of PID controllers because of a simple idea, easy implementation, and fewer parameters to be adjusted [33]. It is a powerful optimization technique that is vital for solving nonlinear problems. PSO is a technique of stochastic population optimization, which imitates the actions of birds when attempting to locate food as a group in a specific area. PSO was first founded in 1995. by R. Eberhart and J. Kennedy based on the behavior of bird flocks. It has a short calculation time and strong results and convergence. Of all stochastic methods, the PSO approach produces a high-quality solution. The basic concept of the PSO algorithm operates by a swarm of particles. In PSO, particles change their locations by moving around in the search space according to some basic criteria.

Fig. 1 presents the movement of particles in the PSO algorithm. The system initially has potential solution known as Particle's initial position. It is currently located at the position denoted as current position. Each particle moves through the problem space with randomly choosing velocity and position. Every particle is following up its best prior position, which is denoted as Personal Best (P_{best}) and its corresponding fitness. There are several P_{best} for each particle in the Swarm with the best of the greatest fitness called the Global Best (G_{best}). The principle concept of PSO is to accelerate each particle to its P_{best} and G_{best} position at each stage of the process. The particle changes the direction of its new movement based on the following:

– its own and swarm's information regarding the best position;

– the direction of its previous movement.

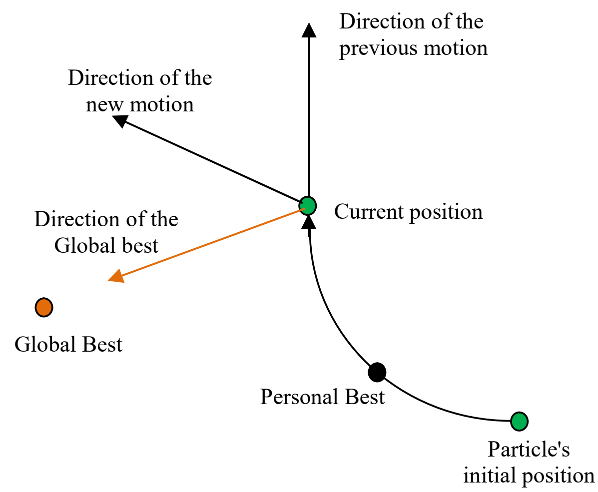

Fig. 1. Motion of particle in PSO algorithm

The modified velocity and position of each particle can be determined using the following equations.

$$v_{ij}^{(t+1)} = w\cdot v_{ij}^t + c_1 r_{1j}^t\left(P_{(best,i)}^t - X_{ij}^t\right) + c_2 r_{2j}^t\left(G_{(best,i)} - X_{ij}^t\right), \quad (10)$$

$$X_{ij}^{t+1} = v_{ij}^{t+1} + X_{ij}^t, \quad (11)$$

where:

– w: intertia coefficient for the velocity of particle i;

– v_{ij}^t, X_{ij}^t: are the velocity and position vector of particle i in dimension j at time t;

– $P_{(best,i)}^t$: is the personal best position of particle i in dimension j found at time t;

– $G_{(best,i)}^t$: is the global best position i of particle in dimension j at t;

– C_1 and C_2: are positive acceleration constants, which are used to the cognitive and social components, respectively;

– r_{1j}^t and r_{2j}^t: are random numbers between 0, 1;

– D: max No. of dimensions;

– P: max. No of particles;

– N: max No. of iterations;

– $v_{ij}^{G_{best}}$: velocity based on G_{best};

– $v_{ij}^{P_{best}}$: velocity based on P_{best};

– f_{ij}: objective function.

The PID controller realizes the specified output of the system by modifying three parameters K_p, K_i and K_d. From the optimization point of view, an optimal solution is to be found in the vector space of three parameters K_p, K_i and K_d in order to make the system optimal. The configuration and structure of the proposed PSO-PID controller are shown in Fig. 2. In the proposed PSO method, each particle contains three members P, I and D. This means that the search space has three dimensions and particles must 'fly' in a three-dimensional space. The flowchart of the PSO-PID control system is shown in Fig. 3.

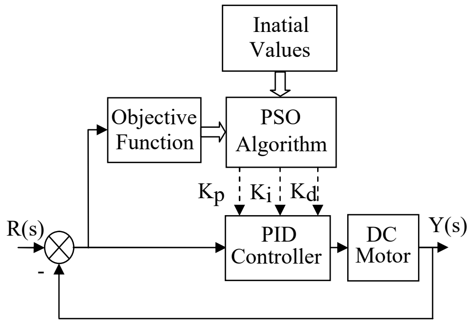

Fig. 2. Overall control design of DC motor based on PSO-PID controller

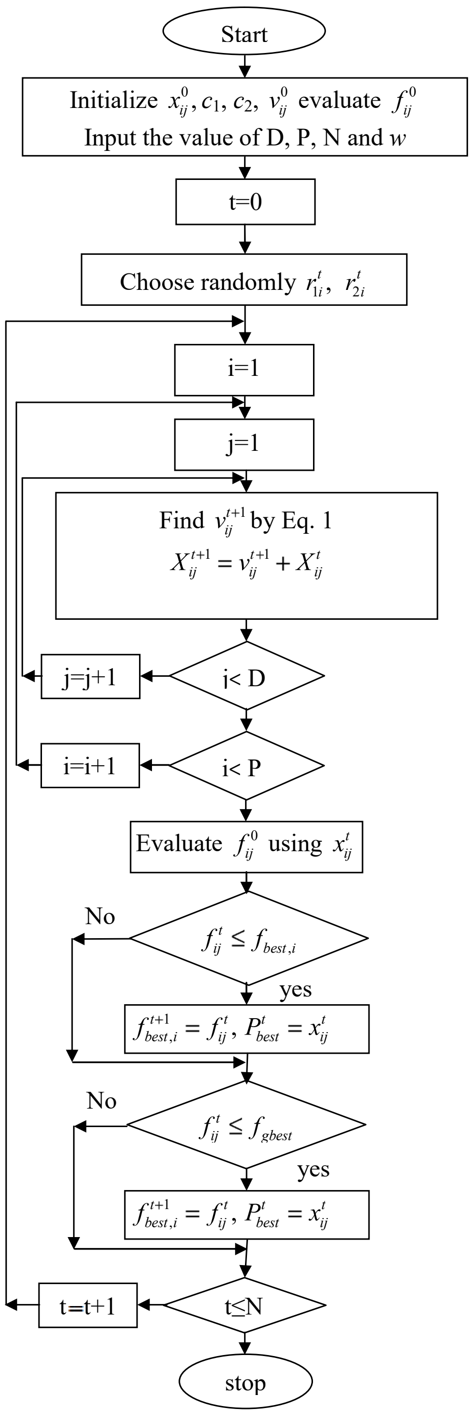

Fig. 3. Flowchart of the PSO algorithm procedure

The presented techniques are used to minimize the maximum overshoot, minimize the rise time. The objective function f_{ij} is the steady-state error (E_{ss}), and it is defined as: f_{ij} (K_p, K_d, K_i)= min (E_{ss}).

4. 4. Modified PSO (MPSO)

The modification of PSO includes two scenarios known as the M1-PSO and M2-PSO. The presented modified PSO is used to set the PID parameters for the DC motor. The error value is minimized to zero by MPSO-PID for controlling the output of the DC motor.

First scenario – M1-PSO. The modification of PSO is done by changing the value of the inertia weight formula (W), social and personal acceleration coefficient (C_1 and C_2). The value of weight (W) and (C_1 and C_2) of each generation are modified by equation (12)–(15). The goal of this parameter modification is to expand the range of particles in the beginning generation (global optimum) and to decrease the area when they are in the last generation (local optimum), therefore, PID parameter values reached are more mature. Four inertia weight functions have been considered as the following equations:

$$W = 0.4 + 0.5\frac{1-\dfrac{iter}{T}}{1+10\left(\dfrac{iter}{T}\right)}, \tag{12}$$

$$W_1 = 0.9 - 0.5\left(\frac{iter}{T}\right), \tag{13}$$

$$W_2 = \frac{1}{1+1.5e^{-2.6d}}, \tag{14}$$

$$W_3 = W_{max} - \frac{W_{max} - W_{min}}{T} * iter \tag{15}$$

$$C_1 = C_2 = C_o W_i \quad i \in \{1,2,3\}, \tag{16}$$

where $W_{\max}$ is the final weight, $W_{\min}$: is the initial weight, T is the maximum iteration number and *iter* is the current iteration number.

Second scenario – M2-PSO. In this scenario, the weight formula (W_i) values of each generation are modified by equation (12)–(16) while the values of C_1 and C_2 are constant. In classic PSO, the values of W, C_1 and C_2 are constant and (W=0.9, C_1=C_2=1.5). The classic PSO process keeps going exploring the best location but the convergence is delayed. A theoretical study [3, 19] showed that the inertia weight should be between [0.4, 0.9].

5. Research results

This section includes the results of the modified PSO scenarios using the benchmark of inertia weight functions for the speed and position control of the DC motor. The simulation is being carried out based on Matlab.

5. 1. Convergence based on PSO for position and speed control of DC motor

The corresponding performance curves of the PSO scenarios are depicted in Fig. 4–7, and show how the PSO algorithm converged to its final values. In more detail, all the cases of weight values (Wi) for the first and second scenario have been examined. As shown in Fig. 4–7, it can be found easily that the solution of the optimization problem by M1-PSO is caused that the fitness function is more optimized than other mentioned approaches such as M2-PSO and classic PSO. More precisely, the weight value (W3) gives the best and fastest solution in all presented scenarios and approaches for both speed and position control. On the other hand, in the case of position control based on W3, it needs 3 iterations to reach the best global value of fitness function in the case of M1-PSO, while it was 17 in the case of M2-PSO. The number of iterations in the case of M1-PSO is 9 iterations, while it was 15 in the case of M2-PSO for speed control.

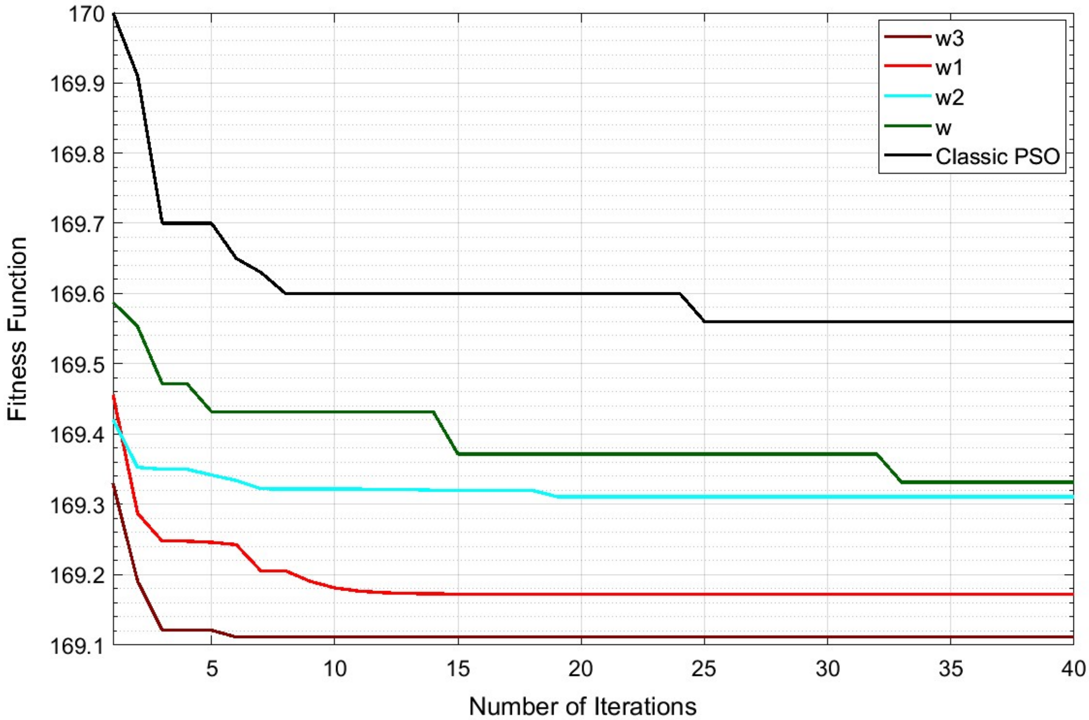

Fig. 4. Convergence curves for position control of DC motor based on M2-PSO

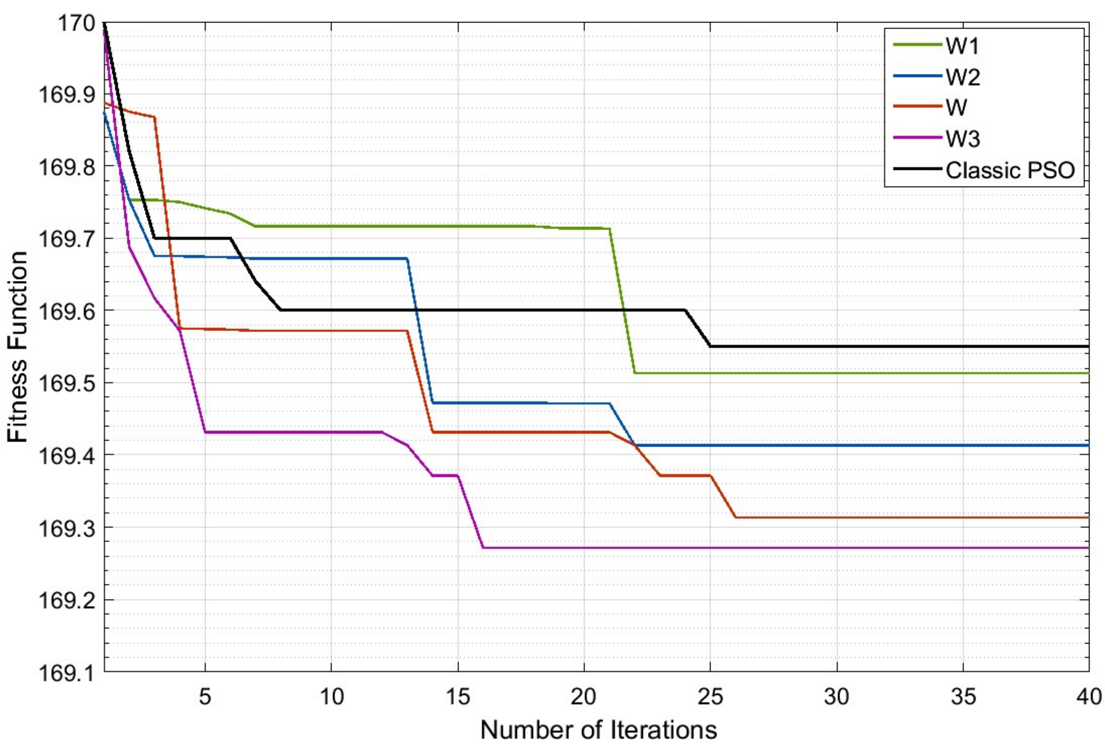

Fig. 5. Convergence curves for position control of DC motor based on M1-PSO

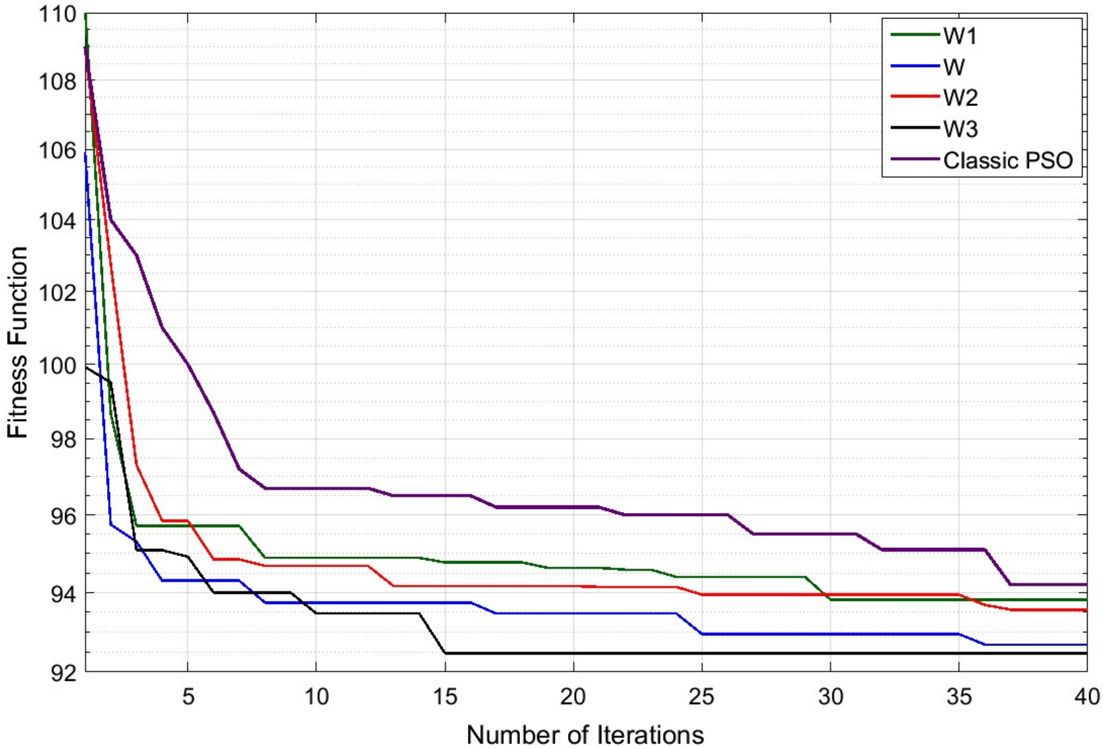

Fig. 6. Convergence curves for speed control of DC motor based on M2-PSO

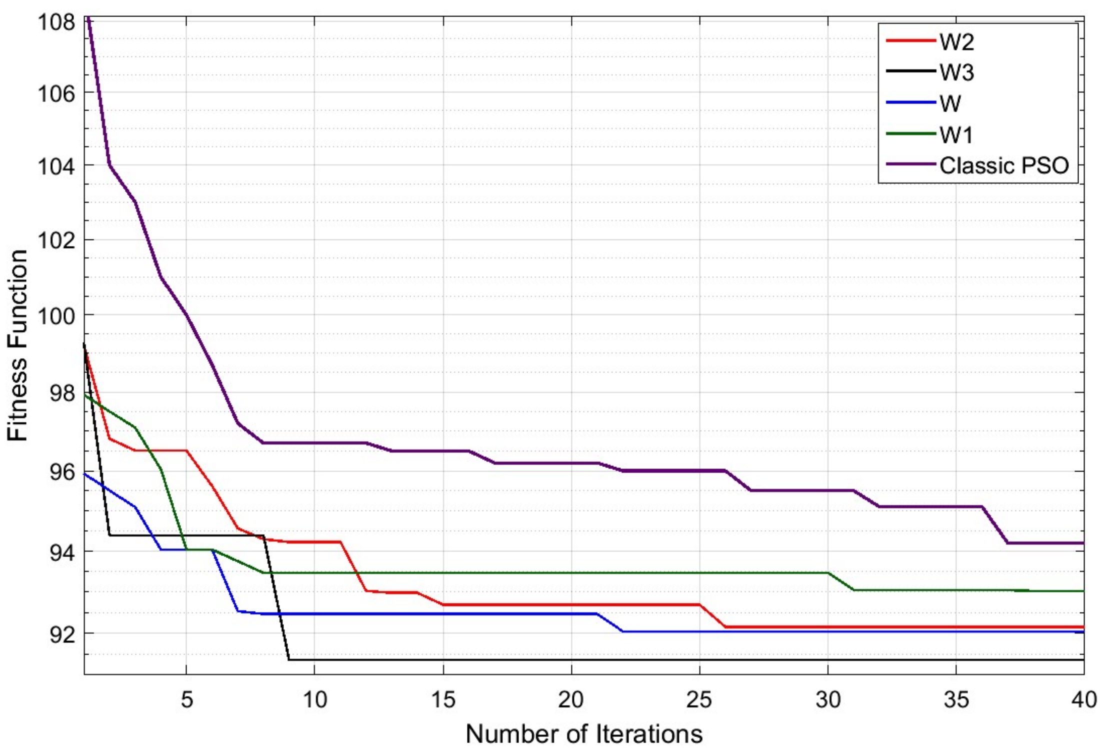

Fig. 7. Convergence curves for speed control of DC motor based on M1-PSO

5. 2. Step response for position and speed control of DC motor

Fig. 8, 9 described the step response curve of position control based on M1- PSO, M2-PSO, classic PSO and only PID controller (traditional values of PID) for the DC motor. Also, Fig. 8, 9 show that the step response based on W3 is better than W, W1, W2, and even the classic PSO method.

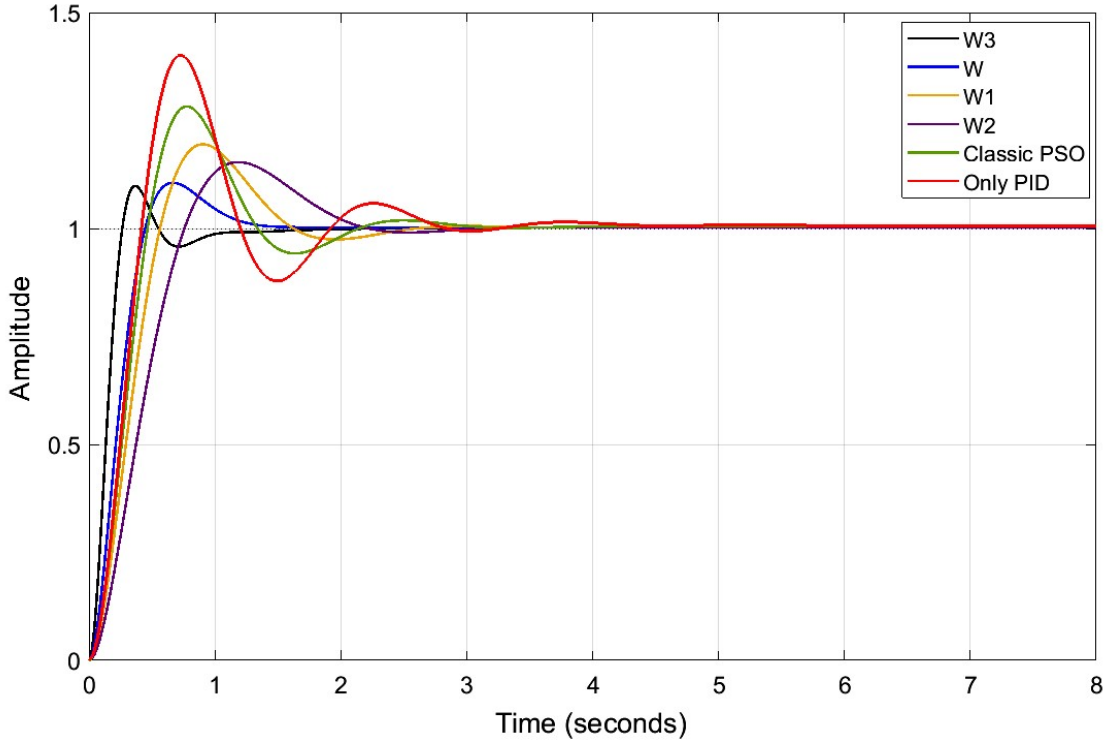

Fig. 8. Step response curve of position control based on M2-PSO

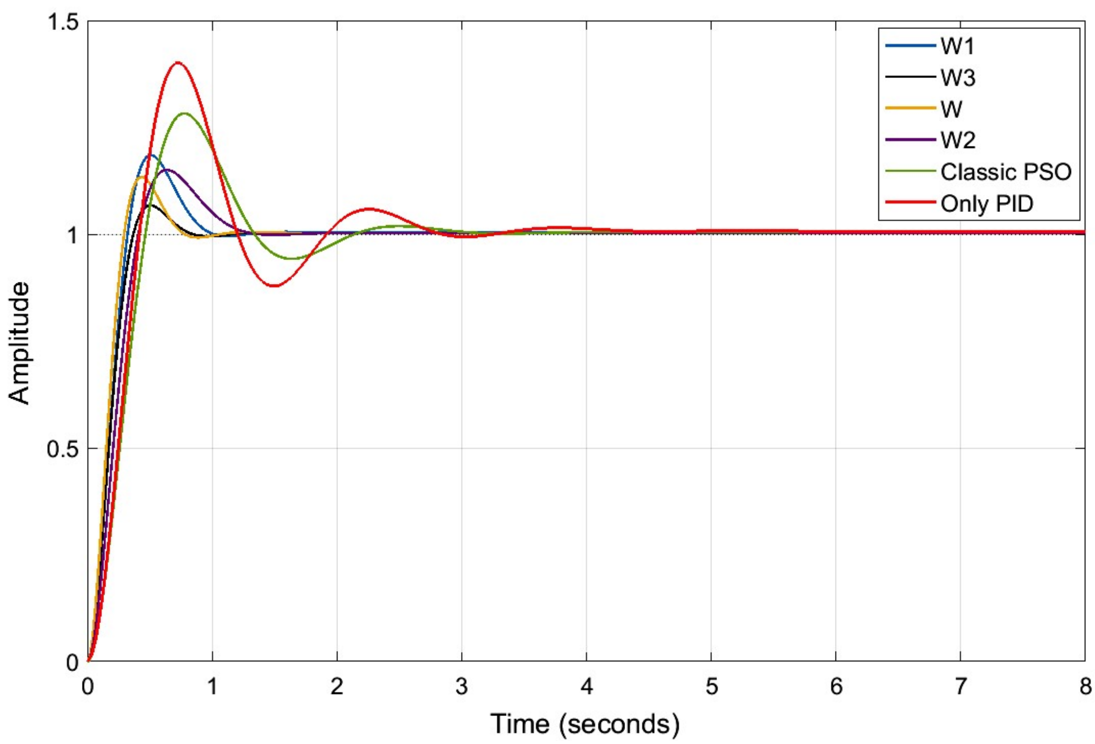

Fig. 9. Step response curve of position control based on M1-PSO

Furthermore, the step response curves of speed control based on PID controller tuning parameters using M1-PSO, M2-PSO, classic PSO and only PID controller are shown in Fig. 10, 11. Also, Fig. 10, 11 show that the step response based on W3 is better and faster than W, W1, W2, and even the classic PSO method.

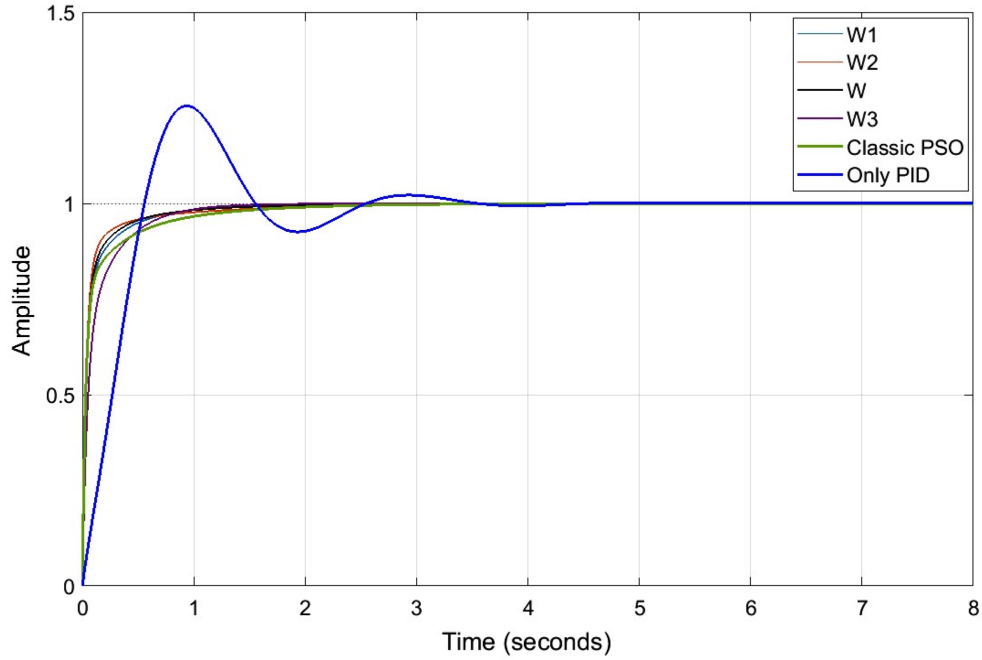

Fig. 10. Step response curve of speed control based on M2-PSO

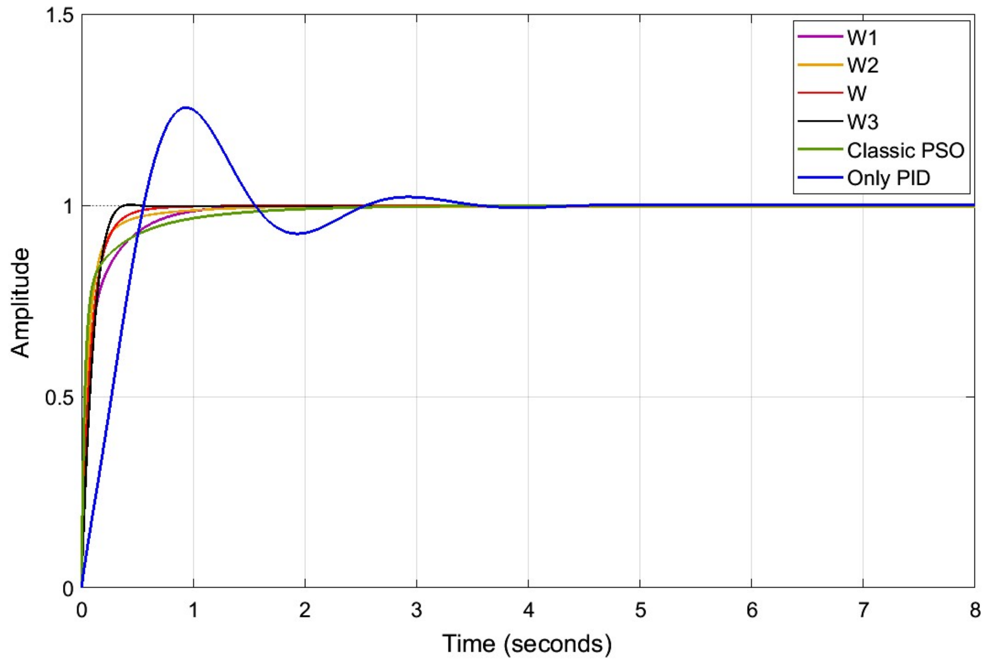

Fig. 11. Step response curve of speed control based on M1-PSO

5. 3. Performance of time response for speed and position control

The settling time (*Ts*), rise time (*Tr*), overshoot percentage, steady-state error (*Ess*), time of steady-state error (t_{Ess}) and the PID values for each case of suggested approaches are presented in Tables 1, 2 for speed and position control, respectively. For speed control, the step response corresponding to the M1-PSO scenario reached the steady-state value within 0.4 sec for speed control based on the W3 case. Indeed, it reached the steady-state value within 0.9 sec, 1.5 sec and 1.75 sec in the case of W, W2 and W1, respectively. While it was 4 sec and 4.65 sec in the used classic PSO and traditional values of the PID controller. Moreover, in the case of speed control, the overshoot for M1-PSO, M2-PSO and classic PSO is zero. While it was 25.3 % in the used traditional PID controller. For the position control, the first scenario based on W3 gives a better response compared to M2-PSO and classic PSO. The value of steady-state time is 1 sec but it was 1.2 sec, 1.45 sec and 1.8 sec by using W, W2 and W1, respectively. While it was 3.7 sec and 4.5 sec in the case of classic PSO and only used PID controller.

Table 2 (position M1) shows that there is a slightly different rise time between the PID controller and PID with M1-PSO but the overshoot obviously can be reduced from

40.2 % to 6.73 % by implementing M1-PSO. Table 2 shows that the rise time reduced from 0.444 sec for only PID controller to 0.193 sec by implementing M1-PSO but the overshoot was reduced from 25.3 % to zero.

Table 1

Summary of the time response for speed control performance

M2-PSO						
W_i	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
W3	1.13	0.325	0	0	1.9	52.166, 71.635, 7.835
W	0.927	0.255	0	0	2.5	79.865, 91.817, 12.191
W2	1.14	0.265	0	0	2.92	105, 92.654, 12.855
W1	0.91	0.21	0	0	3.45	88.265, 96.653, 12.262
M1-PSO						
W_i	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
W3	0.296	0.193	0	0	0.4	63.864, 103.25, 4.173
W	0.479	0.209	0	0	0.9	65.647,101.56, 6.358
W2	0.735	0.198	0	0	1.5	78.545, 101.56, 6.358
W1	0.911	0.385	0	0	1.75	52.166, 71.635, 7.835
Classic PSO						
W=0.9	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
–	0.354	1.39	0	0	4	71.7, 69.35, 13.173
Only PID						
W=0	3.06	0.444	25.3	0.15	4.65	10.264, 67.352,1.109

Table 2

Summary of the time response for position control performance

M2-PSO						
W_i	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
W3	0.917	0.177	9.86	0	2.2	69.83, 1.549, 48.89
W	1.17	0.301	10.6	0	2.5	59.21, 0.832, 22.05
W2	2.01	0.524	15.4	0	3	35.25, 0.973, 8.147
W1	2.16	0.389	19.5	0	3.5	51.69, 0.444, 11.69
M1-PSO						
W_i	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
W3	0.702	0.22	6.73	0	1.0	59.61, 3.103, 32.17
W	0.899	0.22	13.4	0	1,2	76.52, 2.713, 30.16
W2	0.722	0.225	13.46	0	1,45	83.92, 1.45, 38.35
W1	1.14	0.279	15	0	1.8	68.91, 1.356, 22.36
Classic PSO						
W=0.9	T_s (s)	T_r (s)	Over-shoot %	*Ess* (%)	t_{Ess} (s)	P,I,D
–	2.06	0.319	28.2	0	3.7	70.38, 2.459, 13.13
Only PID						
W=0	2.64	0.284	40.2	0.2	4.5	88.29, 11.96, 12.74

6. Discussion of simulation results

The application of modified PSO for speed and position control of the DC motor has been presented. The ITAE has been considered as an objective function for the proposed optimization algorithm. Comparison of the proposed PSO schemes with classical PSO as well as traditional PID has also been studied. The sturdiness analysis of the first scenario of the suggested modified PSO has also been carried out and shown that the proposed MPSO approach has fast searching speed and better performance compared to classical PSO. Moreover, the accuracy and speed of convergence of the proposed method based on M1-PSO are better than those of M2-PSO and even classical PSO.

In other words, the PID controller architecture using M1-PSO causes the overshoot rate in the step response curve to be reduced compared to M2-PSO and classic PSO. The simulation results have shown that the first scenario based on inertia weight function (W3) gives a better response compared to the second scenario and classic PSO and it reaches the global minimum value in fewer iteration numbers compared to the previous study [11, 12]. Indeed, the obtained simulation results encourage to apply the proposed approach in practical control systems. Furthermore, the proposed method accelerates the convergence for the PSO algorithm.

Moreover, the limitations of the PSO technique stem from that it easily suffers from partial optimism, which allows its velocity and direction to be controlled less accurately. Also, there is no general principle of convergence specific to realistic theories of multidimensional difficulties. Input parameters tuning and experimenting with different implementations of the PSO algorithm are sometimes required.

7. Conclusions

1. The main messages brought by this paper are that using the modified PSO algorithm based on speed and position control of the DC motor can be meaningful and the optimization method has indeed been able to tune optimal PID parameters successfully and accurately, which confirmed the rightness of proposed scheme and good behavior of DC motor control, which were not learnt by the perceptron.

2. Based on an analysis aimed at the implementation of the PID controller for the DC motor, I selected a modified PSO algorithm, which makes it possible to improve the response of the controller. Indeed the study shows that it reached the steady state value within 0.4 and 1 sec for speed and position control, respectively.

3. Furthermore, comparative studies have shown that changes of the inertia weight function W3, social and personal acceleration coefficient lead to improved PSO method efficiency, and compared to the second scenario (M2 PSO) and classical PSO, it has very efficient rapid implementation and better response for control systems.

4. A structure of the control system of the DC motor, based on the modified PSO, which derives optimum values for the PID coefficients controller in the process of operation was proposed. Such an approach makes the time response results presented in this work to have accurate results and a better response effect compared to other methods. More precisely using W3 led to the enhancement of DC motor performance and gave the overshoot of 0 % and 6.73 % for speed and position

control, respectively, and the steady-state error became zero for both cases of control. While the settling time (*Ts*) and rise time (*Tr*) became 0.296 and 0.193 sec for speed control and it became 0.702 and 0.22 for position control. For future work, I plan to experimentally realize the proposed control scheme.

Acknowledgments

The author is very grateful to the University of Mosul/College of Engineering for their provided facilities, which helped to improve the quality of this work.

References

1. De la Guerra, A., Alvarez–Icaza, L., Torres, L. (2018). Brushless DC motor control with unknown and variable torque load. IFAC-PapersOnLine, 51 (13), 644–649. doi: https://doi.org/10.1016/j.ifacol.2018.07.353
2. Åström, K. J., Hägglund, T. (2001). The future of PID control. Control Engineering Practice, 9 (11), 1163–1175. doi: https://doi.org/10.1016/s0967-0661(01)00062-4
3. Leena, N., Shanmugasundaram, R. (2014). Artificial neural network controller for improved performance of brushless DC motor. 2014 International Conference on Power Signals Control and Computations (EPSCICON). doi: https://doi.org/10.1109/epscicon.2014.6887513
4. Azman, M. A. H., Aris, J. M., Hussain, Z., Samat, A. A. A., Nazelan, A. M. (2017). A comparative study of fuzzy logic controller and artificial neural network in speed control of separately excited DC motor. 2017 7th IEEE International Conference on Control System, Computing and Engineering (ICCSCE). doi: https://doi.org/10.1109/iccsce.2017.8284430
5. Bennett, S. (2001). The past of PID controllers. Annual Reviews in Control, 25, 43–53. doi: ttps://doi.org/10.1016/s1367-5788(01)00005-0
6. Bennett, S. (1993). Development of the PID controller. IEEE Control Systems, 13(6), 58–62. doi: https://doi.org/10.1109/37.248006
7. Fišer, J., Zítek, P. (2019). PID Controller Tuning via Dominant Pole Placement in Comparison with Ziegler-Nichols Tuning. IFAC-PapersOnLine, 52 (18), 43–48. doi: https://doi.org/10.1016/j.ifacol.2019.12.204
8. Åström, K. J., Hägglund, T. (2004). Revisiting the Ziegler–Nichols step response method for PID control. Journal of Process Control, 14 (6), 635–650. doi: https://doi.org/10.1016/j.jprocont.2004.01.002
9. Åström, K., Hägglund, T. (1995). PID controllers: theory, design, and tuning. Research Triangle Park.
10. Mishra, A. K., Tiwari, V. K., Kumar, R., Verma, T. (2013). Speed control of dc motor using artificial bee colony optimization technique. 2013 International Conference on Control, Automation, Robotics and Embedded Systems (CARE). doi: https://doi.org/10.1109/care.2013.6733772
11. Rodríguez-Molina, A., Villarreal-Cervantes, M. G., Aldape-Pérez, M. (2017). An adaptive control study for a DC motor using meta-heuristic algorithms. IFAC-PapersOnLine, 50 (1), 13114–13120. doi: https://doi.org/10.1016/j.ifacol.2017.08.2164
12. Syafaah, L., Widianto, Pakaya, I., Suhardi, D., Irfan, M. (2017). PID designs using DE and PSO algorithms for damping oscillations in a DC motor speed. 2017 4th International Conference on Electrical Engineering, Computer Science and Informatics (EECSI). doi: https://doi.org/10.1109/eecsi.2017.8239138
13. Potnuru, D., Alice Mary, K., Sai Babu, C. (2019). Experimental implementation of Flower Pollination Algorithm for speed controller of a BLDC motor. Ain Shams Engineering Journal, 10 (2), 287–295. doi: https://doi.org/10.1016/j.asej.2018.07.005
14. Taki El-Deen, A., Mahmoud, A. A. H., R. El-Sawi, A. (2015). Optimal PID Tuning for DC Motor Speed Controller Based on Genetic Algorithm. International Review of Automatic Control (IREACO), 8 (1), 80. doi: https://doi.org/10.15866/ireaco.v8i1.4839
15. Mamchur, D., Yatsiuk, R. (2018). Development of the PID-neurocontroller to compensate for the impact of damages and degradation of induction motor on operation of the electric drive system. Eastern-European Journal of Enterprise Technologies, 5 (2 (95)), 66–77. doi: https://doi.org/10.15587/1729-4061.2018.136466
16. Muniraj, M., Arulmozhiyal, R., Kesavan, D. (2020). An Improved Self-tuning Control Mechanism for BLDC Motor Using Grey Wolf Optimization Algorithm. International Conference on Communication, Computing and Electronics Systems, 315–323. doi: https://doi.org/10.1007/978-981-15-2612-1_30
17. Achanta, R. K., Pamula, V. K. (2017). DC motor speed control using PID controller tuned by jaya optimization algorithm. 2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI). doi: https://doi.org/10.1109/icpcsi.2017.8391856
18. Qi, Z., Shi, Q., Zhang, H. (2020). Tuning of Digital PID Controllers Using Particle Swarm Optimization Algorithm for a CAN-Based DC Motor Subject to Stochastic Delays. IEEE Transactions on Industrial Electronics, 67 (7), 5637–5646. doi: https://doi.org/10.1109/tie.2019.2934030
19. Xie, W., Wang, J.-S., Wang, H.-B. (2019). PI Controller of Speed Regulation of Brushless DC Motor Based on Particle Swarm Optimization Algorithm with Improved Inertia Weights. Mathematical Problems in Engineering, 2019, 1–12. doi: https://doi.org/10.1155/2019/2671792
20. Agarwal, J., Parmar, G., Gupta, R. (2018). Comparative Analysis of PID Controller for Speed Control of DC motor with Intelligent Optimization Algorithms. 2018 International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), 273–277. doi: https://doi.org/10.1109/icacccn.2018.8748475
21. Gupta, S., Deep, K. (2019). A novel hybrid sine cosine algorithm for global optimization and its application to train multilayer perceptrons. Applied Intelligence, 50 (4), 993–1026. doi: https://doi.org/10.1007/s10489-019-01570-w

22. Hashim, N. L. S., Yahya, A., Andromeda, T., Kadir, M. R. R. A., Mahmud, N., Samion, S. (2012). Simulation of PSO-PI Controller of DC Motor in Micro--EDM System for Biomedical Application. Procedia Engineering, 41, 805–811. doi: https://doi.org/10.1016/j.proeng.2012.07.247
23. Kennedy, J., Eberhart, R. (1995). Particle Swarm Optimization. Proceedings of ICNN'95 - International Conference on Neural Networks, 1942–1948. doi: https://doi.org/10.1109/icnn.1995.488968
24. Shi, Y., Eberhart, R. (1998). A modified particle swarm optimizer. 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360). doi: https://doi.org/10.1109/icec.1998.699146
25. Zhan, Z.-H., Zhang, J., Li, Y., Chung, H. S.-H. (2009). Adaptive Particle Swarm Optimization. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 39 (6), 1362–1381. doi: https://doi.org/10.1109/tsmcb.2009.2015956

UDC 656.612.8
DOI: 10.15587/1729-4061.2021.225517

THE SYNTHESIS OF CONTROL SYSTEM TO SYNCHRONIZE SHIP GENERATOR ASSEMBLIES

V. Budashko
Doctor of Technical Sciences, Associate Professor
Educational and Scientific Institute of Automation and Electromechanics
E-mail: bvv@te.net.ua

V. Shevchenko
PhD, Associate Professor
Training & Certifying Center of Seafarers*
E-mail: vash4891@gmail.com

*National University «Odessa Maritime Academy»
Didrikhsona str., 8, Odessa, Ukraine, 65029

This paper considers the construction of principles and the synthesis of a system of effective control over the processes of synchronization of generator sets (GSs) that form a part of the distributed MP-control systems for complex ship technical systems and complexes (STS and C). The tasks of synchronization have been set, the process and database models have been built, the system configurations have been defined. Based on the use of resultant functions, we have determined stages in solving the tasks of control over the frequency adjustment synchronization in a hierarchical sequence. The performance analysis of the STS and C control elements has been carried out; the use of the integrated optimization criteria and dual management principles has been proposed. Practical techniques to manage the GS synchronization have been given. We have solved the problem of high-speed control over the frequency of synchronized objects based on the principles of adjustment. That has made it possible to determine in advance the moments of GS enabling under the deterministic and stochastic statement of the synchronization task. The results of the experimental study into the GS synchronization processes are given; the effectiveness of the proposed GS control has been proven. The principles underlying the construction of procedures to control the GS composition when using the methods of "rigid" and "flexible" thresholds have made it possible to define the optimization criteria and implement a control law that satisfied the condition for an extremum, which is an indicator of the feasibility of the set goal and takes into consideration the limitations of control influences. We managed to design a system in the class of adaptive control systems by the appropriate decomposition of the system's elements by splitting a synchronization task into the task on performance and the task on control under the required conditions. The given examples of the processes where the synchronization failed while using standard synchronizer control algorithms, as well as processes of successful GS synchronization when applying the proposed synchronizer dual control algorithms, have confirmed the reliability of the main scientific results reported here

Keywords: technical operation, synchronization, quality, control system, mathematical modeling

Received date 04.01.2021
Accepted date 05.02.2021
Published date 26.02.2021

1. Introduction

The use of microprocessor (MP) equipment in the ship's electricity generating units implies the automation of control processes, as well as the processes of control and protection of ship technical systems and complexes (STS and C). An important and integrated part of each of these processes is the measurement of electric power parameters, which, for example, are necessary to enable effective automatic synchronization of generator sets (GSs). Namely: the magnitude and nature of load currents in all phases, the voltage, frequency of the current, the phase angle φ of the shift between the current and voltage, power direction, and others.

It is known that aboard a modern sea vessel most of the measurement systems for electric power parameters are associated with the functioning of analog-digital converters (ADC), and, along with the issue of ensuring reliability and accuracy, there is a task to provide for the necessary speed of measurements. Resolving this issue requires finding new ways – based on a systematic approach with accounting for known digital measurement techniques. In a general form, the task to measure the electrical energy parameters in an n-phase system relates to determining the set of values $\{I_1, ..., I_n, U_1, ..., U_n, f, \varphi\}$ according to the selected performance criterion. In this case, $I_1, ..., I_n$ and $U_1, ..., U_n$ are the amplitude values of phase currents and linear voltages.