IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received March 27, 2022, accepted April 9, 2022, date of publication April 13, 2022, date of current version April 21, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3167026

Proportional-Integral-Derivative Parametric Autotuning by Novel Stable Particle Swarm Optimization (NSPSO)

WUDHICHAI ASSAWINCHAICHOTE[1], CHRISSANTHI ANGELI[2], AND JIRAPUN PONGFAI[1]

[1]Department of Electronic and Telecommunication Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand

[2]Department of Electrical and Electronics Engineering, Faculty of Engineering, University of West Attica, 122 43 Athens, Greece

Corresponding author: Wudhichai Assawinchaichote (wudhichai.asa@kmutt.ac.th)

ABSTRACT To improve the performance, robustness and stability of autotuning the proportional integral and derivative (PID) parameter, the novel stable particle swarm optimization (NSPSO) is proposed in this paper. The NSPSO is the combination of the particle swarm and optimization algorithm with the new stable rule to reconsider the survival of the remaining particle in the search space for handling the instability of the system. The new rule is proposed based on proving the stability according to the Lypunov stability theorem. Additionally, to show the method's superiority in performance and robustness, the proposed method is compared with the results of simulations with the particle swarm optimization (PSO), the hybrid particle swarm optimization-grey wolf optimization (PSO-GWO), the whale optimization algorithm (WOA) and the social spider optimization algorithm (SSO) based on a direct current (DC) motor control system. In the comparative performance, the various fitness functions are applied, while the comparative robustness and the changed operation point of the DC motor are applied. After comparing the methods, the proposed method obtains better results than the PSO, PSO-GWO, WOA and SSO in both performance and robustness.

INDEX TERMS DC motor, Lypunov stability, optimization, particle swarm and optimization (PSO), Proportional integral and derivative (PID).

I. INTRODUCTION

The proportional integral and derivative (PID) controller is commonly applied in many fields, such as in wind energy [1], robotics [2]–[5], optical networks [6], hydraulics and pneumatics [7]–[9], industrial processes [10], vehicles [11], and power systems [12] because its structure is simple as a result of its easy implementation, maintenance, and low cost [6], [13]. Nevertheless, its performance relies on the balancing of 3 parameters: proportional gain (K_P), integral gain (K_I) and derivation gain (K_D) [14]. There are parameter effects on the transient curve. For instance, K_P affects the stability of the transient response, K_I affects the steady-state error (E_S) and the maximum overshoot (OS) and K_D affects the improvement for the future response. The conventional method, called Ziegler-Nichols (ZN), is manually tuned by fixing the operating point. In practice, the system is operated by the different operations, and thus, this method is an unsuitable system with different operations [15]–[18].However, many methods, such as Cohen-Coon and phase and gain margin methods have been proposed to overcome the limitations of the conventional methods, but they require experienced designers and more time for tuning [19].

Earlier, the artificial intelligence (AI) evaluation and swarm algorithm was commonly applied to optimally tune PID parameters by continuously minimizing the performance index until it approached the best system response [19], such as the ant lion optimizer (ALO) algorithm [20], the gas brownian motion optimization (GBMO) [21], the bacterial foraging optimization (BFO) algorithm [22], the improved sine cosine algorithm [23], the cuckoo search optimization (CSO) [7], the bat algorithm [24], the ant colony optimized (ACO) [25], the genetic algorithm (GA) [26], [27], the world cup optimization (WCO) [28], the gray wolf optimization algorithm [12], [29], the kidney-inspired algorithm [30], the green leaf-hopper flame optimization algorithm (GLFOA) [31], the improved QUATRE algorithm [32], the fractional-order fish migration optimization

The associate editor coordinating the review of this manuscript and approving it for publication was Ton Duc Do.

algorithm (FOFMO) [33] and the particle swarm optimization (PSO) algorithm [34].

PSO is widely applied to autotune the PID parameter because of its effectiveness and efficiency in handling nonlinear and easy implementation [35], [36]. However, it still has the limitation of falling to the local minima and converging [37], [38]. Many studies have improved the conventional PSO by combining it with the advantages of other algorithms, such as the particle swarm optimization algorithm and the linear-quadratic-regulator (PSO-LQR) [39], the hybrid particle swarm optimization-Grey wolf optimization (PSO-GWO) [38], the modified particle swarm optimization based on dynamic weight and crossover operator (MPSO-IPID) [40], and the improved PSO [36]. Nevertheless, its performance still depends on the setting of the initial particle, and thus, the risk of falling into the local minima still exists [37].

Therefore, this paper proposes the novel stable PSO (NSPSO) to improve the performance, convergence, robustness and stability of autotuning the PID parameter. The method is the improved PSO because it adds the process of determining the survival of the remaining particle in the search space based on sufficient conditions and re-generating the new particle corresponding to the survival particle. To verify the performance, convergence and robustness, the proposed method is compared with the results of simulations with the PSO [44], the PSO-GWO [38], the whale optimization algorithm (WOA) [45] and the social spider optimization algorithm (SSO) based on the direct current (DC) motor system. For comparative performance and convergence, a different fitness function is applied by fixing the operation point of the DC motor. In the comparative robustness part, the changed operation of the DC motor is applied, but the fitness function is fixed. Additionally, the sense of Lypunov stability is used to prove the closed-loop stability of the sufficient conditions. Hence, the contributions of this paper are as follows:

1. NSPSO is proposed to improve the performance, convergence, robustness and stability of the autotuning PID parameter. The proposed method is the improved PSO by reconsidering the remaining particle in the search space.
2. The theoretical sufficient condition to determine the survival particle is proven according to the sense of the Lypunov stability.
3. To verify the superiority of the performance and convergence, the comparative simulation between the proposed method, the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO based on the DC motor are applied with varying fitness functions, but the operation of the DC motor is fixed.
4. To verify the superiority of the robustness, the comparative simulation between the proposed method, the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO based on the DC motor are applied with the operation of the DC motor, but the fitness function is fixed.

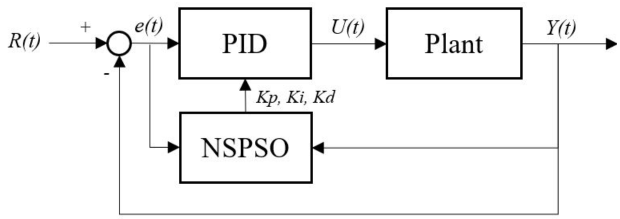

FIGURE 1. Block diagram of the PID controller.

The organization of this paper is as follows: the PID controller and fitness function are described in section II, novel stable particle swarm and optimization is discussed in section III, a simulation and result analysis is presented in section VI and the conclusion and discussion are shared in section V.

II. THE PID CONTROLLER AND THE FITNESS FUNCTION

The proportional integral and derivative controller (PID) are designed based on the derivative of the difference between the reference input ($R(t)$) and the output of the system ($Y(t)$), which this paper applies to brushless DC motor; i.e., $R(t)$ is the reference velocity and $Y(t)$ is the output velocity.

$$U(t) = K_P e(t) + K_I \int_0^t e(t)dt + K_D \frac{de(t)}{dt} \tag{1}$$

where $e(t)$ is an error which is calculated from the derivative of $R(t)$ and $Y(t)$. The parameter of the PID controller is denoted as the proportional gain (K_P), the integral gain (K_I) and the derivative gain (K_D). It is the fact that the controller gains are required to properly design according to the condition of the system since the behavior of the system depends on these controller parameters [15]. In this paper, the novel stable particle swarm optimization (NSPSO) algorithm is proposed to design the PID controller parameter by autotuning. During the process of autotuning, the fitness function is used to determine the quality. Normally, the fitness function is utilized by applying a performance index [19], such as the integral of the absolute error (IAE), the integral of the time multiplied squared error (ITSE), the integral of the time multiplied absolute error (ITAE), and the integral of the squared error (ISE). They are defined as follows [19]:

$$IAE = \int_0^t |e(t)|dt \tag{2}$$

$$ITSE = \int_0^t te(t)^2 dt \tag{3}$$

$$ITAE = \int_0^t t|e(t)|dt \tag{4}$$

$$MSE = \frac{1}{t}\int_0^t (e(t))^2 dt \tag{5}$$

$$ISE = \int_0^t e(t)^2 dt \tag{6}$$

where $e(t)$ is the error in a time domain and t is the time. Additionally, [41] proposed the new fitness function as follows:

$$J(t) = \omega_{ie}\omega_{iae}\int_0^t |e(t)|dt + \omega_{ise}\int_0^t e(t)^2 dt + (\omega_{td}(1+OS(t))(\omega_r t_{rt}(t) + \omega_s t_{st}(t))) \quad (7)$$

where ω_{ie}, ω_{iae}, ω_{ise}, ω_{td}, ω_r and ω_s are the weights of the performance index, such as the integral error (IE), IAE, ISE, overshoot, rise time ($t_{rt}(t)$) and settling time ($t_{st}(t)$). $OS(t)$ is the maximum overshoot.

III. NOVEL STABLE PARTICLE SWARM AND OPTIMIZATION

A novel stable particle swarm optimization (NSPSO) algorithm is proposed to improve the limitation of the particle swarm optimization (PSO) in the local minima by adding the process of regenerating the particle with a bad fitness function based on the remaining particle with a good fitness according to the Levy function. The bad and good fitness functions for each particle is classified by using the acceptance fitness, which is set by the designer to achieve the design requirements. If the fitness function of the particle is more than the acceptance fitness, the particle is interpreted as the good fitness function; otherwise, the particle is interpreted as the bad fitness function.

The PSO was proposed according to the behavior of bird swarming by Eberhart and Kennedy. In each iteration of the PSO, the best solution is found based on the vector of the velocity and the position of each particle, which evaluates the quality according to the fitness function by moving the particle in the searched space [39]. In the searched space with dimension D, the number of particles N in iteration i_{th} with the term local best position or ($pbest$) is stored in the memory with format $p_i = (p_{i1}, p_{i2}, \ldots, p_{ND})$. The best position of the whole particle ($gbest$) is stored in the memory with format $p_g = (p_{g1}, p_g, \ldots, p_{gD})$ [42]. The velocity is updated as follows:

$$v(t+1)_i^d = v(t)_i^d + c_1 \times r_1 \times (p(t)_i^d - x(t)_i^d) + c_2 \times r_2 \times (p(t)_g^d - x(t)_i^d) \quad (8)$$

The position is updated as follows:

$$x(t+1)_i^d = x(t)_i^d + v(t+1)_i^d \quad (9)$$

where r_1 and r_2 are random numbers with the range [0,1], c_1 is the cognitive learning factor and c_2 is the social learning factor. For implementing the PSO, the flow chart is shown in Figure 2.

In Figure 3, the NSPSO is proposed by combining the PSO and the process of determining the survival of the remaining particle in the search space based on the fitness function to prevent the local minima in the PSO. The process of determining the survival of the remaining particle in the search space determines the weak particles and deletes them from the search space. Then, the new position particles are generated according to (9) and $v(t+1)$ is replaced with the Levy optimization function as follows [46]:

$$v(t+1) = [\frac{(1+\beta-1)! \times (sin\frac{1.5\pi}{2})}{(\frac{1+\beta}{2}-1)! \times \beta \times 2(\frac{\beta-1}{2})}]^{\frac{1}{\beta}} \quad (10)$$

where β is the constant that this paper set to be 1.5. Additionally, the process of checking the stability of the new particle is added by determination according to (12) as Theorem 1.

Theorem 1: Each time t generates a new particle in the search space with dimension d as follows:

$$k(t) = 2(c_1 p(t)_g^d + c_2 p(t)_g^d) \quad (11)$$

each new particle is determined based on the equation as follows:

$$C(t) \leq 7k(t) + 4c_1c_2R(E(t) + G(t)) \quad (12)$$

where $E(t) = \frac{1}{3}(e^3(t)e^3(t-1)) + \xi(t)(e(t) + e(t-1) + \frac{e(t-1)+e(t)}{\xi(t-1)})$, $\xi(t) = e(t)e(t-1)$, $G(t) = (1 + OS(t)(t_{rt}(t) + t_{st}(t))) + (1 + OS(t-1)(t_{rt}(t-1) + t_{st}(t-1)))$, *$e(t)$ is the error at time t, $OS(t)$ is the maximum overshoot, $t_{st}(t)$ is the settling time and $t_{rt}(t)$ is the rise time.*

Proof: To verify the stability of generating a new particle, the Lyapunov function is defined as follows based on (7):

$$V(t) = \omega_{ie}\omega_{iae}\int_0^t |e(t)|dt + \omega_{ise}\int_0^t e(t)^2 dt + (\omega_{td}(1+OS(t))(\omega_r t_{rt} + \omega_s t_{st})) \quad (13)$$

The difference in time between $(t-1)$ and (t) of the Lyapunov function is written as follows:

$$\Delta V(t) = V(t) - V(t-1) \quad (14)$$

Therefore, (13) is as follows:

$$\begin{aligned}\Delta V(t) = {} & \omega_{ie}\omega_{iae}\int_0^t |e(t)|dt + \omega_{ise}\int_0^t e(t)^2 dt \\ & + (\omega_{td}(1+OS(t))(\omega_r t_{rt}(t) + \omega_s t_{st}(t))) \\ & - \omega_{ie}\omega_{iae}\int_0^t |e(t-1)|dt - \omega_{ise}\int_0^t e(t-1)^2 dt \\ & - (\omega_{td}(1+OS(t))(\omega_r t_{rt}(t-1) + \omega_s t_{st}(t-1)))\end{aligned} \quad (15)$$

$$\Delta V(t) = \Delta|e(t)| + \frac{\Delta e^3(t)}{3} + \Delta(1 + OS(t-1)) \times (t_{rt}(t-1) + t_{st}(t-1))) \quad (16)$$

$$\Delta V(t) = \Delta|e(t)| + \frac{\Delta e^3(t)}{3} + \Delta t_{rt}(t) + \Delta t_{st}(t) + \Delta OS(t)t_{rt}(t) + \Delta OS(t)t_{st}(t) \quad (17)$$

We consider $\Delta e(t)$, $\Delta t_{rt}(t)$, $\Delta t_{st}(t)$ and $OS(t)$ according to (8) given that $r_1, r_2 = 1$. The equation is as follows:

$$\Delta e(t) = \Delta e(t) + \Delta[2c_1p(t)_i^d - 2c_1e(t) + c_2x(t)_i^d - 2c_2e(t)] \quad (18)$$

Given that $k(t) = 2(c_1x(t)_i^d + c_2x(t)_i^d)$ as follows:

$$R = \frac{1}{4c_1c_2} - \frac{1}{2c_2} - \frac{1}{2c_1} \quad (19)$$

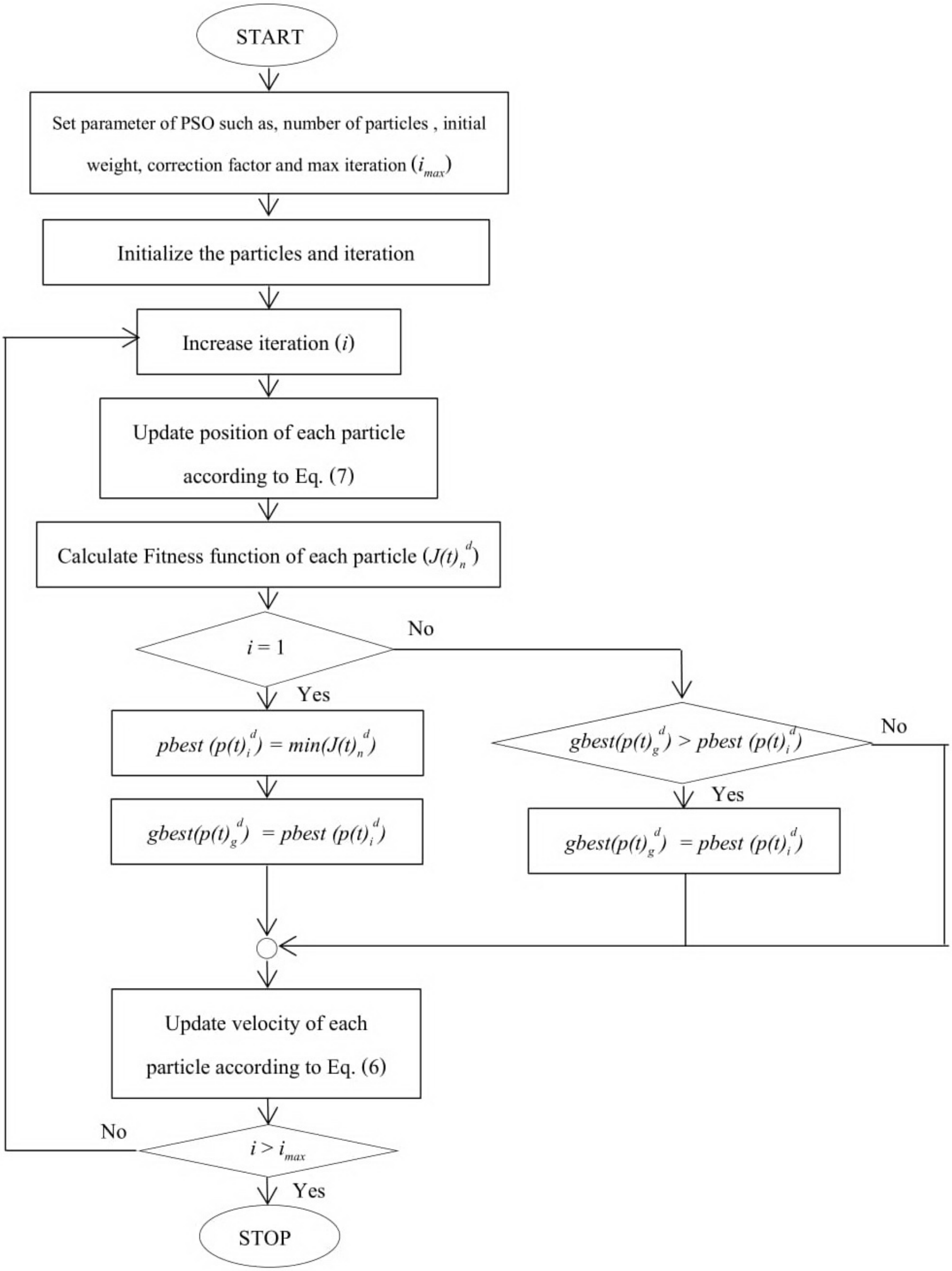

FIGURE 2. The flow chart of PSO.

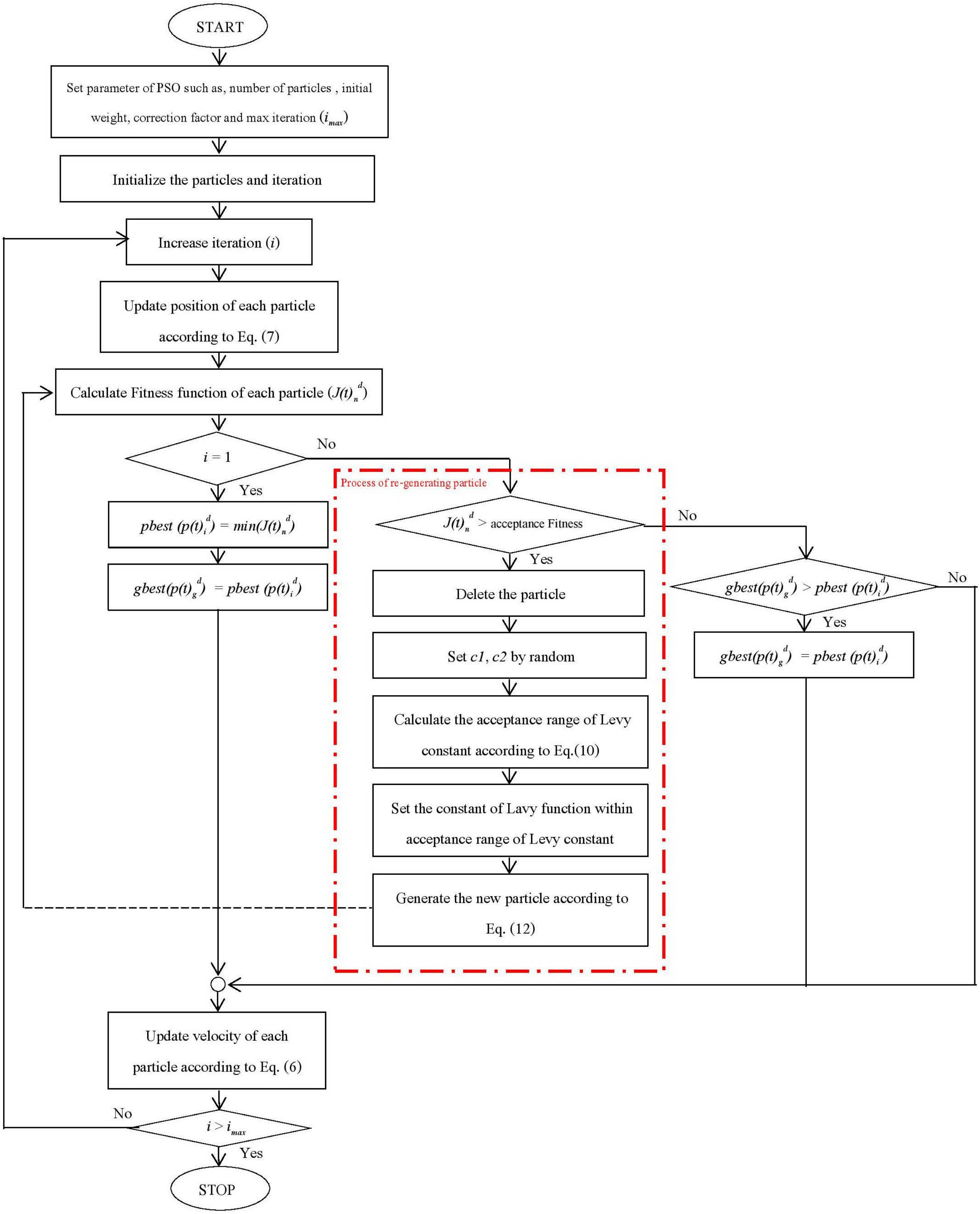

FIGURE 3. The flow chart of the NSPSO.

Hence the equation is as follows:

$$\Delta e(t) = 4c_1c_2R(e(t) + e(t-1)) + k(t) \tag{20}$$

We consider $\Delta t_{rt}(t)$ based on (8) as follows:

$$\Delta t_{rt}(t) = \Delta t_{rt}(t) + \Delta[2c_1p(t)_i^d - 2c_1t_{rt}(t) + c_2x(t)_i^d - 2c_2t_{rt}(t)] \tag{21}$$

(19) is written as follows:

$$\Delta t_{rt}(t) = 4c_1c_2R(t_{rt}(t) + t_{rt}(t-1)) + k(t) \tag{22}$$

We consider $\Delta t_{st}(t)$ based on (8) as follows:

$$\Delta t_{st}(t) = \Delta t_{st}(t) + \Delta[2c_1p(t)_i^d - 2c_1t_{st}(t) + c_2x(t)_i^d - 2c_2t_{st}(t)] \tag{23}$$

From (19), the equation is as follows:

$$\Delta t_{st}(t) = 4c_1c_2R(t_{rt}(t) + t_{st}(t-1)) + k(t) \tag{24}$$

We consider $\Delta OS(t)$ based on (8) as follows:

$$\Delta OS(t) = \Delta OS(t) + \Delta[2c_1p(t)_i^d - 2c_1OS(t) + c_2x(t)_i^d - 2c_2OS(t)] \tag{25}$$

From (19), the equation is as follows:

$$\Delta OS(t) = 4c_1c_2R(OS(t) + OS(t-1)) + k(t) \tag{26}$$

We replace (20), (22), (24) and (26) to (17) as follows:

$$\Delta V(t) = 4c_1c_2R(\frac{e^3(t)}{3} + e^2(t)e(t-1) + e(t)e^2(t-1) + \frac{e^3(t-1)}{3} + e(t) + e(t-1) + t_{rt}(t) + t_{rt}(t-1) + OS(t)t_{rt}(t) + OS(t-1)t_{rt}(t-1) + t_{st}(t) + t_{st}(t-1) + OS(t)t_{st}(t) + OS(t-1)t_{st}(t-1)) + 7k \tag{27}$$

We consider the Lyapunov stability theorem in each sampling time t. In the case of $\Delta V(t) \leq 0$, the new particle is generated according to (12), and thus, the stability of the closed-loop control system of the NSPSO is verified. ■

IV. AN ANALYSIS OF THE SIMULATION AND RESULTS

To verify the performance, robustness and convergence of the proposed method, the results of the simulation compared between the proposed method, the particle swarm optimization (PSO) [44], the hybrid particle swarm optimization-grey wolf optimization (PSO-GWO) [38], the whale optimization algorithm (WOA) [45] and the social spider optimization algorithm (SSO) are presented per the DC motor. For robustness, the minimized cost function according to (7) is applied. During the simulation, each algorithm for comparison is executed based on execution time with 2 seconds and 100 iterations.

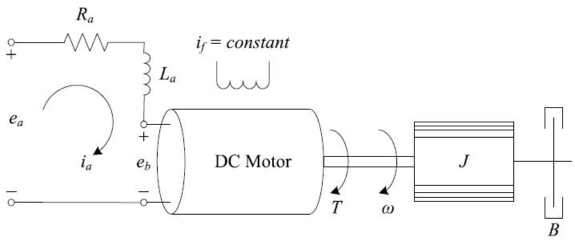

FIGURE 4. The structure of the DC motor [43].

TABLE 1. Point of operation of the DC motor.

Case No.	R_a (Ω)	K
1	0.20	0.009
2	0.20	0.021
3	0.60	0.009
4	0.60	0.021

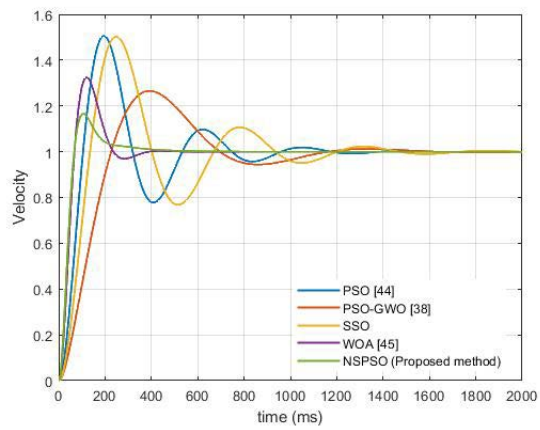

FIGURE 5. The comparison of the performance based on a fitness function as (2).

A. DC MOTOR MODELING

The structure of the DC motor shown in Figure 4 is based on electrical and mechanical principles. In [43], the step of deriving the mathematics model of the DC motor as (28) is shown by the consideration of producing the torque under the armature current as follows:

$$G(s) = \frac{K}{(L_as + R_a)(J_ms + B) + K_bK} \tag{28}$$

R_a is the armature resistance, J_m is the inertia torque of the motor, K is the torque of the motor, B is the motor friction constant and k_b is the constant of the electromotive force. In this paper, J_m is 0.0004 $kg.m^2$, B is 0.0022 $N.m.s/rad$ and K_b is 0.05 $V.s$. R_a and K are varied with 4 cases, as shown in table 1, to verify the robustness [43].

B. NUMERICAL SIMULATION

To verify the performance and convergence of the autotuning PID parameter, the comparative simulation between the proposed method, the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO are performed according to fitness

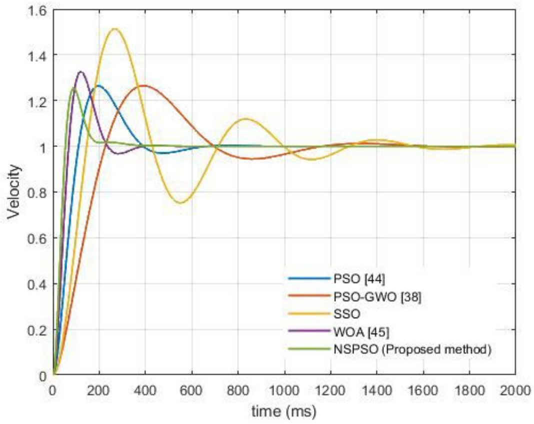

FIGURE 6. The comparison of the performance based on a fitness function as (3).

TABLE 2. A summary of the comparison of convergence.

Fitness Function	Algorithm	K_P	K_I	K_D	$J(t)$
1	PSO [44]	0.422	1.116	0.100	0.944
	PSO-GWO [38]	0.268	0.226	0.057	0.5973
	SSO	0.107	0.177	0.102	1.775
	WOA [45]	0.610	0.900	0.610	0.591
	NSPSO (Proposed method)	0.372	1.157	0.372	0.295
2	PSO [44]	0.323	1.047	0.100	0.345
	PSO-GWO [38]	1.557	1.055	0.071	0.626
	SSO	1.557	1.055	0.071	1.061
	WOA [45]	0.610	0.900	0.610	0.714
	NSPSO (Proposed method)	0.551	1.481	0.792	0.230
3	PSO [44]	0.187	0.495	0.100	0.437
	PSO-GWO [38]	0.243	0.472	0.054	0.409
	SSO	0.124	0.152	0.135	0.287
	WOA [45]	0.610	0.900	0.610	0.356
	NSPSO (Proposed method)	0.372	1.157	0.372	0.273
4	PSO [44]	0.125	0.290	0.100	0.494
	PSO-GWO [38]	0.138	0.288	0.081	0.441
	SSO	0.104	0.140	0.128	0.276
	WOA [45]	0.102	0.150	0.102	0.216
	NSPSO (Proposed method)	0.372	1.157	0.372	0.213

functions (2), (3), (6) and (7). To verify the system performance, the comparative results of the transient response based on each fitness function are shown in Figures5-8, while the comparative results of the convergence curve based on each fitness function are shown in Figures 9-12. The results of K_P, K_I and K_D and minimizing the fitness function are summarized in Table 2. The results obtained for the transient response analysis are shown in Table 3. Additionally, Table 4 shows the summary of the characteristic convergence which is the evaluated speed for each algorithm to approach the best fitness function with (29)-(30) according to [15].

$$\mu = \frac{\sum_{i=1}^{N} f(J_i)}{N} \tag{29}$$

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(f(J_i) - \mu)^2} \tag{30}$$

TABLE 3. A summary of the comparison of the transient response.

Fitness Function	Algorithm	t_s (ms)	t_r (ms)	OS (%)	E_s (10^{-3})
1	PSO [44]	0.917	0.072	50.5	-0.213
	PSO-GWO [38]	1.061	0.161	26.6	0.921
	SSO	1.357	0.093	50.2	0.106
	WOA [45]	0.326	0.046	32.4	-0.001
	NSPSO (Proposed)	0.310	0.045	16.6	-0.001
2	PSO [44]	0.550	0.077	26.4	0.001
	PSO-GWO [38]	1.060	0.161	26.5	0.906
	SSO	1.432	0.100	51.4	6.026
	WOA [45]	0.329	0.046	32.6	0.000
	NSPSO (Proposed)	0.184	0.036	25.4	0.000
3	PSO [44]	0.683	0.145	10.1	0.000
	PSO-GWO [38]	1.059	0.161	26.7	0.000
	SSO	0.850	0.125	23.7	0.000
	WOA [45]	0.327	0.046	32.9	0.000
	NSPSO (Proposed)	0.310	0.045	16.6	0.000
4	PSO [44]	0.664	0.145	14.4	0.000
	PSO-GWO [38]	1.006	0.155	23.0	0.000
	SSO	1.178	0.096	46.0	0.000
	WOA [45]	1.073	0.163	29.6	1.201
	NSPSO (Proposed)	0.310	0.045	16.6	0.000

TABLE 4. A summary of characteristic convergence.

Algorithm	Fitness function 1	Fitness function 2	Fitness function 3	Fitness function 4
PSO [44]	0.317	0.274	0.073	0.06
PSO_GWO [38]	0.384	0.197	0.079	0.068
SSO	0.061	0.058	0.106	0.147
WOA [45]	0.407	0.176	0.095	0.166
NSPSO (Proposed)	0.492	0.299	0.114	0.169

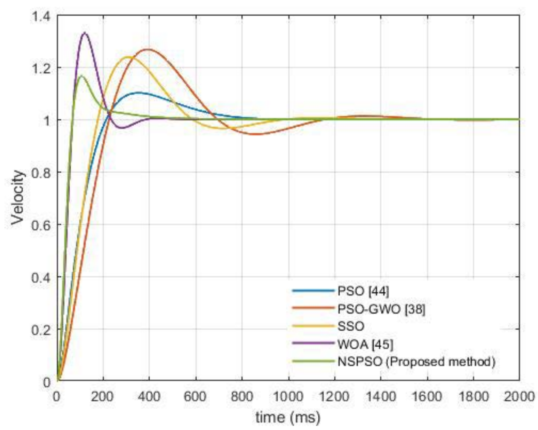

FIGURE 7. The comparison of the performance based on a fitness function as (6).

where $f(J_i)$ is the result of fitness function for each particle and N is the number of particle.

To verify the robustness based on the fitness function as (7), the comparative simulations of transient response analysis with 4 cases as shown in Table 1 by changing the operation point of DC motor for each algorithm are shown in Figures 13-16 and summarized in Table 5.

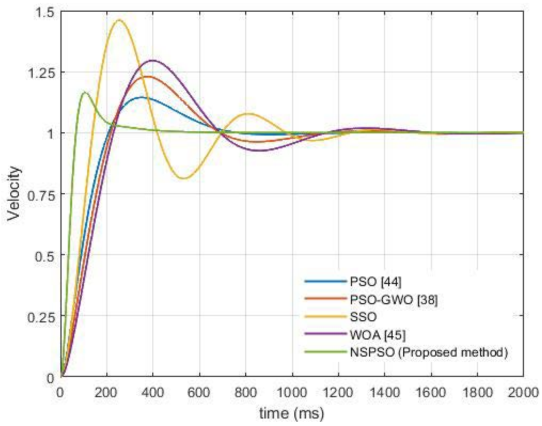

FIGURE 8. The comparison of the performance based on a fitness function as (7).

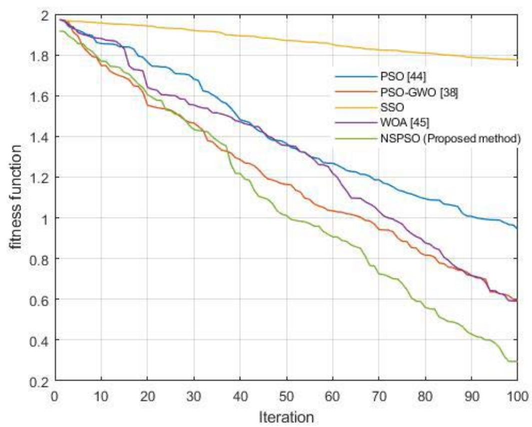

FIGURE 9. The comparison of the convergence based on a fitness function as (2).

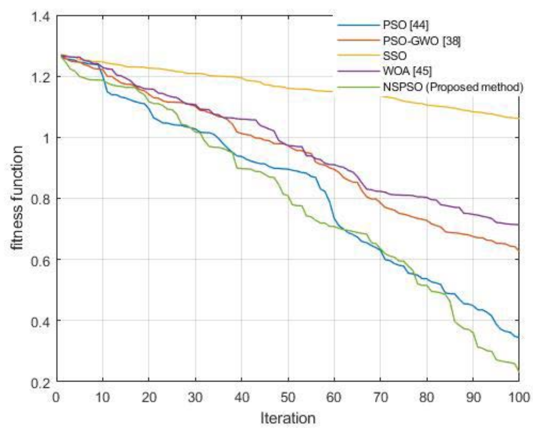

FIGURE 10. The comparison of the convergence based on a fitness function as (3).

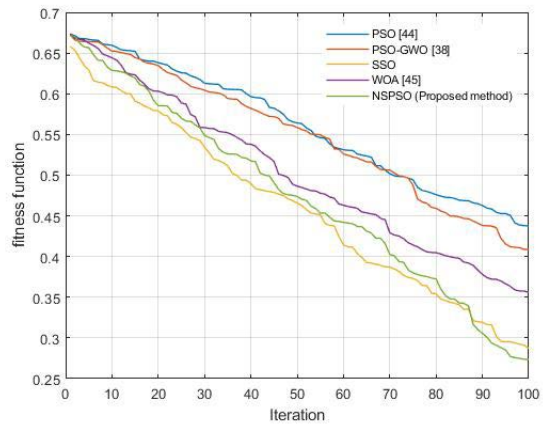

FIGURE 11. The comparison of the convergence based on a fitness function as (6).

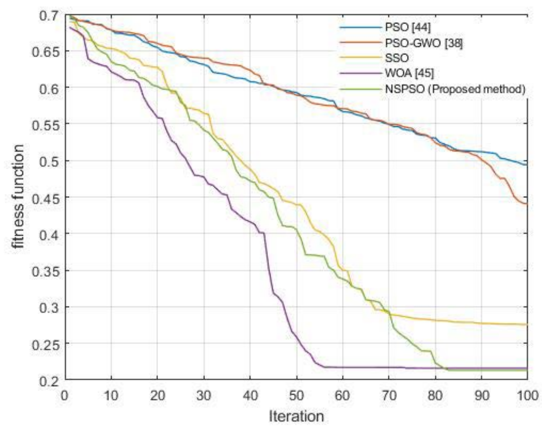

FIGURE 12. The comparison of the convergence based on a fitness function as (7).

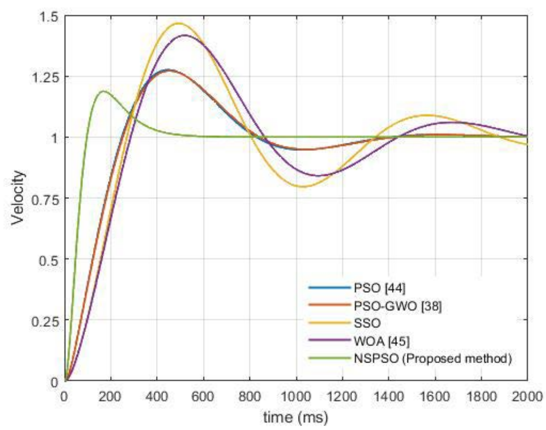

FIGURE 13. The comparison of robustness based on the operation of case 1.

According to the comparative performance, the proposed method provides better performance than the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO for all fitness functions except the fitness function in (7). Figures 7-11 show the effectiveness of the proposed method compared with others approaches. It is necessary to note that although the proposed method may provide the maximum overshoot greater than that of the PSO [44], the value of overshoot can be reduced by increasing the number of iteration of tuning. For the comparative convergence based on minimizing the fitness function, the proposed algorithm provides better results

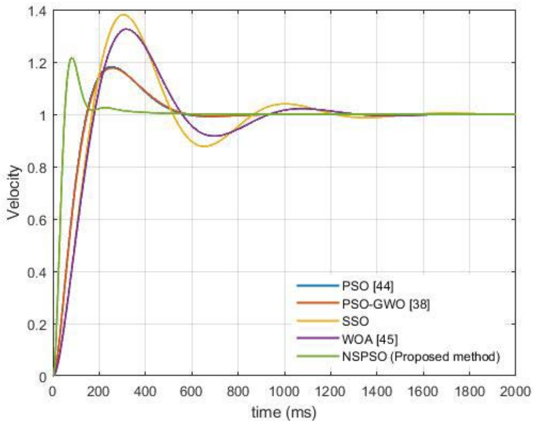

FIGURE 14. The comparison of robustness based on the operation of case 2.

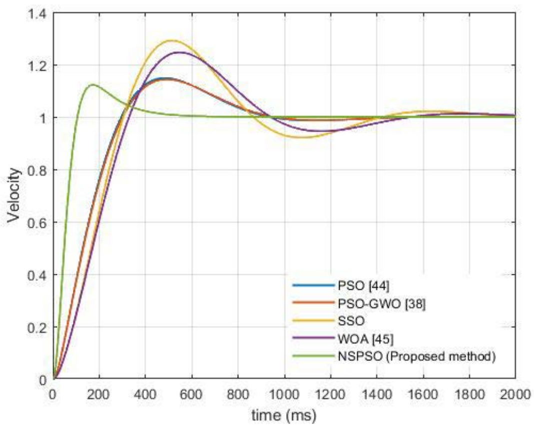

FIGURE 15. The comparison of robustness based on the operation of case 3.

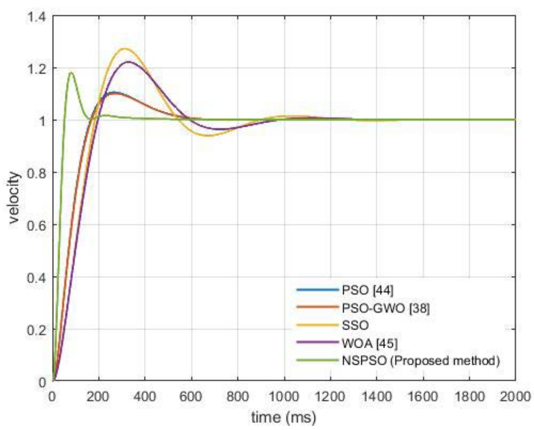

FIGURE 16. The comparison of robustness based on operation of case 4.

than the other algorithms. Regarding the comparative result of characteristic convergence as Table 4 caused by Figures 9-12, the proposed method has the shortest time approaching the optimal value compared with other algorithms such as the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO for all fitness functions with the same initial value.

For comparative robustness by changing the operation point of motor, the proposed method provides better results

TABLE 5. A summary of the comparison of robustness.

Case	Algorithm	t_s (ms)	t_r(ms)	E_s (10^{-3})	OS (%)	$J(t)$
1	PSO [44]	1.270	0.179	∞	27.5	0.142
	PSO-GWO [38]	1.289	0.182	∞	27.1	0.143
	SSO	1.916	0.185	0.031	46.6	0.234
	WOA [45]	1.911	0.205	-0.004	41.6	0.222
	NSPSO (Proposed)	0.426	0.066	0.000	18.7	0.052
2	PSO [44]	0.505	0.101	0.000	18.2	0.071
	PSO-GWO [38]	0.513	0.102	0.000	17.7	0.071
	SSO	1.134	0.119	0.001	38.1	0.124
	WOA [45]	1.111	0.126	0.000	32.5	0.117
	NSPSO (Proposed)	0.267	0.034	0.000	21.7	0.030
3	PSO [44]	0.847	0.211	-0.001	14.8	0.127
	PSO-GWO [38]	0.860	0.215	-0.001	14.3	0.128
	SSO	1.719	0.216	0.001	29.2	0.186
	WOA [45]	1.454	0.238	-0.006	24.6	0.181
	NSPSO (Proposed)	0.412	0.071	0.000	12.2	0.049
4	PSO [44]	0.505	0.112	0.000	10.4	0.068
	PSO-GWO [38]	0.512	0.114	0.000	10.0	0.068
	SSO	0.833	0.129	0.000	27.1	0.107
	WOA [45]	0.859	0.139	0.000	22.0	0.104
	NSPSO (Proposed)	0.140	0.035	0.000	18.0	0.028

than the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO for all operation points based on transient response analysis and convergence. For example, according to the operation point as case 1 and regarding Figure 13, the proposed method obtains the better transient with no steady state error, less settling time and less rise time, while the transient response of the PSO [44] and the PSO-GWO [38] are slowly approaching the steady state. In addition, the WOA [45] and the SSO provide more settling time and rise time than the proposed method.

With respect to the comparative performance and robustness based on the DC motor control system, it is clear that the proposed NSPSO is effective in autotuning the PID parameter.

V. CONCLUSION AND DISCUSSION

The new method to autotune the PID parameter called the novel stable particle swarm optimization (NSPSO) algorithm is proposed in this paper to improve its performance, convergence, robustness, and stability. To check the performance and convergence, the comparison simulation based on DC motor is applied with different fitness functions such as IAE, ITSE, IST, OS, RT and ST. To test the superiority of robustness, the comparative simulation based on DC motor is applied with different operating points of DC motor, and finally the stability is tested according to the sense of Lyapunov stability. As seen in the comparative simulation based on the DC motor, the proposed method provides a grater result for performance, convergence and robustness than the PSO [44], the PSO-GWO [38], the WOA [45] and the SSO due to a reconsideration of the suitability of the remaining particle in the search space according to (12), which proves the stability according to the sense of Lypunov stability. This is included in the process of NSPSO for a newly generated new particle, while the other algorithms only try to modify the value of each particle in the search space. In other words, the performance of other algorithms depends on the initial value of each particle. Sometimes, in the unstable system, their value in the search space approaches the local minima. Therefore, the proposed method can handle unstable systems with respect to internal operation changes, and thus, the proposed theory can be claimed to perform in practical applications. The case of instability from external noise has not yet considered in this paper; however, this issue will be considered in our future research work by proposing the new algorithm of autotuning based on an unpredictable term.

APPENDIX A NOMENCLATURE

Abbreviation	Description
E_S	Steady state error
OS	Maximum overshoot
R	Reference input
Y	Output of the system
e	Error of system
$pbest$	Local best position
$gbest$	Best position of the whole particle
v	Velocity of particle
J	Fitness function
x	Position of particle
c_1	Cognitive learning factor
c_2	Social learning factor
ω_{ie}	Weights of IE
ω_{iae}	Weights of IAE
ω_{ise}	Weights of ISE
ω_{td}	Weights of overshoot
ω_r	Weights of rise time
ω_s	Weights of settling time
β	Levy constant
d	Current dimension
p	Particle in the seareced space
K	Torque of the motor
R_a	Armature resistance

APPENDIX B SYSTEM AND SIMULATION SETTING

Abbreviation	Description	Value
J_m	Inertia torque of the motor	0.0004 $kg.m^2$
B	Motor friction constant	0.0022 $N.m.s/rad$
k_b	Constant of the electromotive force	0.05 V
K_P	Proportional gain	$0\leq K_P \leq 2$
K_I	Integral gain	$0\leq K_I \leq 2$
K_D	Derivation gain	$0\leq K_D \leq 1$
t	Time of tuning in each iteration of tuning	2 s
D	Dimension of searched space	3
N	Number of particles	50
i_{th}	Iteration	100

ACKNOWLEDGMENT

The authors would like to thank the faculty of engineering, King Mongkut's University of Technology Thonburi for supporting the fund of this research.

REFERENCES

[1] H. Habibi, H. R. Nohooji, and I. Howard, "Adaptive PID control of wind turbines for power regulation with unknown control direction and actuator faults," *IEEE Access*, vol. 6, pp. 37464–37479, 2018.

[2] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, "Fuzzy self-tuning PID semiglobal regulator for robot manipulators," *IEEE Trans. Ind. Electron.*, vol. 59, no. 6, pp. 2709–2717, Jun. 2012.

[3] K. M. Goher and S. O. Fadlallah, "Control of a two-wheeled machine with two-directions handling mechanism using PID and PD-FLC algorithms," *Int. J. Autom. Comput.*, vol. 16, no. 4, pp. 511–533, Aug. 2019.

[4] R. K. Mandava and P. R. Vundavilli, "Implementation of modified chaotic invasive weed optimization algorithm for optimizing the PID controller of the biped robot," *Sādhanā*, vol. 43, no. 5, pp. 1–18, May 2018.

[5] R. M. Asl, E. Pourabdollah, and M. Salmani, "Optimal fractional order PID for a robotic manipulator using colliding bodies design," *Soft Comput.*, vol. 22, no. 14, pp. 4647–4659, Jul. 2018.

[6] L. R. R. D. Santos, F. R. Durand, and T. Abrao, "Adaptive PID scheme for OCDMA next generation PON based on heuristic swarm optimization," *IEEE Syst. J.*, vol. 13, no. 1, pp. 500–510, Mar. 2019.

[7] J. Zhao, P. K. Wong, Z. Xie, X. Ma, and X. Hua, "Design and control of an automotive variable hydraulic damper using cuckoo search optimized PID method," *Int. J. Automot. Technol.*, vol. 20, no. 1, pp. 51–63, 2019.

[8] M. Chavoshian, M. Taghizadeh, and M. Mazare, "Hybrid dynamic neural network and PID control of pneumatic artificial muscle using the PSO algorithm," *Int. J. Autom. Comput.*, vol. 17, no. 3, pp. 428–438, Jun. 2020.

[9] H.-P. Ren, J.-T. Fan, and O. Kaynak, "Optimal design of a fractional-order proportional-integer-differential controller for a pneumatic position servo system," *IEEE Trans. Ind. Electron.*, vol. 66, no. 8, pp. 6220–6229, Aug. 2019.

[10] J. P. de Moura, J. V. da Fonseca Neto, and P. H. M. Rego, "Models for optimal online tuning based on computational intelligence of PID controllers applied to operational processes of bulk reclaimers," *J. Control, Autom. Electr. Syst.*, vol. 30, no. 2, pp. 148–159, Apr. 2019.

[11] M. Dangor, O. A. Dahunsi, J. O. Pedro, and M. M. Ali, "Evolutionary algorithm-based PID controller tuning for nonlinear quarter-car electrohydraulic vehicle suspensions," *Nonlinear Dyn.*, vol. 78, no. 4, pp. 2795–2810, Dec. 2014.

[12] M. Ghanamijaber, "A hybrid fuzzy-PID controller based on gray wolf optimization algorithm in power system," *Evolving Syst.*, vol. 10, no. 2, pp. 273–284, Jun. 2019.

[13] N. N. Sari, H. Jahanshahi, and M. Fakoor, "Adaptive fuzzy PID control strategy for spacecraft attitude control," *Int. J. Fuzzy Syst.*, vol. 21, no. 3, pp. 769–781, Apr. 2019.

[14] L. R. R. Dos Santos, F. R. Durand, A. Goedtel, and T. Abrao, "Auto-tuning PID distributed power control for next-generation passive optical networks," *J. Opt. Commun. Netw.*, vol. 10, no. 10, pp. 110–125, 2018.

[15] J. Pongfai, X. Su, H. Zhang, and W. Assawinchaichote, "A novel optimal PID controller autotuning design based on the SLP algorithm," *Expert Syst.*, vol. 37, no. 2, pp. 1–15, Apr. 2020.

[16] B. Verma and P. K. Padhy, "Optimal PID controller design with adjustable maximum sensitivity," *IET Control Theory Appl.*, vol. 12, no. 8, pp. 1156–1165, May 2018.

[17] S. Kiong Nguang, W. Assawinchaichote, P. Shi, and Y. Shi, "H_∞ fuzzy filter design for uncertain nonlinear systems with Markovian jumps: An LMI approach," in *Proc. Amer. Control Conf.*, 2015, pp. 1799–1804.

[18] S. K. Nguang, W. Assawinchaichote, and P. Shi, "H_∞ filter for uncertain Markovian jump nonlinear systems: An LMI approach," *Circuits, Syst. Signal Process.*, vol. 26, no. 6, pp. 853–874, Dec. 2007.

[19] J. Pongfai, X. Su, H. Zhang, and W. Assawinchaichote, "PID controller autotuning design by a deterministic Q-SLP algorithm," *IEEE Access*, vol. 8, pp. 50010–50021, 2020.

[20] M. Raju, L. C. Saikia, and N. Sinha, "Automatic generation control of a multi-area system using ant lion optimizer algorithm based PID plus second order derivative controller," *Int. J. Elect. Power Energy Syst.*, vol. 80, pp. 52–63, Sep. 2016.
[21] A. Zamani, S. M. Barakati, and S. Yousofi-Darmian, "Design of a fractional order PID controller using GBMO algorithm for load–frequency control with governor saturation consideration," *ISA Trans.*, vol. 64, pp. 56–66, Sep. 2016.
[22] K. M. Goher and S. O. Fadlallah, "PID, BFO-optimized PID, and PD-FLC control of a two-wheeled machine with two-direction handling mechanism: A comparative study," *Robot. Biomimetics*, vol. 5, no. 1, pp. 1–16, Dec. 2018.
[23] M. P. Nagarkar, Y. J. Bhalerao, G. J. Vikhe Patil, and R. N. Z. Patil, "GA-based multi-objective optimization of active nonlinear quarter car suspension system—PID and fuzzy logic control," *Int. J. Mech. Mater. Eng.*, vol. 13, no. 1, pp. 1–20, Dec. 2018.
[24] D. Guha, P. K. Roy, and S. Banerjee, "Binary bat algorithm applied to solve MISO-type PID-SSSC-based load frequency control problem," *Iranian J. Sci. Technol., Trans. Electr. Eng.*, vol. 43, no. 2, pp. 323–342, Jun. 2019.
[25] M. Rahman, Z. C. Ong, W. T. Chong, S. Julai, and X. W. Ng, "Wind turbine tower modeling and vibration control under different types of loads using ant colony optimized PID controller," *Arabian J. Sci. Eng.*, vol. 44, no. 2, pp. 707–720, Feb. 2019.
[26] M. M. Gani, M. S. Islam, and M. A. Ullah, "Optimal PID tuning for controlling the temperature of electric furnace by genetic algorithm," *Social Netw. Appl. Sci.*, vol. 1, no. 8, pp. 1–8, Aug. 2019.
[27] T. Samakwong and W. Assawinchaichote, "PID controller design for electro-hydraulic servo valve system with genetic algorithm," *Proc. Comput. Sci.*, vol. 86, pp. 91–94, Jan. 2016.
[28] N. Razmjooy, M. Khalilpour, and M. Ramezani, "A new meta-heuristic optimization algorithm inspired by FIFA world cup competitions: Theory and its application in PID designing for AVR system," *J. Control, Automat. Elect. Syst.*, vol. 27, no. 4, pp. 419–440, 2016.
[29] P. B. de Moura Oliveira, H. Freire, and E. J. Solteiro Pires, "Grey wolf optimization for PID controller design with prescribed robustness margins," *Soft Comput.*, vol. 20, no. 11, pp. 4243–4255, Nov. 2016.
[30] S. Ekinci and B. Hekimoglu, "Improved kidney-inspired algorithm approach for tuning of PID controller in AVR system," *IEEE Access*, vol. 7, pp. 39935–39947, 2019.
[31] S. C. Sahoo, A. K. Barik, and D. C. Das, "A novel green leaf-hopper flame optimization algorithm for competent frequency regulation in hybrid microgrids," *Int. J. Numer. Modelling, Electron. Netw., Devices Fields*, vol. 35, no. 3, pp. 1–13, May 2022.
[32] Z. Z. Qiang, L. S. Jian, and P. J. Shyang, "A PID parameter tuning method based on the improved QUATRE algorithm," *Algorithms*, vol. 14, no. 3, pp. 1–14, 2021.
[33] B. Guo, Z. Zhuang, J.-S. Pan, and S.-C. Chu, "Optimal design and simulation for PID controller using fractional-order fish migration optimization algorithm," *IEEE Access*, vol. 9, pp. 8808–8819, 2021.
[34] A. Sungthong and W. Assawinchaichote, "Particle swam optimization based optimal PID parameters for air heater temperature control system," *Proc. Comput. Sci.*, vol. 86, pp. 108–111, Jan. 2016.
[35] Z. Long, Z. Jiang, C. Wang, Y. Jin, Z. Cao, and Y. Li, "A novel approach to control of piezo-transducer in microelectronics packaging: PSO-PID and editing trajectory optimization," *IEEE Trans. Compon., Packag., Manuf. Technol.*, vol. 10, no. 5, pp. 795–805, May 2020.
[36] B. M Brentan, E. Luvizotto, Jr., I. Montalvo, J. Izquierdo, and R. Pérez-García, "Position control of nonlinear hydraulic system using an improved PSO based PID controller," *Proc. Eng.*, vol. 83, pp. 241–259, Jan. 2017.
[37] H. Zhang, W. Assawinchaichote, and Y. Shi, "New PID parameter autotuning for nonlinear systems based on a modified monkey–multiagent DRL algorithm," *IEEE Access*, vol. 9, pp. 78799–78811, 2021.
[38] A. K. Mishra, S. R. Das, P. K. Ray, R. K. Mallick, A. Mohanty, and D. K. Mishra, "PSO-GWO optimized fractional order PID based hybrid shunt active power filter for power quality improvements," *IEEE Access*, vol. 8, pp. 74497–74512, 2020.
[39] Z. Qi, Q. Shi, and H. Zhang, "Tuning of digital PID controllers using particle swarm optimization algorithm for a CAN-based DC motor subject to stochastic delays," *IEEE Trans. Ind. Electron.*, vol. 67, no. 7, pp. 5637–5646, Jul. 2020.
[40] L. Jia and X. Zhao, "An improved particle swarm optimization (PSO) optimized integral separation PID and its application on central position control system," *IEEE Sensors J.*, vol. 19, no. 16, pp. 7064–7071, Aug. 2019.
[41] A. K. Yadav and P. Gaur, "An optimized and improved STF-PID speed control of throttle controlled HEV," *Arabian J. Sci. Eng.*, vol. 41, no. 9, pp. 3749–3760, Sep. 2016.
[42] Z. Wang, R. Zu, D. Duan, and J. Li, "Tuning of ADRC for QTR in transition process based on NBPO hybrid algorithm," *IEEE Access*, vol. 7, pp. 177219–177240, 2019.
[43] B. Hekimoğlu, "Optimal tuning of fractional order PID controller for DC motor speed control via chaotic atom search optimization algorithm," *IEEE Access*, vol. 7, pp. 38100–38114, 2019.
[44] L. Jia and X. Zhao, "An improved particle swarm optimization (PSO) optimized integral separation PID and its application on central position control system," *IEEE Sensors J.*, vol. 19, no. 16, pp. 7046–7071, 2019.
[45] A. M. Mosaad, M. A. Attia, and A. Y. Abdelaziz, "Whale optimization algorithm to tune PID and PIDA controllers on AVR system," *Ain Shams Eng. J.*, vol. 10, no. 4, pp. 755–767, Dec. 2019.
[46] J. Pongfai, W. Assawinchaichote, P. Shi, and X. Su, "Novel D-SLP controller design for nonlinear feedback control," *IEEE Access*, vol. 8, pp. 128796–128808, 2020.

WUDHICHAI ASSAWINCHAICHOTE received the B.S. degree (Hons.) in electrical engineering from Assumption University, Bangkok, Thailand, in 1994, the M.E. degree in electrical engineering from Pennsylvania State University (Main Campus), PA, USA, in 1997, and the Ph.D. degree in electrical engineering from the University of Auckland, New Zealand, in 2004. He is currently an Associate Professor with the Department of Electronic and Telecommunication Engineering, King Mongkut's University of Technology Thonburi (KMUTT), Bangkok. He has published a research monograph and more than 20 research papers in international refereed journals indexed by SCI/SCIE (Clarivate Analytics). His current research interests include fuzzy control, robust control, optimal control, system and control theory, computational intelligence, and PID controller design. He currently serves as an Associate Editor for the *International Journal of Innovative Computing, Information and Control*, and serves as a Reviewer for IEEE Transactions on Industrial Electronics, IEEE Transactions on Fuzzy Systems, IEEE Transactions on Cybernetics, *Neural Computing and Applications*, and IEEE Access.

CHRISSANTHI ANGELI received the Master of Science degree in intelligent systems from the University of Plymouth, U.K., in 1993, and the Ph.D. degree in intelligent fault detection techniques from the University of Sussex, U.K., in 1999. She is currently a Professor with the Department of Electrical and Electronic Engineering, University of West Attica, Athens, Greece. Her current research interests include real-time expert systems, web-based expert systems, and artificial intelligence techniques for fault detection.

JIRAPUN PONGFAI received the Bachelor of Engineering degree in computer engineering from Naresuan University (NU), Pitsanulok, Thailand, in 2015, and the Master of Engineering degree in electrical and information engineering and the Doctor of Engineering degree in electrical and information engineering technology from the King Mongkut's University of Technology Thonburi (KMUTT), Bangkok, Thailand, in 2017 and 2020, respectively. Her research interests include intelligence control, optimal and predictive controller design, artificial intelligence, and embedded system design. She currently serves as an Editorial Board for *Journal of Electrical and Electronic Engineering* and serves as a Reviewer for IEEE Access.