Novel D-SLP Controller Design for Nonlinear Feedback Control

IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 4, 2020, accepted July 10, 2020, date of publication July 14, 2020, date of current version July 23, 2020.
Digital Object Identifier 10.1109/ACCESS.2020.3009178

Novel D-SLP Controller Design for Nonlinear Feedback Control

JIRAPUN PONGFAI[1], WUDHICHAI ASSAWINCHAICHOTE[1], (Member, IEEE), PENG SHI[2], (Fellow, IEEE), AND XIAOJIE SU[3], (Senior Member, IEEE)
[1]Department of Electronic and Telecommunication Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand
[2]The University of Adelaide, Adelaide, SA 5005, Australia
[3]College of Automation, Chongqing University, Chongqing 400044, China
Corresponding author: Wudhichai Assawinchaichote (wudhichai.asa@kmutt.ac.th)

This work was supported by the Petchra Pra Jom Klao Scholarship.

ABSTRACT Novel nonlinear feedback control based on the dragonfly swarm learning process (D-SLP) algorithm is proposed in this paper. This approach improves the performance, stability and robustness of designing the nonlinear system controller. The D-SLP algorithm is the combination of the dragonfly algorithm (DA) and swarm learning process (SLP) algorithm by applying the DA to the learning process of the SLP algorithm. Furthermore, the estimation of the nonlinear term by using gradient descent is proposed in the process of the D-SLP algorithm. The learning rate is adjusted according to the stable learning rate, which is derived according to the Lyapunov stability theorem. To show the superior performance and robustness of the proposed control method, it is compared with the simulation of designing the controller based on a permanent magnet synchronous motor (PMSM) control system with the online autotuning parameter of a PID controller and LQR controller with two case studies. The conventional SLP algorithm and DA are used to autotune the PID controller, while an artificial bee colony algorithm and a flower pollination algorithm (ABC-FPA) autotune the LQR controller. From the simulation results, the proposed control method can provide a better response than the other control method. Additionally, the global convergence of the D-SLP algorithm is analyzed according to Markov chain modeling and proved to correspond with the policy of global convergence for stochastic search algorithms.

INDEX TERMS Dragonfly algorithm (DA), gradient descent method, Markov chain modeling, nonlinear control, nonlinear estimation, permanent magnet synchronous motor (PMSM), swarm learning process (SLP) algorithm.

I. INTRODUCTION

The nonlinearity of control systems generally appears during practical applications such as complex industrial processes and mechanical, power and traffic systems. For some time, this issue has aggravated the performance of control law and made control difficult [1]–[3]. Therefore, the nonlinear control system has been popularly investigated as the solution for solving this problem [2]–[4]. Many years ago, numerous control methods were proposed to design a nonlinear control system, including sliding mode control (SMC), H-infinity (H_∞) control, linear quadratic regulator (LQR), and fuzzy control [5]–[18].

Reference [14] proposed the SMC to reconstruct the system state, which influences actuator degradation. Reference [15] presented the composite SMC to control PMSM system speed and use load torque for feed-forward compensation. Reference [16] proposed the new sliding-mode reaching law (NSMRL) to optimize the dynamic performance of a permanent magnet synchronous motor (PMSM). Reference [17] proposed the H_∞ controller for aero-engine wireless networked systems. The H_∞ controller is designed based on maximum error first-try once discard scheduling. Reference [18] proposed finite time H_∞ control for hydraulic turbine governing systems. Reference [19] studied applying the digital H_∞ controller design suitable for uninterruptible power supply inverters. Reference [20] proposed the LQR for quadrotor helicopters. This approach tunes the LQR by a feedback gain matrix (K). Reference [21] presented a digital dual-mode linear LQR with feed-forward optimal controller for nonminimum phase boost converters. Reference [22] designed the combination of

The associate editor coordinating the review of this manuscript and approving it for publication was Juntao Fei.

passive-base-isolated (PBI) and active structural control (ASC) by using the LQR. However, the approach requires complex calculation, especially in high-complexity systems, and it is difficult and requires time to solve the mathematical model. Thus, recently, programming methods such as fuzzy logic control, neural network control, backstepping control, model predictive control and so on have been proposed to design the controller. Reference [23] designed a controller for a class of nonlinear pure-feedback multiple-input multiple-output (MIMO) systems with unknown dead-zones by using adaptive fuzzy logic. Reference [24] proposed adaptive fuzzy tracking control designed to address the problem of single input and single output (SISO). Reference [25] presented the fuzzy logic approach to estimate the uncertain function and using fuzzy observer for approximating the measurable state. Reference [26] proposed the Takagi–Sugeno fuzzy model with linear fractional parametric uncertainties to estimate the nonlinear term of dynamic system, while [27] proposed the neural network to approximate the unpredictable term and used the neural network observer to estimate the immeasurable state. Reference [28] proposed an adaptive fuzzy containment control problem for multiple uncertain nonlinear strict-feedback systems with immeasurable states and multiple leaders under directed communication graphs. Reference [29] proposed a neural network based on finite time horizon for optimal nonlinear control. Reference [30] developed a neural network for nonlinear systems with unmodeled dynamics and immeasurable states. Reference [31] investigated adaptive neural control for unknown MIMO systems with time-varying asymmetric output constraints. Reference [30] presented the development of an adaptive neural controller for a class of nonlinear systems with unmodeled dynamics and immeasurable states. Reference [32] proposed a combination of backstepping techniques based on an adaptive consensus tracking control strategy for a class of high-order nonlinear multiagent systems. Reference [33] applied backstepping and robust control for tracking control of quadrotors flying outdoors. Reference [34] proposed a nonlinear model predictive controller (NMPC) with dynamic particle swarm optimization (DPSO) to reduce the cold-start hydrocarbon (HC) emission of an automotive spark-ignited engine. Reference [35] proposed a method of torque control for induction machines to compensate prediction error by model-based prediction. However, the control methods are a complex control algorithm, so they often suffer from high memory capacity [36]–[38] and sensitivity to the initial value, which affects access to the local minima point as a result of system instability [38]. In practice, the memory, convergence and execution time are limited [39]. From the literature, the memory capacity and convergence issues were solved by the SLP algorithm. This algorithm was proposed to autotune the proportional–integral–derivative (PID) controller; nevertheless, its performance depends on the learning process. The learning process of the conventional SLP algorithm is adjusted by the factor of a stochastic process [39]. In the case of nonlinear control systems, when applied to design, the system may approach instability. From the literature, [40] showed that the dragonfly algorithm (DA) provides effective performance of optimal searching during exploration because it is proposed based on metaheuristics, i.e., exploration and exploitation. Consequently, this paper applies the DA to the learning process algorithm, the combination of which is called the dragonfly swarm learning process (D-SLP) algorithm, to design the controller of the nonlinear feedback control system.

The D-SLP algorithm is the combination of the SLP algorithm and DA. The concept of the proposed control method involves improving the learning process of the SLP algorithm by the DA. The process of the D-SLP algorithm applies the gradient descent method to estimate the nonlinear term of the system because the gradient descent method is always proposed to solve the optimization problem [38]. To improve the stability, the stable learning rate is derived corresponding to the Lyapunov stability theorem, which is proposed to be applied for estimating the nonlinear term in this paper. Furthermore, to show the superior performance and robustness of the proposed control method, it is compared with the simulation of designing the controller of PMSM with the PID controller and LQR controller where both controllers are adjusted online. The conventional SLP and DA are applied to autotune the PID parameter, with an artificial bee colony algorithm and a flower pollination algorithm (ABC-FPA) [41] for the LQR controller. In the verification, the two cases of step input and sinusoidal input are performed. Additionally, the global convergence of the proposed control method is analyzed corresponding to Markov chain modeling and verified by the global policy of global convergence for stochastic search algorithms.

Motivated by the above considerations, the D-SLP controller for the nonlinear control system is designed in order to overcome and improve the performance, stability and robustness of nonlinear control systems. Therefore, the contributions of this paper consist of three viewpoints:

1. In lieu of existing algorithms for designing the nonlinear controller in the previous literature, the combination of DA and SLP algorithms, called D-SLP, is proposed.

2. The performance, stability and robustness of nonlinear control systems are improved by applying the gradient descent method to estimate the nonlinear term of the D-SLP algorithm caused by the stable learning rate, which this paper derives based on the theorem of Lyapunov stability. The performance and robustness are verified by comparing the simulation of designing the controller of PMSM with the PID controller and LQR controller. The PID and LQR controllers are designed by online autotuning of the parameters.

3. The global convergence of the D-SLP algorithm is analyzed by using Markov chain modeling and judged according to the policy of global convergence for stochastic search algorithms.

This paper is organized as follows. Section II describes the problem statement of nonlinear control systems.

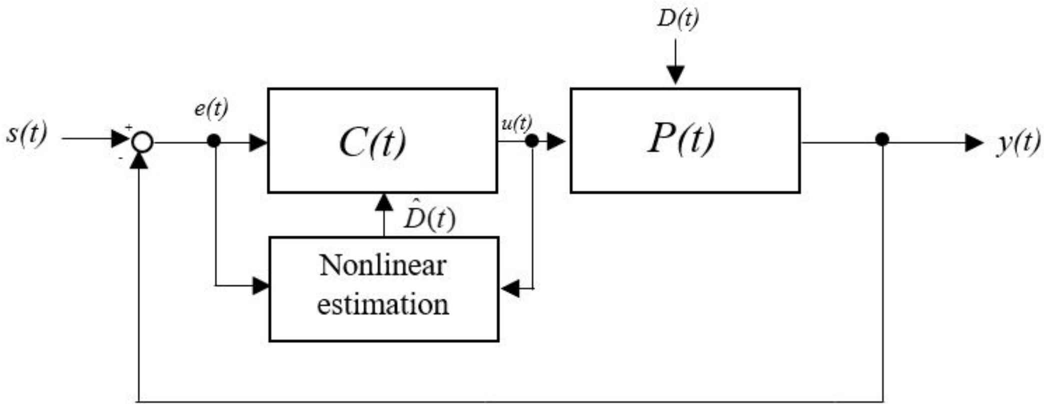

FIGURE 1. The block diagram of the D-SLP controller for the nonlinear control system.

In Section III, the D-SLP algorithm is presented. Section IV presents an illustrative example and discussion. Section V concludes this paper.

II. PROBLEM FORMULATION AND PRELIMINARIES

A. PROBLEM STATEMENT

In this paper, the nonlinear continuous time (CT) control system is considered as follows [42]:

$$\dot{x}(t) = \bar{F}(x(t), u(t)) \tag{1}$$
$$= f(x(t)) + g(u(t)) \tag{2}$$

where $x(t) \in \mathbf{R}^n$ is the state vector. $u(t) \in \mathbf{R}^m$ is the vector of control input, $f(\cdot) \in \mathbf{R}^n$ is the known nonlinear function, and $g(\cdot) \in \mathbf{R}^{n\times m}$ is the matrix of the known nonlinear function. In case nonlinear terms such as the external disturbances $(d(t))$ are added into the system, or parameters of the system differ from the nominal values, the nonlinear CT control system can be modified as follows:

$$\dot{x}(t) = (f + \Delta f)(x(t)) + (g + \Delta g)(u(t)) + d(t) \tag{3}$$
$$= f(x(t)) + g(u(t)) + D(t) \tag{4}$$

where $\Delta f(x(t))$ is the unknown nonlinear function, and $\Delta g(u(t))$ is the matrix of the unknown nonlinear function. $D(t) = \Delta f(x(t)) + \Delta g(u(t)) + d(t)$. In practice, $D(t)$ is the time variance and is unknown; this paper estimates $D(t)$ by using the stable gradient descent method. The block diagram of the proposed control method is introduced as Figure 1.

From Figure 1, $S(t)$ is the setpoint. $y(t)$ is the output. $e(t)$ is the error of the system, and $u(t)$ is the control input. $\hat{D}(t)$ is the nonlinear estimation. $P(t)$ is the system. $C(t)$ is the controller, which this paper proposes as the D-SLP algorithm. The D-SLP algorithm is a combination of the SLP algorithm and dragonfly algorithm produced by applying the operation of the dragonfly algorithm to the learning process of the SLP algorithm.

B. SLP ALGORITHM

The SLP algorithm is proposed for optimal tuning of the PID parameter and improves the performance and convergence of tuning the PID parameter by applying the concepts of the swarm algorithm and learning algorithm [39]. The approach to the optimal value by the SLP algorithm is motivated by student learning. Each student generated must have a score following the standard of the class. The class consists of a learning process that involves classifying the students into a group of bad scores and a group of good scores. The learning of each group is different. The group of bad scores studies the group of good scores and students in the bad group, while the group of good scores studies students in the same group. The student who has the best score in the class is the best student and is represented with the optimal value. Other students must study until they become better students. The flow chart of the SLP algorithm is shown in Figure 2.

From the flowchart of the SLP algorithm in Figure 2, the SLP algorithm consists of 3 operations: initialization, learning operation and selection. Initialization involves generating the student in the classroom; each student is then evaluated with a score based on the standard of the room. If the scores of students correspond to the standard of the room, they can study in the room. Otherwise, those students are sorted out, and a new student is established. For the learning process in the classroom, the students are classified into two groups: bad scores and good scores. The group of bad scores learns based on the students in the group of good scores and group of bad scores, while the group of good scores learns based on the students in the good group. Finally, the best student is selected. Selection is determined based on the score of each student. The mathematical modeling for establishing the new students, group of good scores learning and group of bad scores learning can be written as Equations 5, 6 and 7, respectively.

$$S(t)_{new} = \frac{W_n(t)\sum_{i=1}^{N} f_i(t)s_i(t)}{N} \tag{5}$$

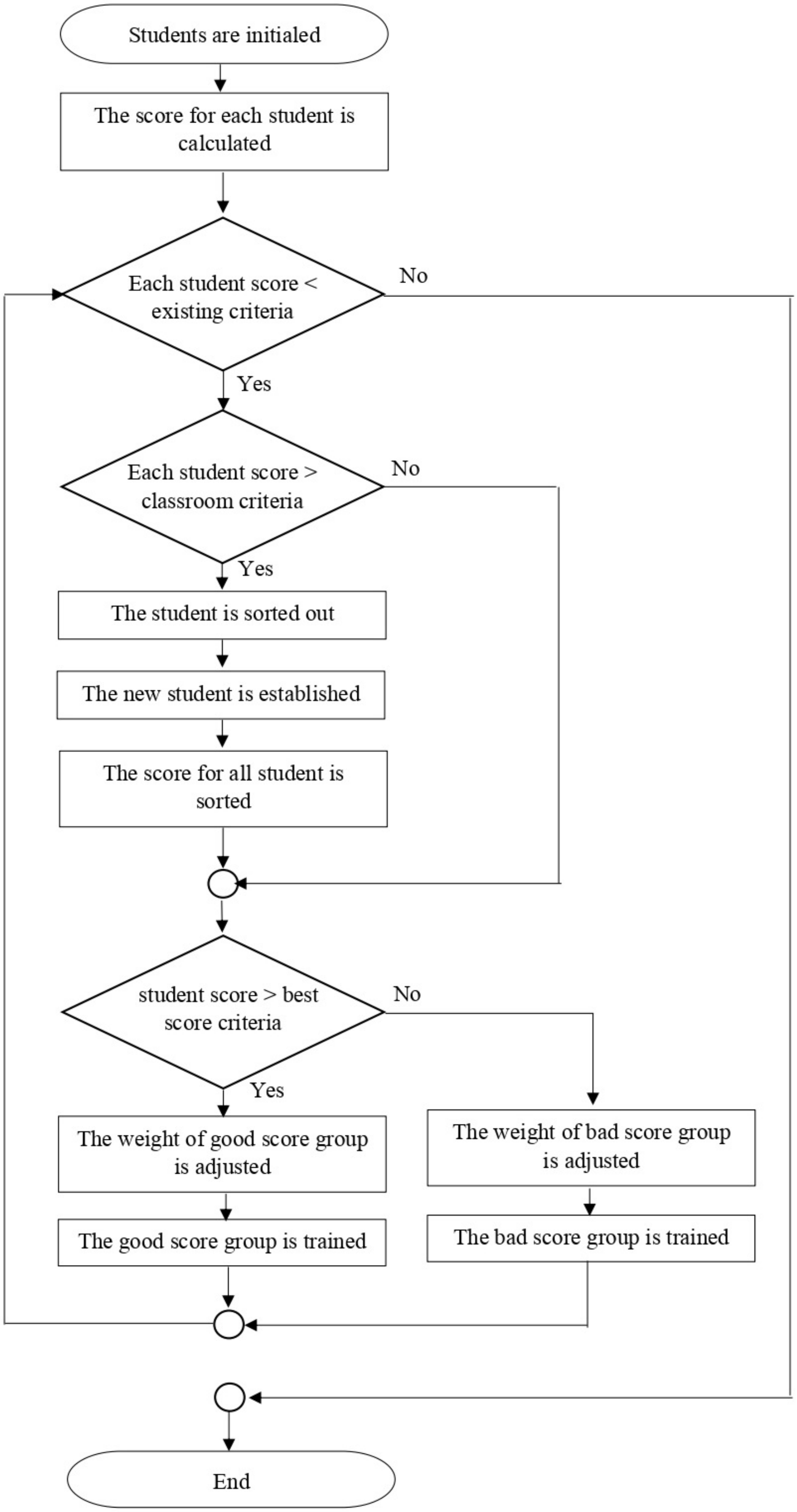

FIGURE 2. The flow chart of the conventional SLP algorithm.

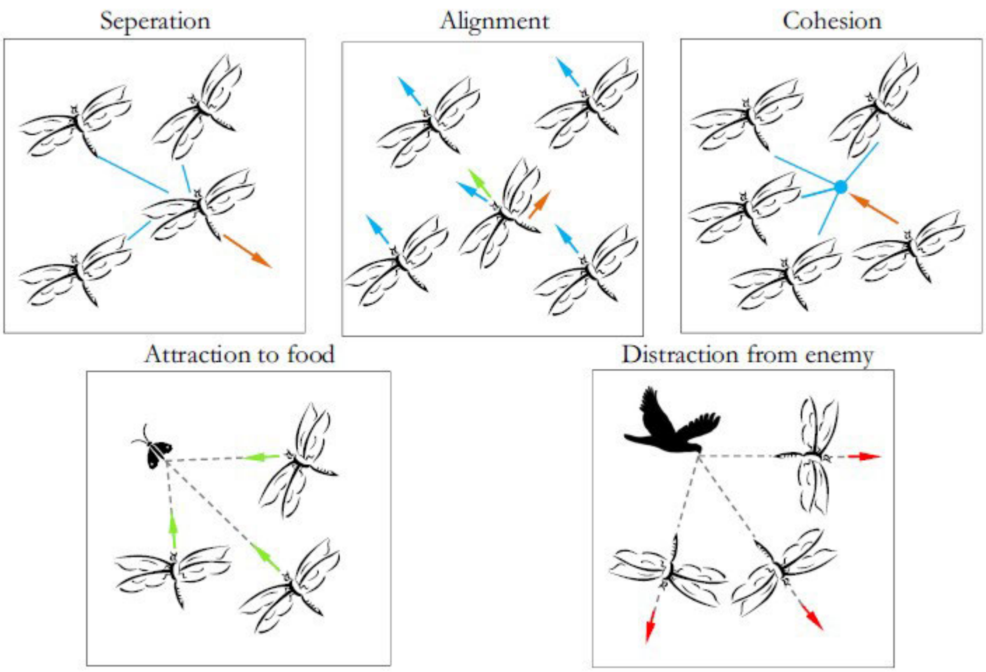

FIGURE 3. The pattern of dragonfly operation [40].

where $S(t)_{new}$ is the new student, N is the number of students in the classroom, $f_i(t)$ is the frequency of s_i in the classroom, $s_i(t)$ is the student in the classroom and $W_n(t)$ is the weight of establishing a new student operation.

$$S(t)_{good} = \frac{W_g(t) \sum_{i=1}^{M} s_i(t)}{N} \tag{6}$$

where $S(t)_{good}$ is the good student, M is the number of good students and $W_g(t)$ is the weight of good student learning.

$$S(t)_{bad} = \frac{W_b(t)(S_B(t) + S_G(t))}{2} \tag{7}$$

where $S(t)_{bad}$ is the bad student, $S_B(t)$ is the badness of a student, $S_G(t)$ is the goodness of a student and $W_b(t)$ is the weight of bad student learning.

Considering the learning process of the SLP algorithm, although classified into groups to quickly approach the optimal value, the training for each student depends on the students in the group. It is possible that the scores of students may exhibit a high distribution, such as when the students take time to become the best, especially in environments with high complexity such as nonlinear systems, and it requires time and causes system instability. Thus, this paper applied the dragonfly algorithm to improve this issue.

C. DRAGONFLY ALGORITHM

The dragonfly algorithm (DA) is a swarm algorithm that was proposed by Mirjalili [40]. It is inspired by dragonfly survival behaviors, i.e., navigating, searching for foods and avoiding enemies. The algorithm classifies the swarming into static swarming and dynamic swarming. Static swarming represents dragonfly hunting, and dynamic swarming represents dragonfly migration. The static algorithm involves dragonflies in small groups for hunting butterflies, mosquitoes, etc., in the small area. The dynamic swarm involves the migration of many dragonflies in one direction over long distances. Hunting and migration are considered to be based on the dragonfly operator, which consists of 5 corrective patterns: separation, alignment, cohesion, attraction and distraction. The operations are shown in Figure 3.

From Figure 3, separation is the avoidance of other dragonflies by each dragonfly in the neighborhood. Alignment indicates the velocity of each dragonfly to match with other dragonflies in the neighborhood. Cohesion is the tendency of each dragonfly to approach the central mass in the neighborhood. Attraction is each dragonfly moving towards the food at the same time, and distraction is each dragonfly avoiding the enemy. In the DA [40], each operation of dragonfly; i.e., the separation, the alignment, the cohesion, the attraction and the distraction is represented by the mathematical model as follows:

The mathematical model for separation is as follows:

$$S_i(t) = -\sum_{j=1}^{N} X(t) - X_j(t) \tag{8}$$

where $X(t)$ is the position of the current dragonfly, $X_j(t)$ is the position of the j^{th} neighboring dragonfly and N is the number of dragonflies. The mathematical model for alignment is as follows:

$$A_i(t) = \frac{\sum_{j=1}^{N} V_j(t)}{N} \tag{9}$$

where $V_j(t)$ is the velocity of the j^{th} neighboring dragonfly. The mathematical model for cohesion is as follows:

$$C_i(t) = \frac{\sum_{j=1}^{N} X_j(t)}{N} - X \tag{10}$$

The mathematical model for attraction is as follows:

$$F_i(t) = X^+ - X \tag{11}$$

where X^+ is the position of food. The mathematical model for distraction is as follows:

$$E_i(t) = X^- - X \tag{12}$$

where X^- is the position of the enemy.

Migration involves updating the position of each dragonfly caused by the dragonfly operation. The behavior of updating the position is denoted as follows:

$$X(t+1) = X(t) + \Delta X(t+1) \tag{13}$$

$$\Delta X(t+1) = (sS_i(t) + aA_i(t) + cC_i(t) \tag{14}$$

$$+fF_i(t) + eE_i(t)) + w\Delta X(t) \tag{15}$$

where $S_i(t), A_i(t), C_i(t), F_i(t)$ and $E_i(t)$ indicate the separation at the i^{th} iteration, alignment at the i^{th} iteration, cohesion at the i^{th} iteration, attraction at the i^{th} iteration, food source at the i^{th} iteration and position of the enemy at the i^{th} iteration, respectively. $\Delta X(t) = X(t) - X(t-1)$. The s, a, c, f, e and w are the weights of separation, alignment, cohesion, attraction, distraction and inertial weight, respectively. To balance each operation, these weights must be properly tuned for the environment.

III. D-SLP CONTROLLER DESIGN

A. D-SLP ALGORITHM

The D-SLP algorithm applies the operation of the DA to the learning process of the SLP algorithm to solve the distribution of information in the space. The algorithm consists of 3 main processing steps corresponding with the SLP algorithm: initialization, learning and selection. In initialization, each dragonfly is generated by stochastic processing. The learning of the proposed control method includes separation, alignment, cohesion, attraction and distraction operations. In this paper, separation, alignment, and cohesion operations adjust $u(t)$ in accordance with the system response, while the attraction and distraction operations are used to estimate the uncertainty term. For selection processing, the best dragonfly is selected based on the cost function ($J(t)$) of each dragonfly, as in Equation 16.

$$J(t) = \int_0^\infty te^2(t)dt \tag{16}$$

where t is the time and $e(t)$ is the error of the system that is calculated from the setpoint - controlled output.

In this paper, the separation, alignment, and cohesion operations adjust $u(t)$, $X(t) = \{x_1, x_2, x_3, \ldots, x_n\}$ is the position of the dragonfly, where n is the number of dragonflies, m is the number of x_i, where $(x_i - x_{i-1})^2 < \hat{D}(t)$ and where $i = 1, 2, 3, \ldots, n$, and $\hat{D}(t)$ is the nonlinear estimation term that is calculated by applying gradient descent as follows:

$$\hat{D}(t+1) = \hat{D}(t) - \eta(t)\frac{\partial J(t)}{\partial e(t)} \tag{17}$$

where $\eta(t)$ is the learning rate, which this paper adjusts according to the stable learning rate as in Equation 21, $e(t)$ is the error of the system and $J(t)$ is the cost function. Therefore, the control signal is adjusted as follows:

$$u(t) = e(t) \times argmin(X(t+1); J(t)) \tag{18}$$

where $X(t+1)$ equals Equation 13 when $m = 0$, and otherwise,

$$\sigma = [\frac{(1+\beta-1)! \times (sin\frac{1.5\pi}{2})}{(\frac{1+\beta}{2}-1)! \times \beta \times 2(\frac{\beta-1}{2})}]^{\frac{1}{\beta}} \tag{19}$$

$$X(t+1) = X(t) + (0.01 \times \frac{\lambda_1\sigma}{|\lambda_2|^{\frac{1}{\beta}}}) \tag{20}$$

where β is the constant (1.5 in this paper) and λ_1 and λ_2 are the random numbers in [0,1] [40].

Theorem 1: Let $z(t) = (x_w(t) + e(t) - \frac{\partial J(t)}{\partial e(t)})$, where $J(t)$ is the best cost function value and $x_w(t) = e_s(t)+e_a(t)+e_c(t)+e_f(t)+e_d(t)$, where $e_s(t)$, $e_a(t)$, $e_c(t)$, $e_f(t)$ and $e_d(t)$ are the errors caused by the separation process, alignment process, cohesion process, attraction process and distraction process, respectively. $e(t)$ is the error of the system. At time t, if the learning rate of estimating the term of nonlinearity ($\eta(t)$) is adjusted based on Equation 21, then the D-SLP controller for the nonlinear feedback system is stable.

$$0 \le \eta_i(t) \le \frac{2e(t)}{z(t)}; \quad z(t) \ne 0 \tag{21}$$

Proof: The Lyapunov function for confirming the stability of the learning factor of the D-SLP algorithm is considered to be caused by the integral of time multiplied squared error (ITSE) of the system, which can be defined as follows:

$$V(t) = \int_0^\infty te^2(t)dt \tag{22}$$

The derivation of the Lyapunov function is written as

$$\Delta V(t) = V(t+1) - V(t) \tag{23}$$

$$= \int_0^\infty t(e^2(t+1) - e^2(t))dt \tag{24}$$

$$= \int_0^\infty t((e(t+1) + e(t))(e(t+1) - e(t)))dt \tag{25}$$

Given that $\Delta e = e(t+1) - e(t)$, $e(t+1) = \Delta e(t) - e(t)$.

$$= \int_0^\infty t((2e(t) + \Delta e(t))\Delta e(t))dt \tag{26}$$

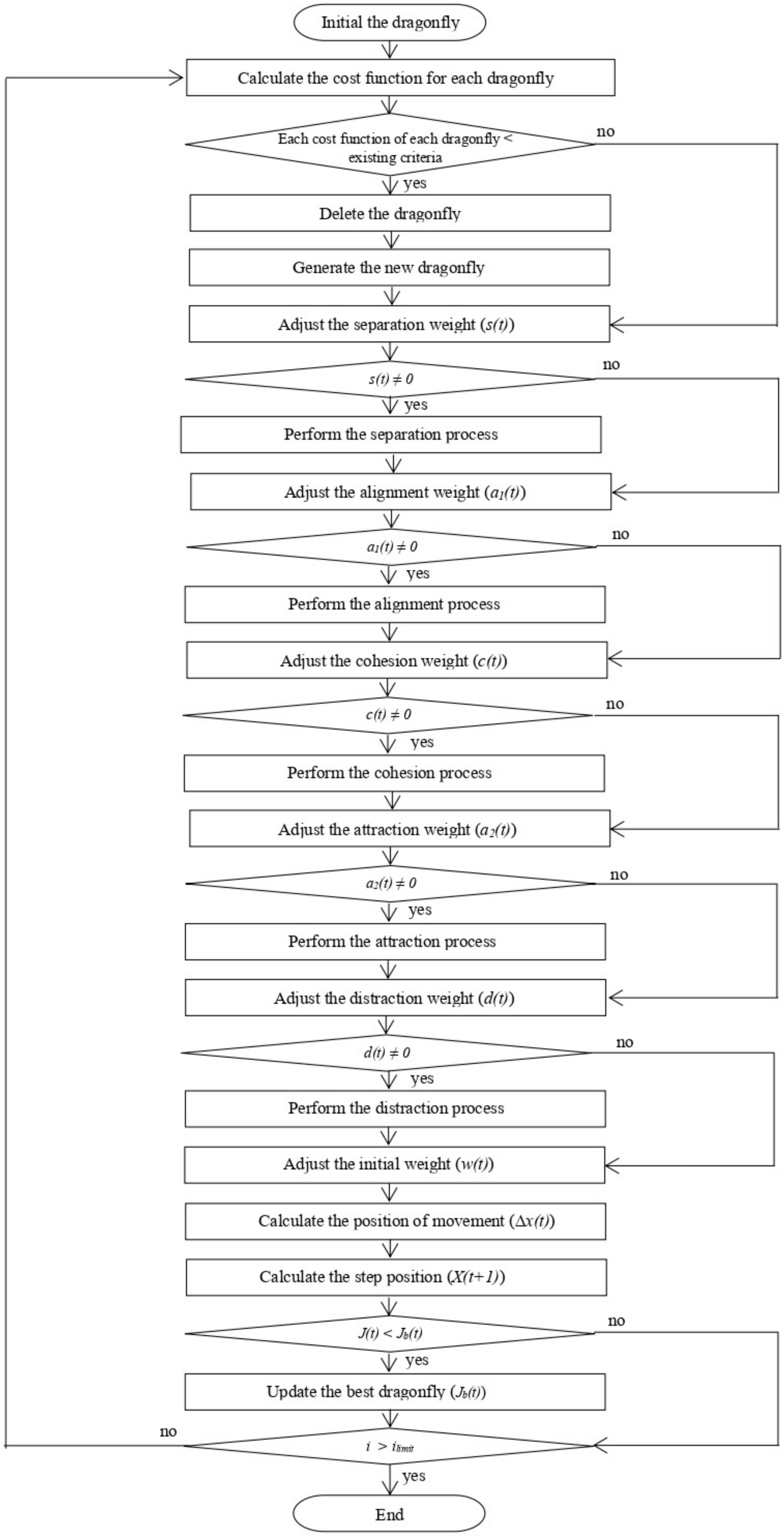

FIGURE 4. The flow chart of the D-SLP algorithm.

In this paper, the learning rate is adjusted for estimation of the nonlinear term of the D-SLP algorithm, which is related to updating the position of the dragonfly as in Equation 15. If the updating of the position is considered to be caused by the error, then $\Delta e(t) = s(t)e_s(t) + a(t)e_a(t) + c(t)e_c(t) + f(t)e_f(t) + d(t)e_d(t) - w(t)e(t+1)$, where $e_s(t)$ is the error from the separation process, $e_a(t)$ is the error from the alignment process, $e_c(t)$ is the error from the cohesion process, $e_f(t)$ is the error from the attraction process, $e_d(t)$ is the error from the distraction process and $s(t)$, $a(t)$, $c(t)$, $f(t)$, $d(t)$ and $w(t)$ are the weights of separation, alignment, cohesion, attraction, distraction and inertial weight, respectively. Letting $s(t) = a(t) = c(t) = f(t) = d(t) = w(t)$, $w(t) = \eta(t)$,

$$\Delta e(t) = \eta(t)(X_w(t) - e(t+1))dt \tag{27}$$

where $X_w(t) = e_s(t) + e_a(t) + e_c(t) + e_f(t) + e_d(t)$. From the gradient descent method error, $e(t+1) = e(t) - \lambda \bigtriangledown e(t)$, where λ is the learning rate, which is set to 1. Therefore, Equation 27 can be written as follows:

$$\Delta e(t) = \eta(t)y(t) \tag{28}$$

where $y(t) = X_w(t) - e(t) - \bigtriangledown e(t)$. When Equation 28 is replaced with Equation 26, the latter can be rearranged as follows:

$$\Delta V(t) = \int_0^\infty t((2e(t) + \eta(t)y(t))(\eta(t)y(t)))dt \tag{29}$$

$$= \int_0^\infty t((2e(t)\eta(t)y(t) + \eta^2(t)y^2(t)))dt \tag{30}$$

Considering the Lyapunov stability theorem at sampling time t, if $\Delta V(t) \leq 0$, the learning factor of the D-SLP algorithm is adjusted according to Equation 21, and the stability of the closed-loop control system by using the D-SLP algorithm is verified. ■

B. MARKOV CHAIN MODELING FOR THE D-SLP ALGORITHM

Definition 1 [43]: If X is a stochastic process in the discontinuous state space of finite state (S), then X is a finite Markov chain when $P(x_{n+1} = i_{n+1}|x_0 = i_0, x_1 = i_1, \ldots, x_n = i_n) = P(x_{n+1} = i_{n+1}|x_n = i_n)$ where for all n are arbitrary states I, $n \geq 0$ and $\{i_0, i_1, i_2, \ldots\} \in I$, P is a probabilistic limitation of x, and $X = \{x_n, n = 0, 1, 2, \ldots\}$.

Definition 2 [43]: If the condition probability $P(x_{n+1} = i_{n+1}|x_n = i_n)$ does not depend on time, it is a homogeneous Markov chain.

The model of the D-SLP algorithm consists of 9 states, i.e., the initial state ($D(0)$), evaluation state ($D_1(t)$), separation state ($D_2(t)$), alignment state ($D_3(t)$), cohesion state ($D_4(t)$), attraction state ($D_5(t)$), distraction state ($D_6(t)$), updating state ($D_7(t)$) and selection state ($D_8(t)$). In the initial state, the dragonfly is generated on the space number n. For state ($D_1(t)$), survival is evaluated based on the cost function, and a new dragonfly is generated when existing dragonflies in the space provide the cost function corresponding to survival criteria. The new dragonfly generation is caused by the best dragonfly in the space. Next, all dragonflies are transferred to the process of dragonfly operation, which consists of 6 operations. The first operation is separation ($D_2(t)$), and after that is a transfer to the alignment operation ($D_3(t)$). Next, the cohesion operation ($D_4(t)$) is performed, followed by the attraction operation ($D_5(t)$). Finally, distraction ($D_6(t)$) is performed. Note that between change state $D_i(t)$ and $D_{i+1}(t)$, where i is the state number, the weight of each state is determined. If the weight is 0, the state is skipped to reduce the execution time. After the dragonfly operation is performed, the updating of the position for each dragonfly ($D_7(t)$) is performed, and the best dragonfly is selected at state ($D_8(t)$). Then, state ($D_8(t)$) is changed to ($D(t+1)$).

The D-SLP algorithm changes the process at which weight adjustment of the conventional dragonfly algorithm occurs from after all operations finish to after the completion of each operation. Thus, the execution of each operation depends on the adjusting weight. The change in the transition state from $D_i(t)$ to $D_{i+1}(t)$, where $i \in 0, 2, 3, \ldots, 8$, is a random transition that is described as $|E^m| = |E|^m = M^m$, where m is not related with time. Thus, according to Definitions 1 and 2, the processing of the D-SLP algorithm exhibits a Markov chain property and represents a homogeneous Markov chain. The state transition matrix of the proposed control method in the case of learning processing can be denoted as $P = E \times S \times A \times C \times F \times D \times U$, where $E, S, A, C, F, D,$ and U are the positive matrices created by evaluation, separation, alignment, cohesion, attraction, distraction and updating, respectively.

In the case of the space consisting of $m+1$ dragonflies, the random transition is $|E^{m+1}| = |E|^{m+1} = M^{m+1}$. The state transition matrix based on $\widetilde{E}, \widetilde{S}, \widetilde{A}, \widetilde{C}, \widetilde{F}, \widetilde{D}, \widetilde{U}$ can be expressed as $\widetilde{P} = \widetilde{E} \times \widetilde{S} \times \widetilde{A} \times \widetilde{C} \times \widetilde{F} \times \widetilde{D} \times \widetilde{U}$, where

$$\widetilde{\mathbf{E}} = \begin{pmatrix} E & (*)^T & \ldots & (*)^T \\ 0 & E & \ldots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & E \end{pmatrix},$$

$$\widetilde{\mathbf{S}} = \begin{pmatrix} S & (*)^T & \ldots & (*)^T \\ 0 & S & \ldots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & S \end{pmatrix},$$

$$\widetilde{\mathbf{A}} = \begin{pmatrix} A & (*)^T & \ldots & (*)^T \\ 0 & A & \ldots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & A \end{pmatrix}$$

$$\widetilde{\mathbf{C}} = \begin{pmatrix} C & (*)^T & \ldots & (*)^T \\ 0 & C & \ldots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & C \end{pmatrix},$$

$$\widetilde{\mathbf{F}} = \begin{pmatrix} F & (*)^T & \dots & (*)^T \\ 0 & F & \dots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & F \end{pmatrix},$$

$$\widetilde{\mathbf{D}} = \begin{pmatrix} D & (*)^T & \dots & (*)^T \\ 0 & D & \dots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & D \end{pmatrix}$$

The updating state ($\widetilde{U}$) involves updating the position of the dragonfly and is a positive number. The state transition for this state can be expressed as follows:

$$\widetilde{\mathbf{U}} = \begin{pmatrix} U_{11} & 0 & \dots & 0 \\ U_{21} & U_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ U_{x1} & U_{x2} & \dots & U_{xy} \end{pmatrix}$$

where U_{11} is a unit matrix, $U_{xy} \neq 0$ and $x \leq y$. The state transition matrix of the D-SLP algorithm based on the Markov chain can be defined as follows:

$$\widetilde{\mathbf{X}} = \begin{pmatrix} P & (*)^T & \dots & (*)^T \\ 0 & P & \dots & (*)^T \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & P \end{pmatrix} \begin{pmatrix} U_{11} & 0 & \dots & 0 \\ U_{21} & U_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ U_{x1} & U_{x2} & \dots & U_{xy} \end{pmatrix}$$
$$= \begin{pmatrix} PU_{11} & 0 & \dots & 0 \\ PU_{21} & PU_{22} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ PU_{x1} & PU_{x2} & \dots & PU_{xy} \end{pmatrix}$$

when $X = PU_{11}$,

$$\mathbf{Y} = \begin{pmatrix} SU_{21} \\ \vdots \\ PS_{x1} \end{pmatrix}, \quad \mathbf{Z} = \begin{pmatrix} PU_{22} & \dots & 0 \\ \vdots & \ddots & \vdots \\ PU_{x2} & \dots & PU_{xy} \end{pmatrix}$$

Thus, the state transition matrix of the D-SLP algorithm based on the Markov chain can be expressed as follows:

$$\widetilde{\mathbf{X}} = \begin{pmatrix} X & 0 \\ Y & Z \end{pmatrix}$$

C. CONVERGENCE ANALYSIS FOR THE D-SLP ALGORITHM

Definition 3 ([43], [44]): The algorithm I achieves global convergence when it achieves the condition as follows:

1. For the iteration result from algorithm I (ζ), if $\zeta \in Z$, then $f(I(x,\zeta)) \leq f(x)$
2. If $B \in Z$, then $\prod_{k=0}^{\infty}(1 - u_k(B)) = 0$, where B is a set of Lebesgue measurables for which $(v(B)) > 0$ and $u_k(B)$ is the size of probabilistic algorithm I in the set of B at the k^{th} iteration.
3. $\lim_{i\to\infty} P[x^i \in R_{e,m}] = 1$, where $R_{e,m}$ is an optimal region, x^i is a generating order of algorithm I and $i \in \{1, \dots, \infty\}$ is the iteration.

Lemma 1 [45]: Letting X and Z be the square matrix, $\widetilde{X}$ achieves the properties of finite Markov chain and homogeneous Markov chain. If the format $\widetilde{X} = \begin{pmatrix} X & 0 \\ Y & Z \end{pmatrix}$, X^k is a unique stable matrix of X^{∞}, i.e., $X^k \rightarrow X^{\infty} = (\pi^T, \pi^T, \dots, \pi^T)$, where $\pi^T = (x_1, x_2, \dots, x_m, 0, \dots, 0)$, $x_i > 0$ (i = 1,..., m), where m is the size of matrix X, and x^{∞} is unique irrespective of the initial condition, $x^k = \lim_{t\to\infty} x^0\widetilde{X}^t = (x_1^{\infty}, x_2^{\infty}, \dots, x_m^{\infty}, 0, \dots, 0)$, where $\sum_{i=1}^{m} x_i^{\infty} = 1$.

Theorem 2: The D-SLP algorithm achieves condition 1 *of Definition 3.*

Proof: The process of the D-SLP algorithm consists of the selecting process, which selects the best dragonfly of the D-SLP algorithm based on the cost function of each dragonfly in the space. The best dragonfly ($f(X)$) is updated to be ($f(x_i)$) when $f(X) \geq f(x_i)$, where $f(x_i)$ is the cost function at the i^{th} iteration. Thus, the D-SLP algorithm achieves condition 1 of Definition 3. ■

Theorem 3: The D-SLP algorithm achieves condition 2 *of Definition 3.*

Proof: According to the D-SLP algorithm modeling based on the Markov chain model, the matrices of E, S, A, C, F, D and U are positive with $P > 0$, where P^0 is the dictatorial distribution, which is limited according to Lemma 1 and P^{∞} is as usual. In the case that the probability of the D-SLP algorithm is in the global state, the probability of D-SLP $= 0$. Thus, $\prod_{i=0}^{\infty}(1 - u_i(B)) = 0$, where $u_i(B)$ is the probabilistic state of the algorithm in the i^{th} iteration in set B. Therefore, condition 2 in Definition 3 is achieved. ■

Theorem 4: The D-SLP algorithm achieves condition 3 *of Definition 3 and the property of global convergence.*

Proof: The process of the D-SLP algorithm updates the best dragonfly $f(X)$ to be $f(x_i)$ when $f(X) \geq f(x_i)$, and then, the other dragonflies are maintained for becoming the best dragonfly, assuming that the best dragonfly is in the optimal region ($R_{e,m}$). This means that when the i^{th} iteration $R_{e,m}$ approaches infinity, the $P(x_i) \in R_{e,m}$, or, in other words, $\lim_{i\to\infty} P(x_i \in R_{e,m}) = 1$. Therefore, it can be concluded that the D-SLP algorithm achieves condition 3 of Definition 3.

From Theorems 2-4, the D-SLP algorithm can satisfy the 3 conditions of Definition 3, and it can be determined that the D-SLP algorithm achieves the property of global convergence. ■

IV. ILLUSTRATIVE EXAMPLE AND DISCUSSION

To illustrate the effectiveness of designing a controller by using the D-SLP algorithm, the tracking error of the PMSM control system is compared with that of the PID controller and LQR. The comparison is performed by comparing the variation of the response of the tracking input for each case study. The PID parameter is designed by parametric autotuning based on the conventional SLP algorithm and conventional DA algorithm. The LQR controller adjusts Q and R based on ABC-FPA [41]. The input signal for verification consists of 2 patterns, here, the step input and sinusoidal signal. In the simulation for both case studies, the motor is started from 0 rpm and the time of simulation is 6 s. The mathematical model of the PMSM system is performed corresponding to reference [46]. The parameters of PMSM are adjusted according to reference [47].

For the first case study, the simulation and comparison of tracking error of the designed PMSM control system are performed by setting the input signal to be 1000 rpm. The iteration number for each method of adjusting the parameter is 30 iterations. The simulation result of each algorithm is shown in Figures 5 to 8. From the simulation results, the variation of control is 95.13 by using the PID controller with the SLP algorithm, 110.27 for the PID controller with the DA, 60.12 for LQR with ABC-FPA [41] and 40.60 for the proposed control method.

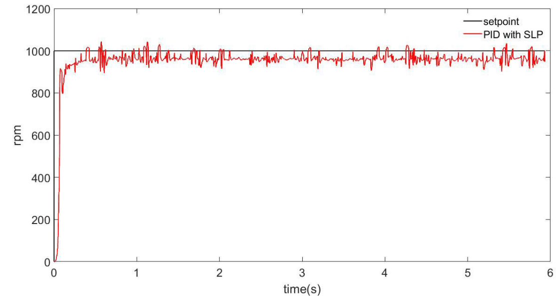

FIGURE 5. Output of tracking step input based on a PID controller by using the SLP algorithm.

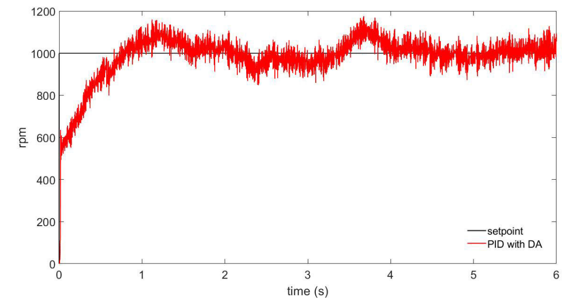

FIGURE 6. Output of tracking step input based on a PID controller by using the DA.

In the second case study, the sinusoidal signal is utilized as the input for simulation and comparison of tracking error to design the PMSM control system. During the simulation, the minimum reference input is −500 rpm, with the maximum input of 500 rpm. From the simulation results of Figures 9 to 12, the variation of control is 457.07 by using the PID controller with the SLP algorithm, 264.05 for the PID controller with the DA, 170.98 for LQR with ABC-FPA [41] and 147.72 for the proposed control method.

From the simulation of the 2 case studies, the proposed control method can provide smoother tracking error than the other designed controller because the proposed control method designs the controller by estimating the nonlinear term and applying the estimation to the online design controller, while the PID controller by using the SLP, DA and LQR with ABC-FPA [41] only includes online adjustment of the parametric controller. Although online adjustment is utilized, adjustment is considered to be caused by the previous response. Thus, some values of adjustment may be proper for the current characteristics of a system. In the case of a highly

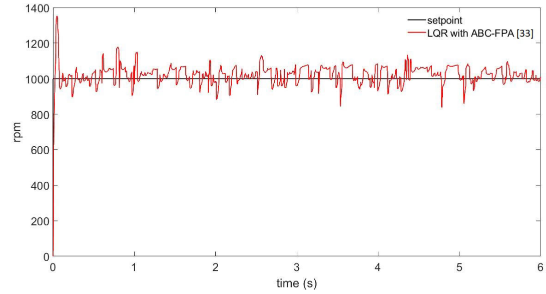

FIGURE 7. Output of tracking step input based on an LQR controller by using ABC-FPA.

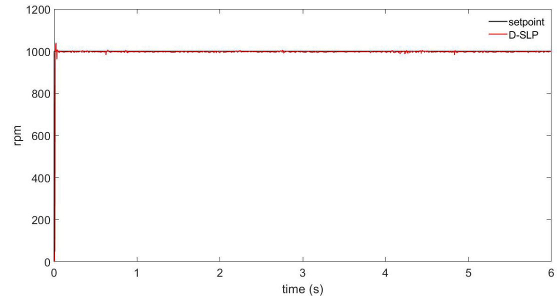

FIGURE 8. Output of tracking step input based on the D-SLP algorithm.

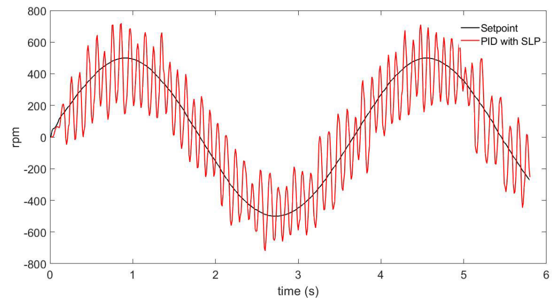

FIGURE 9. Output of tracking sinusoidal input based on a PID controller by using the SLP algorithm.

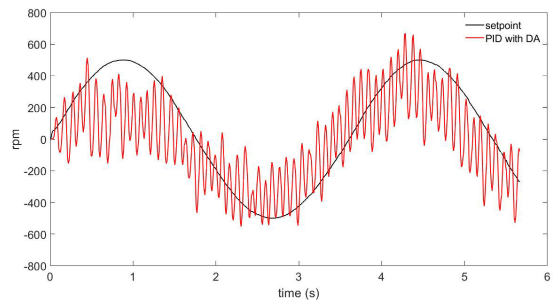

FIGURE 10. Output of tracking sinusoidal input based on a PID controller by using the DA.

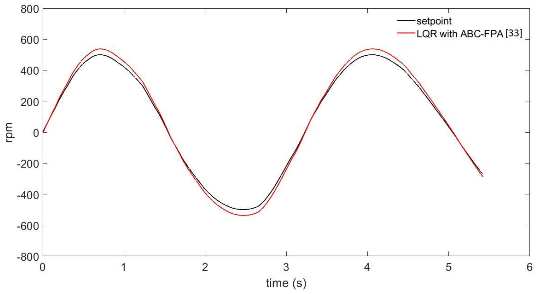

FIGURE 11. Output of tracking sinusoidal input based on an LQR controller by using ABC-FPA.

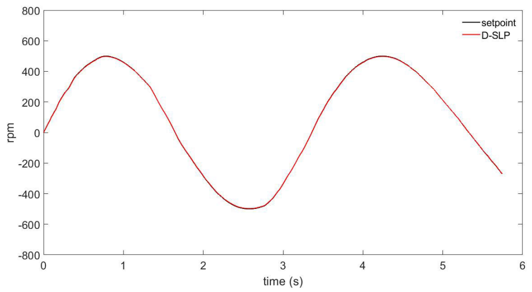

FIGURE 12. Output of tracking sinusoidal input based on the D-SLP algorithm.

dynamic system, the value after adjustment can cause system instability and can require time for adjustment because of the changing characteristics of the system.

Remark 1: From the simulation result, the proposed method can control the nonlinear system and provide the better performance when compared with other designed approach such as the SLP algorithm, the DA and the LQR controller based on the ABC-FPA. Furthermore, the performance comparison of the tracking reference input is also performed for verifying the robustness of designing controller. The proposed method can show the better results when compared with the other designed methods due to the fact that the proposed method estimates the nonlinear term by online adjusting parameters, while the SLP algorithm, the DA and the LQR controller based on the ABC-FPA estimate the parameters based on the previous response. In addition, for the high dynamic system, the difference value between the previous value and the current value is normally high. Those methods may take time to adjust the parameter for properly obtaining the current characteristic of the system or sometimes may not obtain the adjusted parameter since the characteristic of system is always changed.

V. CONCLUSION

In this brief, the D-SLP controller is proposed for designing a nonlinear feedback control system. The controller design consists of estimating the nonlinear term of the system and adjusting the control signal. The process of estimation applies the stable gradient descent method, and design of the control system is considered by using the D-SLP algorithm. The rule of adjusting the learning rate of gradient descent is derived in the sense of the Lyapunov stability function so that the stability of estimating the nonlinear term of the system can be verified. Because the D-SLP algorithm is applied for stochastic processing, the global convergence is also verified based on the convergence rule of the stochastic search algorithm. Additionally, to show the superior performance and robustness of controlling the nonlinear control system, the comparison of simulation results based on the PMSM control system is performed. In the comparison, the proposed control method is compared with the simulation with PID and LQR controllers, for which the parameters of both methods are adjusted online. Furthermore, 2 case studies are performed based on the pattern of input references, namely, step and sinusoidal. According to the simulation results, the proposed control method can provide better performance and robustness than the PID and LQR controller when parameters are adjusted online for both case studies.

Therefore, the simulation study can evaluate whether the theory that is proposed in this paper can claim better performance and robustness with respect to designing the nonlinear control system for practical applications. However, in this paper, we only consider the D-SLP for a general nonlinear control system, which may limit the applications of the designed approach. Therefore, the future work is possible to focus on the D-SLP controller with the unidirectional input constraints and the input dead zones which may exist in some recent research applications such as the pneumatic artificial muscle (PAM) systems [48], and the indispensable oceanic transportation [49].

ACKNOWLEDGMENT

The authors would like to thank the Department of Electronic and Telecommunication Engineering, Faculty of Engineering, King Mongkut's University of Technology Thonburi for their support.

REFERENCES

[1] C. Wu, J. Liu, Y. Xiong, and L. Wu, "Observer-based adaptive fault-tolerant tracking control of nonlinear nonstrict-feedback systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 7, pp. 3022–3033, Jul. 2018.

[2] Y. Zhang, S. Li, and X. Liu, "Neural network-based model-free adaptive near-optimal tracking control for a class of nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6227–6241, Dec. 2018.

[3] D. Wang, D. Liu, Q. Zhang, and D. Zhao, "Data-based adaptive critic designs for nonlinear robust optimal control with uncertain dynamics," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 11, pp. 1544–1555, Nov. 2016.

[4] W. Chen, S. Hua, and H. Zhang, "Consensus-based distributed cooperative learning from closed-loop neural control systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 331–345, Feb. 2015.

[5] N. Chayaopas and W. Assawinchaichote, "A novel approach to robust H_∞ integral control for TS fuzzy systems," *Comput. Appl. Math.*, vol. 37, pp. 954–977, Aug. 2018.

[6] 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.

[7] S. Ruangsang and W. Assawinchaichote, "A novel robust H_∞ fuzzy state feedback plus state-derivative feedback controller design for nonlinear time-varying delay systems," *Neural Comput. Appl.*, vol. 31, no. 10, pp. 6303–6318, Oct. 2019.

[8] X.-H. Chang and G.-H. Yang, "Nonfragile H_∞ filtering of continuous-time fuzzy systems," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1528–1538, Apr. 2011.

[9] W. Assawinchaichote, "A non-fragile H_∞ output feedback controller for uncertain fuzzy dynamical systems with multiple time-scales," *Int. J. Comput. Commun. Control*, vol. 7, no. 1, pp. 8–9, 2012.

[10] S. K. 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.*, 2005, pp. 1799–1804.

[11] K. N. Sing and W. Assawinchaichote, "Fuzzy observer-based controller design for singularly perturbed nonlinear systems: An LMI approach," in *Proc. IEEE Conf. Decis. Control*, Dec. 2002, vol. 26, no. 6, pp. 2165–2170.

[12] N. Kaewpraek and W. Assawinchaichote, "H_∞ fuzzy state-feedback control plus state-derivative-feedback control synthesis for photovoltaic systems," *Asian J. Control*, vol. 18, no. 4, pp. 1441–1452, Jul. 2016.

[13] C. Vorrawan and W. Assawinchaichote, "H_∞ fuzzy integral controller for nonlinear descriptor systems," *Int. J. Innov. Comput., Inf. Control*, vol. 15, no. 2, pp. 741–756, 2019.

[14] T. Ma, L. Li, and H. Li, "Observer-based finite-time adaptive sliding mode control for itô stochastic jump systems with actuator degradation," *IEEE Access*, vol. 8, pp. 18590–18600, 2020.

[15] E. Lu, W. Li, X. Yang, and S. Xu, "Composite sliding mode control of a permanent magnet direct-driven system for a mining scraper conveyor," *IEEE Access*, vol. 5, pp. 22399–22408, 2017.

[16] Y. Wang, Y. Feng, X. Zhang, and J. Liang, "A new reaching law for antidisturbance sliding-mode control of PMSM speed regulation system," *IEEE Trans. Power Electron.*, vol. 35, no. 4, pp. 4117–4126, Apr. 2020.

[17] B. Zhou, S. Xie, and J. Hui, "H_∞ control for T-S aero-engine wireless networked system with scheduling," *IEEE Access*, vol. 7, pp. 115662–115672, 2019.

[18] L. Liu, B. Wang, S. Wang, Y. Chen, T. Hayat, and F. E. Alsaadi, "Finite-time H_∞ control of a fractional-order hydraulic turbine governing system," *IEEE Access*, vol. 6, pp. 57507–57517, 2018.

[19] S. P. Ribas, L. A. Maccari, V. F. Montagner, R. C. de Leão Fontoura de Oliveira, and H. Pinheiro, "Design and implementation of a discrete-time H-infinity controller for uninterruptible power supply systems," *IET Power Electron.*, vol. 7, no. 9, pp. 2233–2241, Sep. 2014.

[20] E. Okyere, A. Bousbaine, G. T. Poyi, A. K. Joseph, and J. M. Andrade, "LQR controller design for quad-rotor helicopters," *J. Eng.*, vol. 2019, no. 17, pp. 4003–4007, Jun. 2019.

[21] M. Zhang, X. Li, J. Liu, and H. Su, "Dual-mode LQR-feedforward optimal control for non-minimum phase boost converter," *IET Power Electron.*, vol. 10, no. 1, pp. 92–102, Jan. 2017.

[22] D. Sato, Y. Chen, K. Miyamoto, and J. She, "A spectrum for estimating the maximum control force for passive-base-isolated buildings with LQR control," *Eng. Struct.*, vol. 199, no. 15, pp. 1–11, 2019.

[23] Y.-J. Liu and S. Tong, "Adaptive fuzzy identification and control for a class of nonlinear pure-feedback MIMO systems with unknown dead zones," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 5, pp. 1387–1398, Oct. 2015.

[24] S. Tong, Y. Li, and S. Sui, "Adaptive fuzzy tracking control design for SISO uncertain nonstrict feedback nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 6, pp. 1441–1454, Dec. 2016.

[25] Y. Chang, Y. Wang, F. E. Alsaadi, and G. Zong, "Adaptive fuzzy output-feedback tracking control for switched stochastic pure-feedback nonlinear systems," *Int. J. Adapt. Control Signal Process.*, vol. 33, no. 10, pp. 1567–1582, 2019.

[26] X.-H. Chang, J. Xiong, and J. H. Park, "Estimation for a class of parameter-controlled tunnel diode circuits," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Aug. 13, 2019, doi: 10.1109/TSMC.2018.2859933.

[27] L. Ma, G. Zong, X. Zhao, and X. Huo, "Observed-based adaptive finite-time tracking control for a class of nonstrict-feedback nonlinear systems with input saturation," *J. Franklin Inst.*, pp. 1–27, Sep. 2019.

[28] W. Wang and S. Tong, "Observer-based adaptive fuzzy containment control for multiple uncertain nonlinear systems," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 11, pp. 2079–2089, Nov. 2019.

[29] H. Xu and S. Jagannathan, "Neural network-based finite horizon stochastic optimal control design for nonlinear networked control systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 472–485, Mar. 2015.

[30] H. Wang, P. Xiaoping Liu, S. Li, and D. Wang, "Adaptive neural output-feedback control for a class of nonlower triangular nonlinear systems with unmodeled dynamics," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 8, pp. 3658–3668, Aug. 2018.

[31] W. Meng, Q. Yang, and Y. Sun, "Adaptive neural control of nonlinear MIMO systems with time-varying output constraints," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 1074–1085, May 2015.

[32] C. L. P. Chen, G.-X. Wen, Y.-J. Liu, and Z. Liu, "Observer-based adaptive backstepping consensus tracking control for high-order nonlinear semi-strict-feedback multiagent systems," *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1591–1601, Jul. 2016.

[33] O. García, P. Ordaz, O. Santos-Sánchez, S. Salazar, and R. Lozano, "Backstepping and robust control for a quadrotor in outdoors environments: An experimental approach," *IEEE Access*, vol. 7, pp. 40636–40648, 2019.

[34] A. Mozaffari, N. L. Azad, and J. K. Hedrick, "A nonlinear model predictive controller with multiagent online optimizer for automotive cold-start hydrocarbon emission reduction," *IEEE Trans. Veh. Technol.*, vol. 65, no. 6, pp. 4548–4563, Mar. 2016.

[35] L. Yan and X. Song, "Design and implementation of Luenberger model-based predictive torque control of induction machine for robustness improvement," *IEEE Trans. Power Electron.*, vol. 35, no. 3, pp. 2257–2262, Mar. 2020.

[36] L. Song and S. Jingzhuo, "Novel generalized predictive iterative learning speed controller for ultrasonic motors," *IEEE Access*, vol. 8, pp. 29344–29353, 2020.

[37] C.-H. Chou, "Genetic algorithm-based optimal fuzzy controller design in the linguistic space," *IEEE Trans. Fuzzy Syst.*, vol. 14, no. 3, pp. 372–385, Jun. 2006.

[38] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 10, pp. 2243–2254, Oct. 2017.

[39] 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.

[40] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, May 2016.

[41] T. Tarczewski and L. M. Grzesiak, "An application of novel nature-inspired optimization algorithms to auto-tuning state feedback speed controller for PMSM," *IEEE Trans. Ind. Appl.*, vol. 54, no. 3, pp. 2913–2925, May 2018.

[42] D. Liu, D. Wang, F.-Y. Wang, H. Li, and X. Yang, "Neural-Network-Based online HJB solution for optimal robust guaranteed cost control of continuous-time uncertain nonlinear systems," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2834–2847, Dec. 2014.

[43] D. Wu, S. Xu, and F. Kong, "Convergence analysis and improvement of the chicken swarm optimization algorithm," *IEEE Access*, vol. 4, pp. 9400–9412, 2016.

[44] F. J. Solis and R. J.-B. Wets, "Minimization by random search techniques," *Math. Oper. Res.*, vol. 6, no. 1, pp. 19–30, Feb. 1981.

[45] L. Ming, Y. Wang, and Y.-M. Cheung, "On convergence rate of a class of genetic algorithms," in *Proc. World Autom. Congr.*, Budapest, Hungary, Jul. 2006, pp. 1–6.

[46] W.-J. Xu, "Permanent magnet synchronous motor with linear quadratic speed controller," *Energy Procedia*, vol. 14, pp. 364–369, Jan. 2012.

[47] J. Yu, P. Shi, W. Dong, B. Chen, and C. Lin, "Neural network-based adaptive dynamic surface control for permanent magnet synchronous motors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 640–645, Mar. 2015.

[48] N. Sun, D. Liang, Y. Wu, Y. Chen, Y. Qin, and Y. Fang, "Adaptive control for pneumatic artificial muscle systems with parametric uncertainties and unidirectional input constraints," *IEEE Trans. Ind. Informat.*, vol. 16, no. 2, pp. 969–979, Feb. 2020.

[49] T. Yang, N. Sun, H. Chen, and Y. Fang, "Neural network-based adaptive antiswing control of an underactuated ship-mounted crane with roll motions and input dead zones," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 3, pp. 901–914, Mar. 2020.

JIRAPUN PONGFAI received the Bachelor of Engineering degree in computer engineering from Naresuan University, Pitsanulok, Thailand, in 2015, and the Master of Engineering degree in electrical and information engineering from the King Mongkut's University of Technology Thonburi, Bangkok, Thailand, in 2017. Her research interests include intelligence control, optimal and predictive controller design, artificial intelligence, and embedded system design.

WUDHICHAI ASSAWINCHAICHOTE (Member, IEEE) 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 articles 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 serves as a Reviewer for the IEEE Transactions on Industrial Electronics, the IEEE Transactions on Fuzzy Systems, the IEEE Transactions on Cybernetics, *Neural Computing and Applications*, and IEEE Access. He currently serves as an Associate Editor for the *International Journal of Innovative Computing, Information and Control*.

PENG SHI (Fellow, IEEE) received the Ph.D. degree in electrical engineering from the University of Newcastle, Australia, in 1994, the Doctor of Science degree from the University of Glamorgan, Wales, in 2006, and the Doctor of Engineering degree from The University of Adelaide, Australia, in 2015. He is currently a Professor with The University of Adelaide. His research interests include system and control theory, intelligent systems, and operational research. He has published widely in these areas. He is a Member-at-Large of Board of Governors, IEEE SMC Society, and was an IEEE Distinguished Lecturer. He is a Fellow of the Institution of Engineering and Technology; and the Institute of Engineers, Australia. He received the Andrew Sage Best Transactions Paper Award from IEEE SMC Society, in 2016, and the Lofti Zaden Best Paper Award from International Conference of Machine Learning and Cybernetics, in 2019. He has served on the Editorial Board for a number of journals, including *Automatica*, the IEEE Transactions on Automatic Control, the IEEE Transactions on Cybernetics, the IEEE Transactions on Fuzzy Systems, the IEEE Transactions on Circuits and Systems, IEEE Access, the IEEE Control Systems Letters, *Information Sciences*, and *Signal Processing*.

XIAOJIE SU (Senior Member, IEEE) received the Ph.D. degree in control theory and control engineering from the Harbin Institute of Technology, China, in 2013. He is currently a Professor and the Associate Dean of the College of Automation, Chongqing University, Chongqing, China. He has published two research monographs and more than 50 research articles in international refereed journals. His current research interests include intelligent control systems, advanced control, and unmanned system control. He currently serves as an Associate Editor for a number of journals, including the IEEE Systems Journal, the IEEE Control Systems Letters, *Information Sciences*, and *Signal Processing*. He is also an Associate Editor of the Conference Editorial Board and the IEEE Control Systems Society. He was named to the 2017, 2018, and 2019 Highly Cited Researchers list from Clarivate Analytics.