IET Control Theory & Applications

WILEY

IET The Institution of Engineering and Technology

ORIGINAL RESEARCH OPEN ACCESS

Dynamic Enzyme-Inspired Nonlinear Optimisation of Feedforward-Compensated PI Controllers for Robust Speed Regulation in Complex DC Motor Systems

Gökhan Yüksek[1] | Serdar Ekinci[2] | Davut Izci[3,4] | Timur Lale[1] | Rıdvan Fırat Çınar[5] | Burcu Bektaş Güneş[6] | Mohit Bajaj[7,8,9] | Vojtech Blazek[10] | Olena Rubanenko[11]

[1]Department of Electrical and Electronics Engineering, Batman University, Batman, Turkey | [2]Department of Computer Engineering, Bitlis Eren University, Bitlis, Turkey | [3]Department of Electrical and Electronics Engineering, Bursa Uludag University, Bursa, Turkey | [4]Applied Science Research Center, Applied Science Private University, Amman, Jordan | [5]Department of Computer Engineering, Batman University, Batman, Turkey | [6]Department of Computer Engineering, Istanbul Gedik University, İstanbul, Turkey | [7]Department of Electrical Engineering, Graphic Era (Deemed to Be University), Dehradun, India | [8]Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman, Jordan | [9]Centre For Research Impact and Outcome, Chitkara University, Rajpura, India | [10]ENET Centre, CEET, VSB-Technical University of Ostrava, Ostrava, Czech Republic | [11]Department of Power Plants and System, Vinnytsia National Technical University, Vinnytsia, Ukraine

Correspondence: Mohit Bajaj (mohitbajaj.ee@geu.ac.in) | Olena Rubanenko (olenarubanenko@vntu.edu.ua)

Received: 29 November 2025 | **Revised:** 2 June 2026 | **Accepted:** 25 June 2026

ABSTRACT

Speed control of direct current (DC) motor systems still remains a challenging nonlinear control problem due to load disturbances, parameter uncertainties, and time-varying operating conditions. Although conventional proportional integral derivative (PID) controllers are widely used because of their simple structure, their fixed-gain configuration limits performance under uncertain and dynamic conditions. To address these limitations, a feedforward-compensated proportional integral (PI + FF) control framework optimised using the dynamic enzyme action optimiser (dEAO) is proposed. The proposed dEAO employs an adaptive bio-inspired search mechanism to improve the exploration–exploitation balance and enhance parameter convergence during controller tuning. In addition, the simultaneous optimisation of PI and feedforward parameters improves transient response, disturbance rejection capability, and steady-state performance. The effectiveness of the proposed framework is validated through extensive numerical simulations and real-time experimental studies on a laboratory-scale DC motor platform. Experimental results confirm that the dEAO-based PI + FF controller achieves superior dynamic performance with a rise time of 1.20 s, settling time of 1.90 s, overshoot of 1.0%, and a negligible steady-state error (8.7×10^{-8}%). The Wilcoxon signed-rank test and statistical analyses over 25 independent runs reveal that the proposed method provides superior consistency and robustness compared to conventional rules and existing state-of-the-art optimisers such as EAO, NRBO, L-SHADE and GA. In addition, the proposed controller maintains stable operation of the DC motor under measurement noise, load disturbances and time-varying references, which demonstrates strong robustness and practical applicability for DC motor control systems.

1 | Introduction

Due to their simple construction and excellent dynamic performance, DC motors are widely adopted as actuators in industrial and mechatronic systems. Therefore, these devices can be considered good candidates for actuation in high-precision and fast-response applications such as robotic manipulators, electric vehicles, automated production lines and aerospace subsystems

This is an open access article under the terms of the Creative Commons Attribution-NonCommercial-NoDerivs License, which permits use and distribution in any medium, provided the original work is properly cited, the use is non-commercial and no modifications or adaptations are made.
© 2026 The Author(s). *IET Control Theory & Applications* published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology.

[1, 2]. For many of these systems, it is critical to maintain a stable rotational speed under time-varying load conditions to achieve task precision, system efficiency and control stability. But in real applications, DC motors are exposed to external disturbances, variations in mechanical loads and system nonlinearities. These factors make the speed control problem more difficult under practical operating conditions [3–5].

In order to overcome these difficulties, conventional control strategies (e.g., proportional-integral (PI) and proportional-integral-derivative (PID) controllers), have found wide industrial applications owing to their simplicity of implementation and adequate steady-state results [6–8]. They can remove static errors and perform adequately under nominal conditions and this is why they are widely used in commercial DC drive systems. However, their feedback-only architecture and fixed-gain nature limit their ability to respond to sudden reference changes, time-varying system dynamics and external disturbances. Therefore, under non-ideal conditions, their performance degrades, particularly in applications with stringent requirements for fast transient response, small overshoot and strong disturbance rejection [9, 10].

Among many structures in PI control attempting to address the shortcomings of the standard PI controller, the addition of a feedforward compensation path has turned out to be an effective solution for system response enhancement. Since the feedforward branch operates on the reference signal itself, it represents a form of anticipatory control, allowing the control system to compensate for known disturbances or setpoint changes in advance of their passing through the feedback loop. This dual-path design compensates for the phase lag inherent in feedback-only schemes and improves transient response and disturbance rejection performance of the closed-loop system [11]. However, the complexity of accurately tuning the related parameters is still very high for feedforward-compensated PI (PI + FF) controllers, and even more so for nonlinear/time-varying systems. Hence, in order to achieve further advances in performance, it is necessary not only to consider structural improvement but also intelligent parameter optimisation algorithms that are adaptive to the system dynamics [12].

The introduction of a feedforward path to the classic PI controller has recently gained significant interest as an effective means of enhancing control performance in systems with measurable disturbances or fast changing references. In DC motor applications, the PI + FF form has been found to offer better transient response, less overshoot and superior tracking accuracy as it pre-shapes the control input according to the reference trajectory. While the PI loop guarantees zero steady-state error, the feedforward branch offers predictive control action. However, the improved controller structure brings a couple more parameters, usually a feedforward gain and time constant, which have to be tuned together with the PI gains. As the parameters are nonlinearly coupled and system dependent, manual tuning is time consuming and suboptimal, especially in the presence of time-varying or uncertain operating conditions. This enduring shortcoming calls for automated multi-parametric optimisation-based tuning schemes, able to tune all parameters at once to meet competing performance criteria [13].

Several studies have focused on improving DC motor speed control based on PI-type controllers with parameters optimised by advanced metaheuristic algorithms. For example, Saravanan et al. implemented new kookaburra and red panda optimisation algorithms to adjust PID controllers for DC motor speed control, resulting in up to 12% quicker rise time and high robustness to parametric uncertainty while employing ITAE as the objective function [14]. Jabari et al. introduced a multistage FOPD(1 + PI) controller based on the pelican optimisation algorithm. In comparison with conventional metaheuristic-based tuning of PID and FOPID controllers, this method resulted in a 28% decrease in rise time, 35% reduction in settling time and 22% reduction in maximum peak overshoot [13]. In the same manner, Çelik et al. presented an exponential nonlinear PI controller for PMDC motors, tuned offline with the salp swarm algorithm. The developed controller ensured robust tracking and torque disturbance rejection on DSP hardware [15]. Chen introduced a feedforward-compensated PID controller to regulate the speed of a DC motor. The control strategy proposed in this paper markedly enhanced dynamic performance when the system was subjected to load disturbance, in particular, alleviating overshoot, reducing settling time and improving steady-state performance over the traditional PID controller [16]. Finally, an investigation using the whale optimisation algorithm, PSO and ant colony optimisation for tuning PID controllers for DC motor systems showed consistent enhancement in minimising steady-state error and improving disturbance robustness under different load and reference conditions [17].

Robustness-oriented PI/PID controller design has recently attracted increasing attention in order to improve stability and disturbance rejection in uncertain dynamical systems [18]. Mao et al. demonstrated that gain and phase margins are meaningful robustness measures of PID-controlled systems and that a suitably designed PID structure can lead to good robustness with respect to modelling errors and external disturbances. Their results further suggested that robustness-oriented tuning based on gain and phase margins has the potential to enhance closed-loop stability properties [19]. Meanwhile, additional PID-based schemes related to filtering and implementation have been studied for practical robustness and reduction of sensitivity to measurement noise and external disturbances [20, 21]. However, the majority of the existing results are still based on traditional feedback-loop tuning methods, and the optimal tuning of feedforward-compensated PI structures by using bio-inspired search algorithms has been seldom studied in the control of DC motors.

In order to tackle these issues, a PI + FF control scheme is put forward in this work based on the dynamic enzyme action optimiser (dEAO) [22]. Contrary to the classical approach to PI/PID tuning, which considers only the feedback-loop parameters, the present scheme performs joint optimisation of the PI parameters and those of the feedforward controller in a unified adaptive optimisation framework. The contributions of this study are summarised as follows:

- A feedforward-compensated PI control structure is developed to improve transient response, disturbance rejection capability and steady-state accuracy in DC motor speed regulation.

- A dEAO-based adaptive optimisation mechanism is introduced for the simultaneous tuning of PI and feedforward controller parameters.
- The proposed scheme is thoroughly investigated under reference tracking, disturbance rejection and noise injection scenarios via numerical simulations as well as real-time experimental studies.
- Results from comparisons with traditional tuning methods and state-of-the-art optimisation algorithms illustrate the improved robustness and dynamic control performance of the proposed approach.

2 | Proposed Dynamic Enzyme Action Optimiser

Based on enzyme-substrate interaction, dEAO is an enhanced form of the basic enzyme action optimiser (EAO) [22]. While EAO provided a reasonable trade-off between exploration and exploitation in solving a variety of optimisation problems, the algorithm is not free from limitations due to the fixed parameter settings and its limited fine-tuning capability in the late stage of the optimisation procedure. The proposed dEAO introduces three main improvements over its predecessor: a sinusoidal adaptation strategy for the adaptive factor (AF), the dynamic randomisation of the enzyme concentration (EC) and a lightweight local search module. All these combined modifications lead to greater flexibility, convergence accuracy as well as robustness of the optimiser, especially in tackling complex multimodal and high-dimensional problems.

2.1 | Initialisation Strategy

The search process begins with the random initialisation of a substrate pool, where each substrate is a candidate solution. This population is uniformly distributed across the problem space. The problem size (D) and the population size (N) are predefined. The procedure used to initialise a substrate is as follows:

$$X_i^{(0)} = LB_d + (UB_d - LB_d) \cdot r_{i,d} \tag{1}$$

where $X_i^{(0)}$ denotes the initial value of the d^{th} dimension of the i^{th} substrate, LB_d and UB_d are the lower and upper bounds for the d^{th} decision variable and $r_{i,d}$ is a uniformly distributed random variable in [0, 1]. The initial global best substrate is identified by evaluating the fitness of all candidates.

2.2 | Adaptive Parameter Control

In contrast to its previous version, which had a deterministic increase of the adaptive factor and a constant enzyme concentration, dEAO has dynamic controls that allow the modification of these two parameters on the fly. The adaptive factor in (2) follows a sinusoidal pattern, allowing a gradual transition from exploratory to exploitative search.

$$AF(t) = 0.5 + 0.5 \cdot sin\left(\frac{\pi \cdot t}{t_{max}}\right) \tag{2}$$

This oscillating characteristic makes the optimiser more responsive to different search stages. To avoid premature convergence, the enzyme concentration as defined in (3) is randomised at each iteration:

$$EC(t) = 0.3 + 0.2 \cdot rand \tag{3}$$

where $rand$ is a uniformly distributed random number in [0, 1]. This randomness introduces variability into the search process and helps the algorithm explore the search space more efficiently.

2.3 | Core Search Process

In the main search loop, two candidate solutions are produced by each substrate, called the exploratory candidate and the differential candidate. In the exploratory candidate stage, the first candidate performs exploration and is calculated by (4).

$$X_i^{First} = \left(X^{Best} - X_i\right) + rand(D) \cdot \sin(AF \cdot X_i) \tag{4}$$

Here, X^{Best} represents the current global best solution and $rand(D)$ generates a random vector for each dimension. The sine function introduces nonlinearity to mimic the flexible nature of enzyme conformations. This candidate is subsequently constrained within the defined variable bounds. The second candidate (differential candidate) leverages a differential mutation approach. Two distinct substrates, S_1 and S_2, are randomly selected from the population (excluding the current substrate). Two trial vectors, A and B, are generated as follows:

$$A = X_i + sc_A \cdot (S_1 - S_2) + ex_A \cdot \left(X^{Best} - X_i\right) \tag{5}$$

$$B = X_i + sc_B \cdot (S_1 - S_2) + ex_B \cdot \left(X^{Best} - X_i\right) \tag{6}$$

where sc_A, $ex_A \in [EC, 1]$ are random vectors, and sc_B, $ex_B \in$ [EC, 1] are scalar values (uniform scaling). These scale parameters regulate the intensity and direction of the search, with the sine term being removed here in order to distinguish the exploitation intensity. Two candidates A and B are tested and the better among them becomes the second candidate. After evaluating the two, the better one replaces the substrate if an improvement is found. Also, if this new candidate is better than the current global best solution, then the global best is updated accordingly.

2.4 | Local Search Enhancement

To better promote convergence in the vicinity of optimal regions, dEAO incorporates a simplified local search mechanism applied to the best solution found at the end of each iteration. This process perturbs each coordinate of the best solution slightly, as follows:

$$\delta_d = 0.005 \cdot (UB_d - LB_d) \tag{7}$$

$$trial_d = X^{Best}(d) + \delta_d \cdot (rand - 1) \tag{8}$$

Here, δ_d determines a small step size proportional to the variable's search range, and $rand$ generates a random number in [0, 1]. If

ALGORITHM 1 | Pseudocode of the dEAO.

Begin
- Randomly initialize substrate population using the initialization equation
- Evaluate the initial fitness of the population
- Identify the global best solution
- For each iteration t = 1 to t_{max} do
  - Update adaptive factor AF using sinusoidal adaptation
  - Randomly compute the enzyme concentration EC
  - For each substrate i do
    - Generate the first candidate using the exploratory formula
    - Clip the candidate within variable bounds
    - Randomly select two distinct substrates S_1, $S_2 \neq$ i
    - Generate candidate A using the first differential formula
    - Generate Candidate B using the second differential formula
    - Select the better of A and B as the second candidate
    - Compare the first and second candidates and select the superior one
    - If the chosen candidate improves fitness:
      - Replace substrate i and, if applicable, update the global best
  - End For
  - Apply local search refinement (as described below)
  - If local refinement yields improvement, update the global best
- End For
- Return the global best solution

End

the trial solution improves the global best's fitness, it is accepted as the new best. This refinement mechanism offers precise local adjustments without imposing significant computational overhead (Algorithm 1).

3 | DC Motor Modelling

In the dynamic model of a DC motor, the magnetic field is usually considered constant during operation. This hypothesis applies to motors having permanent magnets or those where the field windings are externally excited [23, 24]. In this case, the electromagnetic torque produced by the motor is a linear function of the armature current, which is given by the following expression:

$$T(t) = K_t \cdot i(t) \tag{9}$$

where $T(t)$ represents the electromagnetic torque, $i(t)$ is the armature current and K_t is the torque constant, expressed in Nm/A. When the rotor rotates inside the magnetic field, a voltage known as the back electromotive force is generated across the armature terminals that opposes the supply voltage. This voltage is related to the angular velocity $\dot{\theta}(t)$, as shown in (10):

$$e(t) = K_e \cdot \dot{\theta}(t) \tag{10}$$

where, The constant K_e known as the back EMF constant, is measured in $V \cdots /rad$. In SI-compliant systems, the numerical values of K_t and K_e are often equal. The mechanical motion of the rotor is described by Newton's second law for rotational systems. Taking into account the rotor's inertia and viscous damping, the governing equation becomes:

$$J \cdot \ddot{\theta}(t) + b \cdot \dot{\theta}(t) = K_t \cdot i(t) \tag{11}$$

In this equation, J denotes the moment of inertia and b is the viscous friction coefficient. On the electrical side, the dynamics of the armature circuit are modelled using Kirchhoff's voltage law. The total applied voltage consists of contributions from the inductor, resistor and back EMF:

$$L \cdot \frac{di(t)}{dt} + R \cdot i(t) = V(t) - K_e \cdot \dot{\theta}(t) \tag{12}$$

Applying the Laplace transform to the system equations, under the assumption of zero initial conditions, yields the following expressions in the frequency domain:

$$(Js + b) \cdot \Theta(s) = K_t \cdot I(s) \tag{13}$$

$$(Ls + R) \cdot I(s) = V(s) - K_e \cdot s \cdot \Theta(s) \tag{14}$$

By eliminating $I(s)$ and selecting $V(s)$ as the input and $\dot{\theta}(s) = s \cdot \Theta(s)$ as the output, the open-loop transfer function of the system can be derived as:

$$\frac{\dot{\theta}(s)}{V(s)} = \frac{K_t}{(Js + b)(Ls + R) + K_t K_e} \tag{15}$$

The second-order transfer function is a good approximation of the input-output behaviour of an armature-controlled DC motor and is widely used for control engineering analysis [16]. The parameters of the motor are identified with the parameter estimation toolbox in MATLAB/Simulink, and the parameter values are shown in Table 1.

TABLE 1 | DC motor parameters.

Parameter	Symbol	Value	Unit
Moment of inertia of the rotor	J	0.026767	kg/m^2
Motor viscous friction constant	b	0.049813	$N \cdot m \cdot s$
Electromotive force constant	K_e	0.020375	(V/rad)/sec
Motor torque constant	K_t	0.020375	Nm/Amp
Electric resistance	R	0.18644	Ω
Electric inductance	L	0.0063	H

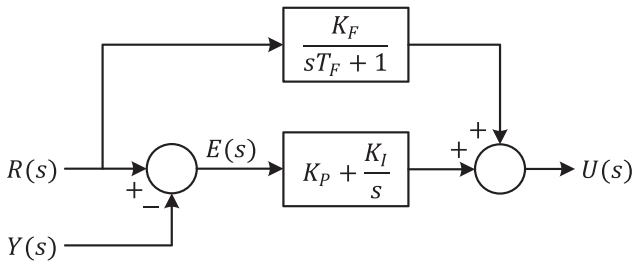


FIGURE 1 | Block diagram of PI + FF controller.

Substituting these values into (15) yields the second-order transfer function of the DC motor, given in (16).

$$G(s) = \frac{0.02038}{0.0001686\, s^2 + 0.005304\, s + 0.009702} \tag{16}$$

This transfer function forms the analytical basis for subsequent controller design and performance evaluation in DC motor speed regulation tasks.

4 | Feedforward-Compensated PI Controller

The control architecture introduced in this paper combines a classic PI controller with a feedforward term to implement efficient speed regulation in DC motors. A classical PI controller suffers from one major drawback due to its purely feedback-based structure, which may lead to inadequate transient response under dynamic operating conditions or sudden load disturbances. In order to mitigate these limitations, a feedforward path is added to predict variations in the reference input and compensate for disturbances before they propagate through the plant. As seen in Figure 1, the PI + FF controller consists of two main branches: the feedforward controller and the feedback controller. The feedback loop uses a PI controller described as

$$C_{PI}(s) = K_P + \frac{K_I}{s} \tag{17}$$

where K_P and K_I denote the proportional and integral gains, respectively. The controller processes the tracking error defined as $E(s) = R(s) - Y(s)$, where $R(s)$ is the reference speed input and $Y(s)$ is the measured motor speed output. The feedforward branch is modelled using a first-order compensator expressed as in (18):

$$C_{FF}(s) = \frac{K_F}{sT_F + 1} \tag{18}$$

where K_F represents the feedforward gain and T_F denotes the feedforward time constant. n contrast to the feedback controller, the feedforward path directly uses the reference signal $R(s)$ to produce anticipatory control effort. This structure enhances transient response and weakens the effect of dynamic lag under rapidly changing operating conditions.

The total control signal $U(s)$ is the sum of the outputs of the PI and feedforward paths. The suggested hybrid configuration facilitates superior reference following, mitigates steady-state error and increases disturbance attenuation capability across different operating conditions. The dEAO is used to find the controller parameters K_P, K_I, K_F and T_F. In contrast with traditional manual or trial-and-error tuning techniques, the presented method casts the parameter tuning process as a constrained optimisation problem. Under this philosophy, each prospective parameter vector $X = [K_P, K_I, K_F, T_F]$ is tested in a closed-loop DC motor model against a set of defined transient response and stability conditions. The gains of the controller affect the closed-loop response in different ways. The proportional term K_P is mostly related to the speed of the transient response and aggressiveness of the control action, while the integral term K_I compensates accumulated errors and eliminates steady-state tracking error. The feedforward gain K_F enhances predictive action to reference variations and measurable disturbances, while the feedforward time constant T_F governs the dynamic smoothing of the feedforward path and avoids oversensitivity to noise or sudden signal changes. The dEAO algorithm progressively searches the parameter space by means of adaptive exploration and exploitation strategies inspired by catalytic reaction dynamics. To evaluate any candidate solution, a composite fitness function is defined as:

$$Fitness = \rho_1 \times m_{os} + \rho_2 \times t_{set} + (1 - \rho_1 - \rho_2) \int_0^{t_{sim}} |e(t)|\, dt \tag{19}$$

where m_{os} is the normalised maximum percent overshoot, t_{set} is the normalised settling time, $e(t) = r(t) - y(t)$ is the tracking error and $t_{sim} = 20$ is the total simulation time. Weighting factors are $\rho_1 = 0.3$ and $\rho_2 = 0.2$ with a tradeoff between overshoot, settling time and total error. The fitness computation is designed such that multiple time-domain performance measures can be considered at the same time during optimisation. Parameter sets that lead to divergence, wild oscillations, or extremely slow responses are naturally penalised under the tested conditions. Thus, the proposed dEAO algorithm converges towards controller parameters that achieve fast transients with less overshoot, superior disturbance rejection and precise steady-state speed regulation. Stability-wise, the proposed PI + FF structure guarantees bounded closed-loop signals for bounded reference and disturbance inputs, as the feedforward compensator is realised as a stable first-order transfer function with positive time constant $T_F > 0$. In addition, due to the composite fitness formulation from (19), unstable or over-oscillatory parameter sets are automatically penalised during the optimisation process.

In particular, the present framework gradually directs the search towards regions of the parameter space associated with stable transient response, low tracking error and good disturbance rejection. Furthermore, the convergence performance in Figure 5 also shows strong and consistent convergence for the presented method. The overall tuning procedure is presented in Figure 2, in which the dEAO-based adaptive parameter tuning scheme for the feedforward-compensated PI controller is integrated into the whole tuning problem, making the PI controller design a systematic procedure for determining the parameters according to the closed-loop dynamic performance rather than relying on trial-and-error gain selection.

5 | Results and Discussions

Following the formulation of the DC motor model and the development of the dEAO-based PI + FF, this section presents

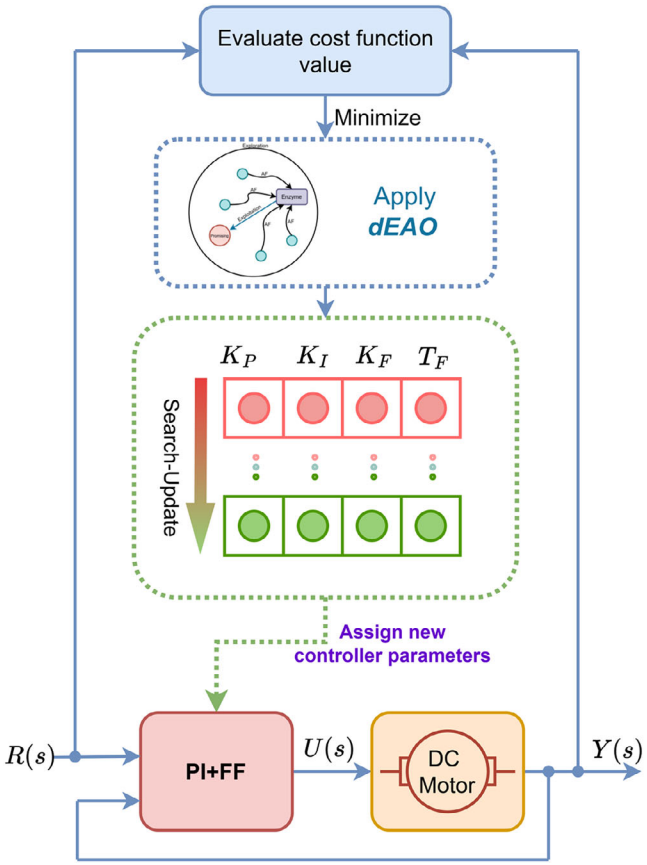


FIGURE 2 | Block diagram of the dEAO-based PI + FF controller parameters in DC motor control.

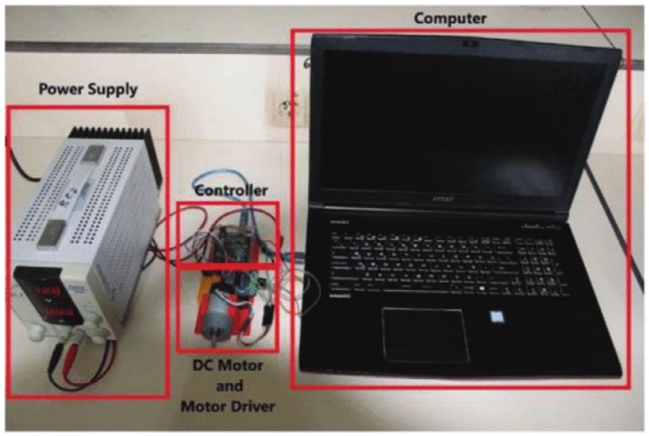


FIGURE 3 | Experimental setup for real-time implementation of dEAO-based PI + FF controller.

TABLE 2 | Statistical comparison of optimisation algorithms for DC motor system.

Optimiser	Minimum	Maximum	Average	Standard deviation
dEAO	**47.1084**	**51.8993**	**48.7636**	**1.2178**
EAO	49.0736	54.6582	50.8265	1.4263
NRBO	52.6577	58.0867	54.4339	1.3444
L-SHADE	51.2802	56.1672	52.9380	1.2742
GA	54.4167	61.4419	56.4693	1.4848

a comparative evaluation based on both simulation results and experimental validations.

5.1 | Compared Algorithms and Experimental Setup

To investigate the efficiency of the proposed dEAO algorithm, a comparative performance evaluation involving four different optimisation algorithms is presented in this study. The algorithms included in the comparison are the original EAO reported in the literature [22] the recently proposed nature-inspired Newton-Raphson-based optimiser (NRBO) [25], the L-SHADE algorithm [26] which has demonstrated strong performance in CEC competitions, and the genetic algorithm (GA) [27], one of the most widely used classical evolutionary techniques. These algorithms are chosen because of their historical importance and their recent success in optimisation research. All the algorithms are tested under the same initial and parameter settings in the experimental study. Within this context, the population size of each algorithm is set to 30 and the total number of iterations is limited to 50. Hence, the computational budget is the same for all the algorithms. To minimise the influence of random behaviour, each algorithm is run 25 times independently for each test case. Hence, one can analyse the average performance as well as the stability properties simultaneously. The cost function is used to evaluate each run of each algorithm. Real-time operation and application of the proposed dEAO-based PI + FF controller are demonstrated on a laboratory-scale DC motor platform depicted in Figure 3. The experimental setup consists of a 12 V DC motor with an IBT-2 H-bridge driver, powered via a regulated DC supply. The controller, previously optimised offline via the dEAO, is directly employed with no further parameter tuning. Real-time implementation is realised using an Arduino-based microcontroller which performs both the control loop and data acquisition. A connection to the host PC is established to capture system responses and monitor controller performance under various dynamic operating conditions.

5.2 | Statistical Performance on Cost Function

During the optimisation, the solutions must not only provide the best results but also have the capability to produce stable and consistent outcomes. In the evaluation with 25 independent runs, minimum, maximum, average and standard deviation values are calculated for each algorithm. The results are shown in Table 2. From the results of the statistical analysis, it can be seen that the proposed dEAO algorithm outperforms the others in both mean cost function value and standard deviation. Although EAO and NRBO algorithms also produced satisfactory results, the obtained mean values lag behind those of dEAO. However, it can be seen that L-SHADE provides limited adaptation at very low iteration counts even though it proved to be a successful algorithm for DC motor system. The GA, however, shows very large mean and standard deviation values, implying that the technique is

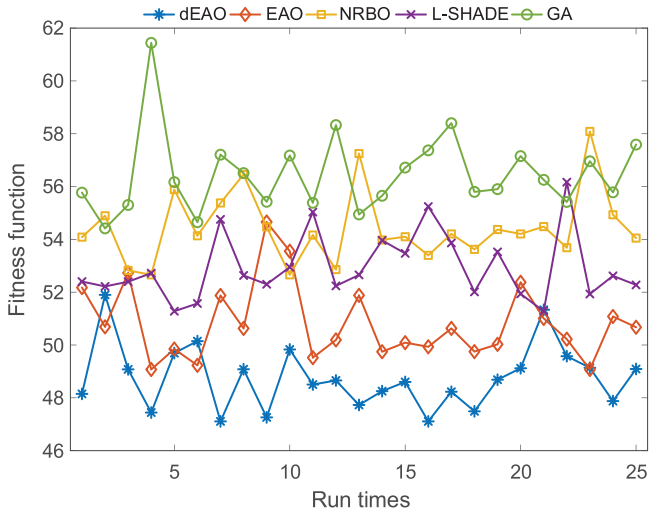


FIGURE 4 | Fitness value variation of dEAO, EAO, NRBO, L-SHADE and GA.

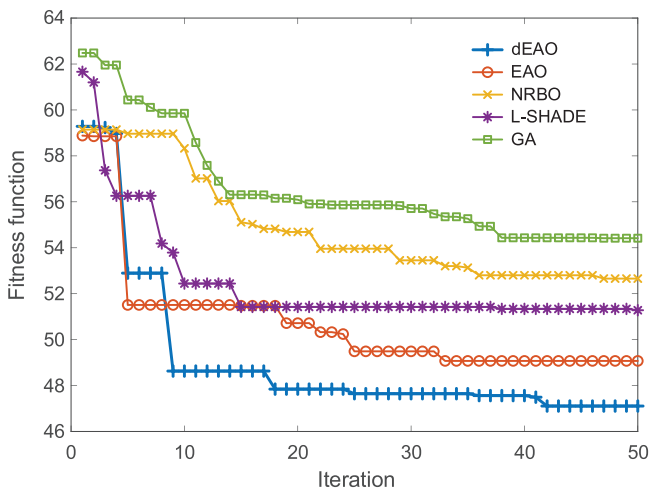


FIGURE 5 | Convergence behaviour of dEAO, EAO, NRBO, L-SHADE and GA.

incapable of delivering sufficiently consistent results for such a short-term search problem.

Figure 4 displays the distribution of the fitness values acquired over 25 runs for each optimisation technique. The line plot also shows that the proposed dEAO is stable and robust since it always obtains lower fitness values with small variance. On the other hand, GA and NRBO demonstrate larger spread and often reach higher fitness values, showing much less stable convergence patterns. This means that dEAO not only obtains the best average performance against other approaches, but also achieves higher repeatability under the same conditions.

The convergence behaviours of the five optimisation algorithms are shown in Figure 5. The proposed dEAO algorithm also shows the best convergence speed and stability, and achieves the lowest fitness value in the least number of iterations. EAO has a smooth and comparatively fast descent to the second-best performance, while NRBO and GA converge slower and to higher fitness values. L-SHADE shows noticeable improvements during the early stages but reaches a plateau prematurely, indicating limited exploitation capability. These results confirm the superior optimisation performance of dEAO in identifying optimal controller parameters.

The Wilcoxon signed-rank test is used to quantitatively demonstrate the effectiveness of the proposed method against other optimisation methods. This non-parametric test examines whether the median of the differences between paired observations is zero, making it useful for comparing outcomes from different tuning approaches. As shown in Table 3, the dEAO-based PI + FF achieves the best performance among all competing strategies with significant p-values well below the 0.05 level, signifying that the achieved enhancements in transient and steady-state performance are not due to chance.

5.3 | Optimised Controller Parameters

Each algorithm developed and compared is run separately on the four basic parameters of the PI + FF controller structure, K_P, K_F, K_I, T_F and parameter optimisation is performed. For each algorithm used in this process, the population size is set to 30 and the maximum number of iterations to 50; each structure is evaluated based on the parameter set that provided the best performance across 25 independent runs. The final gain values are obtained as a result of the optimisation processes are shown in Table 4.

The parameter set obtained with the dEAO algorithm has provided superiority in transient regime performance, such as shorter settling time and lower overshoot. The differences observed in K_P, and K_F values, in particular, have increased the system's early response capability and enabled faster stabilisation. Parameters obtained by algorithms have generally produced balanced performance; classical GA, on the other hand, exhibited a more passive control behaviour with low K_P, and K_F values.

5.4 | Time-Domain Response Performance

The effectiveness of each optimisation algorithm is further assessed by analysing the time-domain performance of the PI + FF controllers they produced. Key transient and steady-state metrics, including rise time (t_{rise}), settling time (t_{set}), percent overshoot (m_{os}), peak time (t_{peak}) and steady-state error (e_{ss}), are extracted from the step response of each control structure. These indicators are summarised in Table 5.

In Figure 6, all the controllers can achieve the reference speed; however, the dEAO-based PI + FF controller is superior to the others in terms of rise time and settling time, while the overshoot and steady-state error are kept at minimum levels. To highlight the fine-grained differences, a zoomed-in view is shown in Figure 7 where it can be noted that dEAO achieves the most balanced performance: fast convergence with the least overshoot. Specifically, the dEAO-based PI + FF exhibited the shortest t_{rise} of 1.1968 s. and the lowest m_{os} as 1.0012%. Additionally, t_{set} is recorded as 1.8964 s, outperforming all other candidates. The e_{ss} achieved by this configuration is negligible (8.73×10^{-8}%),

TABLE 3 | Wilcoxon signed-rank test results.

dEAO versus EAO		dEAO versus NRBO		dEAO versus L-SHADE		dEAO versus GA	
***p*-value**	**Superior**	***p*-value**	**Superior**	***p*-value**	**Superior**	***p*-value**	**Superior**
8.0851E−05	dEAO	1.2290E−05	dEAO	1.3898E−05	dEAO	1.2290E−05	dEAO

TABLE 4 | Optimised PI + FF controller parameters obtained by different algorithms.

Optimiser	K_P	K_I	K_F	T_F
dEAO	0.099396	0.32814	0.28839	0.013752
EAO	0.075973	0.30127	0.29769	0.016747
NRBO	0.052804	0.29353	0.28874	0.016912
L-SHADE	0.064109	0.29332	0.29840	0.040758
GA	0.045118	0.31389	0.27371	0.011320

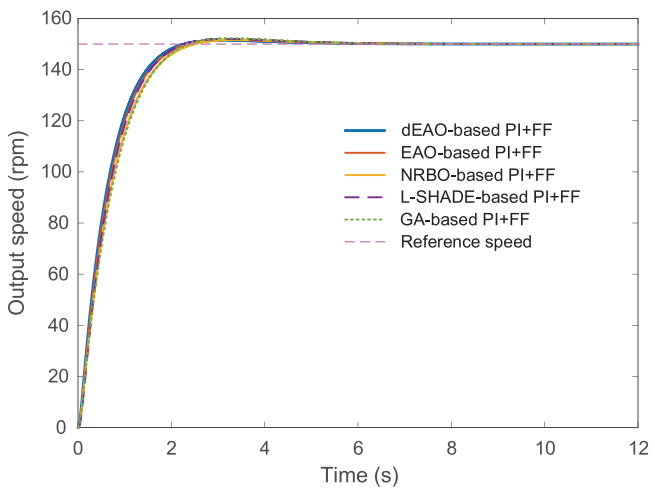


FIGURE 6 | Step response of DC motor speed using PI + FF controllers optimised by different algorithms.

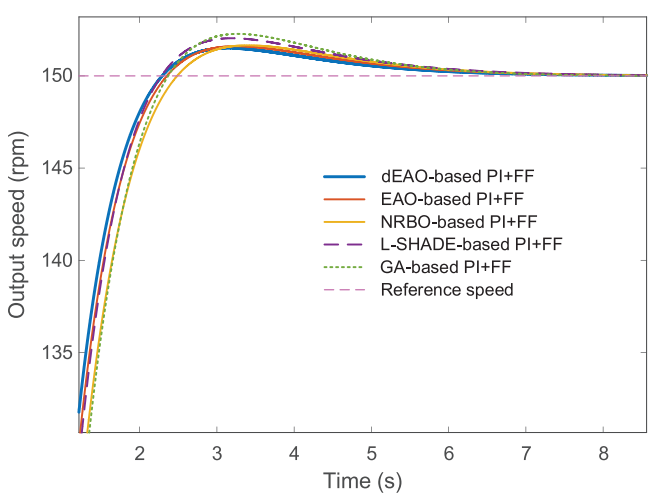


FIGURE 7 | Zoomed-in view of the step responses showing overshoot and settling behaviour of optimised PI + FF controllers.

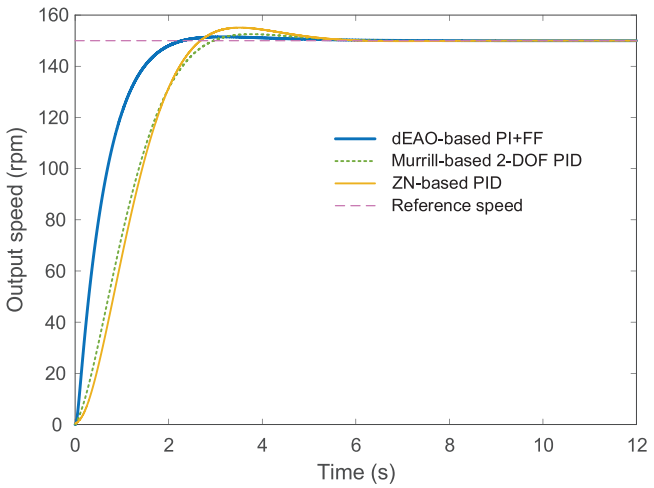


FIGURE 8 | Step response comparison of dEAO-based PI + FF, Murrill-based 2-DOF PID and ZN-based PID controllers.

reflecting high accuracy in maintaining the desired speed. These results affirm the superior transient behaviour and precision of the dEAO-based PI + FF, validating its robustness and suitability for high-performance motion control applications.

5.5 | Comparison With Classical Tuning Rules

To evaluate the effectiveness of the proposed dEAO-based PI + FF controller, a comparative analysis is conducted using classical tuning methods, namely the Murrill-based 2-DOF PID [28] and the Ziegler–Nichols (ZN)-based PID [29] approaches. The goal of this comparison is to explore whether optimisation-based schemes can yield better closed-loop performance than classical heuristic tuning schemes. As seen in Figure 8, the step response curves of all control methods show that the dEAO-based PI + FF controller has a much faster rise time and less overshoot than the classical methods. The ZN-based PID controller has a relatively aggressive behaviour with pronounced overshoot, while the Murrill-designed controller has a more conservative response with a slightly longer settling time.

Quantitative performance metrics are summarised in Table 6. The dEAO-based PI + FF exhibits the shortest t_{rise} as 1.1968 s and lowest m_{os} as 0.0012%, as well as a negligible e_{ss}. In contrast, the ZN-based controller produces the highest m_{os} as 3.3984% and longest t_{set} as 4.4621 s, while the Murrill-based method results in intermediate performance across all parameters.

To provide a more holistic evaluation of control performance, two complementary metrics are considered: the integral of time-weighted absolute error (ITAE) and a composite cost function referred to as the ZLG index, which jointly accounts for overshoot,

TABLE 5 | Time-domain performance metrics of PI + FF controllers optimised by different algorithms.

Control method	t_{rise} (s)	t_{set} (s)	m_{os} (%)	t_{peak} (s)	e_{ss} (%)
dEAO-based PI + FF	1.1968	1.8964	1.0012	3.1480	8.7324E−08
EAO-based PI + FF	1.2391	1.9575	1.0718	3.2470	2.3364E−07
NRBO-based PI + FF	1.3309	2.0834	1.1036	3.4030	9.0324E−08
L-SHADE-based PI + FF	1.2389	1.9420	1.3610	3.1920	1.2316E−07
GA-based PI + FF	1.3502	2.0478	1.5176	3.2720	9.0152E−08

TABLE 6 | Time-domain performance comparison of dEAO-based PI + FF, Murrill-based 2-DOF PID and ZN-based PID controllers.

Control method	t_{rise} (s)	t_{set} (s)	m_{os} (%)	t_{peak} (s)	e_{ss} (%)
dEAO-based PI + FF	1.1968	1.8964	1.0012	3.1480	8.7324E−08
Murrill-based 2-DOF PID	1.8264	2.6716	1.6989	3.7710	1.0556E−07
ZN-based PID	1.7026	4.4621	3.3984	3.5140	1.0519E−07

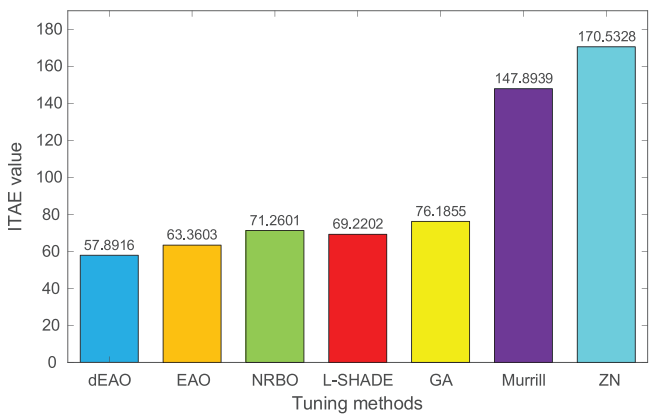


FIGURE 9 | ITAE-based performance comparison of controllers tuned by different optimisation and classical methods.

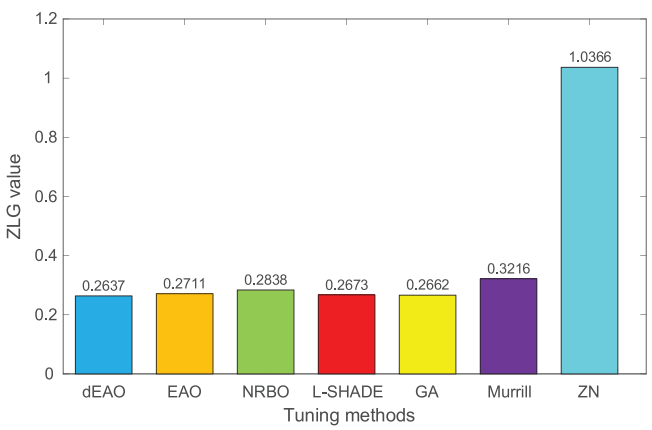


FIGURE 10 | ZLG-based performance comparison of controllers tuned by different optimisation and classical methods.

steady-state error and timing characteristics. These performance criteria are computed according to the definitions given in (20) and (21), with a total simulation time $t_{sim} = 20$ s and a normalisation factor of $\sigma = 1$.

$$ITAE = \int_0^{t_{sim}} t\,|e(t)|\,dt \tag{20}$$

$$ZLG = (1 - e^{-\sigma})\left(\frac{m_{os} + e_{ss}}{100}\right) + e^{-\sigma}(t_{set} - t_{rise}) \tag{21}$$

As seen in Figure 9, the proposed method yields the lowest ITAE value with 57.89, outperforming EAO (63.36), L-SHADE (69.22) and ZN-based tuning, which shows the poorest performance (170.53). A similar trend is evident in the ZLG metric, where dEAO achieves the minimum cost (0.2637), while the ZN-tuned controller records the highest value (1.0366), as illustrated in Figure 10. These findings confirm that advanced metaheuristic approaches offer superior performance in both transient and steady-state domains. By minimising cumulative tracking errors and penalising undesirable dynamic behaviours, such methods prove highly suitable for control tasks requiring fast and accurate responses.

5.6 | Tracking and Disturbance Rejection Capability

In this section, the reference tracking performance and stability against external disturbances of the PI + FF controller tuned with the dEAO algorithm are analysed. In Figure 11, the tracking performance of the system is evaluated by changing the reference signal to different levels over time. As can be seen, the dEAO-based PI + FF is able to track the changing reference values with high accuracy and demonstrate satisfactory performance with both a fast transition time and low steady-state error.

System stability should be evaluated both by monitoring performance and by its response to unexpected external disturbances. For this purpose, as shown in Figure 12, both measurement noise and external disturbance inputs are applied to the system. The external disturbance inputs, represented by the black line, are applied at ±20 rpm levels to simulate instantaneous speed changes; the red line represents measurement noise. The applied disturbance signals test the system's robustness and error suppression capacity.

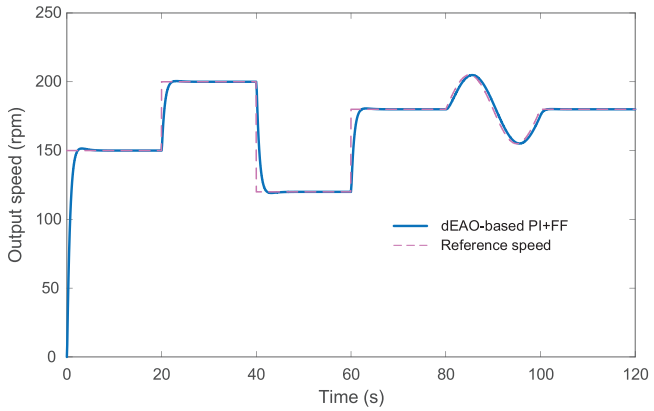


FIGURE 11 | Reference tracking performance of the dEAO-based PI + FF controller under time-varying speed commands.

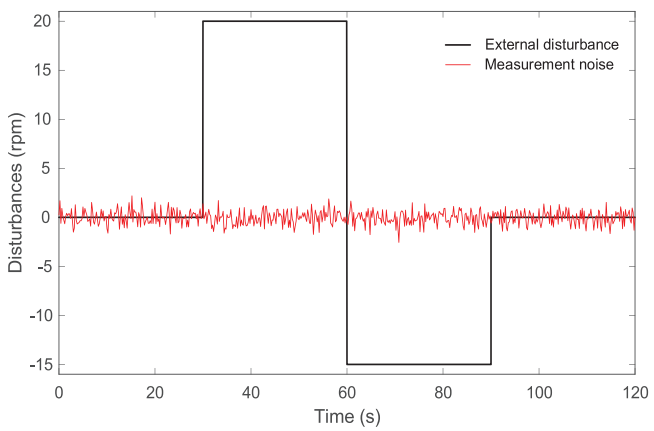


FIGURE 12 | Profile of external load disturbance and measurement noise applied to the system.

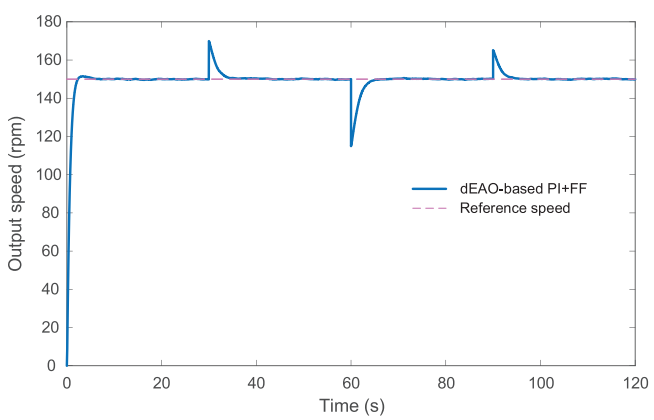


FIGURE 13 | Disturbance rejection performance of the dEAO-based PI + FF controller under external load and measurement noise.

Figure 13 shows the system response under the influence of external disturbances and measurement noise. Although noise is added, the closed-loop behaviour of the dEAO-based PI + FF controller remains stable and the disturbance-induced error in motor speed is quickly brought back to the desired reference value within a short time. The arising oscillations are well damped and no persistent oscillatory behaviour or steady-state drift is observed. These findings illustrate that the proposed control design can maintain tracking performance and dynamic stability under uncertain operating conditions.

Furthermore, the applied disturbance and noise scenarios emulate practical effects commonly encountered in DC motor systems, including sensor imperfections, friction-related fluctuations and unknown external perturbations. Therefore, the obtained results confirm the robustness and practical applicability of the proposed dEAO-based PI + FF control framework for real-world motion control applications.

5.7 | Experimental Results

In order to verify the simulation results and study the real-time applicability of the proposed dEAO-based control scheme, real-time experiments are conducted on a laboratory DC motor system with variable reference input, encoder-based speed measurement, and noise injection capability. The obtained optimal controller parameters through the offline dEAO procedure are utilised directly without further tuning. The first experimental run involved reference tracking under a standard startup profile. As can be seen from Figure 14, all the optimisation-based PI + FF controllers can track the desired reference trajectory with good regulation performance. In general, the NRBO- and EAO-based controllers provide sufficiently smooth transient responses during the initial response stage, whereas the GA- and L-SHADE-based controllers exhibit more pronounced oscillatory behaviours. Although the transient responses of certain other approaches look smoother at the beginning, the dEAO-based PI + FF provides the most balanced performance in terms of rise time, settling time, overshoot and steady-state error when all these factors are considered together. Particularly, the dEAO-based PI + FF exhibits faster convergence to the reference trajectory with low overshoot and negligible steady-state error, which is consistent with the numerical performance indices shown in Table 5.

The second experiment is designed to test the controller under complex, time-varying reference commands, including sharp transitions, plateaus and ramp segments. Figure 15 illustrates that the controller is capable of accurately following the multi-phase reference profile in real time, exhibiting negligible tracking deviation and no visible steady-state drift. Moreover, the measured response of the physical system aligns closely with the modelled output, confirming the fidelity of the simulation model and the effectiveness of the control algorithm in practice. In both experiments, no additional compensations or adaptive mechanisms are employed, underscoring the robustness and plug-and-play characteristics of the proposed method. The absence of overshoot in high-speed segments and the rapid correction during dynamic transitions further support the controller's suitability for real-world motion control applications where precision and stability are critical. Overall, these results validate that the dEAO-based PI + FF control framework both excels in simulated environments and maintains its performance advantages under real-time constraints, including measurement noise, non-idealities and actuation limits. Its consistent behaviour across various dynamic scenarios indicates strong generalisation ability and practical readiness for embedded control applications.

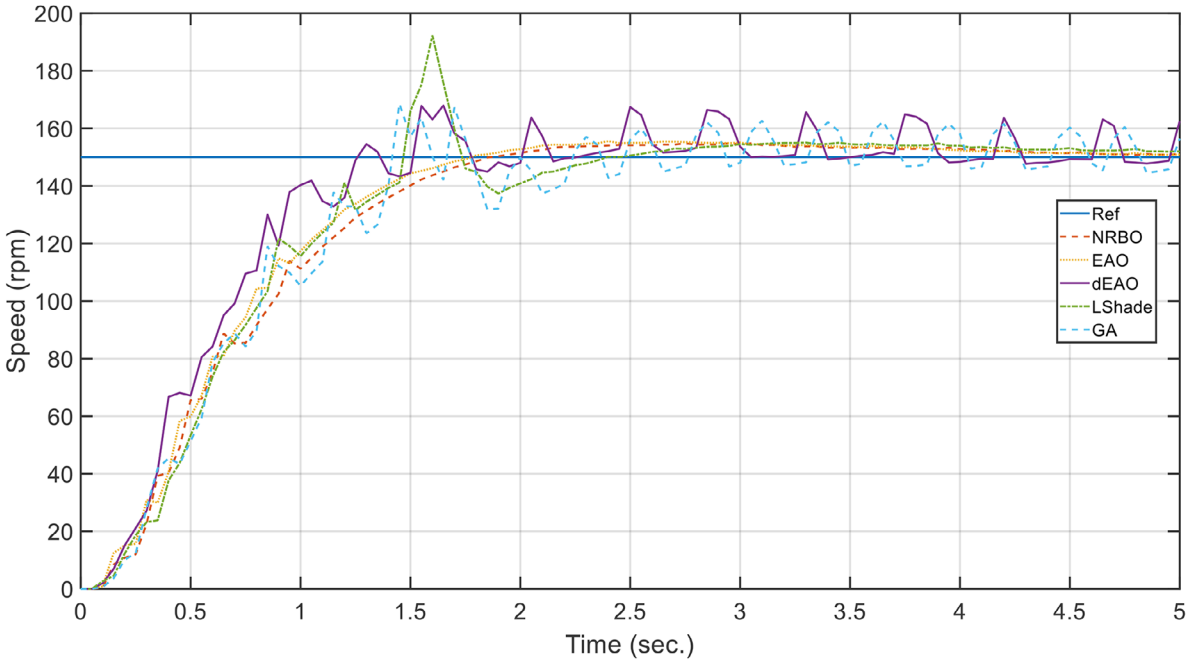


FIGURE 14 | Step response comparison of PI + FF controllers with different optimisers.

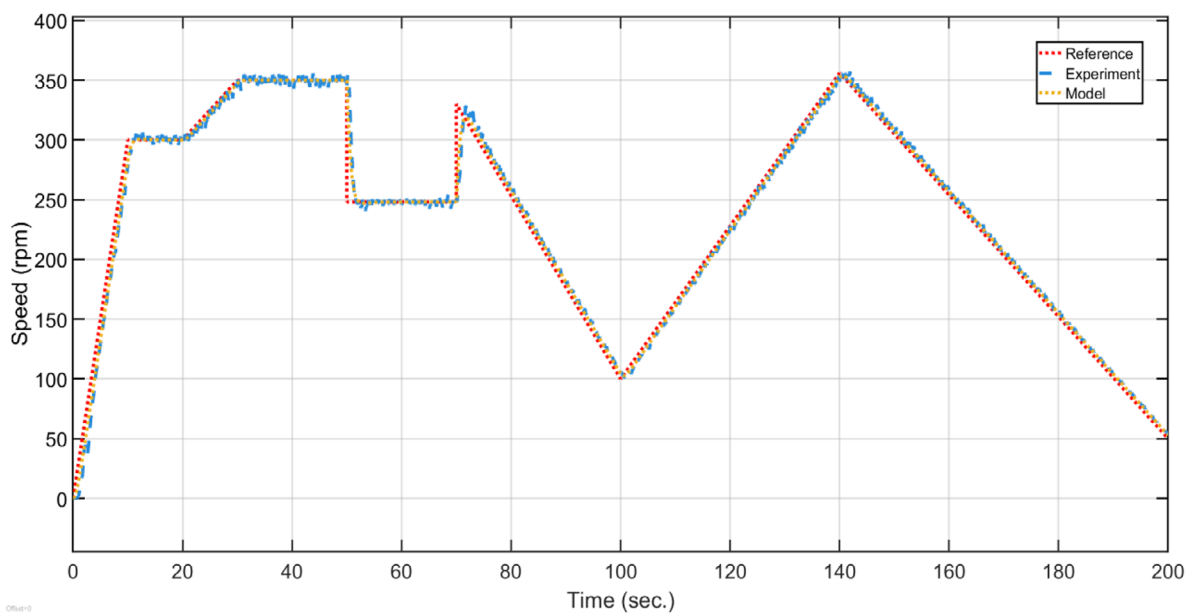


FIGURE 15 | Real-time tracking performance of dEAO-based PI + FF controller.

6 | Conclusions and Future Works

The present investigation formulates a PI + FF control scheme for the speed control of a DC motor optimised using the dEAO under dynamic operating conditions. Different from the standard PI-type schemes, the proposed approach is based on feedforward compensator design and adaptive parameter optimisation to optimise feedback and feedforward controller parameters simultaneously in an integrated optimisation process. The proposed methodology enhances the transient response, disturbance rejection and steady-state control performances. Integrating feedforward compensation with the adaptive search mechanism of the dEAO, the controller showed greater flexibility and robustness when subject to varying operating conditions, external disturbances and measurement noise. The simulation results and experiments showed the superiority of the dEAO-based PI + FF controller over the traditional tuning approaches as well as other advanced optimisation algorithms, namely EAO, NRBO, L-SHADE and GA. The rise time, settling time, overshoot, steady-state error and ITAE performance indices were enhanced remarkably. In addition, the controller exhibited stable reference tracking and robust closed-loop performance against disturbance

and noisy environments. However, this research is not without its limitations. The proposed scheme was tested mainly in predefined operating conditions. Moreover, nonlinear friction dynamics, actuator saturation effects, and more general classes of nonlinear electromechanical systems have not been considered explicitly in this work. In the future, the work may be extended to realise the proposed framework on real-time embedded hardware platforms, namely DSPs and FPGAs, to assess computational efficiency and practical implementation feasibility. Moreover, it is expected that the proposed method can be extended to nonlinear/time-delay system models, to online adaptive parameter tuning, to gain-scheduling type of mechanisms and to the integration with a disturbance observer for enhanced robustness and adaptivity in highly uncertain operating environments. Such extensions might lead to more robust and intelligent control frameworks for advanced motion control applications.

Author Contributions

Gökhan Yüksek: conceptualisation, methodology, software, visualisation, investigation, writing – original draft preparation. **Serdar Ekinci:** conceptualisation, methodology, software, visualisation, investigation, writing – original draft preparation. **Davut Izci:** data curation, validation, supervision, resources, writing – review and editing. **Timur Lale:** conceptualisation, methodology, software, visualisation, investigation, writing – original draft preparation. **Rıdvan Fırat Çınar:** data curation, validation, supervision, resources, writing – review and editing. **Burcu Bektaş Güneş:** data curation, validation, supervision, resources, writing – review and editing. **Mohit Bajaj:** project administration, supervision, resources, writing – review and editing. **Vojtech Blazek:** project administration, supervision, resources, writing – review and editing. **Olena Rubanenko:** project administration, supervision, resources, writing – review and editing.

Acknowledgements

This research is funded by European Union under the REFRESH—Research Excellence For Region Sustainability and High-Tech Industries Project via the Operational Programme Just Transition under Grant CZ.10.03.01/00/22_003/0000048; in part by the National Centre for Energy II under Grant TN02000025 and ExPEDite Project a Research and Innovation Action to Support the Implementation of the Climate Neutral and Smart Cities Mission Project TN02000025; and in part by ExPEDite through European Union's Horizon Mission Programme under Grant 101139527. The authors would like to express their sincere gratitude to Stanislav Misak for his exceptional supervision, project administration, and overall guidance throughout the course of this project. His expertise and support were instrumental to its success.

Conflicts of Interest

The authors declare no conflicts of interest.

Data Availability Statement

All relevant data are within the manuscript. The collection and analysis method complied with the terms and conditions for the source of the data.

References

1. K. Sayed, H. H. El-Zohri, A. Ahmed, and M. Khamies, "Application of Tilt Integral Derivative for Efficient Speed Control and Operation of BLDC Motor Drive for Electric Vehicles," *Fractal and Fractional* 8, no. 1 (January 2024): 61, https://doi.org/10.3390/fractalfract8010061.

2. G. Yuksek, A. Alkaya, and A. N. Mete, "Implementation of Linear Active Disturbance Rejection Control for a DC Motor System," in *2021 13th International Conference on Electrical and Electronics Engineering (ELECO)* (IEEE, 2021), 119–123, https://doi.org/10.23919/ELECO54474.2021.9677843.

3. K. Vanchinathan, K. R. Valluvan, C. Gnanavel, and C. Gokul, "Numerical Simulation and Experimental Verification of Fractional-Order PI$^{\lambda}$ Controller for Solar PV Fed Sensorless Brushless DC Motor Using Whale Optimization Algorithm," *Electric Power Components and Systems* 50, no. 1–2 (January 2022): 64–80, https://doi.org/10.1080/15325008.2022.2135644.

4. M. Khalifa, A. H. Amhedb, and M. Al Sharqawi, "Position Control of Real Time DC Motor Using LabVIEW," *Journal of Robotics and Control (JRC)* 2, no. 5 (2021): 342–348, https://doi.org/10.18196/jrc.25104.

5. J. Niembro-Ceceña, R. Gómez-Loenzo, J. Rodríguez-Reséndiz, O. Rodríguez-Abreo, and Á. Odry, "Auto-Regression Model-Based Off-Line PID Controller Tuning: An Adaptive Strategy for DC Motor Control," *Micromachines* 13, no. 8 (August 2022): 1264, https://doi.org/10.3390/mi13081264.

6. D. Baidya, S. Dhopte, and M. Bhattacharjee, "Sensing System Assisted Novel PID Controller for Efficient Speed Control of DC Motors in Electric Vehicles," *IEEE Sensors Letters* 7, no. 1 (January 2023): 1–4, https://doi.org/10.1109/LSENS.2023.3234400.

7. A. F. Güven, O. Ö. Mengi, M. A. Elseify, and S. Kamel, "Comprehensive Optimization of PID Controller Parameters for DC Motor Speed Management Using a Modified Jellyfish Search Algorithm," *Optimal Control Applications and Methods* 46, no. 1 (January 2025): 320–342, https://doi.org/10.1002/oca.3218.

8. G. Yüksek, "A Novel Delay-Affected Two Degree of Freedom PID Controller Using Physics-Inspired Optimization for Robust Control Applications," *Scientific Reports* 16 (2026): 3453, https://doi.org/10.1038/s41598-025-33481-x.

9. E. Eker, M. Kayri, S. Ekinci, and D. Izci, "A New Fusion of ASO With SA Algorithm and Its Applications to MLP Training and DC Motor Speed Control," *Arabian Journal for Science and Engineering* 46, no. 4 (2021): 3889–3911, https://doi.org/10.1007/s13369-020-05228-5.

10. M. A. Bhayo, J. A. Laghari, M. Aljaidi, et al., "High Precision Experimentally Validated Adaptive Neuro Fuzzy Inference System Controller for DC Motor Drive System," *Scientific Reports* 15, no. 1 (April 2025): 14445, https://doi.org/10.1038/s41598-025-97549-4.

11. D. Izci, E. Köse, and S. Ekinci, "Feedforward-Compensated PI Controller Design for Air–Fuel Ratio System Control Using Enhanced Weighted Mean of Vectors Algorithm," *Arabian Journal for Science and Engineering* 48, no. 9 (September 2023): 12205–12217, https://doi.org/10.1007/s13369-023-07724-w.

12. J. L. Guzmán and T. Hägglund, "Tuning Rules for Feedforward Control From Measurable Disturbances Combined With PID Control: A Review," *International Journal of Control* 97, no. 1 (January 2024): 2–15, https://doi.org/10.1080/00207179.2021.1978537.

13. M. Jabari, S. Ekinci, D. Izci, M. Bajaj, and I. Zaitsev, "Efficient DC Motor Speed Control Using a Novel Multi-Stage FOPD(1 + PI) Controller Optimized by the Pelican Optimization Algorithm," *Scientific Reports* 14, no. 1 (September 2024): 22442, https://doi.org/10.1038/s41598-024-73409-5.

14. G. Saravanan, C. Pazhanimuthu, and P. Naveen, "Performance Improvement of DC Motor Control System Using PID Controller With Kookaburra and Red Panda Optimization Algorithm," *Scientific Reports* 15, no. 1 (2025): 20021, https://doi.org/10.1038/s41598-025-87607-2.

15. E. Çelik, G. Bal, N. Öztürk, et al., "Improving Speed Control Characteristics of PMDC Motor Drives Using Nonlinear PI Control," *Neural Computing and Applications* 36, no. 16 (June 2024): 9113–9124, https://doi.org/10.1007/s00521-024-09568-3.

16. S. Chen, "Study on Load Disturbance Rejection of PID Control in DC Motor Speed Regulation Systems," *Advances in Engineering Innovation* 16, no. 5 (May 2025): 118–122, https://doi.org/10.54254/2977-3903/2025.23523.

17. F. Shafique, M. S. Fakhar, A. Rasool, and S. A. R. Kashif, "Analyzing the Performance of Metaheuristic Algorithms in Speed Control of Brushless DC Motor: Implementation and Statistical Comparison," *PLoS ONE* 19, no. 10 (October 2024): e0310080, https://doi.org/10.1371/journal.pone.0310080.

18. S. Ekinci, G. Yuksek, R. F. Cinar, C. Turkeri, and D. Li, "Gudermannian-Based Error-Shaped PID Controller Tuned via THRO for Nonlinear CSTR Temperature Control," *Chaos, Solitons & Fractals* 209 (August 2026): 118383, https://doi.org/10.1016/j.chaos.2026.118383.

19. Q. Mao, Y. Xu, J. Chen, J. Chen, and T. T. Georgiou, "Maximization of Gain/Phase Margins by PID Control," *IEEE Transactions on Automatic Control* 70, no. 1 (January 2025): 34–49, https://doi.org/10.1109/TAC.2024.3417717.

20. Q. Mao, Y. Xu, J. Chen, and T. T. Georgiou, "Implementation-Oriented Filtered PID Control: Optimization of Robustness Margins," *Automatica* 152 (June 2023): 110974, https://doi.org/10.1016/j.automatica.2023.110974.

21. Q. Mao, J. Chen, F. Xie, L. Dou, B. Tian, and Q. Zong, "Harnessing Optimal Robustness Towards PI Regulators of Linear Perturbed Unstable Systems," *Systems & Control Letters* 205 (November 2025): 106225, https://doi.org/10.1016/j.sysconle.2025.106225.

22. A. Rodan, A.-K. Al-Tamimi, L. Al-Alnemer, S. Mirjalili, and P. Tiňo, "Enzyme Action Optimizer: A Novel Bio-Inspired Optimization Algorithm," *The Journal of Supercomputing* 81, no. 5 (April 2025): 686, https://doi.org/10.1007/s11227-025-07052-w.

23. D. Izci, S. Ekinci, R. M. Rizk-Allah, et al., "Master–Slave Architecture Enhanced and Improved GBO Tuned Cascaded PI - PDN Controller for Speed Regulation of DC Motors," *Optimal Control Applications and Methods* 46 (May 2025): 2137–2152, https://doi.org/10.1002/oca.3313.

24. S. Ekinci, D. Izci, M. H. Almomani, et al., "Advanced Control Parameter Optimization in DC Motors and Liquid Level Systems," *Scientific Reports* 15, no. 1 (January 2025): 1394, https://doi.org/10.1038/s41598-025-85273-y.

25. R. Sowmya, M. Premkumar, and P. Jangir, "Newton-Raphson-Based Optimizer: A New Population-Based Metaheuristic Algorithm for Continuous Optimization Problems," *Engineering Applications of Artificial Intelligence* 128 (February 2024): 107532, https://doi.org/10.1016/j.engappai.2023.107532.

26. R. Tanabe and A. S. Fukunaga, "Improving the Search Performance of SHADE Using Linear Population Size Reduction," in *2014 IEEE Congress on Evolutionary Computation (CEC)* (IEEE, 2014), 1658–1665, https://doi.org/10.1109/CEC.2014.6900380.

27. J. H. Holland, "Genetic Algorithms," *Scientific American* 267, no. 1 (July 1992): 66–72, https://doi.org/10.1038/scientificamerican0792-66.

28. J. D. Rojas, O. Arrieta, and R. Vilanova, *Industrial PID Controller Tuning* (Springer International Publishing, 2021), https://doi.org/10.1007/978-3-030-72311-8.

29. K. Ogata, *Modern Control Engineering*, 4th ed. (Prentice Hall PTR, 2001).