



Comparative Performance Analysis of PI Speed Controller Tuning Techniques for a Vector-Controlled BLDC Motor Drive

Dibyadeep Bhattacharya^{1,2}, Raja Gandhi³, Moumi Pandit^{2*}, Pankaj Singh³, Karma Sonam Sherpa², Rakesh Roy¹

¹Department of Electrical Engineering, National Institute of Technology, Meghalaya, Shillong, India

²Electrical and Electronics Engineering, Sikkim Manipal Institute of Technology, Sikkim Manipal University (SMU), Majitar, India

³Department of Electrical Engineering, Supaul College of Engineering Supaul, 852131, Bihar, India

*Corresponding Author: moumi.p@smit.smu.edu.in

Abstract

The performance of PI based speed controller for Brushless DC motor Drive has been analysed in this paper. BLDC motors demand precise control strategies to mitigate steady-state errors and enhance transient response in high-performance applications like electric vehicles. Traditional manual tuning methods may not provide optimal PI controller gains. Therefore, this study uses three optimization techniques, namely Particle Swarm Optimization (PSO), Firefly algorithm (FA), and Cuckoo Search Algorithm (CSA), to tune the PI speed controller. The performance of the controllers is evaluated under three different operating conditions: start-up, urban driving cycle, and hill-climbing conditions, representing typical real-world operating conditions of an EV. The speed response is analysed in terms of speed overshoot, speed error, speed ripple, rise time, and settling time. The response of the controllers under load variations is also examined to evaluate their ability to maintain the desired speed when an external load acts as a disturbance. In the urban driving cycle, the controller performance is further analysed under frequent speed increase, decrease, and short-duration stops, representing typical traffic conditions. The results show that the optimized PI controllers provide improved speed response compared with the conventional PI controller. Among the considered techniques, the CSA-PI controller provides lower speed overshoot, speed error, and speed ripple, along with a faster rise time, while during loading or uphill condition the FA-PI perform well as it give only 0.2% speed dip during sudden load change. However, this study provides a basis for selecting a suitable PI tuning technique according to the operating requirements of BLDC motor drives used in EV applications.

Keywords: PI controller, PSO, CSA, FA, Optimization, BLDC Motor, Vector Control

1. Introduction

Nowadays, the use of electric vehicles (EVs) is growing very rapidly across the world due to rising demand for clean, efficient, and low-maintenance transportation. This growth is not limited to passenger cars; it is also very significant in the small electric vehicle segment, such as e-rickshaws, electric three-wheelers, scooters, and other light electric vehicles. These vehicles commonly use electric motors such as Brushless DC (BLDC) motors or PMSM due to their high efficiency, compact size, high torque-to-weight ratio, and low maintenance requirements. However, for the proper operation of an electric vehicle, efficient and accurate motor control is very important. Various control techniques are available for controlling BLDC motors, such as Scalar control, direct torque control, and vector control. Among these techniques, vector control is widely used because it provides good dynamic performance, accurate torque and speed control, and smooth motor operation. In a vector-control-based motor drive, the PI controller plays a very important role, particularly in the speed Controller. The PI controller mainly depends on two important parameters: the proportional gain (K_p) and the integral gain (K_i). The proper selection of these two parameters is essential for obtaining good motor performance. If K_p and K_i are selected properly, the motor can achieve fast response, low overshoot, small steady-state error, better speed tracking, and smooth operation. However, improper selection of these gains may result in excessive overshoot, oscillations, slow response, or even instability. Therefore, proper tuning of the PI-controller parameters is a key requirement for achieving high-performance BLDC motor control.

In the literature, there are various ways listed to tune the PI controller of the motor drive, such as Conventional tuning such as trial and error method, which is a very popular method that can provide suitable controller parameters, but it may not always provide the best performance for nonlinear and changing motor operating conditions [1]. Due to this limitation, the metaheuristic optimization approaches come in the scenario as powerful tools that are not dependent on the exact mathematical model and avoid local optima [2] which make it ideal for tuning controllers in given systems. These techniques, such as Particle Swarm Optimization (PSO), Genetic Algorithms (GA), and Grey Wolf Optimizer (GWO) and Whale Optimization (WO), Dragonfly Algorithm, Bat Algorithm, and Cuckoo Search have been widely used in

various engineering applications, including renewable energy systems, wind turbine pitch control, power converters, electric drives, and energy management systems [3]-[11]. In recent years, these optimization techniques have also gained attention in electric motor drives, particularly for tuning speed and current controllers. Metaheuristic optimization techniques have been widely applied to improve the performance of motor-drive controllers by optimizing PI/PID parameters according to objectives such as speed error, settling time, overshoot, and torque ripple. Genetic Algorithm (GA) has been used for PID tuning of DC and BLDC motors, providing improved transient response over conventional tuning [12]. Particle Swarm Optimization (PSO) has been applied extensively for BLDC PI-controller tuning, with reported improvements in speed tracking and dynamic response [13]–[15]. Cuckoo Search (CS) has been employed for DC-motor controller optimization and demonstrated improved regulation characteristics [16]. Grey Wolf Optimization (GWO) has been investigated for both BLDC and PMSM drives, particularly for improving speed tracking and reducing speed ripple [17][18]. Other techniques such as Firefly Algorithm (FA), Modified Firefly Algorithm (MFA), and Whale Optimization Algorithm (WOA) have also been applied to BLDC controller tuning, with studies reporting improvements in convergence, speed regulation, and transient performance [19]–[20].

Collectively, these studies indicate that GA, PSO, CS, GWO, FA/MFA, and WOA can provide systematic alternatives to conventional trial-and-error controller tuning. However, their performance depends on the selected objective function, search limits, motor parameters, controller structure, and operating conditions. Therefore, a comparative evaluation using identical BLDC motor parameters and common performance indices such as settling time, overshoot, steady-state error and torque ripple is necessary to establish the relative effectiveness of different optimization approaches [21]–[24]. The inherent non-linearity and sensitivity of Brushless DC motor frameworks to load disturbances present significant challenges, as conventional tuning often fails to mitigate peak overshoot and settling time issues effectively [24][25] and therefore needs the application of different optimization techniques to tune the gain parameters of PI controllers.

This paper therefore provides a comparative evaluation of metaheuristic optimization techniques—such as Particle Swarm Optimization, the Firefly algorithm, and Cuckoo search algorithm—against traditional analytical tuning methods. By employing a simulation framework, this analysis establishes rigorous performance metrics—including Integral of Time-weighted Absolute Error and root mean squared error—to quantify dynamic response, disturbance rejection, and tracking precision across diverse operational scenarios. This study systematically benchmarks how this diverse metaheuristic strategies outperform empirical tuning by minimizing overshoot and reducing steady-state error during sudden load fluctuations. Due to this limitation, optimization-based techniques come into the scenario, where modern metaheuristic approaches such as Particle Swarm Optimization, Firefly Algorithm, and Cuckoo Search can be used to obtain more suitable PI-controller parameters [26]. In this work, these optimization techniques are applied to tune the PI-controller parameters of the BLDC motor drive, leveraging their ability to navigate complex, multidimensional search spaces without requiring gradient information [27]. Their performance can then be compared with conventional PI tuning based on important parameters such as rise time, settling time, overshoot, steady-state error, and integral error indices [28].

The reviewed literature supports PSO as the most established optimizer for BLDC PI tuning, particularly for speed control across changing loads and for implementations with experimental validation [29-32]. FA and modified FA methods show competitive or superior speed-tracking and convergence results in several studies, including variable-load driving-cycle tests [33, 22]. CSA is especially relevant when the design explicitly constrains load regulation or combines speed settling with torque-ripple reduction [34-36].

In this paper, a vector-controlled BLDC drive is modeled using MATLAB/Simulink to investigate the performance of different PI controller tuning techniques. Three optimization techniques, namely Particle Swarm Optimization (PSO), Firefly Algorithm (FA), and Cuckoo Search Algorithm (CSA), are used to tune the PI speed controller and their performance is compared with the conventional tuning method. The controllers are evaluated based on key speed response parameters, including speed overshoot, peak time, settling time, speed error, and speed ripple. As the literature shows relatively limited research on the application of CSA for PI controller tuning in BLDC drives, this work focuses on a modified CSA-based tuning approach and provides a clear performance comparison with PSO- and FA-based tuning methods.

The present paper provides a brief introduction backed up by a detailed literature survey followed by a brief discussion on the BLDC control drive system and the working principles of the three optimization algorithms used in the next section of methodology. Following methodology, there is a detailed discussion on different operating cases under which the PI-optimized BLDC drive system is tested upon. Finally a detailed conclusion is the last section of the paper.

2. Methodology

This section defines the working of the brushless DC motor drive alongside the PI control and the specific criteria utilized to evaluate comparative tuning effectiveness. The BLDC motor drive is modelled using MATLAB/Simulink with parameters as shown in Table 1, where the PI controller regulates the motor speed by adjusting the reference current based on the error between the desired ω_{ref} and actual speed ω_{act} , as shown in Fig. 1. Therefore, the PI controller gains, namely K_p and K_i , need to be tuned accurately. The PI

controller output for the speed loop is expressed as (1). This PI speed controller generates the reference current as shown in (3), which further used to generate the gate pulse of the inverter. Therefore, PI parameters play a critical role in shaping the system response under frequent speed changes and external load disturbances. As a result, the motor achieves improved speed regulation, faster command tracking, and reduced energy losses. For tuning the PI controller parameters, the total speed error is considered as the objective function, as given in (4). The optimization techniques are then used to obtain the optimal values of K_p and K_i by minimizing the fitness function as shown in (5).

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau \quad (1)$$

where

$u(t)$ is the controller output,

K_p is the proportional gain,

K_i is the integral gain, and

$e(t)$ is the tracking error. For speed control,

$$e(t) = \omega_{ref}(t) - \omega_{act}(t) \quad (2)$$

$$I_{ref} = K_p e(\omega_{ref}(t) - \omega_{act}(t)) + K_i \int_0^t |(\omega_{ref}(t) - \omega_{act}(t))| d\tau \quad (3)$$

$$TSE = \sum_{t=0}^{T_{max}} (\omega_{ref} - \omega_{act})^2 dt \quad (4)$$

$$Fitness\ Function = \min(TSE) \quad (5)$$

This optimization procedures employ metaheuristic algorithms to minimize objective functions such as the total speed error, ensuring a balanced convergence between rise times and settling characteristics. These performance criteria are formally assessed through quantitative metrics including the peak overshoot percentage and the steady-state error magnitude during step-load perturbations. A major advantage of this design is its ability to accurately tune the controller gains without using mathematical model.

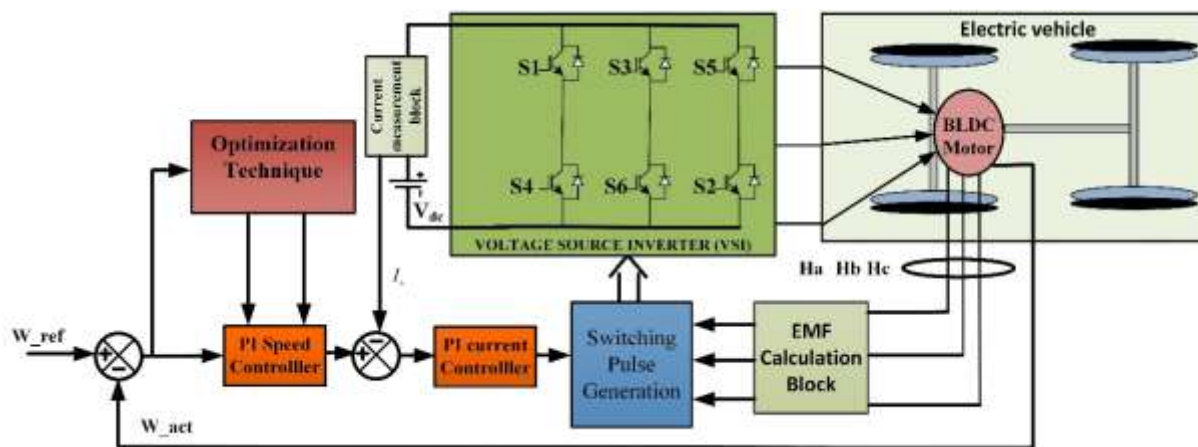


Fig. 1: Block diagram of Optimized PI-based Vector control of BLDC Motor drive

Table 1: BLDC Motor specification

Parameters	Value	Unit
Rated power	1	kW
Rated voltage	48	Volts
Rated current	3	Amp
Rated speed	3000	Rpm
Inductance	0.309	mH
Torque constant	0.33	Nm/A
Resistance	0.75	ohms

A. Conventional PI Controller

The conventional PI tuning relies on fixed gain values derived from classical linear control theory, which frequently results in suboptimal dynamic performance when confronted with the nonlinearities inherent in brushless DC motor drives (Rahmani et al., 2012). These deterministic methods often lack the robustness to handle the complex, time-varying dynamics and high-order nonlinearities prevalent in industrial BLDC systems (Rahmani et al., 2012). Consequently, these heuristic approaches often result in significant speed

ripples and transient oscillations during step-load variations, limiting their efficacy in precision motion control applications. Furthermore, their inability to adapt to fluctuating load conditions and system nonlinearities often leads to reduced efficiency and significant degradation in control accuracy (Bhattacharya et al., 2026)

B. Particle Swarm Optimization based PI

This optimization technique is very popular for PI tuning due to simplicity and flexibility. This technique is inspired by the movement of birds or fish, which is used to automatically find the best PI speed controller gains K_p and K_i values. By facilitating a stochastic exploration of the cost function landscape, the algorithm effectively navigates the non-convex objective space to identify global optimal and values that minimize performance indices like the Integral of Time-weighted Absolute Error.

- Particle Swarm Optimization treats every candidate controller as a particle. For BLDC control, one particle may represent:

$$x_i = [K_{p,\omega}, K_{i,\omega}, K_{p,T}, K_{i,T}] \quad (6)$$

where the first pair tunes the speed loop and the second pair tunes the torque or current loop as in equation (6).

Related Algorithm:

- Define the bounds for K_p and K_i .
- Generate particles with random positions and velocities.
- Simulate the BLDC controller for each particle.
- Calculate the objective function.
- Store each particle's $pbest$ which is the best position previously found by particle i .
- Identify the swarm's $gbest$ which is the best position found by the entire swarm.
- Update velocities.
- Update positions.
- Apply gain and velocity limits.
- Repeat until the stopping condition is reached.
- Return $gbest$ as the optimized gain vector as in equation (7):

$$gbest = \text{argmin}_i(pbest, i)$$

(7)

C. Firefly Algorithm-Based PI Controller

This nature-inspired metaheuristic optimizes controller parameters by simulating the flashing behavior of fireflies, where attraction is proportional to the brightness associated with superior system performance. The algorithm iteratively updates the PI parameters by mimicking the movement of less bright fireflies toward those with higher objective function values, effectively balancing local exploitation and global exploration. This technique addresses the limitations of conventional methods by continuously adjusting gain values to minimize time-domain performance criteria, such as overshoot and settling time, during BLDC motor operation. The firefly-based optimization framework effectively minimizes these performance criteria, facilitating a robust control strategy suitable for demanding electric vehicle propulsion systems. By dynamically adjusting the controller gains to compensate for varying torque demands, this approach enhances overall energy efficiency and tracking accuracy across the typical operating cycles of electric vehicles [22].

Related Algorithm:

- Generate random fireflies within the K_p and K_i bounds.
- Simulate the BLDC controller for each firefly.
- Calculate each objective value J_i .
- Rank fireflies according to brightness.
- Move dimmer fireflies toward brighter fireflies.
- Add the random exploration term.
- Apply gain limits.
- Repeat until the iteration or evaluation limit is reached.
- Select the firefly with the lowest J .
- The best firefly is defined as in equation (8):

$$x^* = \arg \min_{x_i} J(x_i) \quad (8)$$

D. Cuckoo Search Algorithm

The Cuckoo Search Algorithm optimizes the PI gains through stochastic exploration of the controller-parameter search space, enabling effective identification of and values for improved BLDC speed tracking and disturbance rejection. Inspired by the brood parasitism of cuckoo species, this approach leverages Levy flights to maintain population diversity, thereby preventing the algorithm from becoming trapped in local minima during the optimization process. This robust mechanism for population diversity provides a

significant advantage over traditional leader-driven strategies, which often struggle with premature convergence in highly complex or constrained search landscapes.

Related Algorithm Cuckoo Search Algorithm is a population-based optimization method. Each **nest** represents one candidate solution.

For BLDC PI tuning, a candidate can contain the controller gains:

$$x_i = [K_{p,\omega}, K_{i,\omega}, K_{p,T}, K_{i,T}]$$

(9)

where the first two gains belong to the speed loop and the last two to the torque or current loop as in equation (9). The main search equation (10) is:

$$x_i^{t+1} = x_i^t + \alpha \otimes \text{Levy}(\lambda)$$

(10)

x_i^t is the current nest.

α is the step-size factor.

$\otimes$ denotes element-wise multiplication. $\text{Levy}(\lambda)$ is a Lévy-distributed random step.

The original Cuckoo Search formulation uses Lévy flights to generate both short local movements and occasional long-distance movements. This gives the algorithm of local exploitation and global exploration capability.

- Define the gain bounds for K_p and K_i .
- Generate an initial population of nests.
- Simulate the BLDC motor for every nest.
- Calculate the objective function J .
- Select a nest and generate a new solution using a Lévy flight.
- Replace the selected nest if the new solution has lower J .
- Abandon a fraction p_a of the worst nests.
- Keep the best nest found so far.
- Repeat until the maximum number of evaluations or iterations is reached.
- Return the best gain vector as given in equation 11:

$$x^* = \arg \min_{x_i} J(x_i)$$

[11]

3. Results and Discussion

In this section, the performance of the BLDC motor drive is evaluated using the four different PI tuning techniques. Conventional PI, PSO-PI, CSA-PI, and FA-PI. All four methods are tested on the vector-controlled BLDC motor drive, which is modeled in MATLAB/Simulink, and the motor specifications are shown in Table 1. This paper aims to identify the most effective tuning method focuses on key performance indicators such as speed tracking capability, speed overshoot, settling time, and transient response during Disturbance. To ensure the practical assessment, the speed controllers are analyzed under three different operating scenarios that represent real-world EV conditions. The responses obtained from each case are analyzed and discussed in detail to provide a clear comparison of the conventional PI tuning method.

Case 1: Startup condition

In the first scenario, the start-up condition of the BLDC drive is considered, where the reference speed is increased from 0 to 1500 rpm with starting torque of 1 N.m. This condition represents the initial movement of an EV from standstill. The performance of speed controller is compared based on speed overshoot, settling time and speed error and speed ripple. The comparison of the speed response under this condition helps in understanding how quickly and accurately each method can achieve the reference speed. Fig 3 shows the speed response of the BLDC motor drive during startup for all four tuning approach. From Table 2, it can be seen that the maximum speed overshoot of 1.2% is recorded in the conventional PI-based system, while the minimum of 0.53% is observed with the CSA-PI controller. The analysis shows that all optimization techniques provide better results than the conventional PI controller, with CSA-PI giving lower speed ripple, speed error, and rise time. For settling time, a speed band of $\pm 0.1\%$ of the reference speed is considered, and all four controllers settle in nearly the same time. Overall, the CSA-PI controller provides better performance in this case.

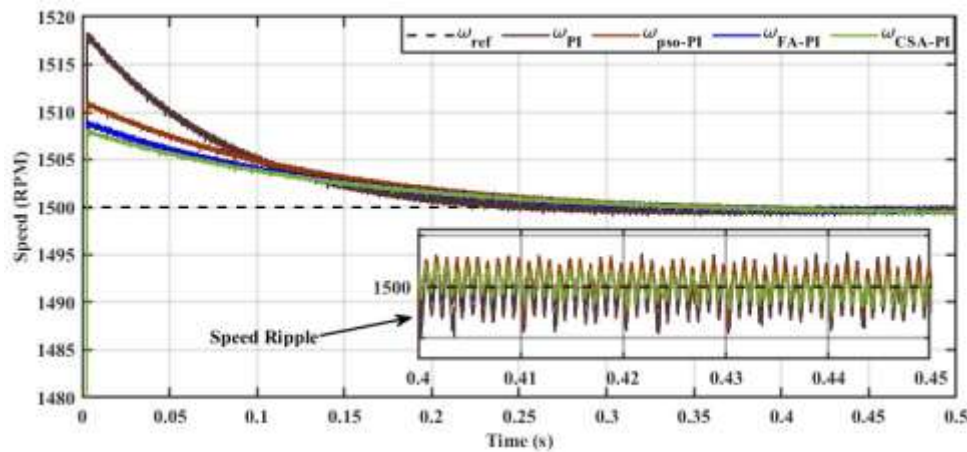


Fig 3: Speed response of the motor at startup from standstill

Table 2: Comparison of Speed response for different PI controller

Method	Speed overshoot	Peak Time	Speed Ripple (%)	Speed Error (%)
Conventional PI	1.2 %	2.9ms	0.02	0.012
PSO-PI	0.6 %	2.75ms	0.046	0.02
FA-PI	0.7 %	2.75ms	0.026	0.013
CSA-PI	0.53 %	2.65ms	0.063	0.028

Case 2: Urban driving Cycles test

In the second scenario, the city environment condition for Ev is tested where the Vehicle frequently experiences speed changes. So to analyze this case reference speed varies in a sequence that represents real traffic conditions, including acceleration, deceleration, and short stop conditions.

The Driving cycle of 0, 750, 1500, 500, 0 rpm are used to examine the performance of all the tuned methods, and the speed responses are shown in Fig 4. From the figure, it can be observed that the conventional PI gives more speed overshoot and more speed drop during the speed transition, whereas the CSA-PI controller has less overshoot and speed drop. The other two optimization techniques also give better responses than the conventional one. This case helps in evaluating how well each PI tuning method can track the speed, handle quick transitions, and maintain stability during frequent speed variation. It is particularly important for urban areas, where a smooth and responsive speed controller is essential for passenger comfort and energy efficiency.

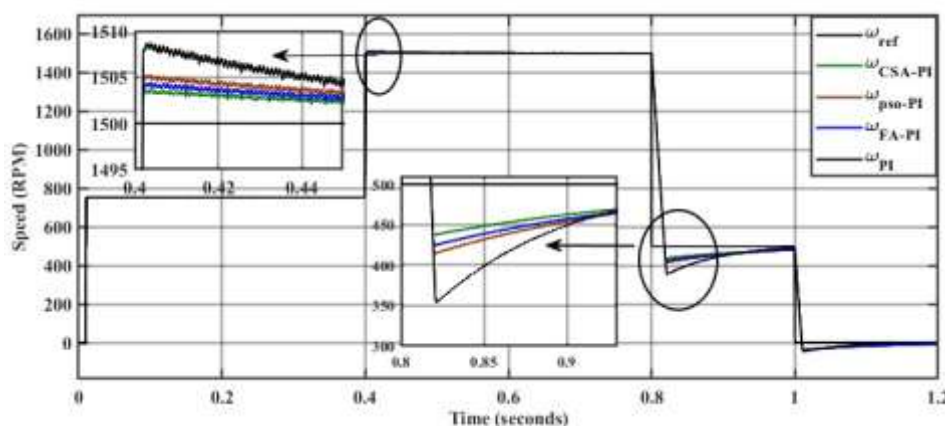


Fig 4: Speed response on City environment condition

Case 3: Hill climbing condition

In this case, an uphill road condition is simulated by gradually increasing the Load torque to experience the motor real time operation. Here the load torques of 2, 3 and 4 (N.m) as shown in Fig.5 is applied to examine the controller performance. As the Vehicle climbs, the required torque increase and the speed tends to drop if the speed controller cannot provide sufficient Signal. Therefore, a proper tuned PI speed controller is required to maintain the speed closer to the reference value and recover faster from the disturbance. The speed response from all four PI tuned method are shown in Fig. 6 from their it can be observed that the conventional PI based motor drive give a high speed dip percentage of 0.53% whereas an optimization-based PI tuning method give the better results. Particularly FA-PI controllers show the least speed drop of 0.2% as shown in Table 3. This behavior is important for electric vehicles, as hill climbing requires high torque and stable speed regulation.

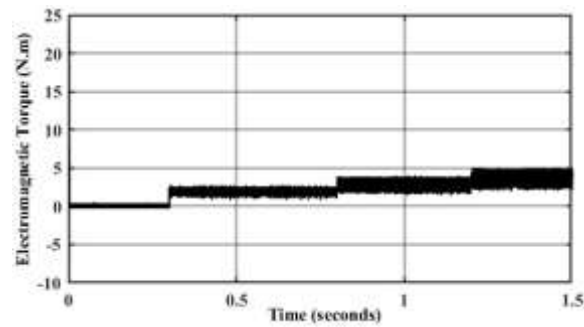


Fig. 5: Load Torque response to simulates uphill road condition

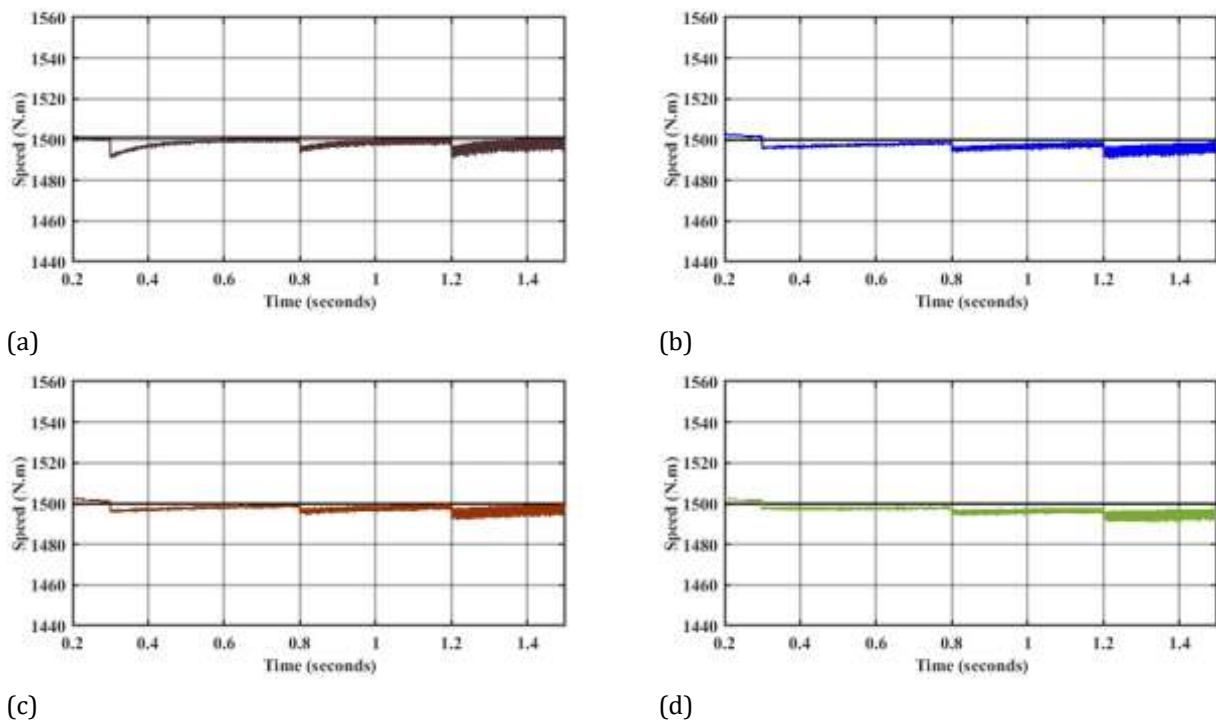


Fig. 6: Hill-Climbing Speed Response at Different Torque Levels using (a) convention PI (b) PSO-PI (c) FA-PI and (d) CAS-PI

Table 3: Speed Response when load Torque is applied to Simulates the uphill road condition

Speed Dip (%) at	Conventional PI	PSO-PI	FA-PI	CSA-PI
2 N.m load Torque	0.53%	0.26 %	0.2 %	0.2 %
3 N.m load Torque	0.33 %	0.26 %	0.2 %	0.23 %
4 N.m load Torque	0.26 %	0.26 %	0.2 %	0.26 %

4. Conclusion

The performance of a vector-controlled BLDC motor drive for electric-vehicle (EV) applications was investigated using four PI-tuning approaches: conventional PI, PSO-PI, CSA-PI, and FA-PI. The controllers were evaluated under three representative EV operating conditions—vehicle startup, urban driving with frequent speed variations, and hill climbing with increasing load torque. The results demonstrate that optimization-based PI tuning can improve speed tracking, transient response, and disturbance rejection compared with conventional PI tuning. During startup, the CSA-PI controller reduced the maximum speed overshoot to 0.53% compared with 1.2% for the conventional PI controller, while also providing lower speed error and ripple. Under the urban driving cycle, the optimization-based controllers maintained closer speed tracking during repeated acceleration and deceleration, with CSA-PI showing comparatively lower overshoot and speed deviation. During hill climbing, the FA-PI controller achieved the lowest speed drop of 0.20%, compared with 0.53% for the conventional PI controller, demonstrating improved speed regulation under increased load torque.

These findings are particularly relevant to BLDC-motor-based electric vehicles, where rapid acceleration, frequent speed changes, regenerative or decelerating operation, and increased load demand during hill

climbing require accurate and robust speed control. The results show that CSA-PI is particularly effective for startup and variable-speed tracking, whereas FA-PI provides improved disturbance rejection during hill-climbing conditions. PSO-PI also provides improved dynamic performance compared with conventional PI tuning. Thus, the study demonstrates that metaheuristic optimization can provide a practical approach for tuning PI controllers in BLDC-based EV propulsion systems. However, the results also indicate that the effectiveness of an optimization technique is operating-condition dependent; therefore, future work can focus on multi-objective or hybrid optimization techniques that simultaneously consider speed tracking, torque ripple, energy consumption, and controller computational effort to achieve robust performance over the complete EV driving cycle.

Reference

1. Jumani, T. A., Mustafa, M. W., Alghamdi, A. S., Rasid, M. M., Alamgir, A., & Awan, A. B. (2020). Swarm Intelligence-Based Optimization Techniques for Dynamic Response and Power Quality Enhancement of AC Microgrids: A Comprehensive Review. *IEEE Access*, 8, 75986–76001. <https://doi.org/10.1109/access.2020.2989133>
2. Enhancing load frequency control and automatic voltage regulation in Interconnected power systems using the Walrus optimization algorithm, Dev, Bhatt, Mondal, Kumar, Kumar, Bajaj, Tuka, Scientific Reports•2024
3. H. Ibrahim, F. Hassan, and A. O. Shomer, "Optimal PID control of a brushless DC motor using PSO and BF techniques," *Ain Shams Engineering Journal*, vol. 5, no. 2, pp. 391–398, 2014, doi: 10.1016/j.asej.2013.09.013.
4. R. Chaudhary, P. Verma, R. Salgotra, et al., "Five Decades of Genetic Algorithms: A Systematic and Bibliometric Review (1975–2025)," *Archives of Computational Methods in Engineering*, 2026, doi: 10.1007/s11831-025-10487-2.
5. S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014, doi: 10.1016/j.advensoft.2013.12.007.
6. M. H. Aljarah, M. Mafarja, A. M. Abualigah, and S. Mirjalili, "A comprehensive survey: Whale Optimization Algorithm and its applications," *Swarm and Evolutionary Computation*, vol. 48, pp. 1–24, 2019, doi: 10.1016/j.swevo.2019.03.004.
7. C. M. Rahman, T. A. Rashid, A. Alsadoon, N. Bacanin, P. Fattah, and S. Mirjalili, "A survey on dragonfly algorithm and its applications in engineering," *Evolutionary Intelligence*, vol. 16, no. 1, pp. 1–21, 2023, doi: 10.1007/s12065-021-00659-x.
8. B. Ur Rehman, K. U. Bangash, M. Arshad, H. Shahid, and S. Shoaib, "Swarm Intelligence for Optimization Problems: A Focus on Whale Optimization Algorithm and Grey Wolf Optimization in Engineering Applications," *IntechOpen*, 2026, doi: 10.5772/intechopen.1014135
9. E. Ikpe and M. F. A. T. Tonmoy, "Review: Evolution of Nature-Inspired Metaheuristic Algorithms: Developments, Trends, and Applications from 1990 to 2025," *Metaheuristic Algorithms with Applications*, 2026, doi: 10.48313/maa.vi.89.
10. K. Dao and T.-T. Nguyen, "A review of the bat algorithm and its varieties for industrial applications," *Journal of Intelligent Manufacturing*, vol. 36, no. 8, pp. 5327–5349, 2025, doi: 10.1007/s10845-024-02506-z.
11. H. Wang and Z. Zhang, "Dragonfly visual evolutionary neural network: A novel bionic optimizer with related LSGO and engineering design optimization," *iScience*, vol. 27, no. 3, Art. 109040, 2024, doi: 10.1016/j.isci.2024.109040.
12. M. A. Ibrahim, A. Kh. Mahmood, and N. S. Sultan, "Optimal PID controller of a brushless DC motor using genetic algorithm," *International Journal of Power Electronics and Drive Systems*, vol. 10, no. 2, pp. 822–830, 2019, doi: 10.11591/ijpeds.v10.i2.pp822-830.
13. W. Xie, "PI controller of speed regulation of brushless DC motor based on particle swarm optimization algorithm with improved inertia weights," *Mathematical Problems in Engineering*, 2019, Art. no. 2671792, doi: 10.1155/2019/2671792.
14. R. Sandip, K. Nagarajan, and T. G. Anbu Mani, "PSO based PI controller for BLDC motor," in *Proc. International Conference on Smart Systems and Sustainable Technologies (ICSSS)*, 2022, doi: 10.1109/ICSSS54381.2022.9782223.
15. J. Shi et al., "Optimizing BLDC motor drive performance using particle swarm algorithm-tuned fuzzy logic controller," *SN Applied Sciences*, vol. 4, Art. no. 293, 2022, doi: 10.1007/s42452-022-05179-6.
16. D. Puangdownreong et al., "Optimal design of I-PD controller for DC motor speed control system by Cuckoo Search," *Procedia Computer Science*, vol. 86, pp. 83–86, 2016, doi: 10.1016/j.procs.2016.05.021.
17. A. Djerioui, A. Houari, M. Ait-Ahmed, M.-F. Benkhoris, A. Chouder, and M. Machmoum, "Grey Wolf based control for speed ripple reduction at low speed operation of PMSM drives," *ISA Transactions*, vol. 74, pp. 111–119, 2018, doi: 10.1016/j.isatra.2018.01.012.
18. R. Pilla et al., "Tuning of extended Kalman filter using grey wolf optimisation for speed control of PMSM drive," *International Journal of Automation and Control*, vol. 15, nos. 4–5, pp. 563–584, 2021, doi: 10.1504/IJAAC.2021.116423.

19. S. Bazi, R. Benzid, Y. Bazi, and M. M. A. Rahhal, "A fast firefly algorithm for function optimization: Application to the control of BLDC motor," *Sensors*, vol. 21, no. 16, Art. no. 5267, 2021, doi: 10.3390/s21165267.
20. S. Banerjee, S. S. Kumar, and A. Alam, "Whale Optimization Algorithm (WOA) based speed control of BLDC motor," in *Proc. International Conference on Sustainable Energy and Future Electric Transportation (SeFeT)*, 2022, doi: 10.1109/SeFeT55524.2022.9909419.
21. Anshory, D. Hadidjaja, and R. B. Jakaria, "BLDC motor: Modeling and optimization speed control using Firefly algorithm," *Dinamik*, vol. 25, no. 2, pp. 51–58, 2020, doi: 10.35315/dinamik.v25i2.7851.
22. D. Bhattacharya, R. Gandhi, C. Kumar, K. S. Sherpa, and R. Roy, "Modified firefly-optimized PI controller for BLDC motor performance under New European Driving Cycle conditions," *International Journal of Power Electronics and Drive Systems*, vol. 17, no. 1, pp. 140–154, 2026, doi: 10.11591/ijpeds.v17.i1.pp140-154.
23. U. Das and P. K. Biswas, "Relative study of classical and fuzzy logic controllers in closed loop BLDC motor drive with GA and PSO optimization techniques," *Journal of Applied Research and Technology*, vol. 19, no. 4, pp. 379–402, 2021, doi: 10.22201/icat.24486736e.2021.19.4.1039.
24. U. Govardhane, P. Mule, S. D. Lokhande, S. Jadhav, D. Tagare, and R. Munje, "PID controllers for BLDC motor: A comparative study," *Zenodo*, 2023, doi: 10.5281/zenodo.8050601.
25. K. S. Kavin, S. K. P., A. Pathak, T. R. Premila, R. Hemalatha, and T. Kumar, "Modelling and analysis of hybrid fuzzy tuned PI controller based PMBLDC motor for electric vehicle applications," *International Journal of Electrical and Electronics Engineering*, vol. 10, no. 2, pp. 8–18, 2023, doi: 10.14445/23488379/ijeee-v10i2p102.
26. Shary, D. K., Nekad, H. J., and Alawan, M. A., "Speed control of brushless DC motors using (conventional, heuristic, and intelligent) methods-based PID controllers," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 30, no. 3, pp. 1359–1368, Jun. 2023, doi: 10.11591/ijeecs.v30.i3.pp1359-1368.
27. U. Das and P. K. Biswas, "Relative study of classical and fuzzy logic controllers in a closed-loop BLDC motor drive with the GA and PSO optimization technique," *Journal of Applied Research and Technology*, vol. 19, no. 4, pp. 379–402, 2021, doi: 10.22201/icat.24486736e.2021.19.4.1039.
28. H. Sarma and A. Bardalai, "An intelligent PID controller tuning for speed control of BLDC motor using driving training-based optimization," *International Journal of Power Electronics and Drive Systems*, vol. 14, no. 4, pp. 2474–2486, Dec. 2023, doi: 10.11591/ijpeds.v14.i4.pp2474-2486.
29. E. H. E. Bayoumi and H. M. Soliman, "PID/PI tuning for minimal overshoot of permanent-magnet brushless DC motor drive using particle swarm optimization," *Electromotion*, vol. 14, no. 4, pp. 198–208, 2007.
30. M. A. Awadallah, E. H. E. Bayoumi, and H. M. Soliman, "Adaptive deadbeat controllers for brushless DC drives using PSO and ANFIS techniques," *Journal of Electrical Engineering*, vol. 60, no. 1, pp. 3–11, 2009.
31. Y. Anagreh, M. B. Fayyad, and A. Anagreh, "Particle swarm optimization based high performance four switch BLDC motor drive," *International Journal of Power Electronics and Drive Systems*, vol. 13, no. 2, pp. 825–834, 2022, doi: 10.11591/ijpeds.v13.i2.pp825-834.
32. C. Kiree, D. Kumpanya, S. Tunyasirirut, and D. Puangdownreong, "PSO-Based Optimal PI(D) Controller Design for Brushless DC Motor Speed Control with Back EMF Detection," *Journal of Electrical Engineering & Technology*, vol. 11, no. 3, pp. 715–723, 2016, doi: 10.5370/JEET.2016.11.3.715.
33. S. Bazi, R. Benzid, Y. Bazi, and M. M. Al Rahhal, "A Fast Firefly Algorithm for Function Optimization: Application to the Control of BLDC Motor," *Sensors*, vol. 21, no. 16, Art. no. 5267, 2021, doi: 10.3390/s21165267.
34. D. Kumpanya, "Optimal PID controller design for BLDC motor speed control system with input tracking and load regulating constrained optimization by Cuckoo Search," *RMUTP Research Journal: Sciences and Technology*, vol. 12, no. 2, pp. 123–137, 2018, doi: 10.14456/jrmutp.2018.28.
35. P. Arul, M. H. Ananda, D. Selvaraj, and R. Dhanalakshmi, "BLDC Motor Control for EVs Using Cuckoo Search Algorithm in PI Controller Tuning," in *Proc. 1st International Conference on Intelligent and Sustainable Power and Energy Systems (ISPES)*, pp. 80–83, 2023, doi: 10.5220/0012508600003808.
36. V. Geetha and S. Thangavel, "Performance Analysis of Direct Torque Controlled BLDC Motor Using Fuzzy Logic," *International Journal of Power Electronics and Drive Systems*, vol. 7, no. 1, pp. 144–151, 2016, doi: 10.11591/ijpeds.v7.i1.pp144-151.