Erciyes University Enginoscope Journal

Research Article
10.28949/[doiprefix].1974477

A Multi-Stage Search-Based PI Tuning Strategy for BLDC Motor Speed Control under Disturbance Scenarios

Imprint:
Select: 2(2)
Year: 2026
Page: 143-159

Ahu ERGÜN SAYDAM[a]
Handan GÜRSOY-DEMİR[b*]

[a] Master's Student, Iskenderun Technical University,
ahusaydam.lee23@iste.edu.tr
[b] Asst. Prof. Dr., Iskenderun Technical University,
handan.gursoy@iste.edu.tr

[*] Corresponding Author

Received: 6/19/2026
Accepted: 7/11/2026
Published: 7/30/2026

Citation:
Ahu ERGÜN SAYDAM, Handan GÜRSOY-DEMİR (2026). A Multi-Stage Search-Based PI Tuning Strategy for BLDC Motor Speed Control under Disturbance Scenarios. *Erciyes University Enginoscope Journal*, 2(2), 143-159.

Abstract

In this study, a multi-stage search-based tuning approach is presented for determining PI controller parameters for brushless direct current (BLDC) motor speed control. The performance of PI-based speed control directly depends on the appropriate selection of proportional and integral gains, especially under disturbance conditions. In this context, the PI controller gains were optimized using Grid Search, Random Search, and Local Search methods. A combined cost function including metrics such as IAE, ITAE, control signal magnitude, overshoot, final error, rise time, and settling time was used to evaluate the candidate gain pairs. In the first stage, the low-cost region in the K_p–K_i parameter space was identified using Grid Search, and then candidate solutions were investigated within the determined range using the Random Search method. In the final stage, nominal performance was fine-tuned using the Local Search method around the best candidate solution obtained from Random Search. The obtained Grid-PI, Random-PI, and Local-PI controllers were tested under nominal operating conditions, measurement noise, measurement noise with actuator saturation, and measurement noise with control delay scenarios. Simulation results showed that the lowest total cost under nominal conditions was obtained with the Local-PI controller. In contrast, under the tested disturbance scenarios, Random-PI produced slightly lower cost values and exhibited more balanced performance across the considered cases. These results indicate that PI gains optimized under nominal conditions may not always produce the lowest cost under disturbance conditions. The proposed approach provides an applicable framework for the systematic and comparative determination of PI parameters in BLDC motor speed control.

Keywords: BLDC motor, PI controller, Speed control, Search-based optimization, Robustness analysis

✵ ✵ ✵

Bozucu Senaryolar Altında BLDC Motor Hız Kontrolü için Çok Aşamalı Arama Tabanlı PI Ayarlama Stratejisi

Öz

Bu çalışmada, fırçasız doğru akım motorunun (BLDC) hız kontrolü için PI denetleyici parametrelerinin belirlenmesine yönelik çok aşamalı arama tabanlı bir ayarlama yaklaşımı sunulmaktadır. PI tabanlı hız kontrol başarımı, özellikle bozucu koşullar altında oransal ve integral kazançların uygun seçimine doğrudan bağlıdır. Bu kapsamda PI denetleyici kazançları Izgara (Grid) Arama, Rastgele (Random) Arama ve Yerel (Local) Arama yöntemleri kullanılarak optimize edilmiştir. Aday kazanç çiftlerinin değerlendirilmesinde IAE, ITAE, kontrol işareti büyüklüğü, aşım, son durum hatası, yükselme süresi ve yerleşme süresi gibi ölçütleri içeren birleşik bir maliyet fonksiyonu kullanılmıştır. İlk aşamada Izgara Arama ile K_p–K_i parametre uzayındaki düşük maliyetli bölge belirlenmiş, ardından Rastgele Search yöntemiyle belirlenen aralıkta aday çözümler araştırılmıştır. Son aşamada ise Rastgele Search sonucunda elde edilen en iyi aday çözüm çevresinde Yerel Search yöntemi ile nominal performansın ince ayarı gerçekleştirilmiştir. Elde edilen Grid-PI, Random-PI ve Local-PI denetleyicileri nominal çalışma koşulu, ölçüm gürültüsü, ölçüm gürültüsü ile eyleyici doygunluğu ve ölçüm gürültüsü ile kontrol gecikmesi senaryoları altında test edilmiştir. Simülasyon sonuçları, nominal durumda en düşük toplam maliyetin Local-PI denetleyicisi ile elde edildiğini göstermiştir. Buna karşılık incelenen bozucu senaryolar altında Random-PI bir miktar daha düşük maliyet değerleri üretmiş ve ele alınan durumlar genelinde daha dengeli bir performans sergilemiştir. Bu sonuçlar, nominal koşulda optimize edilen PI kazançlarının bozucu koşullar altında her zaman en düşük maliyeti üretmeyebileceğini göstermektedir. Önerilen yaklaşım, BLDC motor hız kontrolünde PI parametrelerinin sistematik ve karşılaştırmalı biçimde belirlenmesi için uygulanabilir bir çerçeve sunmaktadır.

Anahtar kelimeler: BLDC motor, PI denetleyici, Hız kontrolü, Arama tabanlı optimizasyon, Sağlamlık analizi

1. INTRODUCTION

Brushless direct current (BLDC) motors are widely used in a broad range of applications, from electric vehicles, robotic systems, and unmanned aerial vehicles to industrial servo drives, owing to their high efficiency, high power density, low maintenance requirements, wide speed operating range, and electronic commutation structure [1]–[3]. However, speed control of BLDC motors cannot be regarded solely as a linear control problem similar to that of conventional direct current motors. Trapezoidal back electromotive force, three-phase inverter switching, rotor-position-dependent commutation structure, current ripples, load torque variations, and drive limitations are important factors that directly affect closed-loop speed control performance [1], [2]. Therefore, controller design in BLDC drive systems requires a systematic evaluation that considers both modeling accuracy and robustness under practical operating conditions. Similarly, recent studies on electromechanical and energy-related systems have also emphasized the importance of experimental performance evaluation, efficiency-oriented design, and reliable operation under practical conditions [4], [5].

In industrial applications, PI and PID controllers still remain among the most widely used controller structures due to their simple structures, low computational cost, ease of implementation, and ability to provide satisfactory performance in many practical systems [6]-[8]. In BLDC motor speed control, PI/PID-based approaches have also been investigated in various forms, including classical tuning rules, fuzzy-logic-supported structures, heuristic optimization algorithms, and adaptive control strategies [9]–[15]. However, a considerable portion of these studies either reduces controller gain selection to a single optimum point that minimizes a specific performance index or focuses only on the speed response under nominal operating conditions. This may limit the reliability of the obtained PI gains, particularly under practical disturbance effects such as measurement noise, actuator saturation, control delay, or drive limitations [16], [17].

Although the PI gain tuning problem is generally formulated as an optimization problem, in practice it is also a parameter-space exploration problem. This is because more than one K_p–K_i pair may yield similar performance under nominal operating conditions. In contrast, these gain pairs may exhibit different dynamic behaviors under measurement noise, actuator saturation, or control delay. Therefore, reporting only the single solution that provides the lowest nominal cost value is not sufficient for a comprehensive understanding of controller behavior. Systematic investigation of the parameter space, identification of low-cost regions, and comparison of the obtained candidate PI controllers under disturbance scenarios can provide a more reliable tuning process.

In this context, Grid Search, Random Search, and Local Search methods have complementary characteristics. Grid Search enables the predefined parameter space to be scanned in a regular and reproducible manner, whereas Random Search can contribute to finding suitable candidate solutions through more flexible sampling within the same range [18]. Local Search, on the other hand, can provide fine tuning of nominal performance by performing a more precise improvement around a previously identified promising candidate solution [19]. Therefore, comparing these methods on the same BLDC closed-loop model using common performance criteria makes it possible to address the PI tuning problem not only as a single optimization outcome but also as an explainable and comparative parameter evaluation problem.

In this study, a multi-stage search-based strategy is presented for PI controller tuning in BLDC motor speed control. The proposed approach consists of three main stages. In the first stage, the PI controller gains are systematically scanned within the K_p–K_i range determined by the Grid Search method, and the low-cost region is identified. In the second stage, candidate solutions are explored in the same parameter region using the Random Search method. In the third stage, local improvement is carried out using the Local Search method based on a suitable initial point obtained from the Random Search results. The resulting Grid-PI, Random-PI, and Local-PI controllers are then evaluated under nominal operating conditions as well as under measurement noise, measurement noise with actuator saturation, and measurement noise with control delay scenarios.

To clarify the novelty of the present work, the proposed approach differs from conventional PI tuning studies in two main aspects. First, the PI tuning problem is not treated only as the search for a single optimum gain pair; instead, the low-cost region of the K_p–K_i parameter space is systematically examined. Second, the obtained PI controllers are not evaluated only under nominal operating conditions, but also under measurement noise, actuator saturation, and control delay scenarios. Therefore, the study provides both a parameter-space-based tuning perspective and a scenario-based robustness comparison for BLDC motor speed control.

The main contributions of this study can be summarized as follows:

• A multi-stage search-based PI tuning framework is proposed for BLDC motor speed control by sequentially using Grid Search, Random Search, and Local Search on the same closed-loop model.

• The PI tuning problem is addressed not only as a single optimum search problem but also as a parameter-space evaluation problem by identifying and interpreting the low-cost region in the K_p–K_i plane.

• The obtained Grid-PI, Random-PI, and Local-PI controllers are comparatively evaluated under nominal operation and three disturbance scenarios, namely measurement noise, measurement noise with actuator saturation, and measurement noise with control delay.

• The results show that the PI gains providing the best nominal performance may not always produce the lowest-cost solution under disturbance scenarios. Thus, the need to consider nominal performance and robustness evaluation together in the PI tuning process is demonstrated.

The remainder of this paper is organized as follows. Section 2 presents the BLDC motor model, closed-loop drive structure, and PI controller formulation. Section 3 explains the proposed multi-stage search-based PI parameter tuning strategy, performance metrics, and total cost function. Section 4 presents the scenario-based robustness evaluation. Section 5 provides the simulation results, method comparisons, and speed response analyses. Finally, the last section summarizes the findings and discusses future research directions.

2. MATHEMATICAL MODEL OF THE BLDC MOTOR AND PI CONTROLLER STRUCTURE

2.1. Mathematical Model Of The BLDC Motor

The system used in this study consists of a three-phase brushless direct current motor driven by a PI controller, an inverter, a commutation logic unit, and a speed feedback block [2]. The BLDC motor is modeled as a star-connected three-phase structure, and the back electromotive force is defined with an ideal trapezoidal profile. The relationship between the electrical angle and the mechanical angle is given in Equation (1).

$$\theta_e(t) = p\theta_m(t) \tag{1}$$

In this equation, p denotes the number of pole pairs, θ_m represents the mechanical rotor angle, and θ_e denotes the electrical angle. The three-phase electrical voltage equations of the BLDC motor are given in Equations (2)–(4).

$$v_a = R_s i_a + L_s \frac{di_a}{dt} + e_a, \tag{2}$$

$$v_b = R_s i_b + L_s \frac{di_b}{dt} + e_b, \tag{3}$$

$$v_c = R_s i_c + L_s \frac{di_c}{dt} + e_c \tag{4}$$

Here, $v_a,\ v_b,\ v_c$ denote the phase voltages; , $i_a,\ i_b,\ i_c$ denote the phase currents; R_s is the stator resistance per phase; L_s is the stator phase inductance; and $e_a,\ e_b,\ e_c$ are the back electromotive forces of the corresponding phases. The trapezoidal back-EMF structure is expressed in Equation (5).

$$e_k = K_e f_k(\theta_e)\omega_m, \qquad k \in a, b, c \tag{5}$$

In the above expression, K_e is the back-EMF constant, $f_k(\theta_e)$ is the trapezoidal shape function of the corresponding phase, and ω_m is the rotor angular speed. The electromagnetic torque is obtained through the interaction of the phase currents and the trapezoidal back-EMF shape functions, as given in Equation (6).

$$T_e = K_t[f_a(\theta_e)i_a + f_b(\theta_e)i_b + f_c(\theta_e)i_c] \tag{6}$$

The mechanical dynamics of the motor are given in Equation (7).

$$J\frac{d\omega_m}{dt} = T_e - T_L - B\omega_m \tag{7}$$

Here, J denotes the rotor inertia, B is the viscous friction coefficient, and T_L is the load torque. In the phase voltage equations, the equivalent phase inductance L_s is used for simplified representation. In the simulation model, however, the d-axis, q-axis, and zero-sequence inductance values of the BLDC motor are defined separately. The main BLDC motor parameters used in the simulation model are given in Table 1.

Table 1. Main BLDC motor parameters used in the simulation model

Parameter	Symbol	Value	Unit
Back-EMF profile	—	Ideal trapezoidal	—
Maximum permanent magnet flux linkage	λ_m	0.5513	Wb
Back-EMF flat-top angle range	—	120	deg
Number of pole pairs	p	1	—
Winding type	—	Star	—
Stator d-axis inductance	L_d	0.00022	H
Stator q-axis inductance	L_q	0.00022	H
Zero-sequence inductance	L_0	0.00016	H
Stator resistance per phase	R_s	0.013	Ω
Rotor inertia	J	0.0001	kg·m^2
Rotor damping coefficient	B	0.2	N·m/(rad/s)

2.2. PI Controller Structure

The speed control of the BLDC motor was performed using a discrete-time PI controller. The error signal between the reference speed and the measured motor speed is defined in Equation (8).

$$e[k] = \omega_{ref}[k] - \omega_m[k] \tag{8}$$

The discrete-time control law of the PI controller is given in Equation (9).

$$u[k] = K_p e[k] + K_i T_s \sum_{n=0}^{k} e[n] \tag{9}$$

In this equation, K_p denotes the proportional gain, K_i denotes the integral gain, and T_s represents the sampling time. The proportional term provides a rapid response to the speed error, whereas the integral term aims to reduce the steady-state error. The controller output was passed through a saturation block to represent the physical voltage limits of the inverter. The saturation operation is given in Equation (10).

$$u_s[k] = sat(u[k], u_{min}, u_{max}) \tag{10}$$

Here, $u_s[k]$ denotes the control signal after saturation, while u_{min} and u_{max} represent the lower and upper control limits, respectively. Owing to the structure given in Equation (10), not only the nominal speed tracking performance of the PI controller but also its behavior under actuator saturation can be evaluated.

3. PROPOSED SEARCH-BASED PI PARAMETER TUNING METHOD

In this study, the K_p and K_i gains of the PI controller used for BLDC motor speed control were determined through a simulation-based offline search evaluation approach. The main objective of the proposed method is not only to obtain a single optimum gain pair but also to systematically investigate the effects of different parameter regions on speed tracking performance. In this context, candidate PI parameters were generated using Grid Search, Random Search, and Local Search methods; each candidate solution was simulated on the same BLDC closed-loop model and evaluated using common performance criteria.

3.1. PI Parameter Search Space

The parameters to be tuned in the PI controller are defined as the proportional gain K_p and the integral gain K_i. The general search space is given in Equation (11).

$$\Omega = \left(K_p, K_i\right); K_{p,min} \leq K_p \leq K_{p,max};\ K_{i,min} \leq K_i \leq K_{i,max} \tag{11}$$

In the above expression, Ω denotes the parameter space containing all candidate PI gain pairs. The search bounds were selected based on preliminary closed-loop simulation trials. The intervals were limited to the region where the BLDC speed control system provided stable tracking, acceptable overshoot, and reasonable control effort, rather than being derived from an analytical stability analysis or a conventional tuning rule.

In this study, the intervals $K_p \in [0.05, 0.18]$ and $K_i \in [4.2, 5.8]$ were used for Grid Search. In the Grid Search stage, both parameter intervals were scanned using 10 equally spaced sampling points, and a total of 100 candidate PI gain pairs were evaluated. In the Random Search stage, 100 random candidate solutions were generated within the same parameter range. In the Local Search stage, the best candidate solution obtained from Random Search was used as the initial point, and a more precise neighborhood search was performed around this point.

3.2. Grid Search Approach

The Grid Search method is based on systematically scanning the defined K_p–K_i search space over a regular grid structure. In this method, the candidate parameters were generated as given in Equations (12) and (13).

$$K_p = K_{p,min} + m\Delta K_p \tag{12}$$

$$K_i = K_{i,min} + n\Delta K_i \tag{13}$$

Here, ΔK_p and ΔK_i denote the sampling steps of the corresponding parameters, while m and n represent the scanning indices. The main function of the Grid Search approach in this study is not only to determine the PI parameter pair that yields the lowest cost but also to make the performance distribution in the K_p–K_i plane visible. Therefore, the Grid Search results were visualized in the form of a cost map and a three-dimensional performance surface. In this way, it was possible to systematically examine which parameter regions produced low cost and which regions resulted in high error, overshoot, or control effort.

3.3. Random Search Approach

In the Random Search method, candidate PI gain pairs were randomly sampled from the defined search space. This process is expressed in Equation (14).

$$\left(K_p^{(j)}, K_i^{(j)}\right) \sim U(\Omega), \qquad j = 1,2,\dots,N_r \tag{14}$$

In this formulation, $U(\Omega)$ denotes the uniform distribution defined over the search space, and N_r represents the number of random trials. In this study, N_r=*100* was selected. For reproducibility, the random sampling process was performed using a fixed random seed, so that the same candidate set and the reported Random-PI gain pair can be regenerated under the same simulation settings. Random Search enables a more flexible sampling of the search space, particularly in cases where the location of the suitable parameter region is not known precisely in advance.

At this stage, the speed response, tracking error, and control voltage of the BLDC motor were recorded for each random K_p–K_i pair. The candidate solutions were then ranked according to the performance metrics and the total cost function. The best candidate parameter pair obtained from Random Search was also simulated again, and the speed response, error signal, and control voltage plots were generated.

3.4. Local Search Approach

The Local Search method was applied to perform a more precise search around a successful candidate solution obtained from Grid Search or Random Search. This hill-climbing-based Local Search was preferred due to its simplicity, reproducibility, and suitability for the two-parameter PI tuning problem. The best candidate obtained from Random Search was used as the initial point. In the reported simulation, this point was $K_p = 0.0811$ and $K_i = 4.99$ and a neighborhood-based search was performed around this point. The neighborhood steps were defined as $\Delta K_p = 0.003$ and $\Delta K_i = 0.03$. At each iteration, the current candidate solution and its neighboring solutions in both axial and diagonal directions were evaluated. Therefore, the candidate neighborhood was defined in (Equation (15))

$$\mathcal{N}(K_p, K_i) = \{(K_p, K_i), (K_p + \Delta K_p, K_i), (K_p - \Delta K_p, K_i), (K_p, K_i + \Delta K_i), (K_p, K_i - \Delta K_i), (K_p + \Delta K_p, K_i + \Delta K_i), (K_p + \Delta K_p, K_i - \Delta K_i), (K_p - \Delta K_p, K_i + \Delta K_i), (K_p - \Delta K_p, K_i - \Delta K_i)\} \tag{15}$$

At the end of each iteration, the candidate solution with the lowest cost was selected as the new center point. The maximum number of iterations was set to 20 based on preliminary simulation trials. Since the Local Search stage refines only two PI parameters around the best Random Search solution, additional iterations provided negligible improvement in the total cost while increasing the simulation time. This approach enables a more precise improvement around the promising initial point determined by Random Search.

3.5. Performance Metrics

For each candidate PI controller, speed tracking performance was evaluated together with error-based metrics and control signal characteristics. The speed tracking error is defined in Equation (16).

$$e(t) = \omega_{ref}(t) - \omega_m(t) \tag{16}$$

Accordingly, the integral absolute error is calculated using Equation (17),

$$IAE = \int_0^T |e(t)| dt \tag{17}$$

the integral of time-weighted absolute error is calculated using Equation (18),

$$ITAE = \int_0^T t|e(t)| dt \tag{18}$$

The effective value of the control signal is given in Equation (19), and the maximum control amplitude is given in Equation (20).

$$U_{rms} = \sqrt{\frac{1}{N}\sum_{k=1}^{N} u^2[k]} \tag{19}$$

$$U_{max} = max|u[k]| \tag{20}$$

In addition, overshoot percentage, rise time, settling time, and final error were calculated to evaluate the transient response quality. The overshoot percentage is expressed in Equation (21).

$$OS(\%) = max\left(0, \frac{\omega_{max} - \omega_{ref}}{|\omega_{ref}|} \times 100\right) \tag{21}$$

The final error is defined in Equation (22).

$$FinalErr = \left|\omega_{ref}(T) - \omega_m(T)\right| \tag{22}$$

3.6. Total Cost Function

A total cost function that jointly considers error magnitude, transient response quality, control effort, and final value accuracy was used to rank the candidate PI parameters according to a common criterion. The total cost function is given in Equation (23).

$$J = IAE + 0.3ITAE + 0.03U_{rms} + OS + 5FinalErr + T_r + 2.0T_s \tag{23}$$

In this cost function, *IAE* represents the total tracking error, *ITAE* represents the errors persisting at later times, U_{rms} represents the control effort, *OS* represents the overshoot, and *FinalErr* represents the steady-state accuracy. In addition, T_r denotes the rise time and T_s denotes the settling time of the speed response. Thus, longer rise and settling times increase the total cost value. The weighting coefficients were selected according to the engineering priorities of the speed control problem. Higher emphasis was given to tracking accuracy, final error, and settling behavior, while a smaller weight was assigned to U_{rms} to avoid excessive penalization of the control effort. Therefore, the controller rankings are interpreted under this common weighting structure.

In this study, the total cost function was used to rank the candidate solutions. However, the final evaluation was not reduced only to the minimum J value; the distribution of low-cost regions in the K_p–K_i parameter space was also analyzed. This approach extends the PI tuning problem beyond the search for a single optimum and provides an explainable parameter-space evaluation.

4. SCENARIO-BASED ROBUSTNESS EVALUATION

Four different scenarios were created to evaluate the PI controller not only under nominal operating conditions but also under disturbance effects that may be encountered in practical drive systems. These scenarios were used to examine the behavior of the BLDC motor speed control system against effects such as measurement noise, actuator limits, and control delay. In each scenario, the same PI parameters were used, and the obtained speed response, tracking error, control signal, and performance metrics were comparatively evaluated.

- **Nominal Operating Scenario**

In the nominal scenario, the BLDC motor closed-loop speed control system was operated without any additional disturbance effect. This scenario was used as the reference case to determine the basic speed tracking performance of the PI controller under ideal conditions. The results obtained from the nominal model were evaluated as the baseline for comparing the performance losses in the other disturbance scenarios.

- **Measurement Noise Scenario**

In the second scenario, the measurement noise effect was modeled by adding a Band-Limited White Noise block to the speed feedback channel. In the Band-Limited White Noise block, the noise power was set to 10, the sample time was set to 0.1 s, and the seed value was set to 23341. In this case, the measured speed signal applied to the controller is given in Equation (24).

$$y_n(t) = \omega_m(t) + n(t) \tag{24}$$

Here, $\omega_m(t)$ denotes the actual motor speed, and $n(t)$ represents the band-limited white noise added to the measurement channel. This scenario was used to investigate the effect of sensor-induced measurement disturbances on the performance of the PI controller.

- **Noise and Actuator Saturation Scenario**

In the third scenario, actuator saturation was also considered in addition to measurement noise. For this purpose, the control signal generated by the PI controller was passed through a saturation block and limited within the interval (-73, 73) V to represent the physical inverter or voltage limits. This limit was selected based on preliminary simulations to activate the saturation effect during the transient response while maintaining stable speed tracking. Lower saturation limits caused excessive tracking degradation, whereas higher limits made the saturation effect less visible. The saturation operation is given in Equation (25).

$$u_s(t) = sat(u(t), -73, 73) \tag{25}$$

In Equation (25), $u(t)$ denotes the control signal generated by the PI controller, while $u_s(t)$ denotes the control signal after saturation. This scenario was used to evaluate the effect of the physical limits of the inverter output voltage or control signal on speed tracking performance in real drive systems. Since saturation may adversely affect the transient response and settling behavior, especially in PI controllers with integral action, this scenario is important for robustness analysis.

- **Noise and Control Delay Scenario**

In the fourth scenario, a Transport Delay block was added to the PI controller output in addition to measurement noise. Thus, the delayed application of the control signal to the motor was modeled. The delayed control signal is given in Equation (26).

$$u_d(t) = u(t - \tau_d) \tag{26}$$

Here, τ_d denotes the control delay time, which was selected as 0.001 s in the simulation model. This scenario represents the effect of digital processing delays, communication delays, or drive-circuit-related delays on closed-loop speed control performance. Since control delay may adversely affect the phase characteristics of the closed-loop system and cause increases in overshoot, oscillation, and settling time, it provides a critical test condition for examining the sensitivity of the PI controller to delay.

In each scenario, the candidate PI controllers were evaluated using the same performance metrics. In this context, IAE, ITAE, overshoot percentage, final error, rise time, settling time, and the effective value of the control signal were calculated. Thus, not only the nominal performance of the PI parameters but also their consistency under different disturbance effects was analyzed. In the robustness evaluation, the reliability of parameters that provided low cost in the nominal scenario but whose performance significantly deteriorated under noise, saturation, or delay was further examined. In contrast, PI parameters that provided low error, limited overshoot, and acceptable control effort under different scenarios were considered more robust candidates.

5. SIMULATION STUDIES AND RESULTS

In this section, the simulation studies performed for tuning the PI controller parameters for BLDC motor speed control are presented. Within the scope of the study, the PI controller parameters were determined using three different search approaches: Grid Search, Random Search, and Local Search. The obtained best parameters were first evaluated under nominal operating conditions and then compared under disturbance scenarios including measurement noise, measurement noise + actuator saturation, and measurement noise + control delay.

5.1. Simulation Flow

The general simulation flow followed in the study is given in Figure 1. This flow summarizes the entire process, starting from the definition of the BLDC motor model and PI controller structure to the determination of performance metrics, PI parameter optimization, selection of the best control parameters, nominal case analysis, and robustness tests.

In the first stage, the BLDC motor speed control system and PI controller structure were created in the Simulink environment. Then, performance metrics such as IAE, ITAE, control signal magnitude, final error, overshoot, rise time, and settling time were considered to evaluate the control performance. Using these metrics, a combined cost function was defined, and all search methods were evaluated according to the same performance criterion.

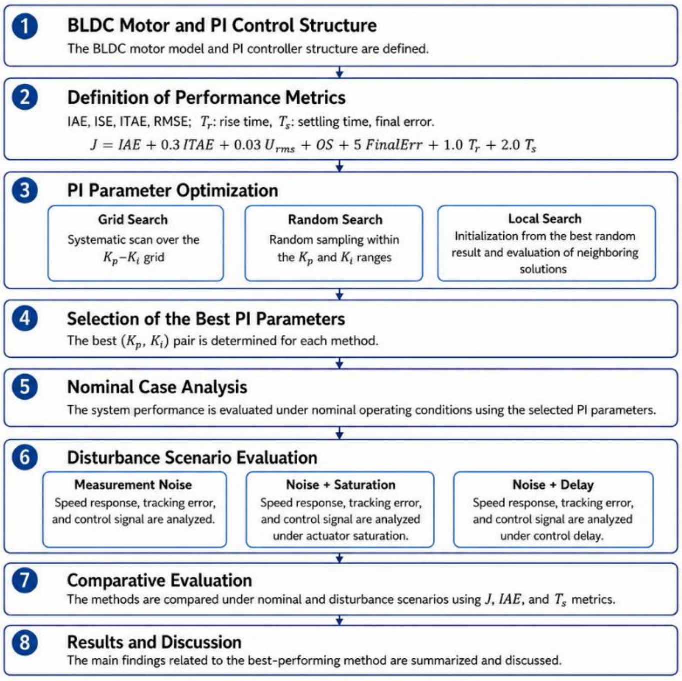

Figure 1. PI parameter optimization and simulation evaluation process.

PI parameter optimization was performed in three stages. First, the K_p–K_i range determined by the Grid Search method was systematically scanned, and the low-cost parameter region was identified. In the second stage, candidate PI parameters were evaluated by random sampling within the same range using the Random Search method. In the final stage, local improvement was performed using the Local Search method, based on the suitable initial point obtained from the Random Search results.

The Grid-PI, Random-PI, and Local-PI controllers obtained as a result of the optimization were tested under nominal and disturbance scenarios. Thus, not only nominal performance but also robustness behavior under effects such as noise, saturation, and delay was evaluated.

5.2. PI Parameter Optimization Results

The best PI parameters obtained from the Grid Search, Random Search, and Local Search methods and the cost function values under nominal conditions are given in Table 2.

Table 2. Best PI parameters and nominal cost values obtained according to the optimization methods.

Method	K_p	K_i	***Nominal (J)***
Grid-PI	0.0788	4.9111	19.965
Random-PI	0.0811	4.9900	20.101
Local-PI	0.0751	4.9300	19.884

As shown in Table 2, all three methods produce PI parameters that are quite close to each other. This indicates that, for the system, there is a low-cost parameter region rather than a single sharp optimum point. Under nominal conditions, the lowest cost value was obtained with the Local-PI controller. However, the Grid-PI and Random-PI results are also very close, and it is observed that all methods provide acceptable performance under nominal conditions.

5.3. Grid Search Results

The cost map and relative cost surface obtained using the Grid Search method is given in Figure 2. These plots show how the cost function changes for different combinations of K_p and K_i parameters.

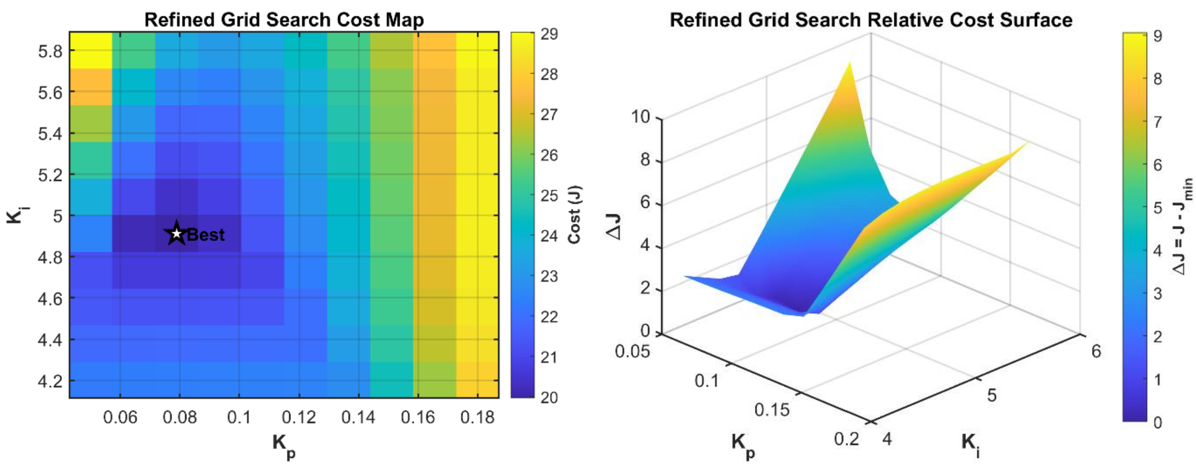

Figure 2. Grid Search results: (a) cost map and (b) relative cost surface.

As clearly observed in Figure 2, the low-cost region is concentrated approximately within the range of $K_p = 0.07 - 0.09$ and $K_i = 4.8 - 5.0$. This result shows that the appropriate operating region in the PI parameter space can be systematically determined using the Grid Search method. The main advantage of the Grid Search method is that it reveals the general distribution of the cost function by regularly scanning the parameter space. In this way, information about the appropriate parameter range was obtained for the subsequent Random Search and Local Search stages.

5.4. Random Search Results

The parameter distribution obtained using the Random Search method is given in Figure 3. In this method, K_p and K_i values were randomly selected within the defined range, and the cost function was calculated for each candidate solution.

As seen in Figure 3, low-cost candidate solutions are concentrated around the low-cost region determined by Grid Search. This indicates that the low-cost region is supported not only by grid-based scanning but also by random sampling. The Random Search method provides a flexible approach, especially in cases where systematically scanning the entire search space is costly. In this study, Random Search produced a performance close to Grid Search under nominal conditions and provided slightly lower cost values under the considered disturbance scenarios.

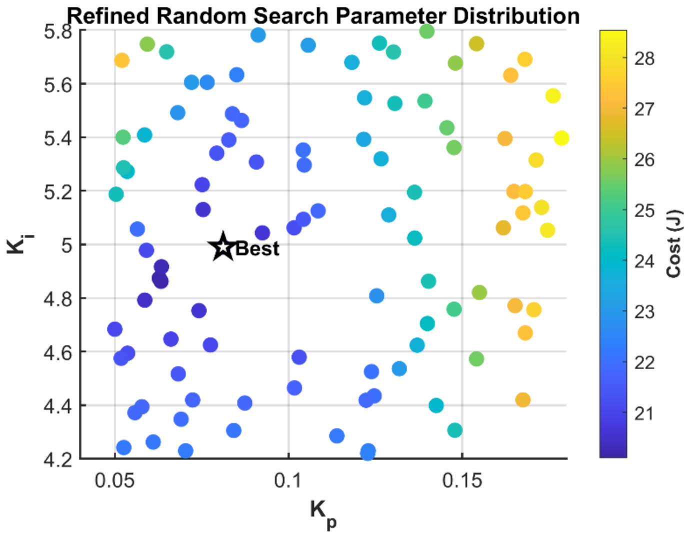

Figure 3. Parameter distribution obtained using the Random Search method

5.5. Local Search Results

The best-cost convergence curve obtained using the Local Search method is given in Figure 4. Local Search aims to improve the cost value by testing neighboring solutions around the suitable initial point obtained from the Random Search result.

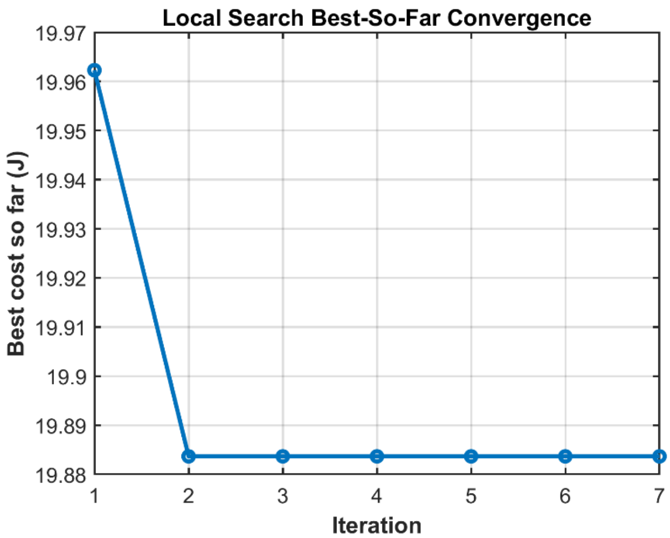

Figure 4. Best-cost convergence curve for the Local Search method

When Figure 4 is examined, it is seen that the cost value decreases rapidly in the first iterations and then becomes stable. This indicates that the initial point obtained by Random Search is suitable and that the Local Search method reaches the local optimum region in a short time. Under nominal conditions, the lowest cost value was obtained with the Local-PI controller. This result shows that the Local Search method can improve nominal performance through fine tuning, especially when a good initial point is used.

5.6. Comparative Results under Nominal and Disturbance Scenarios

The three different PI controllers obtained as a result of the optimization were tested under four different scenarios: nominal condition, measurement noise, measurement noise + actuator saturation, and measurement noise + control delay. The cost function values for these scenarios are given in Table 3.

Table 3. Comparison of cost function (*J*) values under different scenarios.

Scenario	Grid-PI (*J*)	Random-PI (*J*)	Local-PI (*J*)	Best method
Nominal	19.965	20.101	19.884	Local-PI
Noise	39.605	39.249	39.671	Random-PI
Noise + Saturation	42.419	42.081	42.502	Random-PI
Noise + Delay	39.843	39.495	39.938	Random-PI

Table 3 shows that Local-PI achieved the lowest cost under nominal operating conditions. The fact that the Local Search method performs local improvement around the suitable initial point found by Random Search enabled it to reach a slightly lower nominal cost. However, under all scenarios involving disturbance effects, the Random-PI controller produced slightly lower cost values. This result shows that the optimum parameters obtained only under nominal operating conditions may not always be the most robust solution under disturbance effects. This behavior can be explained by the gain values of the two controllers. Random-PI has slightly higher gains $K_p = 0.0811 - K_i = 4.9900$ than Local-PI $K_p = 0.0751 - K_i = 4.9300$. These higher gains provide a stronger corrective action under measurement noise and control delay, whereas Local-PI was mainly fine-tuned to reduce the nominal cost value. Therefore, Local-PI performs better in the nominal case, while Random-PI gives slightly lower cost values under the considered disturbance scenarios.

In the measurement noise scenario, all controllers maintained reference speed tracking; however, due to the noise effect, the cost values increased significantly compared to the nominal condition. The cost values, which were approximately at the level of 20 under nominal conditions, increased to approximately 39 under measurement noise. This increase indicates that the noise effect has a significant influence on error-based performance metrics.

In the actuator saturation scenario, the control voltage was limited to ± 73 V. In this scenario, obtaining $U_{max} = 73$ V for all controllers confirms that the saturation limit was active. In the measurement noise + saturation condition, the cost values were higher compared to the Noise scenario. This indicates that limiting the actuator capacity increases particularly the error integral and time-weighted error metrics. Nevertheless, all controllers maintained stability and continued reference speed tracking.

In the measurement noise + control delay scenario, it is seen that the cost values increased to a limited extent compared to the Noise scenario. This result indicates that the control delay creates an additional deterioration in the transient response; however, the selected PI parameters can maintain system stability. In this scenario as well, the lowest cost value was obtained with the Random-PI controller.

5.7. Comparison of Speed Responses under Different Scenarios

The speed responses of the Grid-PI, Random-PI, and Local-PI controllers under nominal operating conditions, measurement noise, measurement noise + actuator saturation, and measurement noise + control delay scenarios are given in Figure 5.

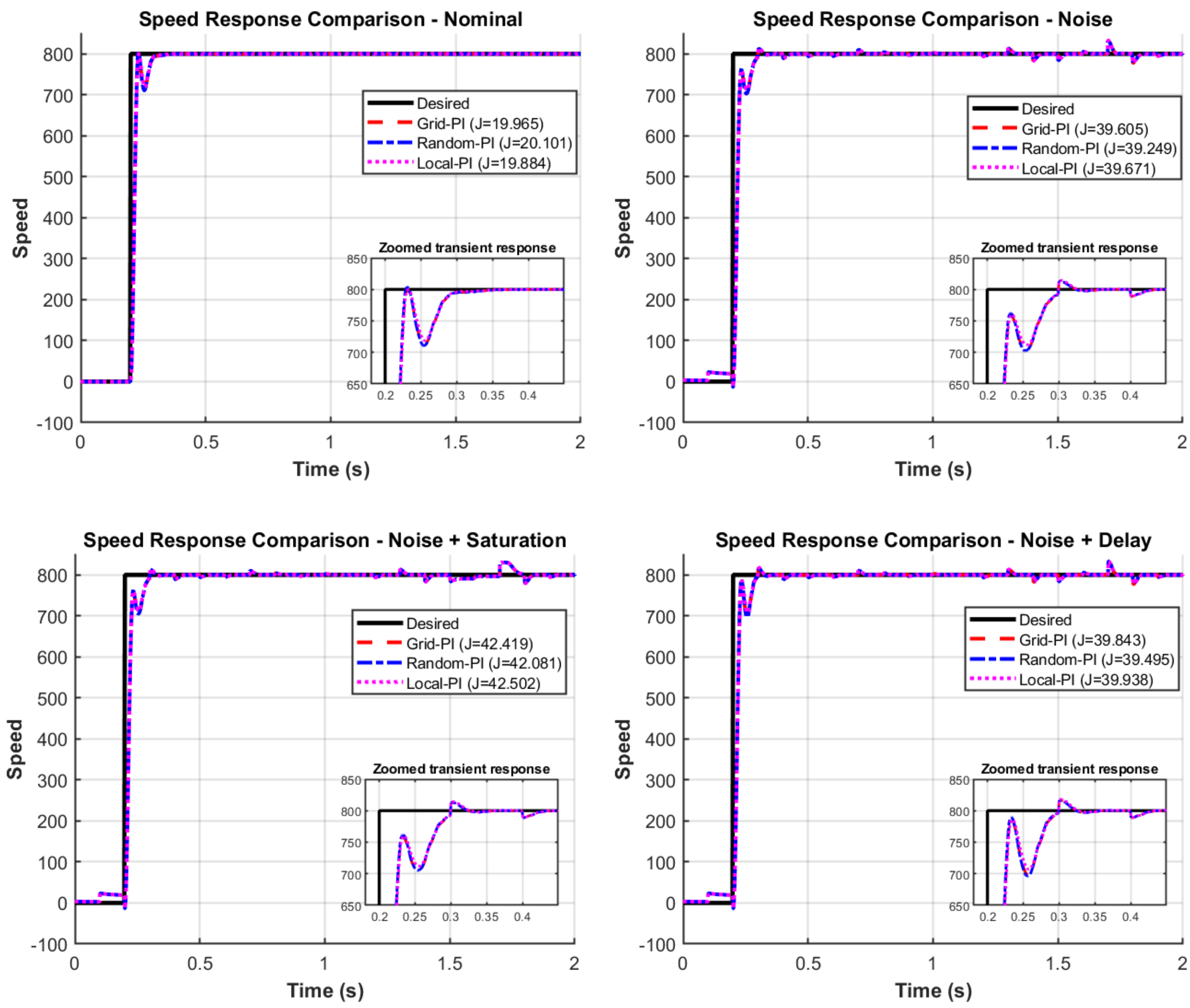

Figure 5. Comparison of speed responses of the Grid-PI, Random-PI, and Local-PI controllers under different operating scenarios.

As seen in Figure 5, all three controllers track the reference speed in a short time and maintain steady-state behavior under all scenarios. Under nominal conditions, the differences among the speed responses of the controllers are quite limited, and the Local-PI controller produces the lowest total cost value. When measurement noise is introduced, small fluctuations occur in the speed response; nevertheless, reference tracking is preserved. In the noise + actuator saturation scenario, the cost values increase, but the system does not lose stability. In the noise + control delay case, a limited deterioration is observed in the transient response; however, all controllers still exhibit acceptable tracking performance around the reference speed.

Overall, the speed responses indicate that the obtained PI gains provide stable and applicable control behavior under all disturbance scenarios. However, when the cost function values are considered, Local-PI stands out under nominal conditions, whereas Random-PI provides slightly lower cost values under the considered disturbance scenarios.

5.8. General Comparison Based on the Cost Function

The comparison of the cost function values under all scenarios is given in Figure 6.

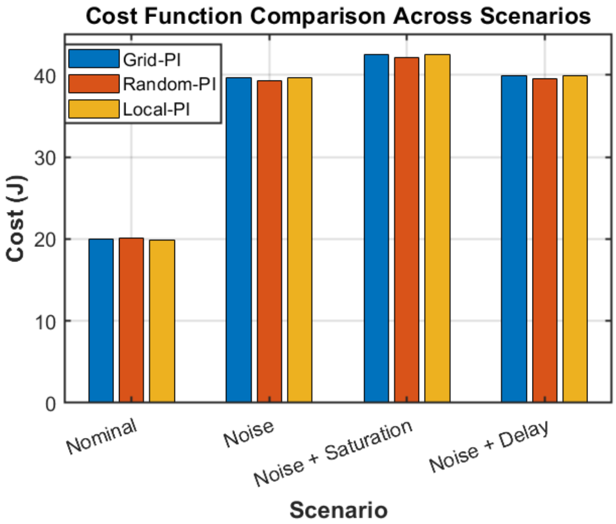

Figure 6. Comparison of cost function values under different scenarios.

Figure 6 shows a clear increase in the cost values from the nominal case to the disturbance scenarios. This increase indicates that effects such as measurement noise and control delay reduce system performance. Nevertheless, all controllers maintained stability and continued reference speed tracking.

While the lowest cost under nominal conditions was obtained with the Local-PI controller, it is seen that the Random-PI controller produced lower cost values under disturbance scenarios. This result shows that, in PI parameter optimization, not only nominal performance but also behavior under disturbance effects should be considered.

6. CONCLUSIONS

In this study, a multi-stage and simulation-based offline search approach was presented for determining the PI controller parameters for BLDC motor speed control. The K_p and K_i parameters of the PI controller were optimized using Grid Search, Random Search, and Local Search methods, and the obtained controllers were compared under both nominal operating conditions and different disturbance scenarios. In the performance evaluation, error-based metrics, transient response indicators, control signal magnitude, overshoot, and final error were considered together.

The Grid Search results enabled the systematic identification of the low-cost region in the K_p -K_i parameter space. The Random Search method generated competitive candidate solutions by performing random sampling within this region. Local Search, starting from the suitable initial point obtained through Random Search, further improved the cost function under nominal conditions. The lowest cost value under nominal operating conditions was obtained with the Local-PI controller, indicating that the Local Search method is effective in terms of fine tuning.

The results obtained under disturbance scenarios revealed a different behavior. In the measurement noise, measurement noise with actuator saturation, and measurement noise with control delay scenarios, the Random-PI controller produced marginally lower cost values. This indicates that PI parameters providing the best performance under nominal conditions do not always constitute the most robust solution under disturbance effects. Therefore, focusing only on nominal performance is not sufficient in PI controller design; the behavior under disturbance effects should also be evaluated.

The findings indicate that Grid Search, Random Search, and Local Search serve complementary roles in the proposed tuning framework. Grid Search reveals the overall structure of the $\mathrm{K_p}$–$\mathrm{K_i}$ parameter space, Random Search helps identify candidate gains that yield slightly lower cost values under the considered disturbance scenarios, and Local Search enables fine adjustment of the nominal response. Accordingly, the proposed approach provides an explainable, reproducible, and comparative framework for PI parameter tuning in BLDC motor speed control.

However, the possibility of convergence to a local minimum in the Local Search stage cannot be completely eliminated, since the method follows a hill-climbing-based local improvement strategy. In future studies, the proposed search-based PI tuning approach can be extended under different reference speed profiles, variable load torque conditions, motor parameter uncertainties, resistance variations, Monte Carlo-based robustness analyses, and different actuator saturation limits. In addition, repeated Random Search runs with different seed values and statistical significance analyses can be conducted to further evaluate the stochastic robustness of the search process. The obtained PI parameters can also be compared with reinforcement learning-based adaptive PI/PID or FOPI/FOPID controller structures to perform a more comprehensive controller performance analysis.

Peer-review: External, Independent.

Acknowledgements:
This study constitutes a part of the first author's master's thesis conducted at Iskenderun Technical University.

Declarations:

1. Statement of Originality:
This manuscript is original, has not been published previously, is not under consideration for publication elsewhere, and has been read and approved by all authors.

2. Author Contributions:
Concept: AES,HG-D; ***Conceptualization:*** AES,HG-D; ***Literature Search:*** AES,HG-D; ***Data Collection:*** AES,HG-D; ***Data Processing:*** AES,HG-D; ***Analysis:*** AES,HG-D; ***Writing – original draft:*** AES,HG-D; ***Writing – review & editing:*** AES,HG-D.

3. Ethics approval:
Not applicable. This study is based exclusively on computer simulations and does not involve human participants, human data, animals, or biological materials.

4. Funding/Support:
The authors received no specific funding or financial support for the research, authorship, and/or publication of this article.

5. Competing Interests:
The authors declare that they have no competing financial or non-financial interests.

6. GenAI Usage Statement:
During the preparation of this manuscript, the authors used ChatGPT solely to improve language, readability, and clarity. All scientific content, methodology, analyses, results, and conclusions were developed, verified, and approved by the authors. The authors take full responsibility for the final content of the manuscript.

7. Sustainable Development Goals:

✵ ✵ ✵

REFERENCES

[1] Lee, B. K., & Ehsani, M. (2003). Advanced simulation model for brushless DC motor drives. Electric Power Components and Systems, 31(9), 841–868. https://doi.org/10.1080/15325000390227191

[2] Jeon, Y. S., Mok, H. S., Choe, G. H., Kim, D. K., & Ryu, J. S. (2000). A new simulation model of BLDC motor with real back EMF waveform. In Proceedings of the 7th Workshop on Computers in Power Electronics (COMPEL 2000), 217-220. IEEE. https://doi.org/10.1109/CIPE.2000.904719

[3] Yamashita, R. Y., Silva, F. L., Santiciolli, F. M., Eckert, J. J., Dedini, F. G., & Silva, L. C. (2018). Comparison between two models of BLDC motor, simulation and data acquisition. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 40(2), 63. https://doi.org/10.1007/s40430-018-1020-0

[4] Burgaç, A. (2026). Energy Harvesting Using Piezoelectric-Film Transducers Excited by Vortex-Induced Vibrations Behind Bluff Bodies for Off-Grid Applications. Isı Bilimi ve Tekniği Dergisi, 46(1), 119-129.

[5] Burgaç, A. (2025). An experimental study on investigation of noise and thrust performance for modified propellers. European Mechanical Science, 9(3), 214-220.

[6] Åström, K. J., & Hägglund, T. (1995). PID controllers: Theory, design, and tuning. The Instrumentation, Systems, and Automation Society.

[7] Oviedo, J. E., Boelen, T., & Van Overschee, P. (2006). Robust advanced PID control (RaPID): PID tuning based on engineering specifications. IEEE Control Systems Magazine, 26(1), 15–19. https://doi.org/10.1109/MCS.2006.1580148

[8] Bel, M., & Gürsoy-Demir, H. (2025). Çift Rotorlu MIMO Sistemi için Sürü Zekâsı Tabanlı LQGI Kontrolcü Tasarımı. Erciyes Üniversitesi Fen Bilimleri Enstitüsü Fen Bilimleri Dergisi, 41(2), 631-646

[9] Xie, W., Wang, J. S., & Wang, H. B. (2019). PI controller of speed regulation of brushless DC motor based on particle swarm optimization algorithm with improved inertia weights. Mathematical Problems in Engineering, 2019(1), 2671792. https://doi.org/10.1155/2019/2671792

[10] Zhang, D., & Wang, J. (2019). Fuzzy PID speed control of BLDC motor based on model design. Journal of Physics: Conference Series, 1303(1), 012124. IOP Publishing. https://doi.org/10.1088/1742-6596/1303/1/012124

[11] Anshory, I., Robandi, I., Wirawan, W., & Jamaaluddin, J. (2019). Identification and implementation hybrid fuzzy logic and PID controller for speed control of BLDC motor. Journal of Physics: Conference Series, 1402(3), 033080. IOP Publishing . https://doi.org/10.1088/1742-6596/1402/3/033080

[12] Varshney, A., Gupta, D., & Dwivedi, B. (2017). Speed response of brushless DC motor using fuzzy PID controller under varying load condition. Journal of Electrical Systems and Information Technology, 4(2), 310–321. https://doi.org/10.1016/j.jesit.2016.12.014

[13] Tabak, A. (2020). Fırçasız doğru akım motorlarının hız kontrolünü gerçekleştirmek için PID/PD kontrolcü tasarımı ve performans incelemesi. Avrupa Bilim ve Teknoloji Dergisi, 19, 145–155. https://doi.org/10.31590/ejosat.707004

[14] Özcan, N., & Gürsoy-Demir, H. (2026). Levy Flight-Augmented Artificial Circulatory System Algorithm for optimal PID tuning in DC motor control. AEU-International Journal of Electronics

and Communications, 204, 156101. 10.1016/j.aeue.2025.156101

[15] Güneş, İ., Eryılmaz, M. M., & Özer, T. (2023). Fırçasız doğru akım motorunun PI tabanlı hız kontrol sisteminin tasarımı ve uygulaması. Journal of the Institute of Science and Technology, 13(2), 983–994. https://doi.org/10.21597/jist.1209249

[16] Ran, M., Wang, Q., & Dong, C. (2016). Anti-windup design for uncertain nonlinear systems subject to actuator saturation and external disturbance. International Journal of Robust and Nonlinear Control, 26(15), 3421–3438. https://doi.org/10.1002/rnc.3524

[17] Chen, G. Y., & Perng, J. W. (2017). PI speed controller design based on GA with time delay for BLDC motor using DSP. In 2017 IEEE International Conference on Mechatronics and Automation (ICMA) (pp. 1174–1179). IEEE. https://doi.org/10.1109/ICMA.2017.8015983

[18] Bergstra, J., & Bengio, Y. (2012). Random search for hyper-parameter optimization. Journal of Machine Learning Research, 13, 281–305.

[19] Aarts, E., & Lenstra, J. K. (Eds.). (2018). Local search in combinatorial optimization. Princeton University Press.