International Scientific Conference 20 UNITECH 26

INTERNATIONAL SCIENTIFIC CONFERENCE
19-21 November 2026
GABROVO

PID-BASED SPEED CONTROL AND DYNAMIC PERFORMANCE ANALYSIS OF A DC MOTOR USING MATLAB/SIMULINK

Ali Riza Motorcu[1*], Hilmi Kuşçu[2]

[1]*Canakkale Onsekiz Mart University, Faculty of Engineering, Department of Industrial Engineering, Terzioglu Campus, 17100, Canakkale, Türkiye*
[2]*Trakya University, Faculty of Engineering, Department of Mechanical Engineering, Ahmet Karadeniz Campus, Edirne, Türkiye*

……
**Corresponding author: armotorcu@comu.edu.tr*

Abstract
Accurate speed regulation of direct-current (DC) motors is essential for electromechanical drive systems requiring rapid transient response, low steady-state error, and stable operation. This study presents an integrated analytical and MATLAB/Simulink-based framework for modeling, PID controller design, and dynamic performance evaluation of an armature-controlled DC motor. The coupled electrical and mechanical dynamics are formulated using physical motor parameters to derive the voltage-to-speed transfer function. A conventional PID controller is tuned using MATLAB/Simulink PID Tuner according to predefined transient-response objectives. The resulting controller gains are Kp = 18.9632, Ki = 55.0564, and Kd = 1.3186. The closed-loop unit-step response exhibits approximately 8.17% maximum overshoot, a 2% settling time of 0.306 s, and negligible steady-state tracking error. Closed-loop pole analysis confirms asymptotic stability. The principal contribution is a consistent and quantitatively characterized PID-control benchmark integrating physical modeling, controller tuning, transient-response assessment, and analytical stability verification. The findings establish a reproducible reference framework for evaluating conventional PID performance, while their applicability remains limited to the investigated nominal linear DC motor model.

Keywords: DC motor, PID control, speed regulation, MATLAB/Simulink, dynamic response.

1. INTRODUCTION

Direct-current (DC) motors are widely used in electromechanical drive applications owing to their favorable speed–torque characteristics, relatively simple mathematical modeling, and convenient speed-control capabilities. Although brushless motors and advanced control techniques have become increasingly prevalent, accurate DC motor speed regulation remains an important control-engineering problem and a useful benchmark for evaluating feedback-control strategies. Recent studies have investigated conventional PID control alongside optimization-based and fractional-order control approaches to improve transient response, tracking accuracy, and robustness [1–3].

The dynamic behavior of an armature-controlled DC motor is determined by the interaction between its electrical and mechanical subsystems. Applied armature voltage determines the armature current and electromagnetic torque, whereas rotor motion produces a back electromotive force that influences the electrical dynamics. Consequently, the design of a speed controller requires an appropriate mathematical representation of these coupled dynamics. The proportional-integral-derivative (PID) controller remains an important industrial control structure because of its relatively simple implementation, intuitive tuning parameters, and ability to modify both transient and steady-state characteristics. PID-based

This is an open access article licensed under Creative Commons Attribution 4.0 International
Print ISSN: 1313-230X, Online ISSN: 3033-1404

motor-speed regulation under different loading conditions has therefore continued to attract research attention [4]. Practical PID implementations may additionally require consideration of phenomena such as actuator saturation and integral windup [5]. Comparative investigations involving PID, fuzzy logic, and sliding-mode control further demonstrate that the selected control strategy can substantially influence the transient performance of DC motor speed-control systems [6].

Recent investigations have emphasized the importance of PID tuning strategies and operating conditions in DC motor control. Taufiqurrahman [7] compared manual tuning, Ziegler–Nichols tuning, and MATLAB/Simulink PID Tuner for DC motor speed regulation under load disturbances. Pambudi et al. [8] investigated PID tuning strategies for DC motor angular-position control under measurement noise. Although these studies address different control objectives and operating conditions, they highlight the importance of evaluating controller performance beyond nominal transient-response characteristics.

Accordingly, the objective of this study is to establish a consistent analytical and simulation-based framework for PID speed regulation of an armature-controlled DC motor. The contribution is focused on maintaining consistency between the physical motor parameters, analytical transfer-function model, MATLAB/Simulink implementation, PID tuning, transient-response metrics, and stability analysis. Rather than proposing a new control algorithm, the study provides an integrated, quantitatively evaluated conventional PID benchmark for the specified DC motor model. The controller is designed to achieve zero steady-state tracking error, overshoot below 10%, and a settling time of approximately 0.3 s. Controller performance is evaluated quantitatively through the step response and analytically through closed-loop pole locations. Accordingly, the analysis establishes a nominal-model reference case for subsequent comparisons under matched motor parameters and evaluation criteria, rather than claiming robustness or superiority over advanced control strategies.

2. MATHEMATICAL MODEL OF THE DC MOTOR

The DC motor is represented using the coupled electrical and mechanical model shown schematically in Fig. 1. The armature circuit consists of resistance R, inductance L, applied voltage $V(t)$, armature current $i(t)$, and back electromotive force $e_b(t)$. The mechanical subsystem is characterized by rotor inertia J, viscous friction coefficient B, electromagnetic torque $T_m(t)$, and angular velocity $\omega(t)$.

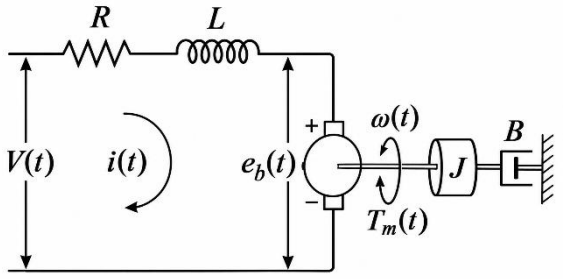


Fig. 1. *Electromechanical model of the armature-controlled DC motor.*

The physical representation in Fig. 1 corresponds to the DC motor model employed in the MATLAB/Simulink analysis. This coupled electromechanical model provides the basis for deriving the voltage-to-angular-speed transfer function [1].

Applying Kirchhoff's voltage law to the armature circuit yields

$$\boldsymbol{V(t) = L\frac{di(t)}{dt} + Ri(t) + e_b(t)} \qquad (1)$$

In Eq. (1), $Ldi(t)/dt$ represents the voltage drop across the armature inductance, $Ri(t)$ represents the resistive voltage drop, and $e_b(t)$ denotes the back electromotive force generated by rotor motion. The back electromotive force is proportional to the angular velocity and is expressed in Eq. (2).

$$e_{b(t)} = K_{e\omega(t)} \quad (2)$$

In Eq. (2), K_e is the back-EMF constant and $\omega(t)$ denotes the angular velocity of the motor shaft. The electromagnetic torque generated by the motor is proportional to the armature current, as given in Eq. (3).

$$T_{m(t)} = K_t i(t) \quad (3)$$

In Eq. (3), K_t denotes the torque constant and $T_m(t)$ represents the electromagnetic torque. The mechanical dynamics of the motor are obtained by applying Newton's rotational equation to the rotor, resulting in Eq. (4).

$$J\frac{d\omega(t)}{dt} + B\omega(t) = T_m(t) \quad (4)$$

In Eq. (4), $J\, d\omega(t)/dt$ represents the inertial torque, while $B\omega(t)$ represents the viscous-friction torque. Accordingly, Eq. (4) relates the electromagnetic torque to the resulting angular motion of the rotor. The nominal model assumes zero external load torque, constant motor parameters, and linear magnetic characteristics. For transfer-function analysis, zero initial conditions are assumed and the Laplace transform is applied to the electrical and mechanical equations. The resulting armature-circuit relationship is given in Eq. (5).

$$V(s) - K_e\Omega(s) = (Ls + R)I(s) \quad (5)$$

Similarly, the mechanical relationship in the Laplace domain is expressed in Eq. (6).

$$(Js + B)\Omega(s) = K_t I(s) \quad (6)$$

Here, $V(s)$, $I(s)$, and $\Omega(s)$ denote the Laplace transforms of the applied armature voltage, armature current, and angular velocity, respectively, while s is the complex Laplace variable. Eliminating $I(s)$ from Eqs. (5) and (6) yields the voltage-to-angular-speed transfer function given in Eq. (7).

$$G(s) = \frac{\Omega(s)}{V(s)} = \frac{K_t}{(Ls+R)(Js+B)+K_tK_e} \quad (7)$$

Equation (7) defines the plant transfer function used throughout the subsequent MATLAB/Simulink implementation and PID-controller design. The physical parameters employed in the analysis are summarized in Table 1.

Table 1. *DC motor parameters used in the simulation*

Parameter	Physical quantity	Value
J	Rotor moment of inertia	0.02 $kg\,m^2$
B	Viscous friction coefficient	0.05 $N\,m\,s/rad$
K_t	Torque constant	0.20 $N\,m/A$
K_e	Back-EMF constant	0.10 $V\,s/rad$
R	Armature resistance	3.0 Ω
L	Armature inductance	0.50 H

Substituting the parameters into Eq. (7) yields

$$G(s) = \frac{0.2}{(0.5s+3)(0.02s+0.05)+0.02} \quad (8)$$

Equation (8) represents the numerical voltage-to-angular-speed transfer function obtained directly from the physical motor parameters listed in Table 1. Dividing the numerator and denominator by the leading denominator coefficient yields the normalized transfer function given in Eq. (9).

$$G(s) = \frac{20}{s^2+8.5s+17} \quad (9)$$

As indicated by Eq. (9), the characteristic polynomial of the uncontrolled DC motor model is $s^2+8.5s+17$. The roots of this characteristic polynomial, which correspond to the open-loop poles of the motor model, are given in Eq. (10).

$$p_1 = -5.2808,\ p_2 = -3.2192 \quad (10)$$

These values are consistent with the open-loop pole calculation performed in MATLAB. Since both poles lie in the open left half of the complex plane, the motor model itself is asymptotically stable.

3. PID CONTROLLER DESIGN AND METHODOLOGY

A unity negative-feedback speed-control architecture is employed, as illustrated in Fig. 2. The reference angular velocity is compared with the actual motor speed, and the resulting tracking error is processed by the PID controller.

The ideal parallel-form PID controller used in this study is expressed in Eq. (11).

$$C(s) = K_p + \frac{K_i}{s} + K_d s \qquad (11)$$

In Eq. (11), K_p is the proportional gain, K_i is the integral gain, and K_d is the derivative gain. The proportional component responds directly to the instantaneous speed error, the integral component contributes to the elimination of steady-state error, and the derivative component provides an anticipatory action based on the rate of change of the error. In the Simulink implementation, the derivative action is filtered, whereas Eq. (11) represents the corresponding ideal PID form used for analytical modeling.

The performance of a conventional PID controller strongly depends on the appropriate selection of its proportional, integral, and derivative gains. Optimization-based PID tuning has therefore also been investigated specifically for DC motor speed regulation [2]. Nevertheless, optimization-based tuning and conventional PID Tuner-based design should be evaluated under consistent plant models and performance criteria to enable meaningful quantitative comparisons.

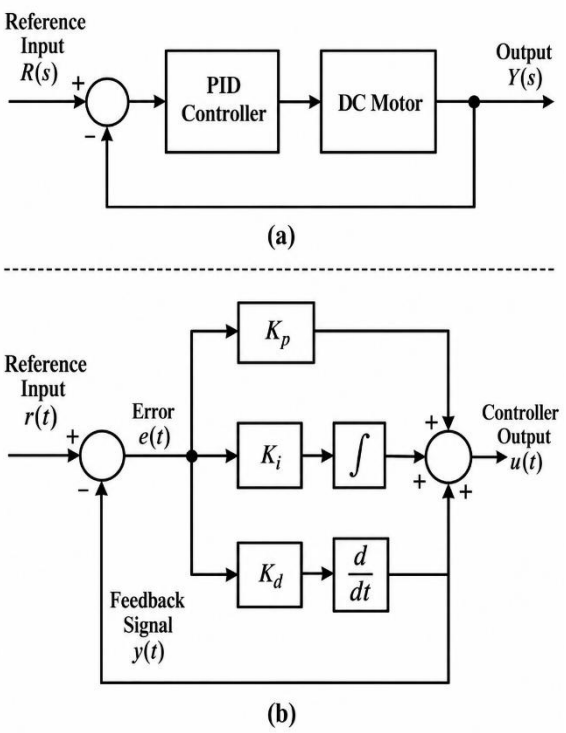


Fig. 2. *Closed-loop PID speed-control structure of the DC motor.*
(a) Overall closed-loop control system.
(b) Internal structure of the PID controller.

The controller-design objectives were specified as:

- overshoot below 10%;
- settling time approximately 0.3 s;
- zero steady-state tracking error.

The controller was tuned using the PID Tuner associated with the MATLAB/Simulink PID Controller block. The controller was configured as a continuous-time, parallel-form, one-degree-of-freedom PID controller, with transfer-function-based tuning. The derivative filter coefficient was N=3182.8718. The corresponding filtered PID controller is represented by $C_f(s)=K_p+K_{i/s}+K_d.N_s/(s+N)$, where N denotes the derivative-filter coefficient. The resulting parameters were

$$K_p = 18.9632,$$
$$K_i = 55.0564,$$
$$K_d = 1.3186 \qquad (12)$$

The PID gains given in Eq. (12) were obtained from the MATLAB/Simulink PID Tuner analysis and used throughout the subsequent closed-loop numerical and stability analyses. For a unity negative-feedback configuration, the relationship between the PID controller *C(s)*, the motor plant *G(s)*, and the resulting closed-loop system is defined by Eq. (13).

$$T(s) = \frac{C(s)G(s)}{1+C(s)G(s)} \quad (13)$$

In Eq. (13), *T(s)* denotes the closed-loop transfer function from the reference angular velocity to the motor angular velocity. Substituting the motor transfer function given in Eq. (9) and the PID controller parameters given in Eq. (12) into Eq. (13) results in the closed-loop transfer function presented in Eq. (14).

$$T(s) = \frac{26.372s^2+379.264s+1101.128}{s^3+34.872s^2+396.264s+1101.128} \quad (14)$$

Equation (14) constitutes the mathematical basis for the subsequent transient-response and closed-loop stability analyses. Its denominator, $s^3 + 34.872\ s^2 + 396.264s + 1101.128$, is the characteristic polynomial of the PID-controlled closed-loop system. Equation (14) represents the ideal PID approximation, whereas the Simulink implementation employs the filtered PID controller defined above, with N=3182.8718. Numerical evaluation of the filtered model yields approximately 8.17% overshoot and a 2% settling time of 0.306 s, consistent with the reported simulation results. The stability of the filtered implementation was additionally verified from its fourth-order closed-loop characteristic polynomial, whose poles lie entirely in the open left half of the complex plane.

4. SIMULINK IMPLEMENTATION

The analytical DC motor model was implemented in MATLAB and MATLAB/Simulink. MATLAB was used to construct the motor transfer function and determine the poles of the uncontrolled motor model. The electrical and mechanical subsystems were defined separately and combined using the series and feedback functions.

The closed-loop Simulink model consists of a reference step input, a summing junction, a PID Controller block, the DC motor model, an output-speed measurement, and a unity negative-feedback path. The PID controller parameters obtained from the tuning procedure were incorporated into the Simulink model to evaluate the transient speed response of the closed-loop system.

```
J=0.02;
B=0.05;
Ke=0.1;
Kt=0.2;
R=3;
L=0.5;
G1=tf(1,[L R]);
G2=Kt;
G3=tf(1,[J B]);
G=series(G1,G2);
GG=series(G,G3);
DC_Motor_TF=feedback(GG,Ke,-1)
kutuplar_DC=pole(DC_Motor_TF)
pzmap(DC_Motor_TF)
```

Fig. 3. *MATLAB code used for transfer-function modeling and open-loop pole analysis of the DC motor.*

The PID parameters were tuned using the transfer-function-based PID Tuner associated with the Simulink PID Controller block, with the aim of achieving the desired compromise between response speed and overshoot. The resulting controller gains were subsequently retained in the closed-loop simulation. The closed-loop performance was evaluated using a unit-step speed reference, with maximum overshoot and settling time determined from the transient response using a 2% settling band.

5. RESULTS AND DISCUSSION

This section evaluates the dynamic performance and stability of the PID-controlled DC motor using

MATLAB/Simulink simulation results and analytical calculations. The findings are discussed in terms of transient-response characteristics, quantitative performance indicators, closed-loop pole locations, and comparisons with relevant studies.

5.1. Transient Speed Response

The resulting closed-loop unit-step response is presented in Fig. 4.

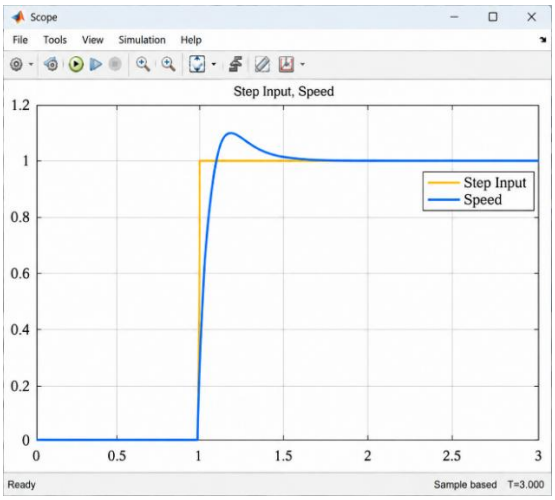


Fig. 4. *Unit-step speed response of the PID-controlled DC motor.*

The tuned controller produced a maximum overshoot of approximately 8.17% and a 2% settling time of approximately 0.306 s, as determined from the closed-loop response analysis. The integral term provides zero steady-state error for the unit-step reference in the ideal model.

5.2. Quantitative Performance Evaluation

Table 2 shows that the overshoot requirement is satisfied and that the 2% settling time is very close to the targeted value of 0.3 s.

Table 2. *Dynamic performance of the tuned PID speed-control system*

Performance indicator	Design objective	Obtained value
Overshoot	<10%	8.17%
Settling time (2%)	≈0.30 s	0.306 s
Steady-state error	0	≈0
Closed-loop stability	Stable	Stable

Importantly, the result should not be interpreted as satisfying a strict $T_s < 0.3$ s requirement because the measured value is approximately 0.306 s. Instead, the tuning provides the intended compromise between rapid response and limited overshoot.

5.3. Stability Analysis

The roots of the characteristic polynomial obtained from the denominator of Eq. (14) determine the closed-loop pole locations of the ideal PID model and therefore provide a direct assessment of system stability. The calculated poles are presented in Eq. (15).

$$p_{1,2} = -15.405 \pm j5.815\,,\; p_3 = -4.061 \quad (15)$$

As shown in Eq. (15), all closed-loop poles have negative real parts. The corresponding pole locations are illustrated in Fig. 5. Since all poles are located in the open left half of the complex s-plane, the ideal PID-controlled closed-loop system is asymptotically stable.

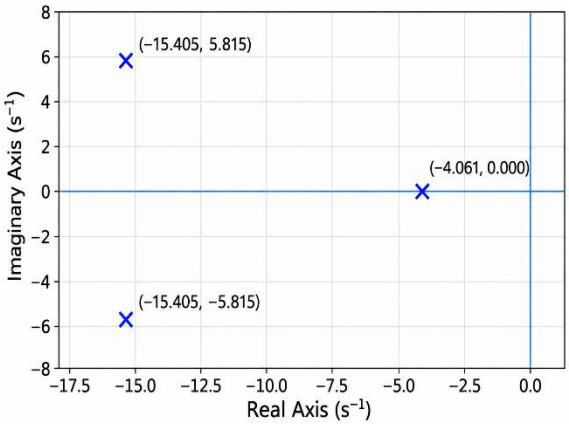


Fig. 5. *Pole locations of the ideal PID-controlled closed-loop system.*

5.4. Relation to Recent Studies

Recent research on DC motor speed regulation has increasingly focused on optimization-based and fractional-order control strategies. Jabari et al. [3] developed a multi-stage fractional-order controller optimized using the Pelican Optimization Algorithm for DC motor speed control. Rizk-Allah et al. [9] investigated an optimization-based fractional-order PID

strategy aimed at improving DC motor speed regulation and disturbance rejection. Optimization of conventional PID parameters has also been investigated; Izci [2] employed a hybrid Lévy-flight and Nelder–Mead approach for PID tuning in DC motor speed regulation. More recently, Ekinci et al. [10] proposed the Gazelle Simplex Optimizer for efficient DC motor speed-control optimization. These studies demonstrate the continuing trend toward optimization-assisted controller tuning for improving dynamic performance.

Quantitative results reported in recent studies provide additional context for the present findings. Jabari et al. [3] reported 0.0016% overshoot and a settling time of 0.0054 s for their optimized fractional-order controller under Mode 1 operating conditions. Almawla et al. [6] reported PID, fuzzy-logic, and sliding-mode settling times of 6.707, 2.165, and 1.627 ms, respectively, for their investigated DC motor system. In comparison, the present conventional PID implementation yielded 8.17% overshoot and a 0.306 s settling time. Because the motor parameters, controller structures, and evaluation conditions differ, these results are presented as quantitative literature benchmarks rather than direct evidence of relative controller superiority. Furthermore, recent comparative studies have investigated the influence of PID tuning procedures and operating conditions on DC motor control performance. Taufiqurrahman [7] examined manual, Ziegler–Nichols, and MATLAB/Simulink PID Tuner-based tuning under load disturbances, while Pambudi et al. [8] investigated tuning-related noise robustness in DC motor position control.

Saravanan et al. [11] investigated PID tuning using Kookaburra and Red Panda optimization algorithms and evaluated transient performance, stability, and robustness under variations in DC motor parameters. Choi and Kim [12] addressed parameter uncertainties and PWM switching noise through a sampled-data speed-control framework. These investigations demonstrate the importance of assessing controller performance beyond nominal operating conditions. In contrast to these optimization-based approaches, the objective of the present study is not to introduce an additional metaheuristic algorithm, but to establish a transparent conventional PID benchmark in which the physical model, controller parameters, transient-response metrics, and stability properties remain analytically traceable. The obtained 8.17% overshoot and approximately 0.306 s settling time demonstrate that conventional PID control can provide satisfactory nominal performance for the investigated linear DC motor model. Nevertheless, the present simulation-based analysis does not establish robustness against substantial parameter uncertainty, actuator saturation, measurement noise, or nonlinear load disturbances. These aspects constitute appropriate directions for subsequent experimental and optimization-based studies.

6. CONCLUSION

This study presented a consistent analytical and MATLAB/Simulink framework for PID-based speed regulation of an armature-controlled DC motor. The electrical and mechanical dynamics were formulated from the physical motor parameters and reduced to a second-order voltage-to-speed transfer function. The resulting open-loop motor model possesses poles at -5.2808 and -3.2192, confirming inherent asymptotic stability. A PID controller was subsequently tuned in MATLAB/Simulink with gains of K_p=18.9632, K_i=55.0564, and K_d=1.3186. The resulting closed-loop response exhibited approximately 8.17% overshoot, a 2% settling time of approximately 0.306 s, and negligible steady-state tracking error. Independent pole analysis of the ideal PID-controlled transfer function confirmed closed-loop asymptotic stability.

The results show that conventional PID control provides an effective and structurally simple benchmark solution for the investigated linear DC motor speed-control problem. The main limitation of the present study is its simulation-based and nominal-model character. Future research should therefore investigate controller robustness under parameter variations, load disturbances, actuator saturation, and measurement noise and should compare conventional PID tuning with optimization-based and fractional-order control strategies using identical motor parameters and performance criteria.

REFERENCES

[1] Hammoodi SJ, Flayyih KS, Hamad AR. Design and implementation speed control system of DC motor based on PID control and MATLAB Simulink. International Journal of Power Electronics and Drive Systems 2020; 11(1):127-134. DOI: 10.11591/ijpeds.v11.i1.pp127-134.

[2] Izci D. Design and application of an optimally tuned PID controller for DC motor speed regulation via a novel hybrid Levy flight distribution and Nelder-Mead algorithm. Transactions of the Institute of Measurement and Control 2021; 43(14):3195-3211. DOI: 10.1177/01423312211019633.

[3] Jabari M, Ekinci S, Izci D, Bajaj M, Zaitsev I. Efficient DC motor speed control using a novel multi-stage FOPD(1+PI) controller optimized by the Pelican optimization algorithm. Scientific Reports 2024; 14:22442. DOI: 10.1038/s41598-024-73409-5.

[4] Rajs V, Rašević NL, Bodić MZ, Zuković MM, Babković KB. PID controller design for motor speed regulation with linear and non-linear load. IFAC-PapersOnLine 2022; 55(4):225-229. DOI: 10.1016/j.ifacol.2022.06.037.

[5] Yapp KKC, Hoo CL, Lai CH. New anti-windup proportional-integral-derivative for motor speed control. Asian Journal of Control 2024; 26(6):2854-2866. DOI: 10.1002/asjc.3390.

[6] Almawla AM, Hussein MJ, Abdullah AT. A comparative study of DC motor speed control techniques using fuzzy, SMC and PID. Journal Européen des Systèmes Automatisés 2024; 57(2):397-406. DOI: 10.18280/jesa.570209.

[7] Taufiqurrahman MA. Analisis perbandingan penalaan PID manual, Ziegler–Nichols, dan PID Tuner MATLAB/Simulink untuk pengendalian kecepatan motor DC terhadap gangguan beban. Jurnal Elektronika dan Otomasi Industri 2026; 13(2). DOI: 10.33795/elkolind.v13i2.10165.

[8] Pambudi AD, Heryanto MA, Wulandari SA. PID tuning for robustness to noise in DC motor angular position control system. JTEIN: Jurnal Teknik Elektro Indonesia 2025; 6(2):113–124. DOI: 10.24036/jtein.v6i2.751.

[9] Rizk-Allah RM, Izci D, Ekinci S, Diabat A, Ezugwu AE, Abualigah L. Incorporating adaptive local search and experience-based perturbed learning into artificial rabbits optimizer for improved DC motor speed regulation. International Journal of Electrical Power & Energy Systems 2024; 162:110266. DOI: 10.1016/j.ijepes.2024.110266.

[10] Ekinci S, Izci D, Yilmaz M. Efficient speed control for DC motors using novel Gazelle Simplex Optimizer. IEEE Access 2023; 11: 105830-105842. DOI: 10.1109/ACCESS.2023.3319596.

[11] Saravanan G, Pazhanimuthu C, Naveen P. Performance improvement of DC motor control system using PID controller with Kookaburra and Red Panda optimization algorithm. Scientific Reports 2025; 15:20021. DOI: 10.1038/s41598-025-87607-2.

[12] Choi YJ, Kim SH. Towards robust sampled-data speed control of DC drive systems: Coping with parameter uncertainties and PWM switching noise. International Journal of Control, Automation and Systems 2025; 23(11):3209–3216. DOI: 10.1007/s12555-025-0580-x.