

# Evaluasi Metode Penalaan PID pada Kendali Kecepatan Motor DC dengan Gangguan Beban

Muhammad Ayyasy Taufiqurrahman<sup>1\*)</sup>

<sup>1</sup> Jurusan Teknik Elektro, Universitas Singaperbangsa Karawang, Jl. HS.Ronggo Waluyo, Karawang, Indonesia

\*Penulis Korespondensi, e-mail: [2210631160097@student.unsika.ac.id](mailto:2210631160097@student.unsika.ac.id)

Received: 18/06/2026

Revised: 13/07/2026

Accepted: 13/07/2026

## ABSTRAK

Motor arus searah (DC) banyak digunakan pada aktuator industri, namun kinerjanya mudah terganggu oleh perubahan beban mendadak sehingga memerlukan sistem kendali yang andal. Penelitian ini membandingkan tiga teknik penalaan pengendali Proportional-Integral-Derivative (PID), yaitu penalaan manual, metode Ziegler-Nichols, dan PID Tuner pada MATLAB/Simulink, untuk pengendalian kecepatan motor DC yang dikenai gangguan beban. Model matematis motor DC diturunkan hingga diperoleh fungsi alih orde dua, kemudian dianalisis secara analitis dan disimulasikan. Karakteristik respons transien, kestabilan melalui kriteria Routh-Hurwitz, galat keadaan tunak, serta ketahanan terhadap gangguan dievaluasi untuk setiap metode. Hasil simulasi menunjukkan ketiga metode menghasilkan sistem stabil dengan galat keadaan tunak mendekati nol. PID Tuner menghasilkan rise time tercepat (0,1105 detik) dan overshoot terkecil (0,92%). Secara keseluruhan, penalaan berbasis PID Tuner menawarkan kompromi performa terbaik dan direkomendasikan untuk aplikasi yang menuntut respons halus dan tahan gangguan, sedangkan penalaan manual menjadi alternatif yang sangat kompetitif ditinjau dari indeks ITAE.

**Kata Kunci:** motor DC; pengendali PID; Ziegler-Nichols; PID Tuner; gangguan beban

## ABSTRACT

Direct current (DC) motors are widely used as industrial actuators, but their performance is easily affected by sudden load changes, which calls for a reliable control system. This study compares three tuning techniques for a Proportional-Integral-Derivative (PID) controller, namely manual tuning, the Ziegler-Nichols method, and the PID Tuner in MATLAB/Simulink, for controlling the speed of a DC motor subjected to a load disturbance. The mathematical model of the DC motor is derived to obtain a second-order transfer function, then analyzed analytically and simulated. The transient response, stability through the Routh-Hurwitz criterion, steady-state error, and disturbance rejection are evaluated for each method. The results show all three methods yield a stable system with a steady-state error approaching zero. The PID Tuner provides the fastest rise time (0.1105 s) and the smallest overshoot (0.92%). Overall, PID Tuner-based tuning offers the best overall performance compromise and is recommended for applications demanding a smooth, disturbance-resistant response, while manual tuning remains a highly competitive alternative in terms of the ITAE index.

**Keywords:** DC motor; PID controller; Ziegler-Nichols; PID Tuner; load disturbance

## 1. PENDAHULUAN

Motor arus searah atau motor DC masih menempati posisi penting dalam berbagai sistem penggerak, mulai dari robotika, kendaraan listrik, konveyor, hingga peralatan rumah tangga. Popularitas tersebut tidak lepas dari kemudahan pengaturan kecepatannya, rentang torsi yang lebar, serta hubungan yang relatif linear antara tegangan masukan dan kecepatan putar [7]. Akan tetapi, ketika motor dioperasikan pada kondisi nyata, beban yang dipikulnya tidak selalu konstan. Perubahan beban yang terjadi secara tiba-

p-ISSN: 2356-0533; e-ISSN: 2355-9195



tiba dapat menyebabkan penyimpangan kecepatan dari nilai yang dikehendaki, dan apabila tidak dikompensasi, penyimpangan ini akan menurunkan ketelitian serta kestabilan kerja sistem secara keseluruhan.

Untuk menjaga agar kecepatan motor tetap berada pada nilai acuan meskipun terjadi gangguan, diperlukan sistem kendali lingkaran tertutup. Di antara sekian banyak strategi kendali, pengendali Proportional-Integral-Derivative (PID) tetap menjadi pilihan yang paling lazim digunakan di lapangan [9]. Struktur PID yang sederhana, kebutuhan komputasi yang ringan, serta kemampuannya menangani sebagian besar proses orde rendah menjadikannya pengendali yang sangat praktis. Komponen proporsional bekerja menanggapi besarnya galat saat ini, komponen integral menghilangkan galat keadaan tunak, sedangkan komponen derivatif memberikan efek peredaman dengan mengantisipasi laju perubahan galat.

Meskipun strukturnya sederhana, keberhasilan pengendali PID sangat bergantung pada ketepatan penentuan ketiga parameternya, yakni penguatan proporsional ( $K_p$ ), penguatan integral ( $K_i$ ), dan penguatan derivatif ( $K_d$ ). Proses penentuan parameter inilah yang dikenal sebagai penalaan (tuning). Pemilihan nilai yang kurang tepat dapat menghasilkan respons yang terlalu lambat, berosilasi berlebihan, bahkan tidak stabil. Oleh karena itu, metode penalaan yang digunakan menjadi penentu utama kualitas kinerja sistem kendali yang dibangun [5].

Terdapat beragam pendekatan untuk melakukan penalaan PID. Pendekatan manual berbasis coba-coba mengandalkan pengalaman perancang dalam menyetel nilai parameter sambil mengamati respons sistem. Cara ini intuitif dan tidak memerlukan perangkat khusus, tetapi hasilnya sangat bergantung pada keterampilan perancang dan cenderung memakan waktu. Sebagai alternatif yang lebih sistematis, Ziegler dan Nichols memperkenalkan metode penalaan empiris yang menentukan parameter PID berdasarkan penguatan kritis dan periode osilasi sistem [1]. Metode ini populer karena prosedurnya baku dan mudah diterapkan, walaupun sering menghasilkan respons dengan overshoot yang cukup besar, sebagaimana ditunjukkan oleh Widagdo dkk. dalam perbandingan metode ZN dengan Chien-Hrones-Reswick pada kendali kecepatan motor DC [12]. Sementara itu, perkembangan perangkat lunak kendali memunculkan pendekatan penalaan otomatis seperti fitur PID Tuner pada MATLAB/Simulink dan algoritma optimasi seperti Particle Swarm Optimization (PSO), yang mengoptimasi parameter dengan mempertimbangkan keseimbangan antara kecepatan respons dan kekakuan sistem [13].

Sejumlah penelitian telah mengkaji pengendalian kecepatan motor DC menggunakan PID dengan berbagai pendekatan penalaan. Ma'arif dkk. menerapkan PID dengan penalaan otomatis berbasis PID Tuner MATLAB pada motor DC nyata menggunakan Arduino dan membuktikan bahwa pendekatan berbasis identifikasi sistem mampu menghasilkan respons transien yang baik [9]. Saputra dkk. juga membuktikan efektivitas PID Tuner MATLAB dalam kendali kecepatan motor DC berbasis PLC dengan performa yang konsisten [11]. Rahayu dkk. membandingkan PID konvensional dengan PID yang dioptimasi menggunakan PSO dan menunjukkan bahwa PSO secara konsisten menghasilkan nilai indeks kinerja integral (IAE, ISE, ITAE) yang lebih kecil [13]. Penelitian terbaru juga memperkuat arah ini: Yıldırım dkk. mengoptimasi parameter PID motor DC menggunakan algoritma PSO di MATLAB/Simulink dan menegaskan keunggulan penalaan berbasis optimasi dibandingkan PID standar [15], sementara Han menyajikan tinjauan komparatif beragam teknik penalaan PID untuk pengaturan kecepatan motor DC, termasuk tanggapannya terhadap gangguan beban [16]. Namun, perbandingan langsung antara penalaan manual, Ziegler-Nichols, dan PID Tuner MATLAB secara serentak pada model yang sama, disertai analisis gangguan beban dan verifikasi analitis lengkap termasuk kriteria Routh-Hurwitz dan teorema nilai akhir, masih jarang dijumpai dalam literatur yang ada.

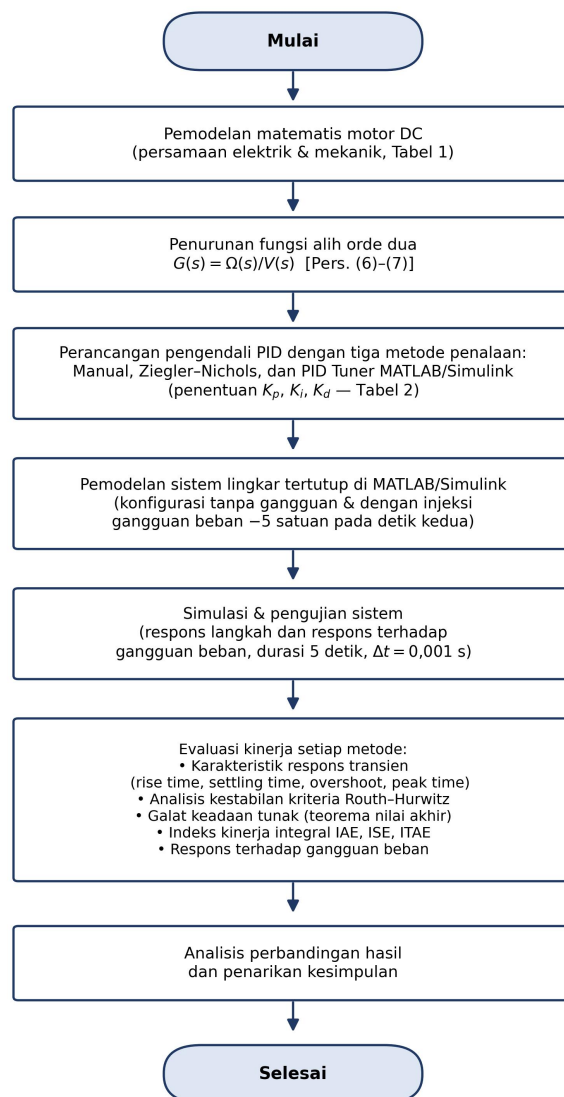
Berdasarkan latar belakang tersebut, penelitian ini bertujuan membandingkan secara menyeluruh kinerja penalaan PID manual, Ziegler-Nichols, dan PID Tuner MATLAB/Simulink untuk pengendalian kecepatan motor DC. Evaluasi dilakukan dari sisi karakteristik respons transien, kestabilan melalui kriteria Routh-Hurwitz, galat keadaan tunak, serta tanggapan sistem ketika diberi gangguan beban. Kontribusi penelitian ini terletak pada penyajian analisis komparatif yang memadukan penurunan model secara analitis dengan verifikasi melalui simulasi, sehingga dapat menjadi acuan praktis dalam memilih metode penalaan



yang paling sesuai. Selanjutnya, makalah ini disusun sebagai berikut: bagian kedua memaparkan metode penelitian yang mencakup pemodelan motor DC dan perancangan pengendali; bagian ketiga menyajikan hasil dan pembahasan; dan bagian keempat memuat kesimpulan.

## 2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan pemodelan dan simulasi. Tahapannya meliputi penurunan model matematis motor DC, penentuan fungsi alih, perancangan pengendali PID dengan tiga metode penalaan, serta pengujian sistem lingkaran tertutup pada kondisi tanpa dan dengan gangguan beban. Seluruh simulasi dilaksanakan menggunakan perangkat lunak MATLAB/Simulink, sedangkan analisis kestabilan dan galat keadaan tunak diverifikasi secara analitis. Alur lengkap tahapan penelitian secara ringkas ditampilkan pada Gambar 1.



Gambar 1. Diagram alir tahapan penelitian

### 2.1 Pemodelan Matematis Motor DC

Motor DC dimodelkan melalui dua persamaan dinamik yang saling berkaitan, yaitu persamaan rangkaian elektrik pada sisi armatur dan persamaan keseimbangan mekanik pada sisi poros. Pada sisi elektrik,

p-ISSN: 2356-0533; e-ISSN: 2355-9195



tegangan masukan motor sama dengan jumlah jatuh tegangan pada induktansi, resistansi armatur, dan tegangan gaya gerak listrik balik (back-EMF) yang sebanding dengan kecepatan sudut motor. Hubungan tersebut dinyatakan oleh persamaan berikut.

$$V(t) = L \frac{di(t)}{dt} + R \cdot i(t) + K\omega(t) \quad (1)$$

Adapun pada sisi mekanik, torsi yang dibangkitkan motor sebanding dengan arus armatur dan digunakan untuk mengatasi momen inersia serta gesekan viskos pada poros. Keseimbangan mekanik ini dirumuskan sebagai berikut.

$$J \frac{d\omega(t)}{dt} + B \cdot \omega(t) = K \cdot i(t) \quad (2)$$

Pada kedua persamaan tersebut,  $V(t)$  menyatakan tegangan masukan motor,  $i(t)$  arus armatur, dan  $\omega(t)$  kecepatan sudut poros. Simbol  $L$ ,  $R$ ,  $J$ ,  $B$ , dan  $K$  berturut-turut adalah induktansi armatur, resistansi armatur, momen inersia, koefisien gesekan, dan konstanta motor. Nilai parameter motor DC yang digunakan dalam penelitian ini dirangkum pada Tabel I.

TABEL I: Parameter motor DC yang digunakan

| Parameter          | Simbol | Nilai | Satuan               |
|--------------------|--------|-------|----------------------|
| Resistansi armatur | R      | 1     | $\Omega$             |
| Induktansi armatur | L      | 0,5   | H                    |
| Momen inersia      | J      | 0,01  | kg·m <sup>2</sup>    |
| Koefisien gesekan  | B      | 0,1   | N·m·s                |
| Konstanta motor    | K      | 0,01  | N·m/A atau V/(rad/s) |

## 2.2 Penurunan Fungsi Alih

Fungsi alih diturunkan dengan menerapkan transformasi Laplace pada Persamaan (1) dan (2) secara terpisah, kemudian mengeliminasi variabel arus armatur. Dengan mengasumsikan kondisi awal nol, Persamaan (1) dalam domain Laplace menjadi:

$$V(s) = (Ls + R)I(s) + K\Omega(s) \quad (3)$$

Sementara itu, transformasi Laplace Persamaan (2) menghasilkan hubungan antara arus armatur dan kecepatan sudut, sehingga arus dapat dinyatakan sebagai:

$$I(s) = \frac{K\Omega(s)}{Js + B} \quad (4)$$

Eksprei  $I(s)$  di atas disubstitusikan ke Persamaan (3), dan setelah penataan ulang suku-suku tegangan diperoleh:

$$V(s) = (Ls + R) \frac{K\Omega(s)}{Js + B} + K\Omega(s) \quad (5)$$

Dengan mengelompokkan dan menyederhanakan, rasio  $\Omega(s)/V(s)$  menghasilkan fungsi alih umum motor DC orde dua sebagai berikut.

$$G(s) = \frac{\Omega(s)}{V(s)} = \frac{K}{[LJ s^2 + (LB + RJ)s + (RB + K^2)]} \quad (6)$$

Dengan memasukkan nilai parameter dari Tabel 1, koefisien penyebut dihitung sebagai:  $LJ = 0,005$ ;  $LB + RJ = 0,5 \times 0,1 + 1 \times 0,01 = 0,06$ ; dan  $RB + K^2 = 1 \times 0,1 + 0,01^2 \approx 0,1001$ . Fungsi alih numerik yang digunakan dalam penelitian ini adalah:



$$G(s) = \frac{0,01}{(0,005s^2 + 0,06s + 0,1001)} \quad (7)$$

### 2.3 Struktur dan Perancangan Pengendali PID

Pengendali PID yang digunakan berbentuk paralel standar, di mana sinyal kendali merupakan jumlah dari aksi proporsional, integral, dan derivatif terhadap galat. Dalam domain Laplace, fungsi alih pengendali dapat dituliskan sebagai berikut.

$$C(s) = Kp + \frac{Ki}{s} + Kd s = \frac{(Kd s^2 + Kp s + Ki)}{s} \quad (8)$$

Apabila pengendali ini dihubungkan dengan plant dalam konfigurasi lingkaran tertutup berumpan-balik satuan, fungsi alih sistem keseluruhan dari acuan ke keluaran dinyatakan oleh persamaan 9 berikut. Tiga metode penalaan diterapkan untuk menentukan nilai  $K_p$ ,  $K_i$ , dan  $K_d$ . Penjelasan ringkas masing-masing metode diuraikan pada subbagian berikut, sedangkan nilai akhir parameternya dirangkum pada Tabel II.

$$T(s) = \frac{C(s)G(s)}{[1 + C(s)G(s)]} \quad (9)$$

#### a) Penalaan Manual (*trial and error*)

Pada penalaan manual, nilai parameter ditentukan secara bertahap berdasarkan pengamatan respons sistem. Penguatan proporsional dinaikkan terlebih dahulu untuk mempercepat respons, kemudian aksi integral ditambahkan untuk menghilangkan galat keadaan tunak, dan terakhir aksi derivatif disetel untuk meredam osilasi. Melalui proses ini diperoleh kombinasi  $K_p = 100$ ,  $K_i = 200$ , dan  $K_d = 10$  yang memberikan respons cukup cepat dengan overshoot kecil.

#### b) Metode Ziegler–Nichols

Metode Ziegler–Nichols osilasi kontinu diterapkan dengan terlebih dahulu menentukan penguatan kritis ( $K_u$ ) dan periode osilasi ( $T_u$ ) pada saat sistem berada di ambang osilasi berkelanjutan. Pada penelitian ini diperoleh  $K_u = 180$  dan  $T_u = 0,8$  s. Parameter PID kemudian dihitung melalui rumus baku Ziegler–Nichols berikut.

$$Kp = 0,6K_u; Ki = \frac{2Kp}{T_u}; Kd = \frac{KpT_u}{8} \quad (10)$$

Substitusi  $K_u = 180$  dan  $T_u = 0,8$  s ke dalam persamaan tersebut menghasilkan nilai-nilai berikut secara eksplisit.

$$Kp = 0,6 \times 180 = 108; Ki = 2 \times \frac{108}{0,8} = 270; Kd = 108 \times \frac{0,8}{8} = 10,8 \quad (11)$$

Dengan demikian diperoleh  $K_p = 108$ ,  $K_i = 270$ , dan  $K_d = 10,8$ .

#### c) PID Tuner MATLAB/Simulink

Metode ketiga memanfaatkan fitur PID Tuner pada MATLAB/Simulink. Berdasarkan model linear plant, perangkat lunak secara otomatis mencari kombinasi parameter yang menyeimbangkan kecepatan respons (response time) dan kekokohan terhadap gangguan (robustness). Proses optimasi ini menghasilkan parameter  $K_p = 120$ ,  $K_i = 250$ , dan  $K_d = 15$ .

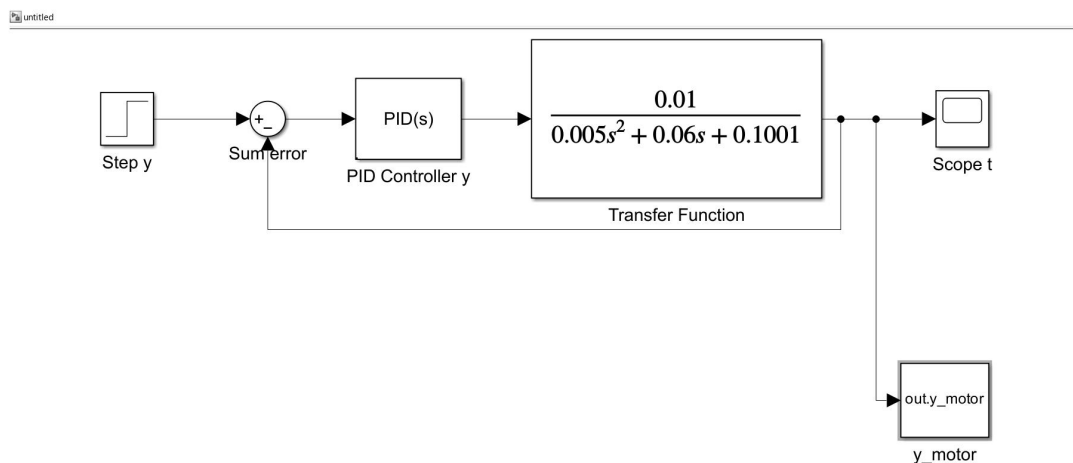
TABEL II: Nilai parameter pengendali PID hasil ketiga metode penalaan



| Metode penalaan     | $K_p$ | $K_i$ | $K_d$ |
|---------------------|-------|-------|-------|
| PID Manual          | 100   | 200   | 10    |
| PID Ziegler–Nichols | 108   | 270   | 10,8  |
| PID Tuner MATLAB    | 120   | 250   | 15    |

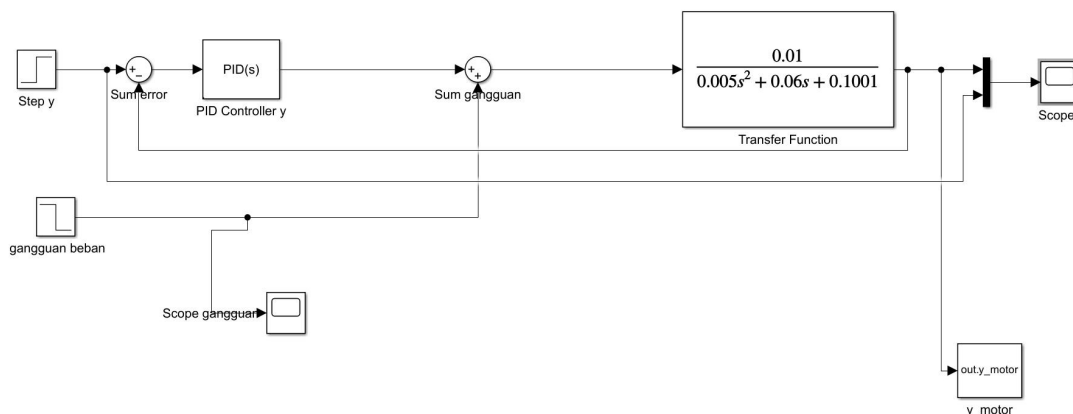
## 2.4 Skenario Simulasi dan Pemodelan Gangguan Beban

Sistem kendali diimplementasikan dalam bentuk diagram blok Simulink yang terdiri atas sumber acuan langkah (step), titik penjumlahan galat, blok pengendali PID, blok fungsi alih motor DC, dan blok pengamat (scope) untuk merekam kecepatan keluaran. Konfigurasi dasar lingkaran tertutup tanpa gangguan ditunjukkan pada Gambar 2.



Gambar 2. Diagram blok lingkaran tertutup kendali kecepatan motor DC di Simulink

Untuk menguji ketahanan sistem, sebuah gangguan beban dimodelkan sebagai sinyal langkah yang dijumlahkan pada keluaran pengendali sebelum masuk ke plant. Gangguan ini bernilai nol pada awal simulasi dan kemudian berubah menjadi sebesar  $-5$  satuan pada detik kedua, sehingga menyerupai pembebanan mendadak yang dialami motor saat beroperasi. Konfigurasi dengan injeksi gangguan beban diperlihatkan pada Gambar 3. Simulasi dijalankan selama 5 detik dengan acuan kecepatan bernilai satu satuan.



Gambar 3. Diagram blok sistem kendali dengan injeksi gangguan beban pada detik kedua

## 2.5 Indeks Kinerja dan Kriteria Evaluasi

Kinerja setiap metode dievaluasi melalui beberapa indikator. Karakteristik respons transien meliputi rise time, settling time, overshoot maksimum, dan peak time yang diperoleh dari respons langkah sistem lingkaran tertutup. Kestabilan diperiksa menggunakan kriteria Routh–Hurwitz pada persamaan karakteristik. Ketelitian keadaan tunak dinilai dari galat keadaan tunak terhadap masukan langkah. Selain itu, kualitas



keseluruhan respons diukur melalui tiga indeks galat berbasis integral, yaitu Integral of Absolute Error (IAE), Integral of Square Error (ISE), dan Integral of Time-weighted Absolute Error (ITAE), yang masing-masing dirumuskan sebagai berikut.

$$IAE = \int |e(t)| dt \quad (12)$$

$$ISE = \int e^2(t) dt \quad (13)$$

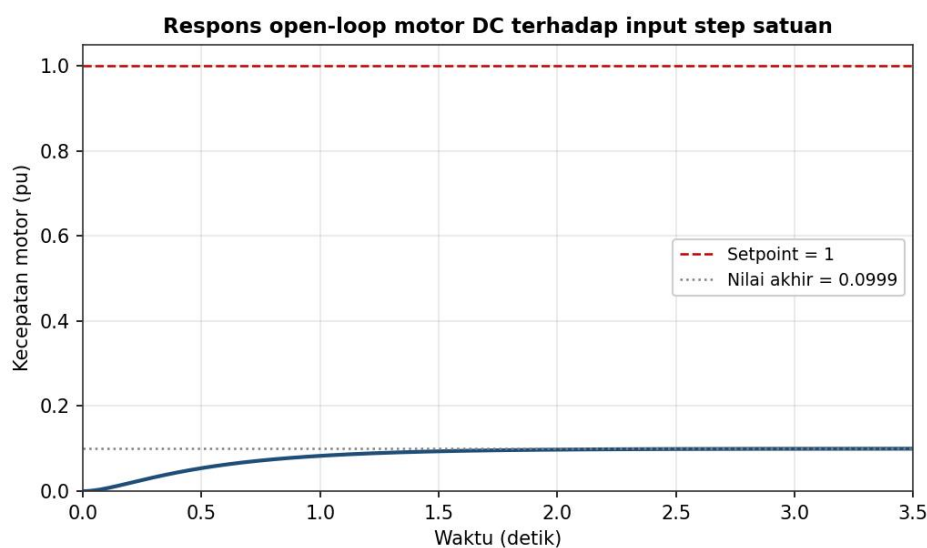
$$ITAE = \int t|e(t)| dt \quad (14)$$

dengan  $e(t)$  menyatakan galat sistem, yaitu selisih antara nilai acuan  $r(t)$  dan keluaran  $y(t)$ . Karena hasil simulasi MATLAB berupa data diskret dengan selang cuplikan  $\Delta t = 0,001$  detik selama 0 hingga 5 detik, integral pada ketiga indeks tersebut dihipotesiskan secara numerik melalui penjumlahan diskret. Nilai IAE, ISE, dan ITAE yang lebih kecil menandakan kinerja kendali yang lebih baik karena galat total sepanjang waktu lebih rendah.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Analisis Respons Open-Loop

Sebelum pengendali diterapkan, perilaku alami motor DC diamati terlebih dahulu melalui respons open-loop terhadap masukan langkah satuan. Berdasarkan fungsi alih pada Persamaan (7), penguatan searah (DC gain) sistem diperoleh dengan mengevaluasi fungsi alih pada  $s = 0$ , yaitu 0,01 dibagi 0,1001 atau sekitar 0,0999. Artinya, jika diberi acuan sebesar satu satuan, keluaran motor secara alami hanya mampu mencapai nilai sekitar 0,0999. Respons open-loop ini ditampilkan pada Gambar 4.



**Gambar 4.** Respons open-loop motor DC terhadap masukan langkah satuan

Selisih yang sangat besar antara keluaran dan acuan menghasilkan galat keadaan tunak sekitar 90,01%. Hal ini menegaskan bahwa motor DC tanpa pengendali sama sekali tidak mampu mengikuti acuan kecepatan yang dikehendaki, sehingga kehadiran pengendali menjadi keharusan. Lebih lanjut, penyebut fungsi alih dapat dinormalisasi ke bentuk baku sistem orde dua untuk mengungkap karakteristik dinamikanya. Dari proses ini diperoleh frekuensi alami tak teredam sebesar 4,474 rad/s dan rasio redaman sebesar 1,341. Karena rasio redaman bernilai lebih besar dari satu, sistem open-loop bersifat teredam-lebih (overdamped);

p-ISSN: 2356-0533; e-ISSN: 2355-9195



responsnya memang tidak beresilasi, tetapi cenderung lambat dan tidak pernah mencapai acuan secara penuh. Karakteristik inilah yang hendak diperbaiki melalui penalaan PID.

### 3.2 Karakteristik Respons Transien Sistem Lingkaran Tertutup

Dengan menerapkan ketiga set parameter PID pada konfigurasi lingkaran tertutup, diperoleh fungsi alih sistem keseluruhan untuk masing-masing metode. Substitusi parameter ke dalam Persamaan (9) menghasilkan fungsi alih lingkaran tertutup berikut. Untuk penalaan manual,

$$T_{manual(s)} = \frac{(0,1s^2 + 1 + 2)}{0,005s^3 + 0,16s^2 + 1,1001s + 2} \quad (15)$$

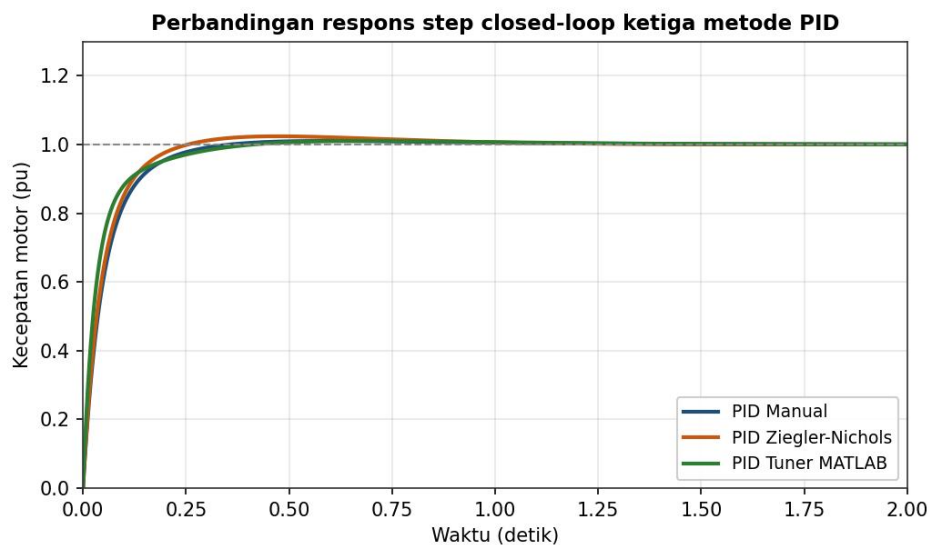
untuk metode Ziegler–Nichols,

$$T_{ZN(s)} = \frac{(0,108s^2 + 1,08s + 2,7)}{0,005s^3 + 0,168s^2 + 1,1801s + 2,7} \quad (16)$$

dan untuk penalaan PID Tuner,

$$T_{tuner(s)} = \frac{(0,15s^2 + 1,2s + 2,5)}{(0,005s^3 + 0,21s^2 + 1,3001s + 2,5)} \quad (17)$$

Respons langkah dari ketiga sistem lingkaran tertutup tersebut disimulasikan dan dibandingkan pada Gambar 5. Terlihat jelas bahwa penerapan pengendali PID mengubah perilaku sistem secara drastis dibandingkan kondisi open-loop: keluaran kini mampu mencapai acuan dengan cepat dan galat keadaan tunak praktis hilang.



**Gambar 5.** Perbandingan respons langkah lingkaran tertutup ketiga metode penalaan PID

Ringkasan kuantitatif karakteristik respons transien ketiga metode disajikan pada Tabel III. Data ini memungkinkan perbandingan yang lebih cermat di antara ketiga metode.

TABEL III: Perbandingan karakteristik respons transien

| Metode | Rise time (s) | Settling time (s) | Overshoot (%) | Peak time (s) |
|--------|---------------|-------------------|---------------|---------------|
|--------|---------------|-------------------|---------------|---------------|



| Metode              | Rise time (s) | Settling time (s) | Overshoot (%) | Peak time (s) |
|---------------------|---------------|-------------------|---------------|---------------|
| PID Manual          | 0,1324        | 0,2570            | 1,03          | 0,5932        |
| PID Ziegler–Nichols | 0,1173        | 0,6235            | 2,39          | 0,4709        |
| PID Tuner MATLAB    | 0,1105        | 0,2871            | 0,92          | 0,5767        |

Dari Tabel III tampak bahwa PID Tuner MATLAB memberikan rise time tercepat, yaitu 0,1105 detik, yang menunjukkan respons awal paling cepat di antara ketiga metode. Sementara itu, metode Ziegler–Nichols memiliki peak time tercepat, yaitu 0,4709 detik, tetapi disertai overshoot terbesar sebesar 2,39% dan settling time paling lama, yaitu 0,6235 detik. Pola ini sejalan dengan karakteristik metode Ziegler–Nichols yang memang dikenal cenderung menghasilkan respons cepat tetapi berosilasi. Sebaliknya, PID Tuner menghasilkan overshoot terkecil sebesar 0,92% dengan rise time yang juga tergolong cepat (0,1105 detik), mencerminkan kompromi yang baik antara kecepatan dan kehalusan respons. Adapun penalaan manual menonjol pada settling time tersingkat, yaitu 0,2570 detik, dengan overshoot yang masih rendah (1,03%).

Perbandingan ini memperlihatkan adanya pertukaran (trade-off) yang khas dalam perancangan kendali. Respons yang sangat cepat sebagaimana dihasilkan Ziegler–Nichols harus dibayar dengan osilasi dan waktu penetapan yang lebih panjang, sedangkan PID Tuner dan penalaan manual lebih mengutamakan kestabilan respons. Untuk aplikasi pengaturan kecepatan motor yang menuntut gerakan halus dan minim lonjakan, overshoot kecil yang ditawarkan PID Tuner maupun penalaan manual jelas lebih menguntungkan.

### 3.3 Analisis Kestabilan dengan Kriteria Routh–Hurwitz

Kestabilan ketiga sistem lingkaran tertutup diuji menggunakan kriteria Routh–Hurwitz terhadap persamaan karakteristik orde tiga dari masing-masing fungsi alih pada Persamaan (15)–(17). Untuk polinomial karakteristik berbentuk  $a_3s^3 + a_2s^2 + a_1s + a_0$ , tabel Routh disusun dalam empat baris. Sistem dinyatakan stabil apabila seluruh elemen pada kolom pertama tabel bernilai positif. Array Routh–Hurwitz untuk ketiga metode penalaan disajikan pada Tabel IV.

TABEL IV : Array Routh–Hurwitz persamaan karakteristik ketiga metode PID

| Baris | PID Manual   | PID Ziegler–Nichols | PID Tuner MATLAB |
|-------|--------------|---------------------|------------------|
| $s^3$ | 0,005 1,1001 | 0,005 1,1801        | 0,005 1,3001     |
| $s^2$ | 0,16 2,0000  | 0,168 2,7000        | 0,21 2,5000      |
| $s^1$ | 1,0376       | 1,1002              | 1,2382           |
| $s^0$ | 2,0000       | 2,7000              | 2,5000           |

Dari Tabel IV terlihat bahwa seluruh elemen pada kolom pertama array Routh–Hurwitz bernilai positif untuk ketiga metode, yakni: 0,005; 0,16/0,168/0,21; 1,0376/1,1002/1,2382; dan 2,0/2,7/2,5. Karena tidak ada perubahan tanda pada kolom pertama, dapat disimpulkan bahwa ketiga sistem lingkaran tertutup bersifat stabil asimtotik. Kondisi ini konsisten dengan respons langkah pada Gambar 5 yang seluruhnya konvergen ke nilai acuan tanpa osilasi yang bertambah besar.

### 3.4 Galat Keadaan Tunak

Galat keadaan tunak terhadap masukan langkah satuan dianalisis menggunakan teorema nilai akhir. Untuk sistem lingkaran tertutup dengan masukan langkah  $R(s) = 1/s$ , galat dalam domain Laplace adalah  $E(s) = [1 - T(s)]/s$ , sehingga galat keadaan tunak diperoleh dengan:

$$e_{ss} = \lim_{s \rightarrow 0} s \cdot E(s) = \lim_{s \rightarrow 0} \frac{s}{s(1 + C(s)G(s))} = \frac{1}{1 + C(0)G(0)} \quad (18)$$

Karena setiap pengendali PID mengandung aksi integral, suku  $K_i/s$  dalam  $C(s)$  menyebabkan  $C(0) \rightarrow \infty$ . Dengan demikian  $C(0) \cdot G(0) \rightarrow \infty$ , dan fungsi alih lingkaran tertutup pada frekuensi nol menjadi:

$$(19)$$



$$T(0) = \frac{C(0)G(0)}{(1 + C(0)G(0))} = \frac{\infty}{(1 + \infty)} = 1 \Rightarrow e_{ss} = 1 - T(0) = 0$$

Hasil ini berlaku untuk ketiga metode penalaan, artinya secara analitis galat keadaan tunak selalu nol selama pengendali memiliki komponen integral. Sebagai verifikasi, galat keadaan tunak juga diamati dari hasil simulasi numerik, sebagaimana dirangkum pada Tabel 5. Galat hasil simulasi pada Tabel V bernilai sangat kecil dan secara praktis dapat dianggap nol, sehingga konsisten dengan prediksi analitis. Perbedaan tipis di antara ketiga nilai tersebut lebih disebabkan oleh sisa numerik proses integrasi diskret dan perbedaan kecepatan konvergensi masing-masing respons, bukan oleh adanya galat tunak yang sesungguhnya. Dengan demikian, ditinjau dari ketelitian keadaan tunak, ketiga metode penalaan sama-sama memuaskan.

TABEL V : Galat keadaan tunak terhadap masukan langkah

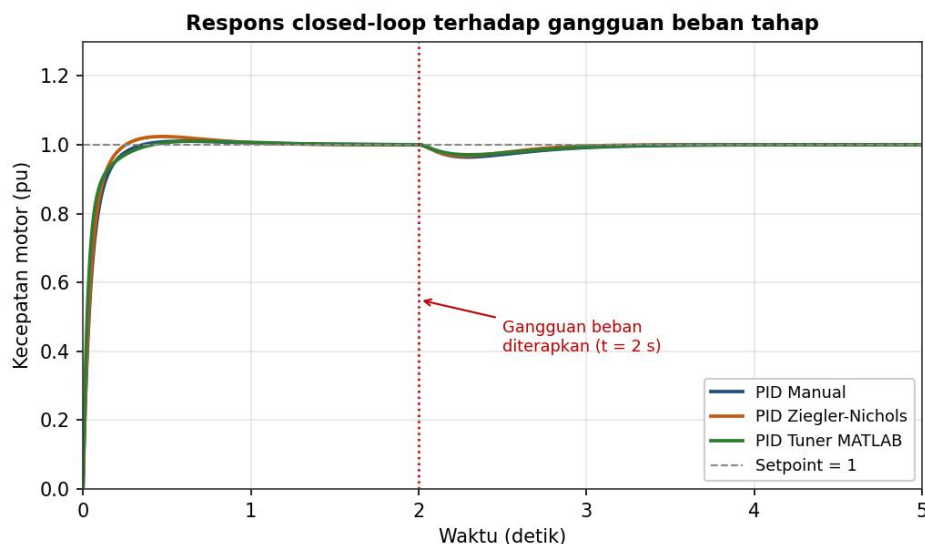
| Metode              | Galat teoretis | Galat hasil simulasi |
|---------------------|----------------|----------------------|
| PID Manual          | 0              | 0,000409             |
| PID Ziegler–Nichols | 0              | 0,001956             |
| PID Tuner MATLAB    | 0              | 0,005568             |

### 3.5 Respons terhadap Gangguan Beban

Pengujian terpenting dalam penelitian ini adalah mengamati perilaku sistem ketika dikenai gangguan beban. Pada sistem berumpan-balik, keluaran terdiri atas dua kontribusi, yaitu kontribusi akibat acuan dan kontribusi akibat gangguan, sebagaimana dinyatakan oleh hubungan berikut.

$$Y(s) = \left[ \frac{C(s)G(s)}{(1 + C(s)G(s))} \right] R(s) + \left[ \frac{G(s)}{(1 + C(s)G(s))} \right] D(s) \quad (20)$$

Suku pertama menggambarkan kemampuan sistem mengikuti acuan, sedangkan suku kedua menggambarkan sejauh mana gangguan  $D(s)$  mempengaruhi keluaran. Pengendali yang baik akan memperkecil pengaruh suku kedua sehingga penyimpangan akibat gangguan dapat ditekan dan keluaran cepat kembali ke acuan. Dalam simulasi, gangguan beban sebesar  $-5$  satuan diterapkan pada detik kedua. Respons ketiga metode terhadap skenario ini ditampilkan pada Gambar 6.



Gambar 6. Respons lingkaran tertutup terhadap gangguan beban langkah pada detik kedua

Gambar 6 memperlihatkan bahwa pada dua detik pertama seluruh metode berhasil membawa keluaran menuju acuan sesuai karakteristik transiennya masing-masing. Begitu gangguan diterapkan pada



detik kedua, keluaran sempat menyimpang dari acuan, namun aksi integral pada ketiga pengendali segera bekerja mengoreksi penyimpangan tersebut hingga keluaran kembali ke nilai satu. Hal ini membuktikan bahwa ketiga metode mampu menolak gangguan beban (disturbance rejection) dan memulihkan kecepatan motor ke nilai yang dikehendaki. Perbedaan terletak pada besar penyimpangan dan kecepatan pemulihan: pengendali dengan respons yang lebih kokoh, yaitu PID Tuner dan penalaan manual, menghasilkan penyimpangan yang lebih kecil serta pemulihan yang lebih mulus dibandingkan Ziegler–Nichols yang penyimpangannya sedikit lebih terasa sebelum stabil kembali.

### 3.6 Pembahasan Komparatif

Untuk melengkapi analisis komparatif, kualitas keseluruhan respons ketiga metode dievaluasi secara kuantitatif menggunakan indeks kinerja berbasis integral, yaitu IAE, ISE, dan ITAE. Ketiga indeks ini dihitung secara numerik menggunakan metode trapesium terhadap sinyal galat selama interval simulasi 0–5 detik dengan selang cuplikan 0,001 detik, sesuai dengan Persamaan (12)–(14). Hasil perhitungan dirangkum pada Tabel 6.

TABEL VI : Indeks kinerja IAE, ISE, dan ITAE ketiga metode penalaan PID

| Metode              | IAE    | ISE    | ITAE   |
|---------------------|--------|--------|--------|
| PID Manual          | 0,0640 | 0,0268 | 0,0095 |
| PID Ziegler–Nichols | 0,0635 | 0,0247 | 0,0112 |
| PID Tuner MATLAB    | 0,0542 | 0,0191 | 0,0097 |

Dari Tabel VI terlihat bahwa PID Tuner MATLAB menghasilkan nilai IAE dan ISE terkecil, yaitu masing-masing 0,0542 dan 0,0191. Hal ini menunjukkan bahwa PID Tuner paling efektif dalam menekan galat absolut kumulatif dan galat besar selama proses transien. Namun, nilai ITAE terkecil diperoleh oleh penalaan manual, yaitu 0,0095, sedikit lebih rendah dibandingkan PID Tuner sebesar 0,0097. Nilai IAE yang lebih kecil menunjukkan bahwa galat absolut kumulatif sepanjang waktu simulasi paling rendah, sedangkan ISE yang terkecil mengindikasikan bahwa galat besar (yang dikuadratkan) paling berhasil ditekan. Nilai ITAE yang rendah membuktikan bahwa galat yang tersisa pada fase akhir simulasi, yang diboboti lebih besar oleh faktor waktu juga paling kecil, mencerminkan kualitas pemulihan yang cepat dan mulus.

Metode Ziegler–Nichols memiliki IAE dan ISE yang sedikit lebih baik dari penalaan manual, namun ITAE-nya paling besar (0,0112). Hal ini konsisten dengan karakteristik overshoot yang lebih tinggi dan settling time yang lebih panjang: galat pada fase awal memang kecil karena respons yang cepat, tetapi galat pada fase akhir (osilasi menuju keadaan tunak) lebih besar dan berlangsung lebih lama, sehingga membebani ITAE. Penalaan manual berada di antara keduanya dengan ITAE terkecil kedua (0,0095), mencerminkan settling time yang singkat meskipun rise time-nya tidak secepat ZN.

Apabila seluruh hasil dirangkum, terlihat gambaran yang utuh mengenai karakter masing-masing metode penalaan. Ditinjau dari kestabilan dan ketelitian keadaan tunak, ketiganya setara: semua sistem stabil menurut Routh–Hurwitz dan memiliki galat keadaan tunak yang praktis nol berkat aksi integral. Perbedaan yang menentukan justru muncul pada kualitas respons transien dan ketahanan terhadap gangguan.

Metode Ziegler–Nichols unggul dalam kecepatan respons awal dan menawarkan prosedur penalaan yang baku serta mudah diterapkan tanpa perangkat lunak khusus. Kelemahannya adalah overshoot yang besar dan settling time yang panjang, sehingga responsnya kurang halus. Untuk banyak aplikasi, hasil Ziegler–Nichols biasanya masih perlu disempurnakan lebih lanjut. Penalaan manual, di sisi lain, mampu memberikan settling time tersingkat dengan overshoot yang rendah, namun proses memperolehnya bergantung pada pengalaman perancang dan relatif memakan waktu karena bersifat coba-coba. Sementara itu, PID Tuner MATLAB/Simulink memberikan keseimbangan terbaik secara keseluruhan, yakni overshoot terkecil, rise time yang cepat, settling time yang baik, serta pemulihan gangguan yang mulus, dengan keunggulan tambahan berupa kemudahan dan kecepatan proses penalaan secara otomatis.

Dengan mempertimbangkan keseluruhan indikator, penalaan berbasis PID Tuner merupakan pilihan paling tepat untuk pengendalian kecepatan motor DC yang menuntut respons halus sekaligus tahan



gangguan. Penalaan manual menjadi alternatif yang sangat kompetitif apabila perangkat lunak penalaan otomatis tidak tersedia, sedangkan metode Ziegler–Nichols paling cocok digunakan sebagai titik awal perancangan yang cepat sebelum dilakukan penyempurnaan parameter lebih lanjut.

#### 4. KESIMPULAN DAN SARAN

Penelitian ini telah membandingkan tiga metode penalaan pengendali PID, yaitu penalaan manual, Ziegler–Nichols, dan PID Tuner MATLAB/Simulink, untuk pengendalian kecepatan motor DC yang dikenai gangguan beban. Hasil menunjukkan bahwa sistem open-loop motor DC tidak layak digunakan karena memiliki galat keadaan tunak mencapai sekitar 90% dan bersifat teredam-lebih sehingga lambat mencapai acuan. Penerapan pengendali PID dengan ketiga metode berhasil memperbaiki kinerja sistem secara signifikan, menghasilkan sistem yang stabil menurut kriteria Routh–Hurwitz dengan galat keadaan tunak yang praktis nol berkat aksi integral.

Dari sisi respons transien, PID Tuner memberikan rise time tercepat (0,1105 detik) sekaligus overshoot terkecil (0,92%). Metode Ziegler–Nichols memiliki peak time tercepat (0,4709 detik), tetapi disertai overshoot terbesar (2,39%) dan settling time terpanjang (0,6235 detik). Penalaan manual menghasilkan settling time tersingkat (0,2570 detik) dengan overshoot rendah (1,03%). Evaluasi indeks kinerja integral menunjukkan bahwa PID Tuner unggul pada indeks IAE dan ISE dengan nilai masing-masing 0,0542 dan 0,0191, sedangkan nilai ITAE terkecil justru diperoleh oleh penalaan manual sebesar 0,0095. Dengan demikian, tidak ada satu metode pun yang unggul pada seluruh parameter; PID Tuner lebih tepat dinilai sebagai metode dengan kompromi performa terbaik secara keseluruhan karena memadukan rise time tercepat, overshoot terkecil, serta respons terhadap gangguan yang halus, meskipun ITAE-nya sedikit di atas penalaan manual. Pada pengujian gangguan beban, ketiga metode mampu menolak gangguan dan memulihkan kecepatan ke acuan, dengan PID Tuner dan penalaan manual menunjukkan penyimpangan yang lebih kecil serta pemulihan yang lebih mulus.

Secara keseluruhan, penalaan berbasis PID Tuner MATLAB/Simulink direkomendasikan sebagai metode dengan kompromi performa terbaik karena menawarkan keseimbangan paling baik antara kecepatan, kehalusan respons, dan ketahanan terhadap gangguan, sementara penalaan manual tetap menjadi alternatif yang sangat kompetitif terutama ditinjau dari indeks ITAE. Penelitian selanjutnya dapat diarahkan pada pengujian dengan variasi profil gangguan yang lebih beragam, penerapan metode penalaan modern berbasis kecerdasan komputasional, serta validasi pada perangkat keras motor DC secara langsung untuk mengonfirmasi hasil simulasi yang diperoleh.

#### DAFTAR PUSTAKA

- [1] J. G. Ziegler and N. B. Nichols, "Optimum settings for automatic controllers," *Transactions of the ASME*, vol. 64, no. 8, pp. 759–768, 1942.
- [2] K. Ogata, *Modern Control Engineering*, 5th ed. Upper Saddle River, NJ: Prentice Hall, 2010.
- [3] N. S. Nise, *Control Systems Engineering*, 7th ed. Hoboken, NJ: John Wiley & Sons, 2015.
- [4] R. C. Dorf and R. H. Bishop, *Modern Control Systems*, 13th ed. New York, NY: Pearson, 2017.
- [5] K. J. Åström and T. Hägglund, *PID Controllers: Theory, Design, and Tuning*, 2nd ed. Research Triangle Park, NC: Instrument Society of America, 1995.
- [6] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 7th ed. Boston, MA: Pearson, 2015.
- [7] R. Krishnan, *Electric Motor Drives: Modeling, Analysis, and Control*. Upper Saddle River, NJ: Prentice Hall, 2001.
- [8] The MathWorks, Inc., "PID Controller Tuning in Simulink," *MATLAB & Simulink Documentation*, Natick, MA, 2023.
- [9] A. Ma'arif, R. Istiarno, and S. Sunardi, "Kontrol Proporsional Integral Derivatif (PID) pada Kecepatan Sudut Motor DC dengan Pemodelan Identifikasi Sistem dan Tuning," *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, vol. 9, no. 2, pp. 374–388, Apr. 2021, doi: 10.26760/elkomika.v9i2.374.
- [10] A. Ma'arif and N. R. Setiawan, "Control of DC Motor Using Integral State Feedback and Comparison with PID: Simulation and Arduino Implementation," *Journal of Robotics and Control (JRC)*, vol. 2, no. 5, pp. 456–461, Sep. 2021, doi: 10.18196/jrc.25122.
- [11] D. Saputra, A. Ma'arif, H. Maghfiroh, P. Chotikunann, and S. N. Rahmadhia, "Design and Application of PLC-based Speed Control for DC Motor Using PID with Identification System and MATLAB Tuner," *International Journal of Robotics and Control Systems*, vol. 3, no. 2, pp. 233–244, Apr. 2023, doi: 10.31763/ijrcs.v3i2.775.



- [12] R. S. Widagdo, F. D. Murdianto, and R. Wahyudianto, "Modelling and Analysis of Ziegler-Nichols and Chien-Hrones-Reswick Tuning PID on DC Motor Speed Control," *Jurnal Teknologi Elektro*, vol. 14, no. 1, pp. 27–33, Jan. 2023, doi: 10.22441/jte.2023.v14i1.005.
- [13] E. S. Rahayu, A. Ma'arif, and A. Cakan, "Particle Swarm Optimization (PSO) Tuning of PID Control on DC Motor," *International Journal of Robotics and Control Systems*, vol. 2, no. 2, pp. 435–447, 2022, doi: 10.31763/ijrcs.v2i2.476.
- [14] H. S. Dakheel, Z. B. Abdullah, and F. N. Abdullah, "Simulation Model of ANN and PID Controller for Direct Current Servo Motor by Using Matlab/Simulink," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 20, no. 4, pp. 736–742, Aug. 2022, doi: 10.12928/TELKOMNIKA.v20i4.23248.
- [15] Ş. Yıldırım, M. S. Bingöl, and S. Savaş, "Tuning PID controller parameters of the DC motor with PSO algorithm," *International Review of Applied Sciences and Engineering*, vol. 15, no. 3, pp. 281–286, 2024, doi: 10.1556/1848.2023.00698.
- [16] X. Han, "Comparative study on PID for DC motor speed regulation," *MATEC Web of Conferences*, vol. 404, art. 02003, 2024, doi: 10.1051/mateconf/202440402003.
- [17] A. M. Putra, H. Maradona, and R. A. Rohmah, "Comparison of Proportional Integral Derivative and Fuzzy Logic Controllers: A Literature Review on the Best Method for Controlling Direct Current Motor Speed," *International Journal of Robotics and Control Systems*, vol. 5, no. 1, pp. 240–265, 2025, doi: 10.31763/ijrcs.v5i1.1701.

