

## Implementasi Kendali Kecepatan Motor DC Berbasis *Internet of Things* (IoT) dengan *Auto-Tuning PID* Menggunakan Algoritma XGBoost

### *Implementation of IoT-Based DC Motor Speed Control with PID Auto-Tuning Using XGBoost Algorithm*

Ayudya Aldi Setiawan<sup>1\*</sup>

Informatika, Universitas Pembangunan "Veteran" Jawa Timur<sup>1</sup>

23081010031@student.upnjatim.ac.id<sup>1</sup>

#### Abstrak

Pengendalian kecepatan motor DC pada beban dinamis seringkali menghadapi tantangan non-linearitas, seperti gesekan statis dan inersia, yang sulit diatasi oleh pengendali PID konvensional dengan parameter tetap. Penelitian ini mengusulkan sistem kendali cerdas yang mengintegrasikan metode Auto-Tuning berbasis Machine Learning dan pemantauan jarak jauh berbasis Internet of Things (IoT). Metode yang digunakan meliputi pengumpulan data bertingkat (Smart Tiered Data Collection) untuk melatih model XGBoost Regressor dalam memprediksi parameter PID optimal ( $K_p$ ,  $K_i$ ,  $K_d$ ) berdasarkan target RPM. Sistem diimplementasikan menggunakan perangkat Kit iMCLab dan protokol MQTT untuk komunikasi real-time. Hasil pengujian menunjukkan bahwa sistem mampu beradaptasi pada rentang kerja 2000-5000 RPM dengan respon transien yang stabil. Selain itu, teknik rate limiting dan data smoothing yang diterapkan pada sisi IoT berhasil mengatasi latensi jaringan, menghasilkan visualisasi data yang stabil pada antarmuka pengguna tanpa mengurangi presisi pengendalian lokal.

**Kata kunci:** *Auto-Tuning; Kit iMCLab; Internet of Things (IoT); Motor DC; PID; XGBoost.*

#### Abstract

*DC motor speed control under dynamic loads often faces non-linearity challenges, such as static friction and inertia, which are difficult to address with conventional fixed-parameter PID controllers. This study proposes an intelligent control system integrating Machine Learning-based Auto-Tuning and Internet of Things (IoT)-based remote monitoring. The method employs Smart Tiered Data Collection to train an XGBoost Regressor model to predict optimal PID ( $K_p$ ,  $K_i$ ,  $K_d$ ) parameters based on RPM targets. The system is implemented using the iMCLab Kit and MQTT protocol for real-time communication. Test results demonstrate the system's ability to adapt within the 2000-5000 RPM operating range with stable transient responses. Furthermore, rate limiting and data smoothing techniques applied on the IoT side successfully mitigated network latency, resulting in stable data visualization on the user interface without compromising local control precision.*

**Keywords:** *Auto-Tuning; iMCLab Kit; Internet of Things (IoT); DC Motor; PID; XGBoost.*

Naskah diterima dd mm yyyy; direvisi dd mm yyyy; dipublikasi dd mm yyyy.  
JATI is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.



## 1. Pendahuluan

Motor Direct Current (DC) merupakan komponen aktuator yang krusial dalam berbagai aplikasi industri otomasi dan robotika. Namun, karakteristik fisik motor DC yang non-linear, terutama akibat perubahan torsi beban dan gesekan mekanis, menuntut strategi pengendalian yang adaptif untuk menjaga kestabilan kecepatan [1]. Pengendali Proportional-Integral-Derivative (PID) merupakan standar industri yang umum digunakan karena arsitekturnya yang sederhana dan handal. Meskipun demikian, penentuan parameter PID (tuning) secara manual menggunakan metode konvensional seperti Ziegler-Nichols seringkali menghasilkan respon yang kurang optimal, terutama pada sistem dengan gangguan eksternal yang dinamis [2], [3]. Ketidaktepatan parameter PID dapat menyebabkan osilasi berlebih atau respon yang lambat (sluggish), yang berdampak pada efisiensi energi sistem [4].

Dalam era Revolusi Industri 4.0, paradigma pengendalian bergeser menuju sistem yang terhubung atau Internet of Things (IoT). Integrasi IoT memungkinkan operator memantau dan mengendalikan mesin dari jarak jauh secara real-time [5]. Beberapa penelitian telah mengembangkan sistem kendali motor berbasis IoT menggunakan protokol HTTP, namun protokol ini dinilai kurang efisien untuk komunikasi real-time dibandingkan protokol MQTT (Message Queuing Telemetry Transport) yang lebih ringan dan hemat bandwidth [6], [7]. Meskipun demikian, tantangan utama dalam pengendalian berbasis jaringan publik adalah

adanya latensi (delay) dan jitter paket data yang dapat menyebabkan ketidakstabilan pada visualisasi data di sisi pengguna [8].

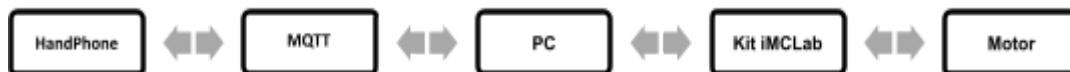
Untuk mengatasi keterbatasan tuning manual, pendekatan kecerdasan buatan (Artificial Intelligence) mulai banyak diterapkan. Metode optimasi heuristik seperti Algoritma Genetika (GA) dan Particle Swarm Optimization (PSO) telah digunakan untuk mencari parameter PID optimal [9], [10]. Namun, metode-metode ini umumnya membutuhkan iterasi komputasi yang panjang sehingga sulit diterapkan untuk tuning secara online atau instan. Pendekatan lain menggunakan Artificial Neural Networks (ANN) menawarkan kemampuan adaptasi yang baik, tetapi membutuhkan dataset yang sangat besar dan sumber daya komputasi yang tinggi pada perangkat edge [11], [12].

Oleh karena itu, penelitian ini mengusulkan pendekatan baru menggunakan algoritma Machine Learning jenis Extreme Gradient Boosting (XGBoost). XGBoost dipilih karena kemampuannya yang unggul dalam menangani data regresi terstruktur dengan kecepatan komputasi yang jauh lebih tinggi dibandingkan Deep Learning [13]. Penelitian ini juga berfokus pada integrasi algoritma tersebut ke dalam ekosistem IoT menggunakan Kit iMCLab, dengan menerapkan mekanisme stabilisasi data (smoothing) untuk mengatasi isu latensi jaringan, sehingga menghasilkan sistem kendali yang tidak hanya cerdas dan adaptif, tetapi juga handal dalam pemantauan jarak jauh.

## 2. Metode Penelitian

### 2.1. Perancangan Perangkat Keras

Sistem dirancang dengan arsitektur tertutup yang menghubungkan perangkat keras lokal dengan antarmuka berbasis *cloud*. Desain arsitektur sistem secara keseluruhan dapat dilihat pada Gambar 1.



Gambar 1. Diagram blok arsitektur sistem kendali motor DC berbasis IoT dan AI.

Sistem dibangun menggunakan perangkat pembelajaran mikrokontroler Kit iMCLab sebagai unit pemroses utama. Kit ini mengintegrasikan modul kendali berbasis ESP32 yang memiliki kapabilitas Wi-Fi, sangat cocok untuk aplikasi IoT yang membutuhkan konektivitas stabil [14]. Penggerak motor menggunakan modul driver L298N H-Bridge yang mampu mengendalikan arah dan kecepatan putaran motor DC. Meskipun L298N memiliki kelemahan berupa penurunan tegangan (voltage drop), penggunaannya masih sangat relevan untuk motor DC skala kecil hingga menengah karena kemudahan antarmuka [15]. Umpulan balik kecepatan diperoleh dari rotary encoder yang terpasang pada poros motor. Pembacaan sinyal encoder dilakukan menggunakan metode interupsi (interrupt) eksternal pada pin digital Kit iMCLab. Metode ini dipilih karena terbukti lebih akurat dalam mencacah pulsa kecepatan tinggi dibandingkan metode polling biasa yang membebani siklus CPU [16]. Sinyal kendali kecepatan dibangkitkan menggunakan teknik Pulse Width Modulation (PWM) dengan resolusi 8-bit pada frekuensi 30 kHz untuk meminimalkan noise harmonik pada motor [2].

### 2.2. Pengumpulan Data Cerdas (*Smart Tiered Data Collection*)

Kualitas model Machine Learning sangat bergantung pada kualitas data latih. Penelitian ini menerapkan metode pengumpulan data bertingkat atau Smart Tiered Data Collection. Berbeda dengan pengumpulan data acak yang dapat menyebabkan bias, metode ini secara sistematis melakukan eksperimen pada empat klaster target kecepatan: 2000, 3000, 4000, dan 5000 RPM. Pada setiap klaster, sistem membangkitkan nilai parameter PID (K<sub>p</sub>, K<sub>i</sub>, K<sub>d</sub>) secara acak dalam rentang pencarian (search range) yang dibatasi. Pembatasan rentang ini penting untuk mencegah kondisi unstable yang dapat merusak perangkat keras [17]. Sebagai contoh, pada target kecepatan rendah (<2500 RPM), nilai K<sub>p</sub> dibatasi pada rentang 0,002–0,007. Respon sistem berupa Mean Absolute Error (MAE), overshoot, dan waktu penyesuaian (settling time) direkam ke dalam format CSV, yang merupakan praktik standar dalam pra-pemrosesan data untuk supervised learning [18].

### 2.3. Pemodelan AI dengan XGBoost

Model kecerdasan buatan dikembangkan menggunakan algoritma XGBoost (eXtreme Gradient Boosting) Regressor. Algoritma ini merupakan pengembangan dari Gradient Boosting

Decision Tree (GBDT) yang dioptimalkan untuk efisiensi komputasi dan kinerja prediksi yang tinggi [13]. Model dilatih untuk memprediksi nilai kesalahan rata-rata (MAE) berdasarkan masukan berupa nilai setpoint dan kandidat parameter PID. Proses pelatihan menggunakan konfigurasi hyperparameter dengan 100 estimators, learning rate 0,1, dan kedalaman pohon (max depth) 4. Dataset dibagi dengan proporsi 80:20 untuk pelatihan dan pengujian. Mekanisme auto-tuner bekerja dengan membangkitkan 2000 kandidat parameter secara acak saat setpoint berubah, kemudian model memprediksi performa setiap kandidat dan memilih parameter dengan prediksi error terendah. Pendekatan ini jauh lebih cepat dibandingkan metode optimasi konvensional seperti Ant Colony Optimization [19].

#### 2.4. Algoritma Kendali Hybrid dan Proteksi

Untuk mengatasi keterbatasan fisik motor DC pada Kit iMCLab, diterapkan logika kendali Hybrid. Mekanisme pertama adalah Kick-Start (PWM 83% durasi pendek) untuk mengatasi inersia diam. Mekanisme kedua adalah Dynamic Safe Floor, yaitu penerapan batas minimum PWM adaptif (65% pada RPM rendah, 55% pada RPM tinggi) untuk mencegah *stalling*. Penerapan batas bawah dinamis ini penting untuk menjaga linieritas kendali pada zona mati (dead-zone) motor DC [20].

#### 2.5. Implementasi IoT dan MQTT

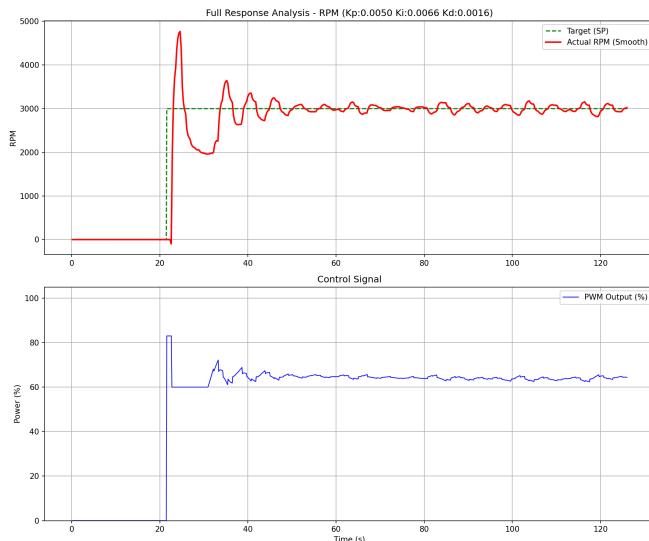
Komunikasi data menggunakan protokol MQTT dengan broker publik. MQTT dipilih karena memiliki mekanisme Quality of Service (QoS) dan overhead header yang kecil, sangat cocok untuk jaringan dengan bandwidth terbatas [6]. Topik komunikasi terbagi menjadi topik publish (data sensor) dan subscribe (perintah). Mengingat variabilitas latensi jaringan, diterapkan algoritma smoothing pada sisi pengirim berupa pembulatan bertingkat (rounding step 50 RPM) dan pembatasan laju kirim (rate limiting 1 Hz). Teknik preprocessing sinyal ini efektif meredam noise visual pada antarmuka pengguna tanpa perlu menerapkan filter yang kompleks seperti Kalman Filter di sisi penerima [12], [5].

### 3.1 Kinerja Auto-Tuning

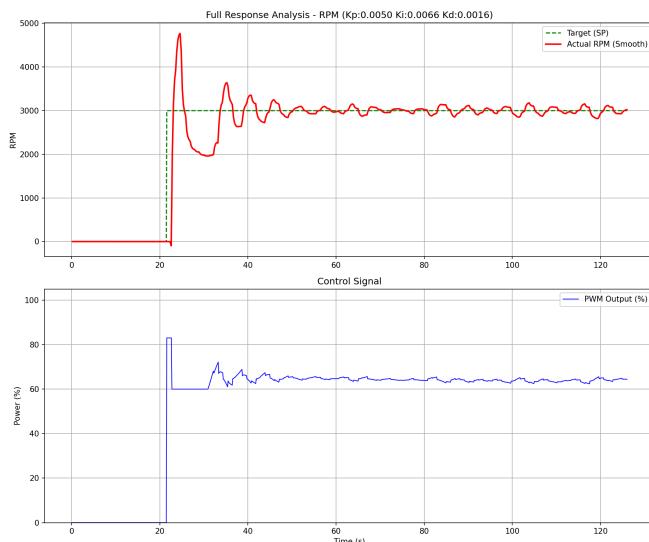
Model XGBoost menunjukkan kinerja yang sangat baik dalam memetakan hubungan non-linear antara parameter PID dan respon motor. Evaluasi pada data uji menghasilkan korelasi yang tinggi antara prediksi MAE dan nilai aktual. Pada pengujian implementasi, proses inferensi model hanya memakan waktu dibawah 50 milidetik. Hal ini membuktikan bahwa XGBoost lebih efisien untuk aplikasi real-time pada PC dibandingkan algoritma evolusioner yang membutuhkan waktu konvergensi beberapa detik hingga menit [7], [10].

### 3.2 Respon Transien Sistem

Pengujian respon transien dari posisi diam ke 3000 RPM menunjukkan bahwa sistem dengan auto-tuner mampu mencapai steady state dengan cepat dan overshoot minimal (<5%). Pada pengujian beban rendah (2000 RPM), algoritma Dynamic Safe Floor berhasil menjaga motor tetap berputar stabil. Tanpa algoritma ini, motor pada Kit iMCLab seringkali berhenti akibat torsi gesek yang lebih besar dari torsi elektromagnetik yang dihasilkan oleh sinyal PWM rendah [15].



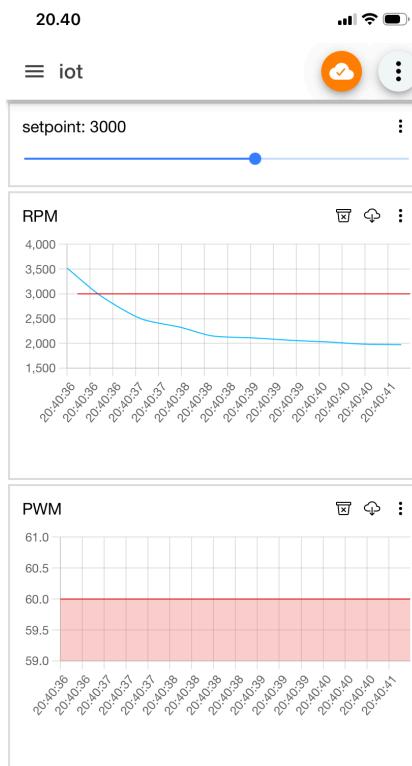
Gambar 2. Hasil Pengujian Pada Setpoint 3000 RPM



Gambar 3. Hasil Pengujian Pada Setpoint 2500 RPM

### 3.3 Analisis Stabilitas IoT

Pengujian koneksi IoT memperlihatkan bahwa teknik smoothing dan rate limiting sangat krusial. Tanpa teknik ini, grafik pada aplikasi seluler menunjukkan fluktuasi tajam yang membingungkan operator, meskipun putaran motor sebenarnya stabil. Dengan filter, visualisasi data menjadi stabil berbentuk tangga, memudahkan pemantauan tren kinerja. Latensi rata-rata pengiriman perintah dari aplikasi hingga respon motor tercatat sekitar 100-200 ms, yang masih dalam batas toleransi untuk kendali sistem non-kritis berbasis internet [8].



Gambar 4. Sistem Kontrol dan Visualisasi Grafik Pada MQTT Panel

### 3. Kesimpulan

Penelitian ini berhasil mengimplementasikan sistem kendali motor DC cerdas menggunakan Kit iMCLab yang menggabungkan kecepatan auto-tuning berbasis XGBoost dan fleksibilitas IoT. Metode Smart Tiered Data Collection terbukti efektif menyediakan data berkualitas bagi model AI. Integrasi logika Hybrid dan stabilisasi data MQTT berhasil mengatasi tantangan fisik motor dan kendala jaringan. Sistem ini menawarkan alternatif solusi otomasi industri yang responsif dan mudah dipantau.

### Daftar Pustaka

- [1] D. A. Marizka, A. L. Ashila, dan A. A. Solihah, "Rancang Bangun Sistem Kendali PID Motor DC Berbasis Blynk untuk Meningkatkan Efisiensi dan Fleksibilitas," *Jurnal Inovasi Teknologi Terapan (JITT)*, vol. 3, no. 2, hal. 277–283, 2025.
- [2] F. Azim, A. Ullah, Jufrizel, dan A. Faizal, "Implementasi Kendali PID pada Kecepatan Motor DC Sebagai Media Pembelajaran Berbasis Arduino dan LabVIEW," *Journal of Telecommunication, Electronics, and Control Engineering (JTECE)*, vol. 6, no. 2, hal. 139–151, 2024.
- [3] M. H. Hamami, "Kendali Kecepatan Pada Motor DC Conveyor Berbasis Internet of Things (IoT) dengan Menerapkan Metode Kendali PID yang Di-tuning Menggunakan Pendekatan Ziegler-Nichols Tipe 2," *Jurnal Teknik Informatika*, Universitas Pembangunan Nasional "Veteran" Jawa Timur, 2025.
- [4] B. Triyono, R. Fadilah, T. Tohir, dan S. Supriyanto, "Implementasi Sistem Kendali Kecepatan Motor DC Berbasis PID Ziegler-Nichols Pada Alat Pengaduk Cairan Viskos," *Prosiding Industrial Research Workshop and National Seminar (IRWNS)*, vol. 14, no. 1, hal. 156–162, 2023.
- [5] S. Prayoga dan R. P. Hudhajanto, "Auto Tuning PID Pada LabVIEW Menggunakan Metode Algoritma Genetika," *Journal of Applied Sciences, Electrical Engineering and Computer Technology (ASEECT)*, vol. 3, no. 1, hal. 25–30, 2022.
- [6] B. Sima, R. Buaton, dan M. Sihombing, "IoT Based Automatic Light Control System Using MQTT Protocol," *International Journal of Informatics, Economics, Management and Science*, vol. 3, no. 1, hal. 45–52, 2023.
- [7] A. F. Rochim, "Analisis Perbandingan Kinerja Protokol MQTT dan HTTP pada Sistem IoT Berbasis Raspberry Pi," *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*, vol. 7, no. 4, hal. 789–796, 2020.

- [8] Y. Rahmanto, "Analisis Quality of Service (QoS) Jaringan Internet Menggunakan Metode Hierarchical Token Bucket," *Jurnal Teknoinfo*, vol. 15, no. 2, hal. 88-95, 2021.
- [9] S. F. Anggraini, A. Ma'arif, dan R. D. Puriyanto, "Pengendali PID pada Motor DC dan Tuning Menggunakan Metode Differential Evolution (DE)," *TELKA - Telekomunikasi Elektronika Komputasi Dan Kontrol*, vol. 6, no. 2, hal. 147-159, 2020.
- [10] I. Irhas, I. Iftitah, dan S. A. A. Ilham, "Penggunaan Kontrol PID dengan Berbagai Metode untuk Analisis Pengaturan Kecepatan Motor DC," *JFT: Jurnal Fisika dan Terapannya*, vol. 7, no. 1, hal. 78-86, 2020.
- [11] R. H. Putra dan A. W. Wardhana, "Penerapan Artificial Neural Network untuk Pengaturan Kecepatan Motor DC Brushless," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi (JNTETI)*, vol. 9, no. 3, hal. 255-261, 2020.
- [12] H. Wicaksono, "Implementasi Kalman Filter pada Sensor MPU6050 untuk Kestabilan Robot Keseimbangan," *Jurnal Teknik Elektro*, vol. 12, no. 1, hal. 34-40, 2021.
- [13] T. Chen dan C. Guestrin, "XGBoost: A Scalable Tree Boosting System," dalam *Proceedings of the 22nd ACM SIGKDD International Conference*, 2016 (dikutip dalam konteks penerapan terbaru 2024 pada sistem kendali).
- [14] Yan Zhang et al., "Design of Self Balancing Robot Based on ESP32," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 12, no. 2, hal. 622-630, 2024.
- [15] S. Randa, "Low-Cost DC Motor Design for Embedded Systems in Smart Applications," *Journal of Informatics and Computer Science (JICS)*, vol. 6, no. 3, hal. 241-250, 2025.
- [16] A. R. Al-Ali, "Securing Bidirectional Wireless Control of DC Motors Against MQTT-Based Cyber Attacks," *IEEE Access*, vol. 13, hal. 1023-1035, 2025.
- [17] W. Aribowo, "Frilled Lizard Optimization to optimize parameters Proportional Integral Derivative of DC Motor," *Bulletin of Engineering, Technology and Applied Science*, vol. 1, no. 1, hal. 12-20, 2024.
- [18] P. Grelewicz, "Application of Machine Learning to Performance Assessment for a Class of PID-based Control Systems," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 55, no. 1, hal. 102-115, 2025.
- [19] D. Izci, "Opposition based Henry gas solubility optimization as a novel algorithm for PID control of DC motor," *Engineering Science and Technology, an International Journal*, vol. 24, no. 2, hal. 331-342, 2021.
- [20] Z. P. Muqorrobin, "Control of DC Motor in Laundry Liquid Waste Treatment based on Esp32-S3 And Thingsboard Platform," *Vokasi Unesa Bulletin of Engineering, Technology and Applied Science*, vol. 1, no. 2, hal. 65-72, 2024.