

Desain Kendali Menggunakan Hybrid PID-BA untuk Mengatur Kecepatan Motor Induksi

¹Titis Amujiati, ²Dwi Ajiatmo, ³Machrus Ali, ⁴Muhlasin

¹ Program Studi Teknik Elektro, Universitas Darul Ulum, Jombang

² Program Studi Teknik Elektro, Universitas Darul Ulum, Jombang

³ Program Studi Teknik Elektro, Universitas Darul Ulum, Jombang

⁴ Program Studi Teknik Elektro, Universitas Darul Ulum, Jombang

¹tiesa027@gmail.com, ²ajiatmo@gmail.com, ³machrus7@gmail.com, ⁴doktormuhlasin@gmail.com

Abstract - Induction motors have several non-linear parameters. This causes the settings for induction motors to be more complicated than DC motors. One of the disadvantages of an induction motor is that it is unable to maintain a constant speed when there is a change in load, so a controller is needed. Artificial Intelligent (AI) based intelligent control to improve conventional control. In this study, the design of a 3-phase induction motor control model uses the PID-Bat Algorithm control to control the speed of a 3-phase induction motor. Then test the control on a 3-phase induction motor model with changes in speed and changes in load torque. The results of the performance of the 3-phase induction motor control model using the PID Bat Algorithm with changes in speed and changes in the load torque load get the better steady-state error, settling time, and overshoot.

Keywords — Induction motor; PID; Bat Algorithm.

Abstrak— Motor induksi memiliki beberapa parameter non-linier. Hal ini menyebabkan pengaturan motor induksi lebih rumit dari pada motor DC. Salah satu kelemahan motor induksi adalah tidak mampu mempertahankan kecepatan yang konstan ketika terjadi perubahan beban, sehingga diperlukan pengontrol. Kontrol cerdas berbasis Artificial Intelligent (AI) untuk meningkatkan kontrol konvensional. Pada penelitian ini, rancang bangun model kendali motor induksi 3 fasa menggunakan kendali Algoritma PID-Bat untuk mengendalikan kecepatan motor induksi 3 fasa. Kemudian uji kendali pada model motor induksi 3 fasa dengan perubahan kecepatan dan perubahan torsi beban. Hasil performansi model kendali motor induksi 3 fasa menggunakan Algoritma PID Bat dengan perubahan kecepatan dan perubahan beban torsi beban mendapatkan error steady state, settling time, dan overshoot yang lebih baik.

Kata Kunci— Motor Induksi; PID; Bat Agoritma

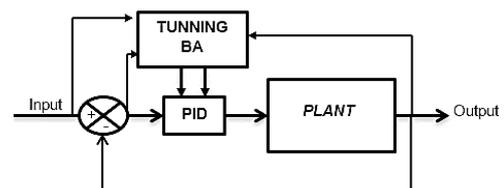
I. PENDAHULUAN (FONT 10)

Motor induksi 3 fasa adalah penggerak yang paling banyak digunakan dalam industri. Hal ini dikarenakan motor induksi memiliki konstruksi yang sederhana, kokoh, harga yang relatif murah, dan perawatan yang mudah sehingga motor induksi mulai menggeser penggunaan motor DC [1][2] di industri. Motor induksi memiliki beberapa parameter non-linier, terutama resistansi rotor, yang nilainya bervariasi untuk kondisi operasi yang berbeda. Hal ini menyebabkan pengaturan motor induksi menjadi lebih rumit dibandingkan

motor DC. Salah satu kelemahan motor induksi adalah tidak mampu mempertahankan kecepatan yang konstan ketika terjadi perubahan beban. Jika terjadi perubahan beban maka kecepatan motor induksi akan berkurang. Untuk mendapatkan kecepatan yang konstan dan meningkatkan unjuk kerja motor induksi terhadap perubahan beban, diperlukan suatu pengontrol.

Kontroler PID digunakan untuk aplikasi kontrol motor karena strukturnya yang sederhana dan algoritma kontrol yang mudah dipahami. Parameter pengontrol umumnya diatur menggunakan trial and error atau metode respons frekuensi Ziegler-Nichols. Kedua metode memiliki hasil yang sukses tetapi membutuhkan waktu yang cukup lama untuk mendapatkan respon sistem yang memuaskan. Dua masalah utama yang dihadapi dalam kontrol motor adalah sifat parameter motor yang berubah-ubah terhadap waktu dalam kondisi operasi dan adanya noise dalam sistem loop. Kontrol cerdas berbasis kecerdasan buatan telah banyak berkembang untuk meningkatkan kontrol konvensional. Oleh karena itu, pada penelitian ini akan dirancang model kendali motor induksi 3 fasa menggunakan kendali PID untuk mengatur kecepatan motor induksi 3 fasa. Kemudian akan menguji kedua kontrol tersebut pada model motor induksi 3 fasa dengan perubahan kecepatan dan perubahan torsi beban. Hasil performansi model kendali motor induksi 3 fasa menggunakan PID dengan perubahan kecepatan dan perubahan torsi beban ditemukan memiliki error steady state, settling time dan overshoot yang lebih baik.

Pada penelitian ini diusulkan untuk mengatur kecepatan motor induksi 3 fasa ini menggunakan kontroler hybrid PID-BA. Pemodelan kontrol PID-BA hybrid yang diusulkan untuk mengontrol kecepatan motor induksi 3 fasa ditunjukkan pada gambar di bawah ini.

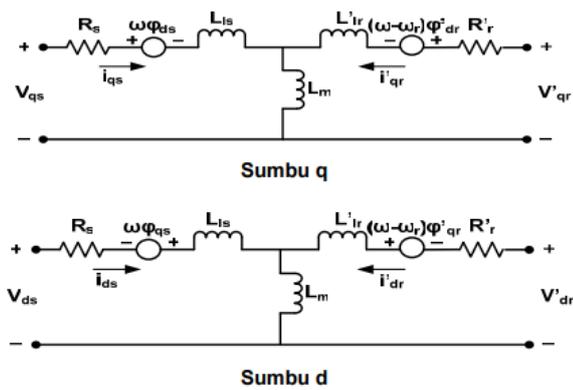


Gambar 1 Blok Diagram Pengaturan Kecepatan Motor Induksi Menggunakan Hybrid PID-BA

II. MATERIAL

A. Motor Induksi

Pada bagian ini, model tinjauan standar motor induksi digunakan. Model rangkaian ini merupakan model yang paling umum digunakan dalam berbagai analisis motor induksi. Model 'imajiner' yang menggambarkan motor induksi sebagai sistem yang terdiri dari vektor yang ditentukan oleh dua koordinat ortogonal dan kutub. Komponen vektor ruang dikaitkan dengan sepasang variabel, yaitu bagian langsung dan kuadratur (d-q). Model rangkaian ekuivalen digambarkan sebagai berikut [3]–[5]:



Gambar 2 Rangkaian Ekuivalen Motor Induksi Tiga Fasa Pada Sumbu d-q.

Tabel 1. Spesifikasi Motor Induksi

Simbol	Information	Rating	Unit
n	Tegangan sumber terukur	380	Volt.
P	Jumlah pole	6	-
f_s	Frekuensi stator	50	Hz
J_{eq}	Momen inersia stator	0.3	$Kg.m^2$
R_s	Tahanan stator	0.08	Ohm
R_r	Tahanan rotor	0.04	Ohm
L_s	Induktansi stator	15.4	mH
L_r	Induktansi rotor	16.38	mH
L_m	Induktansi magnetik	14.60	mH
S	Slip	0.02	-
B_m	Koefisien gesekan	-	$N_m/rad/se$ c

Nilai parameter tersebut kemudian akan disubstitusikan ke dalam bentuk fungsi transfer sebagai berikut:

$$\frac{\omega_m(s)}{V_{sd}(s)} = \frac{n_p L_m (\lambda_{rd} i_{sq}(s) - \lambda_{rq} i_{sd}(s))}{L_r (J_{eq} s + B_m)} \quad (1)$$

$$\frac{\omega_m(s)}{V_{sd}(s)} = \frac{1,4833}{2,16 \times 10^{-2} s^2 + 0,0471s + 9,504} \quad (2)$$

$$\frac{\omega_m(s)}{V_{sd}(s)} = \frac{868}{s^2 + 21,790s + 6970} \quad (3)$$

B. Kontrol Proporsional Integratif Derivatif (PID)

Kendali PID merupakan sistem kendali gabungan antara kendali proporsional, integral, dan turunan. Dalam metode ini, penyetelan dilakukan dalam loop tertutup dimana input referensi yang digunakan adalah fungsi langkah. Kontroler dalam metode ini hanya kontroler proporsional. Kp, dinaikkan dari 0 ke nilai kritis Kp, sehingga diperoleh keluaran yang berosilasi terus menerus dengan amplitudo yang sama. Nilai kritis Kp ini disebut sebagai ultimate gain. Periode akhir, Tu, diperoleh setelah keluaran sistem mencapai kondisi berosilasi terus menerus[6]–[12].

C. Bat Algoritma (BA)

Algoritma kelelawar diusulkan oleh Xin-She Yang, berdasarkan ekolokasi mikroba. Kelelawar biasanya menggunakan ekolokasi untuk mencari makanan. Selama transfer, kelelawar biasanya mengirimkan pulsa pendek, tetapi ketika mereka menemukan makanan, denyut nadi mereka meningkat dan frekuensinya meningkat. Meningkatkan frekuensi berarti penyesuaian frekuensi, yang mempersingkat waktu gema dan meningkatkan akurasi lokasi. Dalam algoritma kelelawar standar, setiap individu memiliki posisi tertentu $x_i(t)$ dan kecepatan $v_i(t)$ dalam ruang pencarian, yang akan diperbarui seiring dengan peningkatan jumlah iterasi. Melalui sejumlah besar studi eksperimental, ditemukan bahwa operator yang berbeda memainkan peran penting dalam kemampuan konvergensi algoritma. Saat operator pengembangan meningkat, kemampuan konvergensi global dari algoritme menjadi lebih lemah; sebagai operator eksplorasi meningkat, akurasi konvergensi tidak akan cukup. Oleh karena itu, dalam makalah ini, kami mengusulkan beberapa strategi pemilihan strategi mandiri. Ide utamanya adalah bahwa individu yang berbeda memilih strategi mana untuk memperbarui posisi sesuai dengan kualitas kebugaran. Dalam tulisan ini, berikut ini menjelaskan langkah-langkah eksekusi dari Bat algoritma standar.

1. Untuk setiap kelelawar, inialisasi posisi, kecepatan, dan parameter dan bangkitkan frekuensi acak dengan Persamaan $x_i(t+1) = x_i(t) + v_i(t+1)$.

2. Perbarui posisi dan kecepatan setiap kelelawar dengan Persamaan $v_i(t+1) = v_i(t) + (x_i(t)-p(t)) f_i$ dan $f_i = f_{min} + (f_{max}-f_{min})\beta$.
3. Untuk setiap kelelawar, buat angka acak ($0 < r < 1$). Perbarui posisi suhu dan hitung nilai kebugaran kelelawar menurut persamaan $x_i(t+1) = hal \rightarrow (t \& A(t))$ jika $r < r_i(t)$.
4. Untuk setiap kelelawar, buat angka acak ($0 < r < 1$). Memperbarui $A_i(t)$ dan $r_i(t)$ dengan Persamaan $A_i(t+1) = SAI(t)$ dan $r_i(t+1) = r_i(0) [1-\exp(-\gamma t)]$ berturut-turut, jika dijalankan $d2 < A_i(t)$ dan $f(x_i(t)) < f(p(t))$.
5. Urutkan setiap individu berdasarkan nilai fitness dan simpan posisi terbaik.
6. Algoritme selesai jika kondisi terpenuhi, jika tidak, lanjutkan ke Langkah 2.

Langkah-langkah rinci dari algoritma kelelawar standar disajikan dalam diagram alir gambar 3.

Strategi di atas dipilih dengan bentuk probabilitas. Oleh karena itu, jumlah individu kelelawar yang memilih strategi berbeda bervariasi dari generasi ke generasi. Setiap strategi menyesuaikan probabilitas yang dipilih berdasarkan hasil evaluasi. Ketika nilai fitness lebih baik, probabilitas strategi akan disesuaikan dengan persamaan di bawah ini.

$$P(N+1) = P(n) + (1 - \lambda) P(n) \quad (4)$$

Jika tidak, dihitung sebagai berikut:

$$P(n+1) = \lambda P(n) \quad (5)$$

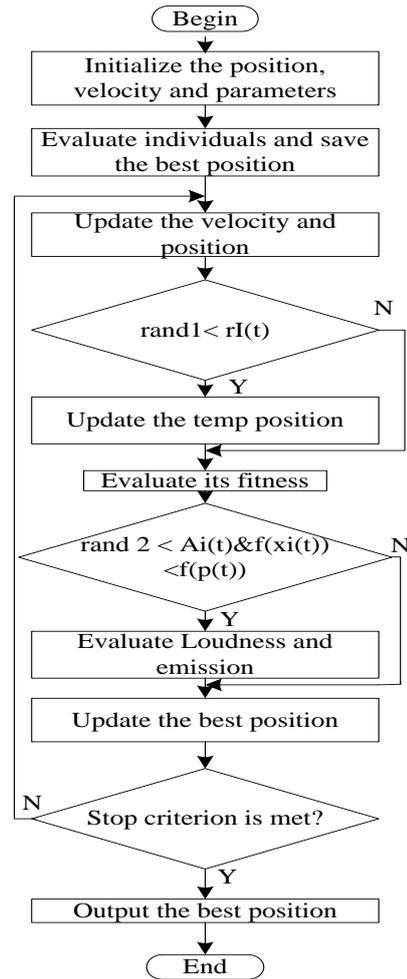
Dimana $\lambda = 0,75$, yang merupakan parameter yang digunakan untuk mengatur laju perubahan probabilitas. Semakin besar, semakin lambat tingkat penurunan probabilitas. Untuk memastikan keragaman populasi, kami mendefinisikan probabilitas batas bawah untuk setiap strategi sebagai 0,01.

Tabel 2 Parameter Bat Algoritma

Parameter	Nilai Rating
<i>Alpha</i>	0.9
<i>Gamma</i>	0.9
<i>Population size</i>	35
<i>Loudness</i>	0.5
<i>Pulse rate</i>	0.6
<i>Minimum frequency</i>	0
<i>Maximum frequency</i>	100
<i>First iteration</i>	1
<i>Number of iterations</i>	2

III. METODE PENELITIAN

Kerangka kerja penulisan paper ini dapat dilihat pada Gambar 3 di bawah ini.

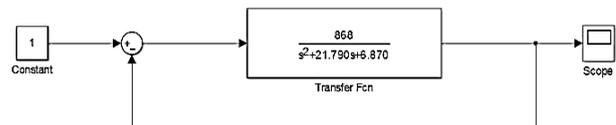


Gambar 3 Flowchart Langkah-Langkah Penelitian

IV. HASIL DAN PEMBAHASAN

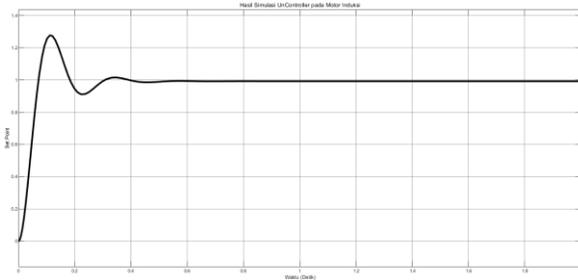
A. Simulasi Sistem Tanpa Kendali PID

Simulasi model motor induksi tanpa kontroler PID dapat dilihat pada Gambar 4 di bawah ini:



Gambar 4 Fungsi Alih Motor Induksi Tanpa Pengontrol

Hasil running program matlab 2019b dapat dilihat pada Gambar 4 dibawah ini:

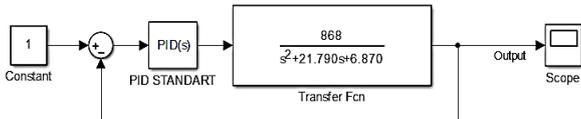


Gambar 5 Hasil Menjalankan Program Simulink Motor Induksi Tanpa Pengontrol.

Berdasarkan hasil grafik di atas dapat diartikan bahwa terjadi overshoot maksimum sebesar 1,277 pada $t = 0,113$ detik, dengan settling time motor induksi pada $t = 3,9$ detik di bawah set point.

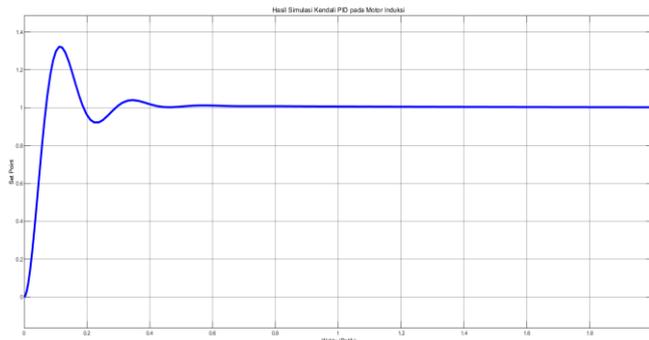
B. Simulasi Sistem Dengan Kontroler PID (PID Standar)

Simulasi model motor induksi dengan kontroler PID dapat dilihat pada Gambar 6 di bawah ini:



Gambar 6 Model Motor Induksi Dengan Pengontrol PID (Standar)

Hasil running program Matlab 2019b dapat dilihat pada Gambar 7 dibawah ini :

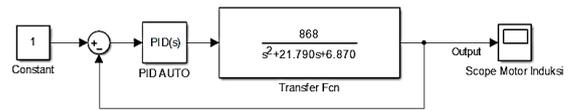


Gambar 7 Hasil Running Program Simulink Motor Induksi Dengan Kontroler PID.

Berdasarkan hasil grafik diatas dapat diartikan bahwa terjadi overshoot maksimum sebesar 1,322 pada $t = 0,112$ detik, dengan settling time motor induksi sebesar 1,7 detik.

C. Simulasi Sistem Dengan PID Auto Tuning MATLAB Controller

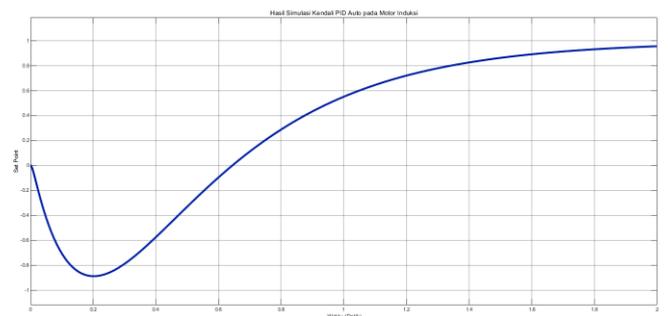
Simulasi model motor induksi dengan PID Auto Tuning Controller dapat dilihat pada Gambar 4.5 di bawah ini:



Gambar. 8 Model Motor Induksi Dengan Pengontrol PID (Penyetelan Otomatis)

Hasil pengaturan kecepatan Motor Induksi auto tuning dengan matlab auto tuning PID diperoleh nilai konstanta $K_p = 0,0127810419353847$, $K_i = 0,0054210671328696$ dan $K_d = -0,0146245116578319$ dengan koefisien filter = 0,56970479527579

Hasil running program matlab 2019b dapat dilihat pada gambar dibawah ini :

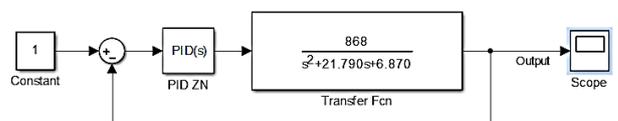


Gambar 9 Hasil Menjalankan Program Auto-Tuning Matlab Motor Induksi Simulink

Berdasarkan hasil grafik di atas dapat diartikan bahwa terjadi overshoot maksimal sebesar 1,096 pada $t = 5$ detik, dengan settling time motor induksi sebesar 25 detik.

D. Simulasi Sistem Dengan Pengontrol PID Ziegler-Nichols

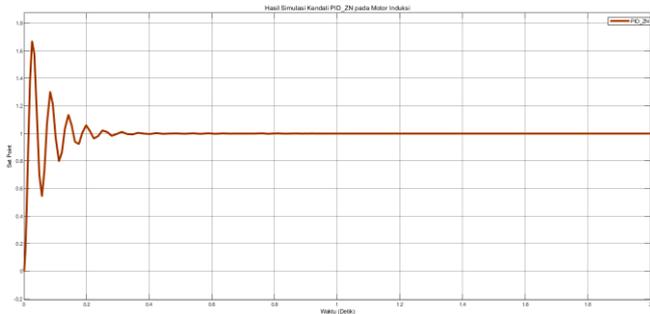
Simulasi model motor induksi Ziegler-Nichols Controller PID dapat dilihat pada gambar di bawah ini:



Gambar 10 Model Motor Induksi Dengan Pengontrol PID Ziegler-Nichols

Hasil perhitungan pengaturan kecepatan K_u dan T_u Motor Induksi dengan matlab auto tuning PID diperoleh nilai konstanta $K_p = 13,32$, $K_i = 1,094$ dan $K_d = 0,015$

Hasil running program matlab 2019b dapat dilihat pada gambar dibawah ini :

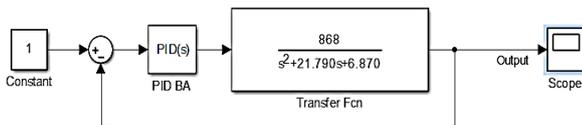


Gambar. 11 Hasil Dari Menjalankan Ziegler-Nichols. Program Simulink Motor Induksi

Berdasarkan hasil grafik diatas dapat diartikan bahwa terjadi overshoot maksimal sebesar 1,68 pada $t = 0,003$ detik, dengan settling time motor induksi sebesar 4,85 detik.

E. Simulasi Sistem Dengan PID-Bat Algoritma

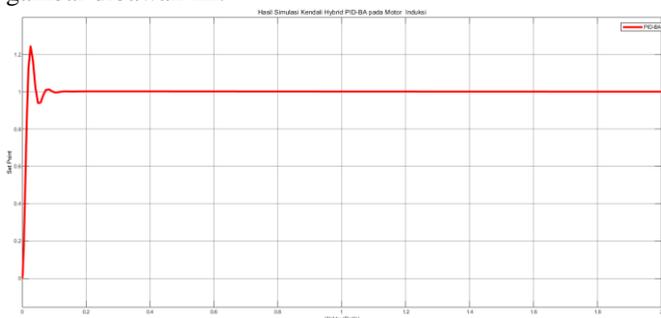
Simulasi model motor induksi dengan PID yang disetel oleh pengontrol Bat Algoritma dapat dilihat pada gambar di bawah ini:



Gambar 12 Model Motor Induksi Dengan Kontrol PID-BA

Hasil auto tuning pengaturan kecepatan Motor Induksi dengan PID tune BA diperoleh nilai konstanta $K_p = 3.6329$, $K_i = 2.6647$ dan $K_d = 0.537$

Hasil running program matlab 2019b dapat dilihat pada gambar dibawah ini.

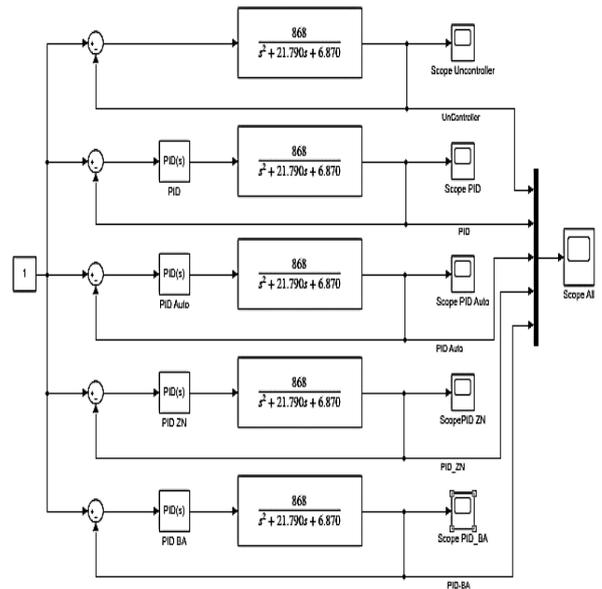


Gambar 13 Hasil Running Program Simulink Motor Induksi Dengan Running Dengan Bat Algoritma.

Berdasarkan hasil grafik diatas dapat diartikan bahwa terjadi overshoot maksimal sebesar 1,24 pada $t = 0,013$ detik, dengan settling time motor induksi sebesar 0,12 detik.

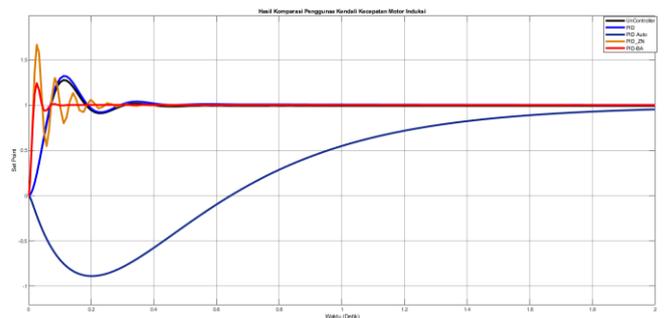
F. Perbandingan Hasil Simulasi Berbagai Pengendali

Simulasi kecepatan motor induksi dengan berbagai pengontrol dapat dilihat pada gambar di bawah ini:



Gbr. 14 Program Simulink Motor Induksi Untuk Berbagai Jenis Pengontrol.

Perbandingan hasil simulasi dari berbagai pengontrol dapat dilihat pada gambar di bawah ini:



Gbr.15 Perbandingan Hasil Menjalankan Motor Induksi Dengan Berbagai Pengontrol

Berdasarkan hasil simulasi di atas, dapat artikan bahwa:

1. Grafik hasil pada PID tanpa pengontrol dapat diartikan bahwa terjadi overshoot maksimum sebesar 1,277 pada $t = 0,113$ detik, dengan settling time motor induksi pada $t = 3,9$ detik
2. Grafik hasil pada kontroler PID standar dapat diartikan bahwa terjadi overshoot maksimal sebesar 1,322 pada $t = 0,12$ detik, dengan settling time motor induksi pada $t = 1,7$ detik.
3. Grafik hasil tuning otomatis pada PID dapat diartikan bahwa terjadi max overshoot sebesar 1,096 pada $t = 5$

- detik, dengan settling time motor induksi pada $t = 25$ detik
4. Grafik hasil pada PID Ziegler-Nichols dapat diartikan bahwa terjadi overshoot maksimal sebesar 1,68 pada $t = 0,003$ detik, dengan settling time motor induksi pada $t = 4,85$ detik.
 5. Grafik hasil pada Algoritma Kelelawar PID dapat diartikan bahwa terjadi max overshoot sebesar 1,24 pada $t = 0,013$ detik, dengan settling time motor induksi pada $t = 0,12$ detik

Untuk mengetahui lebih jelas hasil simulasi diatas dapat dilihat pada tabel 3 dibawah ini.

Tabel 3. Hasil Program Kontrol Motor Induksi Yang Dijalankan Pada Berbagai Metode Kontrol

	No Control	PID standart	PID Auto Tunning	PID ZN	PID BA
Kp	-	1	0,01278	13,32	3,633
Ki	-	1	0,00542	1,094	2,665
Kd	-	0	-0,01462	0,015	0,537
Overshoots (rpm)	1,277	1,322	1,096	1,68	1,24
Settling time (detik)	3,9	1,7	25	4,85	0,12

V. KESIMPULAN

Kecepatan konstan dan kinerja motor induksi yang merespon perubahan beban selama operasi, memerlukan sistem kontrol. Sistem kontrol cerdas berbasis Artificial Intelligent (AI) merupakan pilihan untuk meningkatkan kinerja sistem kontrol konvensional. Motor induksi 3 fasa yang dikendalikan menggunakan PID dengan Tuning Algorithm Kelelawar menghasilkan perubahan kecepatan dan beban torsi dengan kesalahan kondisi tunak, settling time, dan overshoot yang lebih baik dibandingkan dengan metode tidak terkontrol, Standard PID, PID auto tuning, dan Ziegler -Nichols PID digunakan sebagai pembanding dalam penelitian ini. Dengan menjalankan program yang telah dilakukan, didapatkan bahwa pengendali Algoritme Kelelawar tuning PID mendapatkan nilai overshoot maksimum sebesar 1,24 pada $t = 0,013$ detik, dengan settling time yang dicapai oleh motor induksi pada $t = 0,12$ detik. Sehingga kontroler PID dengan tuning Bat Algoritma dapat disebut sebagai kontroler terbaik dengan setting time tercepat dan overshoot yang cukup kecil pada penelitian ini.

VI. DAFTAR PUSTAKA

- [1] M. N. Masrukhan, M. P. Mulyo, D. Ajiatmo, and M. Ali, "Optimasi Kecepatan Motor DC Menggunakan PID dengan Tuning Ant Colony Optimization (ACO) Controller," *Pros. SENTIA-2016*, vol. 8, no. 1, pp. B49–B52, 2016.
- [2] M. Ruswandi Djalal, D. Ajiatmo, A. Imran, and I. Robandi, "Desain Optimal Kontroler Pid Motor Dc Menggunakan Cuckoo Search Algorithm," *Pros. SENTIA-2015*, Vol. 7 No. 1, 2015.
- [3] G. J. Abu-Rub Haitham, Iqbal Atif, *High Performance Control Of AC Drives With Matlab/Simulink Models*. United Kingdom: John Wiley & Sons, Inc., 2012.
- [4] P. C. Krause, O. Wasynczuk, and S. D. Sudhoff, "Analysis of electrical machinery and dirve systems." pp. 1–632, 1986, [Online]. Available: <https://www.wiley.com/en-us/Analysis+of+Electric+Machinery+and+Drive+System+s%2C+3rd+Edition-p-9781118524329>.
- [5] G. M. Ida Bagus, "Analisa Simulasi Multimesin Selama Gangguan Pada Suplai Listrik," Surabaya, 1999. [Online]. Available: <https://repository.its.ac.id/48978/1/2296201007-Undergraduate-Thesis.pdf>.
- [6] M. R. Djalal, D. Ajiatmo, S. Soedibyo, and I. Robandi, "Frequency Control Pltmh dengan Capacitive Energy Storage Menggunakan Cuckoo Search Algorithm," *Sentia 2015*, vol. 7, pp. 73–79, 2015, [Online]. Available: <http://sentia.polinema.ac.id/index.php/SENTIA2015/article/view/176/0>.
- [7] D. Ajiatmo and A. Raikhani, "Desain Optimasi LFC Pada Micro-Hydro Menggunakan Metode Ant Colony Optimization (ACO)," in *Seminar Nasional Inovasi Teknologi UN PGRI Kediri*, 2017, pp. 79–84.
- [8] M. A. Y. Alghifrani, H. Nurohmah, D. Ajiatmo, and M. Ali, "Bat Algorithm Sebagai Optimasi PID Controller Pada Turbin Angin," *Sinarfe7-2*, vol. 2, no. 1, pp. 447–451, 2019, [Online]. Available: <https://ejournal.fortei7.org/index.php/SinarFe7/article/view/89>.
- [9] I. Robandi, D. Ajiatmo, and Muhlasin, "Optimization PI-ACO for Photovoltaic System Battery and Supercapacitor on Electric Vehicle," in *2021 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 2021, pp. 370–375, doi: 10.1109/isitia52817.2021.9502221.
- [10] D. Ajiatmo, Y. M. Safarudin, and D. Mariani, "Desain Optimasi Load Frequency Control Pada PLTMH Dengan Modified Firefly Algorithm (MFA)," in *Prosiding Seminar Nasional NCIET Vol.1 (2020)*, 2020, vol. 1, p. B.406-B.414, [Online]. Available: <http://conf.nciet.id/index.php/nciet/article/view/147/192>.
- [11] H. Dripoyono, S. D. Candra, D. Ajiatmo, and M. Ali, "Penggunaan ACO dan FA Dalam Mengoptimasikan PID Controller Untuk Partial Shading pada Photovoltaic," pp. 35–39.
- [12] D. Ajiatmo and I. Robandi., "Performances Design of Photovoltaic and Battery Parallelization Using PI-MPPT Prototipe Autotuned by Firefly Optimization," in *Proceeding International Symposium For Modern School Development, Social Science And Applied*

Technologies (ISMOSAT) 2016, 2016, pp. 7–12, [Online]. Available: <http://digilib.uinsby.ac.id/6881/13/ProsidingISMOSAT.pdf#page=13>.

[13] X. S. Yang, “Bat algorithm: Literature review and applications,” *Int. J. Bio-Inspired Comput.*, vol. 5, no. 3, pp. 141–149, 2013, doi: 10.1504/IJBIC.2013.055093.