

Modifikasi Desain PID Controller Pada Permanent Magnet Synchronous Motor Dengan Flower Pollination Algorithm

Muhammad Ruswandi Djalal¹, Machrus Ali², Andi Imran³, Herlambang Setiadi⁴

¹Politeknik Negeri Ujung Pandang, Makassar

²Universitas Darul 'Ulum, Jombang

³Institut Teknologi Sepuluh Nopember, Surabaya

⁴The University of Queensland, Australia

E-mail : wandi@poliupg.ac.id

ABSTRACT

The use of PID (Proportional-Integral-Derivative) controllers on a synchronous motor is very widely used, because of its simple, robust structure that is robust and easy to use. The use of PID controllers requires proper parameter setting for optimal performance on the motor. The most commonly used solution is the trial-error method, to determine the proper parameters for PID, but the results do not make the PID controller optimal. Lately there has been a lot of research to optimize PID controller, wrong with smart method. For this purpose, we will use Flower Pollination Algorithm (FPA) optimization method to optimize and determine the exact parameters of PID. FPA is one method that is being adapted from the process of plant pollination, so this concept is adapted and applied to be. From the results that have the FPA method can be well tuned parameters PID, so that the resulting overshoot faster and settling time is very fast. Optimization results K_p 0.9441, K_i 0.9311, K_d 0.0840. In this study will discuss uncontrolled motors, with PID trial-error controller, PID-PSO and PID-FPA.

Keywords: Sync motor, PID, Trial-Error, PSO, FPA, Overshoot, Settling time

ABSTRAK

Penggunaan kontroler PID (Proporsional-Integral-Derivatif) pada sebuah motor sinkron sangat banyak digunakan, karena strukturnya yang sederhana, kokoh yang kuat dan mudah digunakan. Penggunaan kontroler PID diperlukan pengaturan parameter yang tepat untuk kinerja yang optimal pada motor. Solusi yang sering digunakan adalah metode trial-error, untuk menentukan parameter yang tepat untuk PID, namun hasil yang didapat tidak membuat kontroler PID optimal. Belakangan ini sudah banyak penelitian untuk mengoptimasi kontroler PID, salah dengan metode cerdas. Untuk itu pada penelitian ini akan digunakan metode optimasi Flower Pollination Algorithm (FPA), untuk mengoptimasi dan menentukan parameter yang tepat dari PID. FPA adalah salah satu metode yang sedang teradaptasi dari proses penyerbukan tanaman, sehingga konsep inilah yang diadaptasi dan diterapkan menjadi. Dari hasil yang memiliki metode FPA dapat dengan baik menala parameter PID, sehingga overshoot yang dihasilkan semakin cepat dan waktu settling sangat cepat. Hasil optimasi K_p 0.9441, K_i 0.9311, K_d 0.0840. Pada penelitian ini akan membahas motor tanpa kontrol, dengan kontroler PID trial-error, PID-PSO dan dengan PID-FPA.

Kata Kunci: Motor Sinkron, PID, Trial-Error, PSO, FPA, Overshoot, Settling time

1. PENDAHULUAN

Penggunaan motor listrik AC sudah banyak digunakan dari pada motor DC karena arus AC dapat dibangkitkan dan didistribusikan dengan biaya yang lebih murah dari pada arus DC. Selain itu, motor listrik AC memiliki keunggulan dalam hal biaya, ukuran, berat, dan membutuhkan lebih sedikit perawatan dibanding motor DC. Salah satu jenis motor listrik AC adalah motor sinkron. Penggunaan motor sinkron sudah banyak di industry dan sebagai alat pengontrolnya digunakan controller Proportional Integral Derivative (PID). Penggunaan peralatan PID juga memiliki permasalahan tersendiri, yaitu nilai gain yang masih di tuning secara manual.

Kontrol (PID) memiliki kinerja yang baik sehingga banyak digunakan dalam dunia industri. Namun memiliki kelemahan yaitu membutuhkan perhitungan matematik yang rumit untuk mengisi masing-masing nilai gain dari PID.

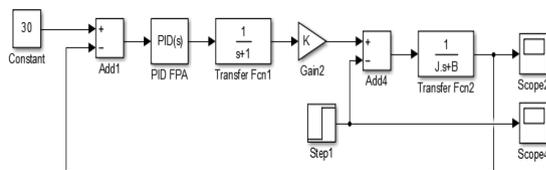
Beberapa metode optimasi berbasis metode konvensional maupun metode cerdas telah banyak digunakan untuk mengoptimasi parameter PID pada motor listrik, diantaranya Artificial Bee Colony [3], Evolutionary Algorithm [4,5,9], Particle Swarm Optimization [6], Bio-Inspired Algorithm [7], Bacterial Foraging [8], Tabu Search [10], Fuzzy Logic [11,12], Cuckoo Search [13] dan Flower [14].

Untuk mengatasi permasalahan di atas, pada penelitian ini diusulkan sebuah metode Algoritma Penyerbukan Bunga (*Flower Pollination Algorithm*) untuk mengontrol kecepatan motor sinkron. Sebagai pembanding digunakan metode Particle Swarm Optimization. Algoritma ini diadopsi dari kejadian alam penyerbukan bunga [1,2].

2. PEMODELAN SISTEM

2.1. Pemodelan Motor Sinkron

Pemodelan motor sinkron yang digunakan ditampilkan sebagai berikut.



Gambar 1. Pemodelan Motor sinkron di Simulink

2.2. Pemodelan Kontroler PID

Kontrol PID adalah salah satu kontrol yang sudah banyak digunakan pada aplikasi industri karena strukturnya yang sederhana, PID kontrol

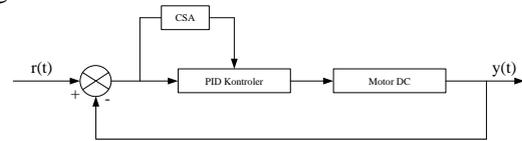
$$u(t) = k_p [e(t) + \frac{1}{T_i} \int_0^t e(t)dt + T_d \frac{de_t}{dt}] \quad (3)$$

Di mana, $u(t)$ adalah nilai kontrol yang dihitung oleh kontroler PID, K_p adalah koefisien proporsional, T_i adalah integral waktu konstan dan T_d adalah diferensiasi time konstan. Fungsi dari ketiga elemen tersebut adalah :

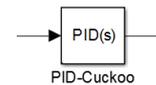
- Proporsional: menggambarkan sinyal deviasi $e(t)$ dari sistem kontrol proporsional. ketika sinyal $e(t)$ ada, kontroler PID menghasilkan efek kontrol segera untuk mengurangi penyimpangan.
- Integral : digunakan untuk menghilangkan kesalahan statis dan meningkatkan stabilitas sistem.
- Diferensial : mencerminkan perubahan penyimpangan sinyal, memperkenalkan sinyal koreksi sebelum penyimpangan nilai sinyal menjadi lebih besar dan mempercepat respon sistem untuk mengurangi pengaturan waktu.

Oleh karena itu, merancang kontroler PID terutama berarti menentukan tiga parameter, serta bagaimana mengkonfigurasi ke tiga parameter PID (K_p, K_i, K_d). Dalam penelitian

ini, *Flower Pollination Algorithm* diusulkan untuk mencari parameter optimal PID. Blok diagram sistem kontrol ditunjukkan pada gambar 1.



Gambar 2. Sistem Kontroler PID-FPA



Gambar 3. Pemodelan PID di Simulink

3. Flower Pollination Algorithm

Flower Pollination Algorithm (FPA) dapat diterapkan di masalah optimisasi memaksimalkan maupun meminimumkan. Dilakukan idealisasi dari karakteristik fenomena penyerbukan bunga di subbab sebelumnya, fenomena *flower constancy*, dan kelakuan organisme penyerbuk dengan aturan-aturan sebagai berikut:

- a. Penyerbukan biotik dan penyerbukan silang dipandang sebagai penyerbukan global dengan organisme penyerbuk pembawa serbuk sari yang melakukan gerakan *Lévy Flights*
- b. Penyerbukan abiotik dan penyerbukan sendiri dipandang sebagai penyerbukan lokal
- c. *Flower constancy* dipandang sebagai peluang reproduksi yang proporsional dengan similaritas dari dua bunga yang terlibat
- d. Penyerbukan lokal dan global diatur oleh peluang berpindah .

Terdapat dua langkah kunci di algoritma ini, yaitu penyerbukan global dan penyerbukan lokal. Pada langkah penyerbukan global, serbuk sari dari bunga dibawa oleh hewan penyerbuk seperti serangga, dan serbuk sari dapat melakukan perjalanan jarak jauh karena serangga dapat terbang dan bergerak di daerah yang luas. Proses ini bisa menghasilkan yang paling cocok, yang kita representasikan dengan Aturan pertama, ditambah dengan fenomena *flower constancy* dapat direpresentasikan secara matematis sebagai:

$$x_i^{t+1} = x_i^t + \gamma L(\lambda) (x_i^t - g^*) \quad (4)$$

dengan x_i^t menyatakan serbuk sari i atau vektor solusi x_i pada iterasi t , dan g^* adalah solusi terbaik di iterasi yang sedang berlangsung.

Parameter L adalah kekuatan penyerbukan, yang pada dasarnya merupakan besar langkah.

Tabel 1. Analogi Penyerbukan Bunga dengan Optimisasi

Masalah Optimisasi: $\max \text{ atau } \min f(x_i), x_i = (x_1, x_2, \dots, x_n) \in S^d \subset R^d, i = 1, 2, \dots, n$	
Fenomena Penyerbukan	Permasalahan Optimisasi
Daerah Cakupan Penyerbukan	Interval nilai $S^d, x \in S^d$
Populasi Bunga	Kandidat Solusi (x_1, x_2, \dots, x_n)
Tingkat kecocokan Bunga	Fungsi objektif f
Kejadian usaha penyerbukan	Iterasi
Gerak Penyerbukan Bunga	Pencarian Solusi pada Daerah Definisi
Dipilihnya Individu yang cocok	Keadaan Optimum

Tabel 2. Pseudo Code Firefly

```

Objektif min atau max  $f(x), x = (x_1, x_2, \dots, x_d)$ 
Inisiasi populasi n bunga (gamet serbuk sari) sebagai solusi acak
Temukan solusi terbaik  $g_*$  di populasi awal
Definisikan switch probability  $p \in [0,1]$ 
Definisikan kriteria berhenti (akurasi atau iterasi maksimum)
While (t < Iterasi Maksimum)
  For i=1:n (seluruh n bunga dari populasi)
    If rand < p,
      Ambil vektor langkah L berdimensi d yang memenuhi distribusi Levy
      Penyerbukan global  $x_i^{t+1} = x_i^t + \gamma L(\lambda)(x_i^t - g^*)$ 
    Else
      Ambil  $\epsilon$  dari distribusi uniform (0,1)
      Lakukan penyerbukan lokal  $x_i^{t+1} = x_i^t + \epsilon(x_i^t - x_k^t)$ 
    End if
  End for
  Evaluasi solusi baru
  Jika solusi yang baru lebih baik, perbarui solusi ini ke dalam populasi
End for
Temukan solusi terbaik  $g_*$ 
End while
Tampilkan solusi terbaik yang ditemukan
    
```

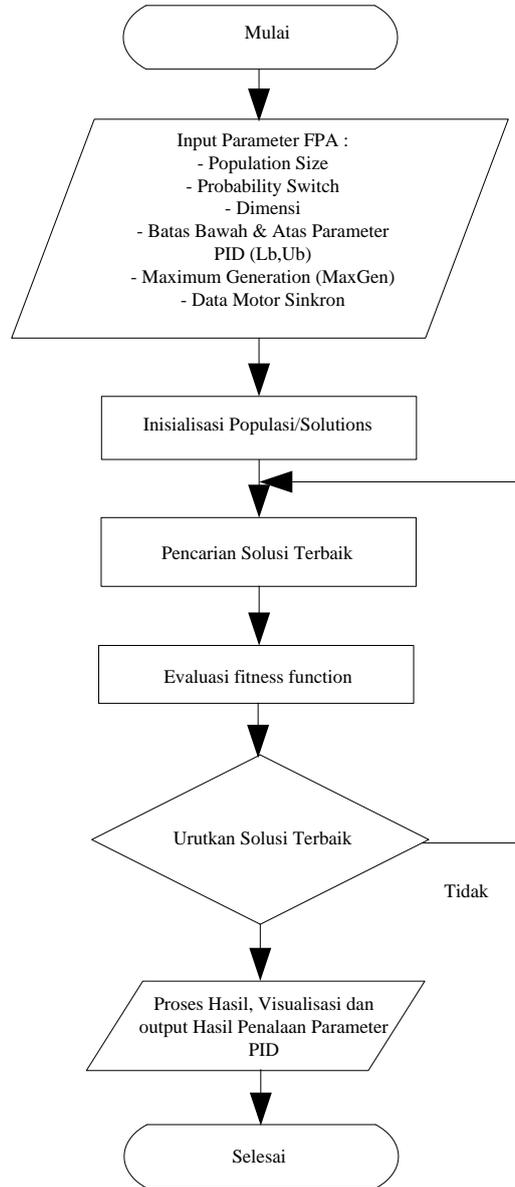
3.1. Penalaan PID dengan FPA

Gambar 5 menunjukkan diagram alir algoritma metode FPA yang digunakan pada penelitian ini untuk menala parameter PID. Fungsi objektif yang digunakan untuk menguji kestabilan sistem adalah dengan *Integral Time Absolute Error* (ITAE).

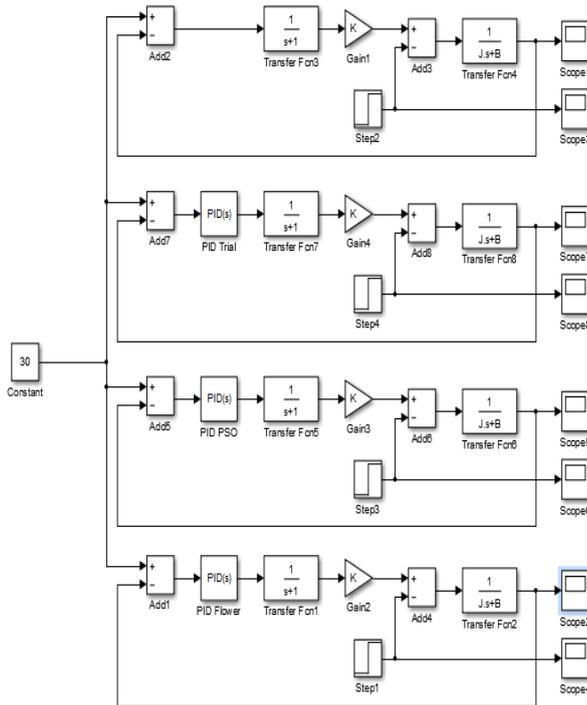
$$ITAE = \int_0^t |\Delta \omega(t)| dt \quad (5)$$

Parameter PID yang ditala oleh FPA adalah K_p, K_i dan K_d . Adapun untuk diagram alir

proses penalaan parameter PID dengan menggunakan metode FPA ditunjukkan oleh *flowchart* pada Gambar 5 dan gambar 4 menunjukkan pemodelan motor induksi pada Simulink Matlab 2015, tanpa kontrol, dengan PID Particle Swarm Optimization, dan PID FPA.



Gambar 4. Flowchart Penelitian



Gambar 5. Pemodelan Motor Sinkron di Simulink

Adapun data parameter-parameter berikut,

Tabel 3. Parameter FPA

Parameter	Nilai
Population Size	20
Probability Switch	0.8
Iteration Parameters	50
Dimension	3
Batas Atas Kp, Ki, Kd	[1 1 0.1]
Batas Bawah Kp, Ki, Kd	[0.1 0.1 0.01]

Berikut hasil optimasi dengan metode CSA.

```

Command Window
New to MATLAB? Watch this Video, see Examples, or read Getting Started.
Automatic solver parameter selection
diagnostic to 'none' in the Diagnostics page
of the configuration parameters dialog
> In fpa_demo>Fun at 140
In fpa_demo at 77
Total number of evaluations: 1000
Best solution=0.94406 0.93114 0.084043 fmin=81954.322

kp_flo =
    0.9441

ki_flo =
    0.9311

kd_flo =
    0.0840

ans =
    0.9441    0.9311    0.0840
    
```

Gambar 6. Hasil Optimasi Kontrol PID Motor Sinkron dengan Flower Pollination Algorithm di MATLAB

Tabel 4. Hasil Optimasi PID dengan FPA

Total number of evaluations: 1000
Best solution=0.94406 0.93114 0.084043
fmin=81954.322
kp_flo =
0.9441
ki_flo =
0.9311
kd_flo =
0.0840
ans =
0.9441 0.9311 0.0840

Hasil optimasi FPA didapatkan nilai *fitness function* sebesar 81954.322, dengan 1000 iterasi, nilai *best solution* merupakan hasil optimasi, yang di mana diketahui sebagai hasil optimasi parameter PID, yaitu Kp, Ki dan Kd. Tabel 4 menunjukkan nilai hasil optimasi parameter PID ditala oleh FPA.

Tabel 4. Parameter PSO

Parameter	Nilai
Jumlah Partikel	30
Maksimal Iterasi	50
Jumlah Variabel	3
Social Constant	2
Cognitive Constant	2
Momentum Inersia	0.9
Batas Atas Kp, Ki, Kd	[40 30 0.1]
Batas Bawah Kp, Ki, Kd	[30 20 0]

```

Command Window
New to MATLAB? Watch this Video, see Examples, or read Getting Started.
Automatic solver parameter selection
diagnostic to 'none' in the Diagnostics page
of the configuration parameters dialog
> In pso_at 81

fitness_terbaik_global =
    8.1261e+04

kp_pso =
    0.7828

ki_pso =
    0.7312

kd_pso =
    0.1831
    
```

Gambar 7. Hasil Optimasi Kontrol PID Motor Sinkron dengan PSO di MATLAB

Dan sebagai perbandingan digunakan metode pembandingan Particle Swarm Optimization

(PSO). Tabel 4 menunjukkan parameter PSO yang digunakan serta gambar 7 hasil optimasi yang diperoleh.

Tabel 4. Hasil Optimasi PID dengan PSO

fitness_terbaik_global =	8.1261e+04
kp_pso =	0.7828
ki_pso =	0.7312
kd_pso =	0.1831

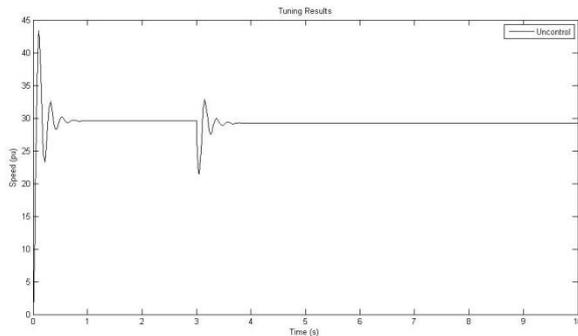
Tabel 5. Hasil Penalaan Parameter PID

Parameter	Trial	PSO	FPA
K_p	0.0925	0.7828	0.9441
K_i	0.0912	0.7312	0.9311
K_d	0.0217	0.1831	0.0840

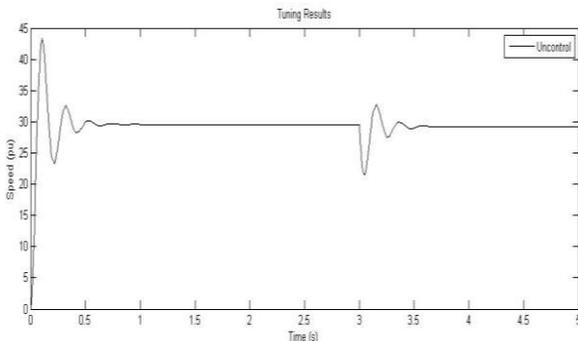
4. HASIL SIMULASI DAN ANALISIS

4.1. Respon Kecepatan Motor Sinkron tanpa Controller

Simulasi pertama adalah simulasi open loop Motor Sinkron tanpa *controller*. Berikut hasil simulasi.



Gambar 8. Respon Kecepatan Motor Sinkron tanpa kontrol, t=10s.



Gambar 9. Respon Kecepatan Motor Sinkron tanpa kontrol, t=5s.

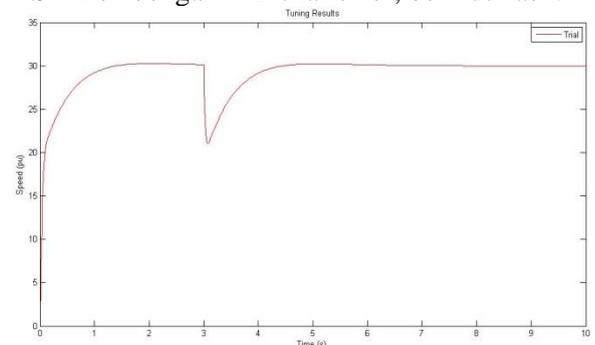
Dari hasil simulasi tanpa kontroler, didapatkan respon kecepatan motor sinkron yang sangat tinggi, hal ini dikarenakan sistem tidak ada

umpan balik, sehingga motor bekerja tanpa ada batasan dan untuk sistem yang seperti ini sangat dihindari. Kemudian ketika ada perubahan beban pada $t = 3s$, respon motor tidak begitu baik merespon perubahan beban, oleh karenanya terdapat osilasi yang sangat besar.

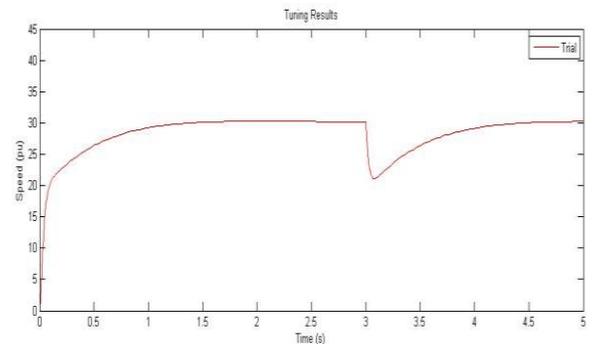
Untuk itu sangat diperlukan desain sistem kontrol yang tepat dengan penambahan kotroler PID pada motor sinkron, sehingga kecepatan yang dihasilkan dapat dikontrol sesuai dengan beban yang dikopel oleh motor sinkron.

4.2. Respon Kecepatan Motor Sinkron dengan PID Trial-Error

Simulasi yang kedua adalah kontrol motor Sinkron dengan PID trial-error, berikut hasil.



Gambar 10. Respon Kecepatan Motor Sinkron dengan PID Trial-Error, t=10s

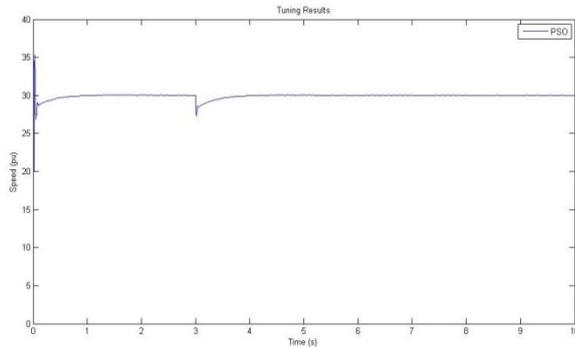


Gambar 11. Respon Kecepatan Motor Sinkron dengan PID Trial-Error, t=5s

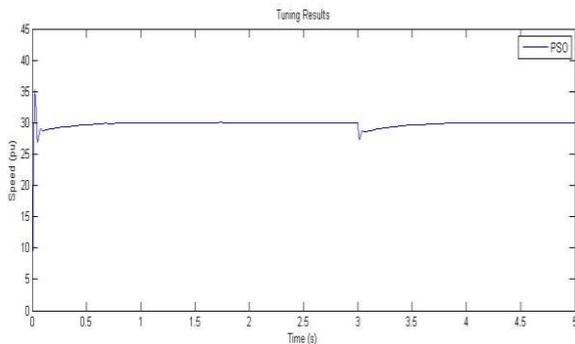
Gambar grafik di atas menunjukkan grafik respon frekuensi motor Sinkron dengan PID Trial-Error. Dari grafik di atas didapatkan respon yang sudah lebih baik dibandingkan dengan system tanpa control, di mana terlihat overshoot yang sudah tidak ada namun *settling time* yang sangat lama. Hal ini berarti, kontrol yang seperti ini sangat tidak diinginkan untuk sistem pengontrolan, hal tersebut dikarenakan respon sistem yang lama akan mengganggu kinerja dari plan yang dikontrol. Kinerja PID pada system ini masih bisa dioptimalkan.

4.3. Respon Kecepatan Motor Sinkron dengan PID PSO

Simulasi yang ketiga adalah kontrol motor sinkron dengan PID PSO, berikut hasil simulasinya.



Gambar 12. Respon Kecepatan Motor Sinkron dengan PID PSO, t=10s

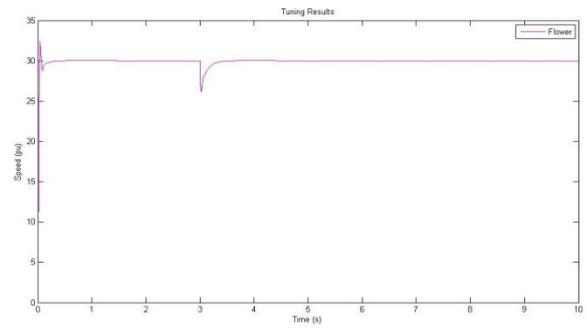


Gambar 13. Respon Kecepatan Motor Sinkron dengan PID PSO, t=5s

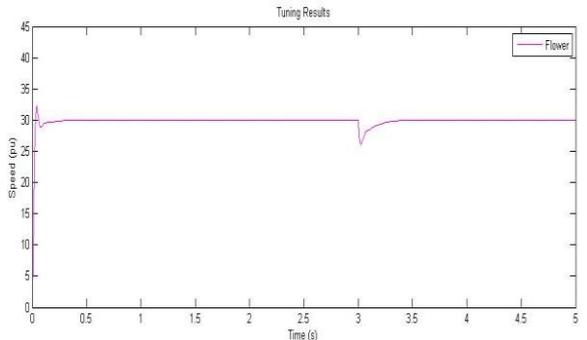
Gambar 12 dan 13 grafik di atas menunjukkan grafik respon frekuensi motor sinkron dengan PID PSO. Dari grafik di atas didapatkan masih terdapat overshoot ketika motor sinkron baru dijalankan namun respon settling time yang sedikit lebih baik dibandingkan dengan control PID trial., di mana sistem sudah semakin cepat berada pada kondisi steady. Respon motor juga lebih baik ketika terjadi perubahan beban pada t = 5s, dan tidak terdapat overshoot ketika terjadi penambahan beban pada poros motor sinkron.

4.4. Respon Kecepatan Motor Sinkron dengan PID FPA

Simulasi terakhir dengan menggunakan metode yang diusulkan dengan menggunakan FPA (Flower Pollination Algorithm). Berikut hasil simulasi.



Gambar 14. Respon Kecepatan Motor Sinkron dengan PID FPA, t=10s

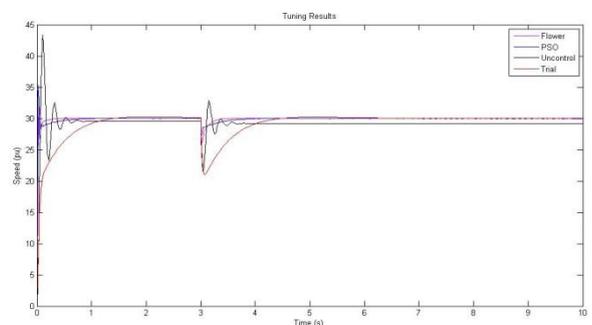


Gambar 15. Respon Kecepatan Motor Sinkron dengan PID FPA, t=5s

Dari grafik di atas, dapat dilihat besar overshoot semakin membaik dibanding dengan metode PSO dan trial sebelumnya.

Ini dapat disimpulkan bahwa dengan menggunakan metode cerdas untuk mengoptimasi parameter PID, hasil yang didapat untuk kinerja PID sangat baik karena menunjukkan respon yang cepat untuk kontrol motor sinkron.

Gambar berikut menunjukkan perbandingan dari kontrol motor sinkron yang telah disimulasikan dan overshoot yang dihasilkan masing-masing metode.



Gambar 16. Perbandingan Respon Kecepatan Motor Sinkron, t=10s

Dibutuhkan desain kontroler motor sinkron yang tepat, dalam hal ini sangat diusulkan untuk

menggunakan kontroler PID karena sangat simple untuk pengontrolan sistem dengan penalaan parameternya menggunakan metode cerdas yang tepat akan didapatkan kinerja yang baik.

5. KESIMPULAN DAN SARAN

5.1. Kesimpulan

Dengan menggunakan metode cerdas *Flower Pollination Algorithm (CSA)* sebagai metode penalaan *PID Controller*, didapatkan hasil penalaan parameter nilai *PID* yang optimal di mana, $k_p_{flo} = 0.9441$, $Ki_{flo} = 0.9311$, $kd_{flo} = 0.0840$.

Dari hasil simulasi dapat disimpulkan, respon kecepatan motor Sinkron dengan PID FPA, didapatkan *settling time* yang sangat cepat dibanding dengan metode trial-error, PSO, di mana sistem semakin cepat menuju kondisi *steady*.

5.2. Saran

Untuk pengembangan studi lanjut dari penelitian ini, dapat menggunakan metode cerdas yang lain untuk menala parameter PID.

6. DAFTAR PUSTAKA

- [1] Yang, X,S. "Cuckoo Search and Firefly Algorithms Theory and Applications". 2013.
- [2] Yang, X,S, "Nature-Inspired Metaheuristic Algorithms", Frome: Luniver Press. 2008.
- [3] Wudai Liao, "Optimization of PID Control for DC Motor Based On Artificial Bee Colony Algorithm". IEEE, 2014.
- [4] X. Wang, and M. Zhang, "Optimizing PID parameters by using improved particle swarm optimization algorithm", Process Automation Instrumentation, no. 2, pp. 1-9, 2004.
- [5] H. He and F. Qian, "The PID parameter tuning based on immune evolutionary algorithm", Microcomputer Information, vol. 27, no. 5, pp. 1174-1176, 2007.
- [6] W. Zhang, "Increment PID controller based on immunity particle swarm optimization algorithm", Microcomputer Information, vol. 28, no.7, pp. 67-69, 2010.
- [7] Nitish Katal. "Optimal Tuning of PID Controller for DC Motor using Bio-Inspired Algorithms", International Journal of Computer Applications.2012.
- [8] Bharat Bhushan, "Adaptive control of DC motor using bacterial foraging algorithm", Applied Soft Computing sciencedirect. 2011.
- [9] Ashu Ahuja, "Design of fractional order PID controller for DC motor using evolutionary optimization techniques".WSEAS Transactions on Systems and Control, 2014.
- [10] Anant Oonsivilai, "Optimum PID Controller tuning for AVR System using Adaptive Tabu Search", 12th WSEAS International Conference on COMPUTERS, Heraklion, Greece, July 23-25, 2008.
- [11] Umesh Kumar Bansal, "Speed Control of DC Motor Using Fuzzy PID Controller", Advance in Electronic and Electric Engineering. 2013.
- [12] K. Premkumar, "Fuzzy PID supervised online ANFIS based speed controller for brushless dc motor". Sciencedirect. Neuro Computing, 2015.
- [13] Djalal.M.R, D. Ajiatmo, A. Imran, I. Robandi, "Desain Optimal Kontroler PID Motor DC Menggunakan Cuckoo Search Algorithm", Seminar Nasional Teknologi Informasi dan Aplikasinya (SENTIA) Politeknik Negeri Malang, 2015.
- [14] D.Lastomo, Djalal.M.R, Widodo, I.Robandi, "Optimization of PID Controller Design for DC Motor Based on Flower Pollination Algorithm", The 2015 International Conference on Electrical, Telecommunication and Computer Engineering (ELTICOM 2015), 2015.