

# DESAIN OPTIMAL KONTROLER PID MOTOR DC MENGGUNAKAN CUCKOO SEARCH ALGORITHM

Muhammad Ruswandi Djalal<sup>1</sup>, Dwi Ajiatmo<sup>2</sup>, Andi Imran<sup>3</sup>, Imam Robandi<sup>3</sup>

<sup>1,2,3,4</sup>Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember

<sup>1</sup>[muhammad.ruswandi13@mhs.ee.its.ac.id](mailto:muhammad.ruswandi13@mhs.ee.its.ac.id), <sup>2</sup>[ajiatmo@gmail.com](mailto:ajiatmo@gmail.com), <sup>3</sup>[andiimran7@gmail.com](mailto:andiimran7@gmail.com), <sup>4</sup>[robandi@ee.its.ac.id](mailto:robandi@ee.its.ac.id)

---

## ABSTRAK

Penggunaan kontroler PID (Proporsional-Integral-Derivatif) pada sebuah motor DC sangat banyak digunakan, karena strukturnya yang sederhana, keandalan yang kuat dan mudah digunakan. Akan tetapi, penggunaan kontroler PID dibutuhkan pengaturan parameter yang tepat untuk mendapatkan kinerja yang optimal pada motor. Umumnya 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 satunya dengan metode cerdas. Untuk itu pada penelitian ini akan digunakan metode cerdas berbasis *Cuckoo Search Algorithm (CSA)*, untuk mengoptimasidan menentukan parameter yang tepat dari PID. CSA adalah salah satu metode cerdas yang terinspirasi dari perilaku burung cuckoo dalam menempatkan telurnya disarang burung lain yang dia pilih secara acak, konsep inilah yang diadaptasi dan diterapkan menjadi algoritma cerdas untuk menyelesaikan masalah optimasi. Dari hasil yang diperoleh metode CSA dapat dengan baik menala parameter PID, sehingga overshoot yang dihasilkan tidak ada dan settling time sangat cepat. Pada penelitian ini juga akan dibahas dan dibandingkan kontroler tanpa PID, dengan PID trial-error dan dengan PID CSA.

**Kata Kunci :** Motor DC, PID, *Trial-Error*, *Cuckoo*, *Overshoot*, *Settling time*

---

## 1. Pendahuluan

Setelah lebih dari setengah abad penelitian dan pengembangan, kontrol PID, dengan struktur sederhana, stabilitas yang baik dan ketahanan yang kuat, memainkan peran penting dalam kontrol industri [1]. Motor DC telah banyak digunakan dalam aplikasi industri. Masalah utama yang sering dibahas dalam kontrol PID adalah penyesuaian parameter. Salah satu teknik yang sering digunakan adalah kontrol konvensional trial-error, namun untuk metode ini sulit untuk menyesuaikan parameter, sehingga dibutuhkan waktu lama [2] serta akurasi kontrol tidak baik. Dalam beberapa tahun terakhir, para peneliti telah banyak menggunakan metode cerdas (*Artificial Intelligent*) untuk penentuan parameter PID motor DC. Para peneliti mulai mempelajari perilaku cerdas dari hewan untuk diaplikasikan untuk menyelesaikan masalah optimasi. Hewan tersebut diantaranya lebah, semut, tawon dan menemukan suatu algoritma perilaku dari kebiasaan atau perilaku dari hewan tersebut.

Pada tahun 2009, seorang peneliti Xin Xhe Yang, menemukan sebuah metode cerdas yang disebut kecerdasan burung cuckoo dan membuat sebuah algoritma penyelesaian yang terinspirasi dari perilaku burung cuckoo dalam menempatkan telurnya. Beberapa metode optimasi berbasis

metode konvensional maupun metode cerdas telah banyak digunakan untuk mengoptimasi parameter PID motor DC, diantaranya Artificial Bee Colony, Particle Swarm Optimization, Tabu Search, Bacterial Foraging, Fuzzy Logic, Genetic Algorithm, dan Neural Network. (Wudai Liao, (2014); X. Wang, (2004); B. Liu, (2007); D. Chen, (2007); H. He, (2007); W. Zhang, (2010); Bhateshvar, Y.K. (2014); Nitish Katal. (2012); Bharat Bhushan. (2011); Ashu Ahuja. (2014); Anant Oonsivilai. (2008); Umesh Kumar Bansal. (2013); K. Premkumar. (2015);)

Pada penelitian ini akan digunakan salah satu metode cerdas (*Artificial Intelligent*) untuk menala parameter PID motor DC yaitu dengan metode *Cuckoo Search Algorithm (CSA)* dan akan dianalisa dan dibandingkan respon kecepatan motor DC dengan metode konvensional PID trial-error serta motor DC tanpa kontroler.

## 2. Pemodelan Sistem [Wudai Liao, 2014]

Motor DC adalah motor listrik yang memerlukan suplai tegangan arus searah (DC) pada kumparan medan untuk diubah menjadi energi gerak mekanik. Kumparan medan pada motor dc disebut stator (bagian yang tidak berputar) dan kumparan jangkar disebut rotor (bagian yang berputar). Motor arus searah, sebagaimana namanya, menggunakan arus langsung yang tidak

langsung/direct-unidirectional. Motor DC memiliki 3 bagian utama untuk dapat berputar yaitu : Kutub Medan, Dinamo dan Komutator.

Untuk merancang dan mensimulasikan suatu sistem kontrol kecepatan motor DC diperlukan adanya model matematika dari plant yang akan dikontrol. Sehingga dalam memudahkan analisis sistem bisa digambarkan ke dalam model statis, model fisis (rangkaiannya listrik) dan dinamik (diagram blok dan transfer function).

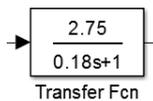
**2.1 PEMODELAN MOTOR DC**

Pemodelan motor DC yang digunakan mengacu pada sebuah paper [1] yang di mana menggunakan pemodelan transfer function. Motor DC yang digunakan di sini adalah menggunakan motor DC magnet permanen, rating kecepatan 1400 rpm dengan kecepatan saat pengukuran 1250 rpm. Dengan memasukkan kecepatan motor DC, maka transfer function motor DC adalah,

$$G(s) = \frac{K_m}{T_m s + 1} \tag{1}$$

Di mana, Km adalah gain mekanik dan Tm adalah waktu konstan mekanik.

$$G(s) = \frac{2.75}{0.18s + 1} \tag{2}$$



**Gambar 1.** Pemodelan Motor DC di Simulink

**2.2 PEMODELAN KONTROLER PID**

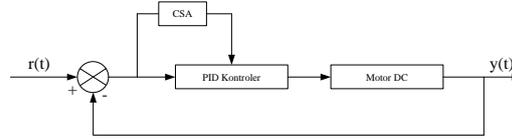
Kontrol PID adalah salah satu kontrol yang sudah banyak digunakan pada aplikasi industri karena strukturnya yang sederhana, PID kontrol

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

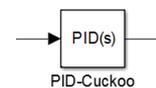
Di mana, u(t) adalah nilai kontrol yang dihitung oleh kontroler PID, Kp adalah koefisien proporsional, Ti adalah integral waktu konstan dan Td adalah diferensial time konstan. Fungsi dari ketiga elemen tersebut adalah :

- Proporsional: menggambarkan sinyal deviasie (t) dari sistem kontrol proporsional. ketikasinyal 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 penyimpangansinyal, memperkenalkan sinyal koreksi sebelum penyimpangan nilai sinyal menjadi lebih besar dan mempercepat responsistem untuk mengurangi pengaturan waktu.

Oleh karena itu, merancang kontroler PID terutama berarti menentukan tiga parameter, serta bagaimana mengkonfigurasi ke tigaparameter PID (Kp, Ki, Kd). Dalam penelitian ini, *Cuckoo Search Algorithm* diusulkan untuk mencari parameter optimal PID. Blok diagram sistem kontrol ditunjukkan padagambar 1.



**Gambar2.** Sistem Kontroler PID-CSA



**Gambar 3.** Pemodelan PID di Simulink

**3 Cuckoo Search Algorithm [Yang, 2009]**

Burung cuckoo dalam berkembangbiak memiliki cara yang beragam dan unik. Dari sekian spesies cuckoo diketahui bahwa 59 diantaranya merupakan cuckoo yang bersifat parasit dalam berkembangbiak yaitu mereka memanfaatkan sarang burung lain yang berbeda spesies untuk menetaskan telur mereka. Bahkan tak jarang burung cuckoo yang menaruh telur pada sarang burung cuckoo yang lain. Ada beberapa jenis cuckoo yang membuang telur dari induk asli pada sarang untuk meningkatkan kemungkinan menetas telur mereka. Mungkin saja terjadi konflik antara burung host dan burung cuckoo pada saat burung cuckoo menaruh telurnya, sehingga burung host membuang telur cuckoo tersebut atau meninggalkan sarang mereka kemudian membuang sarang yang baru. Perilaku parasit yang lain adalah saat burung cuckoo menetas, biasanya telur burung cuckoo menetas lebih awal dari telur burung host, telur burung yang belum menetas tadi dibuang dari sarang agar anak cuckoo tersebut mendapatkan lebih banyak makanan.

4. Setiap cuckoo meletakkan satu telur pada satu waktu dan membuang telur didalamnya pada sarang yang dipilih acak.
  5. Sarang terbaik dengan kualitas telur tertinggi akan dilanjutkan ke generasi berikutnya.
- Jumlah sarang burung yang tersedia berjumlah tetap, dan peluang telur yang diletakkan oleh cuckoo yang ditemukan burung inang sebesar pa ∈ [0,1]. Dalam kasus ini, burung dapat membuang telur cuckoo atau meninggalkan sarang, lalu membuat sarang baru.

Aturan terakhir dapat disederhanakan dengan pendekatan fraksi pa dari n sarang digantikan dengan sarang baru (dengan solusi baru secara acak). Untuk masalah maksimisasi, kualitas atau fitness dari sebuah solusi dapat sebanding dengan nilai dari objective function. Bentuk lain dari fitness dapat didefinisikan dalam cara yang sama dengan

fungsi fitness pada algoritma genetika. Untuk mempermudah, dapat menggunakan representasi yang sederhana bahwa setiap telur dalam sarang merepresentasikan sebuah solusi, dan telur cuckoo merepresentasikan solusi baru, tujuannya adalah untuk menggunakan yang baru dan potensial solusi yang lebih baik (cuckoos) untuk mengganti solusi yang tidak bagus pada sarang. Kemudian telur-telur tadi akan dievolusikan, telur yang lebih akan menggantikan telur lainnya yang diukur oleh fitness, layaknya dalam GA. Dalam suatu sarang host bisa saja terdapat dua telur, dengan kata lain sarang bisa menyimpan lebih dari satu solusi hanya saja untuk menyederhanakan permasalahan, suatu sarang hanya akan menyimpan satu solusi (telur). Berdasarkan tiga aturan, langkah-langkah dasar Cuckoo Search (CS) dapat diringkas sebagai pseudo code yang ditunjukkan pada tabel berikut.

**Tabel2. Pseudo code dari Cuckoo Search (CS)**

```

Begin
Fungsi Objektif  $f(x)$ ,  $x = (x_1, \dots, x_d)^T$ 
Inisialisasi populasi dari  $n$  sarang burung target  $x_i$  ( $i = 1, 2, \dots, n$ )
While ( $t < \text{generasiTotal}$ ) atau (kriteria lain untuk berhenti)
  Evaluasi nilai kualitas dari masing-masing burung cuckoo
  Pilih dari burung cuckoo secara acak dan lakukan random walk
  Jika ( $F_i > F_j$ )
    Gantikan burung cuckoo  $j$  dengan burung cuckoo  $i$ 
  End If
  Reset ulang sarang-sarang dengan kondisi terburuk ( $P_a$ )
  Simpan sarang-sarang yang berhasil lolos
  Urutkan solusi dan cari yang terbaik
End While
  Proses hasil dan visualisasi
End
    
```

Ketika generasi solusi baru  $x^{(t+1)}$  untuk seekor cuckoo  $i$ , Levy flight ditunjukkan sebagai berikut :

$$x_i^{(t+1)} = x_i^{(t)} + \alpha \oplus \text{Levy}(\lambda) \quad (3)$$

$\alpha > 0$  adalah ukuran tahapan yang seharusnya berhubungan dengan skala dari masalah yang menarik. Dalam kebanyakan kasus, dapat menggunakan  $\alpha = 1$ . Levy flight pada dasarnya memberikan jalan yang acak ketika tahapan acak panjang diambil dari distribusi Levy :

$$\text{Levy} \sim u = t^{-\lambda}, \quad (1 < \lambda \leq 3) \quad (4)$$

Mempunyai variasi yang tak terbatas dengan rata-rata tak terbatas. Tahapan penting bagian dari sebuah proses random walk dengan kekuatan rendah, distribusi tahapan panjang dengan bagian terakhir yang berat. Beberapa dari solusi baru seharusnya digenerasi oleh jalan berputar Levy mencapai solusi terbaik sejauh ini, kecepatan akan naik pada pencarian local. Tetapi, sebuah fraksi substansial dari solusi baru seharusnya digenerasi oleh medan acak yang jauh dan lokasi harus cukup jauh dari solusi terbaik, ini akan membuat sistem tidak akan terjebak pada optimasi lokal.

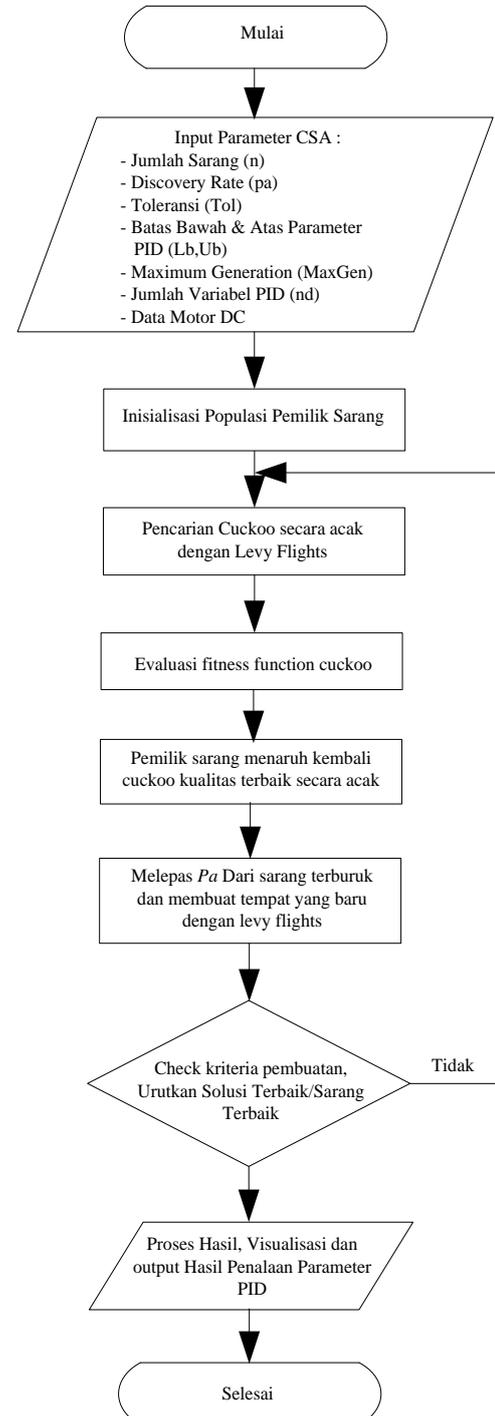
**3.1. PENALAN PID DENGAN CSA**

Gambar 5 menunjukkan diagram alir algoritma metode *Cuckoo Search Algorithm* (CSA) yang digunakan pada penelitian ini untuk menala parameter PID. Fungsi objektif yang digunakan

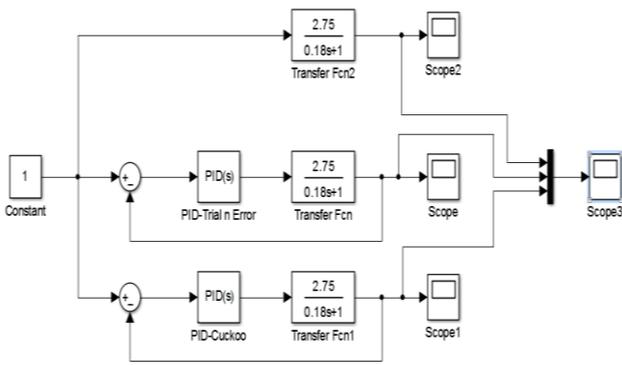
untuk menguji kestabilan sistem adalah dengan *Integral Time Absolut Error* (ITAE).

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

Parameter PID yang ditala oleh CSA adalah  $K_p$ ,  $K_i$  dan  $K_d$ . Adapun untuk diagram alir proses penalaan parameter PID dengan menggunakan metode *Cuckoo Search Algorithm* (CSA) ditunjukkan oleh *flowchart* pada Gambar 5 dan gambar 4 menunjukkan pemodelan motor DC pada Simulink Matlab 2013, tanpa kontrol, dengan trial-error, dan PID Cuckoo.



**Gambar 4.** Flowchart Penelitian



Gambar 5. Pemodelan Motor DC di Simulink

Adapun data parameter-parameter berikut,

Tabel 3. Parameter CSA

Parameter	Nilai
Number of nests	25
Discovery rate of alien eggs/solutions	0.25
Toleransi	1.0e-5
Max Generation	50
Jumlah Variabel (nd)	5
Batas Atas Kp, Ki, Kd	[40 30 0.1]
Batas Bawah Kp, Ki, Kd	[30 20 0]

Berikut hasil optimasi dengan metode CSA.

```

Command Window
In cuckoo_search_new at 79
Total number of iterations=2500

fmin =
    19.3811

bestnest =
    34.6724    25.3488    0.0081

kp_cu =
    34.6724

ki_cu =
    25.3488

kd_cu =
    0.0081

ans =
    34.6724    25.3488    0.0081
    
```

Gambar 6. Hasil Optimasi Kontrol PID Motor DC dengan Cuckoo Search Algorithm di MATLAB

Tabel 4. Hasil Optimasi SMES-PID dengan CSA

Total number of iterations=2500
fmin =
19.3811
bestnest =
34.6724 25.3488 0.0081
kp_cu =
34.6724
ki_cu =
25.3488
kd_cu =
0.0081

Hasil optimasi CSA didapatkan nilai *fitness function* sebesar 19.3811, dengan 50 iterasi, nilai bestnest merupakan sarang terbaik dari burung cuckoo, yang di mana diketahui sebagai hasil optimasi parameter PID, yaitu Kp, Ki dan Kd. Tabel 4 menunjukkan nilai hasil optimasi parameter PID ditela oleh CSA.

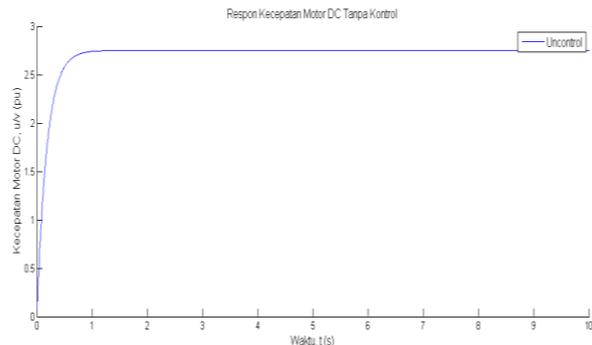
Tabel 5. Hasil Penalaan Parameter PID

Parameter	Trial-Error	Cuckoo Search
<i>Kp</i>	10	34.6724
<i>Ki</i>	0.01	25.3488
<i>Kd</i>	0.01	0.0081

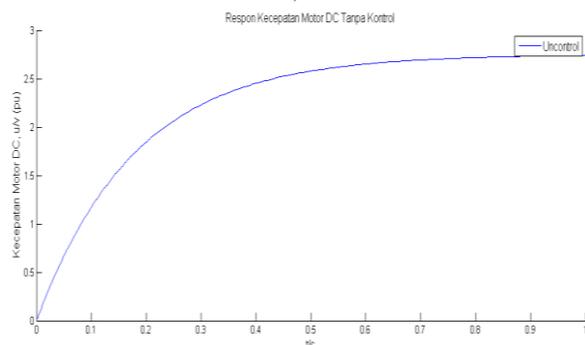
#### 4 Hasil Simulasi Dan Analisis

##### 4.1 RESPON KECEPATAN MOTOR DC TANPA CONTROLLER

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



Gambar 7. Respon Kecepatan Motor DC tanpa kontrol, t=10s.

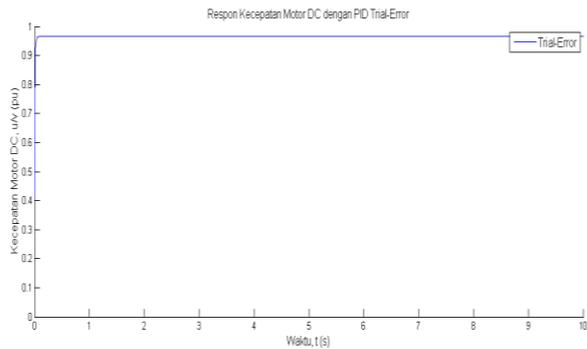


Gambar 7. Respon Kecepatan Motor DC tanpa kontrol, t=1s.

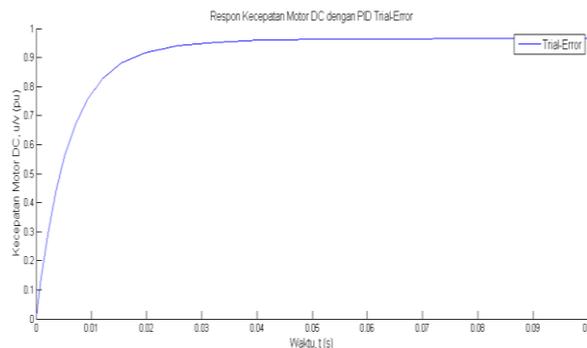
Dari hasil simulasi tanpa kontroler, didapatkan respon kecepatan motor DC yang sangat tinggi, hal ini dikarenakan sistem tidak ada umpan balik, sehingga motor bekerja tanpa ada batasan dan untuk sistem yang seperti ini sangat dihindari. Untuk itu sangat diperlukan desain sistem kontrol yang tepat dengan penambahan kotroler PID pada motor DC, sehingga kecepatan yang dihasilkan dapat dikontrol sesuai dengan beban yang dikopel oleh motor DC.

**4.2 RESPON KECEPATAN MOTOR DC DENGAN PID TRIAL-ERROR**

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



**Gambar 8.** Respon Kecepatan Motor DC dengan PID Trial-Error,  $t=10s$

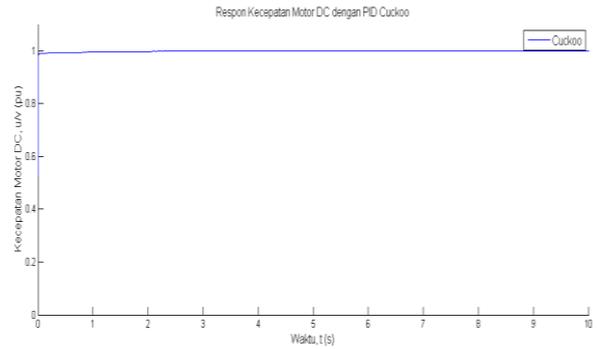


**Gambar 9.** Respon Kecepatan Motor DC dengan PID Trial-Error,  $t=0.1s$

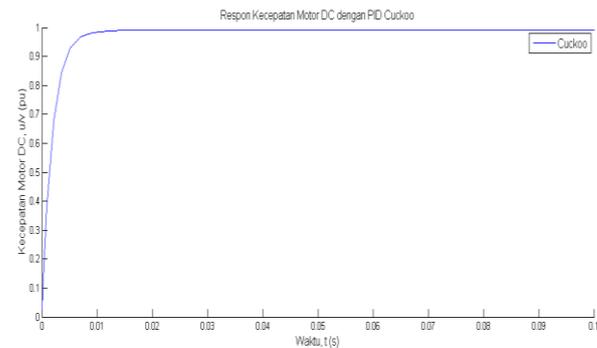
Gambar grafik di atas menunjukkan grafik respon frekuensi motor DC dengan PID Trial-Error. Dari grafik di atas didapatkan *settling time* yang sangat lama, namun masih lebih baik dari yang tanpa kontrol. Dari grafik dapat dilihat, sistem belum berada pada kondisi *steady* pada detik ke  $>10s$ . Hal 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.

**4.3 RESPON KECEPATAN MOTOR DC DENGAN PID CUCKOO**

Simulasi yang ketiga adalah kontrol motor DC dengan PID Cuckoo, berikut hasil simulasinya.

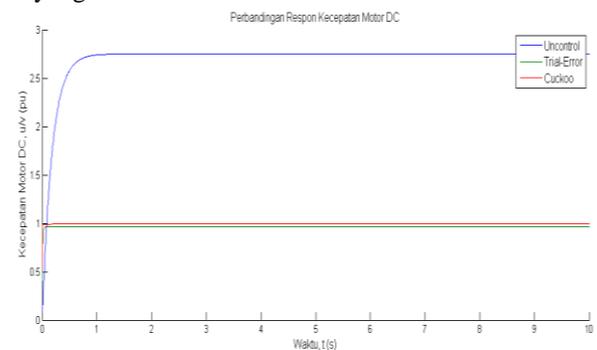


**Gambar 10.** Respon Kecepatan Motor DC dengan PID Cuckoo,  $t=10s$

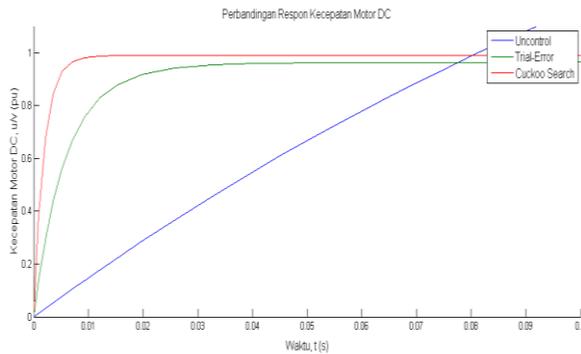


**Gambar 11.** Respon Kecepatan Motor DC dengan PID Cuckoo  $t=0.1s$

Gambar grafik di atas menunjukkan grafik respon frekuensi motor DC dengan PID Cuckoo. Dari grafik di atas didapatkan *settling time* yang sangat cepat dibanding dengan metode trial-error, di mana sistem sudah berada pada kondisi *steady* pada detik ke 0.02s. 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 DC. Gambar berikut menunjukkan perbandingan dari kontrol motor DC yang telah disimulasikan.



**Gambar 12.** Perbandingan Respon Kecepatan Motor DC,  $t=10s$



**Gambar 12.** Perbandingan Respon Kecepatan Motor DC,  $t=0.1s$

Aplikasi motor DC sangat banyak digunakan, karena motor DC sangat cocok untuk aplikasi terutama pengontrolan, sehingga dibutuhkan desain kontroler motor DC yang tepat, dalam hal ini sangat diusulkan untuk menggunakan kontroler PID karena sangat simple untuk pengontrolan sistem dengan penalaan parameter menggunakan metode cerdas yang tepat akan didapatkan kinerja yang baik.

### 5 Kesimpulan

Dengan menggunakan metode cerdas *Cuckoo Search Algorithm (CSA)* sebagai metode penalaan *PID Controller*, didapatkan hasil penalaan parameter nilai *PID* yang optimal di mana,  $K_p$  sebesar 34.6724,  $K_i$  sebesar 25.3488 dan  $K_d$  sebesar 0.0081.

Dari hasil simulasi dapat disimpulkan, respon kecepatan motor DC dengan *PID Cuckoo*, didapatkan *settling time* yang sangat cepat dibanding dengan metode *trial-error*, di mana sistem sudah berada pada kondisi *steady* pada detik ke 0.02s, ini berarti dengan menggunakan metode cerdas untuk mengoptimasi parameter *PID*, hasil yang didapat untuk kinerja *PID* sangat baik karena menunjukkan respon yang cepat untuk kontrol motor DC.

### DAFTAR PUSTAKA

Yang, X.S. (2013) : *Cuckoo Search and Firefly Algorithms Theory and Applications*. 2013.

Yang, X.S. (2008) : *Nature-Inspired Metaheuristic Algorithms*. Frome: Luniver Press. 2008.

Wudai Liao, (2014) : Optimization of PID Control for DC Motor Based On Artificial Bee Colony Algorithm.2014.

X. Wang, and M. Zhang, (2004) : *Optimizing PID parameters by using improved particle swarm optimization algorithm*. Process Automation Instrumentation, no. 2, pp. 1-9, 2004.

B. Liu, J. Tan, and C. Huang, (2007) : *Research and Application of an Improved PID Control Algorithm*. Microcomputer Information, vol. 6, no. 1, pp. 15-17, 2007.

D. Chen, K. Fang, and Q. Chen. (2007) : *Application of genetic algorithm in PID*

*parameters optimization*. Microcomputer Information, vol. 23, no.3, pp. 35-36, 2007.

H. He and F. Qian.(2007) : *The PID parameter tuning based on immune evolutionary algorithm*. Microcomputer Information, vol. 27, no. 5, pp. 1174-1176, 2007.

W. Zhang. (2010) : *Increment PID controller based on immunity particle swarm optimization algorithm*, Microcomputer Information, vol. 28, no.7, pp. 67-69, 2010.

Bhateshvar, Y.K. (2014) : *Power-Frequency Balance With Superconducting Magnetic Energy Storage Using Optimized Intelligent Controller*. Energetika.2014.

Nitish Katal. (2012) : *Optimal Tuning of PID Controller for DC Motor using Bio-Inspired Algorithms*. International Journal of Computer Applications.2012.

Bharat Bhushan. (2011) : *Adaptive control of DC motor using bacterial foraging algorithm*. Applied Soft Computing sciencedirect.2011.

Ashu Ahuja. (2014) : *Design of fractional order PID controller for DC motor using evolutionary optimization techniques*. WSEAS Transactions on Systems and Control.2014.

Anant Oonsivilai. (2008) : *Optimum PID Controller tuning for AVR System using Adaptive Tabu Search*, 12th WSEAS International Conference on COMPUTERS, Heraklion, Greece, July 23-25, 2008.

Umesh Kumar Bansal.(2013) : *Speed Control of DC Motor Using Fuzzy PID Controller*. Advance in Electronic and Electric Engineering.2013.

K. Premkumar. (2015) : *Fuzzy PID supervised online ANFIS based speed controller for brushless dc motor*. Sciencedirect. Neurocomputing.2015.