

PENGEMBANGAN METODA SELF TUNING PARAMETER PID CONTROLLER DENGAN MENGGUNAKAN GENETIC ALGORITHM PADA PENGATURAN MOTOR INDUKSI SEBAGAI PENGGERAK MOBIL LISTRIK

Era Purwanto, Ananto Mukti Wibowo, Soebagio, Mauridhi Hery Purnomo

Electrical Engineering Department of Institute Technology Sepuluh Nopember Surabaya, Indonesia
Kampus ITS, Keputih – Sukolilo, Surabaya 60111
E-mail:era@eepis-its.edu

ABSTRAK

Pada saat ini PID controller masih banyak digunakan di berbagai sektor industri, karena ketangguhannya untuk menghandel permasalahan yang ada di industri, tetapi terdapat satu kekurangan yaitu metode tuning, proses tuning harus dilakukan dengan cara coba - coba. Prosedur tuning lebih banyak dilakukan secara manual dengan metode trial and error yang belum tentu benar, untuk mengatasi permasalahan tersebut dibutuhkan suatu pendekatan alternatif yang dapat mencapai suatu tingkat automasi dari proses tuning dan orang yang melakukan proses tuning tidak harus memiliki pengetahuan luas mengenai sistem pengaturan. Dalam kasus ini metode alternatif yang digunakan adalah Genetic Algorithm (GA). GA diimplementasi untuk mendapatkan kombinasi parameter P,I, dan D dari kontroler PID dalam simulasi pengaturan kecepatan motor induksi tiga fasa sehingga kecepatan motor dapat dipertahankan agar sama dengan kecepatan referensi dalam steady-state. Hasil dari simulasi yang dilakukan menunjukkan bahwa repons kecepatan yang dihasilkan sangatlah bagus.

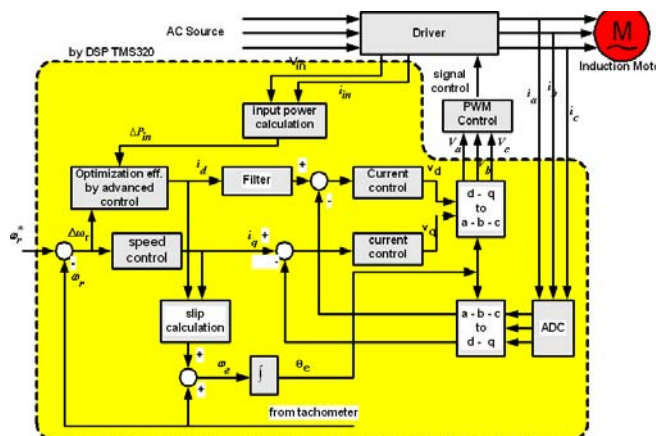
Kata Kunci: self-tuning, genetic Algorithm,dq-model,vector control

1. PENDAHULUAN

Pada saat ini energi listrik masih merupakan energi alternatif untuk digunakan pada alat transportasi, pada sistem ini motor listrik akan menggantikan mesin diesel atau mesin konvensional sebagai penggerak utama, mesin listrik yang dapat digunakan pada saat ini sangat banyak sekali mulai dari motor DC sampai dengan motor AC. Motor dc adalah motor yang paling ideal digunakan pada sistem pengendalian elektrik, karena pengendaliannya dapat dilakukan secara terpisah (*decoupled system*). Namun demikian terdapatnya komutator menyebabkan diperlukannya perawatan rutin dan berkala. Hal ini menyebabkan sistem menjadi kurang efisien. Karena itu penggunaan motor dc perlahan-lahan digeser oleh motor induksi, karena bentuknya kokoh, pemeliharaannya mudah, dan efisien, maka motor induksi banyak digunakan.

Secara konvensional motor induksi dioperasikan pada kecepatan yang tetap sesuai dengan frekuensi sumber tegangan serta jumlah kutub dari motor induksi, tetapi pada mobil listrik diperlukan kecepatan variable terutama pada kondisi jalan yang ramai. Pengaturan kecepatan pada motor induksi jauh lebih sulit dibandingkan dengan motor DC karena tidak ada hubungan yang linier antara arus motor dan torsi yang dihasilkan seperti pada motor DC. Suatu metode yang kemudian disebut sebagai *vector control* digunakan untuk mengatur kecepatan motor induksi pada rentang kecepatan yang besar serta presisi dan respon yang cepat. Namun demikian motor induksi adalah sistem kopel (*coupled*) dimana ada saling ketergantungan antara torka dan fluksi. Pada *vector control*, pengaturan motor induksi ini telah diubah seperti pengaturan motor arus searah, sehingga dalam pengaturan motor

induksi dapat digunakan berbagai kontroler seperti pada pengaturan kecepatan motor arus searah. Dengan berkembangnya metoda Vector control maka kelemahan pada motor induksi dapat diatasi dengan mengubah sistem couple menjadi decouple, sehingga antara komponen torka dan komponen fluksi bisa dikendalikan secara terpisah. Vector control adalah suatu metode pengaturan medan pada motor ac, dimana dari sistem coupled diubah menjadi sistem decoupled. Dengan teknik ini maka pengaturan motor induksi menjadi sangat penting, bahkan menjadi motor yang sangat mendominasi pada sistem pengendalian. Dengan sistem ini arus penguatan dan arus beban motor dapat dikontrol secara terpisah, sehingga torka dan fluksi juga dapat diatur secara terpisah, dengan demikian torsi dan fluksi juga dapat diatur secara terpisah, seperti halnya motor dc. Dari model mekanik diatas maka dikembangkanlah model elektrik seperti ditunjukkan pada gambar 1.



Gambar 1. Sistem Pengaturan Motor Induksi

Motor induksi banyak dipakai dalam industri karena bentuknya sederhana, harganya murah, konstruksinya kokoh (robust), dan pemeliharaannya mudah. Umumnya motor induksi dioperasikan untuk kecepatan konstan. Namun bila beban berubah, maka kecepatan akan berubah, selain itu kecepatan variabel memberikan efisiensi yang lebih tinggi bila beban yang diberikan bervariasi, karena itu diperlukan sistem kontrol untuk dapat memperbaiki kinerjanya.

Penggunaan kontroler PID dalam suatu sistem mempunyai kelemahan, yakni bahwa parameter-parameter dalam kontroler harus selalu diubah (tuned up) bila terjadi perubahan didalam sistem, perubahan tersebut akan menyebabkan terjadinya tuning kembali dari parameter-parameter PID untuk mempertahankan kondisi yang sudah ditentukan, metode tuning yang akan digunakan untuk mencari parameter-parameter kontroler PID adalah dengan menggunakan metoda *Genetic Algorithms*. Sasaran penelitian ini adalah untuk mendapatkan parameter-parameter kontroler PID dengan metode *Genetic Algorithms* dalam pengaturan kecepatan motor induksi tiga fase.

2. DQ MODEL MOTOR INDUKSI

Untuk keperluan analisa diperlukan satu model yang akan dapat digunakan untuk membantu analisa digunakan. Pada penelitian ini model motor induksi yang digunakan adalah model d-q, dimana persamaan adalah sebagai berikut:

$$v_{qs} = r_s i_{qs} + \frac{d\lambda_{qs}}{dt} + \omega_e \lambda_{ds} \quad (1)$$

$$v_{ds} = r_s i_{ds} + \frac{d\lambda_{ds}}{dt} - \omega_e \lambda_{qs} \quad (2)$$

$$v_{qr}' = r_r' i_{qr}' + \frac{d\lambda_{qr}'}{dt} + (\omega_e - \omega_r) \lambda_{dr}' \quad (3)$$

$$v_{dr}' = r_r' i_{dr}' + \frac{d\lambda_{dr}'}{dt} + (\omega_e - \omega_r) \lambda_{qr}' \quad (4)$$

Persamaan untuk fluksi adalah

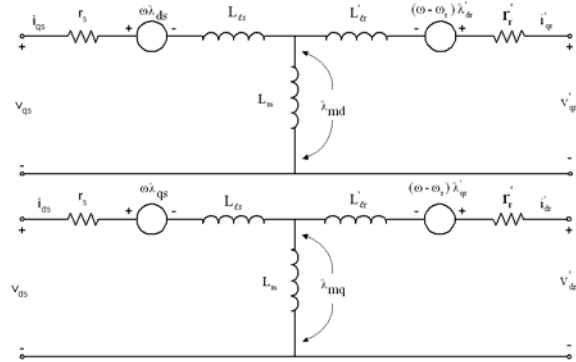
$$\lambda_{qs} = L_{\ell s} i_{qs} + L_m (i_{qs} + i_{qr}') \quad (5)$$

$$\lambda_{ds} = L_{\ell s} i_{ds} + L_m (i_{ds} + i_{dr}') \quad (6)$$

$$\lambda_{ds} = L_{\ell s} i_{ds} + L_m (i_{ds} + i_{dr}') \quad (7)$$

$$\lambda_{dr}' = L_{\ell r}' i_{dr}' + L_m (i_{ds} + i_{dr}') \quad (8)$$

Persamaan (1) s/d (4) dapat dinyatakan oleh diagram ekivalen pada gambar 1.



Gambar 2. Rangkaian ekivalen motor induksi dalam koordinat d-q

Persamaan (1) sampai dengan (9) dapat dinyatakan dalam bentuk matriks sebagai berikut:

$$\bar{V}_{qd} = \bar{Z}_{qd} \bar{I}_{qd} \quad (9)$$

dengan

$$\bar{V}_{qd} = \begin{bmatrix} v_{qs} & v_{ds} & v_{qr}' & v_{dr}' \end{bmatrix}^t \quad (10)$$

$$\bar{I}_{qd} = \begin{bmatrix} i_{qs} & i_{ds} & i_{qr}' & i_{dr}' \end{bmatrix}^t \quad (11)$$

dan \bar{Z}_{qd} adalah matriks impedansi yang dinyatakan oleh:

$$\bar{Z}_{qd} = \begin{bmatrix} r_s + L_s p & \omega_e L_s & L_m p & \omega_e L_m \\ -\omega_e L_s & r_s + L_s p & -\omega_e L_m & L_m p \\ L_m p & (\omega_e - \omega_r) L_m & r_r' + L_r' p & (\omega_e - \omega_r) L_r' \\ -(\omega_e - \omega_r) L_m & L_m p & -(\omega_e - \omega_r) L_r' & r_r' + L_r' p \end{bmatrix} \quad (12)$$

dimana

$$L_s = L_m + L_{\ell s} \quad (13)$$

$$L_r' = L_m + L_{\ell r}' \quad (14)$$

$$p = d/dt \quad (15)$$

Bentuk lain dari pers. 9 dapat ditulis dalam bentuk:

$$\bar{V}_{qd} = \bar{R} \bar{I}_{qd} + \omega_e \bar{F} \bar{I}_{qd} + \omega_r \bar{G} \bar{I}_{qd} + p \bar{L} \bar{I}_{qd} \quad (16)$$

dimana

$$\bar{R} = \begin{bmatrix} r_s & 0 & 0 & 0 \\ 0 & r_s & 0 & 0 \\ 0 & 0 & r_r' & 0 \\ 0 & 0 & 0 & r_r' \end{bmatrix} \quad (17)$$

$$\bar{L} = \begin{bmatrix} L_s & 0 & L_m & 0 \\ 0 & L_s & 0 & L_m \\ L_m & 0 & L_r' & 0 \\ 0 & L_m & 0 & L_r' \end{bmatrix} \quad (18)$$

$$\bar{F} = \begin{vmatrix} 0 & L_s & 0 & L_m \\ L_s & 0 & L_m & 0 \\ 0 & L_m & 0 & L'_r \\ L_m & 0 & L'_r & 0 \end{vmatrix} \quad (19)$$

$$\bar{G} = \begin{vmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & -L_m & 0 & -L'_r \\ L_m & 0 & L'_r & 0 \end{vmatrix} \quad (20)$$

3.KONTROLER PID DENGAN GENETIC ALGORITHMS TUNING

3.1 Response Sistem

Respons suatu sistem kontrol selalu menunjukkan osilasi teredam sebelum mencapai *steady-state*. Penggolongan karakteristik respons transien suatu sistem kontrol terhadap masukan tangga satuan secara grafik ditunjukkan pada gambar 3.

Pengelompokan ini didefinisikan sebagai berikut:

1. **Waktu naik**, t_r

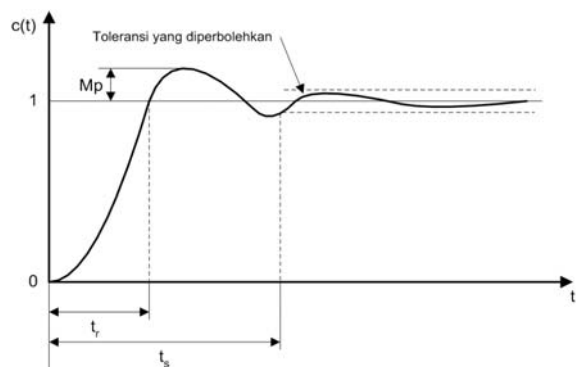
Waktu naik adalah waktu yang diperlukan oleh respons untuk naik dari 10% menjadi 90%, 5% menjadi 95%, atau 0% menjadi 100% dari nilai akhir yang biasa digunakan. Untuk sistem atas redaman waktu naik yang biasa digunakan 10% menjadi 90%.

2. **Overshoot maksimum**, M_p

Overshoot maksimum adalah nilai puncak kurva respons diukur dari satuan.

3. **Waktu steady-state**, t_s

Waktu *steady-state* adalah waktu yang diperlukan untuk merespons kurva agar dapat mencapai dan tetap berada dalam gugus nilai akhir ukuran yang disederhanakan dengan presentase mutlak harga akhirnya (biasanya 2% atau 5%). Waktu *steady-state* tadi dihubungkan tetapan waktu terbesar sistem kontrol.

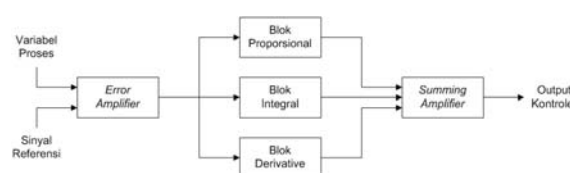


Gambar 3. Kurva respons tangga satuan menunjukkan t_r , M_p dan t_s

3.2 Kontrol PID

Sesuai dengan namanya, kontroler ini merupakan kombinasi dari tiga sistem kontrol yaitu

proportional, integral dan derivative. Jika masing-masing dari ketiga kontroler tersebut berdiri sendiri, hasil yang dicapai kurang bagus sebab masing-masing memiliki kelemahan dan kelebihan sendiri-sendiri. Karena itu kombinasi dari ketiga sistem kontrol tersebut diharapkan dapat mengeliminasi kelemahan masing-masing dan mampu memberikan kontribusi dari kelebihan masing-masing. Blok diagram secara umum dari kontroler PID ditunjukkan pada gambar 3. Tiap elemen menerima sinyal error yang sama dan output dari semua elemen tersebut dijumlahkan melalui *summing amplifier*.



Gambar 4 . Blok Diagram Kontroler PID

Secara umum fungsi dari masing-masing kontroler dalam kontroler PID adalah sebagai berikut:

- **Proportional**

- Berfungsi untuk mempercepat terjadinya respons terhadap sinyal error.
- Bekerja efektif pada daerah sebelum sistem mencapai daerah setpoint/kondisi start.

- **Integral**

- Berfungsi memelihara sinyal kontrol konstan.
- Bekerja efektif pada daerah di mana sistem mencapai set point.

- **Derivative**

- Berfungsi mendapatkan sinyal kontrol dari perubahan errornya.
- Bekerja efektif pada daerah transien.

3.3 Genetic Algorithms

Proses kerja GA diawali dengan inialisasi satu rangkaian nilai random yang disebut *populasi*. Setiap individu di dalam populasi disebut *kromosom*. Sebuah kromosom dapat direpresentasikan dalam bentuk simbol-simbol string biner, floating point, integer, abjad. Kromosom-kromosom ini berkembang melalui beberapa iterasi yang disebut *generasi*. Setiap generasi, kromosom-kromosom ini dievaluasi dengan menggunakan ukuran *fitness* melalui fungsi tujuan (*objective function*) dan batasan-batasan fungsi yang merupakan permasalahan sehingga individu dengan solusi yang terbaik yang terpilih. Untuk menghasilkan generasi selanjutnya ($t+1$) sebagai kromosom baru yang disebut *offspring*, dibentuk melalui penggabungan dua kromosom generasi saat ini (t) dengan menggunakan operator *crossover* dan memodifikasi sebuah kromosom menggunakan operator mutasi. Satu generasi baru dibentuk melalui proses seleksi sesuai dengan *fitness value* kromosom

orang tua dan kromosom yang fit yang akan diturunkan. Kromosom dengan fitness terbesar memiliki probabilitas tertinggi untuk dipilih.

3.4 Tuning Kontroler PID Dengan Genetic Algorithms

Untuk menginisialisasi GA ini, populasi awal dibentuk dari 20 individu yang diambil secara acak dengan memperhatikan batas atas dan batas bawah kromosom dimana tiap individu terdiri atas 3 kromosom bernilai real yang merepresentasikan nilai parameter kontroler PID.

Langkah awal dalam proses GA adalah mengevaluasi tiap kromosom dan memberi nilai fitness. Tiap individu dievaluasi untuk mendapatkan variabel untuk parameter-parameter kontroler PID. Setelah variabel parameter PID didapatkan, variabel-variabel tersebut kemudian dimasukkan dalam model Simulink yang mencerminkan model sebenarnya dari sistem yang digunakan. Setelah simulasi selesai, variabel tersebut dikembalikan lagi kedalam GA agar respons simulasi dievaluasi dan individu tersebut diberi nilai fitness.

Individu yang terpilih untuk generasi berikutnya ditentukan secara probabilistik berdasarkan nilai fitness individu tersebut. Dari individu yang terpilih diatas, generasi berikutnya diperoleh dari proses mutasi dan single-point crossover, dengan parameter sebagai berikut:

- Probabilitas crossover : 0,8
- Probabilitas mutasi : 0,01

Parameter PID diperoleh setelah mengulang iterasi hingga 100 generasi.

4. PENGATURAN KECEPATAN MOTOR INDUKSI DENGAN SELF TUNING KONTROLER PID

4.1 Simulasi Sistem

Motor induksi yang digunakan adalah motor induksi tiga fasa, 4 kutub, tipe sangkar bajing dan dimodelkan oleh blok asynchronous machine. Parameter motor induksi dalam simulasi diberikan pada tabel 1.

Tabel 1. Parameter motor induksi

Data Motor Induksi 3 fasa	
Daya (Watt)	250
Tegangan (Volt)	400
Frekuensi (Hz)	50
R_s (Ohm)	41,18
L_{ls} (H)	0,1744
L_m (H)	0,9854
L'_{lr} (H)	0,2616
R'_r (Ohm)	22,37
J (Kg.m ²)	0,0025

Keterangan: R_s : tahanan kumparan stator
 L_{ls} : induktansi kumparan stator
 L_m : induktansi bersama

L'_{lr} : induktansi kumparan rotor
 R'_r : tahanan kumparan rotor
 J : momen inersia

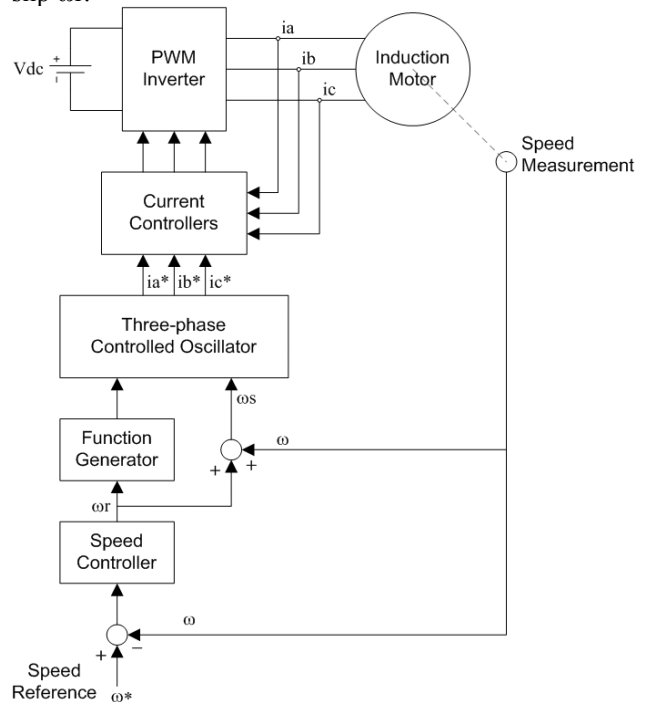
Simulasi diuji dengan kecepatan referensi 100 rad/s pada tiga beban yang berbeda. Besarnya beban ditentukan dengan mengatur besarnya torsi beban (T_L) yaitu untuk $T_L = 0$ N.m, $T_L = 1$ N.m, dan $T_L = 1,5$ N.m.

4.1.1 Pengemudian Motor Induksi Dengan Kecepatan Yang Variabel

Dalam hal ini, digunakan pengemudian motor induksi menggunakan arus stator sinusoidal yang dipengaruhi dan open loop flux control. Torsi motor dikontrol oleh frekuensi rotor (yang juga disebut frekuensi slip). Blok diagram dari pengemudian ini ditunjukkan dalam gambar 5.

Motor induksi mendapat masukan dari current-controlled PWM inverter, yang beroperasi sebagai sumber arus tiga fasa sinusoidal.

Oscillator tiga fasa membangkitkan arus sinusoidal referensi i_a^* , i_b^* , dan i_c^* untuk kontroler arus. Amplitudo dan frekuensi oscillator diatur secara terpisah oleh sinyal I_s (arus stator) dan ω_s (frekuensi stator). Torsi motor diatur oleh frekuensi rotor ω_r (frekuensi slip). Kecepatan motor ω dibandingkan dengan referensi ω^* dan errornya diproses oleh kontroler kecepatan untuk menghasilkan torque command yang merupakan frekuensi rotor ω_r . Frekuensi oscillator ω_s , yang merepresentasikan frekuensi arus stator merupakan hasil penjumlahan kecepatan motor ω dan frekuensi slip ω_r .



Gambar 5. Pengemudian motor induksi dalam operasi kecepatan variabel

Fluksi celah udara diatur dalam suatu open loop oleh function generator yang menghubungkan amplitudo arus I_s dengan frekuensi rotor ω_r berdasarkan persamaan berikut:

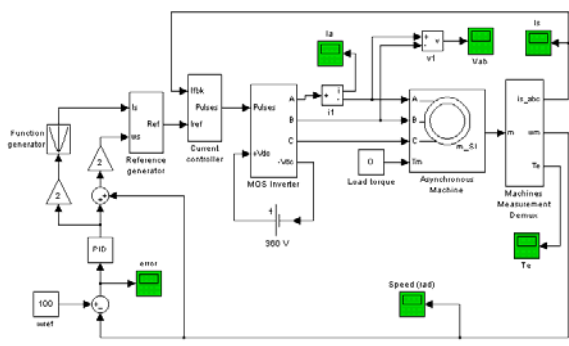
$$I_s = I_m \sqrt{\frac{R_r^2 + (\omega_r L_{lr})^2}{R_r^2 + (\omega_r L_{lr})^2}} \quad (21)$$

dimana I_m = Arus magnetisasi nominal (A)
 R_r = Tahanan rotor (Ω)
 L_{lr} = Induktansi bocor kumparan rotor (H)
 L_{lr} = Induktansi total kumparan rotor (H)

Fungsi dari speed controller adalah untuk mempertahankan kecepatan motor agar sama dengan kecepatan referensi dalam steady-state dan memberikan dinamik yang bagus selama transien.

4.1.2 Simulasi Pengaturan Kecepatan Motor Induksi dengan Kontroler PID Menggunakan MATLAB/SIMULINK dan POWER SYSTEM BLOCKSET

Simulasi dilakukan dengan menggunakan bantuan komputer menggunakan fasilitas *software MATLAB versi 6.1*. Sistem yang diinginkan dimodelkan dalam blok SIMULINK. Gambar 6 menunjukkan skema Simulink untuk model pengaturan kecepatan motor induksi dengan kontroler PID.



Gambar 6. Skema pengaturan kecepatan motor induksi dengan kontroler PID

4.2 Hasil Simulasi dan Analisa

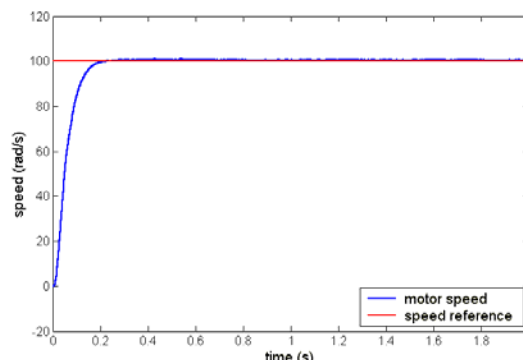
4.2.1 Hasil Simulasi Pengaturan Kecepatan Motor Induksi untuk Torsi Beban = 0 N.m

Dengan metode GA, untuk $T_L = 0$ N.m diperoleh parameter kontroler PID seperti yang terlihat pada tabel 2.

Parameter ini kemudian dimasukkan dalam kontroler PID dalam simulasi model SIMULINK yang dijelaskan pada bab 4.1.2. Respons kecepatan hasil simulasi dapat dilihat pada gambar 7.

Tabel 2. Parameter PID untuk $T_L = 0$ N.m

Generation	K_p	K_i	K_d
1	1.1796	0.11967	0.034183
10	1.6631	0.11967	0.036793
20	1.1796	0.11967	0.020458
30	1.6631	0.11967	0.029835
40	1.1101	0.23938	0.013477
50	1.1151	0.11967	0.0035616
60	1.3285	0.11967	0.013223
70	1.1151	0.11967	0.0093734
80	1.1151	0.11967	0.036128
90	1.1151	0.11967	0.036793
100	1.1151	0.11967	0.036793



Gambar 7. Respons kecepatan motor untuk $T_L=0$ N.m

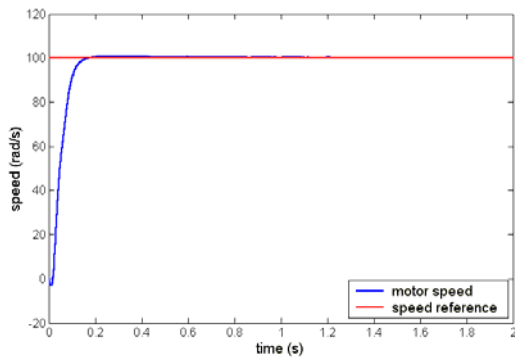
4.2.2 Hasil Simulasi Pengaturan Kecepatan Motor Induksi untuk Torsi Beban = 1 N.m

Dengan metode GA, untuk $T_L = 1$ N.m diperoleh parameter kontroler PID seperti yang terlihat pada tabel 3.

Tabel 3. Parameter PID untuk $T_L = 1$ N.m

Generation	K_p	K_i	K_d
1	0.58586	1.0055	0.0032111
10	0.3165	1.0055	0.0087585
20	0.5263	1.0033	0.0095725
30	0.50107	1.0033	0.0095725
40	0.5263	1.0033	0.0087585
50	0.5263	1.0033	0.0087585
60	0.5263	1.0033	0.0087585
70	0.50107	1.0055	0.0087585
80	0.50107	1.0033	0.0087585
90	0.50107	1.0033	0.0087585
100	0.50107	1.0055	0.0087585

Parameter ini kemudian dimasukkan dalam kontroler PID dalam simulasi model SIMULINK yang dijelaskan pada bab 4.1.2. Respons kecepatan hasil simulasi dapat dilihat pada gambar 8.



Gambar 8. Respons kecepatan motor untuk $T_L=1$ N.m

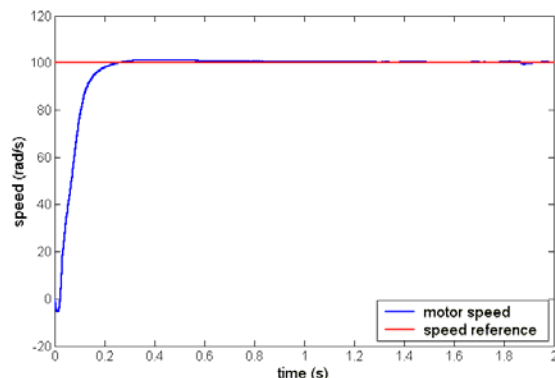
4.2.3 Hasil Simulasi Pengaturan Kecepatan Motor Induksi untuk Torsi Beban = 1,5 N.m

Dengan metode GA, untuk $T_L = 1,5$ N.m diperoleh parameter kontroler PID seperti yang terlihat pada tabel 4.

Tabel 4. Parameter PID untuk $T_L = 1,5$ N.m

Generation	K_p	K_i	K_d
1	0.26611	1.5283	0.0088365
10	0.29538	1.5385	0.016342
20	0.4121	1.5385	0.00035819
30	0.4121	1.5385	0.016342
40	0.4121	1.5283	0.00035819
50	0.4121	1.5385	0.016342
60	0.4121	1.5283	0.00035819
70	0.4121	1.5283	0.016342
80	0.4121	1.5385	0.00035819
90	0.4121	1.5385	0.00035819
100	0.4121	1.5385	0.016342

Parameter ini kemudian dimasukkan dalam kontroler PID dalam simulasi model SIMULINK yang dijelaskan pada bab 4.1.2. Respons kecepatan hasil simulasi dapat dilihat pada gambar 8.



Gambar 9. Respons kecepatan motor untuk $T_L=1,5$ N.m

4.2.4 Perbandingan Respons Kecepatan Antara Tuning Parameter PID Dengan GA Dan Tanpa GA

Dari hasil simulasi diperoleh respons kecepatan motor induksi dengan beban yang berbeda. Untuk melihat unjuk kerja dari tuning parameter kontroler PID dengan GA maka respons kecepataannya dibandingkan dengan dua respons kecepatan untuk sistem yang sama, namun parameter PID-nya ditentukan secara acak. Untuk pembandingan 1, parameter PID yang diambil adalah yang harganya lebih besar dari hasil tuning GA. Sedangkan untuk pembandingan 2, parameter PID yang diambil adalah yang harganya lebih kecil dari hasil tuning GA. Parameter yang dibandingkan adalah: *waktu naik* (t_r), *overshoot maksimum* (M_p), dan *waktu steady-state* (t_s).

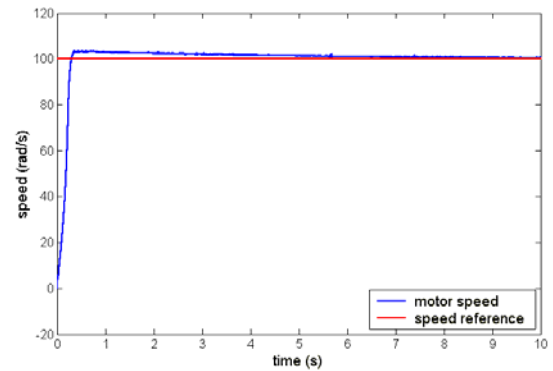
4.2.4.1. Analisis Respons Kecepatan Untuk $T_L = 0$ N.m

Untuk pembandingan pertama, nilai parameter kontroler PID yang diambil adalah sebagai berikut:

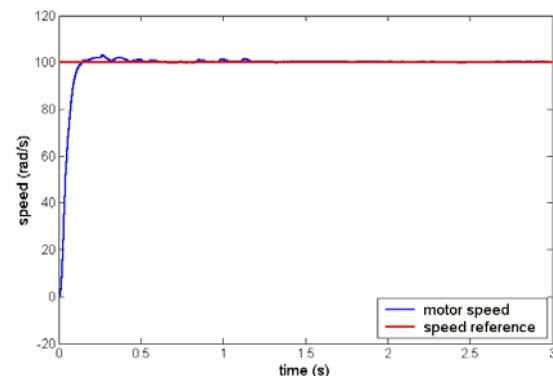
Tabel 5. Parameter PID pembandingan untuk $T_L = 0$ N.m

	Pembandingan 1	Pembandingan 2
K_p	1,6	0,6
K_i	0,3	0,05
K_d	0,04	0,01

Respons kecepatan motor induksi dari simulasi untuk pembandingan 1 dapat dilihat pada gambar 10, sedangkan untuk pembandingan kedua pada gambar 11.



Gambar 10. Respons kecepatan motor dengan parameter PID pembandingan 1 ($T_L=0$)



Gambar 11. Respons kecepatan motor dengan parameter PID pembandingan 2 ($T_L=0$)

Dari hasil simulasi diperoleh parameter respons kecepatan motor induksi seperti yang terlihat di tabel 6.

Tabel 6. Hasil simulasi untuk $T_L = 0$ N.m

		t_r	M_p	t_s
Dengan GA [gambar (6)]		0.2237	1.0278	0.706
Tanpa GA	Pembanding 1 [gambar (9)]	0.304	3.4207	8.6
	Pembanding 2 [gambar (10)]	0.142	3.1492	1.2

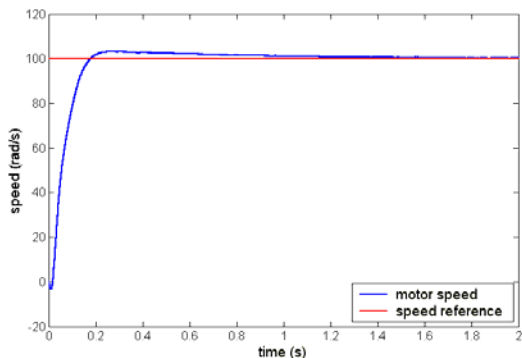
4.2.4.2. Analisis Respons Kecepatan Untuk $T_L = 1$ N.m

Untuk pembanding pertama, nilai parameter kontroler PID yang diambil adalah sebagai berikut:

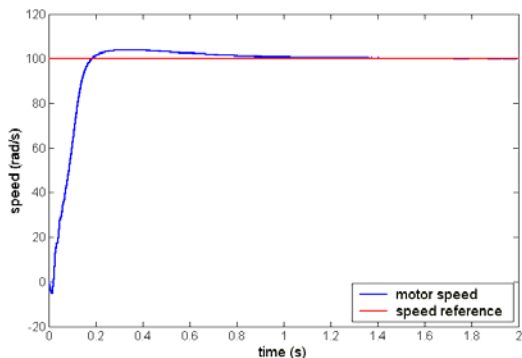
Tabel 7. Parameter PID pembanding untuk $T_L = 1$ N.m

	Pembanding 1	Pembanding 2
Kp	0,7	0,2
Ki	1,2	0,8
Kd	0,01	0,006

Respons kecepatan motor induksi dari simulasi untuk pembanding pertama dapat dilihat pada gambar 12, sedangkan untuk pembanding kedua pada gambar 13.



Gambar 12. Respons kecepatan motor dengan parameter PID pembanding 1 ($T_L=1$)



Gambar 13. Respons kecepatan motor dengan parameter PID pembanding 2 ($T_L=1$)

Dari hasil simulasi pengaturan kecepatan motor induksi dengan kontroler PID untuk kondisi

$T_L = 1$ N.m didapat hasil seperti yang terlihat di tabel 8.

Tabel 8. Hasil simulasi untuk $T_L = 1$ N.m

		t_r	M_p	t_s
Dengan GA [gambar (7)]		0.1730	0.8723	0.443
Tanpa GA	Pembanding 1 [gambar (11)]	0.1786	3.2415	1.477
	Pembanding 2 [gambar (12)]	0.1848	3.8970	1.037

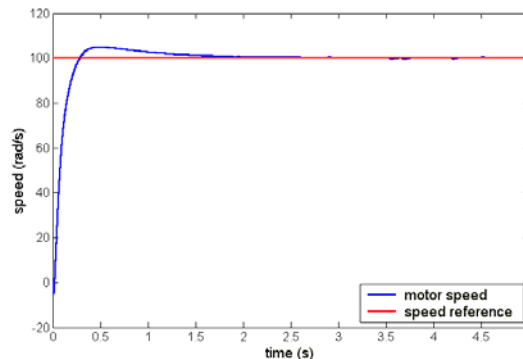
4.2.4.3. Analisis Respons Kecepatan Untuk $T_L = 1,5$ N.m

Untuk pembanding pertama, nilai parameter kontroler PID yang diambil adalah sebagai berikut:

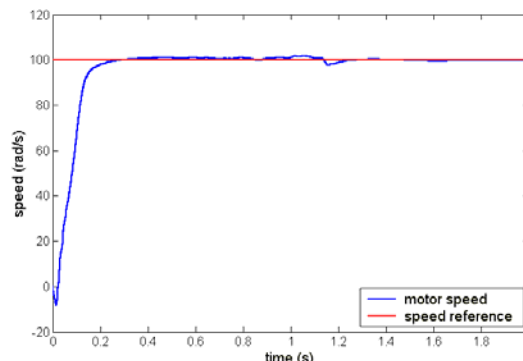
Tabel 9. Parameter PID pembanding untuk $T_L = 1,5$ N.m

	Pembanding 1	Pembanding 2
Kp	0,8	0,2
Ki	1,9	1,3
Kd	0,03	0,005

Respons kecepatan motor induksi dari simulasi untuk pembanding 1 dapat dilihat pada gambar 14, sedangkan untuk pembanding 2 pada gambar 15.



Gambar 14. Respons kecepatan motor dengan parameter PID pembanding 1 ($T_L=1,5$)



Gambar 15. Respons kecepatan motor dengan parameter PID pembanding 2 ($T_L=1,5$)

Dari hasil simulasi pengaturan kecepatan motor induksi dengan kontroler PID untuk kondisi $T_L = 1,5$ N.m didapat hasil seperti yang terlihat di tabel 10.

Tabel 10. Hasil simulasi untuk $T_L = 1,5 \text{ N.m}$

		tr	M_p	ts
Dengan GA [gambar (8)]		0.2582	1.2723	0.9
Tanpa GA	Pembandingan 1 [gambar (13)]	0.288	4.9937	3.52
	Pembandingan 2 [gambar (14)]	0.2903	1.8086	1.232

5. PENUTUP

5.1 Kesimpulan

Berdasarkan simulasi yang dilakukan dan hasil yang diperoleh maka dari penelitian ini dapat disimpulkan sebagai berikut:

1. Dengan menggunakan GA, parameter PID dapat diperoleh dengan cepat dengan hasil optimal.
2. Untuk torsi beban = 0 N.m, dengan GA diperoleh parameter kontroler $K_p = 1.1151$, $K_i = 0.11967$ dan $K_d = 0.036793$. Dengan parameter kontroler PID ini, diperoleh karakteristik respons kecepatan motor induksi seperti yang terlihat di gambar 6 sebagai berikut: $tr = 0.2237$, $M_p = 1.0278$, $ts = 0.706$.
3. Untuk torsi beban = 1 N.m, dengan GA diperoleh parameter kontroler $K_p = 0.50107$, $K_i = 1.0055$ dan $K_d = 0.0087585$. Dengan parameter kontroler PID ini, diperoleh karakteristik respons kecepatan motor induksi seperti yang terlihat di gambar 7 sebagai berikut: $tr = 0.1730$, $M_p = 0.8723$, $ts = 0.443$.
4. Untuk torsi beban = 1,5 N.m, dengan GA diperoleh parameter kontroler $K_p = 0.4121$, $K_i = 1.5385$ dan $K_d = 0.016342$. Dengan parameter kontroler PID ini, diperoleh karakteristik respons kecepatan motor induksi seperti yang terlihat di gambar 8 sebagai berikut: $tr = 0.2582$, $M_p = 1.2723$, $ts = 0.9$.

5.2 Saran

Dari penelitian yang dilakukan, diusulkan beberapa saran:

1. Penelitian ini diharapkan dapat dijadikan langkah awal untuk penelitian yang lebih mendalam tentang *Genetic Algorithms* dan penerapannya dalam sistem tenaga.
2. Untuk pengembangan lebih lanjut, penulis sarankan agar metode tuning kontroler PID dilakukan secara on-line.
3. Penelitian selanjutnya menggunakan program AI lainnya sehingga akan didapatkan perbandingan antara tiap program dan didapatkan mana yang lebih baik dalam tuning parameter kontroler PID.

PUSTAKA

- Chapman, S. J., **Electric Machinery Fundamentals**. Third Edition, Singapore: McGraw-Hill, 1999.
- Davis, L., **Handbook Of Genetic Algorithms**. New York: Van Nostrand Reinhold, 1991.
- Hydro-Quebec and TEQSIM International Inc., **Power System Blockset User's Guide**. TEQSIM International Inc., 1998.
- Michalewicz, Z., **Genetic Algorithms + Data Structures = Evolution Programs**. Third Edition, New York: Springer, 1996.
- Ogata, K., **Modern Control Engineering**. Third Edition, Upper Saddle River, NJ: Prentice-Hall, Inc., 1997.
- Ogata, K., **Teknik Kontrol Otomatik Jilid 1**. Edisi Kedua, Jakarta: Erlangga, 1997.
- Vithayathil, J., **Power Electronic Principles and Application**. McGraw-Hill, 1995.
- Webb, J., and Greshock, K., **Industrial Control Electronics**. Singapore: Maxwell Macmillan, 1992.
- Winter, G., Périaux, J., Galán, M., and Cuesta, P., **Genetic Algorithms in Engineering and Computer Science**. John Wiley & Sons, 1995.