

Perencanaan Optimal Sistem Kontrol AVR (*Automatic Voltage Regulator*) Untuk Memperbaiki Kestabilan Tegangan Dengan Menggunakan Algoritma Genetik

Makalah Tugas Akhir

Disusun Oleh :

Endriyanto NW
L2F301437

Jurusan Teknik Elektro Fakultas Teknik
Universitas Diponegoro Semarang

Abstrak

Automatic Voltage Regulator (AVR) memegang peranan penting pada proses pembentukan profil tegangan terminal suatu generator. Perubahan kondisi beban sebagai perilaku dinamis sistem akan menyebabkan perubahan arus yang mengalir dalam sistem generator yang menyebabkan perubahan tegangan armatur dan terminal generator tersebut. Banyak metode yang digunakan untuk mengontrol stabilisasi profil tegangan terminal generator seperti PID, lag/lead Compesator, lag/lead compensator with stabilizer, Pole Placement dll.

Tugas akhir ini membuat suatu mekanisme kontrol profil tegangan terminal generator dengan menggunakan metode lag/lead compensator with stabilizer. Metode ini akan diimplementasikan bersama dalam model sistem AVR. Model ini sistem kontrol ini sangat murah karena tidak dibutuhkan suatu mekanisme kontrol yang cenderung kompleks seperti PID. Mekanisme kontrol ini hanya membutuhkan sebuah sensor tegangan dan stabiliser komponen lain merupakan komponen murni sistem AVR pada kondisi ikaI terbuka. Permasalahan yang paling utama adalah menentukan besar penguatan (Gain Factor) masing-masing sistem AVR. Pemilihan GF yang tepat akan memberikan unjuk kerja yang optimal begitu juga sebaliknya jika tidak tepat maka stabilisasi profil tegangan terminal tidak akan tercapai.

Pemilahan (Tunning) GF dalam tugas akhir ini menggunakan suatu metode pencarian acak terbimbing (Guided Random Search) dengan fitur seleksi alam dan evolusi genetik, atau lebih dikenal dengan nama algoritma genetik. Metode ini ternyata mampu memberikan proses tunning parameter adaptif AVR dengan kualifikasi yang sangat baik. Perfomansi algoritma genetik dapat ditentukan dengan mengetahui kualitas stabilisasi profil tegangan yang dihasilkan oleh AVR yang di re-design menggunakan algoritma genetik. Selama pengujian algoritma genetik mampu memberikan penyelesaian optimal model AVR yang handal dan berkualias seperti yang direncanakan.

1. Latar Belakang

Mutu energi listrik untuk beberapa parameter seperti fluktuasi frekuensi, fluktuasi tegangan, flicker, harmonisa dan kontinyuitas pelayanan energi. Dari kesemua parameter tersebut fluktuasi tegangan yang paling sering mendapatkan perhatian para ahli dan para operator sistem tenaga. Beberapa negara maju seperti Perancis, Jepang, USA memberikan perhatian dan penanganan khusus dalam permasalahan kestabilan tegangan. Ketidakstabilan tegangan akan menyebabkan ketidakstabilan sistem tenaga secara keseluruhan, terutama sekuritas sistem, kualitas dan kemampuan transfer daya dari

pembangkit ke konsumen, kondisi terparah terjadinya mekanisme *load shedding* ataupun *brown out*. Ada beberapa faktor yang menyebabkan ketidakstabilan tegangan (*voltage collapse*) antara lain unjuk kerja AVR, kenaikan pembebanan saluran transmisi, kendala pengaturan daya reaktif, dinamika OLTC (*on load tap changer*) trafo dan karakteristik beban.

Kestabilan tegangan sistem praktis ditentukan oleh kestabilan sistem regulasi tegangan yang dilakukan oleh sistem eksitasi yang terdapat dalam generator sinkron dan beberapa rangkaian pengontrol lain yang terintegrasi dalam suatu sistem yang disebut AVR. AVR ini memiliki

tugas menjaga profil tegangan terminal pada suatu titik operasi tertentu seperti 220, 380, 13.8kV dll. Fungsi AVR yang lain berkaitan dengan aksi kontrol regulasi daya reaktif dan pengaturan osilasi rotor jika terjadi gangguan.

Model AVR yang digunakan dalam tugas akhir ini adalah model AVR dengan mekanisme kontrol Lag/Lead Compensator dengan menggunakan stabiliser. Komponen-komponen utama dalam model sistem ini terdiri dari amplifier, exciter, model generator, sensor dan controller. Controller yang paling banyak digunakan dalam sistem AVR adalah model kontroler PID, Fuzzy, Adaptif dll. Model AVR yang akan dikembangkan sedemikian rupa mengeliminasi controller sehingga komponen yang dibutuhkan antara lain model AVR utama ditambah dengan stabiliser. Permasalahan yang akan timbul seberapa baik kualitas model AVR yang dihasilkan, dengan melakukan tuning parameter adaptif AVR maka diperoleh semua objektif yang diinginkan.

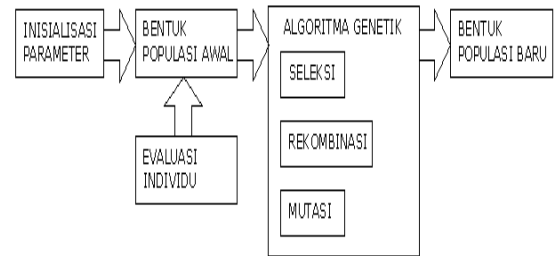
2. Tujuan

Tujuan tugas akhir adalah membentuk sebuah model AVR dengan menggunakan mekanisme Lag/Lead Compensator dengan Stabiliser yang dioptimasi menggunakan algoritma genetik.

3. Dasar Teori

3.1 Algoritma Genetik

Algoritma genetik pertama kali diperkenalkan oleh John Holland pada tahun 1975 di Universitas Michigan untuk keperluan pengembangan pemrograman komputer untuk aplikasi sistem kecerdasan komputer pada penyelesaian permasalahan-permasalahan non-linier. Perkembangan algoritma genetik pada bidang rekayasa dipelopori oleh prof. David E Goldberg, dalam studi analisa optimasi tata letak stasiun kompresor dalam penyaluran gas elpiji. Algoritma genetik didefinisikan sebagai suatu teknik pencarian yang berbasis pada mekanisme evolusi genetik dan seleksi alam. Siklus dasar algoritma genetik dapat dilihat pada gambar 1 berikut.



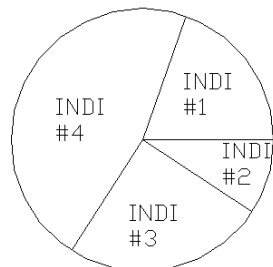
Gambar 1. Siklus Dasar Algoritma Genetik

Siklus algoritma genetik dimulai dengan menginisialisasi semua parameter algoritma genetik seperti ukuran populasi, panjang kromosom dll, kemudian dibentuk populasi awal, populasi awal dibentuk secara acak sehingga pada populasi awal terbentuk individu-individu dengan kromosom acak sehingga kualitas individu pada populasi awal cenderung rendah. Setelah populasi awal terbentuk maka dilakukan proses evaluasi, proses ini bertujuan untuk memberikan penilaian setiap individu yang dihitung berdasarkan persamaan objektif yang dimiliki. Setelah kualitas individu diketahui maka dilakukan ranking terhadap individu terbaik hingga terburuk dan dilakukan pemilihan induk. Proses ini sudah masuk pada proses algoritma genetik. Setelah terpilih induk maka dilakukan rekombinasi dan dilanjutkan dengan mutasi. Setelah satu proses selesai maka akan diperoleh sekumpulan individu baru yang kemudian disebut sebagai generasi baru. Proses ini berlanjut hingga ke suatu generasi yang kemudian akan dinyatakan sebagai generasi saturasi. Pada generasi ini perbedaan kualitas antar individu sudah tidak begitu signifikan.

3.1.1 Mekanisme Seleksi

Seleksi adalah suatu operator algoritma genetik yang berfungsi memilih individu-individu yang akan dijadikan induk pada proses rekombinasi. Metode seleksi yang paling banyak dipergunakan adalah metode *Roulette Wheel Selection (RWS)*. Metode ini sama dengan metode yang dipergunakan pada permainan rolet, dimana setiap angka dinyatakan dalam suatu luasan sektor dalam lingkaran. Terpilingnya suatu sektor dalam satu putaran ditentukan oleh luasan yang dimiliki, semakin luas maka akan semakin sering terpilih. Pada permainan rolet semua sektor memiliki luasan yang sama sehingga kemungkinan yang dimiliki setiap luasan sama. Perbedaan yang mendasar antara permainan rolet

dengan metode RWS adalah pada RWS luasan sektor pada lingkaran rolet sebanding dengan kualitas masing-masing individu. Semakin baik kualitas individu maka semakin luas sektor yang dimiliki olehnya. Gambaran metode ini diperlihatkan pada gambar 2 berikut:



Gambar 2. Lingkaran Rolet RWS

Gambar 2 memperlihatkan bagaimana 4 individu direpresentasikan di dalam suatu roda rolet. Data ke-4 individu tersebut adalah sebagai tabel berikut:

Tabel 1. Statistik kualitas individu (skala 1)

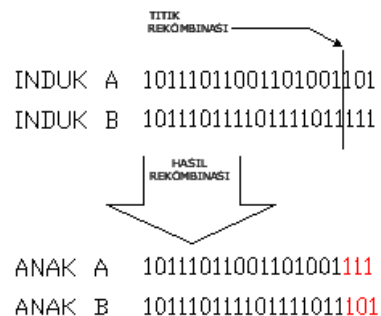
N	Individu	Kualitas
1	Indi #1	0.20
2	Indi #2	0.08
3	Indi #3	0.30
4	Indi #4	0.42

Jika dilihat pada roda rolet maka individu nomer 4 memiliki kemungkinan terpilih terbesar karena memiliki kualitas 0.42 sedangkan individu nomer 2 memiliki kemungkinan terkecil karena memiliki kualitas 0.08. Besar nilai kualitas terhitung menggunakan persamaan objektif suatu permasalahan yang akan dipecahkan. Mekanisme pemilihan dilakukan dengan memutar roda rolet (gambar 2) secara acak kemudian ditunggu hingga roda itu berhenti pada suatu sektor tertentu. Walaupun individu nomer 2 memiliki luasan sektor paling kecil tidak berarti individu ini tidak pernah terpilih pada suatu proses pemilihan, ingat piringan rolet dapat berhenti di semua sektor. Mekanisme rolet memiliki kesamaan dengan seleksi alam siapa yang terkuat pasti memiliki kemampuan terpilih yang terbesar.

3.1.2 Mekanisme Rekombinasi

Rekombinasi adalah suatu proses pertukaran struktur kromosom antara dua induk yang terpilih pada proses seleksi dengan tujuan untuk

menciptakan keragaman materi genetik individu pada generasi baru. Individu hasil dari proses rekombinasi akan tetap mewarisi sifat-sifat yang dimiliki induknya hal ini penting untuk menjaga kualitas individu pada generasi berikutnya. Proses rekombinasi diperlihatkan pada gambar 3 berikut:



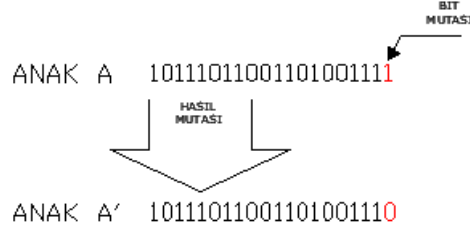
Gambar 3. Proses Rekombinasi

Proses rekombinasi dilakukan dengan cara memilih dua induk dengan kualitas yang baik setelah itu dilakukan proses ekstraksi kromosom setiap induk. Titik rekombinasi ditentukan secara acak, setelah ditentukan dimana titik rekombinasi maka dilakukan pertukaran bit-bit kromosom disebelahkan titik kromosom sehingga terbentuk keturunan yaitu anak A dan B. Kromosom anak sebagian besar masih mewarisi kromosom induk tetapi sebagian lagi sudah terjadi pertukaran materi genetik anatar kromosom. Proses rekombinasi memiliki nilai kemungkinan yang besar dalam satu siklus algoritma genetik karena tujuan utamanya adalah membentuk keragaman individu, semakin tinggi probabilitas rekombinasi maka semakin cepat keragaman terbentuk.

3.1.3 Mekanisme Mutasi

Mutasi adalah operator algoritma genetik yang berguna untuk membentuk individu-individu dengan fitur superior atau memiliki kualitas diatas rata-rata. Selain itu mutasi dipergunakan untuk mengembalikan kerusakan materi genetik akibat proses rekombinasi. Proses rekombinasi terjadi pada level bit kromosom, proses ini akan diimplmentasikan kesemua bit yang terdapat dalam suatu kromosom. Metode mutasi yang paling banyak digunakan adalah metode *flip-bit*, metode ini memiliki algoritma yang sangat sederhana yaitu dengan merubah 1 menjadi 0 atau merubah 0 menjadi 1. Proses terjadinya mutasi ditentukan besar probabilitas mutasi yang

dipergunakan semakin besar semakin sering. Proses mutasi itu sendiri diperlihatkan pada gambar 4 berikut:

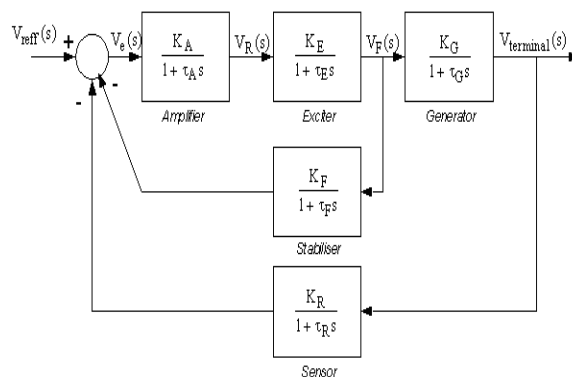


Gambar 4. Proses Mutasi Kromosom

Pada gambar 3 diatas bit yang mengalami mutasi akan dirubah, suatu contoh pada gambar diatas bit yang bernilai 1 (merah) akan dirubah menjadi 0 (merah). Proses mutasi ini tidak boleh sering dilakukan karena perubahan kualitas individu akan terbentuk secara dramatis baik yang menuju baik atau yang buruk, hal ini akan menuju kesuatu fenomena proses pencarian yang bersifat acak dan keluar dari kaidah-kaidah algoritma genetik yang mengimplemntasikan proses pencarian acak yang terbimbing (*Guided random Search*)

3.2 Automatic Voltage Regulator

AVR merupakan elemen penting untuk memebentuk profil tegangan terminal generator yang stabil. AVR dengan model Lag/Lead Compensator yang sertai dengan stabiliser mampu membentuk model AVR yang sederhana namun handal. Model AVR yang dipergunakan seperti pada gambar berikut, yang terdiri dari ampliflier, exciter, generator, sensor dan stabiliser.



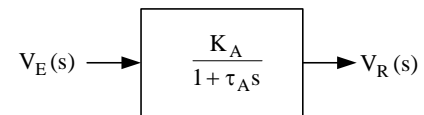
Gambar 5. Model AVR dengan menggunakan stabiliser

3.2.1. Model Amplifier

Dalam sistem eksitasi, *amplifier* dapat berupa *magnetic amplifier*, *rotating amplifier* atau sistem *amplifier* elektronik. Model *amplifier* direpresentasikan dalam sebuah *sistem orde satu* dengan sebuah faktor penguatan dan konstanta waktu. Fungsi alih *amplifier* dapat dilihat pada (1.1).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_A}{1 + \tau_A s} \quad (1.1)$$

Nilai K_A akan bervariasi mulai dari 10 hingga 400, sedangkan nilai konstanta waktu sangat kecil yaitu mulai dari 0.02 hingga 1 detik. Diagram blok *amplifier* dapat dilihat pada gambar 6.



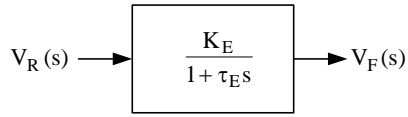
Gambar 6. Representasi Amplifier

3.2.2. Model Exciter

Exciter banyak sekali macamnya, namun dalam era modern dimana perkembangan teknologi *solid state* maju begitu pesat, maka sistem eksitasi menggunakan sumber tegangan AC yang terkontrol melalui SCR sudah banyak dijumpai. Kecepatan switching merupakan aspek yang sangat menguntungkan dari model *exciter* elektronik. Tegangan keluaran sistem *exciter* adalah non-linier dan merupakan fungsi dari tegangan medan disebabkan efek saturasi inti magnet, sehingga dapat dilihat bahwa hubungan antara tegangan terminal dan tegangan medan *exciter* adalah sangat kompleks. Banyak sekali model yang tersedia dengan berbagai level keakuratan tersedia dalam publikasi IEEE. Model *exciter* yang modern dapat didekati dengan mengeliminasi efek saturasi tetapi dengan sangat memperhatikan konstanta waktu. Maka *exciter* dapat dimodelkan dengan sebuah sistem orde satu dimana dengan faktor penguatan K_E dan konstanta waktu T_E . Fungsi alih sistem *exciter* dapat dilihat pada (1.2).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_E}{1 + \tau_E s} \quad (1.2)$$

diagram blok sistem *exciter* dapat dilihat pada gambar 7



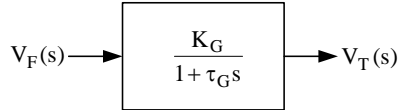
Gambar 7. Representasi Sistem Exciter

3.2.3. Model Generator

Emf yang dibangkitkan oleh generator sinkron merupakan fungsi dari proses magnetisasi, sedangkan tegangan terminal tergantung dari beban. Untuk model yang linier sebuah generator dapat didekati dengan sebuah sistem orde satu, dimana memiliki faktor penguatan K_G dan konstanta waktu T_G . Model generator dinyatakan pada (1.3).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_G}{1 + \tau_G s} \quad (1.3)$$

Konstanta K_G tergantung dari beban dan memiliki nilai antara 0.7 hingga 1.0, sedangkan konstanta waktu T_G memiliki nilai mulai dari 1.0 hingga 2.0 detik. Diagram blok generator dapat dilihat pada gambar 8



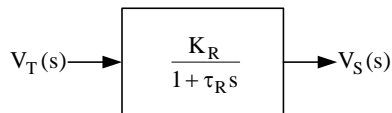
Gambar 8. Representasi Generator

3.2.4. Model Sensor

Tegangan terminal generator disensor menggunakan sebuah trafo tegangan, dan kemudian disearahkan melalui penyearah jembatan. Sensor ini secara sederhana dimodelkan dalam sistem orde satu yaitu seperti pada (1.4).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_R}{1 + \tau_R s} \quad (1.4)$$

dimana K_R memiliki nilai 10 hingga 400, dan T_R memiliki nilai 0.01 hingga 0.06. Blok diagram sistem sensor elektronik adalah gambar 9.



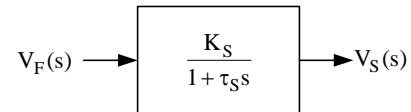
Gambar 9. Representasi Sensor

3.2.5. Stabiliser

Fungsi stabiliser adalah untuk menambahkan sebuah zero pada fungsi alih sistem AVR *uncompensated* agar diperoleh dampak meningkatnya kestabilan relatif. Sebuah stabiliser pada dasarnya merupakan sebuah kompensator lag/lead. Fungsi alih sistem stabiliser dapat dilihat pada (1.5).

$$\frac{V_R(s)}{V_E(s)} = \frac{K_S}{1 + \tau_S s} \quad (1.5)$$

blok diagram model stabiliser dapat dibuat seperti gambar 10.



Gambar 10. Representasi Stabiliser

4. Fungsi alih sistem AVR

Fungsi alih AVR dapat diturunkan dengan mengetahui persamaan perbandingan antara tegangan terminal dengan tegangan referensi.

$$\begin{aligned} \frac{V_T(s)}{V_{Ref}(s)} &= T(s) \\ &= \frac{G(s).E(s).A(s).V_e(s)}{V_e(s) + V_S^{err}(s) + V_R^{err}(s)} \\ &= \frac{G(s).E(s).A(s).V_e(s)}{V_e(s) + S(s).E(s).A(s).V_e(s) + R(s).G(s).E(s).A(s).V_e(s)} \\ &= \frac{G(s).E(s).A(s)}{1 + S(s).E(s).A(s) + R(s).G(s).E(s).A(s)} \end{aligned}$$

dimana,

- A(s) : Fungsi alih *amplifier*
- E(s) : Fungsi alih *exciter*
- G(s) : Fungsi alih generator
- S(s) : Fungsi alih stabiliser
- R(s) : Fungsi alih sensor

Fungsi $F(s)=G(s).E(s).A(s)$, disebut fungsi alih sistem AVR secara *uncompensated*, fungsi $F(s)=1+S(s).E(s).A(s)+R(s).G(s).E(s).A(s)=0$, disebut sebagai persamaan karakteristik sistem AVR, sistem AVR akan stabil jika akar-akar

persamaan karakteristik kesemuanya terletak di sebelah kiri sumbu imajiner bidang kompleks.

4.1 Metode Root Locus

Metode root locus dikembangkan W.R Evans, yang digunakan untuk menentukan nilai eigen sistem kontrol *compensated* dari persamaan *open loop* fungsi alih untuk semua nilai faktor penguatan persamaan karakteristik. Metode *root locus* pada hakikatnya merupakan metode pemetaan semua akar-akar persamaan karakteristik untuk suatu perubahan nilai faktor penguatan mulai dari nol hingga *infinity*. Dengan adanya pemetaan ini diharapkan para *engineer* dapat menentukan besar penguatan sistem agar dapat diperoleh objektif sistem yang direncanakan. Jika obyektif tidak dapat tercapai dengan hanya menggunakan fungsi alih sistem, maka diperlukan aksi kontrol tambahan, yang kemudian dipetakan kembali semua akar persamaan karakteristik yang terbentuk setelah penambahan aksi kontrol tersebut. Tinjau suatu bentuk persamaan fungsi alih sistem kontrol dimana fungsi alih sistem untai terbuka seperti pada (1.7)

$$KG(s)H(s) = \frac{K(s+z_1)(s+z_2)\dots(s+z_m)}{(s+p_1)(s+p_2)\dots(s+p_n)} \quad (1.7)$$

dimana m menyatakan banyaknya zero dan n menyatakan banyaknya kutub. Jika $n > m$ maka akan terdapat $(n-m)$ zero. Untuk persamaan karakteristik sistem kontrol untai tertutup dapat ditulis :

$$-K = \frac{(s+p_1)(s+p_2)\dots(s+p_n)}{(s+z_1)(s+z_2)\dots(s+z_m)} \quad (1.8)$$

jika kita petakan (1.8) ke dalam bidang root locus mulai dari harga $K > 0$ hingga K sama dengan *infinity* harus memenuhi persyaratan berikut :

1. $K = \frac{\text{Perkalian Vektor Kutub}}{\text{Perkalian Vektor Zero}}$
2. $\sum \text{sudut zero GH}(s) - \sum \text{sudut kutub GH}(s) = r * 180^\circ; r = \pm 1, \pm 3, \pm 5, \dots$

4.2 Steady-State Error

Kestabilan suatu sistem kontrol dinilai berdasarkan beberapa kriteria yaitu kriteria kestabilan transien dan kriteria kestabilan tunak. Kestabilan transien dimaksudkan untuk mengamati kelakuan sistem untuk suatu perubahan nilai referensi yang diberikan,

sedangkan kestabilan tunak dimaksudkan untuk mengetahui besar kesalahan yang tercapai untuk suatu durasi waktu yang tak terbatas. Suatu sistem kontrol akan memiliki kondisi kesalahan seperti (1.9).

$$E(s) = R(s) - H(s)C(s) = \frac{R(s)}{1 + KG(s)H(s)} \quad (1.9)$$

Besar kesalahan dalam keadaan *steady state* dapat didekati berdasarkan teorema nilai akhir (SSE) yaitu :

$$e_{ss} = \lim_{s \rightarrow 0} \frac{sR(s)}{1 + KG(s)H(s)} \quad (1.10)$$

Kondisi masukan sistem kontrol dimungkinkan dalam tiga masukan, yaitu :

1. Masukan *step*, maka e_{ss} nya adalah :

$$e_{ss} = \frac{1}{1 + \lim_{s \rightarrow 0} KG(s)H(s)} \quad (1.11)$$

2. Masukan *Ramp*, maka e_{ss} nya adalah

$$e_{ss} = \frac{1}{1 + \lim_{s \rightarrow 0} sKG(s)H(s)} \quad (1.12)$$

3. Masukan *Parabolik*, maka e_{ss} nya adalah :

$$e_{ss} = \frac{1}{1 + \lim_{s \rightarrow 0} s^2 KG(s)H(s)} \quad (1.13)$$

4.3 Kriteria Unjuk Kerja

Unjuk kerja dinamis sistem kontrol memiliki karakteristik inheren pada kawasan waktu. Standar pengujian unjuk kerja sistem kontrol dilakukan menggunakan model sistem orde dua dan diberi masukan unit step. Jika keluaran sistem orde dua dapat diketahui maka secara matematis kriteria unjuk kerja sistem dapat ditentukan. Fungsi alih sistem orde dua standar seperti pada (1.14).

$$G(s) = \frac{\omega_n s}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (1.14)$$

dimana,

ω_n : Frekuensi Alami (rad/sec)

ζ : Koefisien Redaman

Fungsi alih $G(s)$ diatas jika transformasikan ke dalam kawasan waktu untuk masukan *unit step* diperoleh :

$$c(t) = 1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin \left(\sqrt{1-\zeta^2} \omega_n t + \tan^{-1} \left(\frac{\sqrt{1-\zeta^2}}{\zeta} \right) \right) \quad (1.15)$$

sesuai (1.15), *peak time* dirumuskan :

$$t_p = \frac{\pi}{\omega_n \sqrt{1-\zeta^2}} \quad (1.16)$$

percent overshoot (P.O) dirumuskan :

$$P.O = \frac{e^{-\zeta\pi}}{\sqrt{1-\zeta^2}} \times 100\% \quad (1.17)$$

rise time dirumuskan :

$$t_r = t_{|c(t)=0.9} - t_{|c(t)=0.1} \quad (1.18)$$

settling time dirumuskan :

$$t_s = \frac{4}{\zeta\omega_n} \quad (1.19)$$

5. Model Genetik AVR

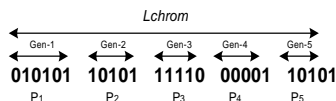
Model sistem AVR yang akan dioptimasi melalui metode algoritma genetik terlebih dahulu diformat ke dalam mekanisme komputasi non-parametrik, yaitu dengan cara mengkodekan parameter-parameter optimal ke dalam format biner yang kemudian tersusun secara sistematis didalam kromosom. Parameter-parameter optimal AVR yang akan dicari meliputi *Gain Amplifier*, *Gain Exciter*, *Gain Generator*, *Gain Stabiliser* dan *Gain Sensor*. Setiap *Gain* komponen akan dimodelkan ke dalam bentuk gen, yang tersusun n bit biner, sedangkan kromosom tersusun atas gen-gen yang mewakili setiap parameter. Jadi panjang kromosom dapat dihitung seperti pada (1.20).

$$L_{chrom} = m * n \text{ bit} \quad (1.20)$$

dimana,

- L_{chrom} : Panjang kromosom
- m : Jumlah gen
- n : Jumlah bit setiap gen.

Contoh, jika kita memiliki 5 buah parameter yang akan dicari selesaian optimalnya dan setiap parameter dikodekan dalam 5 bit biner, maka jumlah gen yang dibutuhkan adalah 5 yakni sama dengan jumlah parameter. Gen-gen tersebut terletak didalam kromosom yang tersusun berjajar seperti pada gambar 11.



Gambar 11. Model Kromosom Biner

Dalam model AVR gen-gen tersebut tidak lain merepresentasikan *Gain Amplifier*, *Gain Exciter*, *Gain Generator*, *Gain Stabiliser* dan *Gain Sensor*. Tingkat kepresisian atau level kuantitasi setiap gain komponen ditentukan berdasarkan (3.4).

$$QL_n = \frac{G_{max}^n - G_{min}^n}{2^{LGen(n)} - 1} \quad (1.21)$$

dimana,

- QL = *Quantitation Level* atau Derajat Kepresisian
- G_{max} = Batas maksimum Gain Komponen
- G_{min} = Batas minimum Gain Komponen
- $LGen$ = Panjang gen (bit)
- n = Jumlah gen

Untuk mendapatkan nilai aktual setiap gain komponen untuk suatu kode gen tertentu dapat dihitung seperti pada (3.5).

$$G_{actual} = \text{decode}(G_n) \times QL_n + G_{min}^n \quad (1.22)$$

dimana,

- G_{actual} : Nilai gain komponen yang sesungguhnya (*True Value*)
- $\text{Decode}(G_n)$: Skalar hasil pengubahan kode biner ke kode desimal
- QL : *Quantitation Level*

Contoh, jika kita anggap parameter kesatu atau gen kesatu mewakili *gain* amplifier dengan batas minimum 1.0 dan maksimum 10.0 dan memiliki kode biner 00111, nilai aktual yang diwakili kode ini dapat dihitung sebagai berikut :

$$\begin{aligned} QL_{Amp} &= \frac{10.0 - 1.0}{2^5 - 1} \\ &= \frac{9.0}{31} \\ &= 0.29 \end{aligned}$$

besar nilai konversi kode dihitung dengan cara berikut :

$$\begin{aligned} \text{decode}(00111) &= 0 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 \\ &\quad + 1 * 2^0 \\ &= 0 + 0 + 0 + 4 + 2 + 1 \end{aligned}$$

$$= 7$$

maka besar nilai aktual gain amplifier dapat dihitung dengan cara berikut :

$$\begin{aligned} G_{Actual}^{Amp} &= \text{decode}(00111).QL_{Amp} + 1.0 \\ &= 7.0.29 + 1.0 \\ &= 3.03 \end{aligned}$$

Untuk mengetahui luas ruang pencarian solusi pemodelan AVR dapat mudah kita tentukan seperti pada (3.6).

$$SS = 2^{L_{chrom}} - 1 \quad (1.23)$$

dimana,

SS : Ruang Pencarian (*Search Space*)

Contoh, jika untuk memodelkan AVR yang optimal diperlukan 5 parameter gain komponen dan setiap gain komponen dikodekan dalam 5 bit biner, maka kita akan memiliki model sebanyak $2^{25}-1$ buah, SS tersebut sangat tidak mungkin dicari menggunakan metode *trial and error* untuk mendapatkan solusi yang paling optimal.

5.1 Fungsi Objektif dan Fungsi *Fitness*

Fungsi objektif digunakan untuk mengetahui atau mengukur secara kualitatif seberapa baik solusi yang dihasilkan atau model yang terbentuk berdasarkan tujuan perencanaan sedangkan fungsi *fitness* digunakan untuk mengetahui secara kualitatif solusi yang dihasilkan berdasarkan fungsi objektif dan pelanggaran kendala (*Constraints Violance*). Dalam algoritma genetik kedua fungsi ini memegang peranan penting dalam proses pencarian solusi optimal, validitas solusi ditentukan berdasarkan seberapa baik fungsi objektif dan fungsi *fitness* dibentuk oleh seorang perancang, sehingga setiap perancang walaupun menggunakan metode ALGEN yang sama akan tetap memberikan hasil akhir yang berbeda. Dalam pemodelan AVR fungsi objektif dibentuk berdasarkan akumulasi kesalahan tanggapan sistem AVR dalam suatu durasi waktu simulasi tertentu. Teori yang mendukung pembentukan fungsi objektif adalah, kinerja suatu sistem kontrol terukur berdasarkan metode ITAE (*Integral of Time Multiplied by Absolute Error*) atau besar akumulasi kesalahan tanggapan sistem dikalikan dengan waktu untuk satu periode simulasi, secara matematis dapat dirumuskan seperti pada (1.24)

$$ITAE = \int_{t=0}^{t_s} t |\varepsilon(t)| dt \quad (1.24)$$

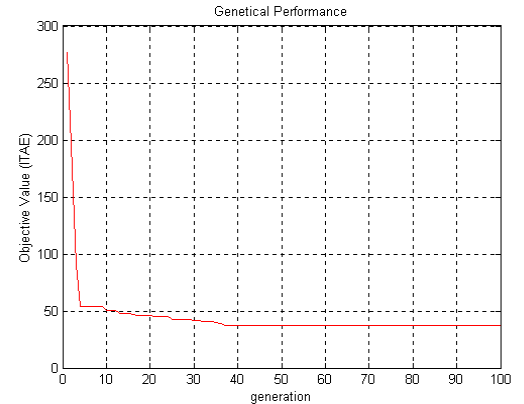
atau secara perhitungan diskrit dapat dihitung sebagai berikut :

$$ITAE = \sum_{t=0, t+\Delta t}^{t_s} t |\varepsilon(t)| \quad (1.25)$$

dimana Δt adalah lebar pencuplikan waktu simulasi.

6. Hasil Pengujian dan Analisa

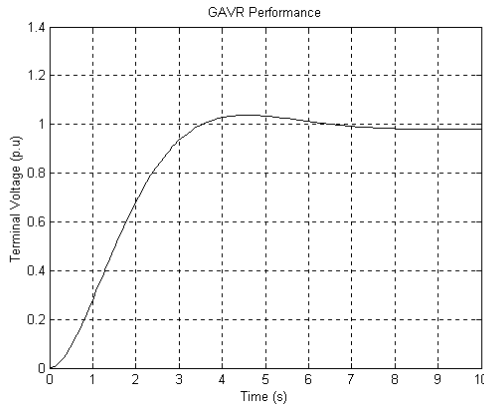
Pengujian unjuk kerja AVR yang dioptimasi menggunakan algoritma genetik diambil untuk beberapa skenario pengujian. Contoh pengujian dilakukan dengan menentukan parameter algoritma genetik sebagai berikut; ukuran populasi=40, jumlah generasi=100, Probabilitas Rekombinasi=0.9, Probabilitas Mutasi=NaN, Gap Generasi=0.9, Konstanta Mutasi=0.7 dan kepresisian=20bit. Hasilnya berikut ini.



Gambar 12. Kurva Unjuk Kerja ALGEN Uji-1

Gambar 12 memperlihatkan bagaimana ALGEN Uji-1 melakukan proses pencarian model terbaik AVR dari generasi ke generasi. Proses pencarian model terbaik AVR dilakukan berdasarkan ITAE (*Integral of Time Multiplied by Absolute Error*) dimana dengan metode ini unjuk kerja AVR dinilai. Nilai ITAE pada generasi pertama ALGEN Uji-1 sangat tinggi yaitu berkisar 275, ITAE yang tinggi ini disebabkan pada generasi pertama semua individu dalam suatu populasi masih memiliki materi genetik yang kurang baik dan terbentuknya pun secara random. 10 generasi pertama ALGEN Uji-1 sudah

menghasilkan individu dengan kualitas yang memadai dengan nilai ITAE yang diberikan jauh lebih kecil dari generasi pertama yaitu berkisar 51. Generasi 10-40 ALGEN Uji-1 tidak memberikan hasil pencarian yang signifikan, hal ini disebabkan terjadi persaingan yang ketat yang menyebabkan nilai ITAE yang diberikan oleh individu terbaik tidak begitu berbeda dengan nilai ITAE individu terbaik generasi sebelumnya. Pada generasi 41-100 karakteristik ALGEN sebagai metode pencarian solusi global (*Global Solution*) tampak jelas. Penampakan garis lurus dalam kisaran generasi 41-100 menginformasikan bahwa individu superior sudah terbentuk, individu superior adalah individu yang memiliki kualitas terbaik dari generasi ke generasi. Informasi gen yang dimiliki individu superior merupakan informasi model AVR terbaik yang tercapai. Dari Lampiran-A dapat dilihat secara detail nilai ITAE setiap individu dari generasi ke generasi untuk mendapatkan solusi yang tepat (*Exact Solution*).



Gambar 13. Tegangan Terminal Generator GAVR Uji-1

Solusi tepat diperoleh pada generasi ke 100 (Lampiran-A) terdapat pada individu ke-3 dan informasi yang dapat diperoleh dari kromosom individu tersebut adalah sebagai berikut :

1. $K_A = 2.9757$
2. $K_E = 1.0605$
3. $K_G = 0.9894$
4. $K_F = 1.0016$
5. $K_R = 0.6974$
6. $ITAE = 37.1188$

Fungsi alih GAVR (*Genetic-Based AVR Model*) dapat dihitung menggunakan (1.6) dan dihasilkan :

$$V_{T}(s) = \frac{78.06s^2 + 3513s + 39030}{V_{\text{reff}}(s) \cdot (s^5 + 59s^4 + 3120s^3 + 49950s^2 + 42590s + 39720)} \quad (1.26)$$

kesalahan keadaan tunak (SSE) GAVR untuk parameter tegangan terminal generator dihitung menggunakan (1.9) dan dihasilkan :

$$E_{ss} = 1 - \frac{39030}{39720} = 1 - 0.9826 = 0.0174 \text{ p.u}$$

Jadi jika tegangan terminal generator diset 13.8 kV, maka keluaran GAVR hanya memberikan tegangan sebesar $0.9826 \cdot 13.8 \text{ kV}$ atau sebesar 13.56 kV. Kurva tanggapan GAVR dapat dilihat pada gambar 13. GAVR yang dihasilkan melalui ALGEN Uji-1, dapat dilakukan perhitungan untuk kriteria sistem kontrol yang lain yaitu *Peak Time*, *Percent Overshoot*, *Rise Time* dan *Settling Time*. Nilai setiap kriteria tersebut dapat dihitung dengan menggunakan (1.16), (1.17), (1.18) dan (1.19) akan menghasilkan :

1. *Peak time* = 3.83067 detik
2. *Percent overshoot* = 19.5273 %
3. *Rise time* = 1.70252 detik
4. *Settling time* = 9.03282 detik

Tabel 2. Hasil pengujian AVR dengan beberapa skenario pengujian dan perbandingan dengan model yang di ada referensi H.Sadaat.

Model	Faktor Penguatan					Unjuk Kerja Sistem						
	K _A	K _E	K _G	K _S	K _R	ITAE	E _{SS}	PT	PO	RT	ST	TC
ALGEN-1	2.9757	1.0605	0.9894	1.0016	0.6974	37.118	0.0174	3.83	19.53	1.702	9.03	29.6
ALGEN-2	2.1007	1.427	0.9985	1.0238	0.6877	36.796	0.0213	3.87	19.41	1.71	9.09	29.8
ALGEN-3	1.0671	1.8386	0.9044	1.0239	0.4226	34.095	0.014	4.37	17.72	1.98	10.04	27.17
ALGEN-4	1.8843	1.4169	0.9962	1.0637	0.6265	36.425	0.025	4.05	18.34	1.8	9.37	29.03
ALGEN-5	3.0773	1.0096	0.9998	1.001	0.6909	36.784	0.0127	3.82	19.7	1.71	9.03	31.8
ALGEN-6	1.2919	1.0127	0.9968	1.0151	0.2259	33.44	0.007	4.29	21.6	1.88	10.26	30.1
ALGEN-7	1.1694	1.1723	0.9975	1.0177	0.2692	33.56	0	4.24	21.52	1.86	10.16	24.48
ALGEN-8	3.0137	1.631	0.9554	1.0268	0.9216	45.96	0.033	3.85	17.69	1.72	8.83	24.31
ALGEN-9	1.1107	2.7401	0.997	1.0012	0.6858	37.68	0.015	3.83	19.74	1.69	9.03	24.31
Model-1#	10	1	1	*	1	*	0.091	0.79	0.25	82.46	19.04	*
Model-2#	10	1	1	2	1	264.6	0.091	6.08	2.95	4.13	8.08	*

Keterangan: PT=Peak Time, PO=Percent Overshoot, RT=Rise Time, ST=Settling Time, TC=Time Consumption, Ess=Steady State Error.

7. Kesimpulan

Dari ke-9 skenario pengujian unjuk kerja algoritma genetik dengan memvariasikan data parameter genetik, untuk memodelkan sistem AVR agar diperoleh tanggapan tegangan terminal generator terbaik, dapat disimpulkan sebagai berikut :

1. Secara global ke-9 skenario pengujian pemodelan sistem AVR menggunakan metode algoritma genetik menghasilkan tanggapan tegangan terminal generator lebih baik dibandingkan model sistem AVR pada referensi H.Sadaat
2. Model AVR terbaik yang berhasil dihasilkan oleh ALGEN Uji-6 dengan kenaikan unjuk kerja 87.36% dan terburuk oleh ALGEN Uji-8 dengan kenaikan 82.63 %.

8. Saran

Pemodelan AVR menggunakan metode algoritma genetik memberikan hasil yang memuaskan untuk semua skenario pengujian untuk pemodelan AVR yang sederhana, untuk penelitian lebih lanjut ada beberapa saran yaitu :

1. Model AVR yang digunakan adalah model AVR standart IEEE, dimana orde sistem jauh lebih tinggi dan memiliki karakteristik tak linier.
2. Model sistem untuk penelitian lebih lanjut dapat menggunakan kasus sistem tenaga *Multimachine* dan *Multiarea*

DAFTAR PUSTAKA

- [1], *Genetic Server and Genetic Library*, www.neurodimension.com
- [2] Byungkyu Park, Carroll J. Messer, Thomas Urbanik II, *Traffic Signal Optimization Program For Oversaturated Conditions: A Genetic Algorithm Approach*, Texas A&M University, 1988
- [3] Davis, Lawrence ED, *Hand Book of Genetic Algorithm*, Van Nostrand Reinhold, New York, 1991

- [4] EuroGP 2001 *14th European Conference on Genetic Programming*, 18-20 April 2001, Lake Corno (Milan), Italy
- [5] Golberg, David, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company. Inc
- [6] Graham, D. *Tuning the PID controller Based on a Genetic Algorithm (GA)*, Paper Implementation of GA in Control Field, 1994
- [7] Ogata, K, *Tenik Kontrol Automatik Jilid I dan 2*
- [8] Kim-Fung-Man : *Genetic Algorithms : Concepts and Designs*, City University of Hongkong.
- [9] Hermana, Nana, *Economic Dispatch dengan Menggunakan Algoritma Genetik*, Proceeding SSTL-II, UNDIP
- [10] Malik, Leevy D., *Optimasi Base Point Unit Pembangkit Thermal Multi Area dengan Algoritma Genetik*, Tugas Akhir, Universitas Diponegoro, Semarang
- [11] Triwiyatno, Aris et.al, *Perbandingan Sistem Eksitasi Konvensional dan Non-Konvensional Terhadap Kestabilan Generator*, Proceeding SSTL-I ITB
- [12] W.Sadaat, H, *Power System Analysis*, McGraw Hill Series.
- [13] Welstead, Stephen T., *Neural Network And Fuzzy Logic Application in C/C++*, John Wiley & Sons, Inc., 1994
- [14] Wood, A, *Power Generation, Operation & Control*, Prentice Hall Series

Mengetahui / menyetujui

Pembimbing I :

Ir. Tejo Sujmadi, MT
NIP.132 162 547

Pembimbing II :

Trias Andromeda, ST MT
NIP.132 283 185