

PENGATURAN FREKWENSI SISTEM TENAGA MENGUNAKAN ALGORITMA GENETIK

Nugroho Agus Darmanto^{*)}

Departemen Teknik Elektro, Universitas Diponegoro Semarang
Jl. Prof. Sudharto SH, Kampus UNDIP Tembalang Semarang 50275, Indonesia

^{*)}E-mail : nugroho.elundip@gmail.com

Abstrak

Load Frequency Control (LFC) adalah salah satu mekanisme dalam suatu sistem tenaga listrik yang digunakan untuk memperbaiki mutu dan keandalan frekuensi sistem. Perencanaan sistem pengendalian frekuensi sistem tenaga membutuhkan metode penyelesaian yang optimal dan sangat fleksibel diaplikasikan ke model sistem yang ada. Makalah ini membahas bagaimana menghasilkan suatu solusi optimal untuk perencanaan sistem pengendalian frekuensi sistem agar diperoleh deviasi frekuensi sistem yang paling kecil jika terjadi perubahan beban. Metode yang digunakan adalah algoritma genetik yang dikembangkan oleh Goldberg yaitu model algoritma genetik sekuensial yang bekerja pada komputer *single* prosesor. Makalah ini mengaplikasikan algoritma genetik dalam beberapa model yaitu GA RWS SSC-10, GA RWS DSC -10, GA RWS SSC-50, GA RWS DSC -50, GA SUS SSC-10, GA SUS DSC-10, GA SUS SSC-50, dan GA SUS DSC-50 yang semuanya dibahas dan dibandingkan. Sebagai uji validasi solusi optimal perencanaan LFC, dibandingkan dengan metode *Optimal Pole Placement (OPP)* yang dikembangkan oleh Saadat.

Kata kunci : *Load Frequency Control, Algoritma Genetik*

Abstract

Load Frequency Control (LFC) is a mechanism in power system that is used to improve the quality and reliability of system frequency. Planning of power system frequency control system requires optimal and very flexible method of completion applied to existing system model. This paper discusses how to generate an optimal solution for system control system frequency planning in order to obtain the smallest system frequency deviation in case of load changes. The method used is genetic algorithm developed by Goldberg which is a sequential genetic algorithm model that works on single-processor computers. This paper applies genetic algorithm in several models of GA RWS SSC-10, GA RWS DSC-10, GA RWS SSC-50, GA RWS DSC-50, GA SUS SSC-10, GA SUS DSC-10, GA SUS SSC-50, and GA SUS DSC-50 which are all discussed and compared. The validation test of LFC optimal planning solution is compared with *Optimal Pole Placement (OPP)* method developed by Saadat.

Keyword: *Load Frequency Control, Genetic Algorithm*

1. Pendahuluan

Pendekatan dasar dalam merencanakan sistem kontrol ada kalanya memerlukan metode coba-coba, baik dalam menentukan elemen kontrol maupun menentukan *setting* yang tepat bagi parameter-parameter pengontrol. Pendekatan yang paling umum dijumpai yaitu dengan menentukan model persamaan matematis setiap sistem, kemudian menentukan parameter kontrolnya, dan pada akhirnya melakukan simulasi komputer untuk mengetahui tanggapan sistem yang optimal.^[1]

Uji unjuk kerja suatu sistem memerlukan orang-orang yang memiliki keahlian yang tinggi dan memiliki

penalaran yang baik tentang sistem kontrol. Model matematis sistem tidak selalu dapat diterapkan pada semua sistem kontrol, hal ini dikarenakan tingkat kesulitan sistem dan kerumitan dalam memecahkan persamaan-persamaan matematis yang tidak linier. Untuk mendapatkan hasil perencanaan sistem kontrol yang optimal, kadang kita terbentur pada sistem yang belum dapat digambarkan dengan baik, sehingga walaupun kita mampu memodelkan sistem secara matematis dengan baik, masih banyak dijumpai kendala.^{[1][2]}

Berangkat dari sifat dan karakteristik algoritma genetik yaitu merupakan teknik pencarian adaptif yang bersifat pencarian banyak titik dan dapat memecahkan

permasalahan dengan tingkat kesulitan yang tinggi, yang ditunjukkan dengan komputasi non-parameter, dapat memberikan dorongan yang kuat pada penentuan dan penalaan bati umpan balik sistem secara optimal dengan melakukan optimalisasi nilai eigen sistem pada bidang kompleks (s -plane) guna memperbaiki kelakuan dinamik sistem LFC (*Load Frequency Control*). Adaptasi *pay-off* dalam algoritma genetik memberikan keuntungan utama yaitu tidak membutuhkan kemampuan matematis yang tinggi dalam optimalisasi nilai eigen, karena dalam adaptasi *pay-off* kita hanya memberikan masukan nilai parameter dan tidak perlu mengetahui bagaimana masukan diolah tetapi kita hanya membutuhkan keluaran yang digunakan untuk penilaian unjuk kerja skemata pada mekanisme algoritma genetik.^{[3][4]}

Algoritma Genetik merupakan salah satu dari beberapa metode analisa heuristic yang sangat berguna untuk membantu menentukan bati umpan balik optimal dengan optimalisasi nilai eigen untuk optimalisasi deviasi frekuensi sistem LFC (*Load Frequency Control*). Dalam menentukan *gen* umpan balik yang optimal dengan optimalisasi nilai eigen sistem pada bidang kompleks menggunakan Algoritma Genetik, dirumuskan hal-hal sebagai berikut :

- Dalam tulisan ini, masalah yang akan dibahas adalah tentang penerapan metode algoritma genetik guna mengatur kestabilan frekuensi tegangan terhadap perubahan beban elektrik generator.
- Hasil studi ini akan dibandingkan dengan metode OPP (*Optimal Pole Placement*) agar dapat diketahui seberapa besar tingkat kebenarannya serta tingkat perbaikan kestabilan frekuensi tegangan.

Dalam penentuan bati umpan balik optimal, analisa dibatasi dengan beberapa hal :

- Metode algoritma genetik yang digunakan hanya menggunakan siklus genetik standar yaitu seleksi dan reproduksi, rekombinasi serta mutasi.
- Kontrol kualitas performansi sistem hanya menggunakan metode ITAE (*Integral of Time Multiplied by Absolute Error*).
- Proses penalaan parameter kontrol bati umpan balik optimal tidak direkomendasikan dalam proses penalaan secara *real-time* dengan menggunakan suatu *hardware* tertentu.
- Ketelitian parameter-parameter yang dioptimasi dipilih untuk kepresisian 10 bit per parameter dan 50 bit per parameter.
- Evaluasi ITAE dilakukan dengan melakukan kuantisasi nilai *error* yang dicuplik setiap 1/100 detik
- Model algoritma genetik yang digunakan bukan algoritma genetik dengan banyak populasi (*Multipopulation*) tetapi populasi tunggal tanpa melalui proses migrasi.

2. Metoda Algoritma Genetik

Algoritma genetik pertama kali dikembangkan oleh John Holland dari Universitas Michigan pada tahun 1975 dengan paper "*Adaptation in Natural and Artificial System*". Dalam paper ini banyak dibicarakan tentang proses adaptasi natural dan pengembangan perangkat lunak yang berbasis pada mekanisme alam.^[3]

2.1 Umum

Algoritma genetik (ALGEN) adalah metode pencarian stokastik global yang memiliki kesamaan pada mekanisme evolusi alam. Algoritma genetik bekerja pada suatu populasi solusi permasalahan. Perkembangan populasi pada generasi berikutnya dihasilkan melalui mekanisme genetik yang dihasilkan dari perkawinan dua individu yang memiliki kualitas di atas rata-rata. Proses perkembangbiakan individu akan terus berlangsung dan akan memberikan hasil yaitu suatu populasi dimana setiap individu memiliki kualitas baik. Proses evolusi akan berlangsung dalam beberapa generasi yang ditentukan. Setiap individu dalam suatu populasi memiliki kromosom yang mempunyai struktur gen yang sama. Setiap kromosom membawa informasi tentang parameter-parameter yang dibutuhkan dalam suatu proses pemecahan masalah.^{[3][4]}

Algoritma genetik memiliki perbedaan yang mendasar dengan metode-metode pencarian solusi optimal dengan basis model matematika kalkulus (*calculus based*). Perbedaan-perbedaan tersebut adalah sebagai berikut :

- Mekanisme optimasi algoritma genetik bekerja berdasarkan kromosom, dimana setiap kromosom menyimpan informasi parameter-parameter yang dibutuhkan untuk penyelesaian masalah. Jadi mekanisme algoritma genetik tidak bekerja langsung pada parameter-parameter tersebut.
- Proses pencarian solusi optimal pada mekanisme algoritma genetik tidak dilakukan pada satu titik pencarian, tetapi pada sekumpulan titik pencarian.
- Algoritma genetik tidak membutuhkan prosedur-prosedur matematis dalam mencari solusi optimal, tetapi algoritma genetik menggunakan informasi langsung dari hasil transfer tiap-tiap parameter ke suatu fungsi yang dapat mewakili tujuan dari proses optimasi yang sedang dilakukan.
- Mekanisme genetik digunakan dalam pemrosesan kode parameter suatu permasalahan, melalui proses seleksi, rekombinasi, dan mutasi untuk memperoleh solusi optimal.
- Proses pencarian solusi optimal menggunakan metode algoritma genetik dengan titik acuan sembarang, untuk menghindari solusi optimal lokal.

- Mekanisme pencarian terbimbing diberikan melalui penilaian terhadap kualitas kode atau kromosom yang dimiliki oleh setiap individu dalam suatu generasi.

2.2 Terminologi Algoritma Genetik

Secara garis besar algoritma genetik banyak memiliki kesamaan dengan mekanisme genetik alami dan seleksi alam. Terminologi algoritma genetik dan genetik alami dapat dilihat pada Tabel 2.1 berikut [6].

Tabel 2.1. Terminologi Algoritma Genetik

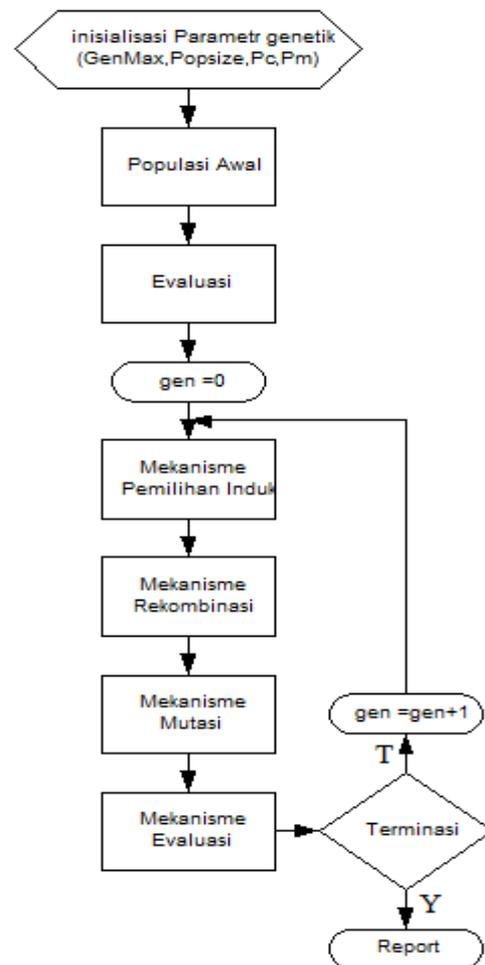
Algoritma Genetik	Sistem Genetik Alami
String	Kromosom
Karakter	Gen
Nilai Karakter	Alele
Posisi Karakter	Lokus
Struktur	Genotip
Parameter	Fenotip

2.3 Siklus Algoritma Genetik

Setelah format permasalahan yang akan dioptimasi sudah dirubah ke format genetik, maka siklus algoritma genetik yang dibutuhkan adalah membentuk populasi awal sebagai bahan dasar untuk membentuk generasi berikutnya.

Mekanisme genetik seperti rekombinasi, mutasi kromosom, serta mekanisme seleksi alam sangat berperan penting dalam proses pembentukan generasi baru yang memiliki individu-individu yang berkualitas. Secara garis besar siklus tersebut dapat dijelaskan seperti gambar 2.1 berikut :

Siklus ini sangat sederhana dalam beberapa hal, antara lain operator genetik yang digunakan dan mekanisme evaluasi individu dalam setiap populasi. Mekanisme laporan hasil pencarian menggunakan algoritma genetik dilakukan untuk mengetahui nilai-nilai setiap parameter yang dihasilkan dan hasil simulasi LFC berupa tanggapan frekuensi sistem terhadap perubahan beban. Mekanisme pemberhentian atau terminasi proses menghasilkan generasi, ditentukan berdasarkan perkiraan jumlah generasi maksimum yang dapat memberikan solusi optimal.



Gambar 2.1. Siklus Sederhana Algoritma Genetik

2.4 Mekanisme Seleksi/Reproduksi [3]

Pada proses reproduksi dibutuhkan dua individu yang digunakan sebagai induk. Secara matematis probabilitas terpilihnya individu sebagai induk sebanding dengan perbandingan antara *fitness* suatu individu dengan rata-rata *fitness* dalam suatu populasi. Jika suatu populasi *m* merupakan anggota suatu generasi *H* pada suatu periode waktu *t* dan kita nyatakan dalam bentuk $m(H,t)$, serta individu $A(t)$ merupakan individu yang hidup dalam populasi $m(H,t)$ maka kemungkinan terpilihnya individu $A(t)$, dinyatakan sebagai berikut :

$$P_i(A(t)) = \frac{f_i}{\sum f_i} \quad (2.1)$$

Jika ukuran populasi dinyatakan sebagai *n*, maka kemungkinan terbentuknya populasi pada periode waktu (*t+1*), atau $m(H, t+1)$ adalah sebagai berikut :

$$m(H, t + 1) = m(H, t) \frac{\sum f_i^{(H,t)}}{\sum f_i} \quad (2.2)$$

Dari persamaan (2.2), terlihat bahwa pertumbuhan suatu populasi dari generasi ke generasi sebanding dengan *fitness* rata-rata dalam suatu populasi $m(H,t)$. Algoritma perkembangan suatu populasi seperti ini dapat didekati dengan algoritma *roulette wheel*. Dalam *roulette wheel* setiap slot yang digunakan memiliki luasan yang sebanding dengan *fitness* setiap individu, dan jumlah slot yang tersedia harus sama dengan jumlah individu dalam suatu populasi $m(H,t)$.

Gambaran tentang *roulette wheel* dapat dijelaskan sebagai berikut. Jika dalam suatu populasi $m(H,t)$ terdapat 4 individu lengkap dengan informasi *fitness*nya, dapat ditabelkan seperti tabel 2.2 berikut :

Tabel 2.2. Data individu dalam populasi $m(H,t)$

No	Individu / Kromosom	Fitness
1	10101010101111101	0.3
2	10101111101010101	0.25
3	00000001111111010	0.35
4	11010101010101010	0.1

Dari tabel 2.2 dapat dibuat representasi dalam bentuk slot-slot *roulette wheel* dimana luasan setiap sektor dalam *roulette wheel* tersebut sebanding dengan *fitness* tiap-tiap individu dalam populasi seperti gambar 2.13. Dalam gambar tersebut diameter *roulette wheel* tidak menjadi permasalahan utama. Yang terpenting luas sektornya harus sebanding dengan nilai-nilai hasil evaluasi tiap individu.



Gambar 2.2 Roulette Wheel

Dari gambar 2.13 di atas, terlihat secara logika individu nomer 3 memiliki peluang tertinggi sebagai kandidat untuk menghasilkan keturunan yaitu 35%, kemudian diikuti 1 dengan peluang 30%, 2 dengan peluang 25%, dan 4 dengan peluang 10%.

Mekanisme *roulette wheel* dalam menentukan kandidat dilakukan dengan cara memutar *roulette wheel* tersebut dan kemudian dilihat individu nomer berapa yang terpilih oleh jarum *roulette*. Proses ini dilakukan dua kali karena untuk menghasilkan keturunan dibutuhkan dua individu

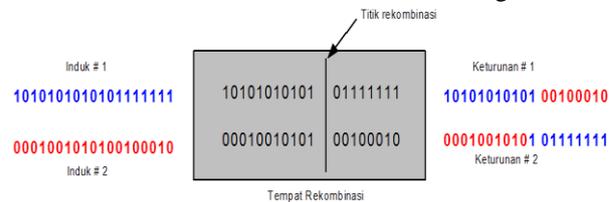
sebagai induk. Setiap keturunan yang dihasilkan ditampung dalam *buffer* individu sementara, dan jumlah keturunan yang ditampung harus sama dengan jumlah individu dalam suatu populasi.

2.5 Mekanisme Rekombinasi (*crossover*)^[3]

Rekombinasi adalah proses pertukaran struktur kromosom antara dua induk yang terpilih dalam proses seleksi dengan tujuan untuk menciptakan keragaman individu-individu baru yang tetap mewarisi sifat-sifat terbaik dari induk-induknya. Dalam algoritma genetik dilakukan atau tidaknya proses rekombinasi ditentukan oleh nilai peluang terjadinya peristiwa rekombinasi yang ditentukan oleh pengguna. Jenis-jenis mekanisme rekombinasi dapat dikelompokkan menjadi rekombinasi sisi (*site crossover*) dan rekombinasi seragam (*uniform crossover*) .

2.5.1 Single Site Crossover (SSC)^[7]

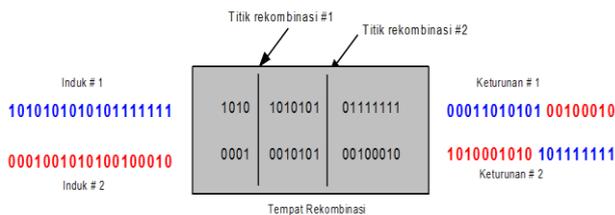
Rekombinasi jenis ini memiliki ciri ada satu titik tempat terjadinya perpindahan struktur kromosom antar kromosom induk. Titik tempat terjadinya awal dari pertukaran struktur kromosom induk ditentukan secara sembarang dalam rentang yang tidak melebihi dari panjang kromosom induk. Jenis rekombinasi seperti ini banyak digunakan karena efektifitas dalam penganekaragaman struktur kromosom individu lebih baik dan materi genetik dari kromosom induk tidak cepat hilang. Contoh mekanisme rekombinasi satu titik adalah sebagai berikut :



Gambar 2.3. Mekanisme Rekombinasi Satu Titik

2.5.2 Double Site Crossover (DSC)^[7]

Rekombinasi jenis ini memiliki ciri ada lebih dari satu titik tempat terjadinya perpindahan struktur kromosom antar kromosom induk. Titik-titik tempat terjadinya awal dari pertukaran struktur kromosom induk ditentukan secara sembarang dalam rentang yang tidak melebihi dari panjang kromosom induk. Rekombinasi jenis ini banyak digunakan jika panjang kromosom sangat panjang lebih dari 500 bit. Contoh mekanisme rekombinasi satu titik adalah seperti gambar 2.15. Titik-titik rekombinasi merupakan permasalahan utama dalam menentukan level efektifitas pembentukan keragaman individu dalam suatu populasi, yang pada akhirnya berpengaruh pada dinamika pencarian solusi optimal yang akan meningkat. Kelajuan tingkat rekombinasi dapat diatur dengan menentukan nilai probabilitas rekombinasi yang sesuai.

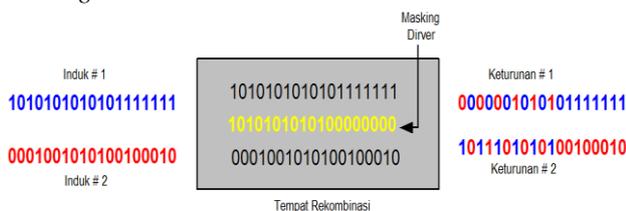


Gambar 2.4. Mekanisme Rekombinasi Banyak Titik

2.5.3 Uniform Crossover (UC)^[7]

Rekombinasi jenis ini tidak mengenal titik rekombinasi, karena modelnya sudah agak berbeda. Dalam rekombinasi ini kita harus menghasilkan kromosom *masking*. Kromosom ini memiliki panjang sama dengan kromosom induk tetapi dihasilkan secara acak.

Penentuan struktur kromosom induk yang harus direkombinasi disesuaikan dengan posisi bit yang berbobot satu pada kromosom *masking*. Rekombinasi jenis ini agak sedikit tidak efektif dalam menciptakan keragaman karena harus sesuai dengan kromosom *masking*.



Gambar 2.5. Mekanisme Rekombinasi Seragam

2.6 Mekanisme Mutasi^[3]

Mutasi adalah operator genetik yang merubah satu atau lebih gen dalam sebuah kromosom dari bentuk aslinya dan menghasilkan sebuah gen baru. Dengan gen baru yang dihasilkan, algoritma genetik dapat menghasilkan solusi yang lebih baik. Mutasi adalah bagian penting dalam penelusuran genetika karena dapat membantu menjaga populasi dari kemacetan pada saat optimasi lokal. Mutasi terjadi selama evolusi sesuai dengan probabilitas mutasi yang telah ditentukan. Probabilitas mutasi sebaiknya memiliki nilai yang kecil (0,001 hingga 0.05). Jika diberikan nilai yang terlalu tinggi dapat menyebabkan penelusuran keturunan kembali pada pola penelusuran acak primitif.

2.7 Simulasi Sederhana Algoritma Genetik^[3]

Untuk memudahkan pemahaman bagaimana algoritma genetik melakukan proses optimasi suatu permasalahan, di sini akan disimulasikan secara sederhana bagaimana algoritma genetik dapat menemukan solusi global dari suatu fungsi objektif $F_0(X) = F(X) = X^2$ untuk batasan nilai $X_{\min s} \leq X \leq X_{\max s}$. Mekanisme pencarian solusi optimal dari permasalahan di atas dapat ditulis sebagai berikut :

- ❑ Membentuk populasi awal beserta kromosom penyusunnya yakni parameter X yang dikodekan dalam biner dan memiliki panjang kromosom L dengan ukuran $N \{N \in \text{bilangan genap}\}$.
- ❑ Melakukan proses decode setiap kromosom ke nilai X aktual.
- ❑ Menghitung nilai *fitness* individu dengan menghitung fungsi objektif berdasarkan nilai aktual X setiap individu.
- ❑ Menghitung kemungkinan terpilihnya suatu individu untuk menjadi induk (P_s) berdasarkan nilai perbandingan *fitness* setiap individu terhadap jumlah *fitness* keseluruhan (mekanisme *roulette wheel*).
- ❑ Menghitung nilai ekspektasi induk berdasarkan nilai perbandingan nilai *fitness* setiap individu terhadap *fitness* rata-rata.
- ❑ Melakukan proses rekombinasi
- ❑ Melakukan proses mutasi
- ❑ Berhasil mendapatkan generasi baru, kemudian kembali ke point (2) hingga kriteria konvergensi tercapai atau aturan pemberhentian proses dilakukan.

Sekarang tiba gilirannya untuk mengimplementasikan prosedur-prosedur di atas ke dalam suatu contoh sederhana, yaitu menemukan berapa nilai X yang dapat membuat fungsi $F(x) = X^2$ bernilai maksimum dalam rentang X mulai 0 hingga 32. Sekuensi proses adalah sebagai berikut :

1. Membentuk populasi awal sebanyak 4 individu dengan panjang kromosom 5 bit. Populasi awal ini dibentuk secara acak, dalam hal ini penulis menggunakan koin logam untuk membentuk populasi awal.
2. Melakukan proses decode yaitu untuk mendapatkan nilai x aktual yaitu dengan algoritma pengubahan bilangan biner ke nilai desimal.
3. Menghitung nilai *fitness* individu berdasarkan nilai fungsi objektif terhadap nilai X.
4. Menghitung probabilitas seleksi (P_s) setiap individu.
5. Menghitung nilai ekspektasi induk dan menentukan berapa kali individu tersebut terpilih berdasarkan mekanisme *roulette wheel*. Proses ini disimulasikan dengan algoritma *floor* nilai ekspektasi individu.

Dari kelima proses di atas dapat ditabelkan sebagai berikut :

Tabel 2.3. Statistik populasi generasi pertama

Individu ke-n	Kromosom	Nilai x	$F_i = X^2$	$P_S = \frac{F_i}{\sum F}$	eksp = $\frac{F_i}{F}$	FLOOR
1	01101	13	169	0.14	0.58	1
2	11000	24	576	0.49	1.97	2
3	01000	8	64	0.06	0.22	0
4	10011	19	361	0.31	1.23	1
N=4	biner		$\Sigma 1170$	$\Sigma 1.00$	$\Sigma 4.00$	$\Sigma 4.00$

Dari tabel 2.3 di atas terlihat individu nomer 1 terpilih satu kali, individu nomer 2 terpilih dua kali, individu nomer 3 tidak pernah terpilih, dan terakhir individu nomer 4 terpilih satu kali. Setelah sebuah induk sudah kita peroleh, sekuensi berikutnya adalah :

1. Melakukan rekombinasi dengan probabilitas $P_c = 1.0$ (selalu terjadi rekombinasi).
2. Melakukan mutasi dengan probabilitas $P_m = 0.001$, nilai ekspektasi untuk operator mutasi adalah $4 \cdot 5 \cdot 0.001 = 0.02$. Jadi untuk satu generasi ini tidak pernah terjadi mutasi karena nilainya $> P_m$.
3. Menghasilkan generasi baru dan siap melakukan proses evaluasi.

Dari ketiga proses di atas, hasilnya dapat dilihat pada tabel 2.4 berikut :

Tabel 2.4. Statistik populasi generasi baru

Nomer Induk	Induk-Induk Terpilih	Pasangan	Titik Rekombinasi	Keturunan	X	Fitness (Fi)
1	01101	2	4	01100	12	144
2	11000	1	4	11001	25	625
3	11000	4	2	11011	27	729
4	10011	3	2	10000	16	256
Jumlah →						$\Sigma 1754$

Dari tabel 2.4 terlihat individu-individu baru dalam hal ini keturunan-keturunan yang dihasilkan ternyata memberikan *fitness* yang rata-rata lebih baik daripada *fitness* induknya. Selanjutnya akan dilakukan proses evaluasi seperti yang sudah dijelaskan seperti mekanisme yang pertama. Proses terminasi ditentukan berdasarkan jumlah maksimum generasi yang dihasilkan atau kriteria konvergensi telah terpenuhi.

2.8 Implementasi

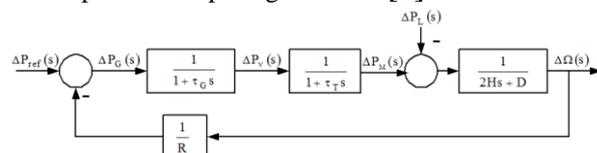
Implementasi algoritma genetik dalam penyelesaian optimal suatu permasalahan harus menganut kaidah-kaidah mekanisme genetik dalam format algoritma genetik. Algoritma genetik bekerja dalam suatu populasi individu dengan karakteristik kromosom dan gen yang berbeda. Kromosom dan gen adalah tempat dimana parameter yang akan dioptimasi berada. Format kromosom yang paling umum dipakai adalah format

biner. Hal ini mengindikasikan setiap parameter-parameter yang akan dicari titik optimalnya harus dikodekan dalam format tersebut. Berarti konsep pertama yang harus diperhatikan, bahwa algoritma genetik bekerja dalam sistem diskrit dan optimasi yang dilakukan bersifat kombinasi (*Combinatorial Optimization*). Berarti jika kita mengkodekan suatu parameter dalam suatu kromosom dengan panjang kromosom 4 bit maka banyaknya nilai kombinasi yang kita miliki adalah 15 buah dan yang dimaksud dengan nilai diskrit dengan adanya 15 buah nilai kombinasi, maka besar setiap bagian parameter yang terdiskritisasi adalah skalar parameter dibagi dengan 15.

Proses pencarian algoritma genetik adalah pencarian acak yang terbimbing (*Guided Random Search*)^[4]. Proses ini membutuhkan informasi tambahan mengenai kualitas kromosom yang dimiliki setiap individu dalam suatu populasi. Kualitas setiap kromosom individu ditentukan berdasarkan fungsi objektif yang terbentuk dari suatu permasalahan yang ada. Suatu besaran atau nilai yang menyatakan kualitas individu disebut sebagai nilai *fitness*. Nilai *fitness* ini selanjutnya akan digunakan sebagai informasi utama dalam pemilihan induk-induk untuk menghasilkan generasi baru. Semakin tinggi nilai *fitness* yang dimiliki suatu individu, maka semakin baik nilai parameter yang dihasilkan dan semakin memuaskan nilai yang dihasilkan oleh suatu fungsi objektif terhadap parameter tersebut. Proses optimasi atau implementasi algoritma genetik dalam suatu permasalahan tertentu, harus dibuat dahulu suatu model yang dapat dibawa ke format optimasi yang bersifat genetik. Proses ini akan dijelaskan pada subbab berikut ini.

2.9 Memodelkan sistem yang akan dioptimasi

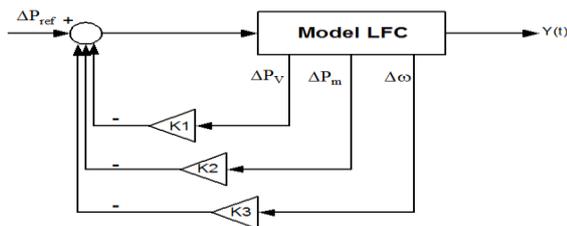
Dalam tulisan ini digunakan sistem LFC untuk mendemonstrasikan mekanisme algoritma genetic guna mencari *setting* optimal untuk faktor penguatan umpan balik agar diperoleh deviasi frekuensi yang kecil jika terjadi perubahan beban. Model LFC sendiri sudah dijelaskan di atas dan kita peroleh diagram bloknya yang telah diperlihatkan pada gambar 2.6[1]



Gambar 2.6. Diagram Blok LFC

Dalam makalah ini deviasi frekuensi sistem tenaga dikompensasi menggunakan sistem umpan balik statik terhadap semua variabel dinamik sistem yang kemudian dijumlahkan dengan sinyal referensi. Metode ini dikenal sebagai metode penempatan nilai eigen optimal (*Optimal Pole Placement*) pada bidang-s.

Metode ini menempatkan beberapa nilai eigen yang dikehendaki di sebelah kiri sumbu imajiner, yang kemudian dengan metode Ackermann dapat ditentukan berapa besar faktor penguatan sistem umpan balik statik yang harus diberikan^[1]. Dalam makalah ini algoritma genetik akan digunakan untuk mencari nilai-nilai penguatan optimal sistem umpan balik statik agar diperoleh posisi nilai eigen di sebelah kiri sumbu imajiner bidang-s dan diperoleh kelakuan sistem yang paling stabil dan memberikan deviasi frekuensi sistem yang terkecil. Sistem yang dikehendaki dalam penelitian ini adalah seperti yang terdapat pada gambar 2.7 .



Gambar 2.7. Model Sistem yang dioptimasi

Dalam gambar ini, objek kontrol adalah representasi model LFC dalam persamaan ruang keadaan, nilai $K_1, K_2,$ dan K_3 adalah faktor penguatan yang akan dicari sehingga diperoleh posisi nilai eigen yang paling optimal dan memberikan kelakuan sistem yang paling baik.

2.10 Model Kromosom

Sistem LFC yang digunakan dalam makalah ini mengandung 3 parameter yang harus ditentukan nilai optimalnya, yaitu : $K_1, K_2,$ dan $K_3,$ yang ketiganya berhubungan dengan $\Delta P_V, \Delta P_m,$ dan $\Delta \omega$ sebagai variabel dinamik dalam LFC. Nilai $K_1, K_2,$ dan K_3 kesemuanya sudah barang tentu memiliki batas yaitu batas minimum dan batas maksimumnya. [3]

Dalam makalah ini, setiap nilai K dikodekan dalam **biner** dengan panjang sama dengan **subchrom**. Maka nilai aktual K dapat kita peroleh dengan persamaan pemetaan berikut ini :

$$K_{\text{actual}} = \text{decode}(K_n) \times \frac{K_{\text{maks}} - K_{\text{min}}}{2^{\text{subchrom}} - 1} + K_{\text{min}} \quad (2.3)$$

dimana,

subchrom : Panjang kode biner tiap parameter.

K_{actual} : Vektor K yang sesungguhnya.

K_{maks} : Vektor batas maksimum vektor $K.$

K_{min} : Vektor batas minimum vektor $K.$

$\text{Decode}(K_n)$: Fungsi biner ke desimal tiap kode parameter.

dan besar ruang pencarian genetik adalah $2^{3 \times \text{subchrom}} - 1.$

2.11 Fungsi Objektif dan Fungsi *Fitness* ^[3]

Fungsi objektif dalam penelitian perilaku sistem LFC dengan menggunakan sistem umpan balik statis adalah bagaimana membuat $\dot{X}(t) = Ax + Bu(t),$ dimana $u(t) = -Kx,$ agar diperoleh deviasi frekuensi sistem yang paling kecil. Dari dua persamaan tersebut diperoleh konstanta kelakuan sistem adalah $(A-B*K).$ Matrik ini memberikan informasi posisi nilai eigen sistem pada bidang-s. Posisi atau lokasi nilai eigen yang dikehendaki sejauh mungkin di sebelah kiri sumbu imajiner. Jadi fungsi objektif yang terbentuk adalah sebagai berikut :

$$F_{\text{Objektif}} = \text{Min} \{ \text{Re}(\lambda) \} = \text{Min} (\text{ITAE}) \quad (2.4)$$

dimana,

ITAE : adalah $\int t|e(t)|dt$ (Indek unjuk kerja sistem)

λ : Nilai eigen sistem

Fungsi *fitness* dibangun berdasarkan fungsi objektif yang terbentuk. Nilai *fitness* harus dinormalisasi antara 0 hingga 1, sehingga kita dapat mengetahui seberapa baik individu-individu yang terbentuk dalam suatu populasi. Jadi fungsi *Fitness* yang terbentuk adalah :

$$F_{\text{Fitness}} = M.F_{\text{Objektif}} \quad (2.5)$$

dimana M adalah konstanta normalisasi F_{fitness} agar bernilai antara 0 dan 1.

2.12 Model Mekanisme Genetik

Mekanisme genetik yang digunakan yaitu proses pemilihan induk menggunakan metode *Roulette Wheel Selection* (RWS) dan *Stochastic Universal Sampling* (SUS). Mekanisme rekombinasi menggunakan model rekombinasi satu titik (*Single Point Crossover*) atau metode SSC dan rekombinasi dua titik (*Double Point Crossover*) atau DSC dan mekanisme mutasi menggunakan metode *flip bit.*

2.13 Kriteria Konvergensi

Kriteria konvergensi dalam makalah ini adalah, pertama jika generasi yang dihasilkan sudah melebihi jumlah maksimum generasi yang ditentukan, dan kedua jika terdapat 20 generasi yang tidak terjadi lagi perubahan nilai *fitness* yang dihasilkan atau sudah terjadi saturasi. Kondisi ini memberikan informasi kepada kita bahwa sudah tidak ditemukannya lagi individu yang memiliki kualitas dengan individu sebelumnya.

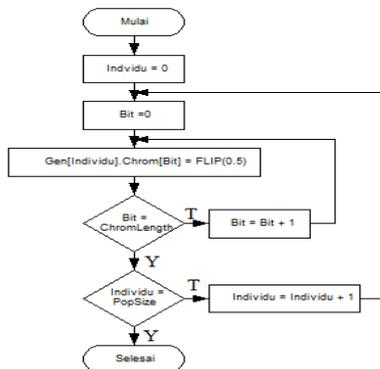
3. Hasil dan Analisa

3.1 Struktur Data

Struktur data yang digunakan dalam algoritma genetik adalah struktur data kolom yang menyatakan posisi semua parameter yang digunakan dalam suatu kromosom. Struktur data dibentuk dengan menggunakan informasi kromosom, parameter, ID individu, parameter-parameter, fitness, nilai objektif dan informasi tambahan mengenai sistem antara lain ITAE dan deviasi frekuensi. Struktur data memegang peranan penting dalam proses pengambilan kembali informasi proses pencarian solusi dari struktur kromosom tiap individu dalam suatu generasi. [3][7]

3.2 Populasi Awal

Populasi awal atau generasi awal adalah sekumpulan individu-individu sebanyak PopSize, dimana kromo-som biner mereka terbentuk berdasarkan mekanisme random. Mekanisme pembentukan populasi awal dapat dilihat seperti gambar 3.3 berikut :



Gambar 3.3. Mekanisme Pembentukan Populasi Awal

3.3 Mekanisme Evaluasi Individu

Fungsi objectif dalam makalah ini ditulis sebagai berikut :

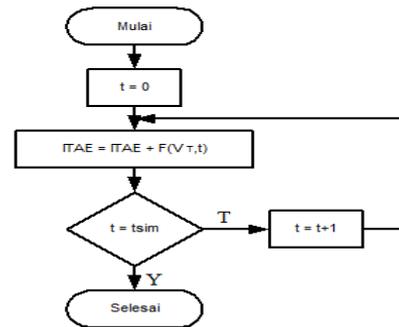
$$ObjValInd = Objective(Tspan)$$

dimana,

ObjValInd : Nilai objektif setiap individu

Tspan : Durasi Simulasi γ

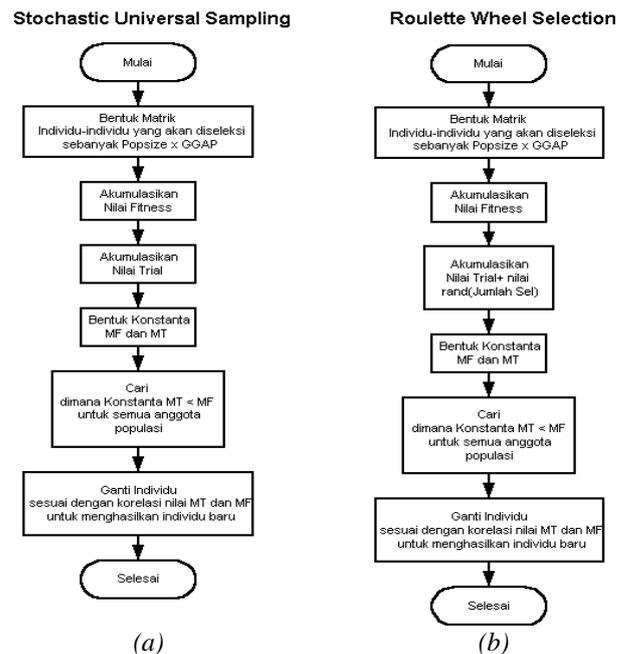
Mekanisme perhitungan nilai ITAE sistem sesuai dengan gambar 3.4 berikut :



Gambar 3.4 Mekanisme Perhitungan ITAE

3.4 Seleksi

Operator seleksi yang digunakan adalah model *Roulette Wheel Selection* (RWS), dimana proses pemilihan induk berdasarkan fitness masing-masing individu dalam suatu populasi. Operator model RWS paling luas digunakan dalam aplikasi algoritma genetik karena operator ini dapat menjaga kualitas individu dalam setiap generasi yang dihasilkan. Mekanisme operator seleksi model RWS dan SUS dapat dilihat pada gambar 3.5 berikut:



Gambar 3.5 : (a) Operator Seleksi Model SUS
(b) Operator Seleksi Model RWS

Akumulasi nilai fitness merupakan penjumlahan parsial untuk menentukan level fitness individu yang akan terpilih sebagai induk. Jika nilai akumulasi nilai tidak bisa melebihi nilai *Rand* maka proses pemilihan induk berdasarkan urutan individu dalam suatu populasi. Sintaks yang digunakan untuk melakukan proses seleksi adalah sebagai berikut :

$$Parents = SelectRWS('TypeSelect', OldChrom)$$

dimana,

Parents : Individu yang terpilih sebagai induk.

OldChrom :Kromosom individu generasi sebelumnya

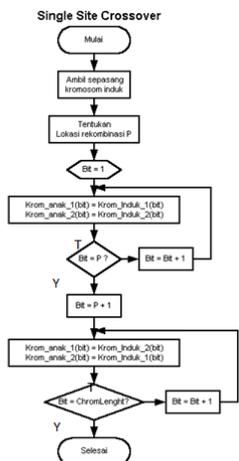
TypeSelect : Tipe Seleksi

3.5 Rekombinasi

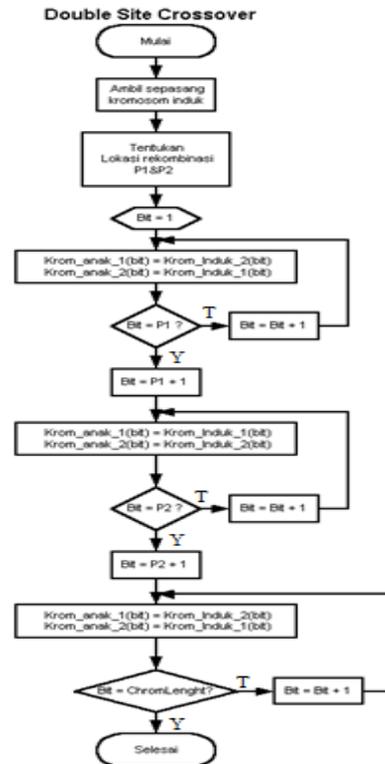
Operator Rekombinasi digunakan untuk menghasilkan keragaman genetik setiap kromosom individu, proses ini dilakukan melalui mekanisme pindah silang melalui sepasang kromosom induk.

Metode operator Rekombinasi yang digunakan dalam makalah ini adalah *Single Site Crossover* (SSC) dan *Double Site Crossover* (DSC). Alasan utama pemilihan model Rekombinasi tipe SSC dan DSC karena kedua model ini banyak direkomendasikan oleh banyak pakar karena efektifitas dalam peragaman gen dengan kerusakan materi genetik yang minimal. Materi genetik adalah setiap bit dalam sebuah kromosom yang memiliki derajat tertinggi.

Mekanisme rekombinasi SSC dapat dilihat pada gambar 3.6 dan DSC pada gambar 3.7 :



Gambar 3.6 : Mekanisme Rekombinasi Tipe SSC



Gambar 3.7 Mekanisme Rekombinasi Tipe DSC

Lokasi titik rekombinasi ditentukan secara *random* dalam rentang $ChromLength-1$. Jika lokasi rekombinasi sama dengan $ChromLength$ maka proses rekombinasi tidak dilakukan. Sintaks yang digunakan untuk melakukan proses rekombinasi adalah sebagai berikut :

$$ChromTMP = Recomb('TypeRec', OldCrom, Pc);$$

dimana,

ChromTMP : Kromosom sementara individu baru.

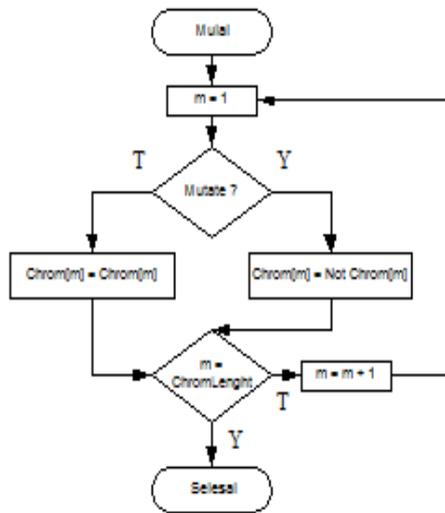
OldChrom :Kromosom Individu generasi sebelumnya

Pc : Probabilitas Rekombinasi.

TypeRec : Jenis rekombinasi yang digunakan.

3.6 Mutasi

Operator Mutasi digunakan untuk mengembalikan materi genetik berharga yang hilang dalam proses rekombinasi. Metode mekanisme Mutasi yang digunakan adalah metode *Flip Bit Mutation* (FBM). Metode FBM sangat tua, digunakan mulai dari generasi John Holland hingga sekarang dan tidak mengalami banyak perubahan, kecuali nilai probabilitas mutasinya. Mekanisme FBM dapat dilihat pada gambar 3.8.



Gambar 3.8 : Mekanisme Flip Bit Mutation (FBM)

Nilai *Mutate* dapat bernilai *True* atau *False* tergantung pada perbandingan nilai P_m dengan suatu bilangan *random* tertentu. Jika lebih besar maka bernilai *True* jika tidak maka sebaliknya. Sintaks yang digunakan adalah sebagai berikut :

$$\text{NewPopChrom} = \text{MutFBM}(\text{ChromTMP}, P_m);$$

dimana,

NewPopChrom: Kromosom individu generasi terbaru
ChromTMP : Kromosom sementara yang dihasilkan pada proses rekombinasi.

P_m : Probabilitas Mutasi.

4. Analisa Dan Pengujian

Analisa hasil pengujian algoritma genetik sebagai metode pencarian solusi optimal dalam makalah ini meliputi beberapa hal yang sangat mendasar, yaitu mengenai unjuk kerja optimal untuk beberapa jenis operator genetik. Operator pertama yang mendapat perhatian disini adalah metode pemilihan induk dan jenis metode operator rekombinasi yang digunakan. Operator seleksi pemilihan induk yang dibandingkan adalah metode SUS (*Stochastic Universal Sampling*) dan RWS (*Roulette Wheel Selection*) sedangkan operator rekombinasi yang dibandingkan adalah SSC (*Single Site Crossover*) dan DSC (*Double Site Crossover*). Kesemua operator ini dibandingkan langsung secara simultan untuk penyelesaian masalah LFC dengan metode lain yaitu dengan menggunakan metode OPP (*Optimal Pole Placement*) yang berbasis teori Ackermann yang ditulis oleh Prof. Hadi Saadat.

4.1 Skenario Pengujian

Setelah semua fungsi yang telah dibuat dapat memberikan hasil simulasi, diperlukan pengujian untuk mengetahui seberapa jauh kebenaran program yang telah dibuat. Uji coba dilakukan untuk mengetahui performansi algoritma

genetik terhadap perubahan nilai-nilai parameternya. Skenario uji coba yang dilakukan yaitu :

1. Model seleksi RWS, model rekombinasi SSC untuk panjang kromosom 30 bit (10 bit/parameter)
2. Model seleksi RWS, model rekombinasi DSC untuk panjang kromosom 30 bit (10 bit/parameter)

Kisaran nilai vektor $K \{k_1, k_2, k_3\}$ dibuat seragam yaitu antara 0 hingga 10 untuk kepresisian 10 bit dan 0 hingga 50 untuk kepresisian 50 bit. Nilai parameter genetik lain dibuat konstan adalah :

1. Generasi Maksimum, bernilai 100 generasi
2. Probabilitas Rekombinasi, bernilai 0.9
3. Probabilitas Mutasi, bernilai 0.7/Panjang kromosom
4. Sela antar generasi (*Generation Gap*) , bernilai 0.9

Parameter-parameter sistem LFC dapat dilihat pada tabel 4.1 berikut :

Tabel 4.1: Parameter LFC (*Load Frequency Control*)

Parameter LFC	Nilai
Konstanta waktu turbin (τ_T)	0.5 sec
Konstanta waktu governor (τ_G)	0.2 sec
Konstanta inersia generator (H)	5 sec
Regulasi Governor (R)	0.05
Perubahan Beban (ΔP)	0.2 p.u
Waktu Simulasi	4 dt
Frekuensi Sistem	60 Hz

4.2. Hasil pengujian LFC Tanpa pengontrolan (*uncompensate*)

Model LFC tanpa menggunakan aksi regulasi sistem umpan balik memiliki fungsi alih berdasarkan persamaan (2.23) dengan nilai $D = 0.8$, sebagai berikut :

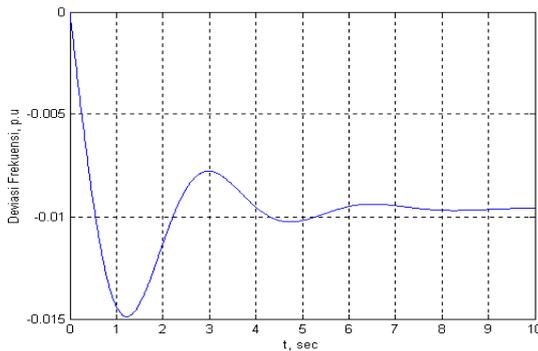
$$\frac{\Delta\Omega(s)}{\Delta P_L(s)} = \frac{-0.1s^2 - 0.7s - 1}{s^3 + 7.08s^2 + 10.56s + 20.8} \quad (4.1)$$

Secara teoritis dengan menggunakan teorema nilai akhir SSE (*Steady State Error*) LFC ini akan memberikan kesalahan keadaan tunak sebesar :

$$\Delta\omega_{ss} = \lim_{s \rightarrow 0} s\Delta\Omega(s) = \frac{1}{D + 1/R} \cdot (-\Delta P_L) = \frac{1}{20.8} (-0.2) = -0.0096 \text{ pu} \quad (4.2)$$

besar penyimpangan frekuensi sistem adalah $\Delta f = -0.0096 \times 60 \text{ Hz} = -0.576 \text{ Hz}$, jadi frekuensi akhir sistem berubah menjadi $60 - 0.576 = 59.42 \text{ Hz}$ untuk perubahan beban sebesar 0.2 p.u.

Tanggapan LFC tanpa menggunakan aksi regulasi terhadap perubahan beban ditunjukkan dalam gambar 4.1 berikut ini :



Gambar 4.1 Tanggapan LFC Uncompensated terhadap perubahan beban

Dari gambar 4.1 di atas diperoleh sistem menuju stabil setelah 6.8 detik dan koefisien ITAE untuk pencuplikan 1/100 diperoleh 9.8061. LFC yang hanya menggunakan regulasi *governor* ternyata memberikan “*Death Band Frequency*” pada frekuensi sistem. Hal ini dikarenakan *governor* memiliki karakteristik aksi regulasi yang tidak mampu mengakumulasi *error* frekuensi, dan aksi *governor* hanya berdasarkan perbandingan perubahan frekuensi terhadap perubahan beban yang konstan. Jadi jika aksi regulasi *governor* diset 5% maka bila terjadi perubahan beban sebesar 0.2 maka besar deviasi frekuensi akan menjadi $0.2 \times 0.05 = 0.01$ p.u

4.2.1. OPP (Optimal Pole Placement).

Metode OPP menghendaki adanya lokasi nilai eigen yang diinginkan pada bidang kompleks (*s-plane*), kemudian akan dicari berapa besar nilai penguatan sistem umpan balik setiap parameter dinamik agar diperoleh tanggapan LFC yang optimal terhadap suatu perubahan beban. Di sini akan digunakan hasil uji coba yang dilakukan oleh Prof. Hadi Saadat. Dalam [1] lokasi nilai eigen yang dikehendaki adalah :

$$\begin{aligned} S_1 &= -2 + j6 \\ S_2 &= -2 - j6 ; \\ S_3 &= -3 \end{aligned}$$

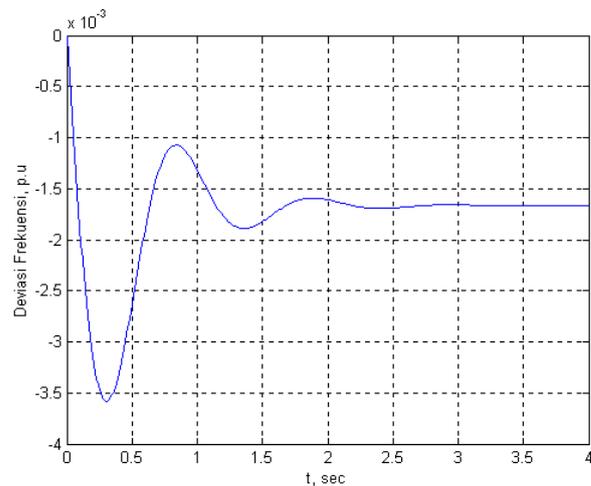
Fungsi alih yang dibentuk oleh ketiga nilai eigen ini seperti persamaan (4.5) :

$$\frac{\Delta\Omega(s)}{-\Delta P_L(s)} = \frac{0.1s^2 + 0.7s + 1}{s^3 + 7s^2 + 52s + 120} \quad (4.5)$$

Kesalahan keadaan tunak untuk fungsi alih (4.5) adalah

$$:\Delta\omega_{ss} = \lim_{s \rightarrow 0} s\Delta\Omega(s) = \frac{1}{120} \cdot (-0.2) = -0.0017 \text{ p.u} \quad (4.6)$$

Tanggapan transien LFC terkompensasi menggunakan OPP terhadap perubahan beban sebesar 0.2 p.u adalah seperti gambar 4.2 :



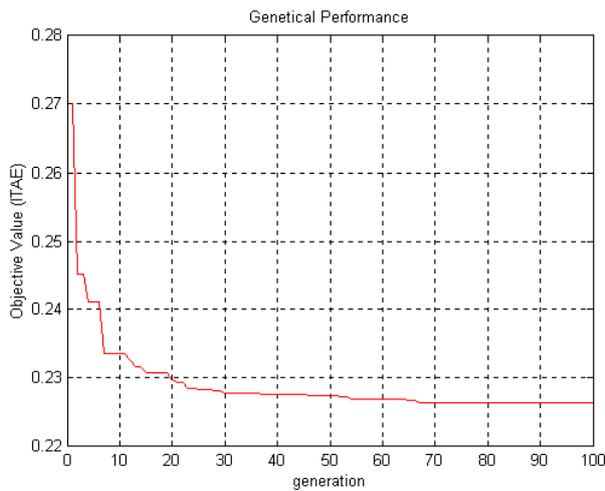
Gambar 4.2 : Respon transien LFC terkompensasi secara OPP

Model kompensasi OPP terhadap LFC memberikan faktor penguatan terhadap variabel dinamik sistem LFC yaitu ΔP_V , ΔP_M , $\Delta\omega$ yaitu secara berturut-turut adalah $K(\Delta P_V) = 4.2$, $K(\Delta P_M) = 0.8$ dan $K(\Delta\omega) = 0.8$. Sistem kompensasi OPP memberikan deviasi frekuensi sistem sebesar $-0.0017 \times 60 \text{ Hz} = -0.102 \text{ Hz}$ dan menyebabkan perubahan frekuensi sistem sebesar $60 \text{ Hz} - 0.102 \text{ Hz} = 59.898 \text{ Hz}$

Indek performansi sistem dihitung menggunakan metode ITAE dengan pencuplikan 1/100 selama periode 4 detik adalah 0.7119 dan mencapai ke kondisi stabil dalam 2.08 detik. Jika dibandingkan dengan model LFC tanpa kompensasi, maka metode OPP memberikan perbaikan terhadap deviasi frekuensi adalah sebesar $\{(59.898-59.42) / 60\} \times 100\% = 0.8\%$ dan kecepatan waktu 4.7 detik lebih cepat dan perbaikan performansi sistem sebesar $9.81 - 0.7119 = 9.81$.

4.2.2. Algoritma Genetik (RWS SSC-10)

Hasil pengujian metode algoritma genetik skenario 1 yaitu pemilihan induk menggunakan metode *Roulette Wheel Selection* (RWS) dan mekanisme rekombinasi menggunakan metode *Single Site Crossover* (SSC), menghasilkan karakteristik pencarian solusi optimal seperti gambar 4.3 dibawah ini :



Gambar 4.3. Karakteristik Pencarian GA/RWS/SSC-10 Dengan menggunakan GA/RWS/SSC-10 perolehan hasil pada generasi pertama yaitu nilai ITAE sama dengan 0.27. Sepuluh generasi pertama proses pencarian sangat efektif, dimana nilai ITAE sistem dapat diperbaiki hingga bernilai 0.234. Generasi 10 hingga generasi 30 mekanisme pencarian solusi masih memberikan hasil signifikan pada performansi sistem yaitu dengan memberikan nilai ITAE sebesar 0.228. Kecenderungan terbentuknya individu dengan kualitas sangat baik mulai terlihat generasi 30 hingga generasi 70 dimana terjadi persaingan sangat ketat antar individu untuk melakukan perkembangbiakan (*breeding*). Hal ini diperlihatkan dengan kenaikan tidak signifikan untuk nilai ITAE sistem pada kisaran generasi 30 hingga 70. Generasi 70 hingga 100 memperlihatkan mekanisme genetik sudah berhasil menghasilkan individu superior yaitu diperlihatkan tidak ditemukan individu lain yang memiliki nilai ITAE lebih baik.

Karakteristik ini diperlihatkan dengan jelas pada gambar 4.3 di atas dimana kurva mengalami saturasi mulai generasi 70 hingga 100. Individu superior ini merupakan keputusan akhir dalam penentuan solusi optimal sistem LFC. Dari individu superior ini dapat diambil struktur kromosomnya untuk memberikan informasi-informasi yang dibutuhkan. Informasi yang dibutuhkan sesuai dengan data struktur yang telah direncanakan. Dari lampiran C diperoleh individu nomer 1 yang terbentuk pada generasi ke-89 merupakan individu superior, dimana mampu bertahan dalam 11 generasi terakhir (89-100). Dari individu ini diperoleh informasi :

1. Nilai ITAE sistem adalah 0.2264
2. Nilai penguatan sistem umpan balik atau vektor K [k1 k2 k3] adalah [10 ,10 ,0]
3. Nilai eigen sistem atau vektor P [s1 s2 s3] adalah [-1.4053 - 9.8278i -1.4053 + 9.8278i -4.2695]
4. Deviasi frekuensi sistem adalah -0.000483 pu

5. Fungsi alih sistem LFC terkompensasi adalah

$$\frac{0.1 s^2 + 0.7 s + 1}{s^3 + 7.08 s^2 + 110.6 s + 420.8}$$

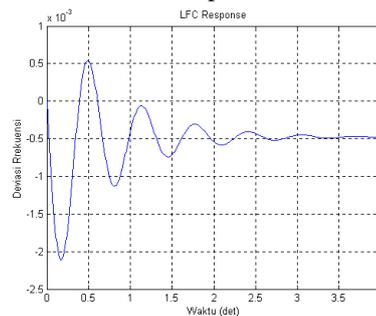
Secara analitis, fungsi alih LFC yang terbentuk adalah :

$$\frac{\Delta\Omega(s)}{-\Delta P_L(s)} = \frac{0.1s^2 + 0.7s + 1}{s^3 + 7.08s^2 + 110.6s + 420.8} \quad (4.7)$$

akan memberikan deviasi frekuensi sistem LFC sebesar :

$$\Delta\omega_{ss} = \lim_{s \rightarrow 0} s\Delta\Omega(s) = \frac{1}{420.8} \bullet (-0.2) = -0.000475 \text{ pu} \quad (4.8)$$

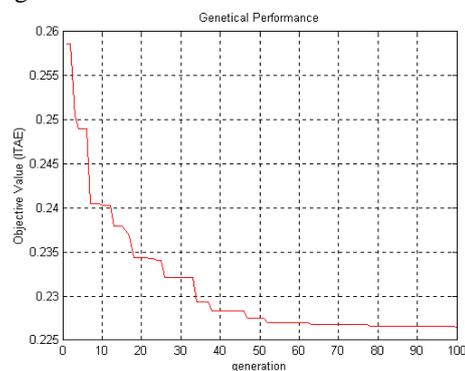
Nilai ini sedikit berbeda dengan yang dihasilkan GA yaitu -0.000483. Hal ini disebabkan GA menentukan nilai deviasi frekuensi berdasarkan nilai terakhir sistem dalam suatu durasi simulasi LFC, sedangkan secara analitis nilai -0.000475 diperoleh pada saat *settling time* yaitu pada detik ke 3.75. Lebih jelasnya lihat gambar 4.4 di bawah ini yang menyatakan tanggapan transien LFC untuk perubahan beban sebesar 0.2 p.u.



Gambar 4.4 : Tanggapan transien individu superior GA RWS SSC-10

4.2.3. Algoritma Genetik (RWS DSC-10)

Hasil pengujian metode algoritma genetik skenario 2 yaitu pemilihan induk menggunakan metode *Roulette Wheel Selection* (RWS) dan mekanisme rekombinasi menggunakan metode *Double Site Crossover* (DSC), menghasilkan karakteristik pencarian solusi optimal seperti gambar 4.5 dibawah ini :



Gambar 4.5. Karakteristik Pencarian GA/RWS/DSC-10 Dengan menggunakan GA/RWS/DSC-10 perolehan hasil pada generasi pertama yaitu nilai ITAE sama dengan

0.2586. Sepuluh generasi pertama proses pencarian sangat efektif, dimana nilai ITAE sistem dapat diperbaiki hingga bernilai 0.2403. Generasi 10 hingga generasi 50 mekanisme pencarian solusi masih memberikan hasil signifikan pada performansi sistem yaitu dengan memberikan nilai ITAE sebesar 0.2275. Kecenderungan terbentuknya individu dengan kualitas sangat baik mulai terlihat generasi 50 hingga generasi 80 dimana terjadi persaingan sangat ketat antar individu untuk melakukan perkembangbiakan (*breeding*). Hal ini diperlihatkan dengan kenaikan tidak signifikan untuk nilai ITAE sistem pada kisaran generasi 50 hingga 80. Dari lampiran C diperoleh individu nomer 33 yang terbentuk pada generasi ke-80 mampu bertahan hingga 4 generasi dengan nilai ITAE sebesar 0.226588, selanjutnya tersingkirkan oleh individu nomer 27 yang terbentuk pada generasi 85 dengan ITAE sebesar 0.226536 dan mampu bertahan hingga ke generasi 99 atau 14 generasi. Tetapi pada generasi ke-100 terbentuk individu baru yang terbaik nomer 21 dengan ITAE sebesar 0.226522. Secara global perbedaan nilai ini tidak begitu signifikan tetapi secara alami harus tetap ditentukan pemenang, maka dari individu superior diperoleh informasi :

1. Nilai ITAE sistem adalah 0.226522
2. Nilai penguatan sistem umpan balik atau vektor K [k1 k2 k3] adalah [10 ,9.99 ,0.068]
3. Nilai eigen sistem atau vektor P [s1 s2 s3] adalah [-1.4027 - 9.8278i -1.4027 + 9.8278i -4.2677]
4. Deviasi frekuensi sistem adalah -0.000483 p.u
5. Fungsi alih sistem LFC terkompensasi adalah

$$\frac{0.1 s^2 + 0.7 s + 1}{s^3 + 7.073 s^2 + 110.5 s + 420.5}$$

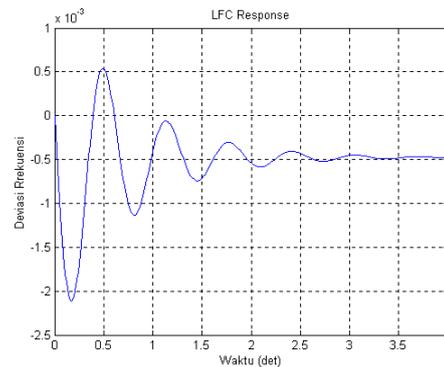
Secara analitis dengan fungsi alih LFC yang terbentuk adalah :

$$\frac{\Delta\Omega(s)}{-\Delta P_L(s)} = \frac{0.1s^2 + 0.7s + 1}{s^3 + 7.08s^2 + 110.6s + 420.5} \quad (4.9)$$

akan memberikan deviasi frekuensi sistem LFC sebesar

$$\Delta\omega_{ss} = \lim_{s \rightarrow 0} s\Delta\Omega(s) = \frac{1}{420.5} \cdot (-0.2) = -0.000476 \text{ pu} \quad (4.10)$$

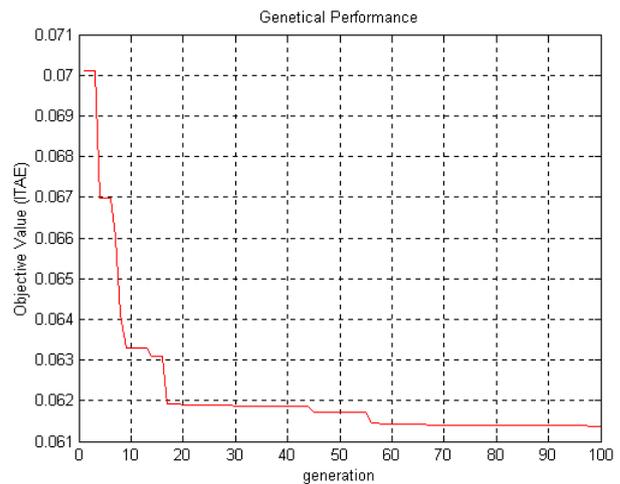
Nilai ini sedikit berbeda dengan yang dihasilkan GA yaitu -0.000483. Hal ini disebabkan GA menentukan nilai deviasi frekuensi berdasarkan nilai terakhir sistem dalam suatu durasi simulasi LFC, sedangkan secara analitis nilai -0.000476 diperoleh pada saat *settling time* yaitu pada detik ke 3.75. Lebih jelasnya lihat gambar 4.6 di bawah ini yang menyatakan tanggapan transien LFC untuk perubahan beban sebesar 0.2 p.u.



Gambar 4.6 : Tanggapan transien individu superior GA RWS DSC-10

4.2.4 Algoritma Genetik (RWS SSC-50)

Hasil pengujian metode algoritma genetik skenario 3 yaitu pemilihan induk menggunakan metode *Roulette Wheel Selection* (RWS) dan mekanisme rekombinasi menggunakan metode *Single Site Crossover* (SSC), menghasilkan karakteristik pencarian solusi optimal seperti gambar 4.7 di bawah ini :



Gambar 4.7. Karakteristik Pencarian GA/RWS/SSC-50

Dengan menggunakan GA/RWS/SSC-50 perolehan hasil pada generasi pertama yaitu nilai ITAE sama dengan 0.070098. Sepuluh generasi pertama proses pencarian sangat efektif, dimana nilai ITAE sistem dapat diperbaiki hingga bernilai 0.06331. Generasi 10 hingga 20 pencarian tetap memberikan hasil yang optimal yaitu memberikan nilai ITAE sebesar 0.06189. Generasi 20 hingga generasi 40 mekanisme pencarian solusi memberikan hasil yang tidak signifikan dan cenderung stagnan pada nilai ITAE = 0.06187. Generasi 40 hingga 60 proses genetik masih tetap berhasil mengasilkan individu yang berkualitas sangat baik dengan ditemukannya nilai ITAE sebesar 0.06141.

Kecenderungan terbentuknya individu dengan kualitas sangat baik mulai terlihat pada generasi 60 hingga generasi 90 dimana terjadi persaingan sangat ketat antar individu untuk melakukan perkembangbiakan (*breeding*). Hal ini diperlihatkan dengan kenaikan tidak

signifikan untuk nilai ITAE sistem pada kisaran generasi 60 hingga 90. Dari lampiran C diperoleh informasi bahwa mulai dari generasi 60 hingga 90 merupakan generasi yang berkualitas. Hal ini diperlihatkan dengan banyaknya individu berlainan yang memiliki nilai ITAE hampir sama dalam kisaran tersebut. Sekali lagi kita tetap harus menentukan individu pemenang, Individu pemenang adalah individu nomer 40 yang terbentuk pada generasi 98,99 dan 100. Informasi yang dapat diperoleh yaitu :

1. Nilai penguatan sistem umpan balik atau vektor K [k1 k2 k3] adalah [49.99 ,49.795 ,0.011]
2. Nilai eigen sistem atau vektor P [s1 s2 s3] adalah [-1.5158 +22.2704i, -1.5158 -22.2704i, -4.0473]
3. Deviasi frekuensi sistem adalah -0.000101 p.u
4. Fungsi alih sistem LFC terkompensasi adalah

$$\frac{0.1 s^2 + 0.7 s + 1}{s^3 + 7.079 s^2 + 510.5 s + 2017}$$

Secara analitis dengan fungsi alih LFC yang terbentuk adalah :

$$\frac{\Delta\Omega(s)}{-\Delta P_L(s)} = \frac{0.1s^2 + 0.7s + 1}{s^3 + 7.079s^2 + 510.5s + 2017} \quad (4.11)$$

akan memberikan deviasi frekuensi sistem LFC sebesar :

$$\Delta\omega_{ss} = \lim_{s \rightarrow 0} s\Delta\Omega(s) = \frac{1}{2017} \bullet (-0.2) = -0.000099 \text{ pu} \quad (4.12)$$

Nilai ini sedikit berbeda dengan yang dihasilkan GA yaitu -0.000101. Hal ini disebabkan GA menentukan nilai deviasi frekuensi berdasarkan nilai terakhir sistem dalam suatu durasi simulasi LFC, sedangkan secara analitis nilai -0.000099 diperoleh pada saat *settling time* yaitu pada detik ke 4.0. Lebih jelasnya lihat gambar 4.8 di bawah ini yang menyatakan tanggapan transien LFC untuk perubahan beban sebesar 0.2 p.u

5. Kesimpulan

Dari uji coba, analisa, dan pembahasan metode algoritma genetik sebagai metode pencarian solusi optimal untuk penyelesaian masalah kontrol optimal stabilitas frekuensi sistem tenaga (LFC) dengan pola pendekatan regulasi faktor penguatan sistem umpan balik statik, dengan berbagai skenario model algoritma genetik yang digunakan dapat disimpulkan sebagai berikut :

1. Algoritma genetik secara umum mampu memperbaiki unjuk kerja sistem LFC secara optimal, dengan memberikan deviasi frekuensi sistem tenaga listrik hampir mendekati nol.
2. Algoritma genetik dimana operator rekombinasi menggunakan metode *Single Site Crossover* (SSC) memberikan hasil akhir lebih baik dibandingkan apabila operator rekombinasi

menggunakan metode *Double Site Crossover* (DSC) untuk semua skenario pengujian baik model 10 bit atau 50 bit.

3. Operator rekombinasi dengan metode *Double Site Crossover* (DSC) untuk model pemilihan induk *Roulette Wheel Selection* (RWS) lebih banyak menciptakan keragaman jenis individu dalam tiap generasi dibandingkan operator rekombinasi menggunakan metode SSC.
4. Operator rekombinasi dengan metode *Single Site Crossover* (SSC) untuk model pemilihan induk *Stochastic Universal Sampling* (SUS) lebih banyak menciptakan keragaman jenis individu dalam tiap generasi dibandingkan operator rekombinasi menggunakan metode DSC.
5. Batasan penguatan vektor penguatan (K) antara 0 hingga 50 memberikan deviasi frekuensi pada sistem LFC lebih kecil dibandingkan dengan batasan penguatan vektor K 0 hingga 10.
6. Sistem LFC tak terkompensasi (*Uncompensated*) memiliki deviasi frekuensi lebih besar dibandingkan kontrol optimal sistem LFC menggunakan metode *Optimal Pole Placement* (OPP) yang dikembangkan oleh Prof. Hadi Saadat.

Aksi kontrol menggunakan aksi regulasi *governor* memberikan deviasi frekuensi sebanding dengan persentase regulasi *governor* yang digunakan.

Daftar Pustaka

- [1]. Hadi Saadat, *Power System Analysis*, Mc Graw Hill Series in Electrical and Computer Engineering
- [2]. Katsuhiko Ogata, *Teknik Kontrol Automatik*, Jilid I, Penerbit Erlangga, 1991.
- [3]. David Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison Wesley Publishing Company Inc
- [4]. Lawrence Davis, *Handbook of Genetic Algorithms*, Van Nostrand Reinhold, New York, 1991
- [5]. Jack Golten & Andy Verwer, *Control System Design and Simulation*, Mc. Graw Hill, 1991
- [6]. LeevyDano Malik, *Optimasi Base Point Pembangkit Thermal Sistem Tenaga Listrik Multi Area via Algoritma Genetik*, Tugas Akhir, 2000, UNDIP
- [7]. G.N. Taranto & D>M> Falcao, *A Genetic Based Control Design for Damping Power System Inter-Area Oscillations*, Proceedings of the 35th IEEE Conference on Decision and Control, Kobe, Japan, 1996
- [8]. Camila Paes Salomon¹, Maurílio Pereira Coutinho¹, Carlos Henrique Valério de Moraes¹, Luiz Eduardo Borges da Silva¹ Germano Lambert-Torres¹ and Alexandre Rasi Aoki², *Applications of Genetic Algorithm in Power System Control Centers*, IUNIFEI - Itajuba Federal University 2LACTEC – Institute of Technology for Development Brazil, 2012