

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S1
KONSENTRASI TEKNIK ENERGI LISTRIK



**OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE
KOMBINASI *EVOLUTIONARY PROGRAMMING-
EVOLUTIONARY STRATEGY-GENETIC ALGORITHM* PADA
SALURAN TRANSMISI 150 KV SUB-SISTEM PAITON-BALI**

SKRIPSI

Disusun Oleh :

FUD NASPUDDIN

01.12.159

SEPTEMBER 2008

LEMBAR PERSETUJUAN

**OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE
KOMBINASI EVOLUTIONARY PROGRAMMING-
EVOLUTIONARY STRATEGY-GENETIC ALGORITHM PADA
SALURAN TRANSMISI 150 kV SUB-SISTEM PAITON-BALI**

SKRIPSI

*Disusun dan Diajukan Untuk Melengkapi dan Memenuhi Syarat-syarat
Guna Mencapai Gelar Sarjana Teknik*

**Disusun Oleh :
FUD NASPUDDIN
Nim. 01.12.159**

Malang, September 2008

**Mengetahui,
Ketua Jurusan Teknik Elektro**

Ir. F. YUDI LIMPRAPTONO, MT.
NIP. Y. 1039500274

**Diperiksa dan Disetujui :
Dosen Pembimbing I**

Dosen Pembimbing II

Ir. H. CHOIRUL SALEH, MT.
NIP. Y. 1018800190

Ir. DJOJO PRATMONO, MT.
NIP. Y. 1018500107

**INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK**

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadirat ALLAH SWT, atas rahmat dan karunia-Nya memungkinkan penulis dapat menyelesaikan penelitian yang berjudul: **“Optimasi Daya Reaktif Menggunakan Metode Kombinasi *Evolutionary Programming-Evolutionary Strategy-Genetik Algorithm* pada Saluran Transmisi 150 kv SUB-Sistem Paiton-Bali ”.**

Penelitian ini bertujuan untuk memenuhi kurikulum akademik yang harus ditempuh oleh setiap mahasiswa ITN Malang guna menyelesaikan pendidikan pada jenjang strata satu jurusan Teknik Elektro konsentrasi Teknik Energi Listrik di Institut Teknologi Nasional Malang.

Atas segala bimbingan, pengarahan dan bantuan yang diberikan, sehingga tersusunnya laporan penelitian ini, maka penulis menyampaikan terima kasih kepada :

1. **Bapak DR. Ir. Abraham Lomi, MSEE** selaku Rector ITN Malang
2. **Ir. Mochtar Asroni, MSME**, selaku Dekan FTI ITN Malang.
3. **Ir. F Yudi Limpraptono, MT**, selaku Ketua Jurusan Teknik Elektro (S1) ITN Malang dan selaku dosen wali saya.
4. **Yusuf Ismail Nakhoda, ST, MT**, selaku Sekretaris Jurusan Teknik Elektro (S1) ITN Malang.
5. **Ir. H. Choirul Saleh, MT**, selaku dosen pembimbing I.
6. **Ir. Djojo Priatmono, MT**, selaku dosen pembimbing II.
7. **Ir. Bambang Priohartono, ST, MT**, selaku dosen penguji I.

DAFTAR ISI

Halaman :

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	xii
DAFTAR GRAFIK	xv
DAFTAR TABEL	xvi
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan.....	3
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian	3
1.6. Sistematika Pembahasan.....	4
1.7. Kontribusi Penelitian.....	5
BAB II SISTEM TENAGA LISTRIK, OPERASI EKONOMIS DAN KARAKTERISTIK UNIT PEMBANGKIT	6
2.1. Sistem Tenaga Listrik.....	6
2.2. Saluran Transmisi.....	7

2.2.1. Saluran Transmisi Pendek	8
2.2.2. Saluran Transmisi Menengah	8
2.2.3. Saluran Transmisi Panjang	10
2.3. Daya Dalam Sistem Tenaga Listrik	11
2.3.1. Daya Aktif (<i>Active Power</i>)	11
2.3.2. Daya Reaktif (<i>Reactive Power</i>)	11
2.3.3. Daya Semu (<i>Apparent Power</i>)	12
2.4. Sistem Per- Unit	13
2.5. Mengubah Dasar Sistem Per-Unit	14
2.6. Sistem Operasi Pada Sistem Tenaga Listrik	15
2.7. Karakteristik Unit Pembangkit	18
2.7.1. Karakteristik <i>Input-Output</i>	18
2.7.2. Karakteristik <i>Heat-Rate</i>	20
2.7.3. Karakteristik <i>Incremental Heat-Rate dan Incremental Fuel Cost</i>	21
2.8. Ekonomi Dispatch	22
2.8.1. Fungsi Bahan Bakar	23
2.8.2. Fungsi Biaya Daya Reaktif Pembangkit	24
2.8.3. Economic Dispatch dengan Mengabaikan Rugi-Rugi Transmisi	24
2.8.4. Economic Dispatch dengan Memperhitungkan Rugi-Rugi Transmisi	27

BAB III OPTIMASI DAYA REKTIF MENGGUNAKAN METODE	
KOMBINASI <i>EVOLUTIONARY PROGRAMMING-</i>	
<i>EVOLUTIONARY STRATEGY-GENETIC ALGORITHM</i>	29
3.1. Analisa Aliran Daya	29
3.1.1. Klasifikasi Bus	30
3.1.2. Metode Newton Raphson.....	31
3.2. Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-</i>	
<i>Genetika Algoritma</i>	34
3.2.1. Metode <i>Evolutionary Programming</i> (EP)	34
3.2.2. <i>Genetic Alorithm</i>	35
3.2.3. <i>Evolutionary Strategy</i>	37
3.3. Parameter Kombinasi <i>Evolutionary Programming-Evolutionary</i>	
<i>Strategy-Genetika Algoritma</i>	39
3.3.1. Jumlah Generasi (MAXGEN).....	39
3.3.2. Ukuran Populasi (POPZISE).....	40
3.3.3. Probabilitas Ceosover (PC).....	40
3.3.4. Probabilitas Mutasi (Pm).....	41
3.3.5. Panjang Kromosom (NVAR).....	41
3.4. Mekanisme Kombinasi <i>Evolutionary Programming-Evolutionary</i>	
<i>Strategy-Genetika Algoritma</i>	42
3.4.1. Pengkodean atau Representasi	42
3.4.2. Fungsi <i>Fitness</i> (Fungsi Evaluasi)	43
3.4.3. Seleksi dan Reproduksi	43

3.4.3.1. <i>Reulette Wheel Selection</i>	43
3.4.3.2. <i>Rank Selektion</i>	44
3.4.4. <i>Crossover</i> (Pindah Silang).....	45
3.4.5. <i>Mutation</i> (<i>Mutasi</i>).....	47
3.4.6. <i>Recombination</i> (<i>Rekombinasi</i>)	48
3.4.7 <i>Competition</i> (<i>Kompetisi</i>)	49
3.4.8. <i>Elitism</i>	49
3.5. <i>Formulasi Masalah Optimasi Daya Reaktif</i>	51
3.6. <i>Adaptasi Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i> ke Masalah Optimasi Daya Reaktif</i>	53
3.6.1. <i>Representasi Solusi</i>	53
3.6.2. <i>Initial Population Generation</i> (<i>Inisial Populasi Generasi</i>).....	53
3.6.3. <i>Evaluasi Fitnes</i>	53
3.6.4. <i>Selektion dan Reproduksi</i>	54
3.6.5. <i>Crossover</i>	54
3.6.6. <i>Statistik</i>	55
3.6.7. <i>Mutation</i>	55
3.6.7.1. <i>Evolutionary Programming</i>	55
3.6.7.2. <i>Genetik Algorithm</i>	56
3.6.7.3. <i>Evolutionary Strategy</i>	57
3.6.8. <i>Competition</i> (<i>Kompetisi</i>).....	57
3.6.9. <i>onvergence Test</i> (<i>Konvergensi</i>).....	58

BAB IV HASIL DAN ANALISA HASIL.....	59
4.1. Program Komputer Optimisasi Daya Reaktif Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i>	59
4.2. Algoritma Program.....	59
4.2.1. Algoritma Program Dasar Permasalahan.....	59
4.2.2. Algoritma Program Perencanaan Optimisasi Daya Reaktif Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i>	60
4.2.3. Algoritma Aliran Daya Newton Raphson.....	61
4.3. Hasil Validasi IEEE 30 Dengan Menggunakan Metode Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i>	65
4.4. Data Pembangkitan Thermal pada Sub-Sistem Paiton Bali.....	68
4.5. Data Pembangkitan dan Pembebanan 150 kV Sub-Sistem Paiton-Bali.	72
4.6. Data Saluran Transmisi 150 kV Sub Sistem Paiton-Bali.....	73
4.7. Prosedur Pelaksanaan Program Perhitungan.....	75
4.8. Hasil dan Analisa Hasil Perhitungan Optimisasi Daya Reaktif Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i> Pada Saluran Transmisi 150 kV Sub-Sistem Paiton Bali.....	82
4.8.1. Hasil Perhitungan Sebelum Optimasi.....	82
4.8.2. Hasil Perhitungan Setelah Optimasi.....	85

4.9.	Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Optimisasi Daya Reaktif Kombinasi <i>Evolutionary Programming-Evolutionary Strategy-Genetika Algoritma</i>	88
4.9.1.	Tingkat Optimum Biaya Pembangkitan Sebelum dan Sesudah Optimasi	88
4.9.2.	Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus Sebelum dan Sesudah Optimasi	89
4.9.3.	Perbandingan Tingkat Rugi-Rugi Daya pada Saluran Sebelum dan Sesudah Optimasi.....	90
BAB V	KESIMPULAN	92
5.1.	Kesimpulan.....	92
5.2.	Saran – saran.....	93
	DAFTAR PUSTAKA	94
	LAMPIRAN	

DAFTAR GAMBAR

Halaman:

Gambar 2.1. Rangkaian Setara Saluran transmisi	7
Gambar 2.2. Rangkaian Setara Saluran Transmisi Pendek	8
Gambar 2.3. Rangkaian Setara Saluran Transmisi Menengah.....	9
Gambar 2.4. Diagram Skema Saluran Transmisi Panjang	10
Gambar 2.5. Representasi Segitiga Daya	12
Gambar 2.6. Unit Boiler-Turbin-Generator	19
Gambar 2.7. Kurva Karakteristik <i>Input-Output</i> Pembangkit Thermal.....	20
Gambar 2.8. Kurva Karakteristik <i>Heat-Rate</i> Unit Pembangkit.....	21
Gambar 2.9. Kurva Karakteristik Incremental <i>Heat-Rate/Fuel Cost</i>	22
Gambar 2.10. N Unit Pembangkit Thermal Melayani Beban P_R	25
Gambar 2.11. N Unit Pembangkit Thermal Melayani Beban P_R Melalui Saluran Transmisi	28
Gambar 3.1. Jumlah Generasi	39
Gambar 3.2. Ukuran Populasi	40
Gambar 3.3. Pengkodean Representasi	42
Gambar 3.4. <i>Reulette Wheel</i>	44
Gambar 3.5. Ilustrasi Operator Dengan <i>One Point Crossover</i>	46
Gambar 3.6. Ilustrasi Operator Dengan <i>Two Point Crossover</i>	46
Gambar 3.7. Ilustasi Operator <i>Crossover</i> Dengan <i>Uniform Crossover</i>	46
Gambar 3.8. Ilustasi Operator Mutasi	47

Gambar 3.9. Pembentukan Next Generation Dalam Algoritma Genetika	50
Gambar 4.1. <i>Flowchart</i> Algoritma Peyelesaian masalah.....	62
Gambar 4.2. <i>FlowChart</i> Program Menggunakan Metode Kombinasi <i>Evolutionary Programming-Evolutionary Strategi-Genetic Algorithm</i>	63
Gambar 4.3. <i>Flowchart</i> Perhitungan Aliran Daya <i>Newton Raphson</i>	64
Gambar 4.4. Tampilan Parameter Kombinasi <i>EP-ES-GA</i> Validasi IEEE 30.....	65
Gambar 4.5. Hasil <i>Load Flow</i>	65
Gambar 4.6. Tampilan Hasil Validasi IEEE 30 Hasil Perhitungan Biaya Pembangkit.....	66
Gambar 4.7. Tampilan Hasil <i>Summary Load Flow</i>	66
Gambar 4.8. Diagram Segaris Jaringan Sistem Tenaga Listrik 150 kV Sub- Sistem Paiton-Bali.....	70
Gambar 4.9. Tampilan Utama Program	75
Gambar 4.10. Tampilan Masukan Data	75
Gambar 4.11. Tampilan Data Bus.....	76
Gambar 4.12. Tampilan Data Saluran.....	76
Gambar 4.13. Tampilan Data Generator.....	77
Gambar 4.14. TampilanParamete Kombinasi <i>EP-ES-GA</i> pada Kondisi Awal (Sebelum Optimasi).....	77
Gambar 4.15. Tampilan Hasil <i>Load Flow</i> pada Kondisi Awal (Sebelum Optimasi).....	78

DAFTAR GRAFIK

Halaman:

Grafik 4.1. Hasil Perhitungan Daya yang Dibangkitkan Sebelum Optimasi.....	84
Grafik 4.2. Hasil Perhitungan Daya yang Dibangkitkan Setelah Optimasi.....	87
Grafik 4.3. Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Total Sebelum dan Setelah Optimasi	91

DAFTAR TABEL

Halaman:

Tabel 4.1.	Perbandingan Hasil Data Referensi Jurnal dengan Data Optimasi	66
Tabel 4.2.	Parameter Unit Pembangkit Thermal.....	68
Tabel 4.3.	Persamaan Biaya Pembangkitan Unit Pembangkit Thermal Paiton dan Bali.....	69
Tabel 4.4.	Penomoran Bus 150 kV Sub-Sistem Paiton-Bali.....	71
Tabel 4.5.	Data Pembangkitan dan Pembebanan 150 kV Sub-Sistem Paiton-Bali	72
Tabel 4.6.	Data Saluran Transmisi 150 kV Sub-Sistem Paiton-Bali	73
Tabel 4.7.	Hasil Perhitungan Tegangan, Sudut Teganagan, Pembangkitan dan Pembebanan Sebelum Optimasi	82
Tabel 4.8.	Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi	83
Tabel 4.9.	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum Optimasi	84
Tabel 4.10.	Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Sebelum Optimasi	84
Tabel 4.11.	Hasil Perhitungan Tegangan, Sudut Teganagan, Pembangkitan dan Pembebanan Setelah Optimasi	85
Tabel 4.12.	Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi.....	86
Tabel 4.13.	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi	87
Tabel 4.14.	Hasil Perhitungan Daya Yang Dibangkitkan Dan Biaya Operasi Setelah Optimasi.....	87

BAB I

PENDAHULUAN

1.1. Latar Belakang

Adanya persoalan dalam menghadapi kebutuhan daya listrik yang tidak tetap dari waktu ke waktu, sehingga menimbulkan permasalahan yaitu bagaimana mengoperasikan suatu sistem tenaga listrik yang selalu dapat memenuhi permintaan daya pada setiap saat, dengan kualitas baik dan harga yang murah.

Variasi tegangan yang besar pada bus akibat dari variasi pembebanan daya yang besar menyebabkan perubahan aliran daya pada setiap saluran. Dari perubahan aliran daya yang besar menimbulkan rugi-rugi daya yang besar pula. Oleh karena itu pada suatu operasi pada beban tertentu, perhitungan ekonomis harus tetap merupakan suatu prioritas atau nilai yang harus diperhitungkan disamping hal-hal lain sehingga nantinya diperlukan suatu rencana operasi yang optimum dengan tetap memenuhi beberapa persyaratan pengoperasian sistem tenaga listrik yaitu antara lain : daya yang dibangkitkan cukup untuk memasok beban dan rugi-rugi daya pada saluran transmisi, tegangan bus sesuai dengan ratingnya serta tidak adanya pembebanan lebih pada unit-unit pembangkit yang beroperasi.

Alokasi daya reaktif yang tepat dalam sistem bisa menghasilkan kondisi optimum, yaitu kondisi dengan rugi-rugi transmisi yang minimum dengan biaya investasi yang murah, sehingga mampu memperbaiki profil tegangan sistem.

Permasalahan besarnya rugi-rugi di saluran telah menjadi perhatian utama khususnya pada saluran transmisi 150 KV.

Masalah-masalah diatas dapat diatasi dan dipecahkan salah satunya dengan menggunakan metode kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm*. Dengan metode kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm* ini akan mengoptimalkan penyaluran daya reaktif untuk mengurangi rugi daya dengan meminimalkan biaya pengoperasian pada jaringan transmisi 150 kV Sub-Sistem Paiton-Bali, sehingga di dapat kualitas suatu sistem tenaga listrik yang baik.

1.2. Rumusan Masalah

Variasi tegangan yang besar pada bus akibat dari variasi pembebanan daya yang besar menyebabkan perubahan aliran daya pada setiap saluran. Dari perubahan aliran daya yang besar menimbulkan rugi-rugi daya yang besar pula. Mengingat pentingnya kestabilan pelayanan sistem tenaga listrik tersebut, maka dapat timbul permasalahan sebagai berikut:

Bagaimana mengoptimalkan penyaluran daya reaktif secara ekonomis dan mengurangi rugi-rugi daya yang terjadi pada saluran transmisi 150 kV Sub-Sistem Paiton-Bali dengan menggunakan metode kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm*.

1.3 Tujuan

Tujuan dari penelitian skripsi ini adalah untuk memperbaiki profil tegangan dan mengurangi rugi-rugi daya dengan biaya pengoperasian yang murah pada Saluran Transmisi 150 kV Sub-Sistem Paiton-Bali dengan cara mengoptimalkan biaya operasional dengan menggunakan metode kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm*.

1.4. Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan, maka pembahasan skripsi ini dibatasi sebagai berikut:

1. Analisis perhitungan optimasi daya reaktif pada saluran transmisi 150 kV Sub-Sistem Paiton-Bali menggunakan metode kombinasi *Evolutionary Programming – Evolutionary Strategy – Genetic Algorithm*
2. Analisa dilakukan dengan asumsi bahwa sistem berada dalam operasi normal.
3. Analisa perhitungan pada pembangkit thermal saja.
4. Tidak membahas masalah peralatan kompensasi dan penempatannya
5. Tidak membahas masalah kontrol tegangan.

1.5. Metodologi Penelitian

Metode yang digunakan dalam pembahasan skripsi ini dilakukan dengan langkah-langkah sebagai berikut:

1. Studi Literatur
-

Yaitu kajian pustaka yang mempelajari teori-teori yang terkait melalui literatur yang telah ada, yang berhubungan dengan permasalahan.

2. Pengumpulan data lapangan yang dipakai dalam obyek penelitian yakni data impedansi saluran transmisi dan data pembebanan saluran transmisi 150 kV Sub-Sistem Paiton-Bali serta data pembangkit termal.
3. Merancang perangkat lunak (*software*) dengan menggunakan metode kombinasi *Evolutionary Programming – Evolutionary Strategy – Genetic Algorithm*
4. Simulasi dan pembahasan masalah
Simulasi dan pembahasan masalah dengan menggunakan bahasa pemrograman Borland Delphi versi 07.
5. Menarik kesimpulan.

1.6. Sistematika Penulisan

Adapun sistematika pembahasan pada skripsi ini adalah sebagai berikut :

BAB I : PENDAHULUAN

Menguraikan latar belakang, rumusan masalah, tujuan penelitian, batasan masalah, metodologi penulisan, sistematika penulisan dan kontribusi penelitian.

BAB II : SISTEM TENAGA LISTRIK, OPERASI EKONOMIS, DAN KARAKTERISTIK UNIT PEMBANGKIT

Menguraikan pembahasan sistem tenaga listrik dan *economic dispatch* secara umum, dan *reactive power dispatch*

BAB II
SISTEM TENAGA LISTRIK, OPERASI EKONOMIS DAN
KARAKTERISTIK UNIT PEMBANGKIT

2.1 Sistem Tenaga Listrik¹³¹

Tiga bagian utama sistem tenaga listrik yaitu : pusat pembangkit tenaga listrik, saluran transmisi dan sistem distribusi yang berhubungan langsung dengan konsumen. Saluran transmisi merupakan penghubung antara pusat pembangkit melalui hubungan antar sistem yang menuju sistem pada sistem yang lain.

Saluran transmisi mempunyai empat parameter yang mempengaruhi kemampuannya dalam menyalurkan daya listrik. Keempat parameter tersebut yaitu : resistansi (R), induktansi (L), kapasitansi (C), serta konduktansi (G).

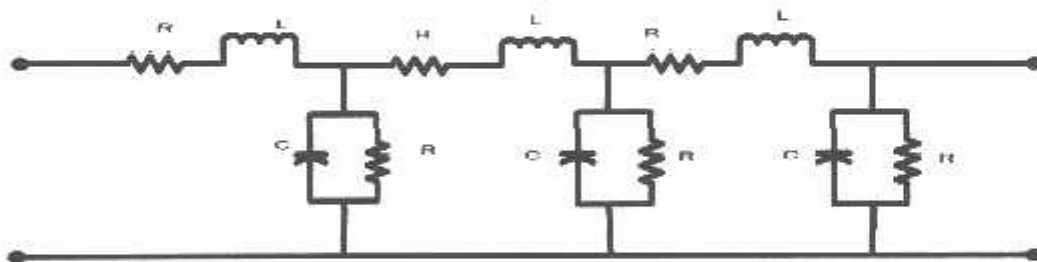
Resistansi umumnya tergantung pada jenis penghantar sedangkan konduktansi menyatakan besarnya arus bocor antar penghantar, antar penghantar dengan tanah, tetapi harganya relatif kecil maka dapat diabaikan. Induktansi adalah parameter rangkaian yang menghubungkan tegangan yang diimbaskan oleh perubahan fluksi akibat perubahan arus, sedangkan kapasitansi suatu saluran transmisi timbul akibat adanya beda potensial antara penghantar dengan tanah, dalam hal ini kapasitansi menyebabkan penghantar bermuatan seperti yang terjadi pada pelat kapasitor.

Impedansi seri terbentuk dari resistansi dan induktansi yang terbagi merata sepanjang saluran. Sedangkan konduktansi dan kapasitansi terdapat diantara

penghantar-penghantar dari saluran fasa tunggal atau diantara penghantar dengan netral dari suatu saluran berfasa tiga membentuk admitansi paralel.

2.2 Saluran Transmisi

Tenaga listrik yang dibangkitkan disalurkan melalui saluran transmisi. Saluran-saluran transmisi ini membawa tenaga listrik dari pusat-pusat tenaga listrik ke pusat-pusat beban. Suatu saluran transmisi tenaga listrik mempunyai empat parameter yang mempengaruhi kemampuan untuk berfungsi sebagai bagian dari sistem tenaga, yaitu resistansi, induktansi, kapasitansi, dan konduktansi. Keempat parameter saluran transmisi tersebut merata disepanjang saluran transmisi. Parameter-parameter tersebut sangat berpengaruh terhadap tegangan bus dan aliran daya yang mengalir pada saluran tersebut.



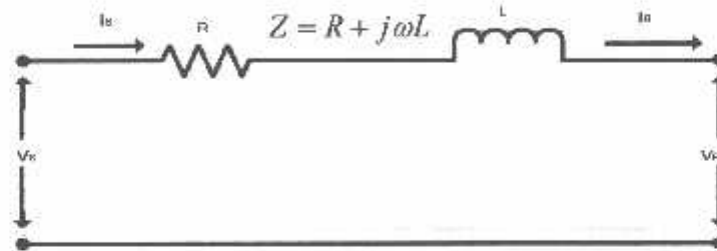
Gambar 2.1^[3]
Rangkaian Setara Saluran Transmisi

Menurut panjangnya, saluran transmisi dapat dikasifikasikan menjadi 3 golongan, yaitu:

1. Saluran transmisi pendek, adalah saluran yang panjangnya < 80 km.
2. Saluran transmisi menengah, adalah saluran yang panjangnya 80-240 km.
3. Saluran transmisi panjang, adalah saluran yang panjangnya > 240 km.

2.2.1 Saluran Transmisi Pendek

Rangkaian ekivalen suatu saluran transmisi pendek diperlihatkan pada gambar 2.2 dimana I_S dan I_R merupakan arus pada ujung pengiriman dan ujung penerimaan, sedangkan V_S dan V_R adalah tegangan saluran terhadap netral pada ujung pengiriman dan ujung penerimaan.



Gambar 2.2^[3]
Rangkaian Setara Saluran Transmisi Pendek

Karena tidak terdapat cabang paralel (*shunt*), arus pada ujung-ujung pengirim dan penerima akan sama besarnya^[3] :

$$I_S = I_R \dots\dots\dots (2.1)$$

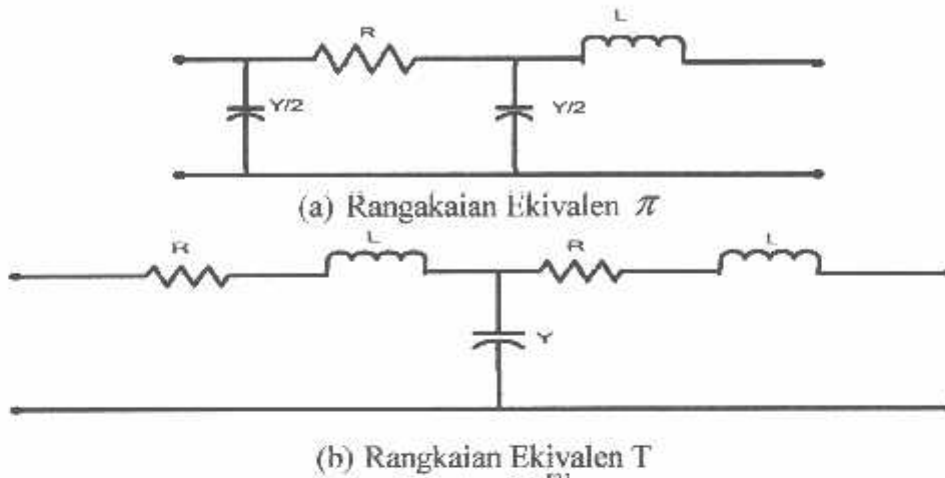
Bila kondisi tegangan pada ujung penerima diketahui, maka tegangan pada ujung sisi kirim adalah^[3] :

$$V_S = V_R + I_R Z \dots\dots\dots (2.2)$$

2.2.2 Saluran Transmisi Menengah

Bertambahnya saluran menyebabkan kapasitansi shunt bertambah besar dan tidak dapat diabaikan. Saluran transmisi jarak menengah pada umumnya digambarkan dengan rangkaian π pada atau rangkaian T, dapat dilihat pada

gambar 2.3(a) dan gambar 2.3(b). Dari dua versi ini rangkaian π lebih umum dipakai dari pada rangkaian T.



(a) Rangkaian Ekivalen π
 (b) Rangkaian Ekivalen T
 Gambar 2.3^[3]
 Rangkaian Setara Saluran Transmisi Menengah^[3]

Untuk rangkaian π berlaku^[3] :

$$V_s = \left[\frac{ZY}{2} + 1 \right] V_R + Z I_R \dots\dots\dots(2.3.)$$

$$I_s = \left[\frac{ZY}{4} + 1 \right] Y V_R + \left[\frac{ZY}{2} + 1 \right] I_R \dots\dots\dots(2.4.)$$

Untuk rangkaian T berlaku^[3] :

$$V_s = \left[\frac{ZY}{2} + 1 \right] V_R + Z I_R \dots\dots\dots(2.5.)$$

$$I_s = \left[\frac{ZY}{4} + 1 \right] Y V_R + \left[\frac{ZY}{2} + 1 \right] I_R \dots\dots\dots(2.6.)$$

dimana :

V_s dan I_s = Tegangan dan Arus sisi kirim

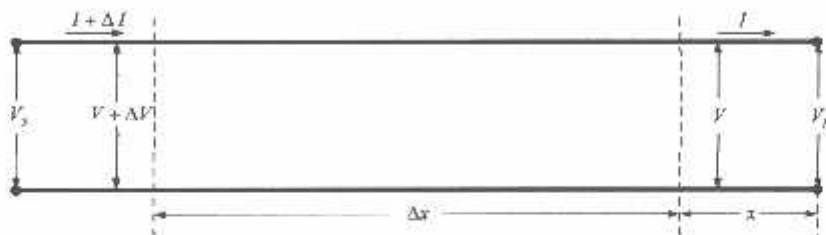
V_R dan I_R = Tegangan dan Arus sisi terima

Z = Impedansi seri total saluran transmisi

Y = Admitansi shunt total saluran

2.2.3 Saluran Transmisi Panjang

Pada saluran panjang parameter-parameter saluran tidak terpusat menjadi satu, melainkan tersebar merata diseluruh panjang saluran, dapat dilihat pada gambar 2.4.



Gambar 2.4^[3]
Diagram Skema Saluran Transmisi Panjang

Persamaan tegangan dan arus pada setiap titik sepanjang saluran transmisi dengan jarak x dari ujung sisi terima dapat ditulis sebagai berikut^[3] :

$$V = \frac{V_R + I_R Z_c}{2} e^{\gamma x} + \frac{V_R - I_R Z_c}{2} e^{-\gamma x} \dots\dots\dots (2.7)$$

$$I = \frac{V_R + I_R}{Z_c} e^{\gamma x} + \frac{V_R - I_R}{Z_c} e^{-\gamma x} \dots\dots\dots (2.8)$$

Persamaan untuk saluran transmisi panjang dapat ditulis dalam bentuk hiperbola sebagai berikut^[3] :

$$V(x) = V_R \cosh \gamma x + I_R Z_c \sinh \gamma x \dots\dots\dots (2.9)$$

$$I(x) = I_R \cosh \gamma x + \frac{V_R}{Z_c} \sinh \gamma x \dots\dots\dots (2.10)$$

dimana :

$$\gamma = \text{Konstanta rambatan pada saluran} = \sqrt{ZY}$$

$$Z_c = \text{Impedansi karakteristik saluran} = \sqrt{\frac{Z}{Y}}$$

2.3 Daya Dalam Sistem Tenaga Listrik^[3]

Dalam sistem tenaga listrik, pembangkit–pembangkit tenaga listrik harus mampu menyediakan tenaga listrik sesuai dengan permintaan beban listrik yang ada. Hal yang harus diperhatikan adalah kondisi sistem yang tetap konstan, dalam hal ini tegangan dan frekwensi harus tetap konstan karena berhubungan dengan daya. Daya listrik yang dibangkitkan dikenal dengan istilah:

2.3.1 Daya aktif (*Active Power*)

Secara umum daya aktif dinyatakan oleh persamaan^[3] :

$$P = [V] [I] \cos \varphi \dots\dots\dots (2.11.)$$

dimana :

V dan I : nilai efektifnya.

P : daya rata–rata yang disebut juga daya aktif.

2.3.2 Daya Reaktif (*Reactive Power*)

Daya reaktif adalah daya yang timbul karena adanya pembentukan medan magnet pada beban–beban induktif (kVAR).

Persamaan daya reaktif adalah^[3]:

$$Q = [V] [I] \sin \varphi \dots\dots\dots (2.12.)$$

Dua macam daya reaktif yaitu : daya reaktif induktif dan daya reaktif kapasitif, dimana keduanya memiliki tanda yang berlawanan. Daya reaktif kapasitif adalah daya yang dibutuhkan oleh kapasitor yang tidak menghasilkan kerja, tetapi tersimpan dalam bentuk energi magnetis atau energi kapasitif.

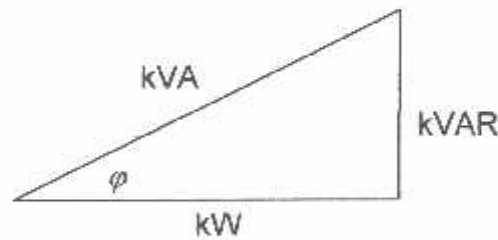
Daya reaktif induktif adalah daya listrik yang dibutuhkan untuk menghasilkan medan magnet yang dibutuhkan oleh alat-alat seperti motor induksi, transformator dan sebagainya.

2.3.3 Daya Semu (*Apparent Power*)

Daya semu merupakan penjumlahan secara vektoris antara daya aktif dan daya reaktif.

Selain diatas dikenal juga istilah faktor daya (*power factor*) adalah perbandingan antara daya aktif dan daya semu, sehingga dapat dirumuskan pada persamaan berikut^[3]:

$$\cos \varphi = \frac{P}{S} = \frac{P}{\sqrt{P^2 + Q^2}} = \frac{kW}{kVA} \dots\dots\dots (2.13)$$



Gambar 2.5^[3]
Representasi Segitiga Daya

dimana :

$\cos \varphi$ = Faktor daya

kW = Daya aktif (P)

kVA = Daya semu (S)

kVAR = Daya reaktif (Q)

2.4 Sistem Per-Unit

Untuk memudahkan proses perhitungan, dalam sistem tenaga listrik digunakan sistem Per-Unit (pu)^[3]:

$$\text{Besaran Per-Unit} = \frac{\text{Besaran sebenarnya}}{\text{Besaran dasar}} \dots\dots\dots(2.14)$$

Rumus-rumus yang digunakan untuk penentuan arus dasar dan impedansi dasar adalah :

➤ Untuk data 1 fasa^[3] :

Arus dasar

$$I_d = \frac{\text{kVA dasar}_{1\phi}}{\text{kV dasar}_{L-N}} \dots\dots\dots(2.15)$$

Impedansi dasar

$$Z_d = \frac{(\text{kV dasar}_{L-N})^2 \times 10^3}{\text{kVA dasar}_{1\phi}} \dots\dots\dots(2.16.)$$

$$Z_d = \frac{(\text{kV dasar}_{L-N})^2}{\text{MVA dasar}_{1\phi}} \dots\dots\dots(2.17.)$$

➤ Untuk data 3 fasa^[3] :

Arus dasar

$$I_d = \frac{\text{kVA dasar}_{3\phi}}{\sqrt{3} \times \text{kV dasar}_{L-L}} \dots\dots\dots(2.18.)$$

Impedansi dasar

$$Z_d = \frac{(\text{kV dasar}_{L-L})^2 \times 10^3}{\text{kVA dasar}_{3\phi}} \dots\dots\dots(2.19.)$$

$$Z_d = \frac{(\text{kV dasar}_{L-L})^2}{\text{MVA dasar}_{3\phi}} \dots\dots\dots(2.20.)$$

Dalam persamaan di atas nilai-nilai besaran diberikan untuk rangkaian satu fasa. Jadi tegangannya adalah tegangan antara fasa dengan tanah dan daya setiap fasa.

Setelah besaran-besaran dasar telah ditentukan maka besaran-besaran itu dinormalisasikan terhadap besaran dasar. Dengan demikian impedansi per satuan dari suatu elemen rangkaian didefinisikan sebagai berikut^[3]:

$$Z = \frac{\text{Impedansi sebenarnya } Z (\Omega)}{\text{Impedansi dasar } Z_d} \dots\dots\dots(2.21.)$$

2.5 Mengubah Dasar Sistem Per-Unit

Kadang-kadang impedansi per-unit untuk suatu komponen dari suatu sistem dinyatakan menurut dasar yang berbeda dengan dasar yang dipilih untuk bagian dari sistem dimana komponen tersebut berada. Karena semua impedansi dalam bagian manapun dari suatu sistem harus dinyatakan dengan dasar impedansi yang sama, maka dalam perhitungannya kita perlu mempunyai cara untuk dapat mengubah impedansi per-unit dari suatu dasar ke dasar yang lain. Dengan mensubstitusikan impedansi dasar yang diberikan dalam persamaan (2.17.) dan (2.20.) ke dalam persamaan (2.21.) maka diperoleh^[3]:

$$Z_u = \frac{(\text{Impedansi sebenarnya, } \Omega) \times (\text{MVA dasar})}{(\text{Tegangan dasar, kV})^2} \dots\dots\dots(2.22)$$

Persamaan (2.22.) memperlihatkan bahwa impedansi per-unit berbanding lurus dengan MVA dasar serta berbanding terbalik dengan kuadrat tegangan

dasar. Untuk mengubah dari impedansi per-unit menurut suatu dasar yang diberikan menjadi impedansi per-unit menurut suatu dasar yang baru, dapat dipakai persamaan berikut^[3] :

$$Z_{\text{baru, per-unit}} = Z_{\text{diberikan, per-unit}} \left(\frac{kV_{\text{diberikan, dasar}}}{kV_{\text{baru, dasar}}} \right)^2 \times \left(\frac{MVA_{\text{baru, dasar}}}{MVA_{\text{diberikan, dasar}}} \right) \dots \dots \dots (2.23)$$

Persamaan ini tidak ada hubungannya dengan pemindahan nilai impedansi dalam ohm dari salah satu sisi transformator ke sisi yang lain.

2.6 Sistem Operasi Pada Sistem Tenaga Listrik^[4]

Seperti telah diketahui bahwa dalam masalah pengaturan beban pada suatu operasi sistem tenaga listrik harus selalu dicapai suatu keadaan operasi yang bisa diandalkan dan cukup ekonomis.

Ada beberapa kerja yang harus dilaksanakan untuk menjamin keandalan sistem operasi antara lain, pengaturan frekuensi dan tegangan sistem untuk berada pada harga normalnya karena adanya perubahan beban pada sistem. Dan seperti yang diketahui dan berulang kali disebutkan bahwa tenaga listrik tidak dapat disimpan sehingga dalam operasinya harus selalu dicapai keseimbangan antara penyediaan dengan pemenuhan kebutuhan daya serta perlu juga diingat bahwa sistem selalu berubah setiap saat. Maka sudah tentu jauh-jauh sebelumnya sudah harus diketahui atau diramalkan keadaan tersebut dengan tepat yaitu keadaan beban pada hari itu dari waktu ke waktu sampai selama 24 jam. Keadaan beban ini digambarkan sebagai kebutuhan daya sebagai fungsi dari waktu yang biasa disebut dengan lengkung beban harian. Lengkung beban harian ini adalah

merupakan sesuatu yang sangat penting disamping karakteristik-karakteristik lainnya sehingga dalam operasi hariannya harus berdasarkan lengkung beban harian yang telah dibuat karena dengan lengkung beban harian ini dapat ditentukan perencanaan operasi pembangkit-pembangkit yang ada, baik itu unit pembangkit *thermal* maupun *hidro*. Tentu saja kebutuhan beban dalam suatu harinya tidak merata akan tetapi dari jam ke jam berbeda sesuai dengan kebutuhan konsumen. Berdasarkan lengkung beban yang telah ada maka dapat ditentukan berapa unit pembangkit yang harus bekerja dan siap bekerja pada hari itu.

Sebagai dasar pertimbangan yang sifatnya umum, untuk menentukan biaya produksi tenaga listrik yang dibutuhkan adalah dengan memperhatikan bahwa dalam keadaan beban minimum maka tenaga listrik yang dibutuhkan diberikan oleh unit pembangkit yang bekerja paling efisien pada keadaan tersebut. Pembangkit ini akan terus beroperasi atau dibebani sampai pada batas efisiensi maksimumnya. Dan apabila ternyata beban masih terus bertambah sedangkan unit pembangkit ini telah mencapai maksimumnya maka selanjutnya beban ditanggung oleh unit pembangkit yang lain yang belum mencapai efisiensi maksimumnya. Dengan dasar operasi yang demikian maka dapat dicapai keadaan operasi yang cukup ekonomis.

Akan tetapi dengan semakin berkembangnya sistem itu sendiri maka diperlukan suatu perencanaan pembangkitan yang optimum dengan biaya operasi yang ekonomis dan harus memperhitungkan rugi-rugi yang terjadi pada saluran transmisi. Mengingat bahwa beban sistem adalah selalu berubah-ubah dari waktu

ke waktu maka perlu untuk membuat secara grafis perubahan beban terhadap waktu.

Oleh karena biaya operasi untuk memproduksi daya listrik, suatu pembangkit *hidro* (PLTA) sangat kecil jika dibandingkan dengan pembangkit *thermal* (PLTU, PLTG, PLTGU, PLTD) maka pembahasan selanjutnya untuk mendapatkan biaya operasi yang ekonomis sebagian besar ditekankan pada unit pembangkit *thermal* saja karena disini akan membutuhkan biaya operasi yang cukup tinggi sehingga usaha penghematan biaya bahan bakar akan sangat berarti. Dengan kata lain dengan mengkoordinasikan operasi pembangkit-pembangkit yang tersedia dengan tepat dan sesuai dengan beban maka akan didapat suatu keadaan operasi yang ekonomis.

Pembahasan mengenai operasi ekonomis adalah merupakan salah satu cara bagaimana menekan biaya produksi dari sistem tenaga listrik. Dalam hal ini maka metode yang dipakai adalah dengan memanfaatkan karakteristik dari menganalisa operasi dari sistem tersebut. Disamping karakteristik dari unit-unit pembangkit perlu juga diketahui karakteristik beban, karena karakteristik bebanlah maka dapat dianalisa pengaturan yang paling ekonomis dari setiap unit pembangkit. Adapun karakteristik yang perlu diketahui dari setiap unit pembangkit adalah :

1. Karakteristik *input* bahan bakar sebagai fungsi dari *output* daya.
 2. Nilai panas sebagai fungsi *output* daya.
 3. Kenaikan jumlah bahan bakar yang dibutuhkan jika terdapat perubahan beban.
-

Ketiga karakteristik tersebut merupakan pedoman menganalisa penjadwalan selanjutnya. Kemudian yang juga perlu diperhitungkan adalah variabel-variabel yang terdapat pada saluran transmisi, karena variabel-variabel ini juga sangat menentukan ekonomis tidaknya penjadwalan pembangkit yang kita tentukan.

Maka untuk mencapai suatu operasi yang ekonomis pada suatu sistem tenaga listrik adalah dengan melakukan penjadwalan pada sistem pembangkit yang ada pada suatu sistem tenaga listrik yang ditinjau tersebut dengan memanfaatkan karakteristik dari setiap masing-masing unit pembangkit yang ada pada dasarnya bertujuan untuk menekan biaya produksi listrik agar harga dari listrik yang dihasilkan dapat ditekan serendah mungkin sehingga dapat memuaskan pemakai listrik.

2.7 Karakteristik Unit Pembangkit

2.7.1 Karakteristik *Input-Output*

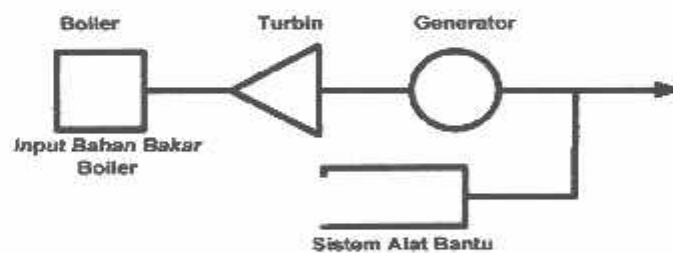
Hal yang paling mendasar dalam operasi pembangkitan yang ekonomis adalah dengan membuat karakteristik *input-output* dari unit pembangkit *thermal*. Karena ini diperoleh dari desain perencanaan atau melalui test pembangkit. Adapun definisi dari karakteristik *input-output* dari pembangkit itu sendiri adalah formula yang menyatakan hubungan antara *input* pembangkit sebagai fungsi dari *output* pembangkit. Sedangkan ciri dari unit *boiler-turbin-generator* dapat digambarkan dalam gambar 2.6, dimana unit ini memuat sebuah *boiler* yang

menghasilkan uap untuk menjalankan turbin yang dikopel dengan rotor dari generator.

Pada pembangkit thermal input diberikan dalam satuan panas Btu/jam atau Kalori/jam dari bahan bakar yang diberikan *boiler* untuk menghasilkan *output* pembangkit. Sedangkan notasi yang digunakan adalah H (MBtu/h) atau dalam satuan yang lain H (MKal/h). Adapapun dalam skripsi ini, perhitungan dilakukan adalah dalam satuan MKal/jam. Selain itu *input* dari pembangkit dapat pula dinyatakan dalam nilai uang yang menyatakan besarnya biaya yang diperlukan untuk bahan bakar. Notasi yang digunakan adalah F (Rp/h). Hubungan antara H dan F dapat dinyatakan dalam rumus sebagai berikut ini^[4]:

$$F = H \times \frac{\text{Rupiah}}{\text{MBtu}} \dots\dots\dots (2.24.)$$

Dimana $\frac{\text{Rupiah}}{\text{MBtu}}$ adalah nilai uang yang diperlukan per satuan panas dari bahan bakar.



Gambar 2.6^[4]
Unit Boiler-Turbin-Generator

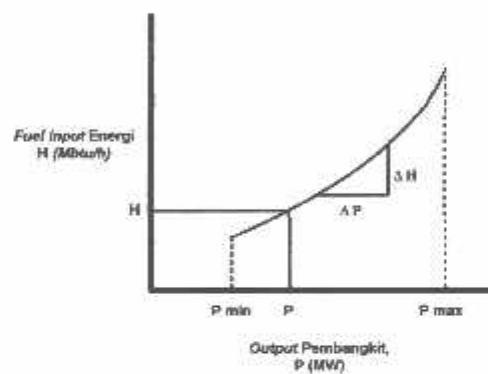
Seperti digambarkan dalam gambar 2.6. maka *output* dari pembangkit tidak hanya dihubungkan dengan sistem saja akan tetapi juga untuk sistem peralatan bantu pembangkit didefinisikan sebagai daya yang dikeluarkan oleh

generator karakteristik *input-output*, daya *output* adalah berupa daya netral dari pembangkit, notasi yang digunakan adalah P (MW).

Persamaan karakteristik *input-output* pembangkit dapat dilihat pada persamaan (2.25.) dan (2.26.) dibawah ini, sedangkan kurva dari karakteristik *input-output* pembangkit dapat dilihat pada gambar 2.7^[4] :

$$H = f(P), \text{ atau } \dots\dots\dots(2.25)$$

$$F = f(P) \dots\dots\dots(2.26)$$

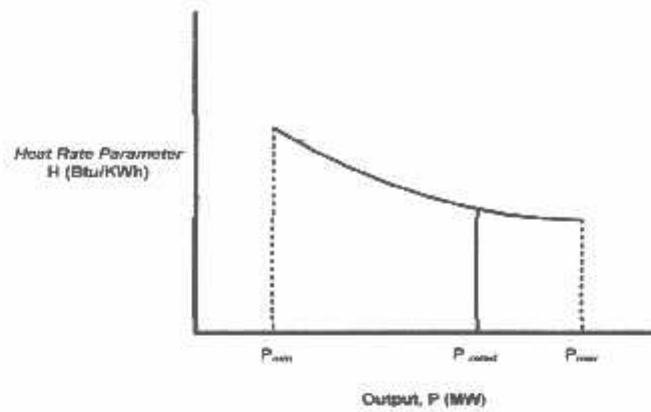


Gambar 2.7^[4]

Kurva Karakteristik *Input-Output* Pembangkit Thermal

2.7.2. Karakteristik *Heat-Rate*

Karakteristik *heat-rate* merupakan karakteristik yang menunjukkan efisiensi dari sebuah mesin. Karakteristik *heat-rate* sebuah unit pembangkit menunjukkan *input* kalor yang diberikan untuk menghasilkan energi sebesar 1 kiloWatt jam pada MegaWatt *output* dari suatu unit. Kurva dari karakteristik *heat-rate* ini dapat dilihat pada gambar 2.8. di bawah ini :



Gambar 2.8^[4]
Kurva Karakteristik *Heat-Rate* Unit Pembangkit

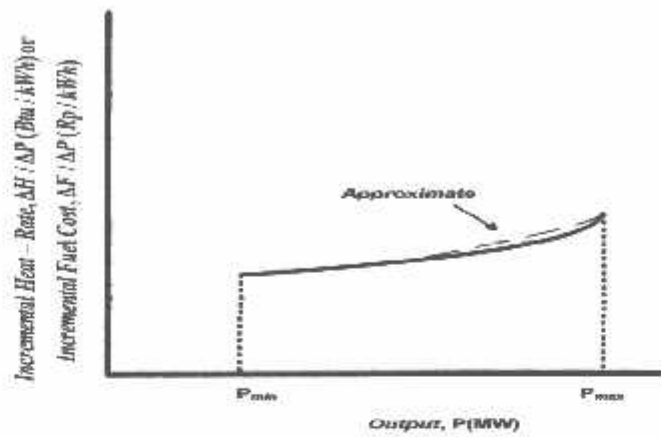
2.7.3 Karakteristik *Incremental Heat-Rate* dan *Incremental Fuel Cost*

Perwujudan yang lain dari karakteristik pembangkit adalah karakteristik *Incremental Heat-Rate* atau perubahan tingkat laju panas dan karakteristik *Incremental Fuel Cost* atau perubahan tingkat laju bahan bakar. Karakteristik ini menyatakan hubungan daya keluaran pembangkit sebagai fungsi *Incremental Heat-Rate* atau *Incremental Fuel Cost*. Karakteristik *Incremental Heat-Rate* ini menunjukkan besarnya perubahan *input* energi bila ada perubahan *output* pada unit pembangkit.

Kurva dari karakteristik *Incremental Heat-Rate* atau *Incremental Fuel Cost* dapat dilihat pada gambar 2.9. Sedangkan persamaan *Incremental Heat-Rate* dan persamaan *Incremental Fuel Cost* dapat dilihat pada persamaan (2.27.) hingga persamaan (2.28.)^[4]

$$\text{Incremental Heat-Rate} = \frac{\Delta H}{\Delta P} \left(\frac{\text{MBtu}}{\text{kWh}} \right) \dots\dots\dots (2.27.)$$

$$\text{Incremental Fuel Cost} = \frac{\Delta F}{\Delta P} \left(\frac{\text{Rupiah}}{\text{kWh}} \right) \dots\dots\dots(2.28.)$$



Gambar 2.9^[4]
Kurva Karakteristik *Incremental Heat-Rate/Fuel Cost*

Bila harga Δ sangat kecil maka dapat dinyatakan dengan persamaan berikut ini ^[4]

$$\text{Incremental Heat-Rate} = \frac{dH}{dP} \left(\frac{\text{MBtu}}{\text{kWh}} \right) \dots\dots\dots(2.29.)$$

$$\text{Incremental Fuel Cost} = \frac{dF}{dP} \left(\frac{\text{Rupiah}}{\text{kWh}} \right) \dots\dots\dots(2.30.)$$

2.8 *Economic Dispatch*^[4]

Dalam pembahasan tentang *Optimal Power Flow* dan operasi pada sistem tenaga listrik yang ekonomis, maka kita selalu membicarakan *economic dispatch*. *Economic dispatch* adalah pembagian pembebanan pada pembangkit-pembangkit yang ada dalam suatu sistem tenaga listrik, secara optimum dan ekonomis pada beban tertentu. Dengan dilakukan *economic dispatch* maka akan didapatkan biaya bahan bakar yang paling murah dalam suatu sistem pembangkit. Oleh karena

beban yang harus ditanggung oleh sistem pembangkit selalu berubah setiap periode waktu tertentu, maka perhitungan *economic dispatch* ini dilakukan untuk setiap harga beban tertentu pula.

2.8.1. Fungsi Biaya Bahan Bakar

Persamaan dari biaya bahan bakar dari setiap unit generator merupakan fungsi dari daya aktif saja, sehingga dengan demikian biaya operasi tiap pembangkit hanya tergantung dari daya aktif saja. Hal ini bisa dilihat dari persamaan di bawah ini^[1]

$$C_F = \sum_{i \in N_g} C_i(P_i) \dots\dots\dots (2.31.)$$

$$C_i(P_i) = a_i + b_i P_i + c_i P_i^2 \dots\dots\dots (2.32.)$$

dimana:

$C_i(P_i)$ = Biaya pembangkitan pada unit generator i

P_i = Besar daya aktif yang terbangkit pada unit generator i

a, b, c = Koefisien persamaan kuadratik.

Akan tetapi sebenarnya dalam suatu sistem tenaga listrik, keberadaan daya reaktif yang terbangkit juga harus diperhitungkan, sehingga daya reaktif yang terbangkit ikut serta secara tak langsung mempengaruhi biaya operasi. Dengan demikian daya reaktif tidak hanya digunakan untuk perhitungan total rugi pada transmisi, akan tetapi juga harus digunakan untuk masalah optimasi.

2.8.2. Fungsi Biaya Daya Reaktif Pembangkit

Biaya ini merupakan hubungan antara biaya pengoperasian dan besarnya daya reaktif yang terbangkit pada unit generator. Hubungan tersebut akan dijelaskan pada persamaan di bawah ini^[1]:

$$C_i(Q_i) = \sum_{i \in Ng} \{C_{gi} |\bar{Q}_{gi}|\} \dots\dots\dots (2.33.)$$

dimana:

$C_i(Q_i)$ = Biaya daya reaktif pembangkit

C_{gi} = Biaya pengoperasian daya reaktif

\bar{Q}_{gi} = Besar daya reaktif yang terbangkit pada unit generator i

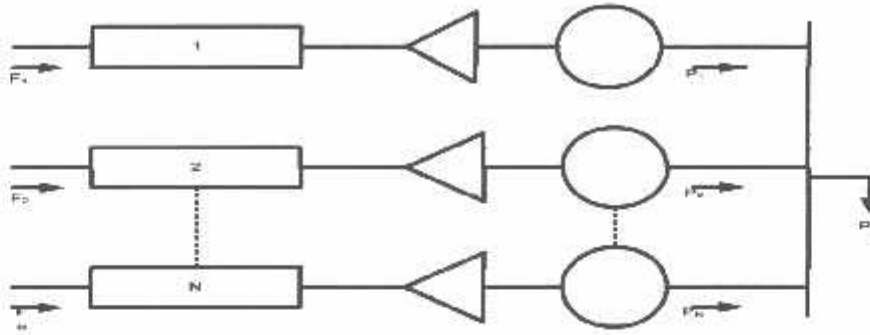
Biaya pengoperasian disini tidak seperti biaya bahan bakar yang merupakan biaya pengoperasian dari pembangkitan daya aktif, tetapi hanya merupakan biaya pengoperasin kecil, seperti biaya pemeliharaan untuk pembangkitan daya reaktif.

2.8.3. *Economic Dispatch* Dengan Mengabaikan Rugi-rugi Transmisi

Dalam sistem tenaga listrik, kerugian transmisi merupakan kehilangan daya yang harus ditanggung oleh sisi pembangkit. Jadi dengan adanya kerugian daya tersebut merupakan tambahan beban bagi sistem tenaga listrik.

Sistem dengan mengabaikan rugi-rugi transmisi dapat dilihat pada gambar 2.10. Sistem ini terdiri dari N buah pembangkit thermal yang dihubungkan pada *single bus bar* yang melayani beban P_R . *Input* dari masing-masing pembangkit ditunjukkan oleh F_i yang mewakili biaya dari satu unit

pembangkit dan *output* dari masing-masing unit P_i adalah daya yang dihasilkan oleh satu unit pembangkit.



Gambar 2.10^[4]
N Unit Pembangkit Thermal Melayani Beban P_R

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah biaya dari masing-masing unit pembangkit. Dan pembatas yang paling penting adalah jumlah *output* dari masing-masing unit pembangkit sama dengan beban di konsumen. Yang menjadi permasalahan adalah meminimumkan total biaya F_T dengan memperhatikan pembatas ϕ bahwa daya yang dihasilkan oleh pembangkit sama dengan beban yang diterima. Secara matematika pernyataan yang tersebut di atas dapat dinyatakan dengan persamaan sebagai berikut^[4]:

$$F_T = F_1 + F_2 + F_3 + \dots + F_N \dots\dots\dots (2.34)$$

$$= \sum_{i=1}^N F_i(P_i)$$

$$\phi = 0 = P_R - \sum_{i=1}^N P_i \dots\dots\dots (2.35)$$

Persamaan di atas adalah pembatas yang merupakan problem dari optimasi dan ini dapat dipecahkan dengan menggunakan kalkulus tingkat lanjut yang melibatkan fungsi *La Grange*. Dimana fungsi ini didapat dengan cara menambahkan pembatas ϕ yang telah dikalikan dengan faktor pengali *La Grange* λ pada fungsi tujuan F_T . Fungsi *La Grange* dapat ditunjukkan dengan persamaan di bawah ini^[4]:

$$L = F_T + \lambda \cdot \phi \dots\dots\dots (2.36.)$$

Persamaan *La Grange* di atas merupakan fungsi *output* pembangkit P_i dan faktor pengali *La Grange* λ . Keadaan optimal dari fungsi tujuan F_T dapat diperoleh dengan operasi gradient dari persamaan *La Grange* sama dengan nol^[4]:

$$\nabla L = 0 \dots\dots\dots (2.37)$$

$$\nabla F_T + \lambda \cdot \phi = 0 \dots\dots\dots (2.38)$$

$$\frac{\partial L}{\partial P} = \frac{\partial F_T}{\partial P_i} + \lambda \cdot \left(\frac{\partial P_R}{\partial P_i} - \frac{\partial P_i}{\partial P_i} \right) = 0 \dots\dots\dots (2.39)$$

$$\text{atau } \frac{\partial F_T}{\partial P_i} + \lambda \cdot (0 - 1) = 0 \dots\dots\dots (2.40)$$

$$\frac{\partial F_T}{\partial P_i} = \lambda \dots\dots\dots (2.41)$$

Persamaan terakhir ini menunjukkan bahwa bila digunakan biaya bahan bakar F_T yang paling minimum maka *incremental cost* setiap unit generator pembangkit harus sama yaitu sebesar λ . Kondisi optimal ini tentunya dengan tetap memperhatikan pembatas yang ada yaitu bahwa daya dari setiap unit generator

pembangkit harus lebih besar atau sama dengan daya *output* minimum dan lebih kecil atau sama dengan daya *output* maksimum yang diijinkan.

Dari N buah pembangkit yang ada dalam sistem tenaga yang telah dibahas dan beban sistem sebesar P_R , maka dapat diambil kesimpulan sebagai berikut^[4]:

$$\frac{\partial F_i}{\partial P_i} = \lambda \quad \text{ada N buah persamaan} \dots \dots \dots (2.42)$$

$$P_{i, \min} \leq P_i \leq P_{i, \max} \quad \text{ada 2 N buah pertidaksamaan} \dots \dots \dots (2.43)$$

$$\sum_{i=1}^N P_i = P_R \dots \dots \dots (2.44)$$

Dari batasan pertidaksamaan pembatas di atas dapat diperluas menjadi^[4]:

$$\frac{\partial F_i}{\partial P_i} = \lambda \quad \text{untuk } P_{i, \min} \leq P_i \leq P_{i, \max} \dots \dots \dots (2.45)$$

$$\frac{\partial F_i}{\partial P_i} \leq \lambda \quad \text{untuk } P_i = P_{i, \max} \dots \dots \dots (2.46)$$

$$\frac{\partial F_i}{\partial P_i} \geq \lambda \quad \text{untuk } P_i = P_{i, \min} \dots \dots \dots (2.47)$$

Karena F_i hanya sebagai fungsi P_i maka $\frac{\partial F_i}{\partial P}$ dapat diganti dengan $\frac{dF_i}{dP_i}$.

2.8.4. *Economic Dispatch Dengan Memperhitungkan Rugi-Rugi Transmisi*

Sistem dengan memperhitungkan rugi-rugi transmisi dapat dilihat pada gambar 2.11. Sistem ini terdiri dari N buah unit pembangkit thermal dihubungkan melalui saluran transmisi yang melayani beban P_R . *Input* dari masing-masing unit ditunjukkan oleh F_i yang mewakili biaya dari satu unit pembangkit dan *output* dari masing-masing unit P_i adalah daya yang dihasilkan oleh satu unit pembangkit.

BAB III

OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE
KOMBINASI *EVOLUTIONARY PROGRAMMING - EVOLUTIONARY*
STRATEGY- GENETIC ALGORITHM

3.1. Analisa Aliran Daya^{[3][5]}

Dalam melayani beban yang dibutuhkan oleh konsumen dan pengoperasian tenaga listrik perlu menganalisa aliran daya, sehingga sistem yang dioperasikan dapat memenuhi persyaratan teknis maupun ekonomisnya. Dalam analisa aliran daya dilakukan perhitungan terhadap tegangan, arus, daya aktif dan reaktif yang terdapat dalam berbagai titik dalam jala-jala jaringan transmisi tenaga listrik.

Tujuan dari analisa aliran daya adalah :

1. Mencari harga magnitude tegangan $|v|$ dan sudut fasa tegangan δ bus beban.
2. Mencari besar daya reaktif Q dan sudut fasa tegangan δ dari generator bus.
3. Untuk mengetahui besar daya aktif dan daya reaktif pada bus slack.
4. Untuk mengetahui apakah semua peralatan pada sistem memenuhi batas-batas yang telah ditetapkan untuk operasi penyaluran daya.
5. Untuk mengetahui kondisi awal pada perencanaan sistem yang baru.
6. Untuk menentukan besar daya yang mengalir disetiap saluran jaringan tenaga listrik.

3.1.1. Klasifikasi Bus^{[3][4]}

Pada setiap bus dari jaringan terdapat parameter-parameter yaitu daya aktif (P), daya reaktif (Q), besar tegangan $|v|$ dan sudut fasa tegangan δ .

Dengan melihat parameter diatas setiap bus dapat diklasifikasikan menjadi 3 bagian:

1. Bus Beban (PQ) atau *Load Bus*

Pada bus ini hanya terdapat kebutuhan daya untuk beban dimana P daya aktif dan Q daya reaktif diketahui, sementara $|v|$ dan δ berubah-ubah sesuai dengan kebutuhan. Oleh karena itu $|v|$ dan δ harus dicari.

2. Bus Generator (PV)

Pada bus ini hanya terdapat daya pembangkitan dimana $|v|$ diatur menggunakan regulator tegangan dan P diatur dengan governor. Sehingga untuk bus ini P dan $|v|$ diketahui sementara itu Q (daya reaktif) dan δ (sudut fasa) dicari.

3. Bus Slack

Pada bus ini $|v|$ dan δ sudah ditentukan besarnya sementara P dan Q dihitung. Biasanya nilai $|v|$ adalah 1 pu, sedangkan sudut fasa tegangan δ berharga nol, karena itu fasor tegangan dari bus dipakai sebagai referensi.

Daya total yang mengalir pada setiap bus dituliskan sebagai berikut (Stevenson, Willism D, Jr, 1996 : 187):

$$S_k = P_k + jQ_k = V_k \hat{I}_k \dots\dots\dots (3.1.)$$

Atau

$$P_k - jQ_k = V_k \sum_{n=1}^N Y_{kn} V_n \dots\dots\dots (3.2.)$$

Dari persamaan $V_k = v_k + j\delta_k$ dan $Y_{k11} = G_{k11} - jB_{k11}$, maka persamaan 3.2. menjadi (Stevenson, Willism D, Jr, 1996 : 187) :

$$P_k - jQ_k = (v_k + j\delta_k) \sum_{n=1}^N (G_{k11} - jB_{k11})(V_n + j\delta_n) \dots\dots\dots (3.3)$$

Bila dituliskan dalam bentuk real dan imajiner maka persamaan di atas menjadi (Stevenson, Willism D, Jr, 1996 : 187) :

$$P_k = \sum_{n=1}^N \{v_k (v_n G_{kn} + \delta_n B_{kn}) + \delta_k (\delta_n G_{kn} - v_n B_{kn})\} \dots\dots\dots (3.4.)$$

$$Q_k = \sum_{n=1}^N \{\delta_k (v_n G_{kn} + \delta_n B_{kn}) - v_k (\delta_n G_{kn} - v_n B_{kn})\} \dots\dots\dots (3.5.)$$

3.1.2. Metode Newton Rapshon¹⁵¹

Proses yang dilakukan adalah membandingkan antara daya yang ditempatkan berdasarkan data ($P_{k, \text{sched}}$ dan $Q_{k, \text{sched}}$) dengan daya hasil perhitungan ($P_{k, \text{calc}}$ dan $Q_{k, \text{calc}}$) menggunakan persamaan (3.4.) dan (3.5.) di atas. Selisih daya yang diterapkan dan perhitungan (ΔP_k dan ΔQ_k) dihitung dengan persamaan (Stevenson, Willism D, Jr, 1996 : 187) :

$$\Delta P_k = P_{k, \text{sched}} - P_{k, \text{calc}} \dots\dots\dots (3.6.)$$

$$\Delta Q_k = Q_{k, \text{sched}} - Q_{k, \text{calc}} \dots\dots\dots (3.7.)$$

Selisih daya dihitung dengan persamaan (3.6.) dan persamaan (3.7.) digunakan untuk menghitung nilai perubahan parameter tegangan bus, yaitu $\Delta|V_k|$ dan $\Delta\delta_k$, yaitu dengan menggunakan elemen Jacobian, sehingga koreksi terhadap nilai parameter tegangan yang telah ditetapkan nilai awal sebelumnya. Elemen Jacobian sendiri merupakan turunan parsial P dan Q terhadap masing-masing variabel pada persamaan (3.4.) dan (3.5.), yang dalam bentuk matriks dituliskan sebagai (Stevenson, Willism D, Jr, 1996 : 187) :

$$\begin{bmatrix} \Delta P_1 \\ \dots \\ \Delta P_{n-1} \\ \Delta Q_1 \\ \dots \end{bmatrix} = \begin{bmatrix} \frac{\partial P_1}{\partial v_1} & \dots & \frac{\partial P_1}{\partial v_{n-1}} & \frac{\partial P_1}{\partial \delta_1} & \dots & \frac{\partial P_1}{\partial \delta_{n-1}} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\partial P_{n-1}}{\partial v_1} & \dots & \frac{\partial P_{n-1}}{\partial v_{n-1}} & \frac{\partial P_{n-1}}{\partial \delta_1} & \dots & \frac{\partial P_{n-1}}{\partial \delta_{n-1}} \\ \frac{\partial Q_1}{\partial v_1} & \dots & \frac{\partial Q_1}{\partial v_{n-1}} & \frac{\partial Q_1}{\partial \delta_1} & \dots & \frac{\partial Q_1}{\partial \delta_{n-1}} \\ \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix} \begin{bmatrix} \Delta v_1 \\ \dots \\ \Delta v_{n-1} \\ \Delta \delta_1 \\ \dots \end{bmatrix}$$

$$\Delta Q_{n-1} = \frac{\partial Q_{n-1}}{\partial v_1} \dots \frac{\partial Q_{n-1}}{\partial v_{n-1}} \frac{\partial Q_{n-1}}{\partial \delta_1} \dots \frac{\partial Q_{n-1}}{\partial \delta_{n-1}} = \Delta \delta_{n-1}$$

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |v| \end{bmatrix} \dots \dots \dots (3.8.)$$

Dimana elemen-elemen jacobian dapat dihitung dengan menurunkan persamaan-persamaan (3.4.) dan (3.5.)

Perubahan nilai tegangan bus dijumlahkan dengan nilai tegangan bus sebelumnya, yang kemudian nilai tegangan bus terbaru ini digunakan untuk menghitung kembali daya $P_{k, calc}$ dan $Q_{k, calc}$ menurut persamaan (3.4.) dan (3.5.) Proses ini terus berulang, yang disebut iterasi hingga mencapai kondisi dimana nilai perubahan daya ΔP dan ΔQ konvergen mencapai suatu nilai minimum yang telah ditentukan (berkisar 0,001 hingga 0,0001 pu)

3.2. Kombinasi *Evolutionary Programming* - *Evolutionary Strategy* - *Genetic Algorithm*

3.2.1. Metode *Evolutionary Programming* (EP)^[4]

Evolutionary Programming merupakan metode yang biasa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. EP ditemukan oleh Lawrence.J. Fogel pada tahun 1960. Metode ini didasarkan pada proses evolusi yang ada dalam makhluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam, yaitu siapa yang kuat dia yang bertahan. Dengan meniru proses ini EP dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata.

b. Parameter-parameter *Evolutionary Programming*

Dalam penggunaan metode *Evolutionary Programming* terdapat beberapa parameter yang digunakan. Parameter-parameter yang digunakan tersebut adalah:

1. Jumlah Generasi (*Maxgen*)
2. Ukuran Populasi (*Popsize*)
3. Probabilitas Mutasi (*Pm*)
4. Panjang Kromosom (*NVAR*)

c. Proses Kerja *Evolutionary Programming*

1. Pengkodean atau Representasi

Langkah pertama kali yang dilakukan dalam penggunaan EP adalah melakukan pengkodean atau representasi terhadap permasalahan yang akan dilakukan, dimana dibentuk oleh serangkaian kromosom. Setiap elemen dalam kromosom ini adalah variabel string yang disebut gen. Selanjutnya beberapa kromosom dibentuk dan berkumpul membentuk populasi. Populasi inilah populasi awal bagi EP untuk awal melakukan pencarian.

2. Fungsi *Fitness* (Fungsi Evaluasi)

Dalam EP, sebuah fungsi *fitness* $f(x)$ harus dirancang untuk masing-masing permasalahan yang akan diselesaikan. Dengan menggunakan kromosom tertentu,

fungsi obyektif atau fungsi evaluasi akan mengevaluasi status masing-masing kromosom. Setiap gen x_i ($i = 1, 2, \dots, N$) dipergunakan untuk menghitung $f_k(x)$ ($k = 1, 2, \dots, \text{POPSIZE}$).

3. Mutasi (*Mutation*)

Operator mutasi digunakan untuk melakukan modifikasi satu atau lebih nilai gen dalam individu yang sama. Tujuannya agar individu-individu yang ada dalam populasi semakin bervariasi.

4. Kompetisi

Dalam tahap ini dihasilkan populasi baru dari populasi yang ada. Melalui penggunaan skema kompetisi setiap individu dalam populasi orang tua (*parent*) maupun anak (*offspring*) akan dikompetisi/bersaing satu dengan yang lainnya. Kompetisi setiap individu dengan lawannya didasarkan pada nilai *fitness* dari setiap individu tersebut.

5. Konvergensi

Proses evolusi merupakan proses yang tidak pernah selesai. Demikian pula *Evolutionary Programming* tidak dapat memperkirakan saat proses berhenti untuk memperoleh solusi yang optimal. Konvergensi merupakan kondisi yang dicapai pada saat populasi kehilangan kondisi keanekaragaman.

3.2.2. Genetic Algorithm ^{[8][10][11][12]}

Genetic Algorithm merupakan metode adaptive yang bisa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Algoritma ini didasarkan pada proses genetik yang ada dalam makhluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam “siapa yang kuat, dia yang bertahan (*survive*)”. Dengan meniru proses ini, Genetic Algorithm dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata.

Genetic Algorithm ditemukan oleh John Holland pada awal tahun 1970 yang dilandasi oleh sifat-sifat evolusi alam. Holland percaya bahwa ini sangat cocok digabungkan dalam sebuah algoritma komputer, menghasilkan sebuah teknik penyelesaian untuk permasalahan-permasalahan yang sulit dengan langkah alami yaitu melalui evolusi. John Holland mulai bekerja dengan algoritma yang dibentuk dari string-string biner 1 dan 0 yang disebut *kromosom*. Seperti halnya alam, algoritma ini menyelesaikan permasalahan-permasalahan dengan menemukan kromosom-kromosom yang baik dengan memanipulasi materi dan sifat (*gene*) kromosom-kromosom. Algoritma ini tidak mengetahui type permasalahan yang akan diselesaikan. Hanya informasi yang telah diberikan dari *evaluasi* berupa nilai fitness setiap kromosom dengan nilai fitness terbaik yang bertahan hidup dan selalu diproduksi.

Sebelum Genetic Algorithm dijalankan, maka sebuah kode yang sesuai (representasi) untuk persoalan harus dirancang. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/string yang terdiri dari komponen genetik terkecil yaitu gen. Pemakaian bilangan seperti integer, floating point dan abjad sebagai *allele* (nilai gen) memungkinkan penerapan operator genetika yaitu proses produksi (*reproduction*), pindah silang (*crossover*), mutasi (*mutation*) untuk menciptakan himpunan titik-titik solusi. Untuk memeriksa hasil optimasi, kita membutuhkan fungsi *fitness* yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, pindah silang dan mutasi untuk menciptakan keturunan (*offspring*).

Jika Algoritma Genetika didesain dengan baik, populasi akan mengalami konvergensi dan akan mendapatkan sebuah solusi yang optimum.

Genetic Algorithm memiliki empat dasar kerja yaitu :

1. Bekerja dengan mengkodekan parameter-parameter permasalahan dan tidak bekerja secara langsung dengan parameter-parameter tersebut.
2. Mencari solusi masalah dari sejumlah populasi kandidat solusi, tidak hanya memproses satu solusi saja.
3. Hanya memperhitungkan fungsi fitness setiap kandidat solusi untuk mendapatkan hasil optimum global.
4. Menggunakan aturan transisi secara probabilistik bukan deterministik.

3.2.3. *Evolutionary Strategy (ES)*^{[8][10][11][12]}

Evolutionary strategy merupakan metode yang biasa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Metode ini didasarkan pada proses evolusi yang ada dalam makhluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti mekanisme alam, dimana individu yang lebih kuat memiliki kemungkinan untuk menjadi pemenang dan mempunyai kesempatan hidup yang lebih besar di dalam lingkungan yang kompetitif. Dengan meniru proses ini ES dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata.

ES merupakan suatu metode strategi optimasi yang merupakan cabang dari *Evolutionary Computation*, yang didalamnya terdiri dari Genetic Algorithm, *Genetic programming*, *Evolutionary Strategy* dan *Evolutionary Programming*.

ES ditemukan oleh Rechenberg pada tahun 1973 dengan seleksi (*selection*), mutasi (*mutation*), populasi (*population*) kemudian dikembangkan oleh Schwefel tahun 1981 yang memperkenalkan *recombination* dan populasi (*population*) individu lebih dari satu. Mereka kemudian menyajikan suatu perbandingan *Evolutionary Strategy* (ES) dengan teknik optimasi yang lebih tradisional.

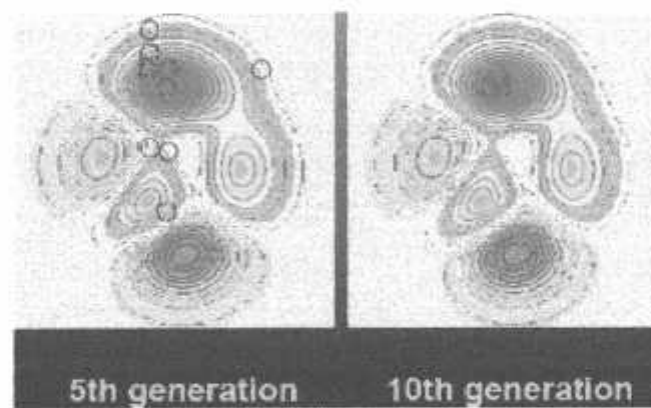
Sebelum ES dijalankan, maka sebuah kode yang sesuai (representasi) untuk persoalan harus dirancang. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/string yang terdiri dari komponen genetik terkecil yaitu gen. Pemakaian bilangan real (*floating point*) sebagai *allele* (nilai gen) memungkinkan penerapan operator ES yaitu proses seleksi (*selection*), *recombination*, mutasi (*mutation*), dan kompetisi (*competition*) untuk menciptakan himpunan titik-titik solusi. Untuk memeriksa hasil optimasi, kita membutuhkan fungsi *fitness* yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, mutasi dan kompetisi untuk menciptakan keturunan (*offspring*).

3.3. Parameter Kombinasi *Evolutionary Programming - Evolutionary strategy - Genetic Algorithm*^[11]

Terdapat beberapa parameter yang digunakan dalam kombinasi evolutionary programming - evolutionary strategy - genetic algorithm. Parameter tersebut digunakan untuk melihat kompleksitas dari kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm*. Parameter yang digunakan tersebut adalah :

3.3.1 Jumlah Generasi (MAXGEN)

Merupakan jumlah perulangan (iterasi) dilakukannya rekombinasi dan seleksi. Jumlah generasi ini mempengaruhi kestabilan output dan lama iterasi (waktu proses *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm*). Jumlah generasi yang besar dapat mengarahkan kearah solusi yang optimal, namun akan membutuhkan waktu yang lama. Sedangkan jika jumlah generasinya terlalu sedikit maka solusi akan terjebak pada local optimum.



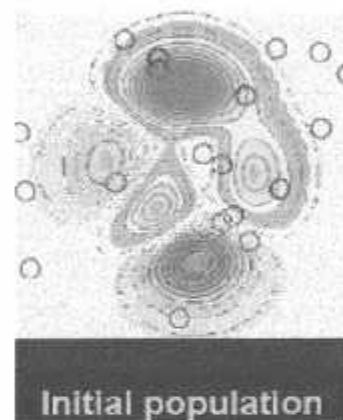
Gambar 3.1 .^[15]

Jumlah Generasi

Sumber : Neuro-Fuzzy and soft Computing” Reference book :Jang et al

3.3.2. Ukuran Populasi (POPSIZE)

Ukuran populasi mempengaruhi kinerja dan efektifitas dari kombinasi evolutionary programming - evolutionary strategy - genetic algorithm. Jika ukuran populasi kecil maka populasi tidak menyediakan cukup materi untuk mencakup ruang permasalahan, sehingga pada umumnya kinerja kombinasi evolutionary programming - evolutionary strategy - genetic algorithm menjadi buruk. Dalam hal ini dibutuhkan ruang yang lebih besar untuk mempersentasikan keseluruhan ruang permasalahan. Selain itu penggunaan populasi yang besar dapat mencegah terjasinya konvergensi pada wilayah local.



Gambar 3.2 .¹¹⁵¹

Ukuran Populasi

Sumber : Neuro-Fuzzy and soft Computing” Reference book :Jang et al

3.3.3 Probabilitas Crossover (P_c)

Probabilitas crossover ini digunakan untuk mengendalikan frekuensi operator crossover. Dalam hal ini, dalam populasi terdapat $P_c \times POPSIZE$ struktur (individu) yang melakukan pindah silang. Semakin besar nilai probabilitas crossover maka semakin cepat struktur baru yang diperkenalkan dalam populasi.

Namun jika probabilitas crossover terlalu besar maka struktur dengan nilai fungsi obyektif yang baik dapat hilang dengan lebih cepat dari seleksi. Sebaliknya jika probabilitas terlalu kecil akan menghalangi proses pencarian dalam proses Genetic Algorithm

3.3.4 Probabilitas Mutasi (P_m)

Mutasi digunakan untuk meningkatkan variasi populasi digunakan untuk menentukan tingkat mutasi yang terjadi, karena frekuensi terjadinya mutasi tersebut menjadi $P_m \times \text{POPSIZE} \times N$, dimana N adalah panjang struktur / gen dalam satu individu. Probabilitas mutasi yang rendah akan menyebabkan gen-gen yang berpotensi tidak dicoba. Dan sebaliknya, tingkat mutasi yang tinggi akan menyebabkan keturunan akan semakin mirip dengan induknya. Dalam kombinasi evolutionary programming - evolutionary strategy - genetic algorithm, mutasi menjalankan aturan penting yaitu

1. Mengganti gen-gen yang hilang sama proses seleksi.
2. Menyediakan gen-gen yang tidak muncul pada saat inisialisasi awal populasi.

3.3.5 Panjang Kromosom ($NVAR$)

Panjang kromosom berbeda-beda sesuai dengan model permasalahan. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/string yang terdiri dari komponen genetik terkecil yaitu gen. Pengkodean dapat memakai bilangan seperti string biner, integer, floating point dan abjad

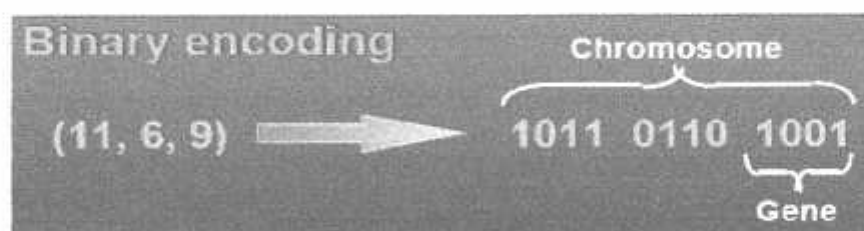
3.4 Mekanisme Kombinasi Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm ^[12]

3.4.1 Pengkodean atau Representasi

Langkah pertama kali yang dilakukan dalam penggunaan kombinasi evolutionary programming - evolutionary strategy - genetic algorithm adalah melakukan pengkodean atau representasi terhadap permasalahan yang akan dilakukan.

Secara umum kombinasi evolutionary programming - evolutionary strategy - genetic algorithm dibentuk oleh serangkaian kromosom yang ditandai dengan x_i ($i = 1, 2, \dots, N$). Setiap elemen dalam kromosom ini adalah variabel string yang disebut gen, berisi nilai-nilai atau allele. Variabel-variabel ini dapat dinyatakan dalam bentuk bilangan real (*floating point*).

Selanjutnya beberapa kromosom dibentuk dan berkumpul membentuk populasi. Populasi inilah populasi awal bagi kombinasi evolutionary programming - evolutionary strategy - genetic algorithm untuk awal melakukan pencarian.



Gambar 3.3 ^[15]

Pengkodean Representasi

Sumber : "Neuro-Fuzzy and soft Computing" Reference book :Jang et al

3.4.2 Fungsi *Fitness* (Fungsi Evaluasi)

Sebuah fungsi *fitness* $f(x)$ harus dirancang untuk masing-masing permasalahan yang akan diselesaikan. Dengan menggunakan kromosom tertentu, fungsi obyektif atau fungsi evaluasi akan mengevaluasi status masing-masing kromosom. Setiap gen x_i ($i = 1, 2, \dots, N$) dipergunakan untuk menghitung $f_k(x)$ ($k = 1, 2, \dots, \text{POPSIZE}$).

Pada permulaan optimasi, biasanya nilai *fitness* masing-masing individu masih mempunyai rentang yang lebar. Seiring dengan bertambah besar generasi, beberapa kromosom mendominasi populasi dan mengakibatkan rentang nilai *fitness* semakin kecil.

3.4.3 Seleksi dan Reproduksi^[12]

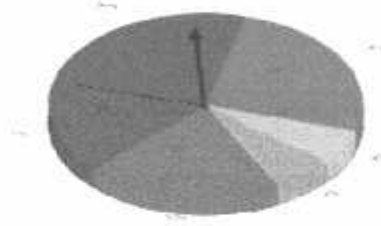
Masalah yang paling mendasar pada proses ini adalah bagaimana proses penyeleksiannya. Menurut teori Darwin proses seleksi individu adalah : "*individu terbaik akan tetap hidup dan menghasilkan keturunan*". Pada proses seleksi ini dapat menggunakan banyak metode seperti *roulette wheel selection*, *rank selection*, *elitesm* dan lain sebagainya.

3.4.3.1 *Roulette Wheel Selection*

Dimana setiap individual memiliki harga *fitness* sehingga didapatkan probabilitas individual $(f(i) / \sum f(i))$ tersebut dicopykan pada populasi yang baru. Untuk individual yang memiliki probabilitas 20% untuk jumlah populasi 10 maka kemungkinan individual tersebut dapat terpilih sebanyak dua kali.

Adapun algoritma dari *roulette-wheel* adalah sebagai berikut :

1. Menjumlahkan fitness dari seluruh anggota populasi.
2. Membangkitkan nilai k , suatu nilai random antara 0 dan total fitnessnya.
3. Menjumlahkan fitness dari kromosom-kromosom dari populasi mulai 0 hingga total fitness lebih besar atau sama dengan nilai k lalu ambil kromosom tersebut



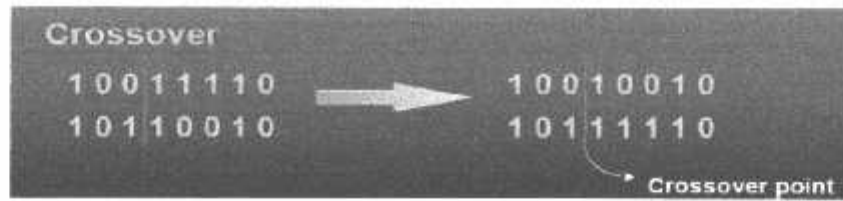
Gambar 3.4 .^[15]
Roulette Wheel

Sumber : *Neuro-Fuzzy and soft Computing*" Reference book :Jang et al

3.4.3.2 Rank Selection

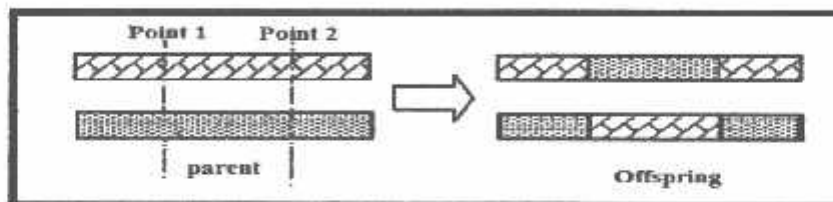
Apabila fitness yang dimiliki oleh suatu kromosom dalam populasi berbeda terlalu jauh dari kromosom lainnya maka hal ini dapat menjadi permasalahan. Misalnya bila kromosom terbaik mempunyai *fitness* yang menyebabkan besarnya tempat yang dimilikinya dalam *roulette wheel* sebesar 90% maka kromosom-kromosom yang lain akan mempunyai peluang yang terlalu kecil untuk diseleksi.

Rank selection pertama kali merangking populasi dan kemudian setiap kromosom diberi nilai fitness baru berdasarkan hasil rangking tersebut. Yang pertama akan mempunyai *fitness* 1, yang kedua akan mempunyai fitness 2 dan

Gambar 3.5.^[15]

Ilustrasi operator dengan *One Point Crossover*

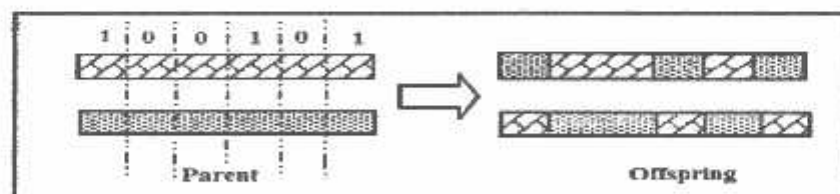
Sumber : *Neuro-Fuzzy and soft Computing* Reference book :Jang et al

Gambar 3.6.^[11]

Ilustrasi operator dengan *Two Point Crossover*

Sumber : *Artificial intelligence (teknik dan aplikasinya)* Sri Kusuma dewi

Untuk *crossover* uniform dibangkitkan suatu nilai random 0 dan 1 sepanjang jumlah kromosom untuk nilai loci. Jika nilai yang dibangkitkan mempunyai nilai 1 maka *allele* parent 2 dan offspring 2 untuk loci tersebut diambil dari *allele* parent 1 dan offspring 2 untuk loci tersebut diambil dari *allele* parent 2. Ilustrasi kerja operator ini digambarkan seperti pada gambar 3.7

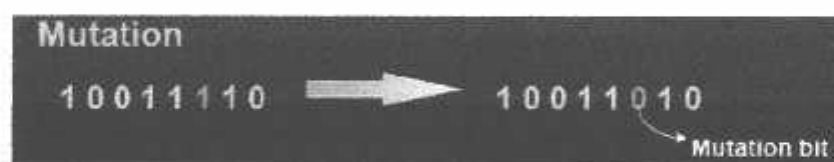
Gambar 3.7.^[11]

Ilustrasi operator *crossover* dengan *uniform crossover*

Sumber : *Artificial intelligence (teknik dan aplikasinya)* Sri Kusuma dewi

3.4.5 Mutation (*Mutasi*)^{[11][12]}

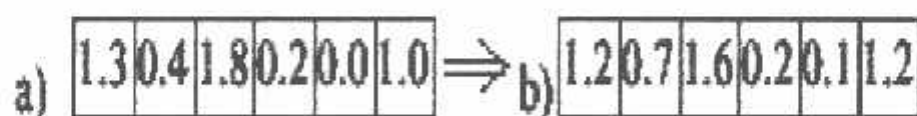
Operator mutasi digunakan untuk memodifikasi satu atau lebih nilai gen dalam satu individu. Cara kerjanya dengan membangkitkan sebuah nilai random r_k dimana $k = 1, 2, \dots, NVAR$ (panjang kromosom). Probabilitas mutasi (P_m) ditentukan dan digunakan untuk mengendalikan frekuensi operator mutasi. Apabila nilai random $r_k < P_m$ maka gen ke-k kromosom tersebut terpilih untuk mengalami mutasi. Proses mutasi dalam kombinasi *evolutionary programming - evolutionary strategy - genetic algorithm* menggunakan operator *Gaussian mutation*, dimana setiap individu akan terpilih secara acak untuk mengalami mutasi berdasarkan nomor acak Gaussian untuk menciptakan individu baru (*offspring*)



Gambar 3.8.^[15]

Ilustrasi operasi mutasi

Sumber : Neuro-Fuzzy and soft Computing" Reference book :Jang et al



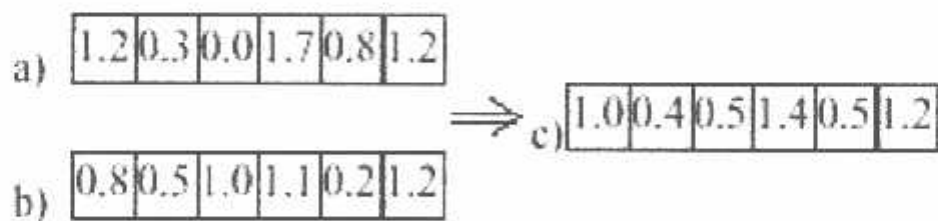
Gambar 3.1.^[11]

Mutasi Gaussian Dari Induk (*parent*) a) Menghasilkan Anak (*offspring*) b)

Fungsi dari operator mutasi adalah untuk menghindari agar solusi masalah yang diperoleh bukan merupakan solusi optimum lokal. Tipe dan implementasi dari operator mutasi bergantung pada jenis pengkodean dan permasalahan yang dihadapi. Seberapa sering mutasi dilakukan dinyatakan dengan suatu probabilitas mutasi, P_m . Posisi elemen pada kromosom yang akan mutasi ditentukan secara random. Mutasi dikerjakan dengan cara melakukan perubahan pada elemen tersebut.

3.4.6 Recombination (Rekombinasi)^{[11][12]}

Dalam *Evolutionary strategies* setelah melakukan *initialization* dan *evaluation*, individual dipilih secara acak menjadi *parents*. Setelah melakukan seleksi *parent* kemudian melakukan rekombinasi. Rekombinasi merupakan metode *intermediate recombination* (rekombinasi menengah) yang hanya dapat digunakan untuk variabel real (dan variabel yang bukan biner). Nilai variabel *offspring* dipilih disekitar dan antara nilai variabel *parent*, dimana vektor dari dua orang tua (*parent*) dirata-ratakan bersama-sama, unsur demi unsur, untuk membentuk suatu keturunan yang baru.



Gambar 3.1^[11]

Recombination dari Induk (*parent*) a) & b) menjadi Anak (*offspring*) c)

Seleksi orang tua (*parent*) menjadi turunan (*offspring*) menjadi lebih mudah, sebagai contoh dalam kaitannya dengan sifat alami penyajian, adalah sangat mudah merata-ratakan dari banyak individu menjadi turunan tunggal. Didalam *Evolutionary Strategy*, N *parent* terpilih secara acak (tidak didasarkan pada fitness terbaik), kemudian N *offspring* dihasilkan melalui penggunaan *intermediate recombination*, dimana variabel pada anak merupakan hasil kombinasi variabel induk.

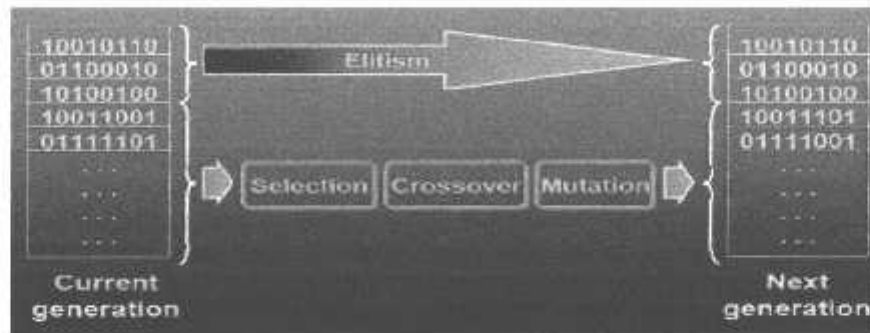
3.4.7 Competition (*Kompetisi*)

Dalam tahap kompetisi, mekanisme seleksi dipakai untuk menghasilkan populasi baru dari populasi yang ada. Melalui penggunaan skema kompetisi setiap individu dalam populasi rang tua (*parent*) maupun anak (*offspring*) akan dikompetisi/bersaing satu dengan yang lainnya. Kompetisi setiap individu dengan lawannya didasarkan pada nilai *fitness* dari setiap individu tersebut. Agar optimal, solusi yang lebih pas atau lebih optimal seharusnya memiliki peluang seleksi yang lebih besar. Individu yang memenangkan dari kompetisi akan digunakan sebagai individu yang baru bagi pembangkitan selanjutnya.

3.4.8 Elitism

Selama membuat populasi baru dengan *crossover* dan *mutasi*, kemungkinan akan terjadi kehilangan kromosom terbaik (*best / few best*). Elitism adalah metode yang pertama kali meng-copy-kan kromosom terbaik (*best / few*

best) kedalam populasi baru. Sisanya dikerjakan dengan cara biasa, yaitu melalui seleksi, *crossover* dan mutasi. Elitism dapat secara cepat meningkatkan performansidari Algoritma Genetika karena *elitism* menghindarkan hilangnya hilangnya solusi terbaik (*best / few best*) yang telah ditemukan. Ilustrasi kerja operator ini dapat digambarkan seperti pada gambar 3.5.



Gambar 3.9.^[15]

Pembentukan Next Generation dalam Algoritma Genetika

Sumber : *Neuro-Fuzzy and soft Computing*" Reference book :Jang et al

3.5 Formulasi Masalah Optimasi Daya Reaktif ^[1]

Permasalahan perencanaan optimasi daya reaktif adalah untuk meminimalkan biaya operasional, dengan persamaan sebagai berikut: (Kwang. Y. Lee, and Frank F. Yang 1998 : 104):

$$C_Q = C_p + C_i(Q_i) \dots\dots\dots(3.8.)$$

$$C_p = \sum_{i \in N_i} d_i C_F^i \dots\dots\dots(3.9.)$$

dimana:

$C_i(Q_i)$ = Biaya daya reaktif pembangkitan.

C_F^i = Biaya oprasi daya nyata pada tingkatan i beban

d_i = Jangka waktu tingkatan i beban

N_i = Banyaknya tingkatan beban

dengan subyek:(Kwang. Y. Lee, and Frank F. Yang 1998 : 104):

$$\underline{P}_g \leq P_g \leq \bar{P}_g \quad (\text{Daya nyata generator})$$

$$\underline{Q}_g \leq Q_g \leq \bar{Q}_g \quad (\text{Daya reaktif generator})$$

$$\underline{T}_k \leq T_k \leq \bar{T}_k \quad (\text{Tap-Setting transformer})$$

$$\underline{V}_i \leq V_i \leq \bar{V}_i \quad (\text{Tegangan bus beban})$$

Tegangan bus beban dan daya reaktif generator merupakan variabel keadaan, yang dapat dibatasi dengan menambahkannya sebagai bentuk penalti kuadrat ke fungsi obyektif sebagai berikut(Kwang. Y. Lee, and Frank F. Yang 1998 : 104):

$$C_Q = C_i + \sum_{i \in N_f} d_i \left\{ C_F + \sum_{i \in N_f} \lambda_{vi} (V_i - \text{Sat}(V_i))^2 + \sum_{i \in N_g} \lambda_{gi} (Q_{gi} - \text{Sat}(Q_{gi}))^2 \right\}. \quad (3.10.)$$

Dimana; $\text{sat}(x)$ adalah fungsi saturasi sebagai berikut:(Kwang, Y. Lee, and Frank

F. Yang 1998 : 104):

$$\text{Sat}(x) = \begin{cases} \underline{x} & \text{if } x < \underline{x} \\ x & \text{if } \underline{x} \leq x \leq \bar{x} \\ \bar{x} & \text{if } x > \bar{x} \end{cases}$$

3.6 Adaptasi Kombinasi Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm Ke Masalah Optimasi Daya Reaktif

Kombinasi *Evolutionary Programming - Evolutionary Strategy - Genetic Algorithm* adalah suatu mekanisme seleksi alam. Individu dari sebuah populasi dikodekan secara real, Populasi pertama dibangkitkan secara random. Generasi baru dibuat dengan mengaplikasikan operator berikut terhadap sebuah populasi yaitu : inisialisasi, rekombinasi, mutasi dan kompetisi dimana tingkat pengoptimalan setiap calon solusi atau individu diukur dengan ketepatannya tergantung pada fungsi tujuan (*objective function*) permasalahan

3.6.1 Representasi Solusi^[1]

Individu dalam populasi menunjukkan calon pada solusi permasalahan. Elemen solusi terdiri dari daya pembangkit yang ditetapkan pada semua bus generator (PV) selain dari bus slack, magnitude tegangan yang ditetapkan pada semua bus generator (PV).

3.6.2 Initial Population Generation (Inisial populasi generasi)^[1]

Di dalam rencana suatu populasi awal bilangan biner diciptakan secara acak. dari string menghadirkan satu solusi mungkin kepada rumusan masalah ,i e., di dalam pencarian suatu daerah yang menghasilkan ketidaksesuaian.

3.6.3. Evaluasi.Fitness^[1] ;

Berikutnya string solusi diubah jadi sistim desimal dan setiap calon solusi diuji dalam lingkungan nya . kebenaran dari tiap calon dievaluasi melalui/sampai

beberapa ukuran yang sesuai, begitu juga sebagai kebalikan menyangkut fungsi biaya : (Kwang. Y. Lee, and Frank F. Yang 1998 : 103).

$$f = 1 / (\alpha + c) \dots\dots\dots (3.11)$$

Jika C adalah biaya fungsional untuk diperkecil dan α adalah suatu fungsi parameter. Algoritma dikemudikan untuk memaksimalkan ukuran fitness ini. Setelah fitness keseluruhan populasi telah ditentukan ya atau tidaknya ukuran penghentian jumlah.dari hal hal manapun. satu. Kemungkinan akan stop algoritma pada beberapa jumlah terbatas dan mengambil hasil sebagai yang terbaik cocok dari populasi itu. Kemungkinan yang lain akan menguji rata-rata fitness dari populasi melebihi beberapa pecahan menyangkut yang terbaik cocok di populasi itu. Jika ukuran adalah satisfiet kemudian kita melanjutkan dengan ke tiga operasi *reproduksi* , *crossorver*, dan *mutasi*

3.6.4 Selection dan Reproduction^[1]

Fitness-yang sebanding *reproduksinya* diakibatkan putaran yang ditirukan dari suatu rolet dihargai kemudi rolet itu. kemudi dibiaskan dengan fitness dari tiap calon solusi. Kemudi adalah spun N di mana N adalah banyaknya string di (dalam) populasi itu. Operasi ini menghasilkan suatu populasi string yang baru yang mencerminkan fitness dari generasi sebelumnya

3.6.5 Crossover^[1]

Berikutnya operasi *crosover* ,di lakukan pada dua string serentak terpilih dari populasi secara acak. *Crossover* melibatkan suatu posisi acak di dalam kedua

string dan menukar bit yang terjadi setelah posisi ini. Di dalam satu generasi crossover operasi dilakukan pada persentase yang ditetapkan pada population. Crossover dapat terjadi pada single position, atau pada jumlah posisi berbeda. Crossover dapat juga dilakukan di dua alat-alat yang berbeda; head-tail crossover (10,13) adalah crossover yang umum, jika bagian paling akhir dari kedua string berubah. Di dalam head tail crossover, pada sisi lain, bagian paling akhir satu string menjadi head dan sebaliknya. string yang lain. head-tail crossover bit berubah semakin sedikit. Dua metoda crossover dapat diubah selama perhitungan berulang-ulang; head-tail crossover menggunakan ketel masak generasi dan yang switch ke tail-tail crossover di (dalam) generasi yang kemudian untuk penyetelan yang baik

3.6.6 Statistics (Statistik)^[11]

Fitness maksimum (f_{max}), fitness minimum (f_{min}), Jumlah fitness $\sum f$, dan fitness rata-rata (f_{avg}) dari generasi dihitung.

3.6.7 Mutation^[11],

3.6.7.1 Evolutionary Programming

Operator mutasi digunakan untuk memodifikasi satu atau lebih nilai gen dalam satu individu. Cara kerjanya dengan membangkitkan sebuah nilai random r_k dimana $k = 1, 2, \dots, NVAR$ (panjang kromosom). Probabilitas mutasi (P_m) ditentukan dan digunakan untuk mengendalikan frekuensi operator mutasi. Apabila nilai random $r_k < P_m$ maka gen ke-k kromosom tersebut terpilih untuk

mengalami mutasi. Proses mutasi dalam *Evolutionary Programming* sama dengan *Evolutionary Strategies* yaitu menggunakan operator *Gaussian mutation*, dimana setiap individu akan terpilih secara acak untuk mengalami mutasi berdasarkan nomor acak *Gaussian* untuk menciptakan individu baru (*offspring*).

$$\text{a) } \boxed{1.3} \boxed{0.4} \boxed{1.8} \boxed{0.2} \boxed{0.0} \boxed{1.0} \Rightarrow \text{b) } \boxed{1.2} \boxed{0.7} \boxed{1.6} \boxed{0.2} \boxed{0.1} \boxed{1.2}$$

Gambar 3.1.^[10]

Mutasi Gaussian Dari Induk (*parent*) (a) Menghasilkan Anak (*offspring*) (b)

Fungsi dari operator mutasi adalah untuk menghindari agar solusi masalah yang diperoleh bukan merupakan solusi optimum lokal. Tipe dan implementasi dari operator mutasi bergantung pada jenis pengkodean dan permasalahan yang dihadapi. Seberapa sering mutasi dilakukan dinyatakan dengan suatu probabilitas mutasi (P_m). Posisi elemen pada kromosom yang akan dimutasi ditentukan secara random. Mutasi dikerjakan dengan cara melakukan perubahan pada elemen tersebut.

3.6.7.2 Genetic Algorithm

Di akhir genetik operator dalam algoritma adalah mutasi. Mutasi dilakukan dengan hemat, secara khas setelah tiap-tiap 100-1000 bit memindahkan dari crossover, dan melibatkan pemillihan suatu nilai string secara acak juga sedikit memposisikan secara acak dan mengubahnya dari 1 ke 0 atau vice-versa. Digunakan untuk tidak mencapai nilai minimum. Setelah mutasi, generasi baru

dilengkapi / menyudahi dan prosedur mulai lagi dengan evaluasi fitness dari populasi.

3.6.7.3 Evolutionary Strategy

Di dalam proses mutasi, setiap terpilihnya orang tua, misalnya P_i dimutasikan dan ditambahkan ke populasinya, sesuai persamaan berikut:

$$P_{i+m,j} = P_{i,j} + N(0, \beta \nabla_{dev}), \quad j = 1, 2, \dots, n, \dots \dots \dots (3.12)$$

dimana:

$P_{i,j}$ = Elemen ke j dari individu ke i

β = Skala mutasi ($0 < \beta < 1$)

∇_{dev} = Variabel deviasi

3.6.8 Competition (Kompetisi)^[1]

Di dalam proses kompetisi, m *parent* akan bersaing dengan m *offspring*. Dan m individu yang pertama dijaga sebagai orang tua untuk proses generasi selanjutnya. Nilai berat W_i dari individu yang ke i dihitung menurut kompetisi berikut ini:

$$W_i = \sum_{t=1}^N W_{i,t} \dots \dots \dots (3.13)$$

dimana N adalah nomor kompetisi generasi secara acak. $W_{i,t}$ untuk yang manapun, 0 untuk kalah dan 1 untuk menang sebagai individu ke i yang bersaing

dengan individu ke r yang dipilih secara acak di dalam populasi yang dikombinasikan itu. Nilai $W_{i,r}$ disampaikan sesuai persamaan berikut :

$$W_{i,r} = \begin{cases} 1 & \text{jika } U_i < \frac{f_r}{f_r + f_i} \\ 0 & \text{jika sebaliknya} \end{cases} \dots\dots\dots (3.14)$$

dimana :

f_r = Fitness dari individu ke r yang dipilih secara acak

f_i = Fitness dari individu ke i

U_i = Set yang terpilih secara acak, $U(0,1)$

Ketika semua $2m$ individu mendapat nilai W_i kompetisinya, mereka akan diatur dalam order turunan menurut nilai W_i yang bersesuaian. Dan m individu yang pertama dipilih dengan fitness f_i menjadi dasar untuk generasi selanjutnya. Kemudian fitness maksimum, minimum, dan fitness rata-rata serta jumlah fitness dari generasi ini dihitung dalam statistik.

3.6.9 Convergence Test (Konvergensi)¹¹

Jika kondisi konvergen tidak dijumpai, proses mutasi dan kompetisi akan berjalan terus dan nomor generasi maksimum dapat digunakan sebagai kondisi konvergen. Kriteria lain, seperti rasio dari fitness rata-rata dan maksimum dari populasi dihitung berulang-ulang sampai:

$$\{f_{avg} / f_{max}\} \geq \delta \dots\dots\dots (3.15)$$

dimana, δ harus mendekati 1. Jika penelitian telah mencapai ketelitian, suatu solusi optimal telah ditemukan untuk optimasi masalah.

BAB IV

HASIL DAN ANALISA HASIL

4.1. Program Komputer Optimasi Daya Reaktif Menggunakan Metode Kombinasi *Evolutionary Programming - Evolutionary Strategy- Genetic Algorithm*

Dalam penyelesaian masalah ini maka diperlukan bantuan program komputer dalam perhitungan yang membutuhkan ketelitian dan keakuratan.

Program komputer dalam skripsi ini dijalankan dengan menggunakan bahasa pemrograman Borland Delphi versi 6.0 dan diaplikasikan pada komputer Pentium 3 dengan Prosesor 667 MHz, Memori 128 Mb.

4.2. Algoritma Program

4.2.1 Algoritma program dasar permasalahan di lakukan dengan langkah-langkah sebagai berikut:

1. Memasukkan inputan data pembangkitan $\{P_{max}, P_{min}, \text{konstanta biaya}\}$, data beban yang meliputi $\{\text{tegangan } (v), \text{ sudut fasa tegangan } (\delta), \text{ daya aktif } (P), \text{ daya reaktif } (Q)\}$, dan data impedansi saluran.
2. Lakukan proses *load flow newton raphson* dan proses metode kombinasi *Evolutionary Programming-Evolutionary Strategy-Genetic Algorithm*.
3. Apakah hasil perhitungan sudah optimal.
4. Jika "Tidak" kembali kelangkah 2.

5. Jika “Ya” maka lanjutkan kelangkah 6.
6. Hitung total biaya pembangkitan.
7. Berhenti.

4.2.2 Algoritma Program Perencanaan Optimasi Daya Reaktif dengan Menggunakan *Evolutionary Programming - Evolutionary Strategy-Genetic Algorithm*:

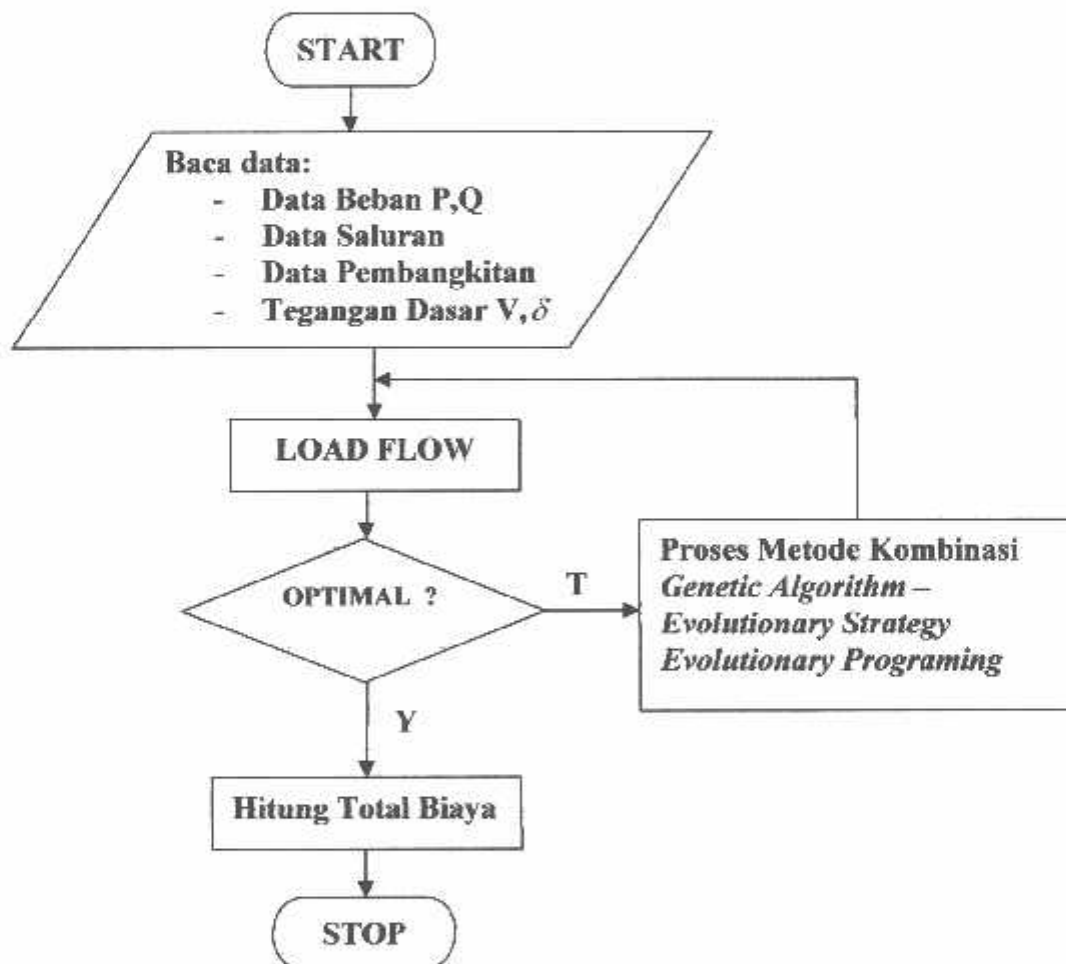
1. Memasukkan inputan data beban yang meliputi tegangan, sudut fasa tegangan, daya aktif, daya reaktif, dan data impedansi saluran.
 2. Menentukan parameter inputan kombinasi EP, ES, GA yang meliputi jumlah populasi maksimum generasi, nilai kemungkinan mutasi dan panjang kromosom tiap-tiap individu.
 3. Generasi =0, populasi=0
 4. Melakukan fitness dari kromosom tiap-tiap individu
 5. Melakukan proses statistik
 6. Melakukan proses seleksi
 7. Melakukan proses crossover
 8. Melakukan proses mutai GA
 9. Proses nomor 6,7,8 diulang sampai offspring sampai jumlah populasi max
 10. Menghitung fitness dari offspring.
 11. Melakukan proses kompetisi evolutionary strategy dan evolutionary programing.
-

12. Melakukan proses elitism.
13. Apakah generasi yang diinginkan sudah terpenuhi (maxgen)
14. Jika “tidak” maka generasi =gen+1,kembali ke langkah 6
15. Jika “ya” maka cetak hasil
16. Berhenti

4.2.3 Sedangkan untuk algoritma aliran daya dengan metode *Newton Rapshon* adalah sebagai berikut :

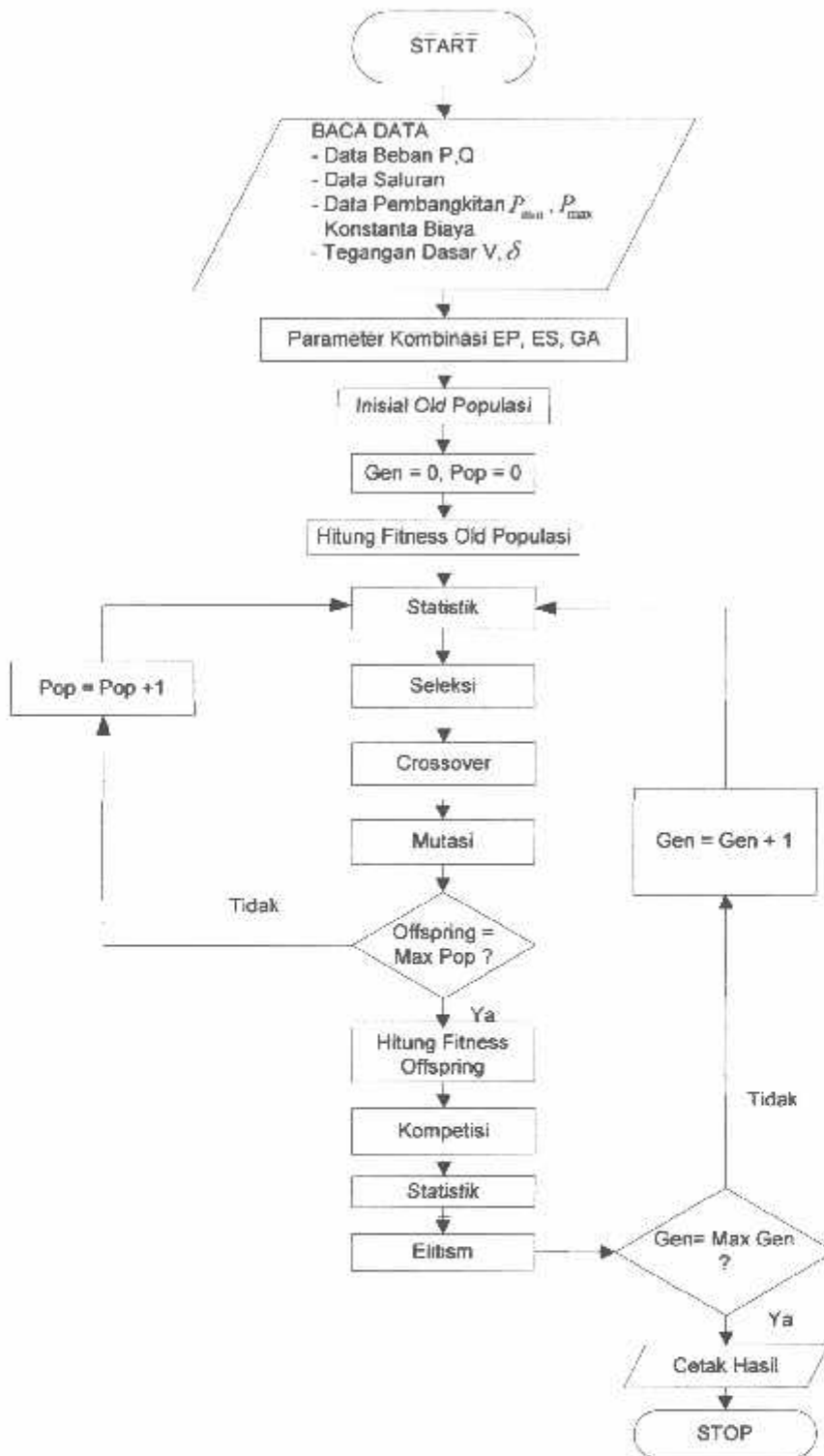
1. Bentuk matriks admitansi bus : $[Y_{bus}]$
 2. Penetapan harga awal tegangan dan sudut fasa untuk semua bus kecuali bus slack $[V_i(0), \delta(0)]$
 3. Menentukan nomor iterasi awal, $k=0$
 4. Menghitung injeksi daya aktif dan reaktif pada setiap bus dengan persamaan kecuali bus slack
 5. Menghitung selisih daya yang dijadwalkan dengan injeksi daya bus dari perhitungan
 6. Menentukan perubahan maksimum pada daya aktif dan daya reaktif
 7. Membandingkan apakah selisih daya sudah sama atau lebih kecil dari ϵ
 8. Jika “ya” hitung daya aktif dan reaktif, tegangan dan sudut fasa tegangan pada setiap bus, serta aliran daya pada saluran dan perhitungan selesai, jika “tidak” lanjutkan ke langkah berikutnya
 9. Membentuk elemen *matriks Jacobian*
-

10. Menghitung faktor koreksi tegangan dan sudut fasa setiap bus kecuali bus slack dan bus generator
11. Menghitung nilai tegangan dan sudut fasa yang baru
12. Mengganti nilai sudut fasa yang lama dengan sudut fasa yang baru, tegangan yang lama dengan tegangan yang baru
13. Perhitungan dilanjutkan ke langkah 4 dengan nilai iterasi yang baru sampai hasil yang didapatkan konvergen.



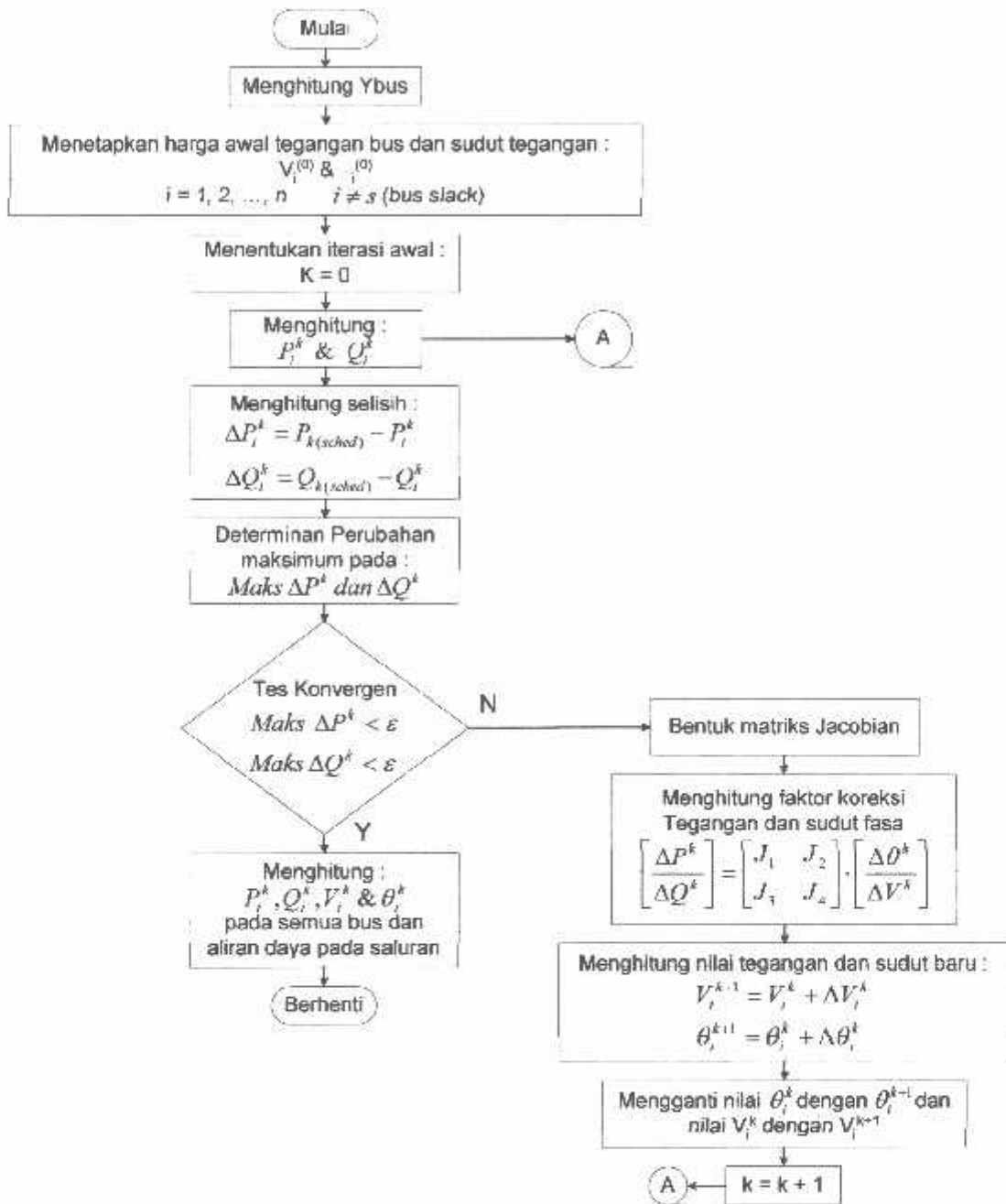
Gambar 4.1

Flowchart Penyelesaian Masalah



Gambar 4.2

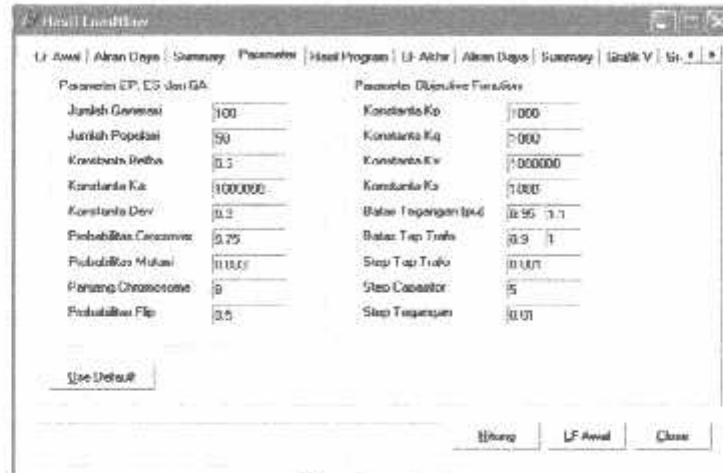
Flowchart Program Menggunakan Metode Evolutionary programming-
Evolutionary Strategy-Genetik Algorithm



Gambar 4.3.
Flowchart Perhitungan Aliran Daya Metode Newton Rapshon

4.3. Hasil Validasi IEEE 30 Dengan Menggunakan Metode *Kombinasi Evolutionary Programming - Evolutionary Strategi - Genetic Algorithm*

1. Tampilan Parameter Validasi



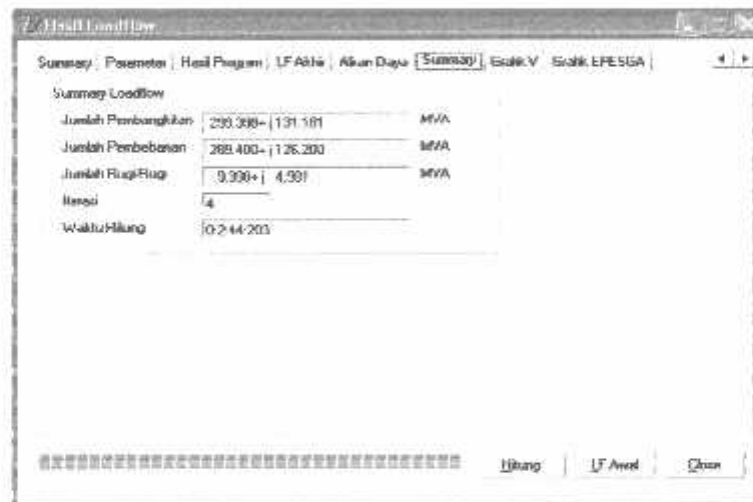
Gambar 4.4.
Tampilan Parameter Kombinasi *EP-ES-GA* Validasi IEEE 30

Bus	Load (kW)	Load (kVar)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sags (rad)	Type Bus
1	0.0000	0.0000	186.398	1.996	0.000	0.000	0.000	1
2	1.06000	-0.05261	40.000	10.385	27.700	12.700	0.000	2
3	1.04995	-0.05959	0.000	0.000	2.400	1.200	0.000	3
4	1.03751	-0.11454	0.000	0.000	7.500	1.000	0.000	2
5	1.02600	-0.15273	25.000	40.788	34.200	19.000	0.000	2
6	1.03464	-0.13465	0.000	0.000	0.800	0.000	0.000	3
7	1.02145	-0.16303	0.000	0.000	22.000	10.900	0.000	3
8	1.04000	-0.14064	22.000	47.630	30.000	30.000	0.000	2
9	1.01351	-0.17282	0.000	0.000	0.800	0.000	0.000	3
10	0.99470	-0.20579	0.000	0.000	5.800	2.000	0.000	3
11	1.03000	-0.15000	11.000	0.204	0.000	0.000	0.000	2
12	1.01156	-0.15247	0.000	0.000	11.200	7.500	0.000	3
13	1.03000	-0.17735	12.000	13.989	0.000	0.000	0.000	2
14	0.99646	-0.21039	0.000	0.000	4.200	1.600	0.000	3
15	0.99062	-0.21192	0.000	0.000	0.200	2.500	0.000	3

Gambar 4.5.
Hasil *Load Flow*



Gambar 4.6.
Tampilan Hasil Validasi IEEE 30
Hasil Perhitungan Biaya Pembangunan



Gambar 4.7
Tampilan Hasil Summary Load Flow

Tabel 4.1.^[1]
Perbandingan Hasil Data Referensi Jurnal Dengan Data Optimasi

No.	No. Bus	Data Referensi		Data Optimasi	
		P_{Gen} (MW)	Q_{Gen} (MVar)	P_{Gen} (MW)	Q_{Gen} (MVar)
1	1	179,162	1,409	186,398	1,996
2	2	48,345	10,827	48,000	10,885
3	5	19,994	47,753	20,000	48,788
4	8	21,790	47,890	22,000	47,638
5	11	11,423	8,064	11,000	8,284
6	13	12,071	13,199	12,000	13,589

Setelah dilakukan proses optimasi maka diperoleh biaya pembangkitan dari data IEEE 30 yaitu sebesar 618,00 \$/hr, sedangkan dari data referensi jurnal diperoleh biaya pembangkitan sebesar 612,16 \$/hr. Dari proses validasi di atas maka didapatkan error sebesar 0,94 %

✓

4.4. Data Pembangkitan Thermal Pada Sub-Sistem Paiton-Bali

Pada skripsi ini akan membahas pembangkit *thermal* yang berada pada Sub-Sistem Paiton-Bali. Pembangkit *thermal* yang dibahas dalam skripsi ini adalah PLTU Paiton 5-6, PLTG Gilimanuk, PLTD Pesanggaran, dan PLTG Pesanggaran.

Untuk data dari bentuk karakteristik semua unit pembangkit yang dibahas dalam skripsi ini beserta kapasitasnya dan *fuel cost* (biaya bahan bakar) yang digunakan dalam perhitungan adalah berdasarkan data dari PT. Indonesia Power dan PT. PJB.

Tabel 4.2.^[17]
Parameter Unit Pembangkit Thermal

Nama Pembangkit	a2	a1	a0	Pmin (MW)	Pmax (MW)
PLTU Paiton 5&6	6.18000	1306.15000	388144.16800	150	740
PLTG Gilimanuk	1.40690	1599.00000	87435.00000	50	133,8
PLTD Pesanggaran	14.29320	1670.00000	88960.00000	21	75
PLTG Pesanggaran	113.90000	5297.00000	139560.00000	15	125,5

Dengan memasukkan data-data pada tabel 4.5. ke persamaan fungsi biaya bahan bakar, maka untuk unit pembangkit thermal PLTU Paiton 5&6 diperoleh sebagai berikut :

Biaya bahan bakar pembangkit^[1]:

$$F_i(G_i) = a_2 P_{Gi}^2 + a_1 P_{Gi} + a_0$$

$$F_i(G_i) = 6.18000 P^2 + 1306.15000 P + 388144.16800$$

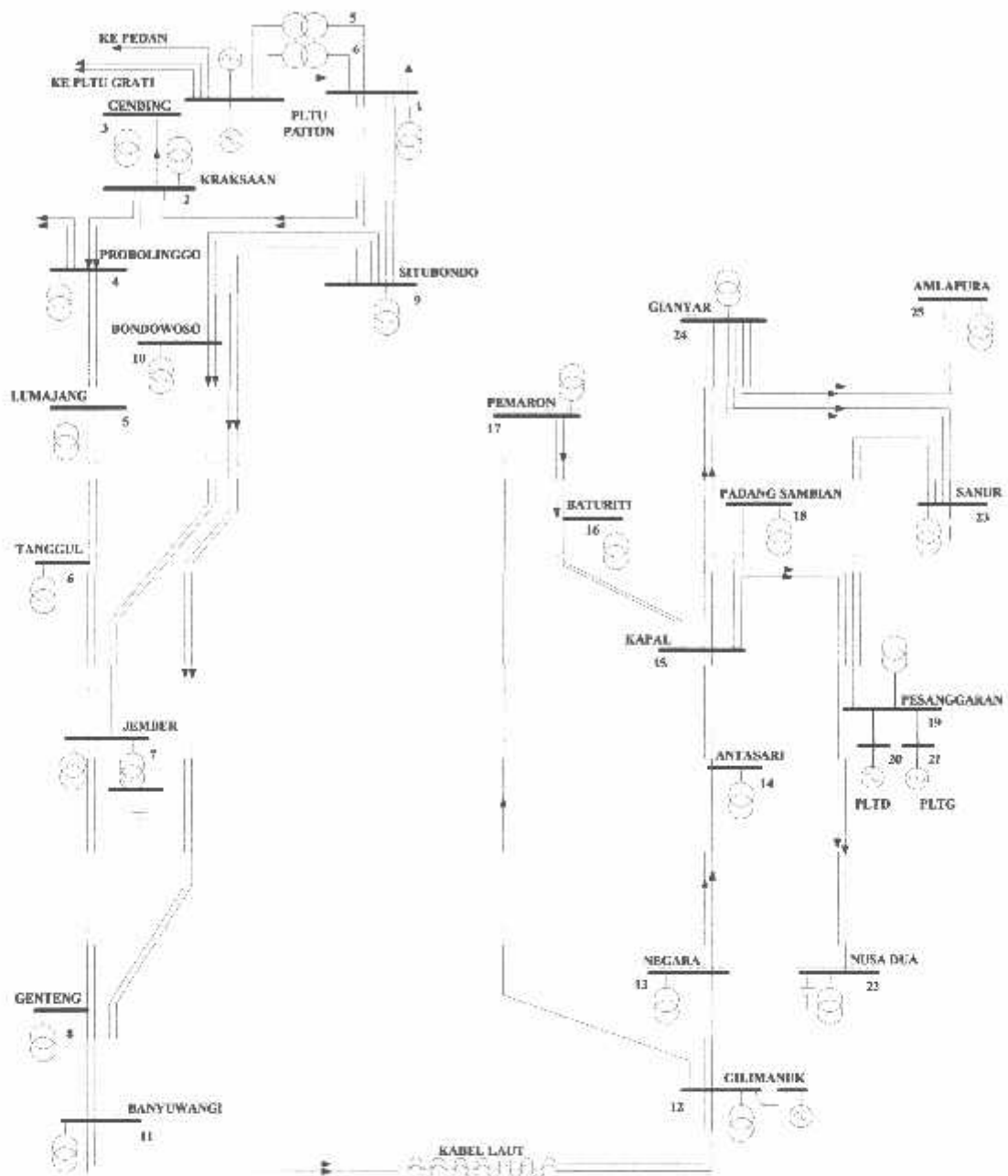
Sedangkan untuk biaya pemeliharaan dari setiap pembangkit yang merupakan biaya pengoperasian daya reaktif, dalam skripsi ini berdasarkan

referensi maka diset sebesar U.S \$ 6,77 / Mvar perjam^[2]. Kurs dollar yang dipakai sebesar Rp 9.500, jadi biaya pengoperasian daya reaktif dari masing-masing pembangkit sebesar Rp 64315 / MVar perjam.

Untuk persamaan biaya pembangkitan dari masing-masing unit pembangkit yang berdasarkan data-data diatas adalah sebagai berikut:

Tabel 4.3.^[17]
 Persamaan Biaya Pembangkitan
 Unit Pembangkit Thermal Paiton dan Bali

No.	Nama Pembangkit	Persamaan Biaya Bahan Bakar (Rp/Jam)	Persamaan Biaya Daya Reaktif (Rp/jam)
1	PLTU Paiton 5&6	$F = 276,246 P^2 + 58384,905 P + 17350044,31$	$C_r(Q_r) = 64315 (Q_r)$
2	PLTG Gilimanuk	$F = 1.40690 P^2 + 1599.00000 P + 87435.00000$	$C_r(Q_r) = 64315 (Q_r)$
3	PLTDPesanggaran	$F = 14.29320 P^2 + 1670.00000 P + 88960.00000$	$C_r(Q_r) = 64315 (Q_r)$
4	PLTGPesanggaran	$F = 113.90000 P^2 + 5297.00000 P + 139560.00000$	$C_r(Q_r) = 64315 (Q_r)$



Gambar 4.8^[13]
 Diagram Segaris Jaringan Sistem Tenaga Listrik 150 kV Sub-Sistem
 Paiton-Bali

Tabel 4.4.^[13]

Penomoran Bus 150 kV Sub-Sistem Paiton-Bali

No. Bus	Nama Bus
1	PAITON
2	KRAKSAAN
3	GENDING
4	PROBOLINGGO
5	LUMAJANG
6	TANGGUL
7	JEMBER
8	GENTENG
9	SITUBONDO
10	BONDOWOSO
11	BANYUWANGI
12	GILIMANUK
13	NEGARA
14	ANTASARI
15	KAPAL
16	BATURITI
17	PEMARON
18	PADANG SAMBIAN
19	PESANGGARAN
20	PLTD PESANGGARAN
21	PLTG PESANGGARAN
22	NUSA DUA
23	SANUR
24	GIANYAR
25	AMPLAPURA

4.6. Data Saluran Transmisi 150 kV Sub-Sistem Paiton-Bali

Pada Sub-Sistem Paiton-Bali terdiri dari 25 bus dan 34 saluran transmisi, dalam hal ini saluran transmisi yang dibahas adalah saluran transmisi 150 kV.

Berdasarkan data pada referensi [15], maka dilakukan pengolahan data untuk saluran dengan jumlah sirkuit 2. Sebagai contoh pada hubungan saluran 1-2 dilakukan perhitungan sebagai berikut :

$$R = \frac{R_1 \times R_2}{R_1 + R_2} = \frac{0.0105 \times 0.0105}{0.0105 + 0.0105} = \frac{1.1025 \times 10^{-4}}{0.021} = 0.0053 \text{ pu}$$

$$X = \frac{X_1 \times X_2}{X_1 + X_2} = \frac{0.0359 \times 0.0359}{0.0359 + 0.0359} = \frac{1.2888 \times 10^{-3}}{0.0718} = 0.0180 \text{ pu}$$

$$B = \frac{B_1 \times B_2}{B_1 + B_2} = \frac{0.0130 \times 0.0130}{0.0130 + 0.0130} = \frac{1.69 \times 10^{-4}}{0.026} = 0.0065 \text{ pu}$$

Untuk selanjutnya dilakukan perhitungan seperti langkah di atas, sehingga didapatkan hasil sebagai berikut :

Tabel 4.6.^[15]
Data Saluran Transmisi 150 kV Sub-Sistem Paiton-Bali

No	Hubungan Bus	Jumlah Sirkuit	R (pu)	X (pu)	B (pu)
1	1 - 2	2	0,0053	0,0180	0,0065
2	1 - 9	2	0,0048	0,0346	0,0251
3	2 - 3	1	0,0100	0,0340	0,0123
4	2 - 4	2	0,0079	0,0269	0,0097
5	4 - 5	2	0,0135	0,0145	0,0166
6	5 - 6	1	0,0167	0,0571	0,0206
7	5 - 7	1	0,0315	0,1077	0,0389
8	6 - 7	1	0,0148	0,0506	0,0189
9	7 - 8	1	0,0337	0,1151	0,0415
10	7 - 10	2	0,0104	0,0353	0,0128

11	7 – 11	1	0,0430	0,1469	0,0530
12	8 – 11	1	0,0177	0,0604	0,0218
13	9 – 10	2	0,0091	0,0310	0,0112
14	9 – 11	2	0,0097	0,0464	0,0340
15	11 – 12	2	0,0009	0,0013	0,0364
16	12 – 13	2	0,0116	0,0336	0,0124
17	12 – 17	1	0,0231	0,0690	0,0476
18	13 – 14	1	0,0270	0,0783	0,0288
19	13 – 15	1	0,0412	0,1194	0,0439
20	14 – 15	1	0,0412	0,0411	0,0151
21	15 – 16	1	0,0232	0,0673	0,0247
22	15 – 17	1	0,0556	0,1054	0,0369
23	15 – 18	1	0,0132	0,0383	0,0141
24	15 – 19	1	0,0105	0,0304	0,0112
25	15 – 22	1	0,0199	0,0379	0,0130
26	15 – 24	2	0,0059	0,0170	0,0063
27	16 – 17	1	0,0183	0,0350	0,0120
28	18 – 19	1	0,0046	0,0134	0,0049
29	19 – 20	1	0,0020	0,0044	0,0000
30	19 – 21	1	0,0020	0,0056	0,0000
31	19 – 22	1	0,0127	0,0243	0,0083
32	19 – 23	2	0,0029	0,0069	0,0024
33	23 – 24	2	0,0061	0,0147	0,0052
34	24 – 25	2	0,0103	0,0298	0,0110

4.7. Prosedur Pelaksanaan Program Perhitungan

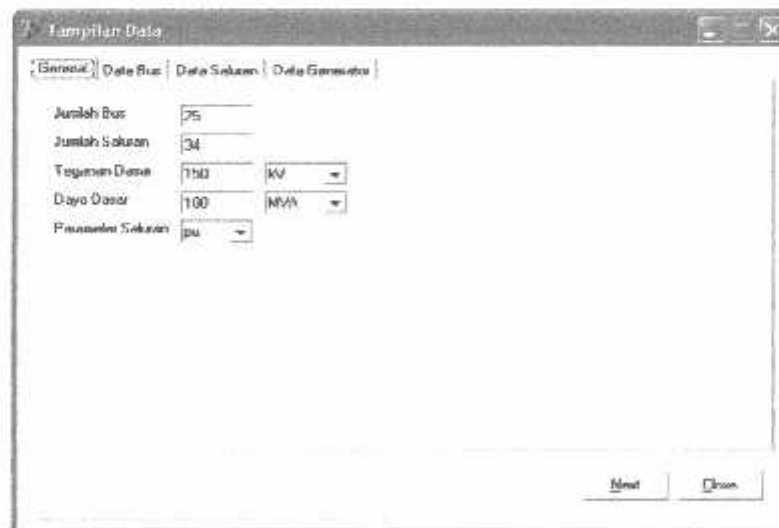
Prosedur menjalankan program perhitungan dengan menggunakan bahasa pemrograman Borland Delphi versi 6.0 dapat dilakukan sebagai berikut :

1. Tampilan utama dari program.



Gambar 4.9.
Tampilan Utama Program

2. Tekan tombol *General* untuk membuka data yang sudah tersimpan.



Gambar 4.10.
Tampilan Masukan Data

3. Kemudian tekan tombol data bus

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Cap (pu)	Type Bus
1	1	0	0	0	33.3	7.9	0	1
2	1	0	0	0	14	6	0	3
3	1	0	0	0	18	8.5	0	3
4	1	0	0	0	36.1	16.5	0	3
5	1	0	0	0	38	16.6	0	3
6	1	0	0	0	22.5	9.7	0	3
7	1	0	0	50	22.2	30	0	3
8	1	0	0	0	41	21.3	0	3
9	1	0	0	0	23	5.9	0	3
10	1	0	0	0	17.2	7	0	3
11	1	0	0	0	37.8	18.7	0	3
12	0.95	0	99.5	0	5.4	2.7	0	2
13	1	0	0	0	11.9	4.4	0	3
14	1	0	0	0	6.2	2.9	0	3
15	**	**	**	**	**	**	**	**

Gambar 4.11.
Tampilan Data Bus

4. Kemudian tekan tombol data saluran

No	Dari	Ke	R (pu)	X (pu)	Lc (pu)	Tr	Tu	Sud (deg)	Kap (MVA)
1	2	3	0.0093	0.0179	0.0065	0	0	0	222
2	1	5	0.0048	0.0345	0.0251	0	0	0	400
3	2	3	0.01	0.034	0.0123	0	0	0	111
4	2	4	0.0079	0.0263	0.0097	0	0	0	222
5	4	5	0.0135	0.046	0.0166	0	0	0	222
6	5	6	0.0167	0.0571	0.0205	0	0	0	133
7	5	7	0.0115	0.0327	0.0289	0	0	0	133
8	6	7	0.0149	0.0505	0.0183	0	0	0	111
9	7	8	0.0337	0.1151	0.0415	0	0	0	133
10	7	10	0.0104	0.0353	0.0129	0	0	0	222
11	7	11	0.043	0.1463	0.053	0	0	0	133
12	8	11	0.0177	0.0604	0.0213	0	0	0	144
13	9	10	0.0081	0.031	0.0112	0	0	0	222
14	9	11	0.0966	0.3407	0.124	0	0	0	300

Gambar 4.12.
Tampilan Data Saluran

5. Kemudian tekan data generator

No	Bus	Qmax (MVAR)	Qmin (MVAR)	a2	a1	a0	Fix Cost	Var Cost	Pmin
1	1	-200.00	300.00	6.18000	1306.15000	399344.168	1.73	5.70	150.
2	12	45.00	80.00	1.40690	1599.00000	87435.00000	1.03	7.20	50.
3	20	-30.00	60.00	14.25330	1670.00000	88960.00000	1.03	7.20	21.
4	21	15.00	25.00	113.90000	5297.00000	139960.000	1.03	7.20	15.

Gambar 4.13.
Tampilan Data Generator

6. Tekan tombol Parameter untuk melihat Parameter Kombinasi *Evolutionary Programming-Evolutionary Strategy-Genetic Algorithm* Pada kondisi awal (Sebelum Optimasi)

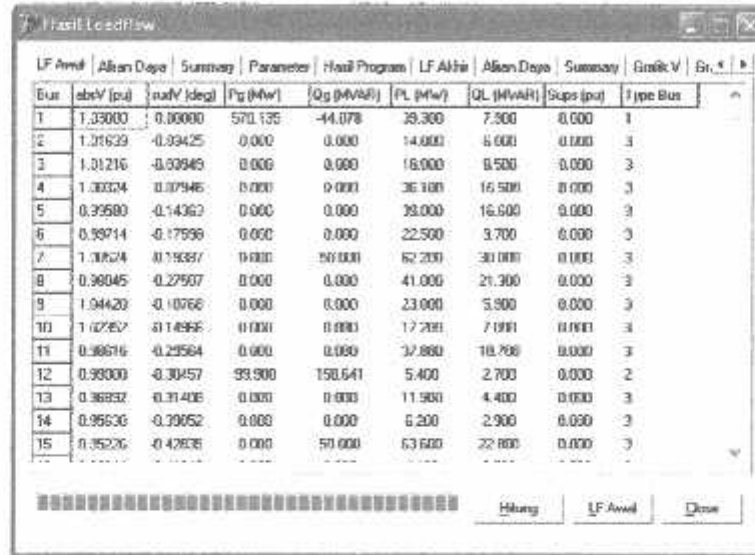
Parameter EP, ES dan GA		Parameter Objective Function	
Jumlah Generasi	100	Konstanta Ka	1000
Jumlah Populasi	50	Konstanta Kp	1000000
Konstanta Beta	0.3	Konstanta Kv	1000
Konstanta Ka	1300000	Batas Tegangan (pu)	0.95 1.05
Konstanta Dev	0.3	Batas Tap Trafo	0.9 1.1
Probabilitas Crossover	0.75	Step Tap Trafo	0.001
Probabilitas Mutasi	0.003	Step Capacitor	5
Paralel Chromosome	8	Step Tegangan	0.01
Probabilitas Flip	0.5		

Use Default

Hitung LF Awal Close

Gambar 4.14.
Tampilan Parameter Kombinasi *EP - ES -GA* pada kondisi awal (Sebelum Optimasi)

7. Tekan *load flow* awal bus untuk melihat hasil perhitungan *Load Flow* pada kondisi awal (sebelum optimasi).

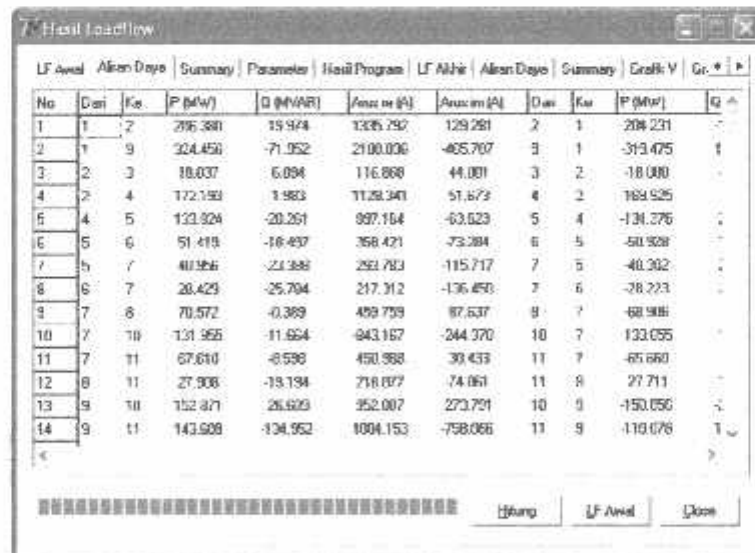


Bus	absV (pu)	radV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Saps (pu)	Type Bus
1	1.33000	0.00000	570.135	-44.078	39.300	7.900	0.000	1
2	1.31639	-0.39425	0.000	0.000	14.000	6.000	0.000	3
3	1.31216	-0.69846	0.000	0.000	16.000	6.500	0.000	3
4	1.30324	-0.37946	0.000	0.000	36.100	16.500	0.000	3
5	0.99580	-0.14362	0.000	0.000	39.000	16.600	0.000	3
6	0.99714	-0.17599	0.000	0.000	22.500	3.700	0.000	3
7	1.38624	0.19397	0.000	0.000	62.200	30.000	0.000	3
8	0.98045	-0.27507	0.000	0.000	41.000	21.300	0.000	3
9	1.04420	-0.10766	0.000	0.000	23.000	5.900	0.000	3
10	1.02762	-0.14966	0.000	0.000	17.200	7.000	0.000	3
11	0.98616	-0.29564	0.000	0.000	37.800	18.700	0.000	3
12	0.98000	-0.30457	99.900	150.641	5.400	2.700	0.000	2
13	0.96932	-0.31406	0.000	0.000	11.900	4.400	0.000	3
14	0.96636	-0.39052	0.000	0.000	6.200	2.900	0.000	3
15	0.95226	-0.42638	0.000	0.000	50.000	63.600	0.000	3

Gambar 4.15.

Tampilan Hasil *Load Flow* Pada Kondisi Awal (Sebelum Optimasi)

8. Tekan aliran daya saluran untuk melihat kondisi aliran daya pada kondisi awal (sebelum optimasi).

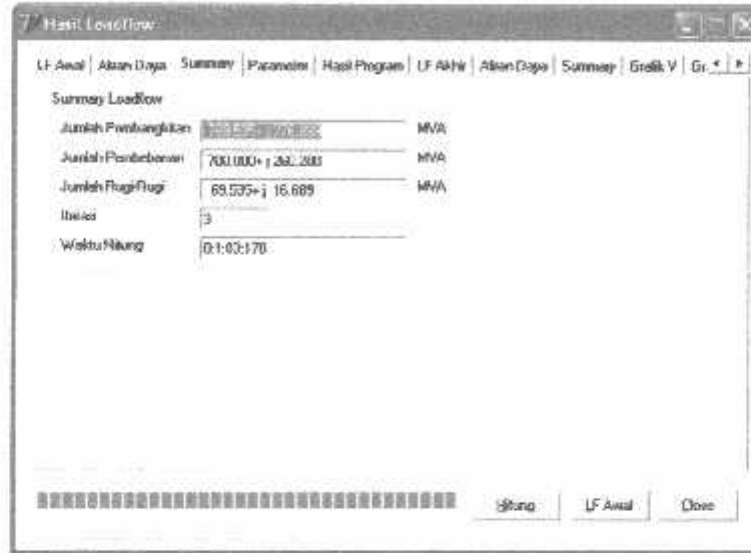


No	Dari	Ko	P (MW)	Q (MVAR)	Amor (A)	Anom (A)	Dari	Ko	P (MW)	Q
1	1	2	296.380	19.974	1306.790	129.281	2	1	296.231	-
2	1	9	324.450	-71.952	2100.030	-405.707	9	1	-313.475	-
3	2	3	18.037	6.094	116.868	44.001	3	2	-18.000	-
4	2	4	172.199	1.983	1129.340	51.672	4	2	169.926	-
5	4	5	132.924	-20.261	997.164	-63.623	5	4	-131.376	-
6	5	6	51.418	-16.497	368.421	-73.284	6	5	-50.928	-
7	5	7	40.986	-21.988	293.783	-115.717	7	5	-40.302	-
8	6	7	20.429	-25.704	217.312	-136.490	7	6	-20.273	-
9	7	8	70.572	-0.369	499.759	87.637	8	7	-68.988	-
10	7	10	131.965	-11.664	943.167	-244.370	10	7	133.055	-
11	7	11	67.610	-8.596	450.988	39.433	11	7	-65.660	-
12	8	11	27.908	-19.194	718.077	74.061	11	8	27.711	-
13	9	10	152.871	26.629	952.007	273.791	10	9	-150.056	-
14	9	11	143.928	-134.952	1094.153	-798.066	11	9	119.076	-

Gambar 4.16.

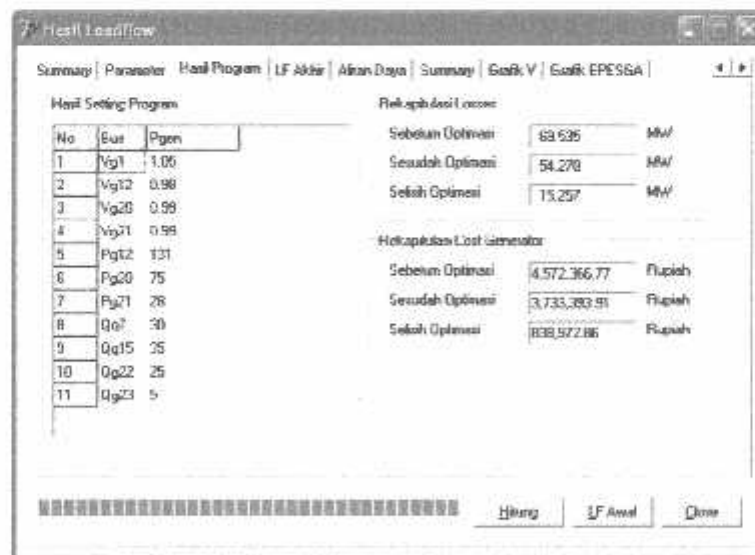
Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi)

9. Tekan tombol *summary*, untuk mengetahui jumlah pembangkitan dan jumlah pembebanan serta rugi – rugi sebelum optimasi



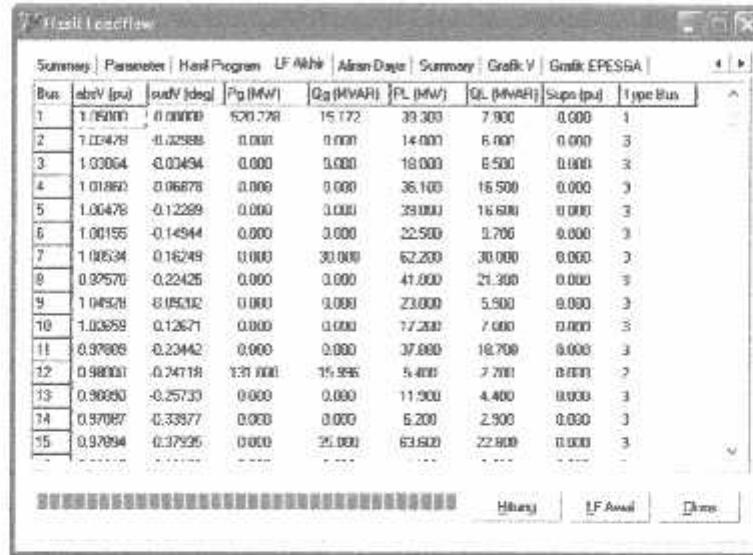
Gambar 4.17.
Tampilan *summary load flow* sebelum optimasi

10. Kemudian tekan tombol Hasil Program, untuk melihat Hasil Program pada kondisi akhir (setelah optimasi)



Gambar 4.18.
Tampilan Hasil Program pada kondisi akhir (Setelah Optimasi)

11. Kemudian tekan tombol *load flow* akhir untuk melihat kondisi *load flow* pada kondisi akhir (setelah optimasi).

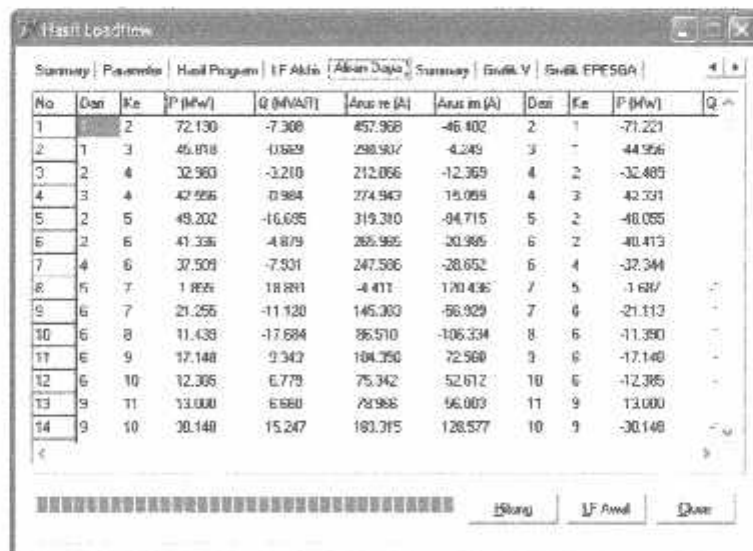


Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Supn (pu)	Type Bus
1	1.05000	0.00000	520.728	15.172	33.303	7.900	0.000	1
2	1.02478	-0.32588	0.000	0.000	14.000	6.000	0.000	3
3	1.03064	-0.00494	0.000	0.000	18.000	6.500	0.000	3
4	1.01862	0.06878	0.000	0.000	36.100	16.500	0.000	3
5	1.00478	-0.12289	0.000	0.000	39.000	16.600	0.000	3
6	1.00155	-0.14944	0.000	0.000	22.500	9.706	0.000	3
7	1.00534	0.16249	0.000	30.000	62.200	30.000	0.000	3
8	0.97570	-0.22425	0.000	0.000	41.000	21.300	0.000	3
9	1.04928	-0.05032	0.000	0.000	23.000	5.900	0.000	3
10	1.03959	0.12671	0.000	0.000	17.200	7.600	0.000	3
11	0.97009	-0.23442	0.000	0.000	37.600	18.700	0.000	3
12	0.98001	-0.24718	130.000	15.996	5.400	7.200	0.000	2
13	0.90990	-0.25733	0.000	0.000	11.900	4.400	0.000	3
14	0.97097	-0.33977	0.000	0.000	6.200	2.900	0.000	3
15	0.97094	-0.37936	0.000	0.000	25.000	63.600	0.000	3

Gambar 4.19.

Tampilan hasil *Load Flow* pada kondisi akhir (Setelah Optimasi)

12. Kemudian tekan tombol aliran daya, untuk melihat aliran daya (setelah optimasi)

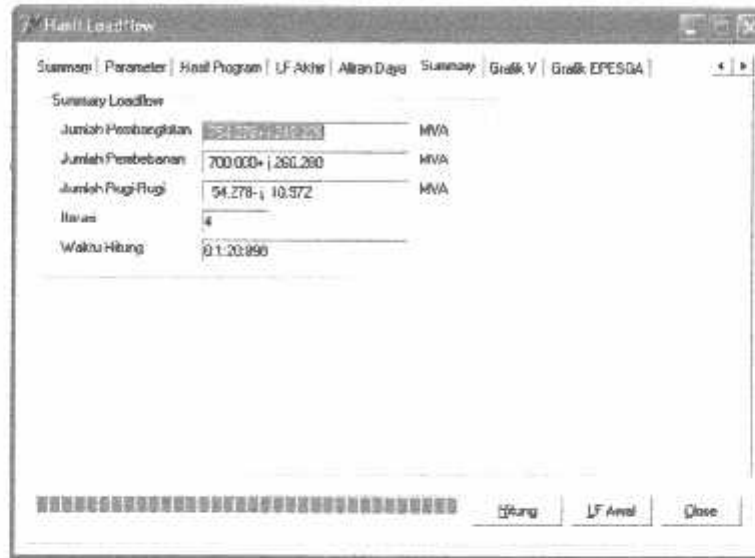


No	Dir	Ke	P (MW)	Q (MVAR)	Ans re (A)	Ans im (A)	Dir	Ke	P (MW)	Q
1	2	2	72.130	-7.308	457.968	-46.402	2	1	-71.221	
2	1	3	46.078	-0.669	298.907	-4.249	3	1	44.956	
3	2	4	32.963	-3.210	212.066	-12.369	4	2	-32.485	
4	3	4	42.996	0.984	274.943	15.089	4	3	42.331	
5	2	5	49.202	-16.635	319.310	-94.715	5	2	-48.055	
6	2	6	41.336	-4.879	265.965	-20.965	6	2	-40.413	
7	4	6	37.509	-7.931	247.596	-28.652	6	4	-32.344	
8	5	7	1.896	18.891	-4.411	170.436	7	5	-1.687	
9	6	7	21.256	-11.120	145.303	-60.929	7	6	-21.113	
10	6	8	11.438	-17.694	86.510	-106.334	8	6	-11.390	
11	6	9	17.148	-9.343	104.250	72.560	9	6	-17.148	
12	6	10	12.305	6.779	75.342	52.612	10	6	-12.385	
13	9	11	13.000	6.680	78.966	56.003	11	9	13.000	
14	9	10	30.148	15.247	181.315	128.577	10	9	-30.148	

Gambar 4.20.

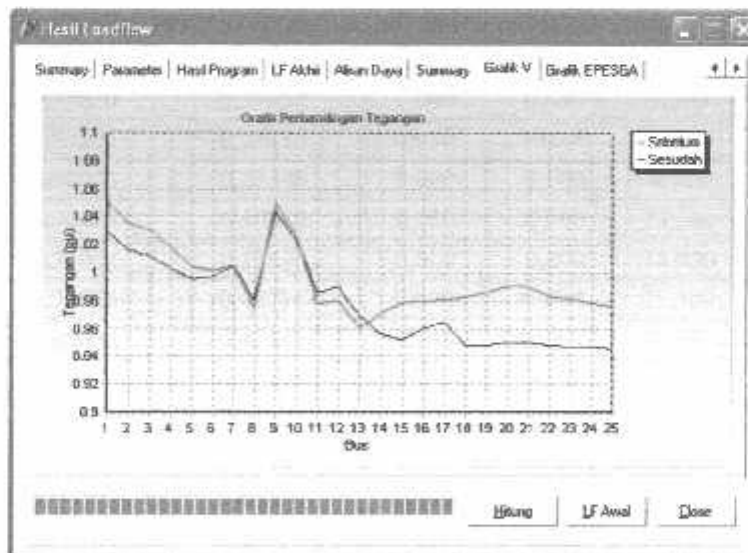
Tampilan aliran daya pada kondisi akhir (Setelah Optimasi)

13. Kemudian tekan tombol *summary*, untuk melihat jumlah pembangkitan dan jumlah pembebanan serta rugi - rugi setelah optimasi



Gambar 4. 21.
Tampilan *Summary Load Flow* setelah optimasi

14. Kemudian tekan tombol *Grafik V*, untuk melihat nilai tegangan sebelum dan setelah optimasi



Gambar 4. 22.
Tampilan nilai tegangan sebelum dan setelah optimasi

Tabel 4.8.
Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi

No	Hubungan Bus	P(MW)	Q(MVAR)	Arus re(A)	Arus im(A)
1	1 – 2	206,380	19,974	1335,792	129,281
2	1 – 9	324,456	-71,952	2100,036	-465,707
3	2 – 3	18,037	6,094	116,868	44,001
4	2 – 4	172,193	1,983	1128,341	51,673
5	4 – 5	133,824	-20,261	897,164	-63,623
6	5 – 6	51,419	-18,497	358,421	-73,284
7	5 – 7	40,956	-23,388	293,783	-115,717
8	6 – 7	28,429	-25,784	217,312	-136,450
9	7 – 8	70,572	-0,389	459,759	87,637
10	7 – 10	-131,855	-11,664	-843,167	-244,370
11	7 – 11	67,610	-8,598	450,968	30,433
12	8 – 11	27,908	-19,194	218,077	-74,061
13	9 – 10	152,871	26,609	952,087	273,791
14	9 – 11	143,609	-134,952	1004,153	-758,066
15	11 – 12	165,658	-182,898	1431,537	-856,515
16	12 – 13	187,221	-24,949	1253,103	217,805
17	12 – 17	71,936	-4,200	470,604	118,292
18	13 – 14	86,646	-13,504	595,708	95,808
19	13 – 15	84,460	-14,700	583,945	83,335
20	14 – 15	78,253	-17,426	550,699	95,333
21	15 – 16	-19,999	-2,834	-119,119	-76,207
22	15 – 17	-21,561	-2,997	-128,593	-81,788
23	15 – 18	24,777	1,767	152,652	83,303
24	15 – 19	28,354	0,516	179,069	85,738
25	15 – 22	23,907	-6,714	171,773	26,763
26	15 – 24	59,465	4,107	366,753	199,077
27	16 – 17	-24,259	0,450	-155,452	-64,794
28	18 – 19	8,305	-3,418	63,046	2,899
29	19 – 20	-32,173	-13,790	-163,699	-183,600
30	19 – 21	-67,200	1,805	-433,088	-188,676
31	19 – 22	21,863	-9,200	166,559	6,565
32	19 – 23	40,771	-8,011	283,361	70,454
33	23 – 24	-9,600	3,804	-72,494	-4,638
34	24 – 25	14,224	1,397	86,372	51,443

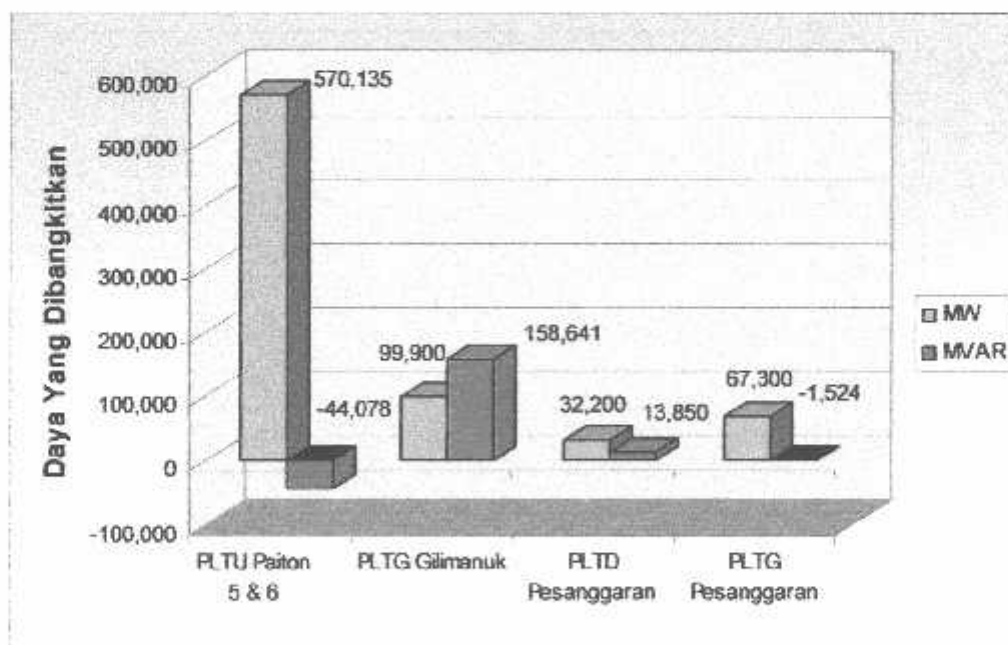
Tabel 4.9.
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum Optimasi

Pembangkitan		Pembebanan		Rugi-rugi		Waktu Komputasi
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	
769.535	276.889	700.000	260.200	69.535	16.689	0:1:03:170

Tabel 4.10.
Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Sebelum Optimasi

No	Nama Pembangkit	Daya Yang Dibangkitkan		Biaya Operasi (Rupiah/jam)
		(MW)	(MVar)	
1	PLTU Paiton 5 & 6	570,135	-44,078	Rp 3.141.662,80
2	PLTG Gilimanuk	99,900	158,641	Rp 261.215,98
3	PLTD Pesanggaran	32,200	13,850	Rp 157.553,76
4	PLTG Pesanggaran	67,300	-1,524	Rp 1.011.934,23
Total				Rp 4.572.366,77

Grafik 4.1
Hasil Perhitungan Daya Yang Dibangkitkan Sebelum Optimasi



4.8.2. Hasil Perhitungan Setelah Optimasi

Tabel 4.11.
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan Dan
Pembebanan Setelah Optimasi

No. Bus	Tegangan (pu)	Sudut Tegangan (deg)	Pembangkitan		Pembebanan	
			Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1	1,05000	0,00000	520,278	15,172	39,300	7,900
2	1,03478	-0,02988	0,000	0,000	14,0	6,0
3	1,03064	-0,03494	0,000	0,000	18,0	8,500
4	1,01860	-0,06878	0,000	0,000	36,100	16,500
5	1,00478	-0,12289	0,000	0,000	39,000	16,600
6	1,00155	-0,14944	0,000	0,000	22,500	9,700
7	1,00534	-0,16249	0,000	30,000	62,200	30,000
8	0,97570	-0,22425	0,000	0,000	41,000	21,300
9	1,04928	-0,09202	0,000	0,000	23,000	5,900
10	1,02659	-0,12671	0,000	0,000	17,200	7,000
11	0,97809	-0,23442	0,000	0,000	37,800	18,700
12	0,98000	-0,24118	131,000	15,996	5,400	2,700
13	0,96090	-0,25733	0,000	0,000	11,900	4,400
14	0,97087	-0,33977	0,000	0,000	6,200	2,900
15	0,97894	-0,37935	0,000	35,000	63,600	22,800
16	0,98015	-0,36109	0,000	0,000	4,100	0,800
17	0,98074	-0,34920	0,000	0,000	23,600	4,300
18	0,98273	-0,38955	0,000	0,000	16,400	7,000
19	0,98584	-0,39194	0,000	0,000	73,300	28,900
20	0,99000	-0,38978	75,000	59,536	0,000	0,000
21	0,99000	-0,39163	28,000	63,524	0,000	0,000
22	0,98203	-0,39743	0,000	25,000	45,500	13,600
23	0,98183	-0,39377	0,000	5,000	50,300	13,500
24	0,97850	-0,39055	0,000	0,000	35,400	7,900
25	0,97631	-0,39473	0,000	0,000	14,200	3,300

Tabel 4.12.
 Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi

No	Hubungan Bus	P(MW)	Q(MVAR)	Arus re(A)	Arus im(A)
1	1 – 2	191,767	34,519	1217,56	219,166
2	1 – 9	289,211	-27,246	1836,261	-172,991
3	2 – 3	18,035	5,997	114,989	42,088
4	2 – 4	157,904	17,762	1013,441	144,776
5	4 – 5	119,938	-3,045	784,497	34,063
6	5 – 6	44,797	-8,882	302,213	-22,056
7	5 – 7	34,269	-13,743	236,835	-62,628
8	6 – 7	21,958	-15,597	159,987	-80,904
9	7 – 8	55,865	6,962	358,102	105,493
10	7 – 10	-112,139	-26,899	-704,966	-296,329
11	7 – 11	49,805	0,446	325,441	56,352
12	8 – 11	13,783	-9,888	106,838	-44,929
13	9 – 10	132,279	38,843	814,206	322,976
14	9 – 11	130,264	-92,896	878,374	-511,667
15	11 – 12	133,074	-113,610	1062,099	-542,496
16	12 – 13	184,932	-78,090	1348,511	-215,369
17	12 – 17	73,230	-23,344	521,678	-35,222
18	13 – 14	85,164	-40,213	642,406	-119,434
19	13 – 15	83,023	-41,361	630,074	-130,920
20	14 – 15	76,430	-45,085	597,995	-116,980
21	15 – 16	-20,471	6,913	-146,934	-7,893
22	15 – 17	-22,004	6,820	-156,387	-12,350
23	15 – 18	25,124	-21,774	213,844	-74,378
24	15 – 19	28,914	-33,021	266,185	-135,967
25	15 – 22	22,244	-18,586	187,587	-61,473
26	15 – 24	57,182	-17,560	406,016	33,123
27	16 – 17	-24,762	10,355	-182,441	6,385
28	18 – 19	8,608	-26,955	123,465	-146,981
29	19 – 20	-74,813	-59,124	-314,827	-562,756
30	19 – 21	-27,902	-63,249	-10,996	-467,354
31	19 – 22	23,566	2,405	141,069	75,903
32	19 – 23	43,131	33,491	183,042	320,719
33	23 – 24	-7,283	25,259	-111,471	139,411
34	24 – 25	14,222	1,263	86,327	44,845

Tabel 4.13.

Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi

Pembangkitan		Pembebanan		Rugi-rugi		Waktu Komputasi
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	
754,278	249,228	700,000	260,200	54,278	10,972	0:1:20:890

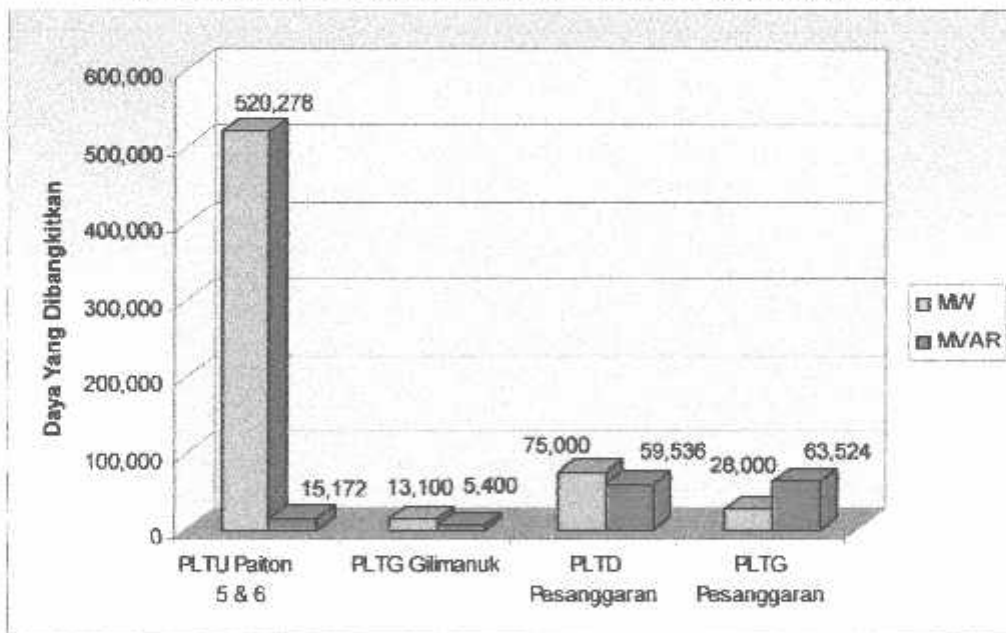
Tabel 4.14.

Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Setelah Optimasi

No	Nama Pembangkit	Daya Yang Dibangkitkan		Biaya Operasi (Rupiah/jam)
		(MW)	(MVar)	
1	PLTU Paiton 5 & 6	520,278	15,172	Rp 2.740.563,25
2	PLTG Gilimanuk	131,000	5,400	Rp 321.047,81
3	PLTD Pesanggaran	75,000	59,536	Rp 294.609,25
4	PLTG Pesanggaran	28,000	63,524	Rp 377.173,60
Total				Rp 3,733,393.91

Grafik 4.2.

Hasil Perhitungan Daya Yang Dibangkitkan Setelah Optimasi



4.9. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Daya Reaktif Menggunakan Metode Kombinasi *Evolutionary Programming-Evolutionary Strategy - Genetic Algorithm*

Dari semua hasil perhitungan diatas maka bisa dibuat tabel-tabel perbandingan sesuai dengan tujuan yang ingin dicapai.

4.9.1. Tingkat Optimum Biaya Pembangkitan Sebelum dan Sesudah Optimasi

Berikut ini adalah tabel yang berisi perbandingan biaya optimum pembangkitan antara hasil perhitungan sebelum dengan sesudah optimasi.

Tabel 4.15.
Perbandingan Tingkat Optimum Biaya Pembangkitan

No	Nama Pembangkit	Sebelum Optimasi	Setelah Optimasi
1	PLTU Paiton 5& 6	Rp 3.141.662,80	Rp 2.740.563,25
2	PLTG Gilimanuk	Rp 261.215,98	Rp 321.047,81
3	PLTD Pesanggaran	Rp 157.553,76	Rp 294.609,25
4	PLTG Pesanggaran	Rp 1.011.934,23	Rp 377.173,60
	Total	Rp 4.572.366,77	Rp 3.733.393,91

Dari tabel tingkat optimum diatas dapat dilihat bahwa besarnya biaya produksi pembangkitan sebelum optimasi adalah Rp 4.572.366,77 sedangkan besarnya biaya produksi pembangkitan setelah optimasi adalah Rp 3.733.393,91 Jadi selisih antara biaya produksi pembangkitan sebelum dengan sesudah optimasi adalah Rp 838.972,86 atau terjadi optimasi biaya sebesar 18.35%

4.9.2. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

Sebelum dan sesudah optimasi

Berikut ini adalah tabel perbandingan tingkat tegangan dan sudut tegangan pada tiap bus.

Tabel 4.16
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

No. Bus	Sebelum optimasi		Sesudah Soptimasi	
	Tegangan (pu)	Sudut Tegangan (deg)	Tegangan (pu)	Sudut Tegangan (deg)
1	1,03000	0,00000	1,05000	0,00000
2	1,01639	-0,03425	1,03478	-0,02988
3	1,01216	-0,03949	1,03064	-0,03494
4	1,00324	-0,07946	1,01860	-0,06878
5	0,99580	-0,14363	1,00478	-0,12289
6	0,99714	-0,17598	1,00155	-0,14944
7	1,00524	-0,19387	1,00534	-0,16249
8	0,98045	-0,27507	0,97570	-0,22425
9	1,04420	-0,10768	1,04928	-0,09202
10	1,02352	-0,14966	1,02659	-0,12671
11	0,98616	-0,29564	0,97809	-0,23442
12	0,99000	-0,30457	0,98000	-0,24118
13	0,96892	-0,31408	0,96090	-0,25733
14	0,95630	-0,39052	0,97087	-0,33977
15	0,95226	-0,42835	0,97894	-0,37935
16	0,96044	-0,41346	0,98015	-0,36109
17	0,96476	-0,40340	0,98074	-0,34920
18	0,94867	-0,43636	0,98273	-0,38955
19	0,94868	-0,43771	0,98584	-0,39194
20	0,95000	-0,43645	0,99000	-0,38978
21	0,9500	-0,43350	0,99000	-0,39163
22	0,94794	-0,44481	0,98203	-0,39743
23	0,94767	-0,44121	0,98183	-0,39377
24	0,94779	-0,43924	0,97850	-0,39055
25	0,94551	-0,44370	0,97631	-0,39473

Dari tabel di atas dapat diketahui bahwa tegangan tiap bus dari sebelum sampai sesudah optimasi mengalami kenaikan, maka sudut tegangannya pada tiap bus mengalami penurunan. salah satu contohnya pada bus 25 tegangan sebelum optimasi adalah 0,94551 pu setelah optimasi tegannya naik menjadi 0,97631 pu, sedangkan sudut tegannya turun dari -0,44370 deg menjadi -0,39473 deg. sehingga dari table tersebut dapat kita simpulkan bahwa profil tegangan dari sebelum sampai sesudah optimasi semakin baik.

4.9.3. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran Sebelum dan Sesudah Optimasi

Berikut ini adalah tabel perbandingan tingkat rugi-rugi daya pada saluran sebelum dan sesudah optimasi.

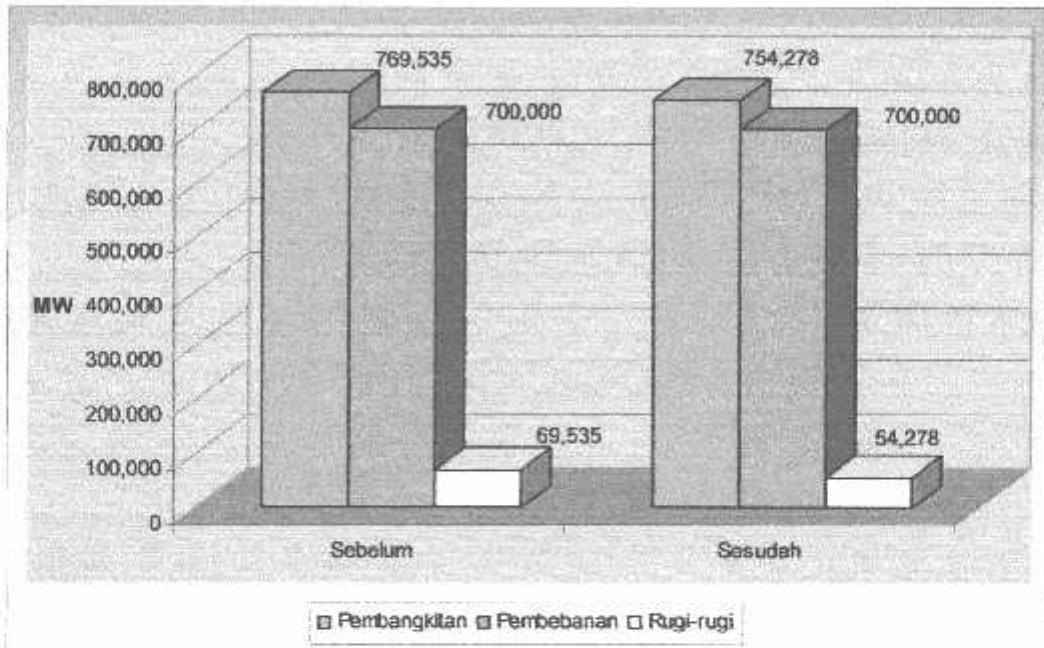
Tabel 4.17.
perbandingan tingkat rugi-rugi daya pada saluran sebelum dan sesudah optimasi

Sebelum Optimasi		Sesudah Optimasi	
Rugi - Rugi		Rugi - Rugi	
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
69,535	16,689	54,278	10,972

Untuk rugi-rugi daya pada saluran sebelum dioptimasi adalah 69,535 MW dan 16,689 MVAR, sedangkan perhitungan rugi-rugi daya setelah optimasi adalah 54,278 MW dan -10,972 MVAR. Dari hasil perhitungan dapat dilihat

bahwa rugi-rugi daya yang ada sebelum dan sesudah optimasi terjadi penurunan yaitu sebesar 15.257 MW dan 27,661 MVAR.

Grafik 4.3.
Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Total
Sebelum dan Setelah Optimasi



Dari grafik 4.3. dapat dilihat bahwa perbandingan total pembangkitan ,pembebanan dan rugi total sebelum dan sesudah optimasi adalah: pada beban tetap pembangkitan total dari sebelum optimasi sampai sesudah optimasi mengalami penurunan dari 769,535 MVA menjadi 754,278 MVA atau mengalami optimasi sebesar 1,98%,dan rugi-rugi totalnya turun(bekurang) dari 69,536 MVA menjadi 54,278 MVA atau mengalami optimasi sebesar 21,94%.

BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah dilakukan analisa perhitungan optimasi daya reaktif dengan menggunakan metode kombinasi *Evolutionary Programming – Evolutionary Strategy - Genetic Algorithm* maka dapat diambil kesimpulan sebagai berikut :

1. Penyelesaian permasalahan optimasi daya reaktif menggunakan metode kombinasi *Evolutionary Programming– Evolutionary Strategy – Genetic Algorithm* pada Sub-Sistem 150 kV Paiton-Bali menghasilkan biaya operasi pembangkitan yang lebih optimum. Biaya produksi pembangkitan sebelum optimasi adalah Rp 4.572.366,77 per jam sedangkan besarnya biaya produksi pembangkitan setelah optimasi adalah Rp. 3.733.393,91 per jam. Jadi selisih antara biaya produksi pembangkitan sebelum dengan sesudah optimasi adalah Rp. 838.972,86 per jam atau terjadi optimasi biaya sebesar 18,35 %.
2. Dengan mengoptimasi daya reaktif menggunakan metode kombinasi *Evolutionary Programming - Evolutionary Strategy – Genetic Algorithm* ternyata rugi-rugi daya sistem turun dari $69,535 + j 16,689$ MVA menjadi $54,278 - j 10,972$ MVA.
3. Untuk tingkat tegangan didapatkan hasil bahwa profil tegangan setelah optimasi daya reaktif menggunakan metode kombinasi *Evolutionary Programming – Evolutionary Strategy – Genetic Algorithm* masih dalam

batas-batas yang diijinkan yaitu antara -5 % sampai +5 % dari tegangan nominal.

5.2. Saran

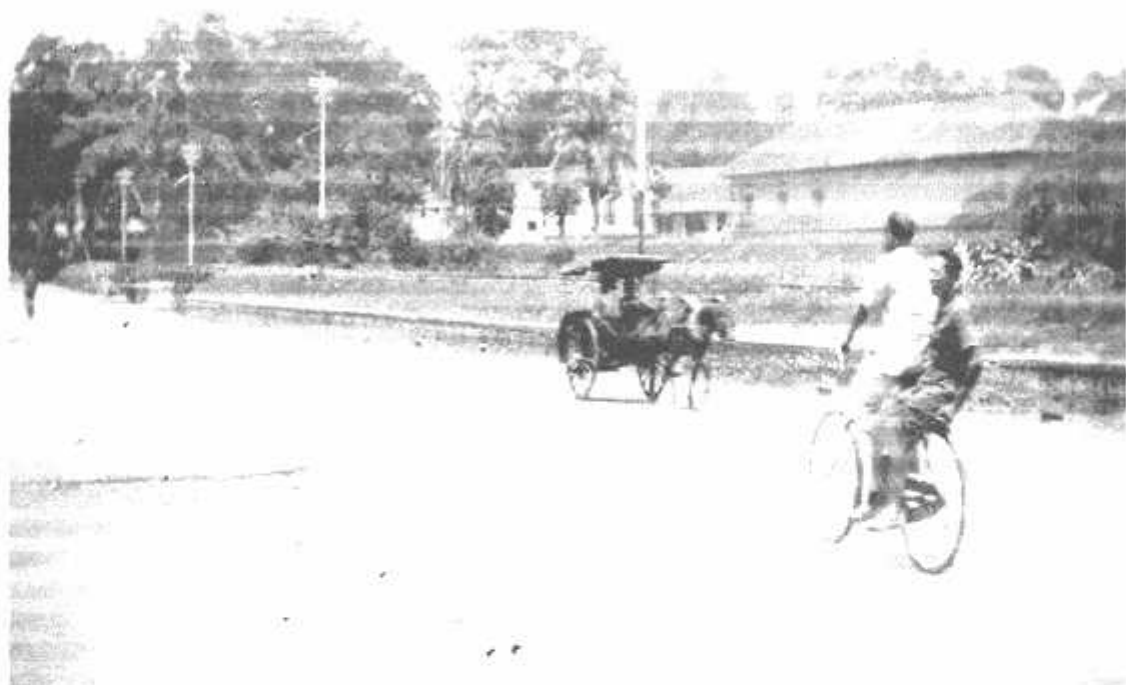
Penggunaan metode kombinasi *Evolutionary Programming – Evolutionary Strategy – Genetic Algorithm* untuk menyelesaikan permasalahan optimasi daya reaktif masih perlu dikembangkan lagi terhadap sistem jaringan yang lebih luas dan penggunaan waktu komputasi yang lebih singkat sehingga nantinya dapat digunakan secara lebih efektif dan efisien dalam pemecahan permasalahan optimasi biaya pembangkitan.

DAFTAR PUSTAKA

- [1]. Kwan Y. Lee, and Frank F. Yang, "*Optimal Reactive Power Planning Using Evolutionary Algorithm : A Comparative Study For Evolutionary Programming, Evolutionary Strategy, Genetic Algorithm, and Linier Programming*", IEEE Trans On Power System, Vol. 13, no 1, February 1998.
 - [2]. Lamont. Jhon W and fu Jian "*Cost Analisis Of Reaktif Power Support.*" IEEE Trans On Power System, Vol. 14, no 3, Agustus 1999.
 - [3]. Stevenson, William D, Jr, "*Analisa Sistem Tenaga Listrik*", Penerbit Erlangga Edisi keempat, 1996.
 - [4]. Wood, Allan J and B. F. Wollenberg, "*Power Generation, Operation, and Control*", John Willey & Sons, Inc, 1996.
 - [5]. Data Saluran Transmisi 150 kV Paiton Bali.
 - [6]. Ir. Yusuf ismail Nahkoda, MT "*Komputasi Cerdas Algoritma Genetika*"
 - [7]. D. E. Goldberg, "*Genetic Algorithm In Search, Optimization And Machine Learning*", Addison-Wesley, 1989.
 - [8]. Syafena, Jatri, Skripsi, "*Ekonomi Dispackt Dengan Optimasi Saya Aktif Dan Reaktif Pada Pembangkit Thermal di Area IV*", 2002.
 - [9]. Spears, William M., "*An Over view of Evolutionary Computation*". European Conference On Machine Learning, 1993.
-

- [10]. Hussain Talik S."An Intruduction in to Evolutionary Computation".Depatment of Computing and Impormation Sience Queen's Univesity, Kingston.
 - [11]. Kusuna Dewi, Sri," *Artifitial Intelligence (Teknik dan Aplikasinya*",Graha Ilmu, Yogyakarta, 2003.
 - [12]. Son Kuswadi,"*Pengendali Cerdas*". Inst. Teknologi Sepuluh November, EE Press, Surabaya, Juli 2000.
 - [13]. Data Load Flow Region 4. Sub Sistem Paiton Bali Selasa 10 Januari 2006 Pukul 19.00. PT PLN (Persero) P3B Region Jawa Bali.
 - [14]. Data "IEE 30 Bus Sistem Tes 150 KV 20 Agustus 1993, UW Archive,Skripsi Slamet Novianto.
 - [15]. **Referensi Book** , Jang Et Al "*Neuro Neuro-Fuzzy and Soft Computing Fuzzy and Soft Compting*"Prentice Hall 1997.
-

LAMPIRAN



Has. Ayo meca cianraui.
(Tj. k. k. c.)
in Melano

Feb. 1948.

DATA KARAKTERISTIK TRANSMISI UPT & SRB

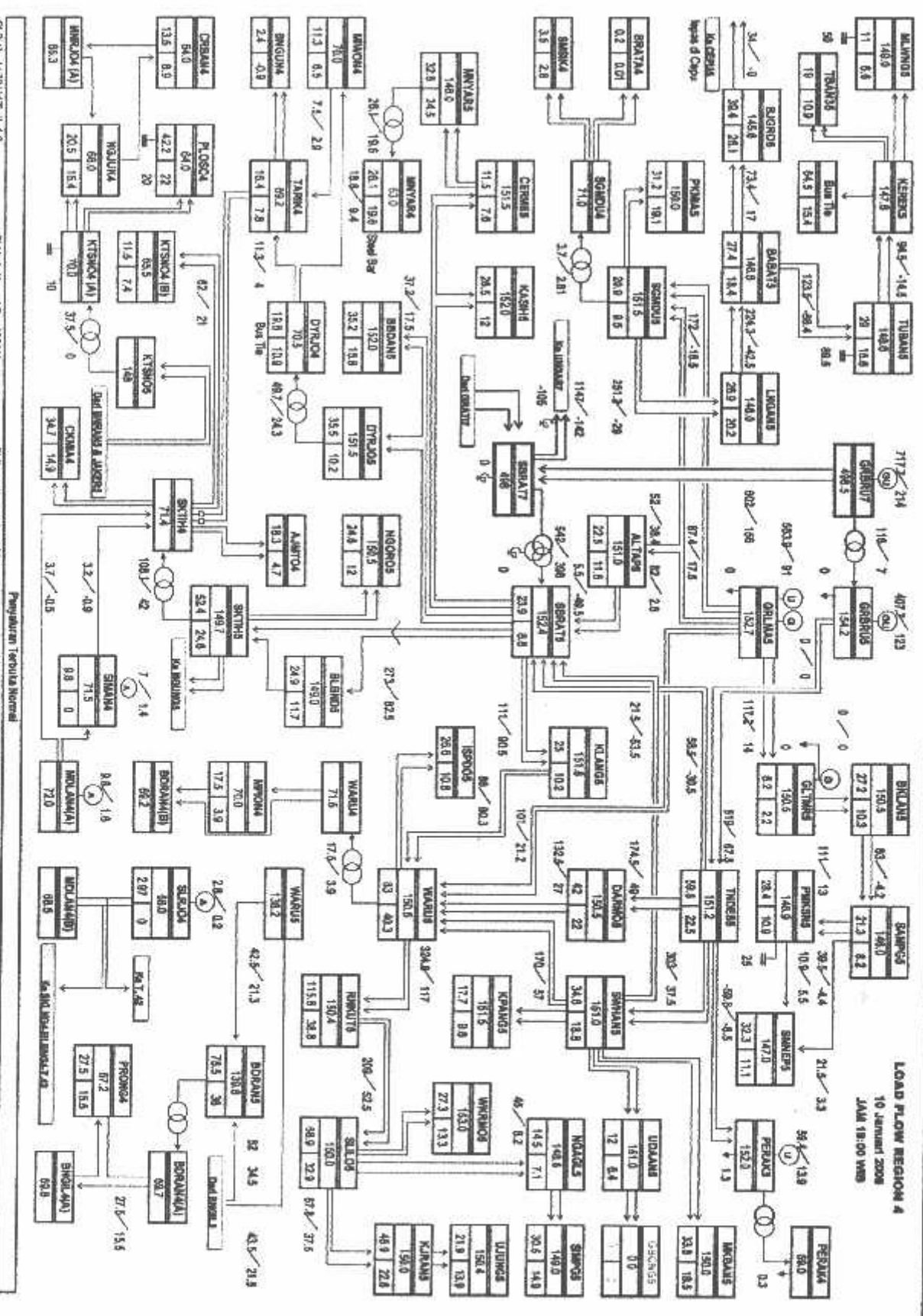
Halaman : 1

Jml Sholat	Darl	Ke	No. Silet	Teg (V)	Jarak (km)	Data Pemfungsian			T (Amp)	Tap CT Dar	T Ke	Tap WT Ke	R (Ohm)	X (Ohm)	B (MVA)	R (Ohm)	X (Ohm)	B (Ohm)	Z ar (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Daya (MVA)
						Jenis	Isi	Ar																	
(1)	(2)												(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)	(13)	(14)	(15)

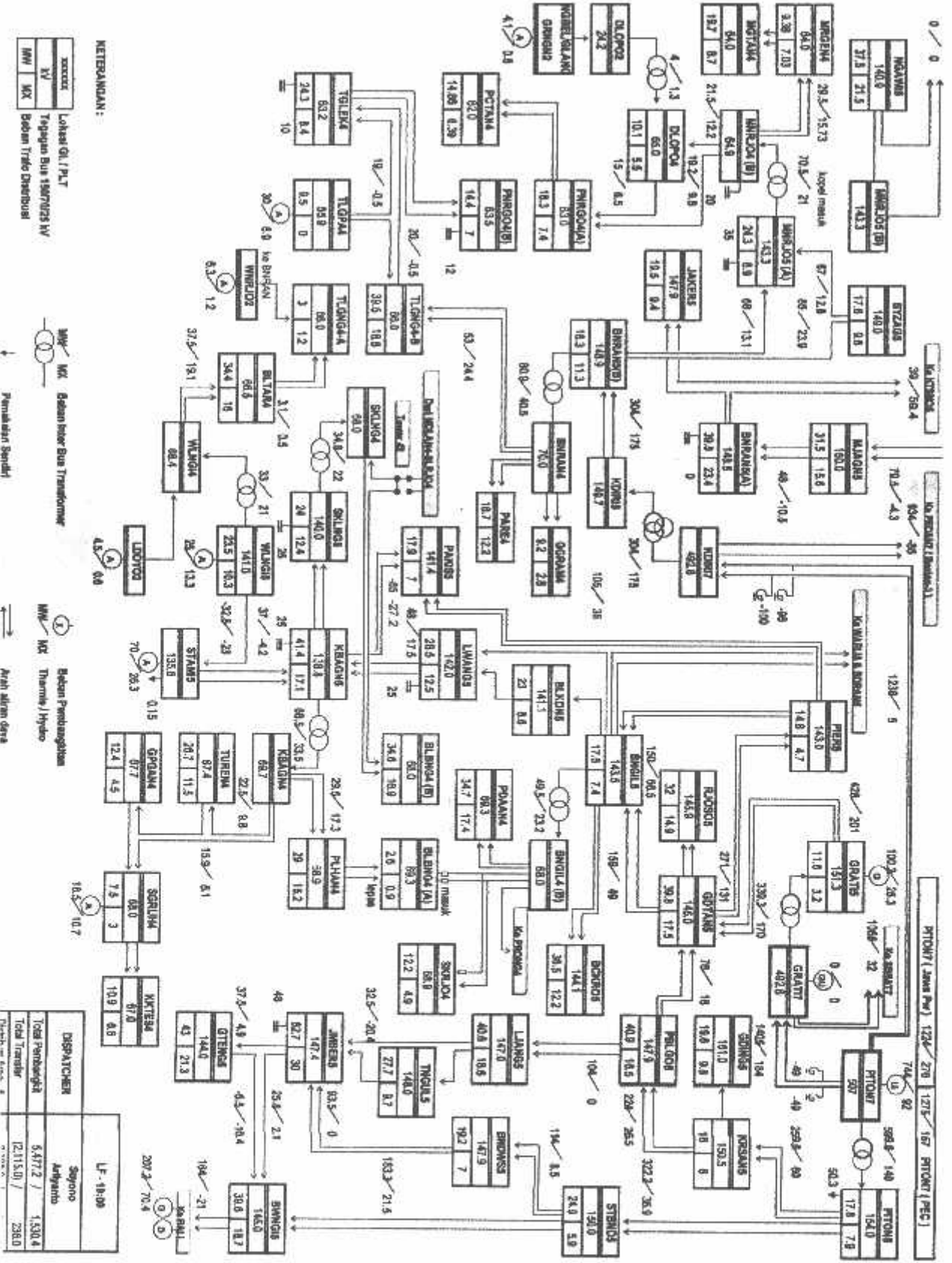
6. UPT Probolinggo

1	PITONT	GRATIT	2	500	83,563	GANNET	41,392.8	2600	2000	2000	0.6251	0.2808	4,0450	2,2179	24,0123	357,7818	24,9113	2,500,000	0.0100	0.0009	0.0099	0.0072	2,424,8000
1	BNGIL5	GOTANS	1	150	16,805	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	6,7270	47,9447	7,0094	225,0000	0.0312	0.0098	0.0299	0.0108	192,2500
1	BNGIL5	GOTANS	2	150	18,805	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	6,7270	47,9447	7,0094	225,0000	0.0312	0.0098	0.0299	0.0108	192,2500
1	BNGIL5	BDRAMS	1	150	9,800	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	3,9229	27,9494	4,0876	225,0000	0.0312	0.0051	0.0174	0.0051	192,2500
1	BNGIL5	WARUS	1	150	9,800	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	3,9229	27,9494	4,0876	225,0000	0.0312	0.0051	0.0174	0.0051	192,2500
1	BNGIL5	LWAKOS	1	150	24,680	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	13,8284	98,9420	14,4852	225,0000	0.0643	0.0181	0.0617	0.0181	192,2500
1	BNGIL5	BLKONS	1	150	27,770	ACSR AW	330 mm ²	740	1000	1000	0.1112	0.4003	2,3530	1,9665	11,1152	79,2278	11,5830	225,0000	0.0315	0.0145	0.0494	0.0145	192,2500
1	BNGIL5	PIERS	1	150	5,150	ACSR ZEBRA	2 x 435 mm ²	1520	2000	1000	0.2807	0.2807	4,0303	1,9931	1,4436	20,7045	2,4621	225,0000	0.0106	0.0096	0.0096	0.0096	420,8760
1	BNGIL5	PIERS	2	150	5,150	ACSR ZEBRA	2 x 435 mm ²	1520	2000	1000	0.2807	0.2807	4,0303	1,9931	1,4436	20,7045	2,4621	225,0000	0.0106	0.0096	0.0096	0.0096	420,8760
1	BNGIL5	BCKROS	1	150	6,200	ACSR ZEBRA	2 x 435 mm ²	1520	1000	1000	0.2807	0.2807	4,0303	2,3094	1,7403	24,9259	2,5641	225,0000	0.0132	0.0101	0.0377	0.0101	420,8760
1	BNGIL5	CKKROS	2	150	6,200	ACSR ZEBRA	2 x 435 mm ²	1520	1000	1000	0.2807	0.2807	4,0303	2,3094	1,7403	24,9259	2,5641	225,0000	0.0132	0.0101	0.0377	0.0101	420,8760
1	BNGIL5	ROKOS	1	150	10,487	ACSR AW	330 mm ²	740	1000	800	0.1112	0.4003	2,3530	1,2291	4,1979	29,9194	4,3742	225,0000	0.0194	0.0055	0.0187	0.0055	192,2500
1	GOTANS	R.OSOS	2	150	10,487	ACSR AW	330 mm ²	740	1000	800	0.1112	0.4003	2,3530	1,2291	4,1979	29,9194	4,3742	225,0000	0.0194	0.0055	0.0187	0.0055	192,2500
1	GOTANS	R.OSOS	2	150	10,487	ACSR AW	330 mm ²	740	1000	800	0.1112	0.4003	2,3530	1,2291	4,1979	29,9194	4,3742	225,0000	0.0194	0.0055	0.0187	0.0055	192,2500
1	PIERS	GRATIS	1	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214	0.0051	192,2500
1	PIERS	GRATIS	2	150	20,000	ACSR	410 mm ²	1420	1000	1000	0.2807	0.2807	4,0303	11,0100	8,4210	120,6090	14,3424	225,0000	0.0637	0.0051	0.0214		

LOAD FLOW REGION 4
10 Januari 2008
JAM 18:00 WIB



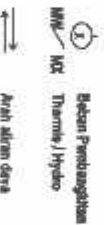
G Subst pnt 70 kV Tank-1,2
 G Rbtg snt 70 kV Pfan-1,2
 G Subst pnt 70 kV Btng Blng 2
 G Himp Kopel Bus 150 kV
 G Himp Kopel Bus 70 kV
 G Wpnt Kopel Bus 70 kV
 G Wpnt Kopel Bus 150 kV
 G Himp Kopel Bus 70 kV
 G Himp Kopel Bus 150 kV
 Pnyuluhan Terutama Normal
 Fnt 150 kV Pr-8grn-1,2 & Cops
 CATATAN:
 Rbtg Penanngnt = rbtg
 Pnt Penanngnt = rbtg



0 / 0

LEGENDA:

XXXXXX	Lokeas GI / PLT	
XY	Topogan Bus 150/70/35 KV	
HW	KX	Deban Tiro Drebusi



DISPATCHER	Seyono
	Arjanto
Total Pembangkit	5.477.2 / 1.530.4
Total Transfer	12.115.0 / 286.0

LF - 18:00


```

unit uEP;

interface

uses uUtils,uGenetic,uLoadflow,uObjFunc,Math,uHasil,uRandom;

type
  TEP=class
  private
    FMaxGen,FPopSize,FLength:integer;
    FBetha,FMin1,FAvg1,FMax1,FSumFitness:double;
    FBatasChrom:TBatasArr1;
    FParent,FChild:TPopDouble1;
    FBestIndi:TIndiDouble1;
    function GetIndividu(const rIndi:TIndiDouble1):TIndiDouble1;
    procedure InitParent;
    procedure Statistik;
    function Seleksi:integer;
    function Mutasi(const rChrom:dArr1;
      const rf1,rfmax:double):dArr1;
    procedure Generasi;
    function FindBestIndi:TIndiDouble1;
    function FindMaxIndi:double;
    procedure SwapIndi(var rIndi1,rIndi2:TIndiDouble1);
    procedure Kompetisi;
    procedure doHitung;
    function getBestChrom:dArr1;
    function getMin:dArr1;
    function getAvg:dArr1;
    function getMax:dArr1;
  protected
    FRandom:TRandomu;
    FMin,FAvg,FMax:dArr1;
  public
    constructor Create(const rMaxGen,rPopSize,rLength:integer;
      const rBetha:double;
      const rBatasChrom:TBatasArr1);
    destructor Destroy;override;
    property BestChrom:dArr1 read getBestChrom;
    property Min:dArr1 read getMin;
    property Avg:dArr1 read getAvg;
    property Max:dArr1 read getMax;
  end;

implementation

```

```

//constructor
constructor TEP.Create(const rMaxGen,rPopSize,rLength:integer;
    const rBeta:double;
    const rBatasChrom:T BatasArr1);
var i:integer;
begin
    inherited Create;
    FMaxGen:=rMaxGen;
    FPopSize:=rPopSize;
    FLength:=rLength;
    FBeta:=rBeta;
    SetLength(FBatasChrom,FLength);
    for i:=0 to FLength-1 do
    begin
        FBatasChrom[i].min:=rBatasChrom[i].min;
        FBatasChrom[i].max:=rBatasChrom[i].max;
    end;
    FRandom:=TRandomu.Create;
end;

//destructor
destructor TEP.Destroy;
begin
    try
        FRandom.Free;
    finally
        inherited Destroy;
    end;
end;

//data processing
function TEP.GetIndividu(const rIndi:TIndiDouble1):TIndiDouble1;
var i:integer;
begin
    SetLength(result.chrom,FLength);
    for i:=0 to FLength-1 do
    begin
        result.chrom[i]:=rIndi.chrom[i];
    end;
    result.fitness:=rIndi.fitness;
end;

procedure TEP.InitParent;
var i,j:integer;
begin
    SetLength(FParent,FPopSize);

```

```

SetLength(FChild,FPopSize);
SetLength(FMin,FMaxGen);
SetLength(FAvg,FMaxGen);
SetLength(FMax,FMaxGen);
for i:=0 to FPopSize-1 do
begin
  SetLength(FParent[i].chrom,FLength);
  for j:=0 to FLength-1 do
  begin
    FParent[i].chrom[j]:=FRandom.NextDouble(FBatasChrom[j].min,
      FBatasChrom[j].max);
  end;
  FParent[i].fitness:=gObjFunc.doHitung(FParent[i].chrom);
end;
end;

```

```

procedure TEP.Statistik;
var i:integer;
begin
  FMin1:=FParent[0].fitness;
  FMax1:=FParent[0].fitness;
  FSumFitness:=FParent[0].fitness;
  for i:=1 to FPopSize-1 do
  begin
    if FMin1>FParent[i].fitness then
    begin
      FMin1:=FParent[i].fitness;
    end;
    if FMax1<FParent[i].fitness then
    begin
      FMax1:=FParent[i].fitness;
    end;
    FSumFitness:=FSumFitness+FParent[i].fitness;
  end;
  FAvg1:=FSumFitness/FPopSize;
end;

```

```

function TEP.Seleksi:integer;
var i:integer;
    partsum,rand:double;
begin
  i:=0;
  rand:=FRandom.NextDouble(0,FSumFitness);
  partsum:=0;
  repeat
    i:=i+1;

```

```

    partsum:=partsum +FParent[i-1].fitness;
until (partsum>=rand) or (i=FPopSize);
result:=i-1;
end;

function TEP.Mutasi(const rChrom:dArr1;
    const rf1,rfmax:double):dArr1;
var i:integer;
    tho:double;
begin
    SetLength(result,FLength);
    for i:=0 to FLength-1 do
    begin
        tho:=FBeta*(FBatasChrom[i].max-FBatasChrom[i].min)*rf1/rfmax;
        result[i]:=rChrom[i]+FRandom.NextGaussian(0,tho);
        if result[i]>FBatasChrom[i].max then
        begin
            result[i]:=FBatasChrom[i].max;
        end;
        if result[i]<FBatasChrom[i].min then
        begin
            result[i]:=FBatasChrom[i].min;
        end;
    end;
end;

procedure TEP.Generasi;
var i,mate:integer;
    fmax:double;
begin
    fmax:=FindMaxIndi;
    for i:=0 to FPopSize-1 do
    begin
        mate:=Seleksi;
        FChild[i].chrom:=Mutasi(FParent[mate].chrom,FParent[mate].fitness,fmax);
        FChild[i].fitness:=gObjFunc.doHitung(FChild[i].chrom);
    end;
end;

function TEP.FindBestIndi:TIndiDouble1;
var i,no:integer;
    min:double;
begin
    min:=FParent[0].fitness;
    no:=0;
    for i:=1 to FPopSize-1 do

```

```

begin
  if min>FParent[i].fitness then
    begin
      min:=FParent[i].fitness;
      no:=i;
    end;
  end;
  result:=getIndividu(FParent[no]);
end;

function TEP.FindMaxIndi:double;
var i:integer;
begin
  result:=FParent[0].fitness;
  for i:=1 to FPopSize-1 do
    begin
      if result<FParent[i].fitness then
        begin
          result:=FParent[i].fitness;
        end;
    end;
  end;
end;

procedure TEP.SwapIndi(var rIndi1,rIndi2:TIndiDouble1);
var tmp:TIndiDouble1;
begin
  tmp:=getIndividu(rIndi1);
  rIndi1:=getIndividu(rIndi2);
  rIndi2:=getIndividu(tmp);
end;

procedure TEP.Kompetisi;
var i,j,pos:integer;
    rand,cek:double;
    w:dArr1;
    tmp:TPopDouble1;
begin
  SetLength(w,FPopSize*2);
  SetLength(tmp,FPopSize*2);
  for i:=0 to FPopSize-1 do
    begin
      tmp[i]:=getIndividu(FParent[i]);
      tmp[FPopSize+i]:=getIndividu(FChild[i]);
    end;
  for i:=0 to 2*FPopSize-1 do
    begin

```

```

w[i]:=0;
for j:=0 to 2*FPopSize-1 do
begin
  repeat
    pos:=FRandom.NextInt(0,2*FPopSize-1);
    until pos<>i;
    rand:=FRandom.NextDouble;
    cek:=(tmp[pos].fitness/(tmp[pos].fitness+tmp[i].fitness));
    if rand<cek then
    begin
      w[i]:=w[i]+1;
    end;
  end;
end;
for j:=0 to 2*FPopSize-2 do
begin
  for j:=i to 2*FPopSize-1 do
  begin
    if w[i]<w[j] then
    begin
      Swap(w[i],w[j]);
      SwapIndi(tmp[i],tmp[j]);
    end;
  end;
end;
for i:=0 to FPopSize-1 do
begin
  FParent[i]:=getIndividu(tmp[i]);
end;
end;

```

```

procedure TEP.doHitung;
var i:integer;
    tmpIndi:TIndiDouble1;
begin
  InitParent;
  Statistik;
  FBestIndi:=FindBestIndi;
  for i:=0 to FMaxGen-1 do
  begin
    Generasi;
    Kompetisi;
    tmpIndi:=FindBestIndi;
    if FBestIndi.fitness>tmpIndi.fitness then
    begin
      FBestIndi:=getIndividu(tmpIndi);
    end;
  end;
end;

```

```

    end;
    Statistik;
    FMin[i]:=FMin1;
    FAvg[i]:=FAvg1;
    FMax[i]:=FMax1;
    frmHasil.pbGen.StepBy(1);
end;
end;

//data outputs

function TEP.getBestChrom:dArr1;
var i:integer;
begin
    doHitung;
    SetLength(result,FLength);
    for i:=0 to FLength-1 do
        begin
            result[i]:=FBestIndi.chrom[i];
        end;
    end;
end;

function TEP.getMin:dArr1;
var i:integer;
begin
    SetLength(result,FMaxGen);
    for i:=0 to FMaxGen-1 do
        begin
            result[i]:=FMin[i];
        end;
    end;
end;

function TEP.getAvg:dArr1;
var i:integer;
begin
    SetLength(result,FMaxGen);
    for i:=0 to FMaxGen-1 do
        begin
            result[i]:=FAvg[i];
        end;
    end;
end;

function TEP.getMax:dArr1;
var i:integer;
begin
    SetLength(result,FMaxGen);

```

```
for i:=0 to FMaxGen-1 do
begin
  result[i]:=FMax[i];
end;
end;

end.
```

```
unit uES;
```

```
interface
```

```
uses uUtils,uGenetic,uLoadflow,uObjFunc,Math,uHasil,uRandom;
```

```
type
```

```
TES=class
```

```
private
```

```
  FMaxGen,FPopSize,FLength:integer;
```

```
  FBeta,FMin1,FAvg1,FMax1,FSumFitness,FDev:double;
```

```
  FBatasChrom:TBatasArr1;
```

```
  FParent,FChild:TPopDouble1;
```

```
  FBestIndi:TIndiDouble1;
```

```
  FBestChrom:dArr1;
```

```
  function GetIndividu(const rIndi:TIndiDouble1):TIndiDouble1;
```

```
  procedure InitParent;
```

```
  procedure Statistik;
```

```
  function Seleksi:integer;
```

```
  function Mutasi(const rChrom:dArr1;
```

```
    const rFl,rFmax:double):dArr1;
```

```
  procedure Generasi;
```

```
  function FindBestIndi:TIndiDouble1;
```

```
  function FindMaxIndi:double;
```

```
  procedure SwapIndi(var rIndi1,rIndi2:TIndiDouble1);
```

```
  procedure Kompetisi;
```

```
  procedure doHitung;
```

```
  function getBestChrom:dArr1;
```

```
  function getMin:dArr1;
```

```
  function getAvg:dArr1;
```

```
  function getMax:dArr1;
```

```
protected
```

```
  FRandom:TRandomu;
```

```

    FMin,FAvg,FMax:dArr1;
public
    constructor Create(const rMaxGen,rPopSize,rLength:integer;
        const rBeta,rDev:double;
        const rBatasChrom:TBatasArr1;
        const rBestChrom:dArr1);
    destructor Destroy;override;
    property BestChrom:dArr1 read getBestChrom;
    property Min:dArr1 read getMin;
    property Avg:dArr1 read getAvg;
    property Max:dArr1 read getMax;
end;

implementation

//constructor
constructor TES.Create(const rMaxGen,rPopSize,rLength:integer;
    const rBeta,rDev:double;
    const rBatasChrom:TBatasArr1;
    const rBestChrom:dArr1);
var i:integer;
begin
    inherited Create;
    FMaxGen:=rMaxGen;
    FPopSize:=rPopSize;
    FLength:=rLength;
    FBeta:=rBeta;
    FDev:=rDev;
    SetLength(FBatasChrom,FLength);
    SetLength(FBestChrom,FLength);
    for i:=0 to FLength-1 do
    begin
        FBatasChrom[i].min:=rBatasChrom[i].min;
        FBatasChrom[i].max:=rBatasChrom[i].max;
        FBestChrom[i]:=rBestChrom[i];
    end;
    FRandom:=TRandomu.Create;
end;

//destructor
destructor TES.Destroy;
begin
    try
        FRandom.Free;
    finally
        inherited Destroy;
    end;
end;

```

```

var i:integer;
begin
  FMin1:=FParent[0].fitness;
  FMax1:=FParent[0].fitness;
  FSumFitness:=FParent[0].fitness;
  for i:=1 to FPopSize-1 do
  begin
    if FMin1>FParent[i].fitness then
    begin
      FMin1:=FParent[i].fitness;
    end;
    if FMax1<FParent[i].fitness then
    begin
      FMax1:=FParent[i].fitness;
    end;
    FSumFitness:=FSumFitness+FParent[i].fitness;
  end;
  FAvg1:=FSumFitness/FPopSize;
end;

```

```

function TES.Seleksi:integer;
var i:integer;
    partsum,rand:double;
begin
  i:=0;
  rand:=FRandom.NextDouble(0,FSumFitness);
  partsum:=0;
  repeat
    i:=i+1;
    partsum:=partsum+FParent[i-1].fitness;
  until (partsum>=rand) or (i=FPopSize);
  result:=i-1;
end;

```

```

function TES.Mutasi(const rChrom:dArr1;
    const rF1,rFmax:double):dArr1;
var i:integer;
begin
  SetLength(result,FLength);
  for i:=0 to FLength-1 do
  begin
    result[i]:=rChrom[i]+FRandom.NextGaussian(0,FDev);
    if result[i]>FBatasChrom[i].max then
    begin
      result[i]:=FBatasChrom[i].max;
    end;
  end;
end;

```

```

    if result[i]<FBatasChrom[i].min then
    begin
        result[i]:=FBatasChrom[i].min;
    end;
end;
end;

```

```

procedure TES.Generasi;
var i,mate:integer;
    fmax:double;
begin
    fmax:=FindMaxIndi;
    for i:=0 to FPopSize-1 do
    begin
        mate:=Seleksi;
        FChild[i].chrom:=Mutasi(FParent[mate].chrom,FParent[mate].fitness,fmax);
        FChild[i].fitness:=gObjFunc.doHitung(FChild[i].chrom);
    end;
end;

```

```

function TES.FindBestIndi:TIndiDouble1;
var i,no:integer;
    min:double;
begin
    min:=FParent[0].fitness;
    no:=0;
    for i:=1 to FPopSize-1 do
    begin
        if min>FParent[i].fitness then
        begin
            min:=FParent[i].fitness;
            no:=i;
        end;
    end;
    result:=getIndividu(FParent[no]);
end;

```

```

function TES.FindMaxIndi:double;
var i:integer;
begin
    result:=FParent[0].fitness;
    for i:=1 to FPopSize-1 do
    begin
        if result<FParent[i].fitness then
        begin
            result:=FParent[i].fitness;
        end;
    end;
end;

```

```

    end;
    end;
end;

procedure TES.SwapIndi(var rIndi1,rIndi2:TIndiDouble1);
var tmp:TIndiDouble1;
begin
    tmp:=getIndividu(rIndi1);
    rIndi1:=getIndividu(rIndi2);
    rIndi2:=getIndividu(tmp);
end;

procedure TES.Kompetisi;
var i,j,pos:integer;
    rand,cek:double;
    w:dArr1;
    tmp:TPopDouble1;
begin
    SetLength(w,FPopSize*2);
    SetLength(tmp,FPopSize*2);
    for i:=0 to FPopSize-1 do
    begin
        tmp[i]:=getIndividu(FParent[i]);
        tmp[FPopSize+i]:=getIndividu(FChild[i]);
    end;
    for i:=0 to 2*FPopSize-1 do
    begin
        w[i]:=0;
        for j:=0 to 2*FPopSize-1 do
        begin
            repeat
                pos:=FRandom.NextInt(0,2*FPopSize-1);
            until pos<>i;
            rand:=FRandom.NextDouble;
            cek:=tmp[pos].fitness/(tmp[pos].fitness+tmp[i].fitness);
            if rand<cek then
            begin
                w[i]:=w[i]+1;
            end;
        end;
    end;
    for i:=0 to 2*FPopSize-2 do
    begin
        for j:=i to 2*FPopSize-1 do
        begin
            if w[i]<w[j] then

```

```

begin
  Swap(w[i],w[j]);
  SwapIndi(tmp[i],tmp[j]);
end;
end;
end;
for i:=0 to FPopSize-1 do
begin
  FParent[i]:=getIndividu(tmp[i]);
end;
end;

procedure TES.doHitung;
var i:integer;
    tmpIndi:TIndiDouble1;
begin
  InitParent;
  Statistik;
  FBestIndi:=FindBestIndi;
  for i:=0 to FMaxGen-1 do
  begin
    Generasi;
    Kompetisi;
    tmpIndi:=FindBestIndi;
    if FBestIndi.fitness>tmpIndi.fitness then
    begin
      FBestIndi:=getIndividu(tmpIndi);
    end;
    Statistik;
    FMin[i]:=FMin1;
    FAvg[i]:=FAvg1;
    FMax[i]:=FMax1;
    frmHasil.pbGen.StepBy(1);
  end;
end;

//data outputs

function TES.getBestChrom:dArr1;
var i:integer;
begin
  doHitung;
  SetLength(result,FLength);
  for i:=0 to FLength-1 do
  begin
    result[i]:=FBestIndi.chrom[i];
  end;
end;

```

```
end;  
end;
```

```
function TES.getMin:dArr1;  
var i:integer;  
begin  
  SetLength(result,FMaxGen);  
  for i:=0 to FMaxGen-1 do  
    begin  
      result[i]:=FMin[i];  
    end;  
end;
```

```
function TES.getAvg:dArr1;  
var i:integer;  
begin  
  SetLength(result,FMaxGen);  
  for i:=0 to FMaxGen-1 do  
    begin  
      result[i]:=FAvg[i];  
    end;  
end;
```

```
function TES.getMax:dArr1;  
var i:integer;  
begin  
  SetLength(result,FMaxGen);  
  for i:=0 to FMaxGen-1 do  
    begin  
      result[i]:=FMax[i];  
    end;  
end;
```

```
end.
```

LAMPIRAN



8 L. de Geer op het Aruispunt

in MALANG 24-3-48



**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : FUD NASPUDDIN
N.I.M. : 01.12.159
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : OPTIMISASI DAYA REAKTIF MENGGUNAKAN
METODE KOMBINASI EP-ES-GA PADA SALURAN
TRANSMISI 150 kV SUB SISTEM PAITON - BALI

Dipertahankan dihadapan Majelis Penguji Skripsi Jenjang Strata Satu (S-1) pada:

Hari : Kamis
Tanggal : 25 September 2008
Dengan Nilai : 80.1 (A) *80.1*

Panitia Ujian Skripsi



Ketua Majelis Penguji

(Ir. Mochtar Asroni, MSME)
NIP. P. 1018100036

Sekretaris Majelis Penguji

(Ir. F. Yudi Limpraptono, MT)
NIP. P. 1039500274

Anggota Penguji

Penguji Pertama

(Ir. Teguh Herbasuki, MT)
NIP. Y 1038900209

Penguji Kedua

(Bambang Prio H, ST, MT)
NIP.Y. 1028400082



PERSETUJUAN PERBAIKAN SKRIPSI

Dari hasil ujian skripsi Jurusan Teknik Elektro jenjang strata satu (S-1) yang diselenggarakan pada :

Hari : Kamis
Tanggal : 25 September 2008

Telah dilakukan perbaikan skripsi oleh :

1. Nama : FUD NASPUDDIN
2. NIM : 01.12.159
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE KOMBINASI EVOLUTIONARY PROGRAMMING-EVOLUTIONARY SRATEGY-GENETIC ALGORITHM PADA SALURAN TRANSMISI 150 KV SUB-SISTEM PAITON-BALI.

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1	Daftar Pustaka	

Anggota Penguji

Ir. Teguh Herbasuki, MT
NIP. 1038900209

Dosen Pembimbing I

Ir. H. Choirul Saleh, MT
NIP. Y. 1018800190

Dosen Pembimbing II

Ir. Djojo Priatmono, MT
NIP. Y. 1018500107



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK

PERSETUJUAN PERBAIKAN SKRIPSI

Dari hasil ujian skripsi Jurusan Teknik Elektro jenjang strata satu (S-1) yang diselenggarakan pada :

Hari : Kamis
Tanggal : 25 September 2008

Telah dilakukan perbaikan skripsi oleh :

1. Nama : FUD NASPUDDIN
2. NIM : 01.12.159
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE KOMBINASI EVOLUTIONARY PROGRAMMING-EVOLUTIONARY STRATEGY-GENETIC ALGORITHM PADA SALURAN TRANSMISI 150 KV SUB-SISTEM PAITON-BALI.

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Abstrsk disempurnakan dan pustaka di perhatikan	
2	Hal 34 – 50 Masukkan BAB 2	
3	Hal 59 – 64 Masukkan BAB 3	
4	Hal 67 lihat	
5	Hal 81 Gambar 4.22 penjelasannya mana ?	

Anggota Penguji

Bambang Prio Hartono, ST, MT
NIP. Y. 1028400082

Dosen Pembimbing I

Ir. H. Choirul Saleh, M.T
NIP. Y. 1018800190

Dosen Pembimbing II

Ir. Djojo Priatmono, MT
NIP. W. 1018500107



PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : FID MASPUDIN
 NIM : 0112159
 Semester : 10
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-1
 Konsentrasi : Teknik Elektronika / Teknik Energi Listrik
 Alamat : JL SEMANGGI TIMUR 1 MALANG

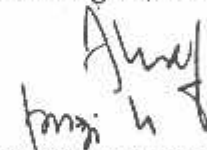
Dengan ini kami mengajukan permohonan untuk mendapatkan persetujuan untuk membuat **SKRIPSI Tingkat Sarjana**. Untuk melengkapi permohonan tersebut, bersama kami lampirkan persyaratan-persyaratan yang harus dipenuhi.

Adapun persyaratan-persyaratan pengambilan **SKRIPSI** adalah sebagai berikut :

1. Telah melaksanakan semua praktikum sesuai dengan konsentrasinya (.....)
2. Telah lulus dan menyerahkan Laporan Praktek Kerja (.....)
3. Telah lulus seluruh mata kuliah keahlian (MKB) sesuai konsentrasinya (.....)
4. Telah menempuh mata kuliah ≥ 134 sks dengan IPK ≥ 2 dan tidak ada nilai E (.....)
5. Telah mengikuti secara aktif kegiatan seminar skripsi yang diadakan Jurusan (.....)
6. Memenuhi persyaratan administrasi (.....)

Demikian permohonan ini untuk mendapatkan penyelesaian lebih lanjut dan atas perhatiannya kami ucapkan terima kasih.

Telah diteliti kebenaran data tersebut diatas
 Recording Teknik Elektro


 (.....)

Malang,200

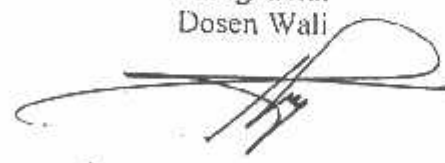
Pemohon


 (FID MASPUDIN.....)

Disetujui
 Ketua Jurusan Teknik Elektro


 Ir. F. Yudi Limpraptono, MT
 NIP. P. 1039500274

Mengetahui
 Dosen Wali


 (Ir. F. YUDI LIMPRAPTONO, MT)

Catatan :

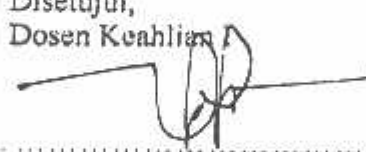
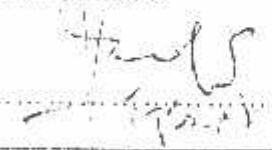
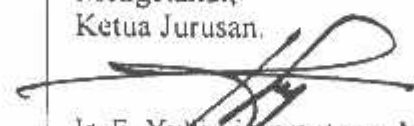

Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1

1. PK 3.88 / 138 2. 81
2.
3. praktikum lengkap



**BERITA ACARA SEMINAR PROPOSAL SKRIPSI
JURUSAN TEKNIK ELEKTRO S-1**

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika

1	Nama Mahasiswa : FUD NASRUDIN	Nim : 0112159		
2	Keterangan	Tanggal	Waktu	Tempat
	Pelaksanaan	25-5-2007	09.00 s/d selesai	Ruang : SEMINAR L-1
3	Spesifikasi judul :			
	a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri	e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya		
4	Judul Proposal yang diseminarkan Mahasiswa	OPTIMISI DAYA REAKTIF PERALA EKONOMIS MENGUNAKAN METODE KOMBINASI EVOLUTIONARY PROGRAMMING - EVOLUTIONARI STRATEGI - GENETIK ALGORITHM PADA SALURAN TRANSMISI 150 KV SUB-SISTEM PATON - BALI		
5	Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian		
6	Catatan : <i>Revisi judul seminar & jurnal yg diambil</i>			
7	Persetujuan Judul Skripsi :			
	Disetujui, Dosen Keahlian I 	Disetujui, Dosen Keahlian II 		
	Mengetahui, Ketua Jurusan  Ir. F. Yudi Limpraptono, MT Nip. Y. 1059500274	Disetujui, Calon Dosen Pembimbing ybs.  Ir. H. CHOIRUL SALIH, MT		

Perhatian :

- *) coret yang tidak perlu
- **) dilingkari a, b, c, atau g. sesuai bidang keahlian.



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

PTN (PERSERO) MALANG
BANK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 4 Juni 2007

Nomor : ITN-297/I.TA/2/2007
Lampiran : -
Perihal : **BIMBINGAN SKRIPSI**
Kepada : Yth. Sdr. Ir. H. CHOIRUL SALEH, MT

Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal Skripsi
untuk Mahasiswa:

Nama : FUD NASPUDDIN
Nim : 0112159
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Kosentrasi : Teknik Energi Listrik S-1

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/I selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

25 Mei 2007 s/d 25 November 2007

Sebagai satu syarat untuk menempuh Program Sarjana Teknik, Jurusan
Teknik Elektro-S1
Demikian atas perhatian serta kerjasamanya yang baik kami sampaikan
terima kasih.



Ketua Jurusan
Teknik Elektro S-1

Ir. F. Yudi Limpraptono, MT
NIP. Y. 1039500274

Tembusan Kepada Yth:

1. Mahasiswa Yang Bersangkutan
2. Arsip

Form S-4a



PERKUMPULAN PENGELOLA PENDIDIKAN UMUM DAN TEKNOLOGI NASIONAL MALANG
INSTITUT TEKNOLOGI NASIONAL MALANG

FAKULTAS TEKNOLOGI INDUSTRI
FAKULTAS TEKNIK SIPIL DAN PERENCANAAN
PROGRAM PASCASARJANA MAGISTER TEKNIK

VI (PERSERO) MALANG
UNIK NIAGA MALANG

Kampus I : Jl. Bendungan Sigura-gura No. 2 Telp. (0341) 551431 (Hunting), Fax. (0341) 553015 Malang 65145
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 4 Juni 2007

Nomor : ITN-298/I.TA/2/2007
Lampiran : -
Perihal : **BIMBINGAN SKRIPSI**

Kepada : Yth. Sdr. Ir. **DJOJO PRIATMONO, MT**

Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal Skripsi
untuk Mahasiswa:

Nama : FUD NASPUDDIN
Nim : 0112159
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Kosentrasi : Teknik Energi Listrik S-1

Maka dengan ini pembimbingan tersebut kami serahkan sepenuhnya
kepada Saudara/I selama masa waktu 6 (enam) bulan, terhitung mulai
tanggal:

25 Mei 2007 s/d 25 November 2007

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Jurusan
Teknik Elektro-S1

Demikian atas perhatian serta kerjasamanya yang baik kami sampaikan
terima kasih.



Ketua Jurusan
Teknik Elektro S-1

Ir. F. Yudi Limpraptono, MT
NIP. Y. 4039500274

Tembusan Kepada Yth:

1. Mahasiswa Yang Bersangkutan
2. Arsip

Form S-4a



FORMULIR BIMBINGAN SKRIPSI

Nama : FUD NASPUDDIN
Nim : 01.12.159
Masa Bimbingan : 25 November 2007 s/d 25 Mei 2008
Judul Skripsi : OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE KOMBINASI EVOLUTIONARY PROGRAMMING-EVOLUTIONARY STRATEGY-GENETIC ALGORITHM PADA SALURAN TRANSMISI 150 kV SUB-SYSTEM PAITON-BALI

No.	Tanggal	Uraian	Parap Pembimbing
	27/11/07	Prab I. Perbaiki Laporan Belahur	es.
		Pendahuluan. Tempoy. dan lain-lain. Jari-jari	
	5/2/08	Ace Prab II.	es.
		Ace Prab III.	es.
	11/8/08	Prab N. Buat tabel	
		sebelum dan setelah Opti	es.
		man (analisa!)	
	15/8/08	Perbaiki dan hasil analisa dan	es.
		presentasi hasil yg. dan buat video	
		Ace. Seminar -	es.

Malang, 200
Dosen Pembimbing,

Ir. H. Choirul Saleh, MT
NIP. Y. 1018800190

Form.S-4b



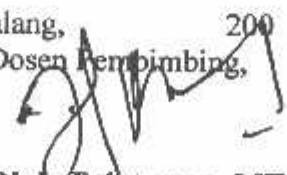
INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

FORMULIR BIMBINGAN SKRIPSI

Nama : FUD NASPUDDIN
Nim : 01.12.159
Masa Bimbingan : 25 November 2007 s/d 25 Mei 2008
Judul Skripsi : **OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE KOMBINASI EVOLUTIONARY PROGRAMMING-EVOLUTIONARY STRATEGY-GENETIC ALGORITHM PADA SALURAN TRANSMISI 150 kV SUB-SYSTEM PAITON-BALI**

No.	Tanggal	Uraian	Parap Pembimbing

Malang, 200
Dosen Pembimbing,


Ir. Djojo Priatmono, MT
NIP. Y. 1018500107

Form.S-4b



PT PLN (PERSERO)
PENYALURAN DAN PUSAT PENGATUR BEBAN JAWA BALI
REGION JAWA TIMUR & BALI

Jalan Suningrat No. 45 Taman Sidoarjo 61257

Telepon : (031) 7882113, 7882114

Cetak Pos : 4119 SBS

Facsimile : (031) 7882578, 7881024

E-mail : region4@pln-jawa-bali.co.id

Website : www.pln-jawa-bali.co.id

Nomor : 186 /330/RJTB/2007
Surat Sdr. No. : ITN-1440/III.TA-2/2/07
Lampiran : 1 (satu) lampiran.
Perihal : Ijin Survey / Pengambilan Data.

12 SEP 2007

Kepada Yth

INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
Di
MALANG

Menunjuk surat Saudara nomor : ITN-1440/III.TA-2/2/07 tanggal 02 Agustus 2007 perihal Survey / Permintaan Data, dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada Mahasiswa Saudara, bernama :

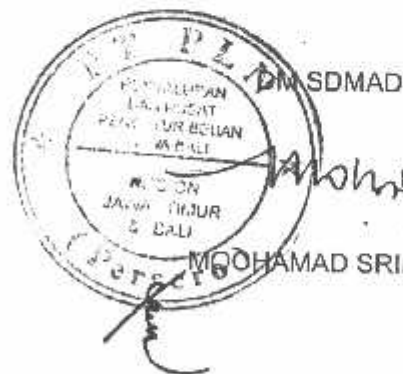
> **Fud Naspuddin**

Nim : 01.12.159

Untuk melakukan Pengambilan Data pada PT. PLN (Persero) P3B Region Jawa Timur dan Bali Bidang Operasi, dengan persyaratan sebagai berikut :

1. Mahasiswa tersebut diatas supaya mengisi dan menanda tangani Surat Pernyataan 1 (satu) lembar bermeteral Rp. 6.000,-
2. Mahasiswa yang bersangkutan agar mematuhi peraturan/ketentuan yang berlaku di PT. PLN (PERSERO) sehingga faktor-faktor kerahasiaan harus benar-benar diutamakan.
3. Semua biaya perjalanan, penginapan, makan dan lain sebagainya tidak menjadi tanggungan PT. PLN (Persero) P3B Region Jawa Timur dan Bali.
4. Buku Laporan Kerja Praktek Mahasiswa tersebut agar dikirimkan kepada PT. PLN (Persero) P3B Region Jawa Timur dan Bali 1 (satu) buah.
5. Untuk informasi lebih lanjut dapat menghubungi PT. PLN (Persero) P3B Region Jawa Timur dan Bali Cq. Bidang SDM & ADMINISTRASI.

Demikian harap maklum dan terima kasih atas perhatian saudara.



Tembusan Yth. :

1. MBSDM PLN P3B JB.
2. DM Operasi PLN P3B RJTB.
3. Sdr. Fud Naspuddin

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : FUD NASPUDDIN
Pria/ Wanita : Pria
Tempat / Tanggal lahir : SLIMBAWA / 1 JUNI 1982
Alamat / No. telepon : JL SEMANGGI TIMUR NO 1 MALANG
08523317774
Pekerjaan : MAHASISWA ITN MALANG

Dengan ini saya menerangkan bahwa :

1. Saya bersedia dan setuju menanggung semua akibat yang ditimbulkan karena kesalahan maupun kelalaian saya dan semua akibat lainnya yang terjadi pada instalasi peralatan milik PLN selama melakukan Training/ Praktek Kerja/ Riset pada PT PLN (Persero) P3B Region Jawa Timur dan Bali, yang telah mendapat ijin dari PT PLN (Persero) P3B Region Jawa Timur dan Bali ;
2. Saya atas peringatan pertama akan membayar sepenuhnya, semua biaya yang langsung menimbulkan kerugian atau kecelakaan , karena kelalaian saya ;
3. Saya akan segera mematuhi semua petunjuk -petunjuk yang diberikan oleh Petugas PT PT PLN (Persero) P3B Region Jawa Timur dan Bali ;
4. Saya sanggup tidak membocorkan hal - hal yang bersifat rahasia perusahaan PT PLN (Persero) P3B Region Jawa Timur dan Bali dan bahan yang saya peroleh dalam Training/ Praktek Kerja/ Riset, dan tidak saya pergunakan untuk hal - hal yang dapat merugikan PT PLN (Persero) P3B Region Jawa Timur dan Bali
5. Saya sanggup menanggung sendiri segala sesuatu untuk keperluan Training/ Praktek Kerja/ Riset termasuk biaya perjalanan , penginapan makan dan sebagainya ;
6. Saya sanggup menyerahkan 1 (satu) buah buku laporan Training/ Praktek Kerja/ Riset kepada PT PLN (Persero) P3B Region Jawa Timur dan Bali, setelah saya presentasikan kepada Manager Bidang SDMAD PT PLN (Persero) P3B Region Jawa Timur dan Bali mengenai tugas Training/ Praktek Kerja/ Riset.
7. Saya tunduk dan akan mentaati semua peraturan yang berlaku di PT PLN (Persero) P3B Region Jawa Timur dan Bali, dan saya sanggup tidak meninggalkan tugas kedinasan selama Training/ Praktek Kerja/ Riset.

Surabaya, 28-8-2007
Yang membuat pernyataan



