

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



***OPTIMAL POWER FLOW MENGGUNAKAN METODE
IMPROVED EVOLUTIONARY PROGRAMMING (IEP)
PADA SUB-SISTEM 150 kV PAITON-BALI***

SKRIPSI

Disusun Oleh :

WIKY INDRA NURCAHYONO

NIM: 00.12.082

SEPTEMBER 2006

INSTITUT TEKNOLOGI NASIONAL MALAYA
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO & I
KONSENTRASI TEKNIK ENERGI LISTRIK

OPTIMAL POWER FLOW MENGGUNAKAN METODE
IMPROVED EVOLUTIONARY PROGRAMMING (IEP)
PADA SUB-SISTEM 150 KV PATUN-BALI

SKRIPSI

Disusun Oleh :

WIKY INDIRA NURCAHYONO

NPM 00.12.002

SEPTEMBER 2008

LEMBAR PERSETUJUAN

**OPTIMAL POWER FLOW MENGGUNAKAN METODE
IMPROVED EVOLUTIONARY PROGRAMMING (IEP)
PADA SUB-SISTEM 150 kV PAITON-BALI**

SKRIPSI

*Disusun Untuk Melengkapi Dan Memenuhi Syarat
Guna Mencapai Gelar Sarjana Teknik*

Disusun Oleh:

**WIKY INDRA NURCAHYONO
00.12.082**



**Mengetahui,
Ketua Jurusan Teknik Elektro**

**Ir. F. Yudi Limpraptono, MT.
NIP. Y. 103 9500 274**

**Menyetujui,
Dosen Pembimbing**

**Ir. Yusuf Ismail Nakhoda, MT.
NIP. Y. 101 8800 189**

**JURUSAN TEKNIK ELEKTRO S-1
KONSENTRASI TEKNIK ENERGI LISTRIK
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI NASIONAL MALANG
2006**

ABSTRAKSI

OPTIMAL POWER FLOW MENGGUNAKAN METODE IMPROVED EVOLUTIONARY PROGRAMMING PADA SUB SISTEM 150 kV PAITON DAN BALI

**(Wiky Indra Nurcahyono, Nim 00.12.082, Teknik Elektro/T.Energi Listrik)
(Dosen Pembimbing: Ir. Yusuf Ismail Nakhoda, MT.)**

Kata Kunci: *Optimal Power Flow, Improved Evolutionary Programming, reassignment strategy.*

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. 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.

Skripsi ini menganalisis permasalahan *Optimal Power Flow* atau dengan menggunakan metode *Improved Evolutionary Programming*. Hasil dari analisa tersebut nantinya dapat digunakan sebagai salah satu acuan dalam operasi pembangkit dan penyaluran daya yang ekonomis dan optimal, terutama mengenai optimasi biaya pembangkitan. Input dari program ini adalah biaya bahan bakar (*fuel cost*) dan hasil perhitungan aliran daya. Sedangkan hasil akhir dari program ini yaitu hasil perhitungan aliran daya, tegangan dan sudut fasa tiap-tiap bus serta biaya pembangkit yang paling optimum.

Analisa dilakukan dengan bantuan program komputer dengan menggunakan bahasa pemrograman Delphi versi 7.0 dan telah sukses diuji cobakan pada sub sistem 150 kV Paiton dan Bali yang terdiri dari 26 bus, dimana telah berhasil dilakukan penghematan biaya pembangkitan sebesar Rp. 662,705.01 - per jam atau terjadi optimasi biaya sebesar 14.52 %.

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadiran Allah SWT, berkat limpahan Rahmat dan Hidayah-Nya, sehingga penyusun dapat menyelesaikan skripsi ini.

Skripsi ini disusun sebagai salah satu persyaratan dalam menyelesaikan studi program strata satu (S-1) jurusan Teknik Elektro, Program Studi Teknik Energi Listrik, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang

Sebelum dan selama penyusunan skripsi ini, penyusun telah banyak mendapatkan bantuan dan bimbingan dari berbagai pihak. Untuk itu pada kesempatan ini penyusun menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. F. Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
4. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Dosen pembimbing.
5. Bapak Choirul Anam, selaku Manager PT. PLN (Persero) P3B Region Jawa Timur Dan Bali.
6. Guruku serta Ibu Bapakku yang selalu memberikan kasih sayang dan dukungan hingga terselesaikannya skripsi ini juga .

7. Serta semua pihak yang turut serta membantu menyelesaikan skripsi ini.
“*Teratai Tunjung*” you give me spirit for finish my job.

Penyusun menyadari akan segala kekurangan yang ada dalam skripsi ini, maka dengan kerendahan hati penyusun mengharapkan kritik dan saran demi penyempurnaan skripsi ini. Akhirnya, kepada semua pihak yang telah bekerja keras dan bersungguh-sungguh hingga terwujudnya skripsi ini, saya menyampaikan penghargaan dan terima kasih.

Malang, September 2006

Penulis

Lembar Persembahan

First I would to thank you Allah SWT, yang menuntunku secara luar biasa, dengan kebesaranMu lah skripsiku ini telah terselesaikan.

Biggest thank's to Ibu bapakku yang selalu memberikan motivasi, dorongan serta cinta kasih sayang kepadaku, dengan penantian yang panjang akhirnya skripsiku selesai. Serta Guruku terimakasih dengan "Teratai Tunjung" you give me spirit to finish my job.

Dosen pembimbingku Ir.Yusuf Ismail Nakhoda, MT. yang nggak bosan2nya memberikan masukan, nasehat serta tentunya bimbingan dalam menyelesaikan tugas akhirku ini matur suwun banget pak...

Kaka'q ku doakan biar dapat kerja terus traktirane buozz....lan ade'q sekolah sing pinter n terima kasih banyak dorongannya, Pa'de Agung suwun banget, de' Mega kuliah tu ga seenak SMU bro tapi have fun aja deh..., de' Jemi mangan2ne rek mari dadian. Blalang tempurku yang selalu mengantarku kemanapun ku pergi dan sekarang tergeletak sakit, Smash temen baruku yang menemaniku hingga ujian akhir thank's bro!!!

Mas Joko terima kasih banyak petuah bijaknya kalo ga' ada mas sekarang skripsiku mungkin belum kelar. My prend "TI", kang Don ndang lulus ble ga' enak sekolah lama2, yoko, reza, kiki, mas Herlan, Adam, mas you-lee. Fitri crewet banget jadi cewe' kan kasian Mukhlis aje nya, n teman seperjuangan KTW.

Sahabat, sohibku, temen sperjuanganku Agung "MUJAEER" ternyata **mission imposible** kita jadi kenyataan ha..ha..ha..., Widya "TV KOTAK" mayak pi2... akhirnya kita lulus ble!!!, Desta "GONDHES BRENGOS" you really my best prend thx akhirnya dengan jurnal yang ente berikan ke ane, ane bisa lulus..., terus nguber tok kecut....thanks guys. Keep fighting for your future...

Temen2 kontrakan puskopat Nyos. Zaenalk's, Ambon, Engkong, Pak Ek'(jok mbanyol tok wae...), Tommy (cepat diajukan Ta nya jangan banyak nunggu), Rully, Lukman AREMA (Mayak man2...), Aryo n Erma dua sejoli yang tak terpisahkan he..he..he...ayo SMANGAT ndang lulus ga usah takut mbodo ae loos pokoknya maju terus pantang mundur, Mojaco, Roby.

Temenku sobat arek kedawung ex SMU 7 Brudien PPM n Astrid kapan ente...??!!, Ditta unibraw sorry..., icang, manusia planet satunya datang tak dijemput pulang tak diantar.

And last for "Al", 'Masih (Sahabatku, temenku, n...q) thank's for all you're my inspiration, I'd like to think that I can go my own way and meet you in the end.

Akhirnya terima kasih pada semua temen2 seangkatan dan semua yang belum kusebutkan dalam lembar persembahanku ini matur suwun sanget I'm only human guys...

DAFTAR ISI

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI.....	iii
KATA PENGANTAR.....	iv
DAFTAR ISI	vi
DAFTAR GAMBAR.....	x
DAFTAR TABEL	xii

BAB I PENDAHULUAN

1.1. Latar Belakang	1
1.2. Rumusan Masalah.....	2
1.3. Tujuan	2
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian.....	3
1.6. Sistematika Penulisan.....	4
1.7. Kontribusi Penelitian	5

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

2.1. Sistem Tenaga Listrik	6
2.2. Karakteristik Unit Pembangkit.....	8
2.2.1. Karakteristik <i>Input-Output</i>	8

2.2.2. Karakteristik <i>Heat-Rate</i>	11
2.2.3. Karakteristik <i>Incremental Heat-Rate dan Incremental Fuel Cost</i>	12
2.3. <i>Economic Dispatch</i>	13
2.3.1. <i>Economic Dispatch Dengan Mengabaikan Rugi – Rugi Transmisi</i>	13
2.3.2. <i>Economic Dispatch Dengan Memperhitungkan Rugi – Rugi Transmisi</i>	16
2.4. Fungsi Biaya Bahan Bakar.....	17
2.5. Sistem Jaringan Transmisi	18
2.5.1. Saluran Transmisi Pendek.....	19
2.5.2. Saluran Transmisi Menengah	20
2.5.3. Saluran Transmisi Panjang.....	21
2.6. Sistem Per-Unit	22

BAB III OPTIMAL POWER FLOW MENGGUNAKAN METODE

IMPROVED EVOLUTIONARY PROGRAMMING

3.1. Analisa Aliran Daya	24
3.1.1. Klasifikasi Bus.....	25
3.1.2. Metode Newton-Raphson.....	26
3.2. <i>Evolutionary Programming</i>	28
3.2.1. Parameter <i>Evolutionary Programming</i>	30
3.2.2. Mekanisme <i>Evolutionary Programming</i>	31
3.3. <i>Simulated Annealing(SA)</i>	34

3.4. Formulasi Masalah <i>Optimal Power Flow</i>	35
3.5. <i>Improved Evolutionary Programming</i> Dalam <i>Optimal Power Flow</i>	38
3.5.1. Representasi Solusi	38
3.5.2 Inisialisasi	38
3.5.3. <i>Inisialisasi Temperature dan Cooling Procedure</i>	39
3.5.4. Mutasi.....	40
3.5.5. Seleksi	42
3.5.6. <i>Reassignment Strategy</i>	43
3.5.7. <i>Criteria Termination</i>	44

BAB IV HASIL DAN ANALISA HASIL

4.1. Program Komputer <i>Optimal Power Flow</i> Menggunakan Metode <i>Improved Evolutionary Programming</i>	45
4.1.1. Algoritma Program	45
4.1.2. Flowchart Program.....	48
4.2. Validasi Program Menggunakan Data Sistem IEEE 30	51
4.2.1. Data Sistem IEEE 30.....	51
4.2.2. Hasil PerhitunganValidasi IEEE 30 menggunakan IEP.....	53
4.3. Data Pembangkitan Thermal Pada Sub Sistem Paiton-Bali.....	56
4.4. Data Pembangkitan dan Pembebanan Sub-Sistem 150 KV Paiton-Bali.....	60
4.5. Data Saluran Transmisi Sub Sistem 150 KV Paiton-Bali.....	61
4.6. Prosedur Pelaksanaan Program Perhitungan.....	62

4.7. Hasil dan Analisa Perhitungan Optimal Power Flow Menggunakan Metode Metode <i>Improved Evolutionary Programming</i> Sub Sistem 150 kVPaiton-Bali	70
4.7.1. Hasil Perhitungan Sebelum Optimasi	70
4.7.2. Hasil Perhitungan Setelah Optimasi	72
4.8. Perbandingan Hasil Perhitungan sebelum dan setelah Optimasi Optimal Power Flow Menggunakan Metode <i>Improved Evolutionary Programming</i>	75
4.8.1. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus	75
4.8.2. Perbandingan Tingkat Rugi-Rugi Daya Pada Saluran	76
4.8.3. Tingkat Optimum Biaya Pembangkitan	76

BAB V KESIMPULAN DAN SARAN

5.1. Kesimpulan	78
5.2. Saran	79

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar 2-1	Unit Boiler-Turbin-Generator	10
Gambar 2-2	Kurva Karakteristik <i>Input-Output</i> Pembangkit Thermal	11
Gambar 2-3	Kurva Karakteristik <i>Heat-Rate</i> Unit Pembangkit.....	11
Gambar 2-4	Kurva Karakteristik <i>Incremental Heat-Rate / Fuel Cost</i>	13
Gambar 2-5	N Unit Pembangkit Thermal Melayani Beban P_R	14
Gambar 2-6	N Buah Pembangkit Thermal Melayani Beban P_R Melalui Saluran Transmisi	17
Gambar 2-7	Rangkaian Setara Saluran Transmisi	18
Gambar 2-8	Rangkaian Setara Saluran Transmisi Pendek.....	19
Gambar 2-9	Rangkaian Setara Saluran Transmisi Menengah.....	20
Gambar 2-9	Rangkaian Setara Saluran Transmisi Menengah.....	21
Gambar 3-1	Ilustrasi Mutasi Gaussian	33
Gambar 4-1	Flowchart Perhitungan sebelum Optimasi	48
Gambar 4-2	Flowchart Program Optimal Power Flow Menggunakan Metode Newton Raphson.....	49
Gambar 4-3	Flowchart Perhitungan Aliran Daya Metode Metode <i>Improved Evolutionary Programming</i>	50
Gambar 4-4	Tampilan Program & Input Data IEEE 30.....	53
Gambar 4-5	Tampilan Parameter IEP	54
	(a) Hasil load Flow	54
	(b) Hasil Perhitungan biaya pembangkitan.....	55
Gambar 4-6	Singel Line Diagram Sub Sistem 150 kV Paiton - Bali.....	58

Gambar 4-7	Tampilan Utama Program	57
Gambar 4-8	Tampilan Masukan Data	58
Gambar 4-9	Tampilan Data Bus	58
Gambar 4-10	Tampilan Data Saluran	59
Gambar 4-11	Tampilan Data Generator.....	59
Gambar 4-12	Tampilan Parameter IEP	59
Gambar 4-13	Tampilan Hasil Load Flow pada Kondisi Awal (Sebelum Optimasi).....	60
Gambar 4-14	Tampilan Hasil Aliran Daya antar Saluran pada Kondisi Awal (sebelum optimasi).	60
Gambar 4-15	Tampilan <i>Summary Load flow</i> Kondisi Awal (Sebelum Optimasi).....	61
Gambar 4-16	Tampilan Hasil Program	62
Gambar 4-17	Tampilan Hasil <i>Load flow</i> pada Kondisi Akhir (Setelah Optimasi).....	62
Gambar 4-18	Tampilan Hasil Aliran Daya antar Saluran pada Kondisi Akhir (Setelah Optimasi)	63
Gambar 4-19	Tampilan <i>Summary Load flow</i> Akhir (Setelah Optimasi).....	63

DAFTAR TABEL

Tabel 4-1	Data generator dan koefisien biaya IEEE 30	51
Tabel 4-2	Data Bus IEEE 30.....	51
Tabel 4-3	Data Saluran Impedansi IEEE 30	52
Tabel 4-4	Perbandingan Data Referensi Jurnal Dengan Data Optimasi	55
Tabel 4-5	Parameter Unit Pembangkit Thermal	56
Tabel 4-6	Persamaan Biaya Pembangkitan Unit Pembangkit Thermal Paiton dan Bali	57
Tabel 4-7	Penomoran Bus Sub Sistem 150 kV Paiton dan Bali.....	59
Tabel 4-8	Data Pembangkitan dan Pembebanan Sub Sistem 150 kV Paiton dan Bali	60
Tabel 4-9	Data Saluran Transmisi Sub Sistem 150 kV Paiton dan Bali	62
Tabel 4-10	Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sebelum Optimasi	70
Tabel 4-11	Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi.....	71
Tabel 4-12	Hasil Perhitungan Aliran Daya Sebelum Optimasi	71
Tabel 4-13	Hasil Perhitungan Daya yang Dibangkitkan dan Biaya Operasi Sebelum Optimasi	72
Tabel 4-14	Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Setelah Optimasi	72
Tabel 4-15	Hasil Perhitungan Aliran Daya antar Saluran Setelah Optimasi.....	73

Tabel 4-16	Hasil Perhitungan Aliran Daya Setelah Optimasi	74
Tabel 4-17	Hasil Perhitungan Daya yang Dibangkitkan dan Biaya Operasi Setelah Optimasi.....	74
Tabel 4-18	Perbandingan Tingkat Tegangan dan Sudut Tegangan pada Tiap Bus Sebelum dan Sesudah Optimasi	75
Tabel 4-19	Perbandingan Tingkat Optimum Biaya Pembangkitan Sebelum dan Setelah Optimasi	77

DAFTAR GRAFIK

Grafik 4-1	Hasil Perhitungan Daya Yang Dibangkitkan Sebelum Optimasi.....	72
Grafik 4-2	Hasil Perhitungan Daya Yang Dibangkitkan Setelah Optimasi.....	74
Grafik 4-3	Perbandingan rugi-rugi daya	76
Grafik 4-4	Tingkat Optimum Biaya Pembangkitan	77

BAB I

PENDAHULUAN

1.1. Latar Belakang

Sistem tenaga listrik secara garis besar dapat dibagi menjadi 3 bagian yaitu: sisi pembangkit tenaga listrik, jaringan transmisi dan beban. 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. 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.

Sehingga koordinasi antara unit-unit pembangkit yang ada pada sistem tenaga listrik sangat diperlukan untuk mencapai biaya operasi yang se-optimum mungkin, dalam hal ini yang dimaksud adalah optimum secara ekonomis, dengan tetap memperhatikan besar beban yang ada dan juga kestabilan tegangan sistem.

Sejak diperkenalkan sebagai *network constrained economic dispatch* oleh Capentier dan didefinisikan sebagai *Optimal Power Flow (OPF)* oleh Dommel dan Tinney ^[1]. OPF dipakai untuk mengoptimalkan fungsi objektif operasi sistem

tenaga listrik yaitu biaya operasi pembangkit thermal dan sekaligus memberikan seperangkat batasan operasi sistem, yang menyangkut batasan yang ditentukan oleh jaringan listrik.^[2]

Dalam skripsi ini akan dibahas metode *Improved Evolutionary Programming* (IEP), yakni suatu metode optimasi yang berusaha menyerupai operasi – operasi seleksi alamiah. IEP secara simultan dapat mengevaluasi banyak titik yang ada dalam ruang lingkup parameter secara sekaligus dan mengarah pada pencapaian *optimum solution*.

1.2. Rumusan Masalah

Berdasarkan hal di atas disusun sebuah pokok permasalahan yaitu seberapa besar kemampuan perhitungan dari metode *Improved Evolutionary Programming* dalam pemecahan masalah *Optimal Power Flow* (OPF) dalam menganalisa penyaluran daya aktif sehingga dapat menekan rugi-rugi daya serta dapat meminimalisasi biaya pembangkitan dalam sistem .

1.3. Tujuan

Tujuan dari penelitian skripsi dengan menggunakan metode *Improved Evolutionary Programming* (IEP) ini adalah:

Untuk memecahkan masalah *Optimal Power Flow* (OPF). Dalam hal ini kriteria yang dibahas adalah pengaturan keluaran daya reaktif dan aktif pada suatu sistem tenaga listrik, dengan memperhatikan batasan-batasan yang ada dalam sistem, sehingga diharapkan diperoleh biaya operasi pembangkitan yang optimal,

1.4. Batasan Masalah

Dalam skripsi ini akan dilakukan analisis terhadap OPF yang menggunakan metode IEP pada sub sistem 150 kV Paiton dan Bali. Agar pembahasan mengarah sesuai tujuan, maka pembahasan dalam skripsi ini dibatasi oleh hal sebagai berikut :

1. Analisis dilakukan dengan asumsi bahwa sistem berada dalam operasi normal.
2. Analisa hanya dilakukan pada pembangkit thermal saja
3. Tidak membahas masalah-masalah berikut, yaitu:
 - a. kapasitas saluran, biaya, *start up* dan *shut down* serta masalah *spinning reserve*.

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.

Pengumpulan data lapangan yang dipakai dalam objek penelitian yakni data impedansi saluran transmisi dan data pembebanan sub sistem 150 kV Paiton dan Bali serta data pembangkit thermal.

Bentuk data yang digunakan:

- a. Data Kuantitatif, yaitu data yang dapat dihitung atau data yang berbentuk angka-angka.
 - b. Data Kualitatif, yaitu data yang berbentuk diagram.
3. Analisa Data yaitu; Analisa perhitungan tegangan, sudut phasa tegangan, aliran daya dan rugi daya pada tiap saluran serta biaya pembangkitan dengan menggunakan metode *Improved evolutionary programming* yang disimulasikan dengan komputer.
 4. Simulasi dan pembahasan masalah

Pembahasan masalah pada skripsi ini disimulasikan dengan komputer berbahasa Pemrograman Delphi versi 7.0. Sehingga dapat di ketahui perbandingan metode yang di terapkan lebih efisien atau ekonomis di bandingkan dengan yang di gunakan pada P3B.

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 : LANDASAN TEORI

Menguraikan pembahasan sistem tenaga listrik dan operasi ekonomis (*economic dispatch*) dan karakteristik unit pembangkit secara umum.

**BAB III : *OPTIMAL POWER FLOW* MENGGUNAKAN METODE
*IMPROVED EVOLUTIONARY PROGRAMMING***

Menguraikan teori dasar dari aliran daya, metode aliran daya *Newton Raphson*, teori EP, adaptasi EP dengan penambahan *improved* ke permasalahan OPF dan prosedur dari IEP.

BAB IV : HASIL DAN ANALISIS HASIL

Menguraikan alur program, hasil validasi, serta hasil perhitungan OPF menggunakan metode IEP.

BAB V : KESIMPULAN DAN SARAN

Memuat intisari dan hasil pembahasan, yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan selanjutnya.

1.7. Kontribusi Penelitian

Kontribusi dari skripsi ini adalah penggunaan *Optimal Power Flow* dengan metode perhitungan *Improved Evolutionary Programming* pemecahan masalah bagi P.T. PLN dalam usaha meminimalkan biaya operasi khususnya biaya pembangkitan yang merupakan biaya terbesar pada sistem tenaga listrik namun tetap memperhatikan kualitas pelayanan daya listrik.

BAB II

KARAKTERISTIK UNIT PEMBANGKIT, OPERASI EKONOMIS

DAN PENGATURAN SALURAN TRANSMISI

2.1. Sistem Operasi Pada Sistem Tenaga Listrik

Seperti telah diketahui 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 frekwensi dan tegangan sistem untuk berada pada harga normalnya karena adanya perubahan beban pada sistem. Dan seperti yang yang diketahui dan berulang kali disebutkan bahwa tenaga listrik tidak dapat disimpan sehingga dalam operasinya harus selalu dicapai keseimbangan antara penyediaan dan pemenuhan kebutuhan daya serta juga perlu diingat bahwasanya beban sistem selalu berubah-ubah 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 sesuatu yang sangat penting disamping karakteristik-karakteristik lainnya sehingga dalam operasi hariannya harus berdasarkan lengkung beban harian yang telah dibuat karena dengan lengkung harian ini dapat ditentukan perencanaan operasi pembangkit-pembangkit yang

ada, baik itu unit pembangkit termal maupun hidro. Tentu saja kebutuhan beban dalam satu harinya tidak merata akan tetapi dari jam ke jam berbeda sesuai kebutuhan konsumen. Berdasarkan pada lengkung beban yang telah ada maka dapat ditentukan berapa unit pembangkit yang bekerja dan siap bekerja pada hari itu.

Sebagai dasar pertimbangan 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 tercapai 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 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 grafik perubahan beban terhadap waktu.

Oleh karena biaya operasi untuk memproduksi daya listrik suatu pembangkit hidro (PLTA) sangat kecil jika dibandingkan dengan pembangkit

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 pada dasarnya bertujuan untuk menekan biaya produksi listrik agar harga dari listrik yang dihasilkan dapat ditekan serendah mungkin sehingga dapat memuaskan pemakai listrik.

2.2. Karakteristik Unit Pembangkit

2.2.1. Karakteristik Input-Output^[1]

Hal yang paling mendasar dalam operasi ekonomis adalah dengan membuat karakteristik input-output dari unit pembangkit termal. Karena ini diperoleh dari desain perencanaan atau melalui tes 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 2.1, dimana unit ini memuat sebuah boiler yang menghasilkan uap untuk menjalankan turbin yang dikopel dengan rotor dari generator.

Pada pembangkit termal, input diberikan dalam satuan panas *Btu/jam* atau *Kalori/jam* dari bahan bakar yang diberikan pada boiler untuk menghasilkan output pembangkit. Sedangkan notasi yang digunakan adalah H (*Mbtu/h*) atau dalam satuan yang lain (*Mkal/h*). Adapun dalam penelitian ini, perhitungan yang dilakukan adalah *Mkal/jam*. Selain itu input dari pembangkit dapat pula

termal (PLTU, PLTGU, PLTD) maka dalam pembahasan selanjutnya untuk mendapatkan biaya operasi yang ekonomis akan ditekankan pada unit pembangkit termal 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 dapat dicapai suatu keadaan operasi yang ekonomis.

Pembahasan mengenai operasi ekonomis 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. Adapun karakteristik yang perlu diketahui dari setiap unit pembangkit adalah :

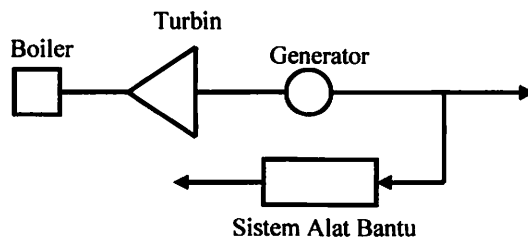
1. Karakteristik input bahan bakar sebagai fungsi dari output daya.
2. Nilai panas sebagai fungsi output daya.
3. Kenaikan jumlah bahan bakar yang dibutuhkan jika terdapat perubahan beban.

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.

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 berikut ini :

$$F = H \times \frac{\text{Rupiah}}{\text{Mbtu}} \dots\dots\dots (2.1)$$

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



Gambar 2.1^[1]
Unit Boiler-Turbin-Generator

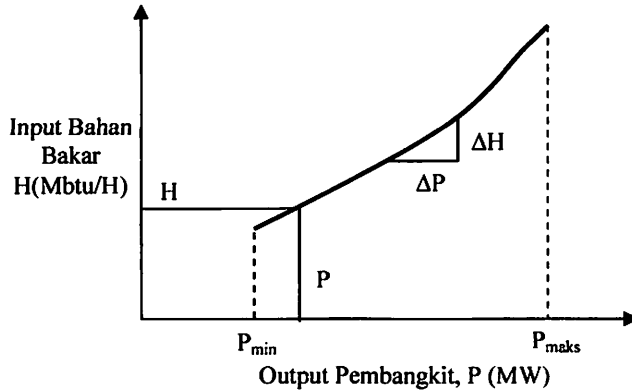
Seperti digambarkan pada gambar 2.1, maka output dari pembangkit tidak hanya untuk dihubungkan dengan sistem saja akan tetapi juga untuk sistem alat bantu dalam pembangkit. Di sini output pembangkit didefinisikan sebagai daya yang dikeluarkan oleh generator untuk beban, diluar untuk keperluan pembangkit itu sendiri. Jadi untuk 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 2.2. dan 2.3 di bawah ini, sedangkan kurva dari karakteristik input-output pembangkit dapat dilihat pada gambar 2.2.

$$H = f(P) \dots\dots\dots (2.2)$$

atau

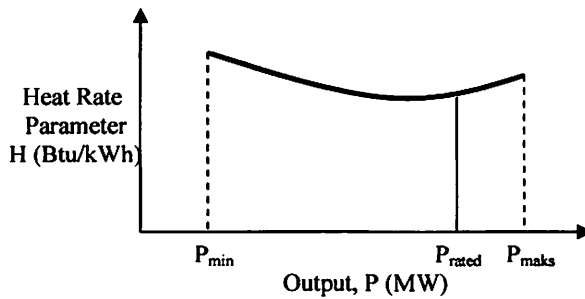
$$F = f(P) \dots\dots\dots (2.3)$$



Gambar 2.2^[1]
Kurva Karakteristik Input-Output Pembangkit Termal

2.2.2. Karakteristik Heat-Rate^[1]

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 2.3 di bawah ini :



Gambar 2.3^[1]
Kurva Karakteristik *Heat-Rate* Unit Pembangkit

2.2.3. Karakteristik *Incremental Heat Rate* dan *Incremental Fuel Cost*^[1]

Perwujudan yang lain dari karakteristik pembangkit adalah karakteristik *incremental heat rate* dan karakteristik *incremental fuel cost*. Karakteristik ini menyatakan hubungan daya output pembangkit sebagai fungsi *incremental heat rate* atau *incremental fuel cost*. Karakteristik *incremental heat rate* ini menunjukkan besarnya perubahan input energi bila ada perubahan output pembangkit pada megawatt output pembangkit.

Kurva karakteristik *incremental heat rate* atau *incremental fuel cost* dapat dilihat pada gambar 2.4. Sedangkan persamaan dari *incremental heat rate* dan persamaan *incremental fuel cost* dapat dilihat pada persamaan 2.4 sampai persamaan 2.7.

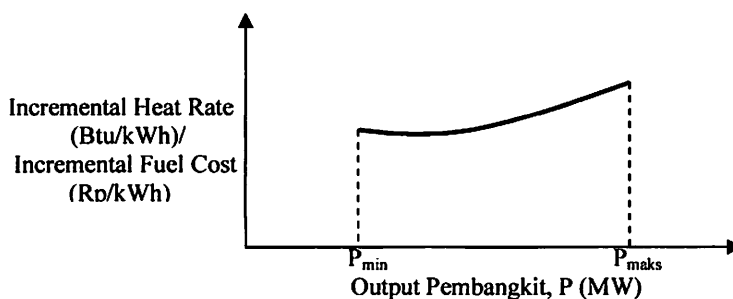
$$\text{Incremental heat-rate} = \frac{\Delta H \left(\frac{\text{MBtu}}{\text{kWh}} \right)}{\Delta P} \dots\dots\dots (2.4.)$$

$$\text{Incremental fuel cost} = \frac{\Delta F \left(\frac{\text{Rupiah}}{\text{kWh}} \right)}{\Delta P} \dots\dots\dots (2.5.)$$

Dengan mengambil harga ΔP mendekati nol, maka dapat dinyatakan dengan persamaan berikut ini :

$$\text{Incremental heat rate} = \frac{dH \left(\frac{\text{MBtu}}{\text{kWh}} \right)}{dP} \dots\dots\dots (2.6.)$$

$$\text{Incremental fuel cost} = \frac{dF \left(\frac{\text{Rupiah}}{\text{kWh}} \right)}{dP} \dots\dots\dots (2.7.)$$



Gambar 2.4^[1]
Kurva Karakteristik *Incremental Heat Rate/Fuel Cost*

2.3. *Economic Dispatch*

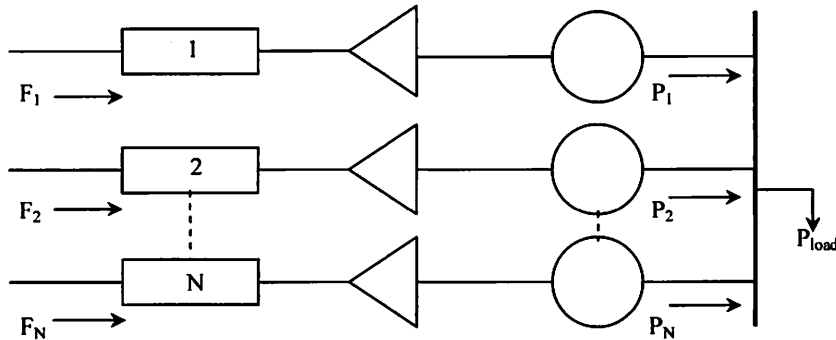
Dalam pembahasan tentang operasi pada sistem tenaga listrik yang ekonomis, maka kita akan selalu membicarakan tentang *economic dispatch*. Pengertian dari *economic dispatch* itu sendiri adalah pembagian pembebanan pada pembangkit-pembangkit yang ada dalam suatu tenaga listrik, secara optimum ekonomis pada beban tertentu. Dengan dilakukannya *economic dispatch* maka akan didapatkan biaya pembangkitan 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.

2.3.1. *Economic Dispatch Dengan Mengabaikan Rugi-rugi Transmisi*^[1]

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 2.5. Sistem ini terdiri dari N buah unit pembangkit termal yang dihubungkan pada

single busbar 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.



Gambar 2.5^[1]
N Unit Pembangkit Termal Melayani Beban P_R

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah dari biaya dari masing-masing unit pembangkit. Dan pembatas yang paling penting adalah bahwa jumlah output dari masing-masing unit pembangkit sama dengan beban di konsumen. Yang menjadi permasalahan adalah meminimalkan 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 berikut :

$$F_T = F_1 + F_2 + F_3 + \dots + F_N \dots \dots \dots (2.8)$$

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

$$\phi = 0 = P_R - \sum_{i=1}^N P_i \dots \dots \dots (2.9)$$

Persamaan di atas adalah pembatas yang merupakan masalah dari optimasi dan ini dapat dipecahkan dengan menggunakan kalkulus tingkat lanjut yang melibatkan fungsi Lagrange. Dimana fungsi didapat dengan cara menambahkan pembatas ϕ yang telah dikalikan dengan faktor pengali Lagrange λ pada fungsi tujuan F_T . Fungsi Lagrange dapat ditunjukkan dengan persamaan di bawah ini :

$$L = F_T + \lambda\phi \dots\dots\dots (2.10)$$

Persamaan Lagrange di atas merupakan fungsi dari output pembangkit P_i dan faktor pengali Lagrange λ . Keadaan optimal dari fungsi tujuan F_T dapat diperoleh dengan operasi gradien dari persamaan Lagrange sama dengan nol.

$$\frac{\partial L}{\partial P_i} = \frac{dF_i(P_i)}{dP_i} - \lambda = 0 \dots\dots\dots (2.11)$$

atau $0 = \frac{dF_i}{dP_i} - \lambda \dots\dots\dots (2.12)$

Persamaan terakhir ini menunjukkan bahwa bila diinginkan biaya bahan bakar, F_T , yang paling minimum maka *incremental cost* setiap unit pembangkit harus sama yaitu sebesar λ . Kondisi optimal ini tentunya dengan tetap memperhatikan pembatas yang ada yaitu bahwa daya setiap unit pembangkit harus lebih besar atau sama dengan daya output minimum dan lebih kecil atau sama dengan daya output maksimum yang diijinkan.

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

$$\frac{dF_i}{dP_i} = \lambda \text{ ada N buah persamaan} \dots\dots\dots (2.13)$$

$$P_i \min \leq P_i \leq P_i \text{ maks ada } 2 \text{ N pertidaksamaan (2.14)}$$

$$\sum_{i=1}^N P_i = P_R \text{ (2.15)}$$

Dari batasan pertidaksamaan pembatas di atas dapat diperluas menjadi :

$$\frac{dF_i}{dP_i} = \lambda \text{ untuk } P_i \min \leq P_i \leq P_i \text{ maks (2.16)}$$

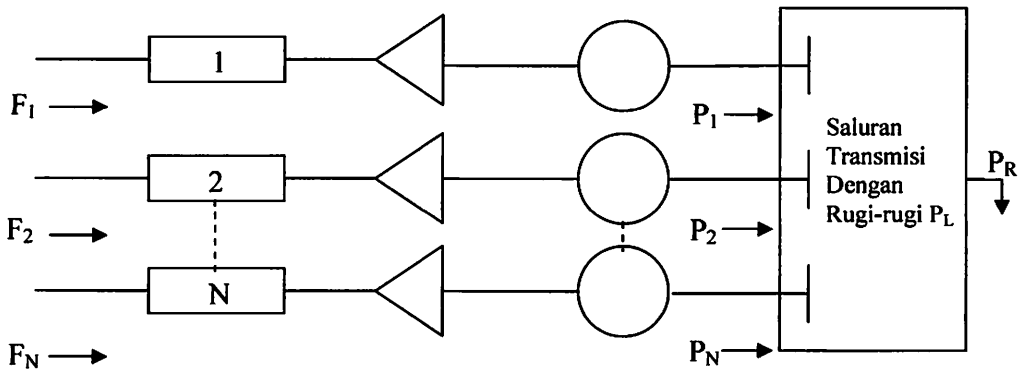
$$\frac{dF_i}{dP_i} \leq \lambda \text{ untuk } P_i = P_i \text{ maks (2.17)}$$

$$\frac{dF_i}{dP_i} \geq \lambda \text{ untuk } P_i = P_i \text{ min (2.18)}$$

2.3.2. *Economic Dispatch Dengan Memperhitungkan Rugi-rugi Transmisi*^[1]

Sistem dengan memperhitungkan rugi-rugi transmisi dapat dilihat pada gambar 2.6. Sistem ini terdiri dari N buah unit pembangkit termal 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.

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah dari biaya dari masing-masing unit pembangkit. Dan batasan yang paling penting adalah bahwa jumlah output dari masing-masing unit pembangkit sama dengan beban di konsumen ditambah rugi-rugi transmisi.



Gambar 2.6^[1]
 N Unit Pembangkit Termal Melayani Beban P_R Melalui Saluran Transmisi

$$P_R + P_L - \sum_{i=1}^N P_i = \phi = 0 \dots\dots\dots (2.19)$$

$$L = F_T + \lambda \phi \dots\dots\dots (2.20)$$

$$\phi = \sum_{i=1}^N P_i - P_R - P_L = 0 \dots\dots\dots (2.21)$$

Persamaan Lagrangennya adalah

$$L = \sum_{i=1}^N F_i - \lambda \left(\sum_{i=1}^N P_i - P_R - P_L \right) \dots\dots\dots (2.22)$$

$$\frac{\partial L}{\partial P_i} = \frac{dF_i}{dP_i} - \lambda \left(1 - \frac{\partial P_L}{\partial P_i} \right) \dots\dots\dots (2.23)$$

2.4. Fungsi Biaya Bahan Bakar^[1]

Biaya bahan bakar merupakan unsur biaya yang penting dalam operasi sistem pembangkit thermal. Fungsi biaya bahan bakar $F_i(P_i)$ untuk setiap

pembangkit terhadap daya keluaran diekspresikan dalam bentuk fungsi kuadrat, yang dapat dinyatakan sebagai berikut:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i \dots \dots \dots (2.24)$$

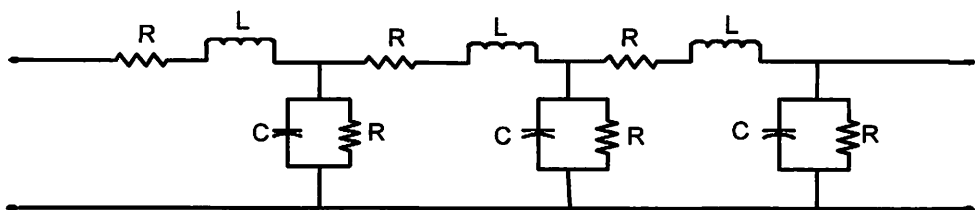
Dimana:

a, b, c = konstanta persamaan dari unit ke - i

P_i = daya keluaran dari unit ke - i

2.5. Sistem Jaringan Transmisi

Pusat-pusat pembangkit tenaga listrik biasanya terletak jauh dari tempat-tempat dimana tenaga listrik tersebut digunakan. Tenaga listrik yang dibangkitkan harus 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.



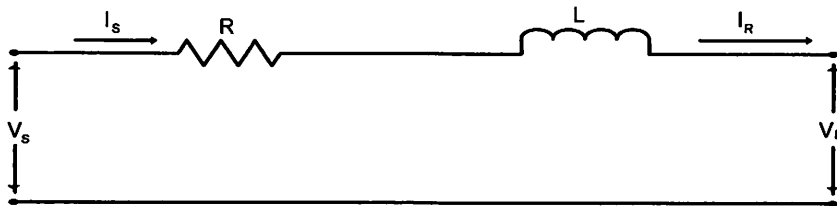
Gambar 2.7^[2]
Rangkaian Setara Saluran Transmisi

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

1. Saluran transmisi pendek : saluran yang panjangnya < 80 km.
2. Saluran transmisi menengah : saluran transmisi yang panjangnya $80\text{--}240$ km.
3. Saluran transmisi panjang : saluran yang panjangnya > 240 km.

2.5.1. Saluran Transmisi Pendek

Rangkaian ekuivalen untuk saluran transmisi pendek diperlihatkan pada gambar 2.8, dimana I_s dan I_R merupakan arus pada ujung pengirim dan ujung penerima, sedangkan V_s dan V_R adalah tegangan saluran terhadap netral pada ujung pengirim dan ujung penerima.



Gambar 2.8^[2]
Rangkaian Setara Saluran Transmisi Pendek

Karena tidak ada cabang paralel (shunt), arus pada ujung–ujung pengirim dan penerima akan sama besar :

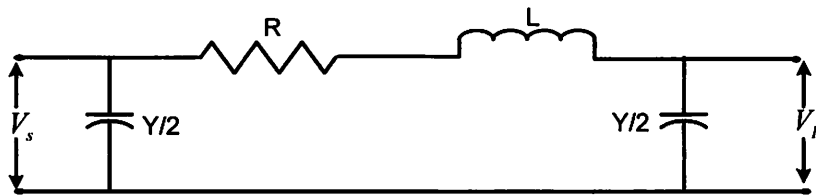
$$I_s = I_R \dots \dots \dots (2.25)$$

Bila kondisi tegangan pada ujung penerima diketahui, maka tegangan pada ujung sisi kirim adalah :

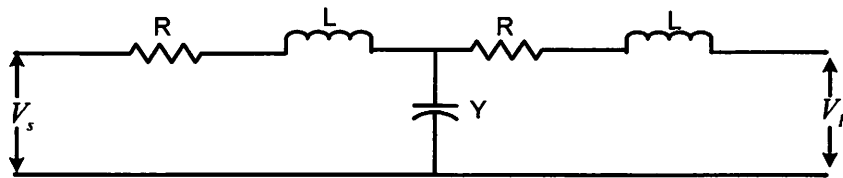
$$V_s = V_R + I_R Z \dots \dots \dots (2.26)$$

2.5.2. Saluran Transmisi Menengah

Bertambahnya saluran menyebabkan kapasitansi shunt bertambah besar dan tidak dapat diabaikan. Saluran transmisi jarak menengah pada umumnya digambarkan dengan rangkaian π atau rangkaian T. Dari dua versi ini rangkaian π lebih umum dipakai dari pada rangkaian T.



(a) Rangkaian Ekivalen π



(b) Rangkaian Ekivalen T

Gambar 2.9^[2]

Rangkaian Setara Saluran Transmisi Menengah

Untuk rangkaian π berlaku :

$$V_s = \left[\frac{ZY}{2} + 1 \right] V_R + ZI_R \dots\dots\dots (2.27)$$

$$I_s = \left[\frac{ZY}{2} + 1 \right] YV_R + \left[\frac{ZY}{4} + 1 \right] I_R \dots\dots\dots (2.28)$$

Untuk rangkaian T berlaku :

$$V_s = \left[\frac{ZY}{2} + 1 \right] V_R + \left[\frac{ZY}{4} + 1 \right] ZI_R \dots\dots\dots (2.29)$$

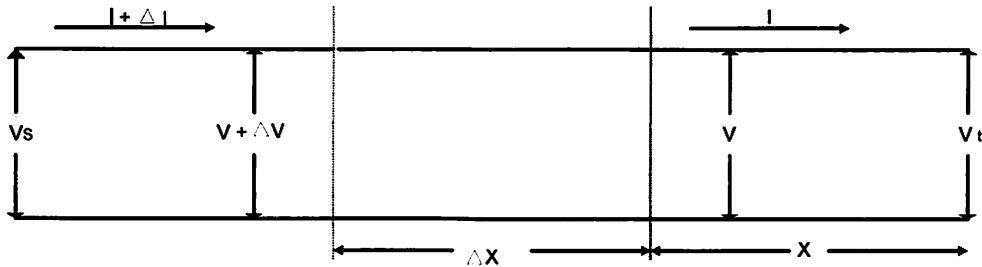
$$I_s = YV_R + \left[\frac{ZY}{2} + 1 \right] I_R \dots\dots\dots (2.30)$$

Dimana :

- V_s, I_s = Tegangan, Arus sisi kirim
- V_R, I_R = Tegangan, Arus sisi terima
- Z = Impedansi seri total saluran transmisi
- Y = Admitansi shunt total saluran

2.5.3. Saluran Transmisi Panjang

Pada saluran panjang parameter-parameter saluran tidak terpusat menjadi satu, melainkan tersebar merata di seluruh panjang saluran.



Gambar 2.10^[2]
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 :

$$V = \frac{V_R + I_R Z_C}{2} e^{yx} + \frac{V_R - I_R Z_C}{2} e^{-yx} \dots\dots\dots (2.31)$$

$$I = \frac{\frac{V_R}{Z_C} + I_R}{2} e^{yx} + \frac{\frac{V_R}{Z_C} - I_R}{2} e^{-yx} \dots\dots\dots (2.32)$$

Persamaan untuk saluran transmisi panjang dapat ditulis dalam bentuk hiperbola sebagai berikut :

$$V_{(x)} = V_R \cosh \tau x + I_R Z_C \sinh \tau x \dots\dots\dots (2.33)$$

$$I_{(x)} = I_R \cosh \tau x + \frac{V_R}{Z_C} \sinh \tau x \dots\dots\dots (2.34)$$

Dimana :

$$\begin{aligned} \tau &= \text{Konstanta rambatan pada saluran} &&= \sqrt{ZY} \\ Z_c &= \text{Impedansi karakteristik saluran} &&= \sqrt{\frac{Z}{Y}} \end{aligned}$$

2.6. Sistem Per-Unit^[3]

Untuk memudahkan proses perhitungan, dalam sistem tenaga listrik digunakan sistem Per-Unit (pu).

$$\text{Besaran Per-Unit} = \frac{\text{Besaran sebenarnya}}{\text{Besaran dasar}} \dots\dots\dots (2.35)$$

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

- Untuk data 1 phasa

Arus dasar

$$I_d = \frac{\text{kVA dasar 1 phasa}}{\text{kV dasar L - N}} \dots\dots\dots (2.36)$$

Impedansi dasar

$$Z_d = \frac{(kV \text{ dasar L - N})^2 \times 10^{-3}}{kVA \text{ dasar 1 fase}} \dots\dots\dots (2.37)$$

$$Z_d = \frac{(kV \text{ dasar L - N})^2}{MVA \text{ dasar 1 fase}} \dots\dots\dots (2.38)$$

■ Untuk data 3 phasa

Arus dasar

$$I_d = \frac{kVA \text{ dasar 3 phasa}}{\sqrt{3} \text{ kV dasar L - L}} \dots\dots\dots (2.39)$$

Impedansi dasar

$$Z_d = \frac{(kV \text{ dasar L - L})^2 \times 10^{-3}}{kVA \text{ dasar 3 phasa}} \dots\dots\dots (2.40)$$

$$Z_d = \frac{(kV \text{ dasar L - L})^2}{MVA \text{ dasar 3 phasa}} \dots\dots\dots (2.41)$$

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

Setelah besaran-besaran dasar telah ditentukan maka besaran-besaran itu dinormalisasikan terhadap besaran dasar. Dengan demikian impedansi persatuan didefinisikan sebagai berikut :

$$Z = \frac{\text{Impedansi sebenarnya } Z(\Omega)}{\text{Impedansi dasar } Z_d(\Omega)} \dots\dots\dots (2.42)$$

BAB III

OPTIMAL POWER FLOW MENGGUNAKAN METODE IMPROVED EVOLUTIONARY PROGRAMMING

3.1. Analisa Aliran Daya^[3]

Dalam melayani beban yang dibutuhkan oleh konsumen dan pengoperasian tenaga listrik perlu dilakukan penganalisaan 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 daya adalah:

1. Mencari harga magnitude tegangan $|v|$ dan sudut fasa tegangan δ bus beban.
2. Mencari daya reaktif Q dan sudut fasa tegangan δ dari generator bus.
3. Untuk mendapatkan 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 daya yang mengalir disetiap saluran jaringan tenaga listrik.

3.1.1. Klasifikasi Bus^[3]

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 (Load Bus) (PQ)

Pada bus ini hanya terdapat kebutuhan daya untuk beban dimana P daya aktif dan Q daya reaktif diketahui, sementara $|v|$ dan δ berubah – ubah menurut kebutuhan. Oleh karena itu, $|v|$ dan δ harus ditentukan (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 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 :

$$S_k = P_k + jQ_k = V_k^* I_k$$

Atau

$$P_k - jQ_k = V_k^* \sum_{n=1}^N Y_{kn} V_n \dots\dots\dots(3.1.)$$

Dari persamaan $V_k = v_k + j\delta_k$ dan $Y_{k11} = G_{k11} - jB_{k11}$, maka persamaan 3.1. menjadi :

$$P_k - jQ_k - (v_k + j\delta_k) \sum_{n=1}^N (G_{kn} - jB_{kn})(V_n + j\delta_n) \dots\dots\dots(3.2.)$$

Bila dituliskan dalam bentuk real dan imajiner maka persamaan di atas menjadi :

$$P_k = \sum_{n=1}^N \{v_k(v_n G_{kn} + \delta_k B_{kn}) + \delta_k(\delta_n G_{kn} - v_k B_{kn})\} \dots\dots\dots(3.3.)$$

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

3.1.2. Metode Newton Rapshon^[5]

Proses yang dilakukan adalah membandingkan antara daya yang ditempatkan berdasarkan data ($P_{k, sched}$ dan $Q_{k, sched}$) dengan daya hasil perhitungan ($P_{k, calc}$ dan $Q_{k, calc}$) menggunakan persamaan (3.3.) dan (3.4.) di atas. Selisih daya yang diterapkan dan perhitungan (ΔP_k dan ΔQ_k) dihitung dengan persamaan :

$$\Delta P_k = P_{k, sched} - P_{k, calc} \dots\dots\dots(3.5.)$$

$$\Delta Q_k = Q_{k, sched} - Q_{k, calc} \dots\dots\dots(3.6.)$$

Selisih daya dihitung dengan persamaan (3.5.) dan persamaan (3.6.) 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 (3.3.) dan (3.4.), yang dalam bentuk matriks dituliskan sebagai:

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

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |v| \end{bmatrix} \dots \dots \dots (3.7.)$$

Dimana elemen-elemen jacobian dapat dihitung dengan menurunkan persamaan-persamaan (3.3.) dan (3.4.)

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 (3.3.) dan (3.4.) 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. *Evolutionary Programming* ^{[6][7][10]}

Evolutionary Programming (EP) merupakan metode stokastik 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 prinsip seleksi alam “siapa yang kuat, dia yang bertahan (*survive*)”. Dengan meniru proses ini, EP dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam dunia nyata.

EP adalah suatu metode strategi optimasi yang merupakan cabang dari *Evolutionary Computation* yang didalamnya terdiri dari Algoritma Genetika, *Genetic Programming*, *Evolutionary Strategies*, dan *Evolutionary Programming*. Perbedaan yang paling mendasar antara *Evolutionary Programming* dengan Algoritma Genetika adalah pada proses operasi. Dalam metode *Evolutionary Programming* tidak menggunakan operasi *crossover* melainkan operasi *competition* (kompetisi).

EP ditemukan oleh Lawrence.J. Fogel dilandasi oleh sifat-sifat evolusi alam. Fogel 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. Fogel mulai bekerja dengan algoritma yang dibentuk dari string-string bilangan real yang disebut *kromosom*. Seperti halnya alam, metode ini menyelesaikan permasalahan-permasalahan dengan menemukan kromosom-kromosom yang baik dengan memanipulasi materi dan sifat (*gene*) kromosom-kromosom. Algoritma ini tidak mengetahui tipe permasalahan yang akan diselesaikan.

Sebelum EP 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 EP yaitu proses produksi (*reproduction*), 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 repereduksi, mutasi dan kompetisi untuk menciptakan keturunan (*offspring*).

EP 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.

3.2.1. Parameter *Evolutionary Programming*

Terdapat beberapa parameter yang digunakan dalam EP. Parameter tersebut digunakan untuk melihat kompleksitas dari EP. Parameter yang digunakan tersebut adalah:

❖ Jumlah Generasi (MAXGEN)

Merupakan jumlah perulangan (iterasi) dilakukannya rekombinasi dan seleksi. Jumlah generasi ini mempengaruhi kestabilan output dan lama iterasi (waktu proses EP). 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 solution*.

❖ Ukuran Populasi (POPSIZE)

Ukuran populasi mempengaruhi kinerja dan efektifitas dari EP. Jika ukuran populasi kecil maka populasi tidak menyediakan cukup materi untuk mencakup ruang permasalahan, sehingga pada umumnya kinerja EP 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 lokal.

❖ Probabilitas Mutasi (P_m)

Mutasi digunakan untuk meningkatkan variasi populasi digunakan untuk menentukan tingkat mutasi yang terjadi, karena frekuensi terjadinya mutasi tersebut menjadi $P_m \times POPSIZE \times N$, 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 EP mutasi menjalankan aturan penting yaitu:

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

❖ 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 memakai *string* bilangan *real*.

3.2.2. Mekanisme *Evolutionary Programming*

A. Pengkodean atau Representasi

Langkah pertama kali yang dilakukan dalam penggunaan EP adalah melakukan pengkodean atau representasi terhadap permasalahan yang akan dilakukan.

Secara umum EP 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 EP untuk awal melakukan pencarian.

B. 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 $f_k(x)$ ($k = 1, 2, \dots, \text{POPSIZE}$).

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.

C. Seleksi^[7]

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 *roulettee wheel selection*, *rank selection*, *elitesm* dan lain sebagainya.

➤ *Roulette Wheel Selection*

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

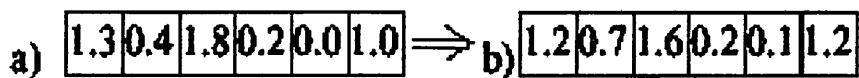
Adapun algoritma dari *roulette-wheel* adalah sebagai berikut:

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.

D. 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 *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*)



Gambar 3.1. ^[13]

Ilustrasi

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.

E. *Competition* (Kompetisi) ^[7]

Dalam tahap kompetisi, mekanisme seleksi dipakai untuk menghasilkan populasi baru dari populasi yang ada. Melalui penggunaan skema kompetisi setiap individu dalam populasi baik orang tua (*parent*) maupun anak (*offspring*) akan dikompetisikan atau 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.

3.3 Simulated Annealing (SA) ^[10]

Optimasi SA adalah suatu simulasi dari proses fisik pendinginan logam dengan pengotoran, di mana suatu partikel cair didinginkan pelan-pelan dari temperatur yang sangat tinggi sampai mengeras pada temperatur rendah. Jumlah iterasi di dalam metode SA adalah penganalisa untuk tingkat temperatur. Sejumlah percobaan dilakukan dalam suatu iterasi. Pada setiap percobaan suatu solusi calon dihasilkan. Manakala solusi calon yang baru mempunyai biaya yang lebih rendah dibanding solusi yang sekarang akan diterima sebagai solusi yang ditemukan kemudian. Cara lainnya jika solusi calon yang baru adalah yang lebih jelek solusi itu diterima dengan suatu kemungkinan penerimaan.

3.4. Formulasi Masalah *Optimal Power Flow*^[10]

Masalah *optimal power flow* berupaya mengoptimalkan performa sistem pembangkitan daya dengan fungsi objektif f dan subyeknya berbagai batasan. Untuk pembagian optimal daya aktif dan reaktif, fungsi obyektif, f , adalah biaya total pembangkit. Tujuan lainnya yaitu mencakup meminimalan kerugian transmisi dan pengoptimalan tingkat tegangan. Secara matematis dapat dirumuskan sebagai berikut :

$$\text{Minimize } F = \sum_{i=1}^{NG} Fi(P_{Gi}) \dots\dots\dots(7)$$

Subjeknya pada batas keseimbangan daya,

$$P_{Gi} - P_{Di} = \sum_{j=1}^N |V_i||V_j||Y_{ij}| \cos(\theta_{ij} - \delta_i + \delta_j) = 0 \quad i=1, \dots, N \dots\dots\dots(8)$$

$$Q_{Gi} - Q_{Di} = - \sum_{j=1}^N |V_i||V_j||Y_{ij}| \sin(\theta_{ij} - \delta_i + \delta_j) = 0 \quad i=1, \dots, N \dots\dots\dots(9)$$

Dan ketidaksamaan yang dikenakan pada operasi dari sistem daya termasuk pembangkitan daya nyata dan daya reaktif,

$$P_{Gi}^{\min} \leq P_{Gi} \leq P_{Gi}^{\max} \quad i=1, \dots, NG \dots\dots\dots(10)$$

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max} \quad i=1, \dots, NG \dots\dots\dots(11)$$

batas tegangan,

$$|V_i^{\min}| \leq |V_i| \leq |V_i^{\max}| \quad i= 1, 2, \dots, N \dots\dots\dots(12)$$

batas keandalan tap trafo

$$t_i^{\min} \leq t_i \leq t_i^{\max} \quad i= 1, 2, \dots, NT \dots\dots\dots(13)$$

dan Saluran atau keandalan beban trafo,

$$|S_{Li}| \leq S_{Li}^{\max} \quad i= 1, 2, \dots, NT \dots \dots \dots (14)$$

dimana,

F = total biaya bahan bakar pembangkitan (\$/hr);

F_i = biaya bahan bakar pembangkitan pada bus i (\$/hr);

P_{Gi} = total daya nyata yang dibangkitkan pada bus ke i (MW);

P_{Di} = total daya nyata yang diminta pada bus ke i (MW);

Q_{Gi} = total pembangkitan daya reaktif pada bus i (MVar);

Q_{Di} = total permintaan daya reaktif pada bus i (MVar);

$|V_i|$ = magnitudo tegangan pada bus i (kV);

$|V_j|$ = magnitudo tegangan pada bus j (kV);

$|Y_{ij}|$ = magnitudo dari element ij th Y_{bus} ;

θ_{ij} = sudut ij th element Y_{bus} ;

δ_i = sudut tegangan pada bus i ;

δ_j = sudut tegangan pada bus j ;

P_{Gi}^{\max} = pembangkitan max daya nyata pada bus i (MW);

P_{Gi}^{\min} = pembangkitan min daya nyata pada bus i (MW);

Q_{Gi}^{\max} = pembangkitan max daya reaktif pada bus i (MVar);

Q_{Gi}^{\min} = pembangkitan min daya reaktif pada bus i (MVar);

$|V_i^{\max}|$ = magnitudo tegangan max pada bus i (kV);

$|V_i^{\min}|$ = magnitudo tegangan min pada bus i (kV);

t_i^{\max} = batas max tap seting trafo i ;

t_i^{\min} = batas min tap seting trafo i;

t_i = tap seting trafo i;

$|S_{Li}|$ = line or trafo beban pada line i (MVA);

S_{Li}^{\max} = line or trafo beban pada line i (MVA);

N = total nomer dari bus;

NG = total nomer dari pembangkitan;

NT = total nomer drai trafo;

NL = total nomer saluran;

Untuk menangani batasan ketidaksaman dari suatu variable yang mencakup besarnya tegangan bus beban dan variabel keluaran termasuk pembangkitan daya nyata pada slack bus, output pembangkitan daya reaktif, dan beban saluran, penambahan fungsi obyektif bisa dituliskan seperti dibawah ini;

$$F = \sum_{i=1}^{NG} F_i(P_{Gi}) + K_p h(P_{Gi}) + K_q \sum_{i=1}^{NG} h(Q_{Gi}) + K_v \sum_{i=1}^{NLB} h(|V_{Li}|) + K_s \sum_{i=1}^{NL} h(|S_{Li}|) \dots (15)$$

Dimana K_p , K_q , K_v , dan K_s adalah penalty konstan dari pembangkitan daya nyata pada slack bus, pembangkitan daya reaktif dari semua bus PV dan slack bus, secara berturut-turut. $h(P_{Gi})$, $h(Q_{Gi})$, $h(|V_{Li}|)$, and $h(|S_{Li}|)$ merupakan fungsi penalty dari pembangkitan daya nyata pada slack bus, besarnya tegangan pada semua bus PQ secara berturut-turut. NLB adalah nomer dari PQ bus. Fungsi penalty bisa di tulis seperti di bawah ini;

$$h = \begin{cases} (x - x^{\max})^2 & \text{jika } x > x^{\max} \\ (x^{\min} - x)^2 & \text{jika } x < x^{\min} \\ 0 & \text{jika } x^{\min} \leq x \leq x^{\max} \end{cases}$$

dimana $h(x)$ fungsi penalty dari variabel x , x^{\max} , dan x^{\min} limit atas dan limit bawah dari suatu variable x .

3.5. Improved Evolutionary Programming Untuk Masalah Optimal Power Flow^[10]

Improved Evolutionary Programming adalah merupakan bentuk *Evolutionary Programming* dan *Simulated Annealing* (SA) algorithm. IEP awalnya membagi solusi tempat menjadi beberapa subtempat. Berbagai macam operator mutasi telah dipakai untuk memperbesar penyelusuran keanekaragaman. Mekanisme seleksi dengan meng-*update* berdasarkan pada *schedule annealing* dari SA untuk digunakan dalam menghindari *entrapment* pada *local optimum*. *Reassignment strategy* dari setiap individu di desain untuk setiap populasi agar bias saling bertukar informasi.

3.5.1. Representasi Solusi^{[6] [7] [9]}

Individu dalam populasi menunjukkan calon pada solusi OPF. Tiap individu terdiri dari variable control OPF termasuk pembangkitan daya nyata yang ditetapkan pada semua bus PV dan tidak termasuk bus slack, *magnitude* tegangan untuk semua bus PV termasuk bus slack

3.5.2. Inisialisasi^{[6] [7] [9] [10]}

Memastikan semua individu dalam populasi yang utuh ditemakan secara keseluruhan pada space slisi yang berbeda, jika total populasi adalah dibagi kedalam M subpopulasi maka tiap populasi yang dipilih secara acak dalam

disesuaikan dengan solusi subspace divisi. Terdiri dari SP individual dan tiap element dari individu yang ditulis seperti persamaan di bawah ini:

$$x_i = x_i^{\min} + u.(x_i^{\max} - x_i^{\min}) \dots\dots\dots(16)$$

dimana

- x_i adalah i th element dari individual dalam suatu subpopulation
- x_i^{\min} subinterval atau interval batas bawah dari i th elemen individual
- x_i^{\max} subinterval atau interval batas atas dari i th elemen individual
- u uniform random nomer pada interval [0,1].

3.5.3. Inisialisasi Temperature dan Cooling Procedure^[10]

Inisialisasi *temperature* dari tiap subpopulasi telah ditentukan oleh nilai obyektif yang berbeda diantara individu yang terbaik dan yang paling buruk dalam suatu populasi, dan kemungkinan dari penerimaan individu yang buruk kedalam suatu individu yang terbaik. Ini terlihat pada persamaan dibawah ini:

$$T_{o,m} = - (F_{max,m} - F_{min,m}) / \ln p_r \dots\dots\dots(17)$$

Dimana

$T_{o,m}$ nital temperature dari m th subpopulation.

$F_{max,m}$ nilai obyektif dari individu yang paling buruk dalam m th subpopulation.

$F_{min,m}$ nilai obyektif dari individu yang terbaik dalam m th subpopulation.

p_r kemungkinan individu yang paling buruk diterima oleh individu yang terbaik.

Setelah *strategy reassignment*, maka *temperature* diturunkan oleh fungsi *temperature annealing* dimana dituliskan sebagai berikut:

$$T_{0,m} = \lambda^{(r-1)} \cdot T_{0,m} / \ln p_r \dots\dots\dots(18)$$

dimana

$T_{r,m}$ temperature mth *subpopulation* setelah *r*th *reassignment*.

$T_{0,m}$ inisial temperature dari mth *subpopulation*

λ rata-rata penurunan

p_r nomer dari *reassignment*

Salah satu sifat dari SA ini merupakan Optimasi dasar untuk mencari suatu simulasi dari proses fisik pendinginan logam dengan pengotoran dimana suatu partikel cair didinginkan pelan – pelan dari temperature yang sangat tinggi sampai mengeras pada temperature rendah. Jumlah iterasi di dalam SA adalah menganalisa untuk tingkat *temperature*. Sejumlah percobaan dilakukan dalam suatu iteasi. Pada setiap percobaan suatu solusi calon dihasilkan. Manakala solusi calon yang baru mempunyai biaya yang lebih rendah dibanding solusi yang sekarang akan diterima sebagai solusi yang ditemukan kemudian. Cara lainnya jika solusi calon yang baru adalah yang lebih jelek solusi itu diterima dengan suatu kemungkinan penerimaan.

3.5.4. Mutasi^{[6] [7] [9] [10]}

Setiap subpopulasi, mutasi digunakan untuk membentuk *offspring* baru pada setiap subpopulasi. Pada subpopulasi *offspring* dengan individual

dibangkitkan dengan menggunakan operator mutasi pada subpopulasi *parent*. Dua operator mutasi termasuk *Gaussian* dan *Cauchy* digunakan untuk memperbesar perbedaan didalam suatu subpopulasi yang berbeda-beda.

Tiap-tiap elemen dari suatu individu *offspring* ditulis seperti persamaan di bawah ini:

$$x'_i = x_i + \xi_m \cdot \sigma_i \dots\dots\dots(19)$$

$$\sigma_i = a^{(r-1)} \cdot (x_i^{\max} - x_i^{\min}) \dots\dots\dots(20)$$

dimana

x'_i nilai dari *i*th elemen individu *offspring*

x_i nilai dari *i*th elemen individu *parent*

σ_i ukuran langkah mutasi untuk setiap *i*th elemen dari individu

ξ_m gangguan dari *m*th subpopulasi seperti $N(0,1)$, $C(0,1)$, dll.

a nomer positif *constant slightly* kurang dari satu

r nomer *reassignment*

x_i^{\max} subinterval atau interval dari batas atas dari *i*th elemen suatu individu

x_i^{\min} subinterval atau interval dari batas bawah dari *i*th elemen suatu individu

$N(0,1)$ nomer acak *Gaussian* dengan 0 sebagai penyimpangan kesatuan unit

$C(0,1)$ nomer acak *Cauchy*.

Jika nilai yang dimutasi melebihi batas yang diinginkan dari *space division* maka akan diperbaharui kembali seperti pada persamaan (19) sampai batas yang ditentukan

3.5.5. Seleksi^[10]

Untuk menghindari *entrapping* saat *optimum local*, IEP menggunakan strategy SA untuk pemilihan populasi *parent* pada generasi berikutnya. Tiap individual *offspring* diseleksi untuk menghasilkan individu *parent* baru berdasarkan pada peningkatan nilai *objective parent* dan *temperature annealing*. Saat pencarian dimulai, *temperature* yang tinggi mengijinkan beberapa individual *offspring* yang lebih rendah untuk bertahan dengan kemungkinan penerimaan yang tinggi, yaitu dengan cara diversifikasi. Saat iterasi ditingkatkan, kemudian *temperature* diturunkan dalam rangka mencari individu *parent* diantara *neighbourhood offspring* yang lebih rendah. Kemungkinan diterimanya individu *offspring* tersebut dalam *kth* dan *mth* sebagai subpopulasi bisa ditulis sbb:

$$p_{k,m} = \min \{1, \exp(-\Delta / T_{r,m})\} \dots \dots \dots (21)$$

dimana

$p_{k,m}$ penerimaan kemungkinan dari *kth* individu *offspring* didalam *mth* subpopulasi;

Δ nilai objective yang berbeda diantara *kth* individu *offspring* dan bersesuaian dengan individu *parent*

$T_{r,m}$ *temperature annealing* dari *mth* subpopulasi setelah *rth* ditugaskan kembali.

Jika $p_{k,m}=1$, k th individu *offspring* dari m th subpopulasi akan diseleksi menjadi individu *parent* baru untuk generasi berikutnya. Selain itu, nomer acak yang seragam, U , dalam interval $[0,1]$ dihasilkan dan dibandingkan dengan $p_{k,m}$. Jika $p_{k,m} > U$, maka individu *offspring* dalam k th bisa diterima. Maka dengan begitu *parent* yang bersesuaian bisa terpilih.

Sebagai tambahan, jaminan yang dicapai oleh individu yang terbaik dalam tiap subpopulasi bisa diterapkan. Jika individu terbaik telah dicapai dalam tiap subpopulasi tidak termasuk dalam subpopulasi *parent* yang baru, maka dimana individu dengan nilai tertinggi akan digantikan oleh individu yang terbaik.

3.5.6. *Reassignment Strategy*^[10]

Setelah evolusi dari semua subpopulasi dengan pembangkitan G_{max} , *Reassignment strategy* untuk semua individu dibawa keluar untuk memadukan dan menukar informasi pencarian dari semua subpopulasi sedemikian hingga *convergence* prematur yang disebabkan oleh solusi space kecil dalam mencari populasi tunggal bisa dihindarkan. *Reassignment strategy* individu dalam suatu subpopulasi ditugaskan kembali untuk memperbesar keanekaragaman dari suatu subpopulasi dan menyediakan beberapa individu dengan mutasi lainnya untuk melanjutkan evolusi baru. Operator mutasi yang berbeda digunakan dalam pengertian *hybrid* untuk individu tertentu, menuju ke arah explorasi yang lebih kuat.

Untuk melaksanakan *Reassignment strategy* setiap individu dari semua subpopulasi telah digabungkan. Kemudian keseluruhan populasi akan diacak dan dibagi ke dalam bentuk baru yaitu *M subpopulation*.

3.5.7. Criteria Termination^[10]

Setelah itersi G_{max} pada tiap populasi, keseluruhan populasi telah digabungkan dan ditugaskan kembali. IEP berakhir saat nomer *Reassignment* melebihi nilai (R_{max}) yang dibutuhkan.

BAB IV

HASIL DAN ANALISA HASIL

4.1. Program Komputer Optimal Power Flow Menggunakan Metode *Improved Evolutionary Programming*

Dalam penyelesaian masalah ini diperlukan bantuan program komputer dalam perhitungan yang membutuhkan ketelitian dan keakuratan. Program komputer dalam skripsi ini di jalankan dengan menggunakan bahasa Pemrograman Bordland Delphi versi 7.0 dan merupakan program yang mudah diaplikasikan .

4.1.1. Algoritma Program

Algoritma program optimasi dilakukan dengan langkah-langkah sebagai berikut:

1. Memasukkan Inputan data beban yang meliputi tegangan (V), sudut fasa tegangan (δ), daya aktif (P), daya reaktif (Q) dan data impedansi saluran, data pembangkit Pmak, Pmin, konstanta biaya.
2. Lakukan proses *load flow Newton Raphson*.
3. Lakukan proses *Improved Evolutionary Programming*.
4. Apakah objective function sudah terpenuhi?
5. Jika “tidak” maka melakukan perhitungan *load flow* dan *IEP* lagi.
6. Jika “ya” maka tampilkan hasil optimasi dan perhitungan berhenti.

Sedangkan untuk Algoritma aliran daya dengan metode *Newton Raphson* adalah sebagai berikut:

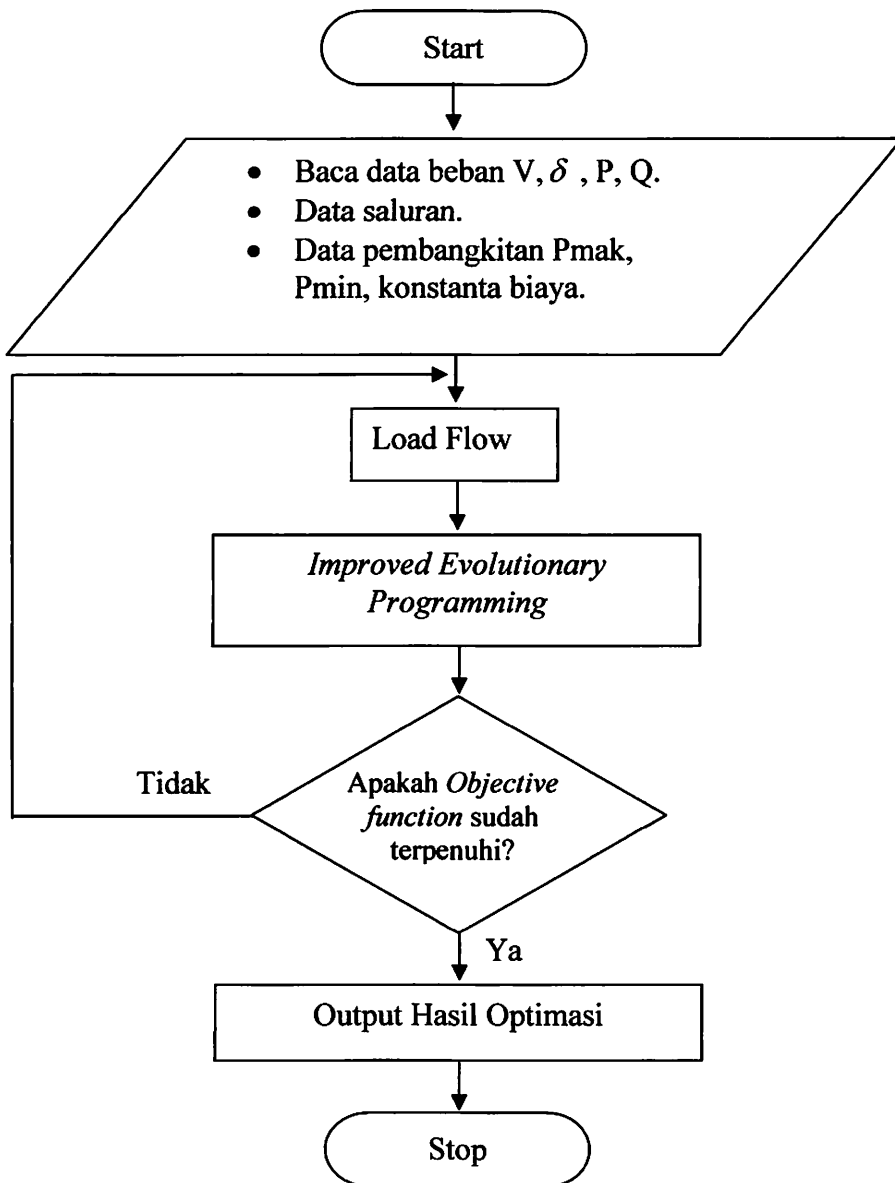
1. Bentuk matriks admitansi bus : [Y_{bus}]

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 di jadwalkan dengan injeksi daya bus dari perhitungan.
6. Menentukan perubahan maksimum pada daya aktif dan 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 element matriks Jacobian.
10. Menghitung faktor koreksi tegangan dan sudut fasa setiap bus kecuali bus slack dan bus generator.
11. Menghitung nilai sudut fasa yang lama dengan sudut fasa yang baru, tegangan yang lama dengan tegangan yang baru.
12. Mengganti nilai sudut fasa yang baru, tegangan yang lama dengan tegangan yang baru.
13. Perhitungan di lanjutkan ke langkah 4 dengan nilai iterasi yang baru sampai hasil yang di dapatkan konvergen.

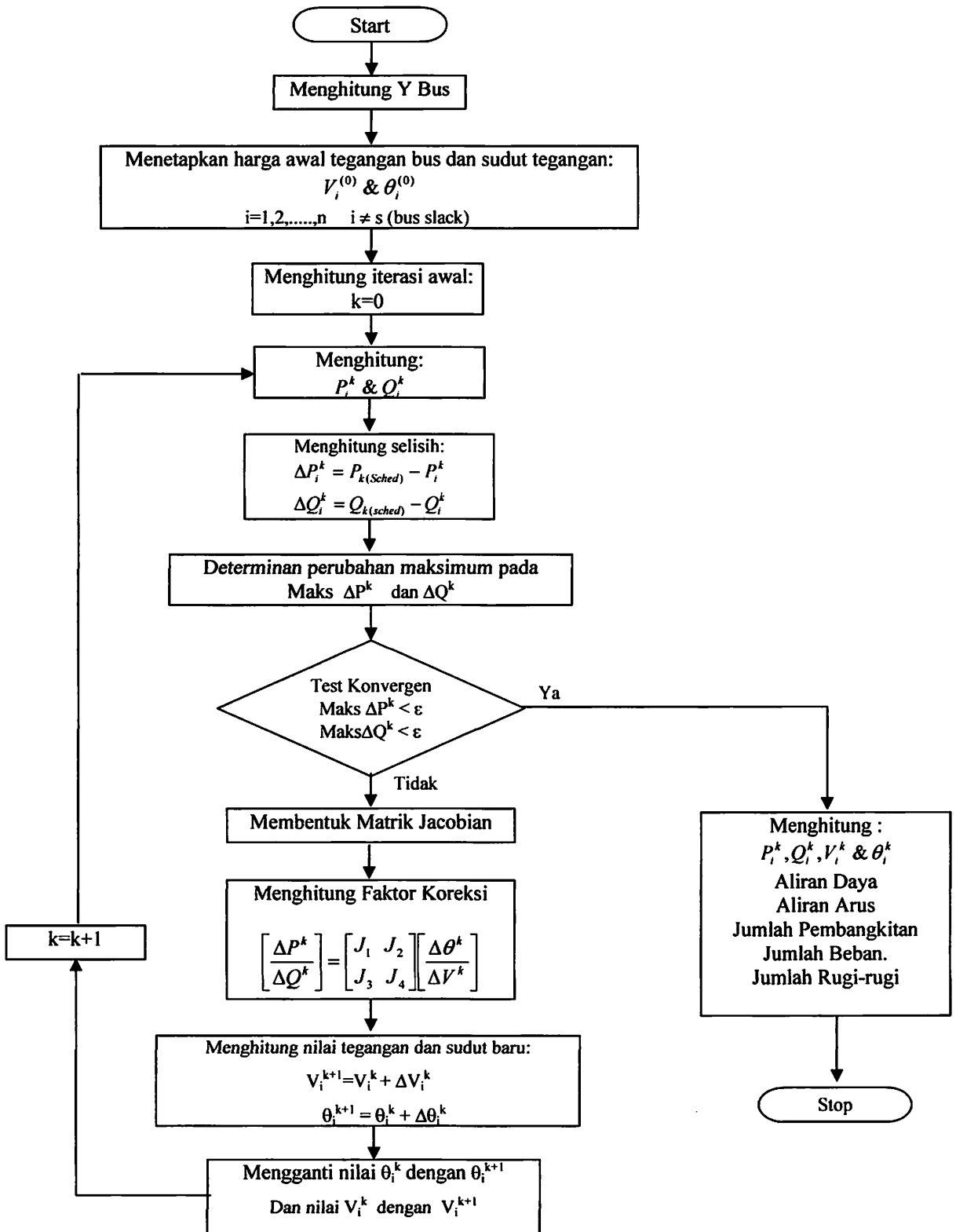
Selanjutnya untuk Algoritma program OPF menggunakan metode IEP dilakukan dengan langkah-langkah sebagai berikut:

1. Memasukan inputan data beban yang meliputi tegangan (V), sudut fasa tegangan (δ), daya aktif (P), daya reaktif (Q), dan data impedansi saluran, data pembangkitan P_{maks} , P_{min} , konstanta biaya.
2. Menentukan parameter inputan *evolutionary programming* yang meliputi jumlah populasi, maksimum generasi, nilai kemungkinan mutasi, dan panjang kromosom tiap-tiap individu.
3. Generasi = 0, Populasi = 0.
4. Melakukan inialisasi calon solusi space division pada I^{st} dan M^{th} subpopulasi
5. Melakukan inialisasi temperature pada I^{st} dan M^{th} subpopulasi
6. Set Gen = 1 pada I^{st} dan M^{th} subpopulasi.
7. Melakukan proses mutasi pada I^{st} dan M^{th} subpopulasi.
8. melakukan seleksi pada I^{st} dan M^{th} subpopulasi.
9. Apakah generasi yang diinginkan sudah terpenuhi (max Gen).
10. Jika “ Tidak “ maka generasi = gen + 1, kembali ke langkah 7
11. Jika “ Ya “ maka perhitungan dilanjutkan ke langkah berikutnya.
12. Melakukan proses *reassignment strategy*
13. Apakah Reassignment yang diinginkan sudah terpenuhi (max R).
14. Jika “ Ya “ maka perhitungan berhenti
15. Jika “ Tidak “ maka $r = r + 1$, dilanjutkan ke langkah berikutnya
16. Melakukan Cooling Temperature
17. Jika menghasilkan individu baru pada I^{st} dan M^{th} subpopulasi proses berlanjut ke langkah 8.

4.1.2. Flowchart Program.

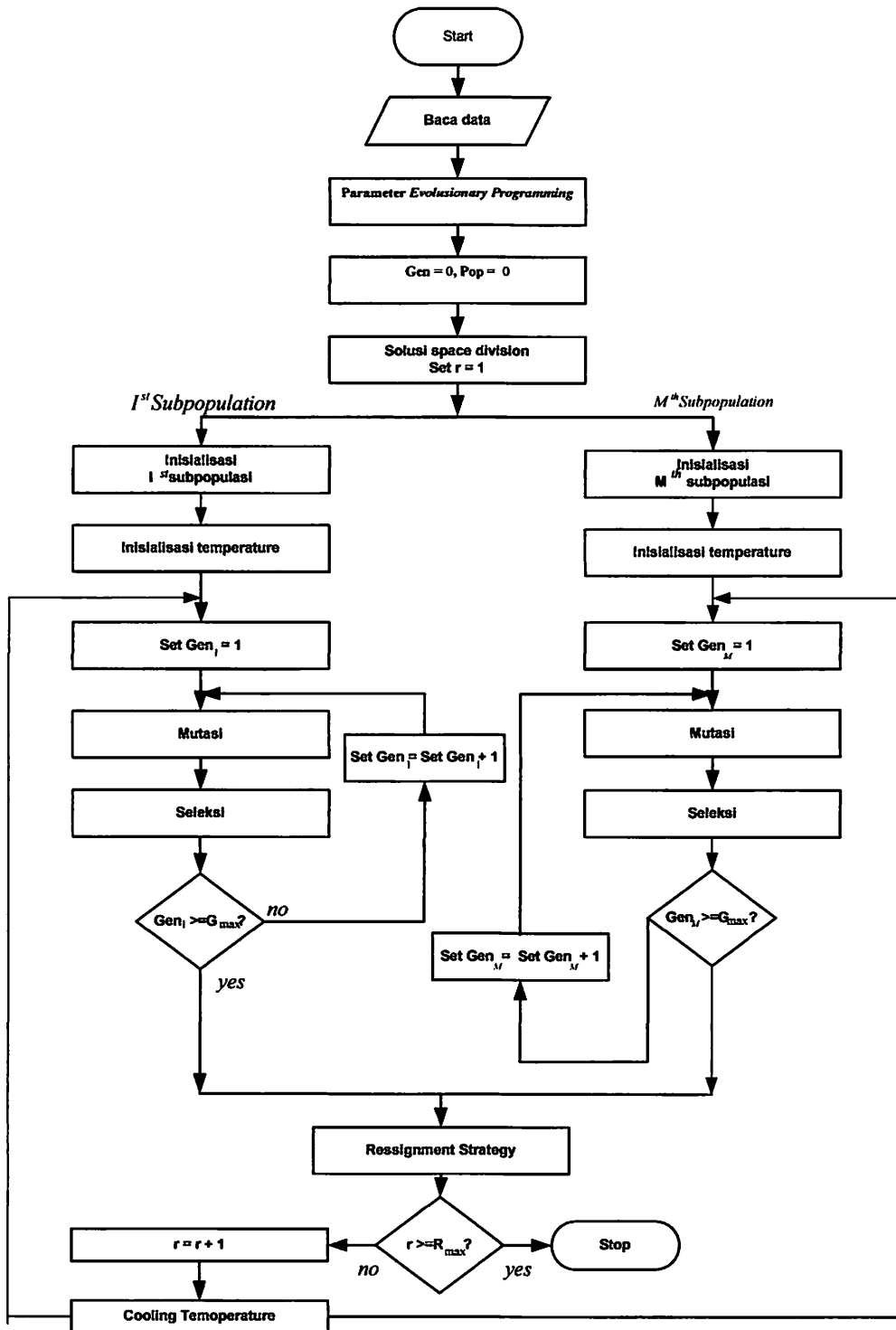


Gambar 4.1
Flowchart Perhitungan Program Optimasi



Gambar 4.2

Flow Chart Perhitungan Aliran Daya dengan Metode Newton Raphson



Gambar 4.3
Flowchart Perhitungan Aliran Daya Metode Improved Evolutionary Programming

4.2. Validasi Program Menggunakan Data Sistem IEEE 30-Bus

4.2.1. Data Sistem IEEE 30-Bus

Tabel 4.1
Data Generator dan Konstanta Biaya IEEE 30-Bus

No	No Bus	P_G^{Min} MW	P_G^{Max} MW	Q_G^{Min} MVAR	Q_G^{Max} MVAR	Konstanta Biaya		
						A	B	C
1	1	50	200	-20	150.0	0.00375	2.0000	0.0000
2	2	20	80	-20	60.0	0.01750	1.7500	0.0000
3	5	15	50	-15	62.5	0.01750	1.0000	0.0000
4	8	10	35	-15	50.0	0.00834	3.2500	0.0000
5	11	10	30	-10	40.0	0.02500	3.0000	0.0000
6	13	12	40	-15	45.0	0.02500	3.0000	0.0000

Tabel 4.2
Data Bus IEEE 30 Bus

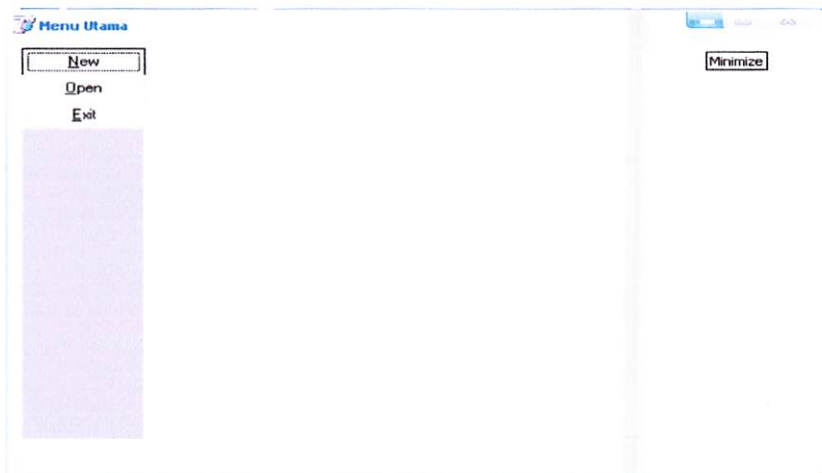
No Bus	Tegangan (pu)	P_{Gen} (MW)	Q_{Gen} (MVAR)	P_{Load} (MW)	Q_{Load} (MVAR)	Type Bus
1	1.05	0.00	0.00	0.00	0.00	0.00
2	1.0338	0.00	57.56	0.00	21.70	12.70
3	1.00	0.00	0.00	0.00	2.400	1.20
4	1.00	0.00	0.00	0.00	7.600	1.60
5	1.0058	0.00	24.56	0.00	94.20	19.00
6	1.00	0.00	0.00	0.00	0.00	0.00
7	1.00	0.00	0.00	0.00	22.80	10.90
8	1.0230	0.00	35.00	0.00	30.00	30.00
9	1.00	0.00	0.00	0.00	0.00	0.00
10	1.00	0.00	0.00	0.00	5.80	2.00
11	1.0913	0.00	17.93	0.00	0.00	0.00
12	1.00	0.00	0.00	0.00	11.20	7.50
13	1.0883	0.00	16.91	0.00	0.00	0.00
14	1.00	0.00	0.00	0.00	6.20	1.60
15	1.00	0.00	0.00	0.00	8.20	2.50
16	1.00	0.00	0.00	0.00	3.50	1.80
17	1.00	0.00	0.00	0.00	9.00	5.80
18	1.00	0.00	0.00	0.00	3.20	0.90
19	1.00	0.00	0.00	0.00	9.50	3.40
20	1.00	0.00	0.00	0.00	2.20	0.70
21	1.00	0.00	0.00	0.00	17.50	11.20
22	1.00	0.00	0.00	0.00	0.00	0.00
23	1.00	0.00	0.00	0.00	3.200	1.60
24	1.00	0.00	0.00	0.00	8.700	6.70
25	1.00	0.00	0.00	0.00	0.00	0.00
26	1.00	0.00	0.00	0.00	2.30	2.30
27	1.00	0.00	0.00	0.00	0.000	0.00
28	1.00	0.00	0.00	0.00	0.000	0.00
29	1.00	0.00	0.00	0.00	0.400	0.90
30	1.00	0.00	0.00	0.00	10.60	1.90

Tabel 4.3
Data Saluran IEEE 30-Bus

No	Dari Bus	Ke Bus	R(pu)	X(pu)	B(pu)
1	1	2	0.0192	0.0575	0.0264
2	1	3	0.0452	0.1852	0.0204
3	2	4	0.0570	0.1737	0.0184
4	3	4	0.0132	0.0379	0.0042
5	2	5	0.0472	0.1983	0.0209
6	2	6	0.0581	0.1763	0.0187
7	4	6	0.0119	0.0414	0.0045
8	5	7	0.0460	0.0102	0.0102
9	6	7	0.0267	0.0820	0.0085
10	6	8	0.0120	0.0420	0.0045
11	6	9	0.0000	0.2080	0.0000
12	6	10	0.0000	0.5560	0.0000
13	9	11	0.0000	0.2080	0.0000
14	9	10	0.0000	0.1100	0.0000
15	4	12	0.0000	0.2560	0.0000
16	12	13	0.0000	0.1400	0.0000
17	12	14	0.01231	0.2559	0.0000
18	12	15	0.0602	0.1304	0.0000
19	12	16	0.0945	0.1987	0.0000
20	14	15	0.2210	0.1997	0.0000
21	16	17	0.0824	0.1932	0.0000
22	15	18	0.1070	0.2185	0.0000
23	18	19	0.0639	0.1292	0.0000
24	19	20	0.0340	0.0680	0.0000
25	10	20	0.0936	0.2090	0.0000
26	10	17	0.0324	0.0845	0.0000
27	10	21	0.0348	0.0749	0.0000
28	10	22	0.0727	0.1499	0.0000
29	21	22	0.0116	0.0236	0.0000
30	15	23	0.1000	0.2020	0.0000
31	22	24	0.1150	0.1790	0.0000
32	23	24	0.1320	0.2700	0.0000
33	24	25	0.1885	0.3292	0.0000
34	25	26	0.2544	0.3800	0.0000
35	24	27	0.1093	0.2087	0.0000
36	28	27	0.0000	0.3960	0.0000
37	27	29	0.2198	0.4153	0.0000
38	27	30	0.3202	0.6027	0.0000
39	29	30	0.2399	0.4533	0.0000
40	8	28	0.0636	0.2000	0.0214
41	6	28	0.0169	0.0599	0.0065

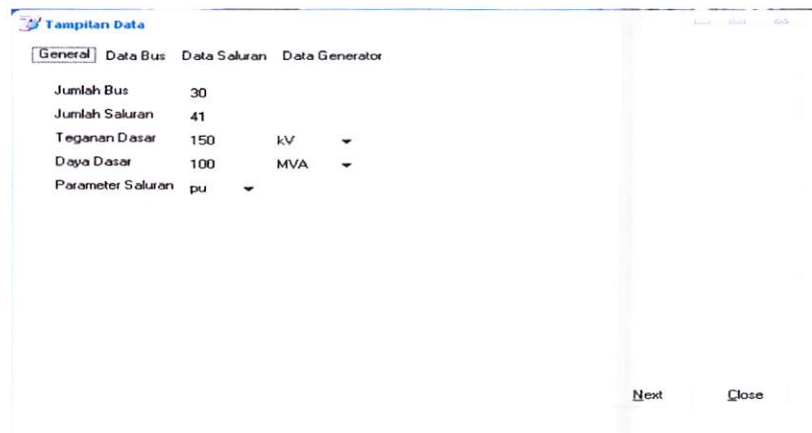
4.2.2 Hasil Perhitungan Validasi IEEE 30 Dengan Menggunakan Metode *Improved Evolutionary Programming*

Tampilan utama dari program computer ini dapat dilihat pada gambar dibawah ini:

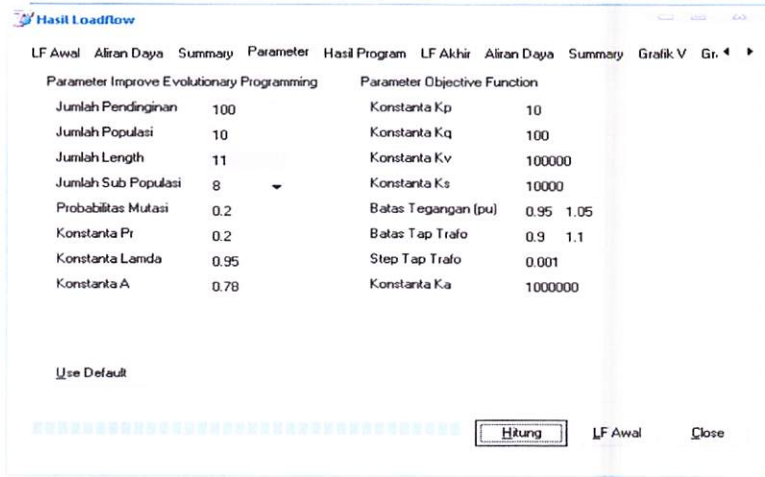


Gambar 4.4
(a) Tampilan program

Kemudian setelah itu tekan tombol open untuk membuka file yang sudah tersimpan.



Gambar 4.4
(b) Tampilan input data IEEE-30



Gambar 4.5
Tampilan Parameter IEP

Setelah menetapkan parameter tersebut, maka komputasi dapat dilakukan

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.04954	0.00000	174.911	-14.876	0.000	0.000	0.000	1
2	1.03580	-0.06648	22.863	5.026	27.700	12.700	0.000	2
3	1.02789	-0.09061	0.000	0.000	2.400	1.200	0.000	3
4	1.02231	-0.10870	0.000	0.000	7.600	1.600	0.000	3
5	1.04710	-0.17292	32.500	61.060	94.200	19.000	0.000	2
6	1.02213	-0.12471	0.000	0.000	0.000	0.000	0.000	3
7	1.02477	-0.15325	0.000	0.000	22.800	10.900	0.000	3
8	1.02636	-0.12627	33.129	40.887	30.000	30.000	0.000	2
9	1.01353	-0.15005	0.000	0.000	0.000	0.000	0.000	3
10	0.99193	-0.18848	0.000	0.000	5.800	2.000	0.000	3
11	1.04850	-0.10597	22.513	18.128	0.000	0.000	0.000	2
12	1.00557	-0.18231	0.000	0.000	11.200	7.500	0.000	3
13	1.02875	-0.16607	12.000	17.130	0.000	0.000	0.000	2
14	0.98983	-0.19872	0.000	0.000	6.200	1.600	0.000	3
15	0.98521	-0.19963	0.000	0.000	8.200	2.500	0.000	3

(a) Hasil load flow

Hasil Loadflow

LF Awal Aliran Daya Summary Parameter Hasil Program LF Akhir Aliran Daya Summary Grafik V Gr. ◀ ▶

Hasil Setting Program			Rekapitulasi Biaya		
No	Bus	Pgen			
1	Pg2	22.863	Sebelum Optimasi	634	Rp
2	Pg5	32.500	Sesudah Optimasi	634	Rp
3	Pg8	33.129	Selisih Optimasi	0	Rp
4	Pg11	22.513	Rekapitulasi Losses		
5	Pg13	12.000	Sebelum Optimasi	7.520	MW
6	Vg1	1.050	Sesudah Optimasi	8.517	MW
7	Vg2	1.036	Selisih Optimasi	-0.997	MW
8	Vg5	1.047			
9	Vg8	1.026			
10	Vg11	1.049			
11	Vg13	1.029			

Hitung LF Awal Close

(b) Hasil perhitungan biaya pembangkitan

Tabel
4.4

Perbandingan Hasil Data Referensi Jurnal dengan Data Optimasi

No Unit	No Bus	Hasil Referensi			Hasil program		
		V _G (pu)	P _G (MW)	Biaya (\$/h)	V _G (pu)	P _G (MW)	Biaya (\$/h)
1	1	1.05000	144.961	250.835	1.04984	173.911	311.735
2	2	1.03380	57.560	164.81	1.04358	22.863	58.17
3	5	1.00580	24.560	63.379	1.04988	32.500	80.259
4	8	1.02300	35.000	98.746	1.02052	33.129	87.966
5	11	1.09130	17.930	28.847	1.03357	22.513	52.827
6	13	1.08830	16.910	27.858	1.04946	12.000	22.878
TOTAL			296.921	634.475		296.916	634.465

Setelah Dilakukan Proses Optimasi maka di peroleh biaya pembangkitan dari data IEEE 30 yaitu sebesar 634.475\$/h sedangkan biaya total hasil program sebesar 634.465\$/h.

Sehingga dari proses validasi diatas didapatkan error / kesalahan perhitungan untuk keluaran daya aktif sebesar 0.001687%, sedangkan untuk biaya pembangkitan sebesar 0.00157 %.

4.3 Data Pembangkitan Thermal Pada Sub Sistem Paiton dan Bali

Pada skripsi ini akan membahas pembangkit thermal yang berada pada sub sistem Paiton dan Bali. Pembangkit thermal yang dibahas dalam skripsi ini adalah PLTU Paiton 1-2, 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 4.5
Parameter Unit Pembangkit Thermal

Nama Pembangkit	a2	a1	a0	Pmin (MW)	Pmax (MW)
PLTU Paiton 1&2	6.180000	1306.150	388144.168	150	700
PLTG Gilimanuk	1.406900	1599.000	87435.000	50	133.8
PLTD Pesanggaran	14.293200	1670.000	88960.000	21	75
PLTG Pesanggaran	113.900000	5297.000	139560.000	15	125.5

Dengan memasukkan data-data pada tabel 4.5. ke persamaan fungsi biaya bahan bakar, maka untuk unit pembangkit thermal PLTU Paiton 1&2 diperoleh sebagai berikut :

Biaya bahan bakar pembangkit:

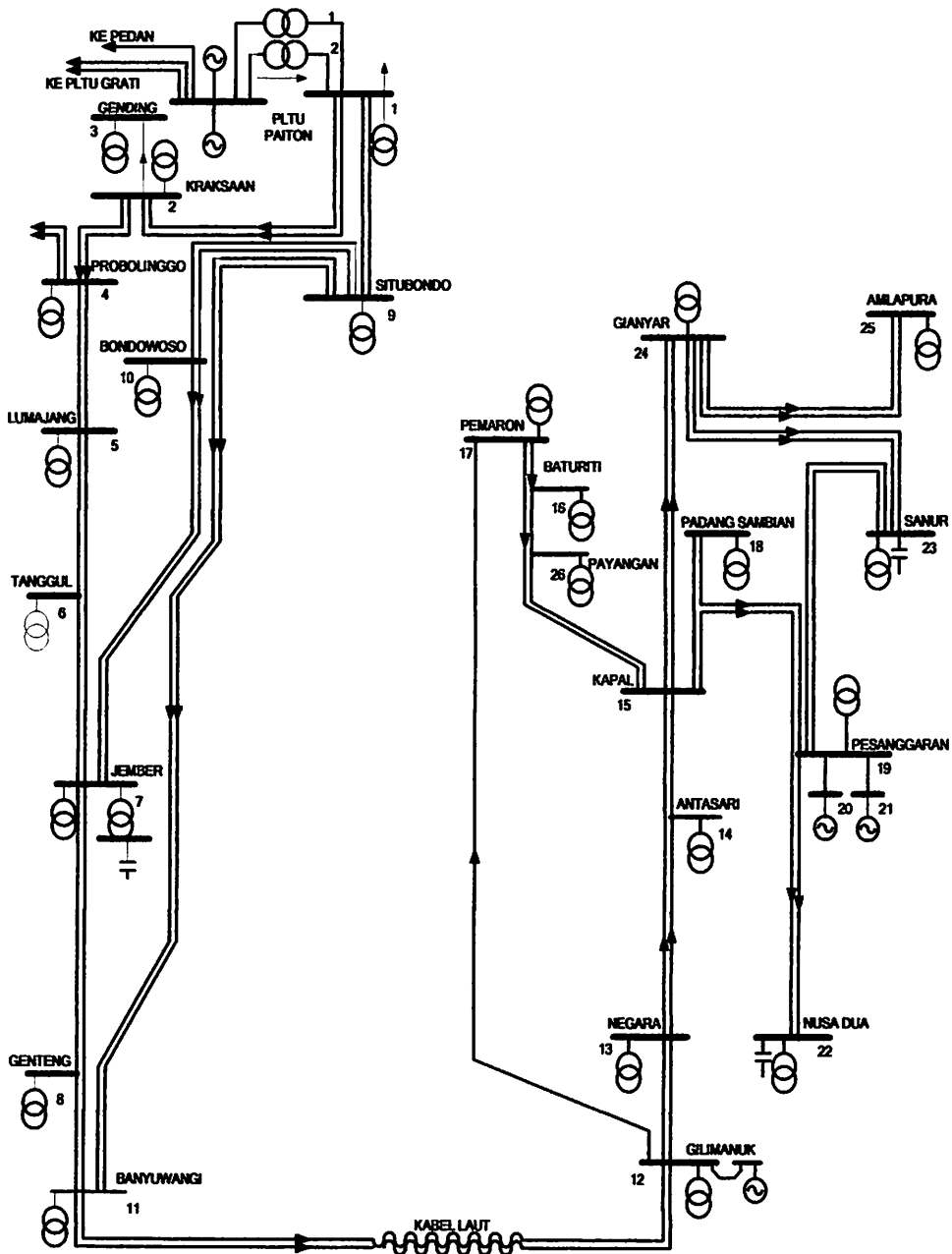
$$F_i(G_i) = a_2 P_{Gi}^2 + a_1 P_{Gi} + a_0$$

$$F_1(G_1) = 6.180000 P^2 + 1306.150 P + 388144.168$$

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

Tabel 4.6
 Persamaan Biaya Pembangkitan
 Unit Pembangkit Thermal Paiton dan Bali

No.	Nama Pembangkit	Persamaan Biaya Bahan Bakar (Rp/Jam)
1	PLTU Paiton 1&2	$F = 6.180000 P^2 + 1306.150 P + 388144.168$
2	PLTG Gilimanuk	$F = 1.406900 P^2 + 1599.000 P + 87435.000$
3	PLTDPesanggaran	$F = 14.293200 P^2 + 1670.000 P + 88960.000$
4	PLTGPesanggaran	$F = 113.900000 P^2 + 5297.000 P + 139560.000$



Gambar 4.6
 Diagram Segaris Jaringan Sistem Tenaga Listrik Sub Sistem
 150 kV Paiton-Bali

Tabel 4.7
 Penomoran Bus 150 kV Sub Sistem Paiton dan 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
26	PAYANGAN

4.4 Data Pembangkitan dan Pembebanan 150 Kv Sub Sistem Paiton dan Bali

Berdasarkan data referensi [11] maka dilakukan pengolahan data untuk mengubah dasar tegangan ke dalam pu. Serta perhitungan P_{load} dan Q_{load} .

Pada bus nomor 1, maka :

$$\text{Tegangan Dasar} = 150 \text{ Kv}$$

$$\text{Tegangan Sebenarnya} = 155 \text{ Kv}$$

$$\text{Tegangan (pu)} = \frac{\text{Tegangan sebenarnya}}{\text{Tegangan dasar}}$$

$$\text{Tegangan (pu)} = \frac{155}{150} = 1.033 \text{ pu}$$

$$P_{\text{beban}} = 11.7 \text{ MW}$$

$$Q_{\text{beban}} = 5.7 \text{ MVAR}$$

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

Tabel 4.8.^[11]
Data Pembangkitan dan Pembebanan
150 Kv Sub Sistem Paiton dan Bali
Hari Rabu, Tanggal 13 juni 2006, Pukul : 18:00 WIB

No	Nama Bus	Tegangan	Pgenerator	Qgenerator	Pload	Qload	Tipe
		(pu)	(MW)	(MVAR)	(MW)	(MVAR)	
1	PAITON	1.03066	0.000	0.000	14.500	6.000	1
2	KRAKSAAN	1.01333	0.000	0.000	5.300	6.000	3
3	GENDING	1.01400	0.000	0.000	19.700	8.700	3
4	PROBOLINGGO	1.00000	0.000	0.000	39.700	17.200	3
5	LUMAJANG	0.98666	0.000	0.000	39.700	15.600	3
6	TANGGUL	1.00000	0.000	0.000	27.200	9.400	3
7	JEMBER	1.00000	0.000	0.000	39.800	11.700	3
8	GENTENG	0.97333	0.000	0.000	62.200	30.000	3
9	SITUBONDO	1.00667	0.000	0.000	24.900	9.600	3
10	BONDOWOSO	0.99333	0.000	0.000	19.700	7.000	3
11	BANYUWANGI	0.98667	0.000	0.000	38.100	14.000	3
12	GILIMANUK	0.99066	99.900	0.000	5.600	2.000	2
13	NEGARA	0.97333	0.000	0.000	11.600	4.100	3

14	ANTASARI	0.96267	0.000	0.000	6.400	3.100	3
15	KAPAL	0.95066	0.000	30.000	54.700	17.400	3
16	BATURITI	0.94400	0.000	0.000	4.800	0.400	3
17	PEMARON	0.95600	0.000	0.000	0.000	0.000	3
18	PDG SAMBIAN	0.94100	0.000	0.000	28.900	11.700	3
19	PESANGGARAN	0.96400	0.000	0.000	64.900	21.000	3
20	PLTD PSGRAN	0.92000	33.600	0.000	0.000	0.000	2
21	PLTG PSGRAN	0.92000	79.000	0.000	0.000	0.000	2
22	NUSADUA	0.96400	0.000	25.000	56.700	16.300	3
23	SANUR	0.96400	0.000	25.000	56.700	16.300	3
24	GIANYAR	0.94133	0.000	0.000	29.700	5.600	3
25	AMLAPURA	0.93466	0.000	0.000	13.700	6.700	3
26	PAYANGAN	0.94866	0.000	0.000	15.200	3.300	3

Ket: 1 : bus slack
2 : bus generator
3 : bus beban

4.5 Data Saluran Transmisi 150 kV Sub Sistem Paiton dan Bali

Pada sub sistem Paiton dan Bali terdiri dari 26 bus dan 35 saluran transmisi, dalam hal ini saluran transmisi yang dibahas adalah saluran transmisi 150 kV.

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

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

$$X = \frac{X_1 \times X_2}{X_1 + X_2} = \frac{0.0359 \times 0.0359}{0.0359 + 0.0359} = \frac{1.2888 \times 10^{-3}}{0.0718} = 0.0179 \text{ pu}$$

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

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

Tabel 4.9 ^[11]
Data saluran transmisi sub sistem 150 kV Paiton- Bali.

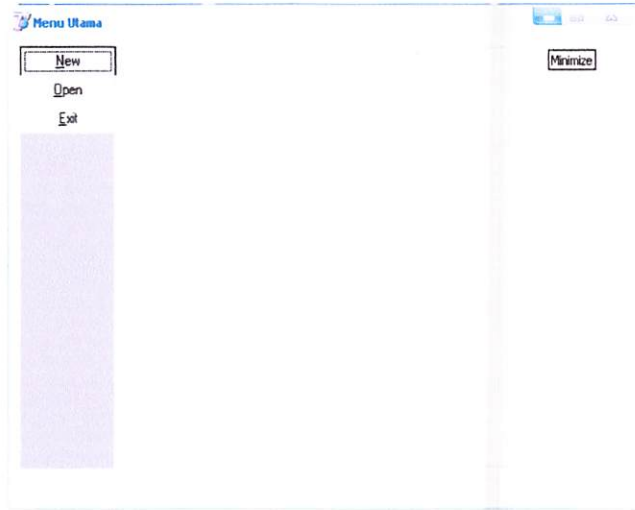
No Saluran	Hubungan Bus	Jumlah sirkuit	R (pu)	X (pu)	B (pu)
1	1-2	2	0.0053	0.0179	0.0065
2	1-9	2	0.0048	0.0346	0.0251
3	2-3	1	0.0100	0.0340	0.0123
4	2-4	2	0.0079	0.0269	0.0097
5	4-5	2	0.0135	0.0460	0.0166
6	5-6	1	0.0167	0.0571	0.0206
7	5-7	1	0.0315	0.1077	0.0389
8	6-7	1	0.0148	0.0506	0.0183
9	7-8	1	0.0337	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	0.0966	0.0457	0.0340
15	11-12	2	0.0016	0.0035	0.0000
16	12-13	1	0.0116	0.0336	0.0124
17	12-17	2	0.0399	0.1314	0.0502
18	13-14	1	0.0270	0.0783	0.0288
19	13-15	1	0.0412	0.1194	0.0439
20	14-15	1	0.0142	0.0411	0.0151
21	15-26	1	0.0132	0.0383	0.0075
22	16-26	1	0.0171	0.0494	0.0091
23	15-17	1	0.0556	0.1054	0.0369
24	15-18	1	0.0105	0.0304	0.0112
25	15-19	1	0.0105	0.0304	0.0112
26	15-22	1	0.0293	0.0561	0.0195
27	15-24	2	0.0059	0.0170	0.0063
28	16-17	1	0.0194	0.0371	0.0129
29	18-19	1	0.0045	0.0130	0.0048
30	19-20	1	0.0020	0.0044	0.0000
31	19-21	1	0.0020	0.0056	0.0000
32	19-22	1	0.0127	0.0243	0.0084
33	19-23	2	0.0037	0.0070	0.0025
34	23-24	2	0.0078	0.0149	0.0052
35	24-25	2	0.0103	0.0298	0.0110

4.6 Prosedur pelaksanaan program perhitungan.

Prosedur menjalankan program perhitungan dilakukan setelah memasukkan seluruh data ke dalam program dengan menggunakan bahasa

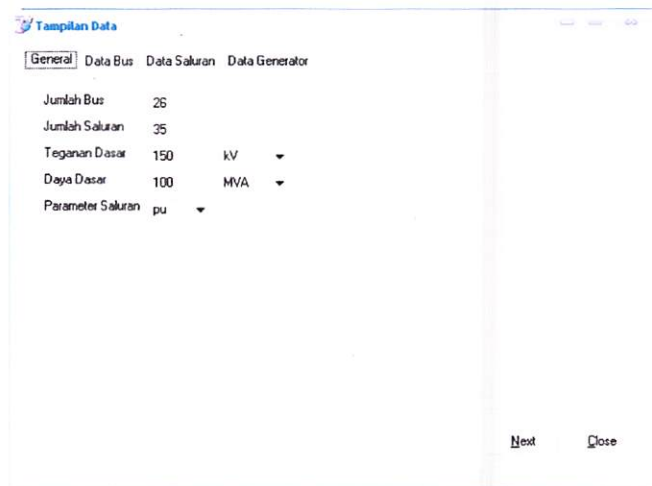
Pemrograman Borland Delphi versi 7.0. Mengenai prosedur jalannya program dapat dilakukan sebagai berikut:

1. Tampilan utama program.



Gambar 4.7
Tampilan Utama Program

2. Kemudian setelah itu tekan tombol open untuk membuka file yang sudah tersimpan.



Gambar 4.8
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	1.03066	0	0	0	14.5	6	0	1
2	1.01333	0	0	0	5.3	6	0	3
3	1.014	0	0	0	19.7	8.7	0	3
4	1	0	0	0	39.7	17.2	0	3
5	0.98666	0	0	0	39.7	15.6	0	3
6	1	0	0	0	27.2	9.4	0	3
7	1	0	0	0	39.8	11.7	0	3
8	0.97333	0	0	0	62.2	30	0	3
9	1.00667	0	0	0	24.9	9.6	0	3
10	0.99333	0	0	0	19.7	7	0	3
11	0.98667	0	0	0	38.1	14	0	3
12	0.99066	0	99.9	0	5.6	2	0	2
13	0.97333	0	0	0	11.6	4.1	0	3
14	0.96267	0	0	0	6.4	3.1	0	3
15	0.952	0	0	0	5.7	1.4	0	3

Gambar 4.9
Tampilan Data Bus

4. Kemudian tekan data saluran.

No	Dari	Ke	R (pu)	X (pu)	Lc (pu)	Tr	Tu	Su (deg)	Kap (MVA)
1	1	2	0.0053	0.0179	0.0065	0	0	0	222
2	1	9	0.0048	0.0346	0.0251	0	0	0	480
3	2	3	0.01	0.034	0.0123	0	0	0	111
4	2	4	0.0079	0.0269	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	0.0206	0	0	0	111
7	5	7	0.0315	0.1077	0.0389	0	0	0	111
8	6	7	0.0148	0.0506	0.0183	0	0	0	111
9	7	8	0.0337	0.1151	0.0415	0	0	0	111
10	7	10	0.0104	0.0353	0.0128	0	0	0	222
11	7	11	0.043	0.1469	0.053	0	0	0	111
12	8	11	0.0177	0.0604	0.0218	0	0	0	111
13	9	10	0.0091	0.031	0.0112	0	0	0	222
14	9	11	0.0355	0.1157	0.034	0	0	0	111

Gambar 4.10
Tampilan Data Saluran

5. Tekan tombol data generator

No	Bus	Qmin (MVAR)	Qmax (MVAR)	a2	a1	a0	Fix Cost	Var Cost	Pmin
1	1	-200.00	300.00	6.18000	1306.15000	388144.168	0.73	5.10	150.
2	12	-45.00	80.00	1.40690	1599.00000	87435.0000	1.03	7.20	50.0
3	20	-20.00	60.00	14.29320	1670.00000	88960.0000	1.03	7.20	21.0
4	21	-15.00	75.00	113.90000	5297.00000	139560.000	1.03	7.20	15.0

Gambar 4.11
Tampilan Data Generator

6. Tekan tombol next, kemudian tekan tombol *Use Default* untuk memasukkan parameter IEP atau berikan parameter yang sudah ditentukan.

Parameter Improve Evolutionary Programming		Parameter Objective Function	
Jumlah Pendinginan	80	Konstanta Kp	10
Jumlah Populasi	10	Konstanta Kq	250
Jumlah Length	7	Konstanta Kv	2000
Jumlah Sub Populasi	2	Konstanta Ks	900
Probabilitas Mutasi	0.2	Batas Tegangan (pu)	0.95 1.05
Konstanta Pr	0.01	Batas Tap Trafo	0.9 1.1
Konstanta Lamda	0.9	Step Tap Trafo	0.001
Konstanta A	0.9	Konstanta Ka	1000000

Use Default

Hitung Lf Awal Close

Gambar 4.12
Tampilan Parameter IEP.

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

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sup (pu)	Type Bus
1	1.04553	0.00000	506.329	28.655	14.500	6.000	0.000	1
2	1.02916	-0.02984	0.000	0.000	5.300	6.000	0.000	3
3	1.02476	-0.03549	0.000	0.000	19.700	8.700	0.000	3
4	1.01089	-0.07058	0.000	0.000	39.700	17.200	0.000	3
5	0.99457	-0.12645	0.000	0.000	39.700	15.600	0.000	3
6	0.98956	-0.15448	0.000	0.000	27.200	9.400	0.000	3
7	0.99247	-0.16620	0.000	0.000	39.800	11.700	0.000	3
8	0.95656	-0.24397	0.000	0.000	62.200	30.000	0.000	3
9	1.04168	-0.09619	0.000	0.000	24.900	9.600	0.000	3
10	1.01639	-0.13101	0.000	0.000	19.700	7.000	0.000	3
11	0.96609	-0.24984	0.000	0.000	38.100	14.000	0.000	3
12	0.96850	-0.25701	102.811	76.045	5.600	2.000	0.000	2
13	0.94961	-0.26842	0.000	0.000	11.600	4.100	0.000	3
14	0.94566	-0.34267	0.000	0.000	6.400	3.100	0.000	3
15	0.94616	-0.37870	0.000	30.000	54.700	17.400	0.000	3

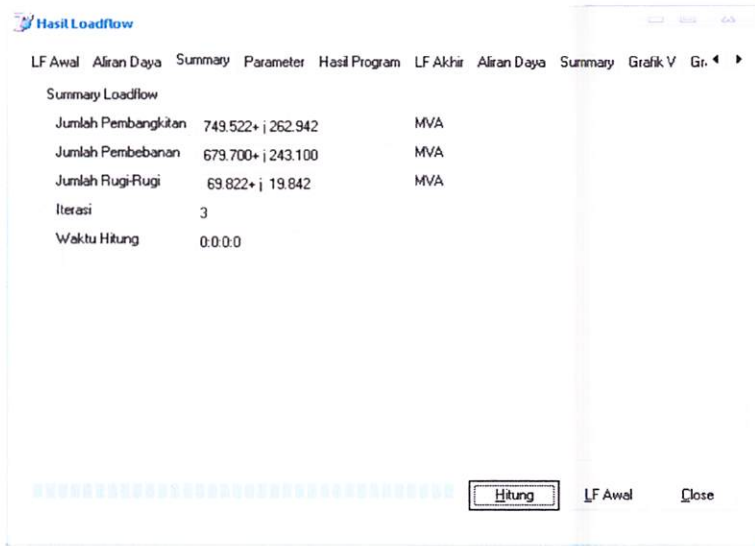
Gambar 4.13
Tampilan hasil *Load Flow* pada Kondisi Awal (Sebelum Optimasi)

8. Tekan tombol *Aliran Daya* untuk melihat kondisi aliran daya pada kondisi awal (sebelum optimasi).

No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	2	201.375	28.700	1302.561	185.640	2	1	-199.308	-
2	1	9	321.148	-59.907	2077.295	-387.498	9	1	-316.339	-
3	2	3	19.743	6.320	128.144	45.729	3	2	-19.700	-
4	2	4	174.265	10.762	1140.800	108.274	4	2	-171.929	-
5	4	5	132.228	-12.420	885.201	-13.685	5	4	-129.853	-
6	5	6	51.428	-13.996	356.436	-44.618	6	5	-50.952	-
7	5	7	38.724	-18.832	276.252	-88.935	7	5	-38.169	-
8	6	7	23.753	-20.997	182.320	-111.935	7	6	-23.611	-
9	7	8	78.951	-2.509	523.136	82.626	8	7	-76.825	-
10	7	10	-126.146	-24.278	-800.328	-318.324	10	7	127.876	-
11	7	11	69.176	-15.876	475.543	-17.683	11	7	-67.045	-
12	8	11	14.626	-31.761	157.345	-181.664	11	8	-14.420	-
13	9	10	149.582	39.057	926.219	350.470	10	9	-147.576	-
14	9	11	141.850	-137.835	997.176	-781.449	11	9	-107.858	-

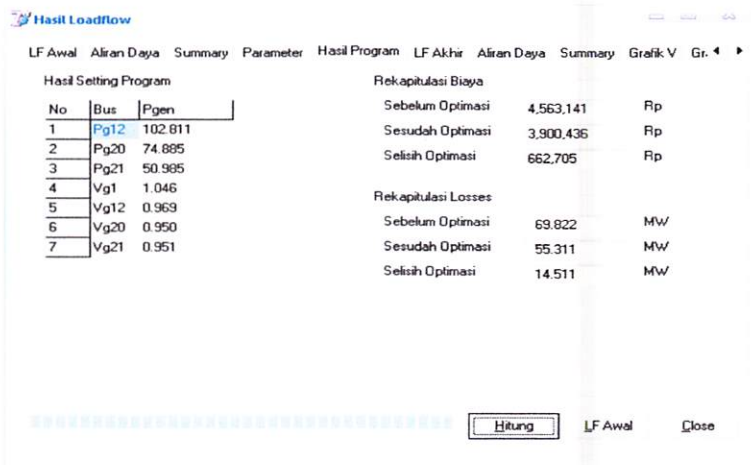
Gambar 4.14
Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi)

9. Tekan tombol *Summary* untuk melihat *summary load flow* (sebelum optimasi)



Gambar 4.15
Tampilan *Summary Load Flow* Sebelum Optimasi

10. Tekan tombol Hasil Program untuk melihat Rekapitulasi Biaya Pembangkitan.



Gambar 4.16
Tampilan Hasil Program pada Kondisi Akhir (Setelah Optimasi)

11. Tekan tombol *Load Flow* akhir untuk melihat hasil perhitungan pada kondisi akhir (setelah optimasi).

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.04553	0.00000	506.329	28.655	14.500	6.000	0.000	1
2	1.02916	-0.02984	0.000	0.000	5.300	6.000	0.000	3
3	1.02476	-0.03549	0.000	0.000	19.700	8.700	0.000	3
4	1.01089	-0.07058	0.000	0.000	39.700	17.200	0.000	3
5	0.99457	-0.12645	0.000	0.000	39.700	15.600	0.000	3
6	0.98956	-0.15448	0.000	0.000	27.200	9.400	0.000	3
7	0.99247	-0.16620	0.000	0.000	39.800	11.700	0.000	3
8	0.95656	-0.24397	0.000	0.000	62.200	30.000	0.000	3
9	1.04168	-0.09619	0.000	0.000	24.900	9.600	0.000	3
10	1.01639	-0.13101	0.000	0.000	19.700	7.000	0.000	3
11	0.96609	-0.24984	0.000	0.000	38.100	14.000	0.000	3
12	0.96850	-0.25701	102.811	76.045	5.600	2.000	0.000	2
13	0.94961	-0.26842	0.000	0.000	11.600	4.100	0.000	3
14	0.94566	-0.34267	0.000	0.000	6.400	3.100	0.000	3
15	0.94616	-0.37870	0.000	30.000	54.700	17.400	0.000	3

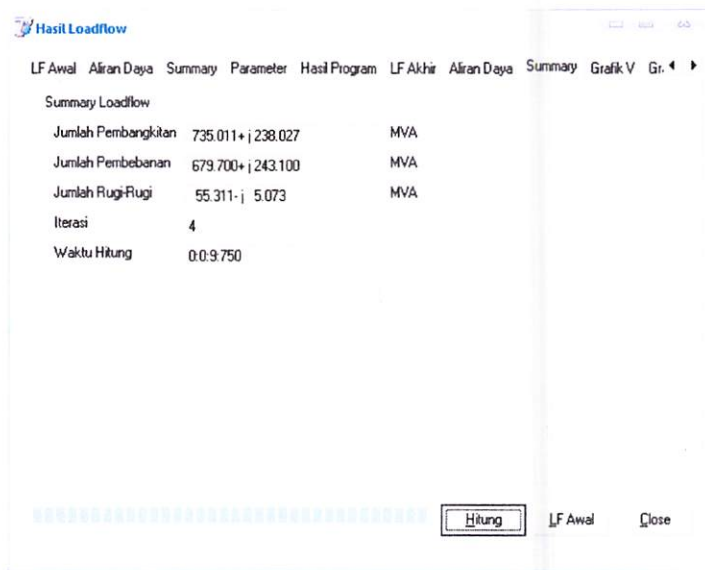
Gambar 4.17
Tampilan Hasil *Load Flow* pada Kondisi Akhir (Setelah Optimasi)

12. Tekan tombol *Aliran Daya* untuk melihat hasil perhitungan pada kondisi akhir (setelah optimasi).

No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	2	191.670	40.857	1222.153	260.519	2	1	-189.805	-
2	1	9	300.159	-18.202	1913.912	-116.062	9	1	-296.193	-
3	2	3	19.742	6.249	126.621	44.277	3	2	-19.700	-
4	2	4	164.763	23.709	1062.241	185.357	4	2	-162.693	-
5	4	5	122.993	1.477	808.416	66.917	5	4	-120.993	-
6	5	6	47.029	-6.395	318.127	-2.764	6	5	-46.652	-
7	5	7	34.264	-11.204	237.312	-45.534	7	5	-33.873	-
8	6	7	19.452	-13.027	142.994	-66.596	7	6	-19.376	-
9	7	8	68.047	9.443	440.301	138.172	8	7	-66.400	-
10	7	10	-110.244	-34.298	-692.235	-349.658	10	7	111.643	-
11	7	11	55.646	-1.410	370.207	52.499	11	7	-54.288	-
12	8	11	4.200	-18.297	59.211	-116.673	11	8	-4.146	-
13	9	10	133.020	46.793	818.617	379.842	10	9	-131.343	-
14	9	11	138.273	-97.720	940.901	-537.517	11	9	-113.381	1

Gambar 4.18
Tampilan *Aliran Daya* pada Kondisi Akhir (Setelah Optimasi)

13. Kemudian tekan tombol *summary* untuk melihat *summary load flow* setelah optimasi.



Gambar 4.19
Tampilan *Summary Load Flow* Setelah Optimasi

4.7. Hasil dan Analisis Hasil Perhitungan *Optimal Power Flow* Menggunakan Metode *Improved Evolutionary Programming* Pada Saluran Transmisi 150 kV Sub Sistem Paiton dan Bali

4.7.1. Hasil Perhitungan Sebelum Optimasi

Tabel 4.10
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sebelum Optimasi

Bus No.	Tegangan		Pembangkit		Pembebanan	
	Magnitude (pu)	Sudut (deg)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1	1.03066	0.00000	537.022	-25.207	14.500	6.000
2	1.01575	-0.03295	0.000	0.000	5.300	6.000
3	1.01128	-0.03875	0.000	0.000	19.700	8.700
4	1.00011	-0.07820	0.000	0.000	39.700	17.200
5	0.98917	-0.14119	0.000	0.000	39.700	15.600
6	0.98791	-0.17329	0.000	0.000	27.200	9.400
7	0.99431	-0.18842	0.000	0.000	39.800	11.700
8	0.96997	-0.28223	0.000	0.000	62.200	30.000
9	1.04080	-0.10635	0.000	0.000	24.900	9.600
10	1.01655	-0.14672	0.000	0.000	19.700	7.000
11	0.98591	-0.29696	0.000	0.000	38.100	14.000
12	0.99066	-0.30569	99.900	229.421	5.600	2.000
13	0.96873	-0.31018	0.000	0.000	11.600	4.100
14	0.93978	-0.38020	0.000	0.000	6.400	3.100
15	0.92700	-0.41544	0.000	30.000	54.700	17.400
16	0.94180	-0.39806	0.000	0.000	4.800	0.400
17	0.94180	-0.38667	0.000	0.000	0.000	0.000
18	0.91966	-0.42371	0.000	0.000	28.900	11.700
19	0.91941	-0.42338	0.000	0.000	64.900	21.000
20	0.92000	-0.42156	33.600	-2.954	0.000	0.000
21	0.92000	-0.41771	79.000	-18.318	0.000	0.000
22	0.91820	-0.43348	0.000	25.000	56.700	16.300
23	0.91874	-0.42760	0.000	25.000	56.700	16.300
24	0.92065	-0.42572	0.000	0.000	29.700	5.600
25	0.91723	-0.42985	0.000	0.000	13.700	6.700
26	0.93177	-0.41139	0.000	0.000	15.200	3.300

Tabel 4.11
 Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi

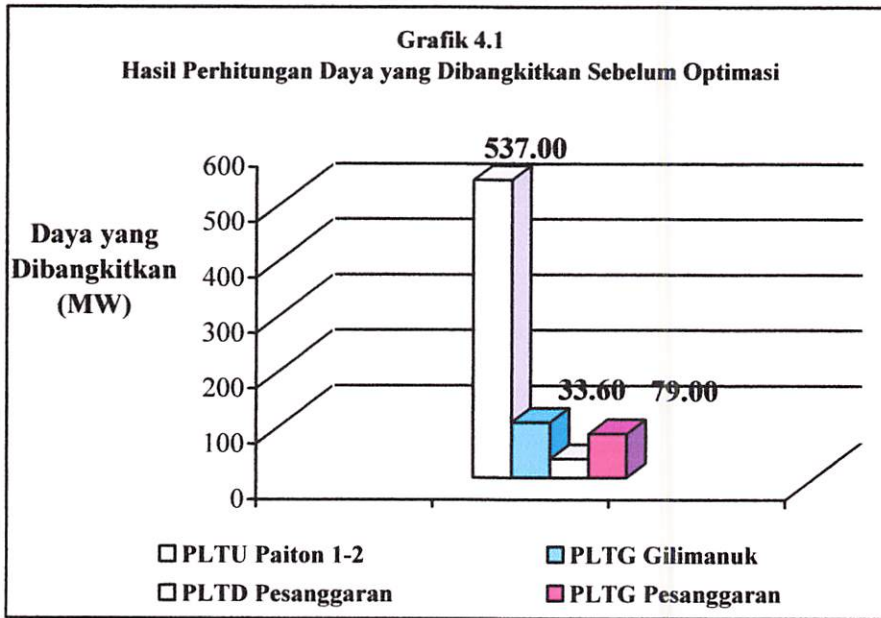
No.	Saluran		Daya	
	Dari	Ke	Aktif (MW)	Reaktif (MVAR)
1	1	2	201.375	28.700
2	1	9	321.148	-59.907
3	2	3	19.743	6.320
4	2	4	174.265	10.762
5	4	5	132.228	-12.420
6	5	6	51.428	-13.996
7	5	7	38.056	-18.832
8	6	7	23.672	-20.997
9	7	8	78.110	-2.509
10	7	10	-126.098	-24.278
11	7	11	69.810	-15.876
12	8	11	14.871	-31.760
13	9	10	149.122	39.057
14	9	11	141.454	-137.835
15	11	12	151.120	-201.985
16	12	13	182.341	15.172
17	12	17	61.221	7.972
18	13	14	84.123	6.859
19	13	15	82.908	5.543
20	14	15	76.887	2.904
21	15	26	-11.120	-8.452
22	16	26	27.445	9.778
23	15	17	-27.342	-9.087
24	15	18	27.900	11.954
25	15	19	27.058	12.914
26	15	22	27.663	-1.289
27	15	24	56.902	14.590
28	16	17	-31.898	-9.489
29	18	19	-1.361	1.834
30	19	20	-33.573	3.013
31	19	21	-78.845	18.753
32	19	22	29.529	-11.369
33	19	23	43.573	-14.257
34	23	24	-13.219	-5.309
35	24	25	13.727	4.920

Tabel 4.12
 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)	
749.522	262.942	679.700	243.100	69.822	9.842	0 :0:0:0

Tabel 4.13
 Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Sebelum Optimasi

No	Nama Pembangkit	Daya Yang Dibangkitkan (MW)	Biaya Operasi (Rupiah / Jam)
1	PLTU Paiton 1 & 2	537.02	Rp 2,871,841.887
2	PLTG Gilimanuk	99.90	Rp 261,215.976
3	PLTD Pesanggaran	33.60	Rp 283,075.651
4	PLTG Pesanggaran	79.00	Rp 1,268,872.9
Total		749.52	Rp 4,563,141.424



4.7.2. Hasil Perhitungan Setelah Optimasi

Tabel 4.14
 Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Setelah Optimasi.

Bus No.	Tegangan		Pembangkit		Pembebanan	
	Magnitude (pu)	Sudut (deg)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1	1.04553	0.00000	506.329	28.655	11.700	6.000
2	1.02916	-0.02984	0.000	0.000	16.500	6.500
3	1.02476	-0.03549	0.000	0.000	17.500	6.700
4	1.01089	-0.07058	0.000	0.000	39.300	19.800
5	0.99457	-0.12645	0.000	0.000	39.800	17.500
6	0.98956	-0.15448	0.000	0.000	23.400	9.300
7	0.99247	-0.16620	0.000	0.000	58.900	30.300

8	0.98751	-0.24397	0.000	0.000	39.800	21.800
9	1.05074	-0.09619	0.000	0.000	24.400	5.800
10	1.03176	-0.13101	0.000	0.000	17.800	7.400
11	0.99071	-0.22244	0.000	0.000	37.800	14.800
12	0.99270	-0.22836	102.811	76.045	5.600	2.000
13	0.97327	-0.23828	0.000	0.000	11.700	4.600
14	0.96492	-0.31025	0.000	0.000	6.000	2.800
15	0.96277	-0.34557	0.000	30.000	54.300	23.500
16	0.96822	-0.33182	0.000	0.000	4.400	0.500
17	0.97098	-0.32217	0.000	0.000	24.800	8.800
18	0.96031	-0.35315	0.000	0.000	32.800	13.300
19	0.96240	-0.35235	0.000	0.000	76.400	28.900
20	0.96363	-0.34864	74.885	14.701	0.000	0.000
21	0.96700	-0.34873	50.985	38.626	0.000	0.000
22	0.95917	-0.36054	0.000	25.000	51.400	19.300
23	0.96009	-0.35651	0.000	25.000	60.300	17.700
24	0.95888	-0.35496	0.000	0.000	33.700	9.600
25	0.95547	-0.35891	0.000	0.000	14.200	7.000
26	0.94701	-0.37351	0.000	0.000	15.200	3.300

Tabel 4.15
 Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi

No.	Saluran		Daya	
	Dari	Ke	Aktif (MW)	Reaktif (MVAR)
1	1	2	191.670	40.857
2	1	9	300.139	-18.202
3	2	3	19.742	6.249
4	2	4	164.763	23.709
5	4	5	122.993	1.477
6	5	6	47.029	-6.393
7	5	7	34.264	-11.042
8	6	7	19.452	-13.542
9	7	8	68.012	9.014
10	7	10	-110.244	-34.003
11	7	11	55.720	-1.703
12	8	11	4.290	-18.606
13	9	10	133.029	46.311
14	9	11	138.151	-97.800
15	11	12	133.498	-126.942
16	12	13	170.814	-41.347
17	12	17	60.369	-12.047
18	13	14	78.236	-21.257
19	13	15	76.125	-22.392
20	14	15	70.295	-25.277
21	15	26	-11.483	1.208
22	16	26	-26.839	-0.436
23	15	17	-19.232	0.530
24	15	18	23.023	-7.096
25	15	19	18.848	-14.301
26	15	22	21.963	-10.905
27	15	24	52.423	5.507
28	16	17	-22.112	0.507
29	18	19	-3.837	-16.499

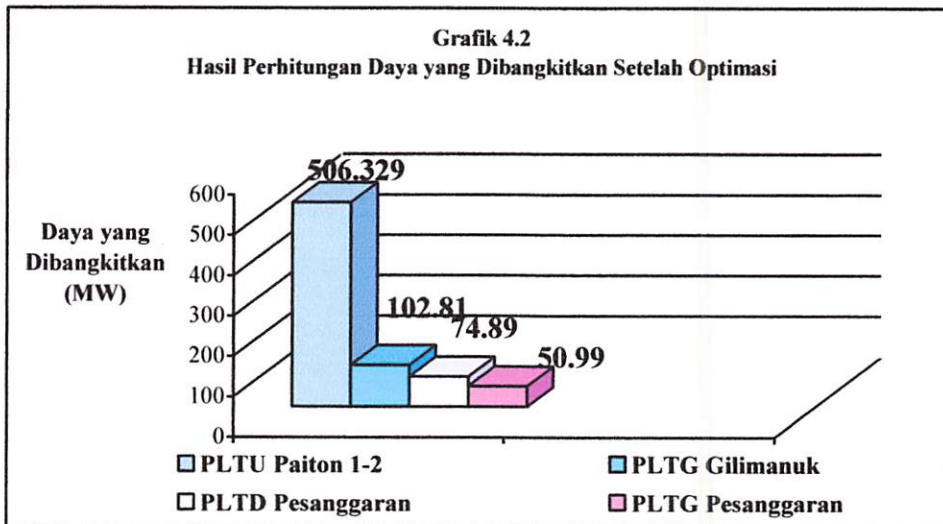
30	19	20	-74.878	-14.329
31	19	21	-50.402	-38.863
32	19	22	32.720	-2.396
33	19	23	48.116	5.038
34	23	24	-8.665	14.551
35	24	25	13.726	4.821

Tabel 4.16
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)	
735.011	238.027	679.700	243.100	55.311	-5.073	0:0:10:188

Tabel 4.17
Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Setelah Optimasi

No	Nama Pembangkit	Daya Yang Dibangkitkan (MW)	Biaya Operasi (Rupiah / Jam)
1	PLTU Paiton 1 & 2	506.329	Rp 2,633,846.557
2	PLTG Gilimanuk	102.811	Rp 266,700.265
3	PLTD Pesanggaran	74.885	Rp 294,170,861
4	PLTG Pesanggaran	50.985	Rp 705,707,83
Total		735.010	Rp 3,900,436.413



4.8. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Optimal Power Flow Menggunakan Metode Parallel Evolutionary Programming.

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

4.8.1. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

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

Tabel 4.18
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

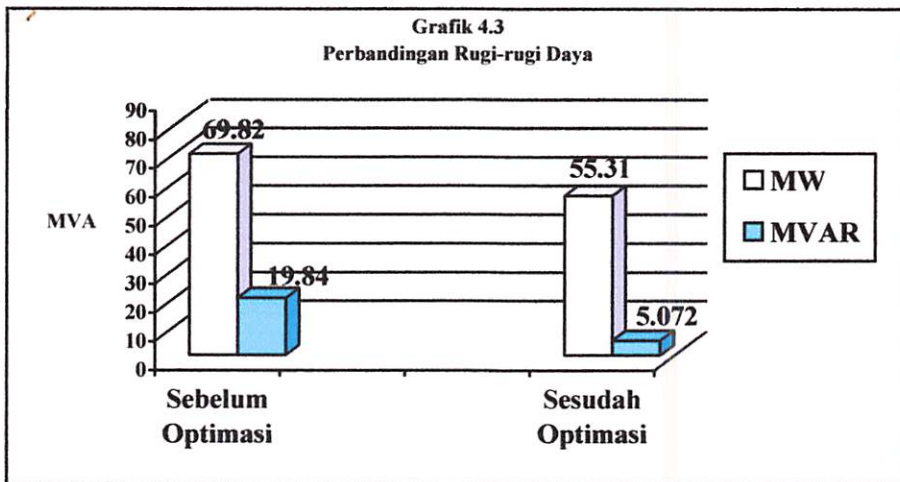
Bus No	Sebelum Optimasi		Sesudah Optimasi	
	Tegangan (pu)	Sudut Tegangan (deg)	Tegangan (pu)	Sudut Tegangan (deg)
1	1.03066	0.00000	1.04553	0.00000
2	1.01575	-0.03295	1.02916	-0.02984
3	1.01128	-0.03875	1.02476	-0.03549
4	1.00011	-0.07820	1.01089	-0.07058
5	0.98917	-0.14119	0.99457	-0.12645
6	0.98791	-0.17329	0.98956	-0.15448
7	0.99431	-0.18842	0.99247	-0.16620
8	0.96997	-0.28223	0.98751	-0.24397
9	1.04080	-0.10635	1.05074	-0.09619
10	1.01655	-0.14672	1.03176	-0.13101
11	0.98591	-0.29696	0.99071	-0.22244
12	0.99066	-0.30569	0.99270	-0.22836
13	0.96873	-0.31018	0.97327	-0.23828
14	0.93978	-0.38020	0.96492	-0.31025
15	0.92700	-0.41544	0.96277	-0.34557
16	0.94180	-0.39806	0.96822	-0.33182
17	0.94180	-0.38667	0.97098	-0.32217
18	0.91966	-0.42371	0.96031	-0.35315
19	0.91941	-0.42338	0.96240	-0.35235
20	0.92000	-0.42156	0.96363	-0.34864
21	0.92000	-0.41771	0.96700	-0.34873
22	0.91820	-0.43348	0.95917	-0.36054
23	0.91874	-0.42760	0.96009	-0.35651
24	0.92065	-0.42572	0.95888	-0.35496

25	0.91723	-0.42985	0.95547	-0.35891
26	0.93177	-0.41139	0.94701	-0.37351

Dari tabel diatas dapat di lihat bahwa tegangan pada tiap bus masih dalam batas yang di perbolehkan yaitu antara -5% sampai +5% dari tegangan nominal.

4.8.2. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran

Untuk rugi-rugi daya pada saluran sebelum dioptimasi adalah 69.82 MW dan 19.84 MVAR, sedangkan perhitungan rugi-rugi daya setelah optimasi adalah 55.31 MW dan 5.073 MVAR. Dari hasil perhitungan dapat dilihat bahwa rugi-rugi daya yang ada sebelum dan sesudah optimasi terjadi penurunan yaitu sebesar 39.317 MW dan 19.638 MVAR.



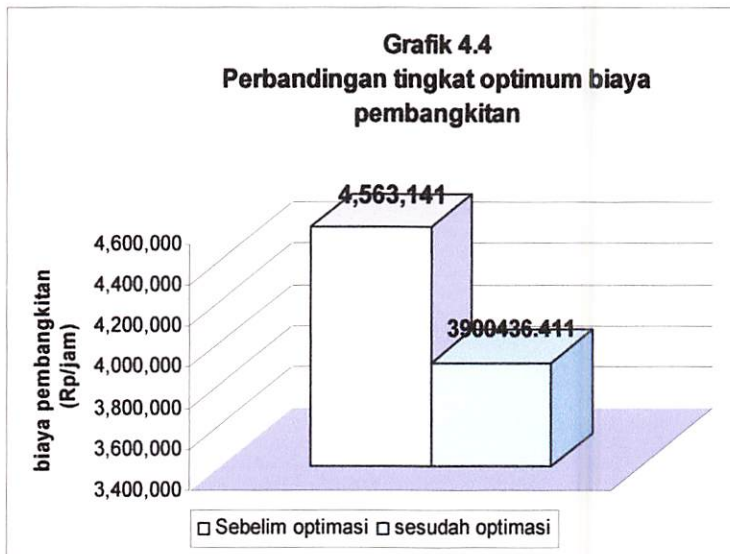
4.8.3. Tingkat Optimum Biaya Pembangkitan

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

Tabel 4.19
Perbandingan Tingkat Optimum Biaya Pembangkitan

No	Nama Pembangkit	Sebelum Optimasi	Setelah Optimasi
1	PLTU Paiton 1 & 2	Rp 2,871,841.887	Rp 2,633,846.557
2	PLTG Gilimanuk	Rp 261,215.976	Rp 266,700.265
3	PLTD Pesanggaran	Rp 283,075.651	Rp 294,170.861
4	PLTG Pesanggaran	Rp 1,268,872.9	Rp 705,707.83
Total		Rp 4,563,141.424	Rp 3,900,436.411

Dari tabel tingkat optimum diatas dapat dilihat bahwa besarnya biaya produksi pembangkitan sebelum optimasi adalah Rp 4,563,141.424,- per jam , sedangkan besarnya biaya produksi pembangkitan setelah optimasi adalah Rp3,900,436.411,- per jam. Jadi selisih antara biaya produksi pembangkitan sebelum dengan sesudah optimasi adalah Rp 662,705.413 - per jam atau terjadi optimasi biaya sebesar 14.52%.



BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah dilakukan analisis perhitungan *Optimal Power Flow (OPF)* dengan menggunakan metode *Improved Evolutionary Programming (IEP)* maka dapat diambil kesimpulan sebagai berikut:

1. Dari hasil perhitungan *loadflow* awal(sebelum optimasi)menggunakan metode Newton Rapshon, maka total daya yang dibangkitkan sebesar 749.522MW, total pembebanan 679.700 MW dan rugi-ruginya sebesar 85.192 MW.

Setelah dilakukan optimasi menggunakan metode IEP maka *loadflow* akhirnya (setelah optimasi), untuk total daya pembangkitan sebesar 735.011 MW, total pembebanan sebesar 679.700 MW dan rugi-ruginya sebesar 54.680 MW

2. Penyelesaian masalah OPF dengan menggunakan metode *Improved Evolutionary Programming* pada sub sistem 150 kV Paiton-Bali untuk melayani permintaan beban puncak sebesar 735.011 MW menghasilkan biaya operasi yang lebih Optimum. Biaya produksi pembangkitan sebelum optimasi adalah Rp 4,563,141.424 per jam, sedangkan besarnya biaya produksi pembangkitan setelah optimasi adalah Rp 3,900,436.411, - per jam. Jadi selisih antara biaya produksi pembangkitan sebelum dengan sesudah optimasi adalah Rp 662,705.01, - per jam atau terjadi optimasi biaya sebesar 14.52 %.

3. Tegangan tiap bus masih dalam batas yang diijinkan yaitu antara -5 % sampai +5% dari tegangan nominal 150 kV.

5.2 Saran

Penggunaan metode *Improved Evolutionary Programming* untuk menyelesaikan permasalahan OPF masih perlu dikembangkan lagi terhadap sistem jaringan yang lebih luas minimal mengganti area penelitian, serta membahas kapasitas saluran, biaya *start up* dan *shut down*, Serta masalah *Spinning Reserve* sehingga hasil yang diperoleh dari penelitian lebih lengkap dan akurat, karena menyertakan seluruh komponen yang berhubungan dengan masalah optimalisasi dalam penyaluran daya listrik.

DAFTAR PUSTAKA

- [1] Dommel, H. W. and W. F. Tinney, "Optimal Power Flow Solutions", IEEE Trans. Power Appar. Syst., Vol. PAS – 87, pp. 1866-1879, Oct 1968.
- [2] Bakirtzis, Anastasios G., Pandel N. Biskas, Christoforos E. Zoumas, Vasilos Petridis, "Optimal Power Flow by Enhanced Genetic Algorithm", IEEE Transactions On Power Systems, vol. 17. No. 2. May 2002.
- [3] Stevenson, William D., Jr, "Analisa Sistem Tenaga Listrik", Penerbit Erlangga edisi ke-empat, 1996.
- [4] Wood, Allan J. and B. F. Wollwnberg, "Power Generation, Operation, and Control", John Wiley & Sons, Inc., 1996.
- [4] Zuhail, "Dasar Teknik Tenaga Listrik dan Elektronika Daya", PT Gramedia Utama, Jakarta, 1995.
- [5] Syafena, Jatri, Skripsi, "Economic Dispatch Dengan Optimasi Daya Aktif dan Reaktif Pada pembangkit Thermal di Area IV", 2002
- [6] Yuryevich, J. and K.P. Wong, "Evolutionary Programming Based Optimal Power Flow Algorithm", IEEE Trans.on Power System, Vol. 14, No. 4, pp.1245-1250, Nov. 1999.
- [7] Luke, Brian T., "Overview of Evolutionary Programming Methods", Learning From the Web.net.
- [8] Kusumadewi, Sri, "Artificial Intelligence (Teknik dan Aplikasinya)", Graha Ilmu, Yogyakarta, 2003
- [9] C.H. Lo, Member, IEEE, C. Y. Chung, Member IEEE, G. H. M. Nguyen, Member, IEEE, and K. P. Wong, fellow, IEEE "Parallel Evolutionary Programming for Optimal Power Flow". IEEE Trans on Power System, DRPT, April 2004
- [10] Ongsakul., W. and Tantimaporn, T. 2004. *Optimal Power Flow Using Improved Evolutionary Programming. Electricity Supply Industry in Transition.*
- [11]. Data Load Flow Region Jawa Bali, Sub-Sistem 150 kV Paiton dan Bali, Tanggal 30 Maret 2005 Pukul 18.00 WIB, PT. PLN(Persero) P3B Region Jawa Timur dan Bali, Waru.



FORMULIR BIMBINGAN SKRIPSI

Nama : WIKY INDRA NURCAHYONO
Nim : 60.12.082
Masa Bimbingan : 27 Juni 2006 s/d 27 Desember 2006
Skripsi : **OPTIMAL POWER FLOW MENGGUNAKAN
METODE IMPROVED EVOLUTIONARY PROGRAMMING
PADA SUB-SISTEM 150 kV PAITON-BALI**

No.	Tanggal	Uraian	Parap Pembimbing
1.	16-08-2006	Konsultasi bab I, II	
2.	19-08-2006	Konsultasi perbaikan bab I & II sekaligus pengajuan bab III	
3.	28-08-2006	Konsultasi perbaikan bab III & melakukan test uji validasi	
4.	30-08-2006	Cek kembali error hasil uji validasi dengan journal	
5.	01-09-2006	Konsultasi validasi dan pengajuan bab IV	
6.	02-09-2006	Konsultasi perbaikan bab IV dan pengajuan makalah seminar	
7.	07-09-2006	Periksa kembali selisih biaya program, apakah sudah optimal	
8.	09-09-2006	Perbaiki dan periksa lagi masalah selisih total biaya	
9.	11-09-2006	Acc makalah seminar hasil	
10.	21-09-2006	Konsultasi seluruh bab dan Acc Ujian Skripsi	

Malang, September 2006
Dosen Pembimbing,

Ir. Yusuf Ismail Nakhoda, MT
Nip. Y. 1018800189



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

LEMBAR BIMBINGAN SKRIPSI

Nama Mahasiswa : WIKY INDRA NURCAHYONO
N.I.M. : 00.12.082
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : *OPTIMAL POWER FLOW MENGGUNAKAN
IMPROVED EVOLUTIONARY PROGRAMMING (IEP)
PADA SUB-SISTEM 150 kV PAITON-BALI*

Tanggal Mengajukan Skripsi : 08 April 2006
Tanggal Menyelesaikan Skripsi : 23 September 2006
Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT
Telah dievaluasi dengan nilai : 86 (Delapan Puluh Enam)

Malang, September 2006

Mengetahui,
Ketua Jurusan Teknik Elektro

Ir. F. Yudi Limpraptono, MT
NIP. Y. 103 9500 274

Menyetujui,
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 101 8800 189



LEMBAR BIMBINGAN SKRIPSI

Nama Mahasiswa : WIKY INDRA NURCAHYONO
N.I.M. : 00121082
Jurusan : Teknik Elektro 2-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : OPTIMAL POWER FLOW MENGGUNAKAN IMPROVED EVOLUTIONARY PROGRAMMING (IEP) PADA SUB-SISTEM 150 KV PATON-BALI

Tanggal Mengajukan Skripsi : 08 April 2006
Tanggal Menyetujui Skripsi : 23 September 2006
Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT
Telah divalidasi dengan nilai : 86 (Delapan Puluh Enam)

Malang, September 2006

Menyetujui,
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT
NIP. Y. 101 8800 189

Mengetahui,
Ketua Jurusan Teknik Elektro

Ir. R. Yudi Himpigtono, MT
NIP. Y. 103 9500 274



BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI

Nama Mahasiswa : WIKY INDRA NURCAHYONO
N.I.M. : 00.12.082
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : *OPTIMAL POWER FLOW MENGGUNAKAN METODE
IMPROVED EVOLUTIONARY PROGRAMMING (IEP)
PADA SUB-SISTEM 150 kV PAITON-BALI*

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

Hari : Senin
Tanggal : 25 September 2006
Dengan Nilai : 78,95 (B+)

Panitia Ujian



Ketua
(Ir. Mochtar Asroni, MSME)
Nip. Y. 101 8100 030

Sekretaris
(Ir. F. Yudi Limpraptono, MT)
NIP. Y. 103 9500 274

Anggota Penguji

Penguji Pertama

(Ir. Choirul Saleh, MT)
Nip. Y. 101 8800 190

Penguji Kedua

(Ir. Eko Nurcahyo)
Nip. P. 102 8700 172

PT PLN (PERSERO)

**PENYALURAN DAN PUSAT PENGATUR BEBAN JAWA BALI
REGION JAWA TIMUR & BALI**

Jalan Satrio No. 45 Taman Sidoarjo 61257

Telp. (031) 7882113, 7882114

Fax. (031) 7882119

Facsimile : (031) 7882578, 7881024

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

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

Surat No. : 182 /330/RJTB/2006.
No. : ITN-1663/III.TA/2/2006.
: 1 (satu) lampiran.
: Ijin Survey/ Pengambilan Data.

24 JUL 2006

Kepada

Yth. Dekan Fakultas Teknik.
Institut Teknologi Nasional Malang
Di

MALANG

Menunjuk surat Saudara nomor : ITN-1663/III.TA/2/2006 tanggal 16 Mei 2006 perihal : Survey/ Permintaan Data, dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada Mahasiswa Saudara, bernama :

• **WIKY INDRA N**

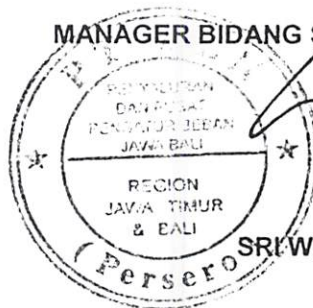
Nim : 00.12.082

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

1. Mahasiswa tersebut diatas supaya mengisi dan menanda tangani Surat Pernyataan 1 (satu) lembar bermeterai 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.

MANAGER BIDANG SDM & ADMINISTRASI,



Tembusan Yth. :

1. M.SDMO PT PLN (Persero) P3B.
2. MOPHAR PT PLN (Persero) RJTB.
3. Sdr. Wiky Indra N
4. AsMan Operasi.

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : WIKY INDRA N
Pria / Wanita : Pria
Tempat Tanggal lahir : Malang 2 Januari 1982
Alamat No. telepon : Jl. Tirtagangga 31 / 0391-313370
Kecamatan : Mahan

Sehubungan dengan ini saya menerangkan bahwa :

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 ;

Saya atas peringatan pertama akan membayar sepenuhnya, semua biaya yang langsung menimbulkan kerugian atau kecelakaan, karena kelalaian saya ;

Saya akan segera mematuhi semua petunjuk -petunjuk yang diberikan oleh Petugas PT PLN (Persero) P3B Region Jawa Timur dan Bali ;

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

Saya sanggup menanggung sendiri segala sesuatu untuk keperluan Training/ Praktek Kerja/ Riset termasuk biaya perjalanan, penginapan makan dan sebagainya ;

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.

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, 24-7-2006
Yang membuat pernyataan


WIKY I. N.

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : WIKY INDEA N
Pria/ ~~Wanita~~ : Pria
Tempat / Tanggal lahir : Malang 2 Januari 1987
Alamat / No. telepon : Jl Tirtagangga 31 / 0291 242270
Pekerjaan : Mahasiswa

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, 24-7-2006
Yang membuat pernyataan

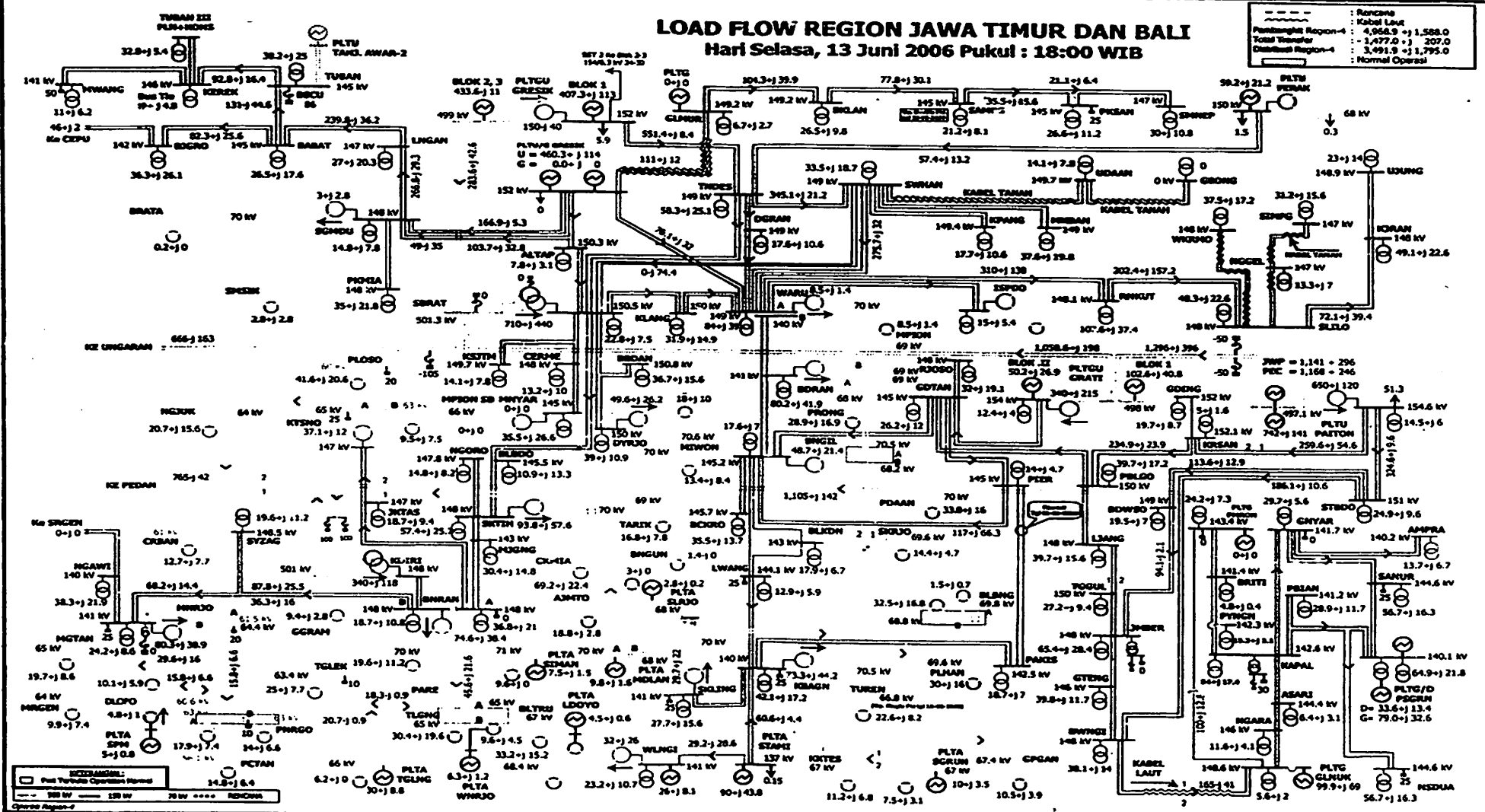
Materai
Rp.6000

WIKY INDEA N

LOAD FLOW REGION JAWA TIMUR DAN BALI

Hari Selasa, 13 Juni 2006 Pukul : 18:00 WIB

-----	: Rencana
-----	: Kabel Lint
-----	: Pembangkit Region-4
-----	: Total Transfer
-----	: Demand Region-4
-----	: Normal Operasi

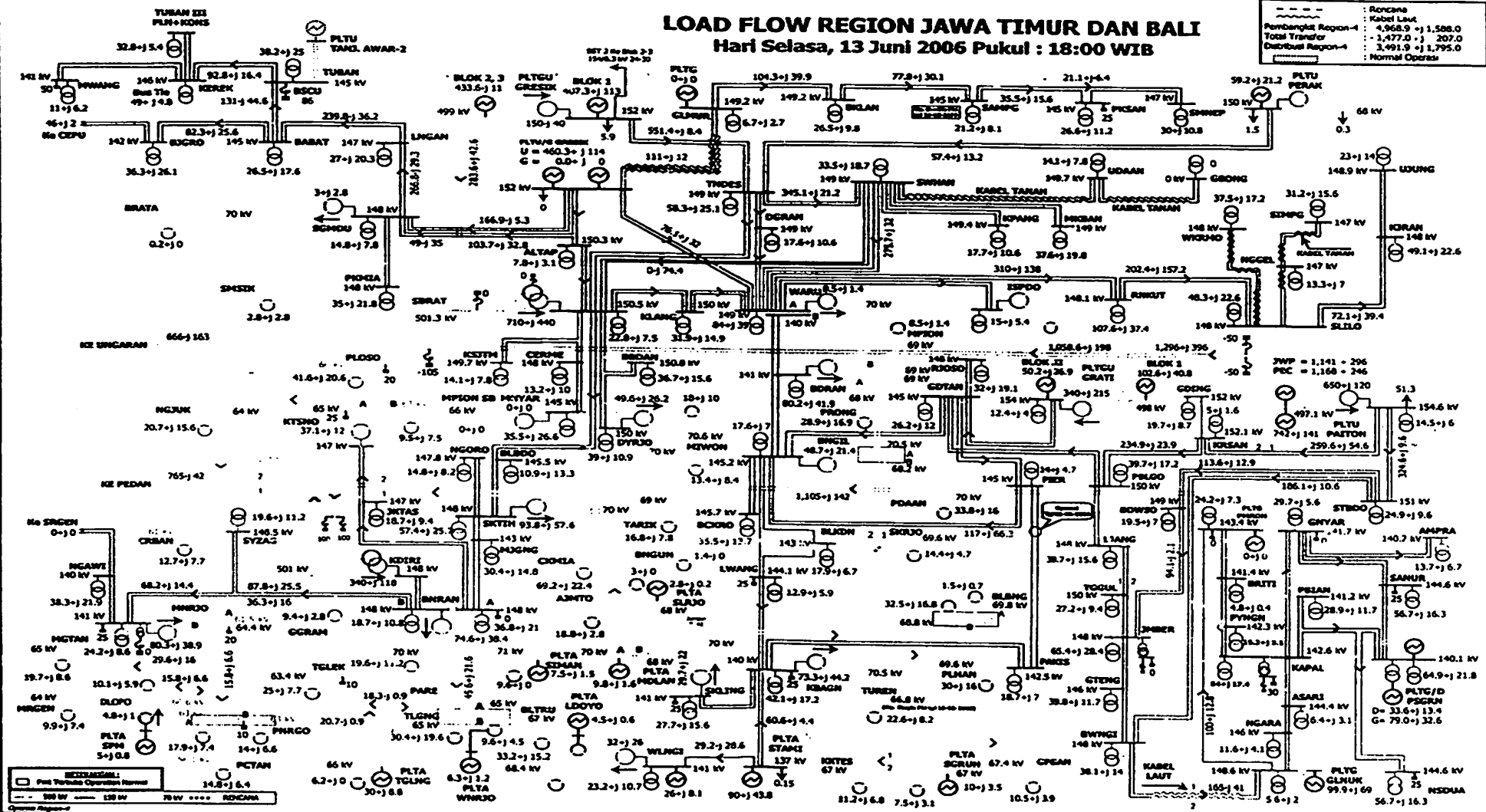


-----	: Rencana
-----	: Kabel Lint
-----	: Pembangkit Region-4
-----	: Total Transfer
-----	: Demand Region-4
-----	: Normal Operasi

LOAD FLOW REGION JAWA TIMUR DAN BALI

Hari Selasa, 13 Juni 2006 Pukul : 18:00 WIB

-----	Rencana
-----	Kabel Laut
-----	Pembangkit Region-4
-----	Total Transfer
-----	Distribusi Region-4
-----	Normal Operasi
	4,968.9 +j 1,588.0
	1,477.0 +j 207.0
	3,491.9 +j 1,795.0



LEGENDA:
 - - - - - Rencana
 ———— Pilih Perbaikan Operasi Normal
 150 kV ———— 138 kV ———— 70 kV

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jml SRB	Dart	No Sekt	No Top (m)	Jarak (km)	Data Pemfaktoran		Jenis	Ked (Amp)	Top CT Dart No	Top VIT Dart No	R (Ohm/km)	X (Ohm/km)	B (Ohm/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z ad (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Daya (MVA)
					Basis	Top																	
1	PITOM7	GRA117	1	500	88,363	GANNET	443928	2800	2000	2000	0,0251	0,2808	4,0490	2,2179	24,8123	357,7818	24,9113	2,500,000	0,0100	0,0099	0,0099	0,0072	2,424,6000
1	BNG15	GOTAN5	1	150	16,805	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	1,9695	6,7270	47,9447	7,0094	225,0000	0,0312	0,0088	0,0289	0,0108	192,2500
1	BNG15	GOTAN5	2	150	16,805	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	1,9695	6,7270	47,9447	7,0094	225,0000	0,0312	0,0088	0,0289	0,0108	192,2500
1	BNG15	WAR15	1	150	9,800	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	1,1466	3,9229	27,9594	4,0876	225,0000	0,0182	0,0051	0,0174	0,0063	192,2500
1	BNG15	LIWANG5	1	150	34,689	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	4,0645	13,8824	98,9420	14,4652	225,0000	0,0643	0,0181	0,0517	0,0063	192,2500
1	BNG15	BLDKN5	1	150	27,770	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	3,2546	11,163	79,2278	11,5830	225,0000	0,0515	0,0145	0,0484	0,0178	192,2500
1	BNG15	PIERS	2	150	5,150	ACSRAW	2 x 435 mm ²	1620	2000	1000	0,3870	0,2807	4,0203	1,9931	14,456	20,7045	2,4621	225,0000	0,0109	0,0089	0,0064	0,0047	420,8760
1	BNG15	ROKRO5	1	150	6,200	ACSRAW	2 x 435 mm ²	1620	2000	1000	0,3870	0,2807	4,0203	1,9931	14,456	20,7045	2,4621	225,0000	0,0109	0,0089	0,0064	0,0047	420,8760
1	BNG15	ROKRO5	2	150	6,200	ACSRAW	2 x 435 mm ²	1620	2000	1000	0,3870	0,2807	4,0203	2,3994	1,7403	24,9295	2,9641	225,0000	0,0132	0,0107	0,0077	0,0056	420,8760
1	GOTAN5	RUSOS	1	150	10,487	ACSRAW	330 mm ²	740	1000	800	0,1172	0,4003	2,6530	1,2,31	4,1919	24,9295	2,9641	225,0000	0,0132	0,0107	0,0077	0,0056	420,8760
1	GOTAN5	RUSOS	2	150	10,487	ACSRAW	330 mm ²	740	1000	800	0,1172	0,4003	2,6530	1,2,31	4,1919	24,9295	2,9641	225,0000	0,0194	0,0055	0,0187	0,0087	192,2500
1	PIERS	GRA115	1	150	30,000	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	11,6100	8,4210	120,6990	14,3424	225,0000	0,0637	0,0516	0,0374	0,0271	192,2500
1	PELGG5	GOTAN5	1	150	33,827	ACSRAW	330 mm ²	740	800	1000	0,1172	0,4003	2,6530	3,9645	13,5409	96,5084	14,1094	225,0000	0,0627	0,0176	0,0602	0,0217	192,2500
1	PELGG5	GOTAN5	2	150	33,827	ACSRAW	330 mm ²	740	800	1000	0,1172	0,4003	2,6530	3,9645	13,5409	96,5084	14,1094	225,0000	0,0627	0,0176	0,0602	0,0217	192,2500
1	PELGG5	LIANG5	1	150	51,692	ACSRAW	330 mm ²	740	600	800	0,1172	0,4003	2,6530	6,0583	20,6923	147,4773	21,5610	225,0000	0,0958	0,0269	0,0332	0,0332	192,2500
1	PELGG5	LIANG5	2	150	51,692	ACSRAW	330 mm ²	740	600	800	0,1172	0,4003	2,6530	6,0583	20,6923	147,4773	21,5610	225,0000	0,0958	0,0269	0,0332	0,0332	192,2500
1	KRSAN5	PELGG5	1	150	30,239	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	3,5440	12,1047	86,2719	12,6128	225,0000	0,0561	0,0158	0,0358	0,0184	192,2500
1	KRSAN5	PELGG5	2	150	30,239	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	3,5440	12,1047	86,2719	12,6128	225,0000	0,0561	0,0158	0,0358	0,0184	192,2500
1	KRSAN5	GONIG5	1	150	19,137	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	2,2429	7,6605	54,5979	7,9821	225,0000	0,0355	0,0160	0,0340	0,0160	192,2500
1	KRSAN5	PELGG5	1	150	20,204	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	2,2429	7,6605	54,5979	7,9821	225,0000	0,0355	0,0160	0,0340	0,0160	192,2500
1	PITONS	KRSAN5	1	150	20,204	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	2,2429	7,6605	54,5979	7,9821	225,0000	0,0355	0,0160	0,0340	0,0160	192,2500
1	PITONS	KRSAN5	2	150	20,204	ACSRAW	330 mm ²	740	1000	1000	0,1172	0,4003	2,6530	2,2429	7,6605	54,5979	7,9821	225,0000	0,0355	0,0160	0,0340	0,0160	192,2500
1	PITONS	STB05	1	150	55,433	ACSRAW	2 x 435 mm ²	1620	2000	2000	0,3870	0,2807	4,0203	21,4526	15,5600	222,8571	26,5015	225,0000	0,1178	0,0953	0,0882	0,0692	415,6800
1	PITONS	STB05	2	150	55,433	ACSRAW	2 x 435 mm ²	1620	2000	2000	0,3870	0,2807	4,0203	21,4526	15,5600	222,8571	26,5015	225,0000	0,1178	0,0953	0,0882	0,0692	415,6800
1	BNG14	POAN4	1	70	9,700	ACSRAW	300 MCM	440	250	400	0,2175	0,3871	2,9580	2,1098	3,7549	28,6732	4,3070	49,0000	0,0879	0,0431	0,0766	0,0014	53,3456
1	BNG14	BOAN4	1	70	9,700	ACSRAW	300 MCM	440	250	400	0,2175	0,3871	2,9580	2,1098	3,7549	28,6732	4,3070	49,0000	0,0879	0,0431	0,0766	0,0014	53,3456
1	BNG14	PRONG4	1	70	11,700	ACSRAW	300 MCM	440	400	400	0,2180	0,3859	2,9580	2,5056	4,5150	34,6905	5,1857	49,0000	0,1058	0,0521	0,0921	0,0017	53,3456
1	BNG14	BLBNG4	1	70	40,000	ACSRAW	300 MCM	300	200	200	0,3866	0,3594	3,2020	15,6640	14,3360	128,0800	21,0869	49,0000	0,4303	0,3156	0,2926	0,0063	36,3170
1	BNG14	BLBNG4	2	70	40,000	ACSRAW	300 MCM	300	200	200	0,3866	0,3594	3,2020	15,6640	14,3360	128,0800	21,0869	49,0000	0,4303	0,3156	0,2926	0,0063	36,3170
1	GRAT15	GOTAN5	1	150	18,500	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	4,1575	3,0156	43,1901	8,8445	225,0000	0,0393	0,0318	0,0202	0,0167	420,8760
1	GRAT15	GOTAN5	2	150	10,743	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	4,1575	3,0156	43,1901	8,8445	225,0000	0,0393	0,0318	0,0202	0,0167	420,8760
1	GOTAN5	PIERS	1	150	11,500	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	3,2781	46,2335	5,4979	225,0000	0,0244	0,0186	0,0143	0,0104	420,8760	
1	GOTAN5	PIERS	2	150	11,500	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	3,2781	46,2335	5,4979	225,0000	0,0244	0,0186	0,0143	0,0104	420,8760	

RXY, deskrip, tgl 06/08/2006

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Halaman: 11

Jml Sheet	Dati	No Sheet	Tag (V)	Jarak (m)	Data Pembarutan		Jenis	Nom (Amp)	Tag CT	Dati	Kø	Tag WT	Dati	Kø	R (Ohm/km)	X (Ohm/km)	B (Mg/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z sat (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Daya (MVA)
					U	Jenis																					
1	PBLG05	1	150	51,892	ACSR DOVE	330 mm ²	740	600	600	800	800	800	800	800	0,1172	0,4003	2,8530	6,0.35	20,6922	147,4773	21,5610	225,0000	0,0958	0,0289	0,0920	0,0332	192,2530
1	PBLG05	2	150	51,892	ACSR DOVE	330 mm ²	740	600	600	800	800	800	800	800	0,1172	0,4003	2,8530	6,0.35	20,6922	147,4773	21,5610	225,0000	0,0958	0,0289	0,0920	0,0332	192,2530
1	LANG05	1	150	32,080	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	7,0941	24,2002	172,6921	25,2473	225,0000	0,1122	0,0315	0,1077	0,0389	192,2530
1	LANG05	1	150	60,530	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	3,3343	11,3985	81,1679	11,8666	225,0000	0,0527	0,0148	0,0506	0,0183	192,2530
1	TNGUL5	1	150	28,450	ACSR AW	330 mm ²	740	1000	800	800	800	800	800	800	0,1172	0,4003	2,8530	9,6801	33,0628	235,6435	74,4507	225,0000	0,1531	0,0430	0,1449	0,0530	192,2530
1	JMBERS	1	150	82,285	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	7,5838	25,8026	184,6119	26,9900	225,0000	0,1200	0,0337	0,1151	0,0530	192,2530
1	JMBERS	1	150	64,708	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	4,6484	15,6787	113,1557	16,5432	225,0000	0,0735	0,0207	0,0706	0,0255	192,2530
1	JMBERS	1	150	39,662	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	7,5838	25,8026	184,6119	26,9900	225,0000	0,1200	0,0337	0,1151	0,0530	192,2530
1	JMBERS	2	150	39,662	ACSR AW	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	7,5838	25,8026	184,6119	26,9900	225,0000	0,1200	0,0337	0,1151	0,0530	192,2530
1	JMBERS	2	150	55,433	ACSR ZEبرا	2 x 435 mm ²	1620	2000	2000	1250	1250	1250	1250	1250	0,3870	0,2807	4,0203	21,4526	15,5800	222,8573	26,5015	225,0000	0,1178	0,0353	0,0922	0,0302	420,8760
1	STBDOS	1	150	55,433	ACSR ZEبرا	2 x 435 mm ²	1620	2000	2000	1250	1250	1250	1250	1250	0,3870	0,2807	4,0203	21,4526	15,5800	222,8573	26,5015	225,0000	0,1178	0,0353	0,0922	0,0302	420,8760
1	STBDOS	2	150	56,433	ACSR ZEبرا	2 x 435 mm ²	1620	2000	2000	1250	1250	1250	1250	1250	0,1172	0,4003	2,8530	4,0789	13,8248	99,2445	14,5094	225,0000	0,0645	0,0181	0,0819	0,0223	192,2530
1	STBDOS	2	150	34,786	ACSR AW	330 mm ²	740	800	800	1600	1600	1600	1600	1600	0,1172	0,4003	2,8530	4,0789	13,8248	99,2445	14,5094	225,0000	0,0645	0,0181	0,0819	0,0223	192,2530
1	STBDOS	2	150	34,786	ACSR AW	330 mm ²	740	800	800	1600	1600	1600	1600	1600	0,1172	0,4003	2,8530	4,0789	13,8248	99,2445	14,5094	225,0000	0,0645	0,0181	0,0819	0,0223	192,2530
1	STBDOS	2	150	74,200	ACSR DOVE	2 x 340 mm ²	1200	800	800	1600	1600	1600	1600	1600	0,0986	0,2815	4,0740	4,3481	20,8873	302,2908	21,3351	225,0000	0,0948	0,0193	0,0928	0,0480	311,7600
1	STBDOS	2	150	74,200	ACSR DOVE	2 x 340 mm ²	1200	800	800	1600	1600	1600	1600	1600	0,0986	0,2815	4,0740	4,3481	20,8873	302,2908	21,3351	225,0000	0,0948	0,0193	0,0928	0,0480	311,7600
1	BWNGIS	1	150	33,990	ACSR DOVE	330 mm ²	740	800	800	1250	1250	1250	1250	1250	0,1370	0,4003	2,8530	3,9789	13,9902	98,8594	14,1607	225,0000	0,0628	0,0183	0,0804	0,0218	192,2530
1	BWNGIS	1	150	7,990	ACSR DOVE	477 MM	655	600	600	800	800	800	800	800	0,1370	0,3966	2,8800	1,0946	3,1698	23,0112	3,3526	225,0000	0,0149	0,0048	0,0141	0,0482	170,1690
1	BWNGIS	1	150	4,417	SUBMARE CABLE	330 mm ²	500	600	600	800	800	800	800	800	0,0870	0,1310	73,0000	0,3980	0,9987	326,3830	0,7031	225,0000	0,0031	0,0017	0,0026	0,0734	120,8070
1	BWNGIS	1	150	4,829	ACSR HAWK	300 mm ²	465	600	600	800	800	800	800	800	0,1370	0,3966	2,8800	1,0946	3,1698	23,0112	3,3526	225,0000	0,0149	0,0048	0,0141	0,0482	170,1690
1	BWNGIS	2	150	7,990	ACSR HAWK	477 MM	655	600	600	800	800	800	800	800	0,0870	0,1310	73,0000	0,3980	0,9987	326,3830	0,7031	225,0000	0,0031	0,0017	0,0026	0,0734	120,8070
1	BWNGIS	2	150	4,389	SUBMARE CABLE	300 mm ²	500	600	600	800	800	800	800	800	0,0870	0,1310	73,0000	0,3980	0,9987	326,3830	0,7031	225,0000	0,0031	0,0017	0,0026	0,0734	120,8070
1	BWNGIS	2	150	4,280	OSS	300 mm ²	465	600	600	800	800	800	800	800	0,0870	0,1310	73,0000	0,3980	0,9987	326,3830	0,7031	225,0000	0,0031	0,0017	0,0026	0,0734	120,8070

7. UPT Jember

DATA KARAKTERISTIK TRANSMISI UPT 4 SRB

Halaman 1

Jml Bikit	Darl	Ke	No. Bikit	Tag (M)	Jarak (km)	Data Pengantar		Jenis	I Nom (Amp)	Tap CT Darl No	Tap VIT Darl No	R (Ohm)	X (Ohm)	B (Ohm)	R (Ohm)	X (Ohm)	B (Ohm)	Z ad (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Days (MVA)
1	GLUNUK	CH GLUNUK	1	150	1,720	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	0.2366	0.6822	4.9636	0.7217	225.0000	0.0032	0.0010	0.0030	0.0014	167.5710
1	GLUNUK	CH GLUNUK	2	150	7,72	ACSR HAWK	477 MCM		645	800	300	0.1370	0.3966	2.6320	0.2366	0.6822	4.9636	0.7217	225.0000	0.0032	0.0010	0.0030	0.0014	167.5710
1	CH GLUNUK	CH KIPANG	1	150	4,200	OPC	3 x 300 mm ²		462	800	800	0.0870	0.1310	73.0000	0.3634	0.5502	306.6000	0.6605	225.0000	0.0029	0.0016	0.0024	0.0016	120.2716
1	CH GLUNUK	CH KIPANG	2	150	4,200	OPC	3 x 300 mm ²		462	800	800	0.0870	0.1310	73.0000	0.3634	0.5502	306.6000	0.6605	225.0000	0.0029	0.0016	0.0024	0.0016	120.2716
1	GLUNUK	NGARAS	1	150	38,040	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	5.2115	15.0867	109.5652	15.9614	225.0000	0.0709	0.0232	0.0671	0.0247	167.5710
1	GLUNUK	NGARAS	2	150	38,040	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	5.2115	15.0867	109.5652	15.9614	225.0000	0.0709	0.0232	0.0671	0.0247	167.5710
1	GLUNUK	PAROHAS	1	150	75,900	ACSR HAWK	2 x 477 MCM		1200	800	1250	0.0685	0.2045	2.7880	5.1992	15.5216	211.6992	16.3692	225.0000	0.0728	0.0231	0.0660	0.0247	335.1420
1	NGARAS	ASARIS	1	150	44,400	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	6.0828	17.6990	127.8720	18.6300	225.0000	0.0828	0.0270	0.0703	0.0268	167.5710
1	NGARAS	ASARIS	2	150	44,400	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	6.0828	17.6990	127.8720	18.6300	225.0000	0.0828	0.0270	0.0703	0.0268	167.5710
1	KAPALAS	ANSRIS	1	150	23,310	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	3.1935	9.2447	67.1328	9.7808	225.0000	0.0435	0.0142	0.0339	0.0151	167.5710
1	KAPALAS	ANSRIS	2	150	23,310	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	3.1935	9.2447	67.1328	9.7808	225.0000	0.0435	0.0142	0.0339	0.0151	167.5710
1	KAPALAS	GAYARAS	1	150	19,210	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	2.6318	7.6187	55.3248	8.0604	225.0000	0.0358	0.0117	0.0319	0.0125	167.5710
1	KAPALAS	GAYARAS	2	150	19,210	ACSR HAWK	477 MCM		645	800	800	0.1370	0.3966	2.8800	2.6318	7.6187	55.3248	8.0604	225.0000	0.0358	0.0117	0.0319	0.0125	167.5710
1	KAPALAS	NSDULAS	1	150	20,310	ACSR PARTRIDGE	135 mm ²		400	400	400	0.2136	0.4090	2.7650	4.4664	8.5313	57.6371	9.6297	225.0000	0.0428	0.0199	0.0379	0.0130	103.9200
1	KAPALAS	NSDULAS	1	150	0,960	OEO XLPE	3 x 240 mm ²		400	800	1250	0.0870	0.1320	73.0000	0.0835	0.1267	70.0800	0.1518	0.0700	15.1768	8.3380	12.6720	354.6334	0.6928
1	KAPALAS	BRITIS	1	150	38,170	TACSR	180 mm ²		400	800	1250	0.1370	0.3970	2.8800	5.2293	15.1535	109.9296	16.0304	225.0000	0.0712	0.0232	0.0673	0.0247	103.9200
1	KAPALAS	BRITIS	1	150	21,700	TACSR	180 mm ²		400	800	800	0.1370	0.3970	2.8800	2.9729	8.6149	62.4690	8.1134	225.0000	0.0405	0.0132	0.0333	0.0141	103.9200
1	BRITIS	PNGANS	1	150	28,010	TACSR	180 mm ²		400	800	1250	0.1370	0.3970	2.8800	3.6374	11.1200	80.6888	11.7635	225.0000	0.0523	0.0171	0.0484	0.0162	103.9200
1	GNVARS	SANURS	1	150	16,500	ACSR PARTRIDGE	135 mm ²		400	200	1000	0.1370	0.4090	2.7650	2.2441	6.7320	45.6390	7.5989	225.0000	0.0339	0.0157	0.0299	0.0163	103.9200
1	GNVARS	SANURS	2	150	16,500	ACSR HAWK	477 MCM		645	800	1000	0.1370	0.3970	2.8800	2.2441	6.7320	45.6390	7.5989	225.0000	0.0339	0.0157	0.0299	0.0163	103.9200
1	GNVARS	AMPRAS	1	150	33,760	ACSR HAWK	477 MCM		645	400	1000	0.1370	0.3966	2.8800	4.6251	13.3992	97.2268	14.1656	225.0000	0.0306	0.0206	0.0289	0.0106	167.5710
1	GNVARS	AMPRAS	2	150	33,760	ACSR HAWK	477 MCM		645	400	1000	0.1370	0.3966	2.8800	4.6251	13.3992	97.2268	14.1656	225.0000	0.0306	0.0206	0.0289	0.0106	167.5710
1	SAUNRS	PGANS	1	150	7,740	ACSR PARTRIDGE	135 mm ²		400	400	400	0.2136	0.4090	2.7650	1.6533	3.1579	21.4098	3.5945	225.0000	0.0158	0.0073	0.0140	0.0083	103.9200
1	SAUNRS	PGANS	2	150	7,740	ACSR HAWK	477 MCM		400	400	400	0.2136	0.4090	2.7650	1.6533	3.1579	21.4098	3.5945	225.0000	0.0158	0.0073	0.0140	0.0083	103.9200
1	NSDULAS	PGANS	1	150	13,410	ACSR PARTRIDGE	135 mm ²		400	200	200	0.1370	0.3970	2.8800	2.8680	3.0768	22.3200	3.2546	225.0000	0.0274	0.0047	0.0137	0.0083	103.9200
1	NSDULAS	PGANS	2	150	13,410	ACSR PARTRIDGE	135 mm ²		400	200	200	0.1370	0.3970	2.8800	2.8680	3.0768	22.3200	3.2546	225.0000	0.0274	0.0047	0.0137	0.0083	103.9200
1	BRITIS	PAROHAS	1	150	19,280	ACSR PARTRIDGE	135 mm ²		400	200	200	0.2136	0.4090	2.7650	4.1182	7.8662	53.3285	8.8790	225.0000	0.0356	0.0183	0.0330	0.0120	103.9200
1	BRITIS	PAROHAS	2	150	19,280	ACSR PARTRIDGE	135 mm ²		400	200	200	0.2136	0.4090	2.7650	4.1182	7.8662	53.3285	8.8790	225.0000	0.0356	0.0183	0.0330	0.0120	103.9200
1	KAPALAS	PRIANS	1	150	9,120	ACSR HAWK	477 MCM		645	400	1000	0.1370	0.3966	2.8800	1.2494	3.9170	26.2656	3.8267	225.0000	0.0170	0.0056	0.0161	0.0089	167.5710
1	PRIANS	PGANS	1	150	7,600	ACSR HAWK	477 MCM		645	400	1000	0.1370	0.3966	2.8800	1.0412	3.0142	21.8890	3.1899	225.0000	0.0142	0.0046	0.0134	0.0069	167.5710

Sub Region Bali

```

unit uMenu;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,
  Dialogs, ComCtrls, StdCtrls, ExtCtrls;

type
  TfrmMenu = class(TForm)
    Panel1: TPanel;
    btnNew: TButton;
    btnOpen: TButton;
    btnExit: TButton;
    StatusBar1: TStatusBar;
    Panel2: TPanel;
    OpenDialog1: TOpenDialog;
    procedure btnExitClick(Sender: TObject);
    procedure btnNewClick(Sender: TObject);
    procedure btnOpenClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmMenu: TfrmMenu;

implementation

uses uInputLFChild, uComplex, uUtils, uLoadflow, uHasil;

{$R *.dfm}

procedure TfrmMenu.btnExitClick(Sender: TObject);
begin
  Application.Terminate;
end;

procedure TfrmMenu.btnNewClick(Sender: TObject);
begin
  try
    if frmInput=nil then
      begin
        frmInput:=TfrmInputLFChild.Create(Application);
      end;
    frmInput.Caption:='Input Data';
    frmInput.btnNext.Caption:='&Save';
    frmInput.ShowModal;
  finally
    frmInput.Free;
  end;
end;

procedure TfrmMenu.btnOpenClick(Sender: TObject);
var NamaFile, Nama:sring;
    output:TextFile;
    i, j, Typ, dari, ke, Nbus, Nsal, Param, Ngen, NCable:integer;
    Cap, absV, sudV, Pg, Qg, PL, QL, CapSal, Pmin, Pmax, Harga, Length:double;
    R, X, Lc, Tr, Tu, Su, VKonst, PKonst, Pbase, Vbase, dia:double;
begin
  try
    if OpenDialog1.Execute then
      begin
        NamaFile:=OpenDialog1.FileName;
        AssignFile(output, NamaFile);
        Reset(output);
        Readln(output, Nbus);
        Readln(output, Nsal);

```

```

Readln(output, Vbase);
Readln(output, VKonst);
Readln(output, Pbase);
Readln(output, PKonst);
Readln(output, param);
gParamLF.Vbase:=Vbase;
gParamLF.VKonst:=VKonst;
gParamLF.Pbase:=Pbase;
gParamLF.PKonst:=PKonst;
if Param=1 then
begin
  gParamLF.ParamBranch:=-pbPu;
end
else if Param=2 then
begin
  gParamLF.ParamBranch:=pbOhm;
end;
gParamLF.MaxIterasi:=15;
gParamLF.Toleransi:=0.0001;
try
  frmInput:=TfrmInputLFChild.Create(Application);
  frmInput.edtNbus.Text:=IntToStr(Nbus);
  frmInput.edtNsal.Text:=IntToStr(Nsal);
  frmInput.edtVbase.Text:=FloatToStr(Vbase);
  if VKonst=1 then
  begin
    frmInput.cmbVKonst.Text:='V';
  end
  else if VKonst=1000 then
  begin
    frmInput.cmbVKonst.Text:='kV';
  end
  else if VKonst=1000000 then
  begin
    frmInput.cmbVKonst.Text:='MV';
  end;
  frmInput.edtPbase.Text:=FloatToStr(Pbase);
  if PKonst=1 then
  begin
    frmInput.cmbPKonst.Text:='VA';
    frmInput.fgBus.Cells[3,0]:='Pg (W)';
    frmInput.fgBus.Cells[4,0]:='Qg (VAR)';
    frmInput.fgBus.Cells[5,0]:='PL (W)';
    frmInput.fgBus.Cells[6,0]:='QL (VAR)';
    frmInput.fgBranch.Cells[9,0]:='Kap (VA)';
  end
  else if PKonst=1000 then
  begin
    frmInput.cmbPKonst.Text:='kVA';
    frmInput.fgBus.Cells[3,0]:='Pg (kW)';
    frmInput.fgBus.Cells[4,0]:='Qg (kVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (kW)';
    frmInput.fgBus.Cells[6,0]:='QL (kVAR)';
    frmInput.fgBranch.Cells[9,0]:='Kap (kVA)';
  end
  else if PKonst=1000000 then
  begin
    frmInput.cmbPKonst.Text:='MVA';
    frmInput.fgBus.Cells[3,0]:='Pg (MW)';
    frmInput.fgBus.Cells[4,0]:='Qg (MVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (MW)';
    frmInput.fgBus.Cells[6,0]:='QL (MVAR)';
    frmInput.fgBranch.Cells[9,0]:='Kap (MVA)';
  end;
  if param=1 then
  begin
    frmInput.cmbParam.Text:='pu';
    frmInput.fgBranch.Cells[3,0]:='R (pu)';
    frmInput.fgBranch.Cells[4,0]:='X (pu)';
    frmInput.fgBranch.Cells[5,0]:='Lc (pu)';
    frmInput.fgBus.Cells[7,0]:='Cap (pu)';
  end
end

```

```

else if param=2 then
begin
frmInput.cmbParam.Text:='ohm';
frmInput.fgBranch.Cells[3,0]:='R (ohm)';
frmInput.fgBranch.Cells[4,0]:='X (ohm)';
frmInput.fgBranch.Cells[5,0]:='Lc (ohm)';
frmInput.fgBus.Cells[7,0]:='Cap (ohm)';
end;
SetLength(gBus,Nbus);
for i:=0 to Nbus-1 do
begin
Readln(output,absV,sudV,Pg,Qg,PL,QL,Cap,Typ);
gBus[i].absV:=absV;
gBus[i].sudV:=sudV;
gBus[i].Pgen:=Pg;
gBus[i].Qgen:=Qg;
gBus[i].PL:=PL;
gBus[i].QL:=QL;
gBus[i].Cap:=Cap;
gBus[i].typeBus:=Typ;
frmInput.fgBus.Cells[0,i+1]:=IntToStr(i+1);
frmInput.fgBus.Cells[1,i+1]:=FloatToStr(absV);
frmInput.fgBus.Cells[2,i+1]:=FloatToStr(sudV);
frmInput.fgBus.Cells[3,i+1]:=FloatToStr(Pg);
frmInput.fgBus.Cells[4,i+1]:=FloatToStr(Qg);
frmInput.fgBus.Cells[5,i+1]:=FloatToStr(PL);
frmInput.fgBus.Cells[6,i+1]:=FloatToStr(QL);
frmInput.fgBus.Cells[7,i+1]:=FloatToStr(Cap);
frmInput.fgBus.Cells[8,i+1]:=IntToStr(typ);
end;
SetLength(gBranch,Nsal);
for i:=0 to Nsal-1 do
begin
Readln(output,dari,ke,R,X,Lc,Tr,Tu,Su,CapSal);
gBranch[i].dari:=dari;
gBranch[i].ke:=ke;
gBranch[i].R:=R;
gBranch[i].X:=X;
gBranch[i].Lc:=Lc;
gBranch[i].Tr:=Tr;
gBranch[i].Tu:=Tu;
gBranch[i].Su:=Su;
gBranch[i].KapSal:=CapSal;
frmInput.fgBranch.Cells[0,i+1]:=IntToStr(i+1);
frmInput.fgBranch.Cells[1,i+1]:=IntToStr(dari);
frmInput.fgBranch.Cells[2,i+1]:=IntToStr(ke);
frmInput.fgBranch.Cells[3,i+1]:=FloatToStr(R);
frmInput.fgBranch.Cells[4,i+1]:=FloatToStr(X);
frmInput.fgBranch.Cells[5,i+1]:=FloatToStr(Lc);
frmInput.fgBranch.Cells[6,i+1]:=FloatToStr(Tr);
frmInput.fgBranch.Cells[7,i+1]:=FloatToStr(Tu);
frmInput.fgBranch.Cells[8,i+1]:=FloatToStr(Su);
frmInput.fgBranch.Cells[9,i+1]:=FloatToStr(CapSal);
end;
Readln(output,Ngen);
if Ngen<>0 then
begin
frmInput.fgGen.PowCount:=Ngen+1;
SetLength(gGenLF,Ngen);
for i:=0 to Ngen-1 do
begin
Readln(output,dari,R,X,Lc,Tr,Tu,Su,CapSal,Pmin,Pmax);
gGenLF[i].bus:=dari;
gGenLF[i].Qmin:=R;
gGenLF[i].Qmax:=X;
gGenLF[i].a2:=Lc;
gGenLF[i].a1:=Tr;
gGenLF[i].a0:=Tu;
gGenLF[i].FixCost:=Su;
gGenLF[i].VarCost:=CapSal;
gGenLF[i].Pmin:=Pmin;
gGenLF[i].Pmax:=Pmax;

```

```

    frmInput.fgGen.Cells[0,i+1]:=IntToStr(i+1);
    frmInput.fgGen.Cells[1,i+1]:=IntToStr(gGenLF[i].bus);
    frmInput.fgGen.Cells[2,i+1]:=RealToStr(gGenLF[i].Qmin,2);
    frmInput.fgGen.Cells[3,i+1]:=RealToStr(gGenLF[i].Qmax,2);
    frmInput.fgGen.Cells[4,i+1]:=RealToStr(gGenLF[i].a2,5);
    frmInput.fgGen.Cells[5,i+1]:=RealToStr(gGenLF[i].a1,5);
    frmInput.fgGen.Cells[6,i+1]:=RealToStr(gGenLF[i].a0,5);
    frmInput.fgGen.Cells[7,i+1]:=RealToStr(gGenLF[i].FixCost,2);
    frmInput.fgGen.Cells[8,i+1]:=RealToStr(gGenLF[i].VarCost,2);
    frmInput.fgGen.Cells[9,i+1]:=RealToStr(gGenLF[i].Pmin,2);
    frmInput.fgGen.Cells[10,i+1]:=RealToStr(gGenLF[i].Pmax,2);
end;
end
else
begin
    frmInput.fgGen.RowCount:=2;
end;
CloseFile(output);
frmInput.Caption:='Tampilan Data';
frmInput.btnNext.Caption:='&Next';
frmInput.ShowModal;
finally
    frmInput.Free;
end;
end;
except
    MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);
end;
end;
end.

```



```
mas11: TForm1
200
44
: 610
= 451
cons = [biSystemMenu, biMinimize]
. = 'Hasil Loadflow'
: clBtnFace
arset = DEFAULT_CHARSET
lor = clWindowText
ight = -11
me = 'MS Sans Serif'
yle = []
teOrder = False
n = poScreenCenter
e = FormCreate
erInch = 96
ght = 13
PageControl1: TPageControl
= 8
: 8
. = 585
t = 345
ePage = TabSheet3
der = 0
t TabSheet3: TTabSheet
tion = 'LF Awal'
geIndex = 2
ect fgBus: TStringGrid
eft = 0
op = 0
idth = 577
eight = 313
olCount = 9
efaultColWidth = 30
efaultRowHeight = 18
owCount = 2
ptions = [goFixedVertLine, goFixedHorzLine, goVertLine, goHorzLine, goRangeSelect, goEc
abOrder = 0
olWidths = (
30
60
61
67
65
61
60
61
61)

t TabSheet4: TTabSheet
tion = 'Aliran Daya'
geIndex = 3
ect fgBranch: TStringGrid
eft = 0
op = 4
idth = 577
eight = 309
olCount = 13
efaultColWidth = 35
efaultRowHeight = 18
owCount = 2
abOrder = 0
olWidths = (
35
35
35
72
73
71
72
```

35
72
72
72
71)

```
t TabSheet9: TTabSheet
tion = 'Summary'
geIndex = 8
ect GroupBox6: TGroupBox
eft = 8
op = 8
idth = 369
eight = 153
aption = 'Summary Loadflow'
abOrder = 0
bject Label8: TLabel
Left = 16
Top = 24
Width = 104
Height = 13
Caption = 'Jumlah Pembangkitan'
nd
bject Label9: TLabel
Left = 16
Top = 48
Width = 99
Height = 13
Caption = 'Jumlah Pembebanan'
nd
bject Label10: TLabel
Left = 16
Top = 72
Width = 83
Height = 13
Caption = 'Jumlah Rugi-Rugi'
nd
bject Label11: TLabel
Left = 16
Top = 96
Width = 28
Height = 13
Caption = 'Iterasi'
nd
bject Label12: TLabel
Left = 16
Top = 120
Width = 66
Height = 13
Caption = 'Waktu Hitung'
nd
bject lblGen: TLabel
Left = 304
Top = 24
Width = 30
Height = 13
Caption = 'lblGen'
nd
bject lblLoad: TLabel
Left = 304
Top = 48
Width = 34
Height = 13
Caption = 'lblLoad'
nd
bject lblLoss: TLabel
Left = 304
Top = 72
Width = 32
Height = 13
```

Caption = 'IDILoss

nd
object edtSumGen: TEdit
Left = 128
Top = 24
Width = 169
Height = 21
TabOrder = 0

nd
object edtSumLoad: TEdit
Left = 128
Top = 48
Width = 169
Height = 21
TabOrder = 1

nd
object edtSumLoss: TEdit
Left = 128
Top = 72
Width = 169
Height = 21
TabOrder = 2

nd
object edtIterasi: TEdit
Left = 128
Top = 96
Width = 57
Height = 21
TabOrder = 3

nd
object edtTime: TEdit
Left = 128
Top = 120
Width = 169
Height = 21
TabOrder = 4

nd
: TabSheet1: TTabSheet
Caption = 'Parameter'

TabIndex = 4
object GroupBox4: TGroupBox
Left = 256
Top = 8
Width = 233
Height = 257
Caption = 'Parameter Objective Function'

TabOrder = 0
object Label21: TLabel
Left = 16
Top = 24
Width = 64
Height = 13
Caption = 'Konstanta Kp'

nd
object Label22: TLabel
Left = 16
Top = 48
Width = 64
Height = 13
Caption = 'Konstanta Kq'

nd
object Label23: TLabel
Left = 16
Top = 72
Width = 64
Height = 13
Caption = 'Konstanta Kv'

nd
object Label24: TLabel
Left = 16

```
Top = 120
Width = 100
Height = 13
Caption = 'Batas Tegangan (pu)'
nd
Object Label25: TLabel
Left = 16
Top = 144
Width = 77
Height = 13
Caption = 'Batas Tap Trafo'
nd
Object Label29: TLabel
Left = 16
Top = 168
Width = 72
Height = 13
Caption = 'Step Tap Trafo'
nd
Object Label30: TLabel
Left = 16
Top = 96
Width = 63
Height = 13
Caption = 'Konstanta Ks'
nd
Object Label31: TLabel
Left = 16
Top = 192
Width = 64
Height = 13
Caption = 'Konstanta Ka'
nd
Object edtKp: TEdit
Left = 136
Top = 24
Width = 65
Height = 21
TabOrder = 0
nd
Object edtKq: TEdit
Left = 136
Top = 48
Width = 65
Height = 21
TabOrder = 1
nd
Object edtKv: TEdit
Left = 136
Top = 72
Width = 65
Height = 21
TabOrder = 2
nd
Object edtVmin: TEdit
Left = 136
Top = 120
Width = 33
Height = 21
TabOrder = 3
nd
Object edtVmax: TEdit
Left = 168
Top = 120
Width = 33
Height = 21
TabOrder = 4
nd
Object edtTapTrafoMin: TEdit
Left = 136
Top = 144
Width = 33
```

```
Height = 21
TabOrder = 5
nd
object edtTapTrafoMax: TEdit
Left = 168
Top = 144
Width = 33
Height = 21
TabOrder = 6
nd
object edtStepTrafo: TEdit
Left = 136
Top = 168
Width = 65
Height = 21
TabOrder = 7
nd
object edtKs: TEdit
Left = 136
Top = 96
Width = 65
Height = 21
TabOrder = 8
nd
object edtKa: TEdit
Left = 136
Top = 192
Width = 65
Height = 21
TabOrder = 9
nd

object GroupBox2: TGroupBox
Left = 8
Top = 8
Width = 233
Height = 257
Caption = 'Parameter Improve Evolutionary Programming'
TabOrder = 1
object Label15: TLabel
Left = 16
Top = 24
Width = 95
Height = 13
Caption = 'Jumlah Pendinginan'
nd
object Label16: TLabel
Left = 16
Top = 48
Width = 76
Height = 13
Caption = 'Jumlah Populasi'
nd
object Label17: TLabel
Left = 16
Top = 72
Width = 69
Height = 13
Caption = 'Jumlah Length'
nd
object Label18: TLabel
Left = 16
Top = 96
Width = 98
Height = 13
Caption = 'Jumlah Sub Populasi'
nd
object Label19: TLabel
Left = 16
Top = 120
Width = 88
Height = 13
```

```
Caption = 'PROBABILITAS MUTASI'
nd
bject Label20: TLabel
  Left = 16
  Top = 192
  Width = 3
  Height = 13
nd
bject Label32: TLabel
  Left = 16
  Top = 144
  Width = 61
  Height = 13
  Caption = 'Konstanta Pr'
nd
bject Label33: TLabel
  Left = 16
  Top = 168
  Width = 83
  Height = 13
  Caption = 'Konstanta Lamda'
nd
bject Label34: TLabel
  Left = 16
  Top = 192
  Width = 58
  Height = 13
  Caption = 'Konstanta A'
nd
bject edtMaxGen: TEdit
  Left = 136
  Top = 24
  Width = 65
  Height = 21
  TabOrder = 0
nd
bject edtPopSize: TEdit
  Left = 136
  Top = 48
  Width = 65
  Height = 21
  TabOrder = 1
nd
bject edtLength: TEdit
  Left = 136
  Top = 72
  Width = 65
  Height = 21
  Color = clScrollBar
  ReadOnly = True
  TabOrder = 2
nd
bject edtPmutasi: TEdit
  Left = 136
  Top = 120
  Width = 65
  Height = 21
  TabOrder = 3
nd
bject edtPr: TEdit
  Left = 136
  Top = 144
  Width = 65
  Height = 21
  TabOrder = 4
nd
bject edtLamda: TEdit
  Left = 136
  Top = 168
  Width = 65
  Height = 21
  TabOrder = 5
```

```
object edtA: TEdit
  Left = 136
  Top = 192
  Width = 65
  Height = 21
  TabOrder = 6
nd
object cmbSubPop: TComboBox
  Left = 136
  Top = 96
  Width = 65
  Height = 21
  ItemHeight = 13
  TabOrder = 7
  Items.Strings = (
    '2'
    '4'
    '8')
nd
object btnUseDefault: TButton
  Left = 8
  Top = 280
  Width = 89
  Height = 25
  Caption = '&Use Default'
  TabOrder = 2
 OnClick = btnUseDefaultClick
nd
object TabSheet2: TTabSheet
  Caption = 'Hasil Program'
  PageIndex = 4
object GroupBox3: TGroupBox
  Left = 8
  Top = 8
  Width = 257
  Height = 297
  Caption = 'Hasil Setting Program'
  TabOrder = 0
object fgHasil: TStringGrid
  Left = 8
  Top = 24
  Width = 241
  Height = 257
  ColCount = 3
  DefaultColWidth = 35
  DefaultRowHeight = 18
  RowCount = 2
  TabOrder = 0
  ColWidths = (
    35
    38
    74)
nd
object GroupBox5: TGroupBox
  Left = 272
  Top = 8
  Width = 297
  Height = 105
  Caption = 'Rekapitulasi Biaya'
  TabOrder = 1
object Label16: TLabel
  Left = 16
  Top = 24
  Width = 84
  Height = 13
  Caption = 'Sebelum Optimasi'
nd
object Label17: TLabel
```

```
Top = 90
Width = 28
Height = 13
Caption = 'Iterasi'
nd
object Label5: TLabel
Left = 16
Top = 120
Width = 66
Height = 13
Caption = 'Waktu Hitung'
nd
object lblGen2: TLabel
Left = 304
Top = 24
Width = 30
Height = 13
Caption = 'lblGen'
nd
object lblLoad2: TLabel
Left = 304
Top = 48
Width = 34
Height = 13
Caption = 'lblLoad'
nd
object lblLoss2: TLabel
Left = 304
Top = 72
Width = 32
Height = 13
Caption = 'lblLoss'
nd
object edtSumGen2: TEdit
Left = 128
Top = 24
Width = 169
Height = 21
TabOrder = 0
nd
object edtSumLoad2: TEdit
Left = 128
Top = 48
Width = 169
Height = 21
TabOrder = 1
nd
object edtSumLoss2: TEdit
Left = 128
Top = 72
Width = 169
Height = 21
TabOrder = 2
nd
object edtIterasi2: TEdit
Left = 128
Top = 96
Width = 57
Height = 21
TabOrder = 3
nd
object edtTime2: TEdit
Left = 128
Top = 120
Width = 169
Height = 21
TabOrder = 4
nd
TabSheet8: TTabSheet
Caption = 'Grafik V'
```



```
geIndex = 0
ect Chart1: TChart
eft = 8
op = 8
idth = 561
eight = 305
ackWall.Brush.Color = clWhite
ackWall.Brush.Style = bsClear
radient.Visible = True
itle.Text.Strings = (
'Grafik Perbandingan Tegangan ')
ottomAxis.Title.Caption = 'Bus'
eftAxis.Automatic = False
eftAxis.AutomaticMaximum = False
eftAxis.AutomaticMinimum = False
eftAxis.Maximum = 1.10000000000000000000
eftAxis.Minimum = 0.90000000000000000000
eftAxis.Title.Caption = 'Tegangan (pu)'
iew3D = False
abOrder = 0
bject Series1: TLineSeries
Marks.ArrowLength = 8
Marks.Visible = False
SeriesColor = clBlue
Title = 'Sebelum'
Pointer.InflateMargins = True
Pointer.Style = psRectangle
Pointer.Visible = False
XValues.DateTime = False
XValues.Name = 'X'
XValues.Multiplier = 1.00000000000000000000
XValues.Order = loAscending
YValues.DateTime = False
YValues.Name = 'Y'
YValues.Multiplier = 1.00000000000000000000
YValues.Order = loNone
nd
bject Series2: TLineSeries
Marks.ArrowLength = 8
Marks.Visible = False
SeriesColor = clRed
Title = 'Sesudah'
Pointer.InflateMargins = True
Pointer.Style = psRectangle
Pointer.Visible = False
XValues.DateTime = False
XValues.Name = 'X'
XValues.Multiplier = 1.00000000000000000000
XValues.Order = loAscending
YValues.DateTime = False
YValues.Name = 'Y'
YValues.Multiplier = 1.00000000000000000000
YValues.Order = loNone
nd
l
ct TabSheet10: TTabSheet
tion = 'Grafik IEP'
geIndex = 9
ect Chart2: TChart
eft = 8
op = 8
idth = 561
eight = 305
ackWall.Brush.Color = clWhite
ackWall.Brush.Style = bsClear
radient.Visible = True
itle.Text.Strings = (
'Grafik Improved Evolutionary Programming')
ottomAxis.Title.Caption = 'Iterasi'
eftAxis.Title.Caption = 'Fitness'
iew3D = False
```

```
Order = 0
Object Series3: TLineSeries
Marks.ArrowLength = 8
Marks.Visible = False
SeriesColor = clBlack
Title = 'Min'
Pointer.InflateMargins = True
Pointer.Style = psRectangle
Pointer.Visible = False
XValues.DateTime = False
XValues.Name = 'X'
XValues.Multiplier = 1.00000000000000000000
XValues.Order = loAscending
YValues.DateTime = False
YValues.Name = 'Y'
YValues.Multiplier = 1.00000000000000000000
YValues.Order = loNone
end
```

```
Object Series4: TLineSeries
Marks.ArrowLength = 8
Marks.Visible = False
SeriesColor = 33023
Title = 'Avg'
Pointer.InflateMargins = True
Pointer.Style = psRectangle
Pointer.Visible = False
XValues.DateTime = False
XValues.Name = 'X'
XValues.Multiplier = 1.00000000000000000000
XValues.Order = loAscending
YValues.DateTime = False
YValues.Name = 'Y'
YValues.Multiplier = 1.00000000000000000000
YValues.Order = loNone
end
```

```
Object Series5: TLineSeries
Marks.ArrowLength = 8
Marks.Visible = False
SeriesColor = clBlue
Title = 'Max'
Pointer.InflateMargins = True
Pointer.Style = psRectangle
Pointer.Visible = False
XValues.DateTime = False
XValues.Name = 'X'
XValues.Multiplier = 1.00000000000000000000
XValues.Order = loAscending
YValues.DateTime = False
YValues.Name = 'Y'
YValues.Multiplier = 1.00000000000000000000
YValues.Order = loNone
end
```

```
Panel1: TPanel
Width = 800
Height = 352
Caption = '585'
Color = 57
Order = 1
OnClose = btnCloseClick
btnClose: TButton
Caption = '512'
Color = 16
Height = 65
Width = 25
Caption = '&Close'
Order = 0
OnClick = btnCloseClick

btnLFAwal: TButton
Caption = '440'
```

```
Left = 16
Top = 48
Width = 85
Height = 13
Caption = 'Sesudah Optimasi'
nd
object Label13: TLabel
Left = 16
Top = 72
Width = 73
Height = 13
Caption = 'Selisih Optimasi'
nd
object lblLoss1: TLabel
Left = 224
Top = 24
Width = 38
Height = 13
Caption = 'lblLoss1'
nd
object Label14: TLabel
Left = 224
Top = 48
Width = 38
Height = 13
Caption = 'lblLoss2'
nd
object lblLoss3: TLabel
Left = 224
Top = 72
Width = 38
Height = 13
Caption = 'lblLoss3'
nd
object edtSebelumOpt: TEdit
Left = 128
Top = 24
Width = 89
Height = 21
TabOrder = 0
nd
object edtSesudahOpt: TEdit
Left = 128
Top = 48
Width = 89
Height = 21
TabOrder = 1
nd
object edtSelisihOpt: TEdit
Left = 128
Top = 72
Width = 89
Height = 21
TabOrder = 2
nd
object GroupBox7: TGroupBox
Left = 272
Top = 120
Width = 297
Height = 105
Caption = 'Rekapitulasi Losses'
TabOrder = 2
object Label26: TLabel
Left = 16
Top = 24
Width = 84
Height = 13
Caption = 'Sebelum Optimasi'
nd
object Label27: TLabel
Left = 16
```

```
Top = 40
Width = 85
Height = 13
Caption = 'Sesudah Optimasi'
id
Object Label28: TLabel
Left = 16
Top = 72
Width = 73
Height = 13
Caption = 'Selisih Optimasi'
id
Object lblLoss21: TLabel
Left = 224
Top = 24
Width = 38
Height = 13
Caption = 'lblLoss1'
id
Object lblLoss22: TLabel
Left = 224
Top = 48
Width = 38
Height = 13
Caption = 'lblLoss2'
id
Object lblLoss23: TLabel
Left = 224
Top = 72
Width = 38
Height = 13
Caption = 'lblLoss3'
id
Object edtBaseLoss: TEdit
Left = 128
Top = 24
Width = 89
Height = 21
TabOrder = 0
id
Object edtOptLoss: TEdit
Left = 128
Top = 48
Width = 89
Height = 21
TabOrder = 1
id
Object edtSelisihLoss: TEdit
Left = 128
Top = 72
Width = 89
Height = 21
TabOrder = 2
id
TabSheet5: TTabSheet
Caption = 'LF Akhir'
TabIndex = 5
Object fgBus2: TStringGrid
Left = 0
Top = 0
Width = 577
Height = 313
ColCount = 9
DefaultColWidth = 30
DefaultRowHeight = 18
RowCount = 2
Options = [goFixedVertLine, goFixedHorzLine, goVertLine, goHorzLine, goRangeSelect, goEd
TabOrder = 0
ColWidths = (
30
```

61
67
65
61
60
61
61)

```
t TabSheet6: TTabSheet
Caption = 'Aliran Daya'
PageIndex = 6
Object fgBranch2: TStringGrid
Left = 0
Top = 4
Width = 577
Height = 309
ColCount = 13
DefaultColWidth = 35
DefaultRowHeight = 18
RowCount = 2
TabOrder = 0
ColWidths = (
  35
  35
  35
  72
  73
  71
  72
  35
  35
  72
  72
  72
  72
  71)
```

```
t TabSheet7: TTabSheet
Caption = 'Summary'
PageIndex = 7
Object GroupBox1: TGroupBox
Left = 8
Top = 8
Width = 369
Height = 153
Caption = 'Summary Loadflow'
TabOrder = 0
Object Label1: TLabel
Left = 16
Top = 24
Width = 104
Height = 13
Caption = 'Jumlah Pembangkitan'
Object Label2: TLabel
Left = 16
Top = 48
Width = 99
Height = 13
Caption = 'Jumlah Pembebanan'
Object Label3: TLabel
Left = 16
Top = 72
Width = 83
Height = 13
Caption = 'Jumlah Rugi-Rugi'
Object Label4: TLabel
Left = 16
```

```
    10  
    th = 65  
    ght = 25  
    tion = '&LF Awal'  
    Order = 1  
    lick = btnLFAwalClick
```

```
t btnHitung: TButton  
t = 360  
    = 16  
    th = 73  
    ght = 25  
    tion = '&Hitung'  
    oled = False  
    Order = 2  
    lick = btnHitungClick
```

```
t pbGen: TProgressBar  
t = 8  
    = 16  
    th = 345  
    ght = 17  
    Order = 3
```

```
nAbout: TfrmAbout
192
14
323
= 270
cons = [biSystemMenu, biMinimize]
= 'About'
clBtnFace
charset = DEFAULT_CHARSET
color = clWindowText
height = -11
name = 'MS Sans Serif'
style = []
zOrder = False
x = poScreenCenter
width = 96
height = 13
```

```
Input1: TInput1
307
32
596
= 438
cons = [biSystemMenu, biMinimize]
  clBtnFace
Charset = DEFAULT_CHARSET
Color = clWindowText
Height = -11
Name = 'MS Sans Serif'
Style = []
TabOrder = False
Top = poScreenCenter
OnOpen = FormCreate
WidthInch = 96
Height = 13
PageControl1: TPageControl
  = 8
  = 569
  = 329
Page = TabSheet1
Order = 0
TabSheet1: TTabSheet
  Caption = 'General'
Object Label1: TLabel
  Left = 16
  Top = 16
  Width = 54
  Height = 13
  Caption = 'Jumlah Bus'
Object Label2: TLabel
  Left = 16
  Top = 40
  Width = 72
  Height = 13
  Caption = 'Jumlah Saluran'
Object Label3: TLabel
  Left = 16
  Top = 64
  Width = 74
  Height = 13
  Caption = 'Tegangan Dasar'
Object Label4: TLabel
  Left = 16
  Top = 88
  Width = 56
  Height = 13
  Caption = 'Daya Dasar'
Object Label5: TLabel
  Left = 16
  Top = 112
  Width = 87
  Height = 13
  Caption = 'Parameter Saluran'
Object edtNbus: TEdit
  Left = 112
  Top = 16
  Width = 57
  Height = 21
  TabOrder = 0
  OnChange = edtNbusChange
Object edtNsal: TEdit
  Left = 112
  Top = 40
```



```

    Left = 57
    Height = 21
    TabOrder = 1
    OnChange = edtNsalChange

    TEdit edtVbase: TEdit
    Left = 112
    Top = 64
    Width = 57
    Height = 21
    TabOrder = 2

    TEdit edtPbase: TEdit
    Left = 112
    Top = 88
    Width = 57
    Height = 21
    TabOrder = 3

    TComboBox cmbParam: TComboBox
    Left = 112
    Top = 112
    Width = 57
    Height = 21
    ItemHeight = 13
    TabOrder = 4
    OnChange = cmbParamChange
    Items.Strings = (
      'pu'
      'ohm')

    TComboBox cmbVKonst: TComboBox
    Left = 176
    Top = 64
    Width = 65
    Height = 21
    ItemHeight = 13
    TabOrder = 5
    Items.Strings = (
      'V'
      'kV'
      'MV')

    TComboBox cmbPkonst: TComboBox
    Left = 176
    Top = 88
    Width = 65
    Height = 21
    ItemHeight = 13
    TabOrder = 6
    OnChange = cmbPkonstChange
    Items.Strings = (
      'VA'
      'kVA'
      'MVA')

    TTabSheet2: TTabSheet
    Caption = 'Data Bus'
    PageIndex = 1

    TStringGrid fgBus: TStringGrid
    Left = 0
    Top = 0
    Width = 561
    Height = 297
    ColCount = 9
    DefaultColWidth = 30
    DefaultRowHeight = 18
    RowCount = 2
    Options = [goFixedVertLine, goFixedHorzLine, goVertLine, goHorzLine, goRangeSelect, goEd
    TabOrder = 0
    ColWidths = (
```

```
60
61
67
65
61
60
61
61)

: TabSheet3: TTabSheet
Caption = 'Data Saluran'
PageIndex = 2
Object fgBranch: TStringGrid
Left = 0
Top = 0
Width = 561
Height = 297
ColCount = 12
DefaultColWidth = 30
DefaultRowHeight = 18
RowCount = 2
Options = [goFixedVertLine, goFixedHorzLine, goVertLine, goHorzLine, goRangeSelect, goEd
TabOrder = 0
ColWidths = (
30
40
40
61
61
61
61
61
61
61
61
61
67)
```

```
: TabSheet4: TTabSheet
Caption = 'Data Generator'
PageIndex = 3
Object fgGen: TStringGrid
Left = 0
Top = 0
Width = 553
Height = 297
ColCount = 12
DefaultColWidth = 30
DefaultRowHeight = 18
RowCount = 2
Options = [goFixedVertLine, goFixedHorzLine, goVertLine, goHorzLine, goRangeSelect, goEd
TabOrder = 0
ColWidths = (
30
35
71
73
61
62
59
60
60
64
62
162)
```

```
Panel1: TPanel
= 8
```

```
Order = 1  
= 569  
= 57  
Order = 1  
: btnClose: TButton  
= 496  
= 16  
Width = 65  
Height = 25  
Caption = '&Close'  
Order = 0  
OnClick = btnCloseClick
```

```
: btnNext: TButton  
= 424  
= 16  
Width = 65  
Height = 25  
Order = 1  
OnClick = btnNextClick
```

```
SaveDialog1: TSaveDialog  
Filter = 'Text File (*.txt)|*.txt|Data File (*.dat)|*.dat'  
= 296  
Width = 352
```

uLoadflowBU

```
uLoadflow;  
face  
uUtils,uComplex;  
ParamBranch=(pbOhm,pbPu);  
ParamLF=record  
  MaxIterasi,Iterasi:byte;  
  Vbase,Vkonst,Pbase,Pkonst,Zbase,Ibase,Toleransi:double;  
  ParamBranch:TParamBranch;  
  PmGen,SumLoad,SumLoss:TCmplx;  
;  
TBus=record  
  Vs,sudV,Pgen,Qgen,PL,QL,Cap:double;  
  nBus:byte;  
;  
TBusArr1=array of TBus;  
TBranch=record  
  rI,ke:integer;  
  X,Lc,Tr,Tu,Su,KapSal,Length,JenisCable:double;  
  Sji,Aij,Aji:TCmplx;  
;  
TBranchArr1=array of TBranch;  
TGenLF=record  
  n:integer;  
  Qin,Qmax,a2,a1,a0,FixCost,VarCost,Pmin,Pmax:double;  
;  
TGenLFArr1=array of TGenLF;  
TSwitch=record  
  rI,ke:integer;  
;  
TSwitchArr1=array of TSwitch;  
function CalcCostGenLF(const rGen:TGenLF;  
  const rPgen:double):double;  
procedure DecodeCommDataToLFData(const rBus:TBusArr1;  
  var rNbus,rNsal:integer;  
  var rV,rSg,rSL:CxArr1;  
  var rCap:dArr1;
```

uLoadflowBU

```
var rTypBus:iArr1;
const rBranch:TBranchArr1;
var rZ,rTp:CxArr2;
var rLc,rTr:dArr2);
procedure DecodeData(var rParamLF:TParamLF;
var rNbus:integer;
var rV,rSg,rSL:CxArr1;
var rCap:dArr1;
var rTypBus:iArr1;
var rZ,rTp:CxArr2;
var rLc,rTr:dArr2);
function FindSumGen(const rNbus:integer;
const rTypBus:iArr1):integer;
procedure Admitansi(const rNbus:integer;
const rZ,rTp:CxArr2;
const rLc,rTr:dArr2;
const rCap:dArr1;
var rY:CxArr2);
function RecToPolar(const rData:CxArr1):CxArr1;
function PolarToRec(const rData:CxArr1):CxArr1;
function MaxDataArray(const rData:dArr1):double;
procedure AliranDaya(const rNbus:integer;const rV:CxArr1;
const rY:CxArr2;
const rLc:dArr2;
var rAlirS:CxArr2);
procedure DayaGen(const rNbus:integer;
const rV,rSL:CxArr1;
const rY:CxArr2;
const rTyp:iArr1;
var rSg:CxArr1);
procedure DayaSlack(const rNbus:integer;
const rAlirS:CxArr2;
const rTyp:iArr1;
const rSL:CxArr1;
var rSg:CxArr1);
procedure ArusBranch(const rNbus:integer;
const rV:CxArr1;
const rLc:dArr2;
const rY:CxArr2;
var rArus:CxArr2);
procedure UpdateAkhir(const rNbus,rNsal:integer;
var rParamLF:TParamLF;
const rV,rSg,rSL:CxArr1;
const rAlir,rArus:CxArr2;
var rBus:TBusArr1;
var rBranch:TBranchArr1);overload;
procedure UpdateAkhir(const rNbus:integer;
var rParamLF:TParamLF;
var rSg,rSL:CxArr1;
var rAlir,rArus:CxArr2);overload;
```

uLoadflowBU

```
function FindVarControl(var rBus:TBusArr1):iArr1;overload;  
function FindVarControl(var rSg:CxArr1;  
var rTypeBus:iArr1):iArr1;overload;  
function FindBatasControl(var rBus:TBusArr1;  
var rParamLF:TParamLF):TBatasArr1;
```

```
Bus:TBusArr1;  
Branch:TBranchArr1;  
ParamLF:TParamLF;  
GenLF:TGenLFArr1;
```

Implementation

```
function CalcCostGenLF(const rGen:TGenLF;  
const rPgen:double):double;
```

```
Result:=0;  
if rPgen>0 then  
begin  
Result:=rGen.a2*sqr(rPgen)+rGen.a1*rPgen+rGen.a0;
```

```
procedure DecodeCommDataToLFData(const rBus:TBusArr1;  
var rNbus,rNsal:integer;  
var rV,rSg,rSL:CxArr1;  
var rCap:dArr1;  
var rTypBus:iArr1;  
const rBranch:TBranchArr1;  
var rZ,rTp:CxArr2;  
var rLc,rTr:dArr2);  
var i,ke:integer;
```

```
rNbus:=high(rBus)+1;  
rNsal:=high(rBranch)+1;  
rV:=array[0..rNbus-1] of real;  
rSg:=array[0..rNbus-1] of real;  
rSL:=array[0..rNbus-1] of real;  
rCap:=array[0..rNbus-1] of real;  
rTypBus:=array[0..rNbus-1] of integer;  
for i:=0 to rNbus-1 do  
begin  
rBus[i].real:=rBus[i].absV;  
rBus[i].imag:=rBus[i].sudV;  
rBus[i].Pgen:=rBus[i].Pgen;  
rBus[i].Qgen:=rBus[i].Qgen;  
rBus[i].PL:=rBus[i].PL;  
rBus[i].QL:=rBus[i].QL;  
rBus[i].Cap:=rBus[i].Cap;  
rBus[i].typeBus:=rBus[i].typeBus;
```

uLoadflowBU

```
;
length(rZ,rNbus,rNbus);
length(rTp,rNbus,rNbus);
length(rLc,rNbus,rNbus);
length(rTr,rNbus,rNbus);
i:=0 to rNsal-1 do
in
ri:=rBranch[i].dari-1;
ke:=rBranch[i].ke-1;
r[dari,ke].real:=rBranch[i].R;
r[dari,ke].imag:=rBranch[i].X;
c[dari,ke]:=rBranch[i].Lc;
r[dari,ke]:=rBranch[i].Tr;
p[dari,ke].real:=rBranch[i].Tu;
p[dari,ke].imag:=rBranch[i].Su;
;

procedure DecodeData(var rParamLF:TParamLF;
var rNbus:integer;
var rV,rSg,rSL:CxArr1;
var rCap:dArr1;
var rTypBus:iArr1;
var rZ,rTp:CxArr2;
var rLc,rTr:dArr2);
j:integer;
if rParamLF.MaxIterasi=0 then rParamLF.MaxIterasi:=15;
if rParamLF.Toleransi=0 then rParamLF.Toleransi:=0.0001;
rParamLF.Zbase:=sqr(rParamLF.Vbase*rParamLF.Vkonst)/
(rParamLF.Pbase*rParamLF.Pkonst);
rParamLF.Ibase:=(rParamLF.Vbase*rParamLF.Vkonst)/rParamLF.Zbase;
i:=0 to rNbus-1 do
in
rSg[i].real:=rSg[i].real/rParamLF.Pbase;
rSg[i].imag:=rSg[i].imag/rParamLF.Pbase;
rSL[i].real:=rSL[i].real/rParamLF.Pbase;
rSL[i].imag:=rSL[i].imag/rParamLF.Pbase;
;
if rParamLF.ParamBranch=pbOhm then
in
i:=0 to rNbus-1 do
begin
for j:=0 to rNbus-1 do
begin
if rZ[i,j].imag<>0 then
begin
rZ[i,j].real:=rZ[i,j].real/rParamLF.Zbase;
rZ[i,j].imag:=rZ[i,j].imag/rParamLF.Zbase;
rLc[i,j]:=rLc[i,j]/rParamLF.Zbase;
```

uLoadflowBU

```
end;
nd;
f;

i:=0 to rNbus-1 do
in
j:=0 to rNbus-1 do
gin
rZ[i,j].imag<>0 then
egin
rZ[j,i]:=Cmplx(rZ[i,j]);
rLc[j,i]:=rLc[i,j];
nd;
d;
}

ion FindSumGen(const rNbus:integer;
const rTypBus:iArr1):integer;
integer;

lt:=0;
:=0 to rNbus-1 do
n
TypBus[i]=2 then
gin
ic(result);
f;

dure Admitansi(const rNbus:integer;
const rZ,rTp:CxArr2;
const rLc,rTr:dArr2;
const rCap:dArr1;
var rY:CxArr2);
,k:integer;
:double;
':CxArr2;
n,Za,Ca:TCmplx;

length(CY,rNbus,rNbus);
:=Cmplx(1.0,0.0);
:=0 to rNbus-1 do
n
j:=0 to rNbus-1 do
gin
rZ[i,j].imag<>0 then
egin
CY[i,j]:=Divide(sum,rZ[i,j]);
```


uLoadflowBU

```

nd;
d;
;
Length(rY,rNbus,rNbus);
i:=0 to rNbus-1 do
in
r j:=0 to rNbus-1 do
gin
f j=i then
egin
rY[i,j]:=Cmplx(0.0,0.0);
for k:=0 to rNbus-1 do
begin
rY[i,j].real:=rY[i,j].real+CY[i,k].real;
rY[i,j].imag:=rY[i,j].imag+CY[i,k].imag+rLc[i,k];
end;
nd
lse
egin
rY[i,j].real:=-CY[i,j].real;
rY[i,j].imag:=-CY[i,j].imag;
nd;
d;
;
i:=0 to rNbus-1 do
in
r j:=0 to rNbus-1 do
gin
f rTr[i,j]>0 then
egin
rY[i,i].real:=rY[i,i].real-CY[i,j].real;
rY[i,i].imag:=rY[i,i].imag-CY[i,j].imag-rLc[i,j];
rY[i,i].real:=rY[i,i].real+CY[i,j].real/sqr(rTr[i,j]);
rY[i,i].imag:=rY[i,i].imag+CY[i,j].imag/sqr(rTr[i,j]);
rY[j,j].real:=rY[j,j].real-CY[i,j].real;
rY[j,j].imag:=rY[j,j].imag-CY[i,j].imag-rLc[i,j];
rY[j,j].real:=rY[j,j].real+CY[i,j].real;
rY[j,j].imag:=rY[j,j].imag+CY[i,j].imag;
rY[i,j].real:=-1*CY[i,j].real/rTr[i,j];
rY[i,j].imag:=-1*CY[i,j].imag/rTr[i,j];
rY[j,i].real:=rY[i,j].real;
rY[j,i].imag:=rY[i,j].imag;
{CLc:=TComplex.Create(0.0,aLc[i,j]);
result[i,i]:=result[i,i]-Cx[i,j]-CLc;
result[i,i]:=result[i,i]+Cx[i,j]/sqr(at);
result[j,j]:=result[j,j]-Cx[i,j]-CLc;
result[j,j]:=result[j,j]+Cx[i,j];
result[i,j]:=-Cx[i,j]/at;
result[j,i]:=result[i,j];
CLc.Free;}

```

uLoadflowBU

```

nd;
d;
;
:=0 to rNbus-1 do
in
  :=0 to rNbus-1 do
  gin
  rTp[i,j].real<0 then
  egin
  a:=rTp[i,j].real*cos(rTp[i,j].imag);
  b:=rTp[i,j].real*sin(rTp[i,j].imag);
  rY[i,i].real:=rY[i,i].real-CY[i,j].real;
  rY[i,i].imag:=rY[i,i].imag-CY[i,j].imag-rLc[i,j];
  rY[i,i].real:=rY[i,i].real+CY[i,j].real/(sqr(a)+sqr(b));
  rY[i,i].imag:=rY[i,i].imag+CY[i,j].imag/(sqr(a)+sqr(b))+rLc[i,j];
  Za:=Cmplx(CY[i,j]);
  Za:=Negative(Za);
  Ca:=Cmplx(a,b);
  sum:=Divide(Za,Ca);
  rY[j,i]:=Cmplx(sum);
  Ca:=Conjugate(Ca);
  sum:=Divide(Za,Ca);
  rY[i,j]:=Cmplx(sum);
  {sUpfc:=TComplex.Create(aUpfc[i,j].tap*cos(aUpfc[i,j].sudut),
    aUpfc[i,j].tap*sin(aUpfc[i,j].sudut));
  CLc:=TComplex.Create(0.0,aLc[i,j]);
  result[i,i]:=result[i,i]-Cx[i,j]-CLc;
  result[i,i]:=result[i,i]+Cx[i,j]/sqr(sUpfc.Abs)+CLc;
  result[i,j]:=-Cx[i,j]/conj(sUpfc);
  result[j,i]:=-Cx[i,j]/sUpfc;
  CLc.Free;
  sUpfc.Free;}
nd;
d;
;
:=0 to rNbus-1 do
in
  Cap[i]<0 then
  gin
  Y[i,i].imag:=rY[i,i].imag+rCap[i];
  d;
;
;
function RecToPolar(const rData:CxArr1):CxArr1;
NData:integer;
s,sud:double;

ata:=high(rData)+1;
.length(result,NData);

```

uLoadflowBU

```
i:=0 to NData-1 do
in
s:=getAbs(rData[i]);
d:=getAngleRad(rData[i]);
result[i]:=Cmplx(abs,sud);
;

function PolarToRec(const rData:CxArr1):CxArr1;
NData:integer;
real,imag:double;
begin
NData:=high(rData)+1;
Length(result,NData);
i:=0 to NData-1 do
in
real:=rData[i].real*cos(rData[i].imag);
imag:=rData[i].real*sin(rData[i].imag);
result[i]:=Cmplx(real,imag);
;

function MaxDataArray(const rData:dArr1):double;
NData:integer;
begin
NData:=high(rData)+1;
result:=abs(rData[0]);
i:=1 to NData-1 do
in
if result<abs(rData[i]) then
begin
result:=abs(rData[i]);
end;
;

procedure AliranDaya(const rNbus:integer;const rV:CxArr1;
const rY:CxArr2;
const rLc:dArr2;
var rAlirS:CxArr2);
i,j:integer;
result,tmp1,tmp2:TCmplx;
begin
Length(rAlirS,rNbus,rNbus);
i:=0 to rNbus-1 do
in
for j:=0 to rNbus-1 do
begin
rAlirS[i,j].real:=0.0;
rAlirS[i,j].imag:=0.0;
```

uLoadflowBU

```

if j > i then
begin
if rY[i,j].imag < 0 then
begin
Lca:=Cmplx(0.0,rLc[i,j]);
tmp1:=Multiply(Multiply(Conjugate(rV[i]),Subtract(rV[i],rV[j])),
    Negative(rY[i,j]));
tmp2:=Multiply(Multiply(Conjugate(rV[i]),rV[i]),Lca);
rAlirS[i,j]:=Conjugate(Add(tmp1,tmp2));
//result[i,j]:=conj(aV[i])*(aV[i]-aV[j])*(-aY[i,j])+
//    conj(aV[i])*aV[i]*dLc;
//result[i,j]:=conj(result[i,j]);
end;
end;
end;

procedure DayaGen(const rNbus:integer;
const rV,rSL:CxArr1;
const rY:CxArr2;
const rTyp:iArr1;
var rSg:CxArr1);
j:integer;
n:double;

:=0 to rNbus-1 do
begin
n:=0.0;
if Typ[i]=2 then
begin
for j:=0 to rNbus-1 do
begin
//sum:=sum+(Fi*(Ej*Gij+Fj*-Bij)-Ei*(Fj*Gij-Ej*-Bij));
sum:=sum+(rV[i].imag*(rV[j].real*rY[i,j].real+rV[j].imag*-rY[i,j].imag)-
    rV[i].real*(rV[j].imag*rY[i,j].real-rV[j].real*-rY[i,j].imag));
end;
rSg[i].imag:=sum+rSL[i].imag;
end;
end;

procedure DayaSlack(const rNbus:integer;
const rAlirS:CxArr2;
const rTyp:iArr1;
const rSL:CxArr1;
var rSg:CxArr1);
i:integer;
nP,sumQ:double;

```

uLoadflowBU

```

i:=0 to rNbus-1 do
in
Type[i]=1 then
gin
sumP:=0.0;
sumQ:=0.0;
for j:=0 to rNbus-1 do
egin
if rAlirS[i,j].imag<>0 then
begin
sumP:=sumP+rAlirS[i,j].real;
sumQ:=sumQ+rAlirS[i,j].imag;
end;
end;
Sg[i].real:=sumP+rSL[i].real;
Sg[i].imag:=sumQ+rSL[i].imag;
d;
;

procedure ArusBranch(const rNbus:integer;
const rV:CxArr1;
const rLc:dArr2;
const rY:CxArr2;
var rArus:CxArr2);
j:integer;
c,tmp1,tmp2:TCmplx;

length(rArus,rNbus,rNbus);
length(rArus,rNbus,rNbus);
:=0 to rNbus-1 do
n
j:=0 to rNbus-1 do
gin
rY[i,j].imag<>0 then
egin
xLc:=Cmplx(0,rLc[i,j]);
tmp1:=Multiply(Subtract(rV[i],rV[j]),Negative(rY[i,j]));
tmp2:=Multiply(rV[i],xLc);
Arus[i,j]:=Conjugate(Add(tmp1,tmp2));
/result[i,j]:=(aV[i]-aV[j])*(-aY[i,j])+aV[i]*xLc;
/result[i,j]:=Conj(result[i,j]);
nd
se
egin
Arus[i,j].real:=0.0;
Arus[i,j].imag:=0.0;
nd;
l;

```

uLoadflowBU

```
procedure UpdateAkhir(const rNbus,rNsal:integer;
var rParamLF:TParamLF;
const rV,rSg,rSL:CxArr1;
const rAlir,rArus:CxArr2;
var rBus:TBusArr1;
var rBranch:TBranchArr1);
dari,ke:integer;
rParamLF.SumGen:=Cmplx(0.0,0.0);
rParamLF.SumLoad:=Cmplx(0.0,0.0);
for i:=0 to rNbus-1 do
begin
rBus[i].absV:=rV[i].real;
rBus[i].sudV:=rV[i].imag;
rBus[i].Pgen:=rSg[i].real*rParamLF.Pbase;
rBus[i].Qgen:=rSg[i].imag*rParamLF.Pbase;
rBus[i].PL:=rSL[i].real*rParamLF.Pbase;
rBus[i].QL:=rSL[i].imag*rParamLF.Pbase;
rParamLF.SumGen:=Add(rParamLF.SumGen,rBus[i]);
rParamLF.SumLoad:=Add(rParamLF.SumLoad,rBus[i]);
end;
rParamLF.SumGen:=Multiply(rParamLF.SumGen,rParamLF.Pbase);
rParamLF.SumLoad:=Multiply(rParamLF.SumLoad,rParamLF.Pbase);
rParamLF.SumLoss:=Subtract(rParamLF.SumGen,rParamLF.SumLoad);
for i:=0 to rNsal-1 do
begin
rBranch[i].dari:=rBranch[i].dari-1;
rBranch[i].ke:=rBranch[i].ke-1;
rBranch[i].Sij:=Multiply(Cmplx(rAlir[dari,ke]),rParamLF.Pbase);
rBranch[i].Sji:=Multiply(Cmplx(rAlir[ke,dari]),rParamLF.Pbase);
rBranch[i].Aij:=Multiply(Cmplx(rArus[dari,ke]),rParamLF.Ibase);
rBranch[i].Aji:=Multiply(Cmplx(rArus[ke,dari]),rParamLF.Ibase);
end;
end;
procedure UpdateAkhir(const rNbus:integer;
var rParamLF:TParamLF;
var rSg,rSL:CxArr1;
var rAlir,rArus:CxArr2);
dari:integer;
rParamLF.SumGen:=Cmplx(0.0,0.0);
rParamLF.SumLoad:=Cmplx(0.0,0.0);
for i:=0 to rNbus-1 do
begin
rBus[i].real:=rSg[i].real*rParamLF.Pbase;
rBus[i].imag:=rSg[i].imag*rParamLF.Pbase;
rBus[i].real:=rSL[i].real*rParamLF.Pbase;
```

uLoadflowBU

```
rSL[i].imag:=rSL[i].imag*rParamLF.Pbase;
rParamLF.SumGen:=Add(rParamLF.SumGen,rSg[i]);
rParamLF.SumLoad:=Add(rParamLF.SumLoad,rSL[i]);

rParamLF.SumLoss:=Subtract(rParamLF.SumGen,rParamLF.SumLoad);
for i:=0 to rNbus-1 do
begin
  for j:=0 to rNbus-1 do
  begin
    if rAlir[i,j].imag <> 0 then
    begin
      rAlir[i,j]:=Multiply(rAlir[i,j],rParamLF.Pbase);
      rArus[i,j]:=Multiply(rArus[i,j],rParamLF.lbase);
    end;
  end;
end;

function FindVarControl(var rBus:TBusArr1):iArr1;
var Nbus,sa:integer;

sa:=high(rBus)+1;
for i:=0 to Nbus-1 do
begin
  if rBus[i].typeBus <> 3 then
  begin
    result:=i;
    c(sa);
  end;
end;

for i:=0 to Nbus-1 do
begin
  if rBus[i].typeBus=3 then
  begin
    if rBus[i].Qgen <> 0 then
    begin
      result:=i;
      c(sa);
    end;
  end;
end;

result:=length(result,sa);
end;

for i:=0 to Nbus-1 do
begin
  if rBus[i].typeBus <> 3 then
  begin
    result[sa]:=i;
    c(sa);
  end;
end;
```

uLoadflowBU

```
:=0 to Nbus-1 do
n
Bus[i].typeBus=3 then
gin
rBus[i].Qgen<>0 then
egin
result[sa]:=i;
inc(sa);
nd;
l;
```

```
ion FindVarControl(var rSg:CxArr1;
var rTypeBus:iArr1):iArr1;
Nbus,sa:integer;
```

```
is:=high(rSg)+1;
0;
:=0 to Nbus-1 do
n
TypeBus[i]<>3 then
gin
ic(sa);
l;
```

```
:=0 to Nbus-1 do
n
TypeBus[i]=3 then
gin
rSg[i].imag<>0 then
egin
inc(sa);
nd;
l;
```

```
length(result,sa);
0;
:=0 to Nbus-1 do
n
TypeBus[i]<>3 then
gin
sult[sa]:=i;
ic(sa);
l;
```

```
:=0 to Nbus-1 do
n
TypeBus[i]=3 then
gin
```


d;
;

uLoadflowBU

uLoadflowBU

```
rSg[i].imag<0 then
begin
result[sa]:=i;
inc(sa);
end;
l;

function FindBatasControl(var rBus:TBusArr1;
var rParamLF:TParamLF):TBatasArr1;
Nbus,sa:integer;

sa:=high(rBus)+1;
l;
for i:=0 to Nbus-1 do
begin
if Bus[i].typeBus<3 then
begin
inc(sa);
end;
end;

for i:=0 to Nbus-1 do
begin
if Bus[i].typeBus=3 then
begin
if rBus[i].Qgen<0 then
begin
inc(sa);
end;
end;
end;

sa:=length(result,sa);
l;
for i:=0 to Nbus-1 do
begin
if Bus[i].typeBus<3 then
begin
result[sa].min:=0.95;
result[sa].max:=1.05;
inc(sa);
end;
end;

for i:=0 to Nbus-1 do
begin
if Bus[i].Qgen<0 then
begin
result[sa].min:=0;
result[sa].max:=rBus[i].Qgen/rParamLF.Pbase;
inc(sa);
end;
end;
```