

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



SKRIPSI

**OPTIMASI DAYA REAKTIF SECARA EKONOMIS
MENGUNAKAN METODE *EVOLUTIONARY PROGRAMMING*
DAN *EVOLUTIONARY STRATEGY* PADA SALURAN
TRANSMISI 150 KV SUB SISTEM PAITON-BALI**

Disusun oleh :

**LUKMAN KHAKIM
00.12.058**

DESEMBER 2005

LEMBAR PERSETUJUAN

**OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN
METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY
STRATEGY* PADA SALURAN TRANSMISI 150 kV
SUB SISTEM PAITON-BALI**

SKRIPSI

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

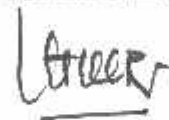
Disusun Oleh :

**LUKMAN KHAKIM
NIM. 00.12.058**


**Mengetahui,
Ketua Jurusan Teknik Elektro**

**Ir. F. Yudi Limpraptono, MT
NIP. Y.103 950 0274**

**Menyetujui,
Dosen Pembimbing**

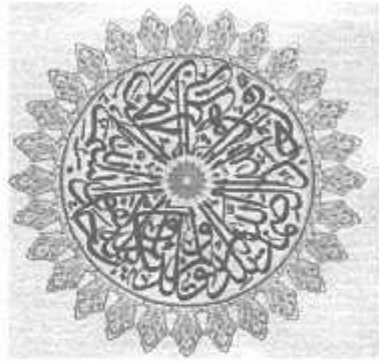


**Ir. H. Choiri
NIP. 130703042**

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

LeMBAR PErsEmbaHaN

EE...sPeCial FoR....



ALLOH SWT atas semua rahmat dan hidayah-MU

Serta kemudahan2 yang Engkau berikan kepadaQ

Serta JunjunganQ Nabi MUHAMMAD SAW yang telah memberi

sauri teladan kepada seluruh umat manusia

*Hanya kepada-MU aku berserah diri dan minta petunjuk Dalam
hidup ini dengan iman dan taqwa*

EbeZ@Q n MemeZ@Q yang tak pernah henti membakar

aku n dukungan Finansialnya (maffin Q ya..DosoQ

kpdmU Terlalu bnfr), serta KKq@si_Pmlz@ an 2

@dikQ@ yang rajin Thanks ats doa, Supot an

biJaksanan kalian shgg Q bisa enCink-kon kuliah,

semoga Allah SWT selalu memberikan rahmat, hidayah

pada keluarga Q...Amiiiiin..

For

Dosen Pembimbing

Bapak, Ir.H.CHOIRI yang telah membimbing dan membantu saya dalam penyusunan skripsi ini. Mohon maaf atas kesalahan yang pernah Qmi perbuat(WalO CaPex ato sakit Bpk Tetap SmAnGat (Bantu Qmi) lebih lgi pAS Qmi kOmpRse PasTi Buat MaLU BpK, semoga Smoga Bapak memahaminya. Dan Qmi doakan Bpk Sbg DoseN FaVoritt Qmi sMua en doakan Qmi agr jd orang yang Berguna Dunia n aKhirat amiiiiiiiiiiiiiiiiiiii...

«(AriZZssss JeMBeR→sink Menehi Jurnal IEEE SkripZi Iki Pas tuku Lalapan En nGajaRi Aq{thanXs for u}. HenDra→(konco nggarab skripzi mbulot JgH kwatir TerNyata Lulus dEnGan Nilai pasZan n Dibantai KabisZsahlll Tp ojo mEneNg ae Yan KompRe) en Yan© (My Beast Pren tP rd CrZY, ojo maleZ, qm emank Anak Mami N SiMuka BayI(ZeBra)) n Andi 3 Dharmika→(maz Gun= u tmn+SahabatH yank PantanX NyerAH,knP Rambutmu gk prnh Panjang ooo..lp! khan u maaz gun, Cuit..1000x lariiz PuLsane) n Deni CrUUT→(makaciH Tumpangan ke P3b Waru N Makan gRatIZnya heee..:) n dik guDeL-Bogel teloen-somgGlonG(eko W_{aria})→ojo Bojak PunyaQ dang.. MasA gk Pny InisiatIF snDR... en mamaD→ jGn mau di kwn Gudel. Tri Panjen→(ojo Kesusu aE), BenI, ali tjuZZ→knp kamu gk T-ruzzz gk nyesel ta? Aq deain oj cpt luluZ jagn Cr rumus pentanahan aj .fisu/(mediun), SonY→Wah op ya...TonY.NenY si LadieZ FirsT Kenthank...MaZZ jayenk KisruhH, B.Puji..)>

AreQ Kontrakn PUSKOPAD A12 Ngopi Truz pantang mUleZ ati2 kEncinK ManieeZ@AbaZZ(FarHaT)→Cpt diagarap skrimise Ojo NgodeW ae(Sinak2 MerPati©),JoKO kLatEn→ojo mullah muleh sawi luluse EngKa/suwun geleNe ngeterke uQ nek ndi ae!, eko(ToGel)/pak TukanK→yee...nyeke saiki todo rajin...Aq IkUt berBelu SingKaWa atas Kejauln Jm 1 mLm aqLiarlunnu BocoooR,JohAn(QnTi)→ojo mBojo ae moso" Muleh ae, Budi Beset→"jgn Ada pErSainGan dAntar

AdiQ kelas N akakak Kelas", Kakt00z→Ojo LemEs Tetap semangat yooo!,LuQmaN→malang ojo neng WaGIR ae, maktum→sint menempuh hidp brU dG DdunK cpt buat ank yang Bnyk, Zaenal n Ali→sing rukun, Wahid→ojo angel2 garapane.wiki, widia,desta,aGung ponorogo(cpt dimarekne Kincir"E ndank digoleki tante G,kukuh,sGY, agung PK, mardiono, Emi,erma n ArYo, AriZZ BTK, Tomy, Rully, SamZi/ZU→ arpan, aZyik, Agus Gajah,agus bima,irpan, puGuh, adi bgs n mery, Harie, adek,daud, piere,fini zakiah, sanTo "sory gk sido dek kdr, knp kPl u ky Bola?", udin, Tanto→Q sK Gaya Lho. yusaX, Kasur(surYa)←aljabar Vektor 'A',Mia, hadll(andle),Mya,Antony99 suwun donate, komank98, farid98, gandul elka, nanank elka, mumuN elka,jabriX, Ditoo, IwanZ, NiwaN Iwak UB, Dani Iwak UB, Diana Pertanian UB←kM uDh LuluZ ya..nt Hr Raya aq Krmh U, AgunK Zipi←km luLuZ jg trnytL,BahruLL 01→ojo ngiler', Andie CEPU thanks ats u Question en masukannYa kmjoko medlun,dedi dukur, Ziget OcoZ,siget Kallmantan Rendi 01', Jati 01'→*cpt lulus selak TuweK', RoMa 01', HandQqo,antonsmbr sr,tH011b,hAfIZULLaH(MengOng),JUNeD, dan sbgn... aqu wez dhoweh omonk wiz pokoknya ST 00' Kbeh aE, sink Lali gk Tak SebutKan I'm sorry.....

ZaM UgRo beserta KelUarga

MakaCih ya mas atas programnya Walo RaDa gk ValIDD tp gk pa2 ya.. lucu aj hiii..., karena dengan program SamPean aku bisa lulus dari ITN,aku sangat menghargai usaha keras dari mas tp Ojo diaWur kasiHan khan...?, semoga Allah membalas semua kebaikan sampeyan

©©BebEraPa Orank Yng Gk bS aQ SbUtIn, Suwun Ats DO'A Kalian wlo KaliAn Sk Buat Aq PuZZZZINg (©Ok©→swon ats doa n tahajudmu..aq tak akn melupakan hr ultahmu kaQ?-EL*_*→supportmu thanks ya!!-P→si crewet-De→iBu Perawat yg Baek-Dw→Zintailali deNgon Swajarnya km km ga prah tau kapn km akn membencinya dan Berilah Dengan Swajarnya km km jk bakal tau knp km akaN mencintanya Hrya Cinta knD Allah SWT yg palinK keKalll' iK→si-CombLankq-V_teX-N.....!)"\$*Smoga Budi kalian-Kalian Smua dibalas dG Diberi Hidayah n Karunia Allah SWT*



[Handwritten signature]

ABSTRAKSI

OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON-BALI

(Lukman Khakim, Nim. 00.12058, Teknik Elektro Energi Listrik S-1)
(Dosen Pembimbing : Ir.H Choiri)

Kata Kunci: *Optimal Reactive Power Planning, Evolutionary Programming, Evolutionary Strategy, Optimasi Biaya Pembangkitan.*

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 optimasi daya reaktif secara ekonomis dengan menggunakan metode *Evolutionary Programming* dan *Evolutionary Strategy*. Hasil dari analisa tersebut nantinya dapat digunakan sebagai salah satu acuan dalam operasi pembangkitan dan penyaluran daya yang ekonomis dan optimal, terutama mengenai optimasi biaya pembangkitan. Input dari program ini adalah biaya bahan bakar (*fuel cost*), biaya pemeliharaan 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 pembangkitan yang optimum.

Analisa dilakukan dengan bantuan program komputer dengan menggunakan bahasa pemrograman Delphi versi 7.0 dan telah sukses dicoba pada sub sistem 150 kV Paiton-Bali yang terdiri dari 25 bus, dimana telah berhasil dilakukan penghematan biaya pembangkitan sebesar Rp.131.666.094,- per-jam atau terjadi optimasi biaya sebesar 31,8980 %.

KATA PENGANTAR

Dengan rahmat Allah SWT, dan mengucapkan syukur kehadiran-Nya atas karunia yang dilimpahkan kepada saya sehingga dapat menyelesaikan skripsi yang berjudul "OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON-BALI".

Skripsi ini bertujuan untuk memenuhi kurikulum akademik yang harus ditempuh oleh setiap mahasiswa ITN Malang dalam menempuh sekaligus mengakhiri pendidikan pada jenjang S-1 pada Jurusan Teknik Elektro Konsentrasi Teknik Energi Listrik.

Saya menyadari bahwa skripsi ini masih jauh dari kesempurnaan, karena itu saran dan kritik membangun sangat saya harapkan.

Atas segala bimbingan, pengarahan dan bantuan yang diberikan, sehingga tersusun skripsi ini, maka penulis menyampaikan terima kasih kepada;

1. Bapak Dr. Ir. Abraham Lomi, MSEE, selaku Rektor ITN Malang.
2. Bapak Ir. F Yudi Limpraptono, MT, selaku Kepala Jurusan Teknik Elektro ITN Malang.
3. Bapak Ir. H. Choiri, selaku Dosen Pembimbing.
4. Bapak Ir. Totok Prawiro, selaku pembimbing lapangan di PT. PLN (persero) P3B Jawa-Bali.

Akhirnya saya mengharapkan skripsi ini berguna dan bermanfaat bagi rekanmahasiswa khususnya Jurusan Teknik Elektro.

Malang, September 2005

Penyusun

DAFTAR ISI

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

| | | |
|--------|---|----|
| 2.2.1. | Saluran Transmisi Pendek | 7 |
| 2.2.2. | Saluran Transmisi Menengah | 8 |
| 2.2.3. | Saluran Transmisi Panjang | 10 |
| 2.3. | Daya Dalam Sistem Tenaga Listrik | 11 |
| 2.3.1. | Daya Aktif (<i>Active Power</i>) | 11 |
| 2.3.2. | Daya Reaktif (<i>Reactive Power</i>) | 11 |
| 2.3.3. | Daya semu (<i>Apperent Power</i>) | 12 |
| 2.4. | Sistem Per-Unit | 13 |
| 2.4.1. | Mengubah Dasar Sistem Per-Unit..... | 14 |
| 2.5. | Sistem Operasi Pada Sistem Tenaga Listrik | 15 |
| 2.6. | Karakteristik Unit Pembangkit | 18 |
| 2.6.1. | Karakteristik <i>Input-Output</i> | 18 |
| 2.6.2. | Karakteristik <i>Heat Rate</i> | 20 |
| 2.6.3. | Karakteristik <i>Incremental Heat-Rate</i> dan <i>Incremental Fuel Cost</i> | 21 |
| 2.7. | <i>Economic Dispatch</i> | 22 |
| 2.7.1. | Fungsi Biaya Bahan Bakar | 22 |
| 2.7.2. | Fungsi Biaya Daya Reaktif Pembangkit | 23 |
| 2.7.3. | <i>Economic Dispatch</i> Dengan Mengabaikan Rugi-rugi Transmisi | 24 |
| 2.7.4. | <i>Economic Dispatch</i> Dengan Memperhitungkan Rugi-Rugi Transmisi .. | 27 |

BAB III OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE

EVOLUTIONARY PROGRAMMING DAN EVOLUTIONARY

***STRATEGY*** 28

| | | |
|------|---------------------------|----|
| 3.1. | Analisa Aliran Daya | 28 |
|------|---------------------------|----|

| | | |
|----------|--|----|
| 3.1.1. | Klasifikasi Bus | 30 |
| 3.1.2. | Metode <i>Newton Raphson</i> | 31 |
| 3.2. | <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 33 |
| 3.2.1. | <i>Evolutionary Programming</i> | 33 |
| 3.2.2. | <i>Evolutionary Strategy</i> | 34 |
| 3.3. | Parameter <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 35 |
| 3.3.1. | Jumlah Generasi (Max Gen) | 36 |
| 3.3.2. | Ukuran Populasi (Popsiz)e..... | 36 |
| 3.3.3. | Probabilitas Mutasi (Pm) | 36 |
| 3.3.4. | Panjang Kromosom | 37 |
| 3.4. | Mekanisme <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 37 |
| 3.4.1. | Pengkodean (<i>Initialization</i>) | 37 |
| 3.4.2. | Fungsi Evaluasi (<i>Fungsi Fitness</i>) | 37 |
| 3.4.3. | Seleksi (<i>Selection</i>) | 38 |
| 3.4.3.1. | <i>Roulette Wheel Selection</i> | 38 |
| 3.4.3.2. | <i>Rank Selection</i> | 39 |
| 3.4.4. | Rekombinasi (<i>Recombination</i>)..... | 39 |
| 3.4.5. | Mutasi (<i>Mutation</i>) | 40 |
| 3.4.6. | Kompetisi (<i>Competition</i>)..... | 41 |
| 3.5. | Formulasi Masalah Optimasi Daya Reaktif | 42 |
| 3.6. | Adaptasi <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> Ke Masalah Optimasi Daya Reaktif..... | 43 |
| 3.6.1. | Representasi Solusi | 43 |

| | |
|---|----|
| 3.6.2. Inisialisasi (<i>Initialization</i>) | 44 |
| 3.6.3. Statistik (<i>Statistics</i>) | 44 |
| 3.6.4. Mutasi (<i>Mutation</i>) | 44 |
| 3.6.4.1. Mutasi <i>Evolutionary Programming</i> | 44 |
| 3.6.4.2. Mutasi <i>Evolutionary Strategy</i> | 45 |
| 3.6.5. Kompetisi (<i>Competition</i>)..... | 45 |
| 3.6.6. Konvergensi (<i>Konvergence</i>)..... | 46 |
| | |
| BAB IV HASIL DAN ANALISA HASIL | 42 |
| 4.1. Program Komputer Optimasi Daya Reaktif Menggunakan Metode <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 47 |
| 4.1.1. Algoritma Program | 47 |
| 4.1.2. <i>Flowchart</i> Program | 51 |
| 4.2. Validasi Data IEEE 30 Untuk Menyelesaikan Permasalahan Optimasi Daya Reaktif Secara Ekonomis Menggunakan Metode <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 54 |
| 4.2.1. Hasil Validasi IEEE 30 Dengan Menggunakan Metode <i>Evolutionary</i> <i>Programming</i> dan <i>Evolutionary Strategy</i> | 56 |
| 4.3. Data Pembangkitan Thermal Pada Sub Sistem Paiton-Bali | 58 |
| 4.4. Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton-Bali | 62 |
| 4.5. Data Saluran Transmisi 150 kV Sub Sistem Paiton-Bali..... | 63 |
| 4.6. Prosedur Pelaksanaan Program Perhitungan | 65 |

| | | |
|--------------|--|-----------|
| 4.7. | Hasil dan Analisis Hasil Perhitungan Optimasi Daya Reaktif Menggunakan Metode <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> Pada Saluran Transmisi 150 kV Sub Sistem Paiton-Bali..... | 72 |
| 4.7.1. | Hasil Perhitungan Sebelum Optimasi | 72 |
| 4.7.2. | Hasil Perhitungan Setelah Optimasi | 75 |
| 4.8. | Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Daya Reaktif Menggunakan Metode <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 78 |
| 4.8.1. | Tingkat Optimum Biaya Pembangkitan | 78 |
| 4.8.2. | Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus . | 80 |
| 4.8.3. | Perbandingan Tingkat Rugi-rugi Daya Pada Saluran | 81 |
| BAB V | KESIMPULAN DAN SARAN | 82 |
| 5.1. | Kesimpulan | 82 |
| 5.2. | Saran..... | 82 |

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

| | | |
|--------------|---|----|
| Gambar 2.1. | Rangkaian Setara Saluran Transmisi | 7 |
| Gambar 2.2. | Rangkaian Setara Saluran Transmisi Pendek | 8 |
| Gambar 2.3. | Rangkaian Setara Saluran Transmisi Menengah | 9 |
| Gambar 2.4. | Diagram Skema Saluran Transmisi Panjang | 10 |
| Gambar 2.5. | Representasi Segitiga Daya | 12 |
| Gambar 2.6. | Unit Boiler-Turbin-Generator | 19 |
| Gambar 2.7. | Kurva Karakteristik <i>Input-Output</i> Pembangkit Thermal | 20 |
| Gambar 2.8. | Kurva Karakteristik <i>Heat-Rate</i> Unit Pembangkit | 20 |
| Gambar 2.9. | Kurva Karakteristik <i>Incremental Heat-Rate/Fuel Cost</i> | 21 |
| Gambar 2.10. | N Unit Pembangkit Thermal Melayani Beban P_R | 24 |
| Gambar 2.11. | N buah Pembangkit Thermal Melayani Beban P_R Melalui Saluran Transmisi | 27 |
| Gambar 3.1. | <i>Intermediate recombination</i> dari Induk (<i>parent</i>) a) dan b) menjadi Anak (<i>offspring</i>) c)..... | 40 |
| Gambar 3.2. | Mutasi Gaussian Dari Induk (<i>Parent</i>) (a) Menghasilkan Anak (<i>Offspring</i>) (b) | 41 |
| Gambar 4.1. | <i>Flowchart</i> Perhitungan Dasar Permasalahan | 51 |
| Gambar 4.2. | <i>Flowchart</i> Program Optimasi Menggunakan Metode <i>Evolutionary Programming</i> dan <i>Evolutionary Strategy</i> | 52 |
| Gambar 4.3. | <i>Flowchart</i> Perhitungan Aliran Daya Metode <i>Newton Rapshon</i> | 53 |
| Gambar 4.4. | Tampilan Parameter Validasi IEEE 30 | 56 |

| | | |
|--------------|---|----|
| Gambar 4.5. | Hasil <i>Load Flow</i> | 57 |
| Gambar 4.6. | Tampilan Hasil Validasi IEEE 30 Hasil Perhitungan Biaya Pembangkitan | 57 |
| Gambar 4.7. | Diagram Segaris Jaringan Sistem Transmisi Listrik 150 kV Paiton-Bali | 60 |
| Gambar 4.8. | Menu Utama Program | 65 |
| Gambar 4.9. | Tampilan Data | 65 |
| Gambar 4.10. | Tampilan Data Bus..... | 66 |
| Gambar 4.11. | Tampilan Data Saluran | 66 |
| Gambar 4.12. | Tampilan Data Generator | 67 |
| Gambar 4.13. | Tampilan Parameter | 67 |
| Gambar 4.14. | Tampilan Parameter Dengan Menekan <i>Used Default</i> | 68 |
| Gambar 4.15. | Tampilan <i>Load flow</i> Pada Kondisi Awal (Sebelum Optimasi) | 68 |
| Gambar 4.16. | Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi).. | 69 |
| Gambar 4.17. | Tampilan <i>Summary</i> (Sebelum Optimasi)..... | 69 |
| Gambar 4.18. | Tampilan Hasil Program | 69 |
| Gambar 4.19. | Tampilan <i>Load flow</i> Pada Kondisi Akhir (Setelah Optimasi) | 70 |
| Gambar 4.20. | Tampilan Hasil Aliran Daya Pada Kondisi Akhir (Setelah Optimasi) .. | 71 |
| Gambar 4.21. | Tampilan <i>Summary</i> (Setelah Optimasi) | 71 |

DAFTAR TABEL

| | | |
|-------------|---|----|
| Tabel 4.1. | Data Generator Dan Koefisien Biaya IEEE 30 | 54 |
| Tabel 4.2. | Data Bus IEEE 30 | 54 |
| Tabel 4.3. | Saluran Impedansi IEEE 30 | 55 |
| Tabel 4.4. | Perbandingan Hasil Data Referensi Jurnal Dengan Data Optimasi | 58 |
| Tabel 4.5. | Parameter Unit Pembangkit Thermal..... | 58 |
| Tabel 4.6. | Persamaan Biaya Pembangkitan Unit Pembangkit Thermal Paiton-Bali .. | 59 |
| Tabel 4.7. | Penomoran Bus 150 kV Sub Sistem Paiton-Bali | 61 |
| Tabel 4.8. | Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton-Bali | 62 |
| Tabel 4.9. | Data Saluran Transmisi 150 kV sub Sistem Paiton-Bali | 64 |
| Tabel 4.10. | Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sebelum Optimasi | 72 |
| Tabel 4.11. | Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi | 73 |
| Tabel 4.12. | Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum Optimasi | 74 |
| Tabel 4.13. | Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi Sebelum Optimasi | 74 |
| Tabel 4.14. | Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Setelah Optimasi | 75 |
| Tabel 4.15. | Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi | 76 |
| Tabel 4.16. | Total Peembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi | 77 |

| | |
|--|----|
| Tabel 4.17. Hasil Perhitungan Daya Yang Dibangkitkan dan Biaya Operasi | |
| Setelah Optimasi | 77 |
| Tabel 4.18. Perbandingan Tingkat Optimum Biaya Pembangkitan | 78 |
| Tabel 4.19. Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran | |
| Sebelum dan Setelah Optimasi | 79 |
| Tabel 4.20. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus | 80 |

DAFTAR GRAFIK

| | | |
|-------------|---|----|
| Grafik 4.1. | Hasil Perhitungan Daya Yang Dibangkitkan Sebelum Optimasi | 74 |
| Grafik 4.2. | Hasil Perhitungan Daya Yang Dibangkitkan Setelah Optimasi | 77 |
| Grafik 4.3. | Perbandingan Biaya Pembangkitan Sebelum dan Setelah Optimasi | 79 |
| Grafik 4.4. | Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum dan Setelah Optimasi | 79 |
| Grafik 4.5. | Perbandingan Profil Tegangan | 81 |
| Grafik 4.6. | Perbandingan Rugi-rugi Daya | 81 |

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dengan adanya persoalan peningkatan kebutuhan daya listrik dari waktu ke waktu, sehingga timbul suatu permasalahan yaitu bagaimana operasi sistem tenaga listrik selalu bisa memenuhi permintaan daya untuk setiap saat, dengan kualitas baik dan ekonomis. Oleh karena itu pada suatu operasi sistem tenaga listrik pada beban tertentu perhitungan ekonomis harus tetap merupakan suatu prioritas utama sehingga nantinya diperlukan suatu rencana operasi yang optimum dengan tetap memenuhi beberapa persyaratan pengoperasian sistem tenaga listrik, antara lain daya yang dibangkitkan cukup untuk memasok beban dengan memperhitungkan rugi-rugi daya pada saluran transmisi, tegangan bus sesuai dengan rating yang telah ditentukan, serta tidak adanya beban lebih pada unit-unit pembangkit yang beroperasi.

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

Dalam pemecahan permasalahan tersebut metode yang digunakan adalah *evolutionary programming* dan *evolutionary strategy*, yakni dua metode optimasi yang menyerupai operasi-operasi seleksi alamiah dengan tujuan mengoptimasi daya reaktif secara ekonomis dan mengurangi rugi-rugi daya pada saluran transmisi 150 kV sub sistem Paiton-Bali.^[1]

1.2. Rumusan Masalah

Berdasarkan latar belakang tersebut diatas permasalahan yang timbul:

- Bagaimana kondisi profil tegangan tiap bus pada saluran transmisi 150 kV sub sistem Paiton-Bali.
- Bagaimana cara mengoptimalkan penyaluran daya reaktif sehingga dapat mengurangi rugi-rugi daya
- Bagaimana cara mengurangi biaya pengoperasian pembangkit pada saluran transmisi 150 kV sub sistem Paiton-Bali.

1.3. Tujuan

Tujuan dari skripsi ini adalah memperbaiki profil tegangan dan optimasi daya reaktif untuk mengurangi rugi-rugi daya dengan biaya operasional yang ekonomis pada saluran transmisi 150 kV sub sistem Paiton-Bali menggunakan metode *evolutionary programming* dan *evolutionary strategy*.

1.4. Batasan Masalah

Agar pembahasan mengarah sesuai dengan tujuan, maka pembahasan ini dibatasi oleh hal berikut :

Analisa perhitungan hanya dilakukan pada pembangkit termal.

- Analisa dilakukan dengan asumsi bahwa sistem berada dalam operasi normal.
 - Tidak membahas biaya *start up*, *shut down*, dan biaya *spinning reserve*.
 - Tidak membahas masalah pengaruh kontrol tegangan.
-

- Tidak membahas kabel laut.
- Tidak membahas masalah peralatan kompensasi dan penempatannya.
- Analisa perhitungan optimasi daya reaktif pada saluran transmisi 150 kV sub sistem Paiton-Bali menggunakan metode *evolutionary programming* dan *evolutionary strategy*.

1.5. Metodologi Penelitian

Metodologi yang digunakan dalam pembahasan skripsi ini adalah sesuai langkah-langkah berikut :

1. Studi literatur yaitu kajian pustaka yang mempelajari teori-teori yang terkait melalui literatur yang telah ada dan berhubungan dengan permasalahan.
 2. Pengumpulan data lapangan yang dipakai dalam objek penelitian yakni; data pembangkit thermal, data impedansi saluran transmisi dan data pembebanan saluran transmisi 150 kV sub sistem Paiton-Bali.
 3. Menganalisa dan menghitung besar tegangan, sudut fasa tegangan, aliran daya pada tiap saluran serta biaya pembangkitan menggunakan metode *evolutionary programming* dan *evolutionary strategy* kemudian menarik kesimpulan dari analisa tersebut.
-

1.6. Sistematika Penulisan

Adapun sistematika pembahasan pada skripsi ini adalah sebagai berikut :

BAB I : PENDAHULUAN

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

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

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

BAB III : OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY*

Menguraikan teori dasar dari aliran daya, metode aliran daya *Newton Raphson*, teori serta adaptasi *evolutionary programming* dan *evolutionary strategy* ke permasalahan optimasi daya reaktif secara ekonomis.

BAB IV : HASIL DAN ANALISIS HASIL

Menguraikan alur program, hasil validasi, serta hasil perhitungan optimasi daya reaktif menggunakan metode *evolutionary programming* dan *evolutionary strategy*.

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

Adapun kontribusi dari skripsi ini adalah penggunaan dari metode *evolutionary programming* dan *evolutionary strategy* dalam mengalisa penyaluran daya reaktif diharapkan dapat menekan rugi-rugi daya dengan biaya pembangkitan yang murah didalam sistem tenaga listrik dan jika diterapkan oleh pemerintah khususnya PT. PLN(Persero) sebagai perusahaan penyedia energi listrik di Indonesia selanjutnya akan menambah keuntungan.

BAB II

SISTEM TENAGA LISTRIK, OPERASI EKONOMIS DAN KARAKTERISTIK UNIT PEMBANGKIT

2.1. Sistem Tenaga Listrik^[3]

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

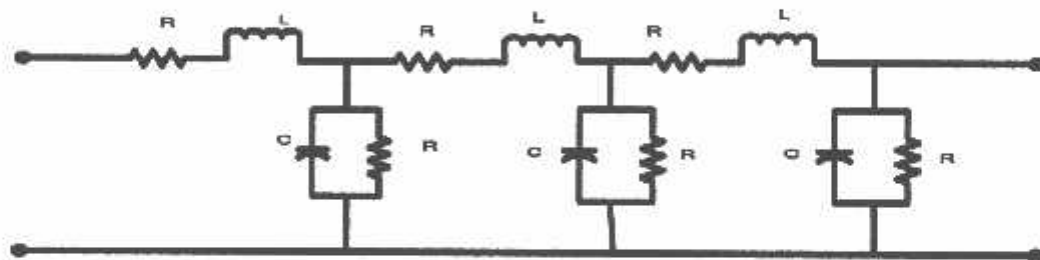
Saluran transmisi mempunyai empat parameter yang mempengaruhi kemampuannya dalam menyalurkan daya listrik. Keempat parameter tersebut yaitu : resistansi (R), induktansi (L), kapasitansi (C), serta konduktansi (G).

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

Impedansi seri terbentuk dari resistansi dan induktansi yang terbagi merata sepanjang saluran. Sedangkan konduktansi dan kapasitansi terdapat diantara penghantar-penghantar dari saluran fasa tunggal atau diantara penghantar dengan netral dari suatu saluran berfasa tiga membentuk admitansi paralel.

2.2. Saluran Transmisi^[3]

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



Gambar 2.1.
Rangkaian Setara Saluran Transmisi^[3]

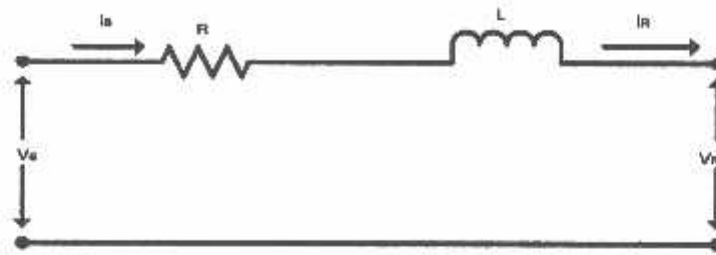
Menurut panjangnya, saluran transmisi dapat dikasifikasikan menjadi 3 golongan, yaitu :

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

2.2.1. Saluran Transmisi Pendek^[3]

Rangkaian ekivalen untuk saluran transmisi pendek diperlihatkan pada gambar 2.2 dimana I_s dan I_r merupakan arus pada ujung pengiriman dan ujung

penerimaan. Sedangkan V_S dan V_R adalah tegangan saluran terhadap netral pada ujung pengiriman dan ujung penerimaan.



Gambar 2.2.
Rangkaian Setara Saluran Transmisi Pendek^[3]

Karena tidak ada cabang paralel (shunt), arus pada ujung-ujung pengiriman dan penerimaan akan sama besar :

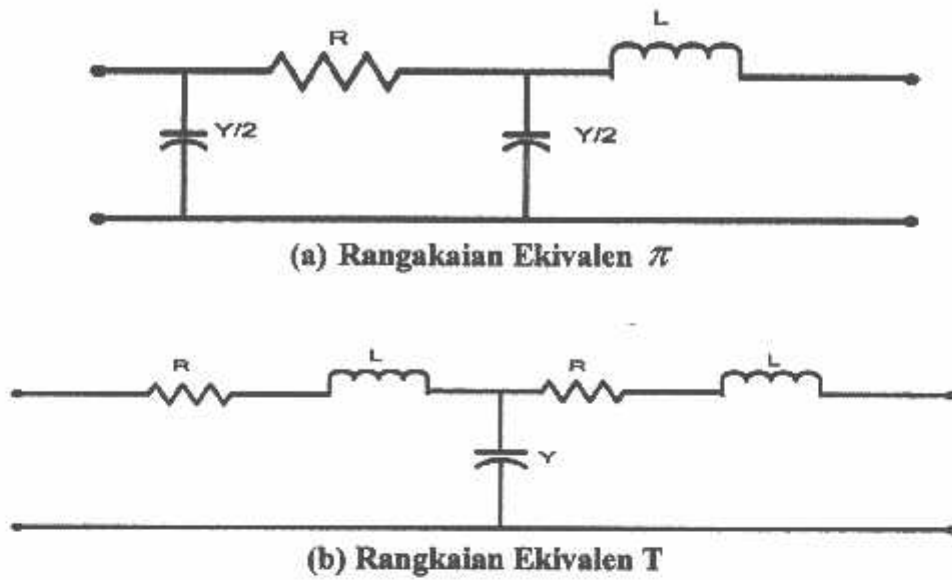
$$I_S = I_R \dots\dots\dots (2.1.)$$

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

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

2.2.2. Saluran Transmisi Menengah^[3]

Bertambahnya saluran menyebabkan kapasitansi shunt bertambah besar dan tidak dapat diabaikan. Saluran transmisi jarak menengah pada umumnya digambarkan dengan rangkaian π pada atau rangkaian T, dapat dilihat pada gambar 2.3(a) dan gambar 2.3(b). Dari dua versi ini rangkaian π lebih umum dipakai dari pada rangkaian T.



Gambar 2.3.
Rangkaian Setara Saluran Transmisi Menengah⁽³⁾

Untuk rangkaian π berlaku :

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

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

Untuk rangkaian T berlaku :

$$V_s = \left[\frac{ZY}{2} + 1 \right] V_R + \left[\frac{ZY}{4} + 1 \right] ZI_R \dots\dots\dots(2.5.)$$

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

Dimana :

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

2.2.3. Saluran Transmisi Panjang^[3]

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



Gambar 2.4.

Diagram Skema Saluran Transmisi Panjang^[3]

Persamaan tegang dan arus pada setiap titik sepanjang saluran transmisi dengan jarak x dari ujung sisi terima dapat ditulis sebagai berikut :

$$V = \frac{V_R + I_R Z_C}{2} e^{yX} + \frac{V_R - I_R Z_C}{2} e^{-yX} \dots\dots\dots(2.7.)$$

$$I = \frac{\frac{V_R}{Z_C} + I_R}{2} e^{yX} + \frac{\frac{V_R}{Z_C} - I_R}{2} e^{-yX} \dots\dots\dots(2.8.)$$

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

$$V_{\infty} = V_R \cosh \tau x + I_R Z_C \sinh \tau x \dots\dots\dots(2.9.)$$

$$I_{\infty} = I_R \cosh \tau x + \frac{V_R}{Z_C} \sinh \tau x \dots\dots\dots(2.10.)$$

Dimana :

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

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

2.3. Daya Dalam Sistem Tenaga Listrik^[3]

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

2.3.1 Daya aktif (*Active power*)^[3]

Secara umum daya aktif dinyatakan oleh persamaan :

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

Dimana :

V dan I nilai efektifnya.

P adalah daya rata-rata yang disebut juga daya aktif.

2.3.2 Daya Reaktif (*Reactive power*)^[3]

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

Persamaan daya reaktif adalah :

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

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

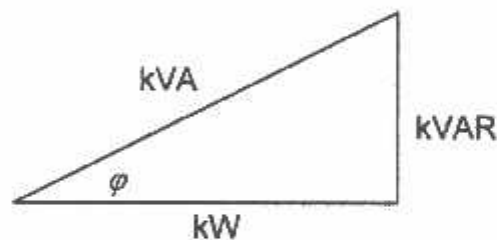
Daya reaktif induktif adalah daya listrik yang dibutuhkan untuk menghasilkan medan magnet yang dibutuhkan oleh alat-alat seperti motor induksi, transformator dan sebagainya.

2.3.3. Daya Semu (*Apparent power*)^[3]

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

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

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



Gambar 2.5.
Representasi Segitiga Daya^[3]

Dimana :

- $\cos \varphi$ = Faktor daya
- KW = Daya aktif
- KVA = Daya semu
- KVAR = Daya reaktif

2.4. Sistem Per-Unit^[3]

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

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

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

➤ Untuk data 1 phasa

Arus dasar

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

Impedansi dasar

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

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

➤ Untuk data 3 fasa

Arus dasar

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

Impedansi dasar

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

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

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

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

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

2.4.1. Mengubah Dasar Sistem Per-Unit⁽³⁾

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

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

Persamaan (2.22.) memperlihatkan bahwa impedansi per-unit berbanding lurus dengan MVA dasar serta berbanding terbalik dengan kuadrat tegangan dasar. Untuk mengubah dari impedansi per-unit menurut suatu dasar yang

diberikan menjadi impedansi per-unit menurut suatu dasar yang baru, dapat dipakai persamaan berikut:

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

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

2.5. Sistem Operasi Pada Sistem Tenaga Listrik^[2]

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

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

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

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

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

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

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

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

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

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

2.6. Karakteristik Unit Pembangkit^[2]

2.6.1. Karakteristik *Input-Output*^[2]

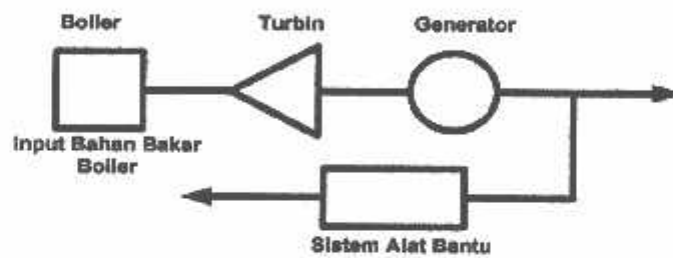
Hal yang paling mendasar dalam operasi pembangkitan yang ekonomis adalah dengan membuat karakteristik *input-output* dari unit pembangkit thermal. Karena ini diperoleh dari desain perencanaan atau melalui test pembangkit. Adapun definisi dari karakteristik *input-output* dari pembangkit itu sendiri adalah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

Pada pembangkit thermal input diberikan dalam satuan panas Btu/jam atau Kalori/jam dari bahan bakar yang diberikan boiler untuk menghasilkan *output* pembangkit. Sedangkan notasi yang digunakan adalah H (MBtu/h) atau dalam satuan yang lain H (MKal/h). Adapaun dalam skripsi ini, perhitungan dilakukan adalah dalam satuan MKal/jam. Selain itu *input* dari pembangkit dapat pula

dinyatakan dalam nilai uang yang menyatakan besarnya biaya yang diperlukan untuk bahan bakar. Notasi yang digunakan adalah F (Rp/h). Hubungan antara H dan F dapat dinyatakan dalam rumus sebagai berikut ini :

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

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



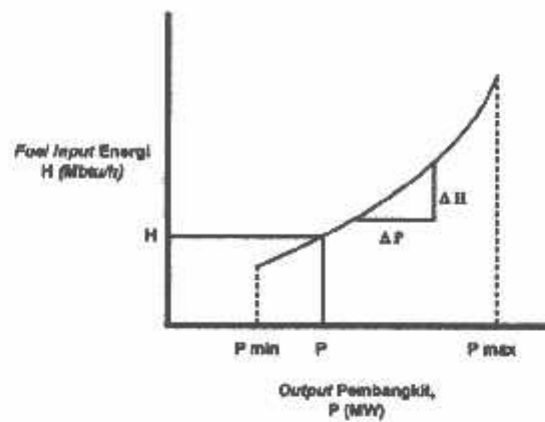
Gambar 2.6.
Unit Boiler-Turbin-Generator^[2]

Seperti digambarkan dalam gambar 2.6. maka *output* dari pembangkit tidak hanya dihubungkan dengan sistem saja akan tetapi juga untuk sistem peralatan bantu pembangkit didefinisikan sebagai daya yang dikeluarkan oleh generator karakteristik *input-output*, daya *output* adalah berupa daya netral dari pembangkit, notasi yang digunakan adalah P (MW).

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

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

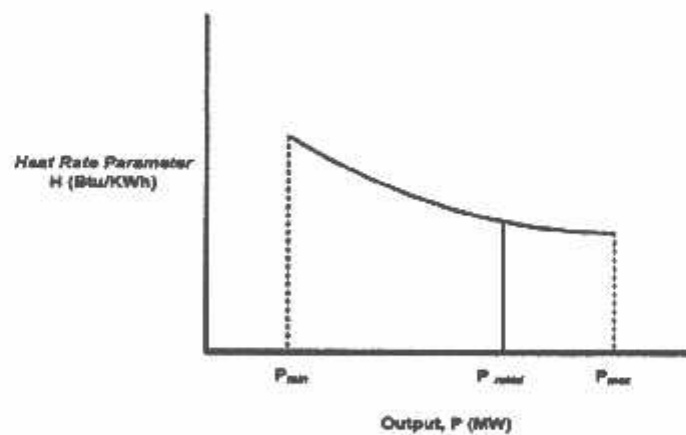
$$F = f(P) \dots\dots\dots (2.26.)$$



Gambar 2.7.
Kurva Karakteristik *Input-Output* Pembangkit Thermal^[2]

2.6.2. Karakteristik *Heat-Rate*^[2]

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



Gambar 2.8.
Kurva Karakteristik *Heat-Rate* Unit Pembangkit^[2]

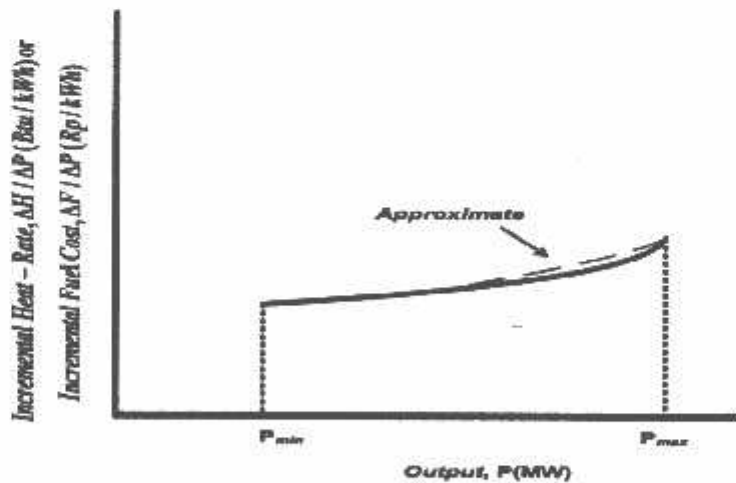
2.6.3 Karakteristik Incremental Heat-Rate dan Incremental Fuel Cost^[2]

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

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

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

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



Gambar 2.9.
Kurva Karakteristik *Incremental Heat-Rate/Fuel Cost*^[2]

Bila harga Δ sangat kecil maka dapat dinyatakan dengan persamaan berikut ini :

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

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

2.7 Economic Dispatch^[2]

Dalam pembahasan tentang *Optimal Power Flow* dan operasi pada sistem tenaga listrik yang ekonomis, maka kita selalu membicarakan *economic dispatch*. *Economic dispatch* adalah pembagian pembebanan pada pembangkit-pembangkit yang ada dalam suatu sistem tenaga listrik, secara optimum dan ekonomis pada beban tertentu. Dengan dilakukan *economic dispatch* maka akan didapatkan biaya bahan bakar yang paling murah dalam suatu sistem pembangkit. Oleh karena beban yang harus ditanggung oleh sistem pembangkit selalu berubah setiap periode waktu tertentu, maka perhitungan *economic dispatch* ini dilakukan untuk setiap harga beban tertentu pula.

2.7.1 Fungsi Biaya Bahan Bakar^[2]

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

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

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

Dimana:

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

P_i = Besar daya aktif yang terbangkit pada unit generator i

a, b, c = Koefisien persamaan kuadrat.

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

2.7.2. Fungsi Biaya Daya Reaktif Pembangkit^[1]

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

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

Dimana:

$C_i(Q_i)$ = Biaya daya reaktif pembangkit

C_{gr} = Biaya pengoperasian daya reaktif

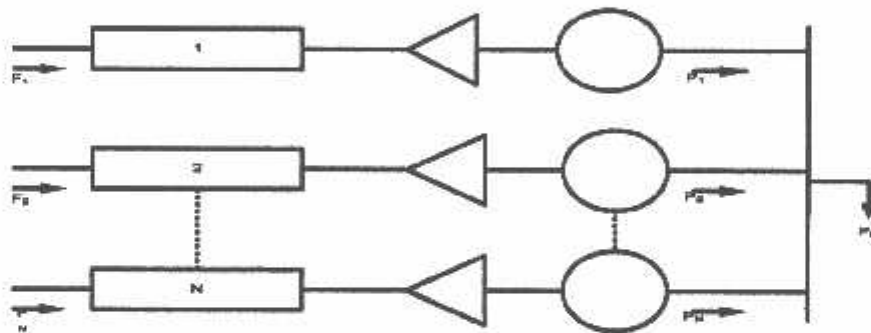
\bar{Q}_{gr} = Besar daya reaktif yang terbangkit pada unit generator i

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

2.7.3. *Economic Dispatch Dengan Mengabaikan Rugi-rugi Transmisi*^[2]

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

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



Gambar 2.10.
N Unit Pembangkit Thermal Melayani Beban P_R ^[2]

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah biaya dari masing-masing unit pembangkit. Dan pembatas yang paling penting adalah jumlah *output* dari masing-masing unit pembangkit sama dengan beban di konsumen. Yang menjadi permasalahan adalah meminimumkan total biaya F_T dengan memperhatikan pembatas ϕ bahwa daya yang dihasilkan oleh pembangkit

sama dengan beban yang diterima. Secara matematika pernyataan yang tersebut di atas dapat dinyatakan dengan persamaan sebagai berikut :

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Dari batasan pertidaksamaan pembatas di atas dapat diperluas menjadi :

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

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

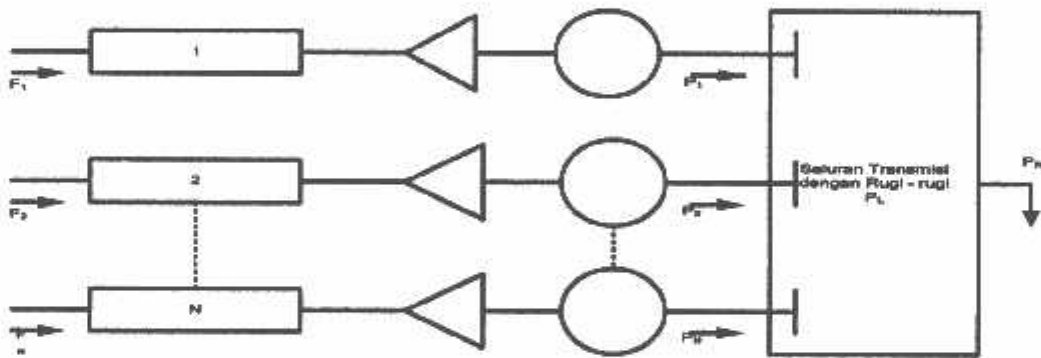
$$\frac{\partial F_i}{\partial P_i} \geq \lambda \quad \text{untuk } P_i = P_{i \min} \dots \dots \dots (2.47.)$$

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

2.7.4. Economic Dispatch Dengan Memperhitungkan Rugi-Rugi Transmisi^[2]

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

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 dan rugi-rugi transmisi.



Gambar 2.11.
N buah Pembangkit Thermal Melayani Beban P_R Melalui Saluran Transmisi^[2]

$$P_R + P_L - \sum_{i=1}^N P_i = \phi = 0 \dots\dots\dots (2.48.)$$

$$L = F_T + \lambda \phi \dots\dots\dots (2.49.)$$

$$\phi = \sum_{i=1}^N P_i - P_R - P_L = 0 \dots\dots\dots (2.50.)$$

Persamaan La Grange nya adalah :

$$L = \sum_{i=1}^N F_i - \lambda \left(\sum_{i=1}^N P_i - P_R - P_L \right) \dots\dots\dots (2.51.)$$

$$\frac{\partial L}{\partial P_i} = \frac{dF_i}{dP_i} - \lambda \left(1 - \frac{\partial P_L}{\partial P_i} \right) = 0 \dots\dots\dots (2.52.)$$

BAB III

OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY*

3.1. Analisa Aliran Daya^{[3][4]}

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

Tujuan dari analisa aliran daya adalah:

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

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

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

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

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

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

2. Bus Generator (PV)

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

3. Bus Slack

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

Daya total yang mengalir pada setiap bus dituliskan sebagai berikut :

$$S_k = P_k + jQ_k = V_k \cdot I_k^* \dots \dots \dots (3.1.)$$

Atau

$$P_{kj}Q_k = V_k \cdot \sum_{n=1}^N Y_{kn} V_n \dots\dots\dots (3.2)$$

Dari persamaan $V_k = v_k + j\delta_k$ dan $Y_{k11} = G_{k11} - jB_{k11}$, maka persamaan 3.2. menjadi :

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

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

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

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

3.1.2. Metode Newton Rapshon^[5]

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

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

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

Selisih daya dihitung dengan persamaan (3.6.) dan persamaan (3.7.) digunakan untuk menghitung nilai perubahan parameter tegangan bus, yaitu $\Delta|V_k|$

dan $\Delta\delta_k$, yaitu dengan menggunakan elemen Jacobian, sehingga koreksi terhadap nilai parameter tegangan yang telah ditetapkan nilai awal sebelumnya. Elemen Jacobian sendiri merupakan turunan parsial P dan Q terhadap masing-masing variabel pada persamaan (3.4.) dan (3.5.), yang dalam bentuk matriks dituliskan sebagai :

$$\begin{array}{c}
 \left[\begin{array}{c} \Delta P_1 \\ \dots \\ \Delta P_{n-1} \\ \Delta Q_1 \\ \dots \\ \Delta Q_{n-1} \end{array} \right] = \left[\begin{array}{cccccc}
 \frac{\partial P_1}{\partial v_1} & \dots & \frac{\partial P_1}{\partial v_{n-1}} & \frac{\partial P_1}{\partial \delta_1} & \dots & \frac{\partial P_1}{\partial \delta_{n-1}} \\
 \dots & \dots & \dots & \dots & \dots & \dots \\
 \frac{\partial P_{n-1}}{\partial v_1} & \dots & \frac{\partial P_{n-1}}{\partial v_{n-1}} & \frac{\partial P_{n-1}}{\partial \delta_1} & \dots & \frac{\partial P_{n-1}}{\partial \delta_{n-1}} \\
 \frac{\partial Q_1}{\partial v_1} & \dots & \frac{\partial Q_1}{\partial v_{n-1}} & \frac{\partial Q_1}{\partial \delta_1} & \dots & \frac{\partial Q_1}{\partial \delta_{n-1}} \\
 \dots & \dots & \dots & \dots & \dots & \dots \\
 \frac{\partial Q_{n-1}}{\partial v_1} & \dots & \frac{\partial Q_{n-1}}{\partial v_{n-1}} & \frac{\partial Q_{n-1}}{\partial \delta_1} & \dots & \frac{\partial Q_{n-1}}{\partial \delta_{n-1}}
 \end{array} \right] \left[\begin{array}{c} \Delta v_1 \\ \dots \\ \Delta v_{n-1} \\ \Delta \delta_1 \\ \dots \\ \Delta \delta_{n-1} \end{array} \right]
 \end{array}$$

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

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

Perubahan nilai tegangan bus dijumlahkan dengan nilai tegangan bus sebelumnya, yang kemudian nilai tegangan bus terbaru ini digunakan untuk

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

3.2. *Evolutionary Programming* dan *Evolutionary Strategy*^{[6][7][8]}

3.2.1. *Evolutionary Programming*^{[6][7][8]}

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

Evolutionary programming ditemukan oleh Lawrence.J. Fogel pada tahun 1960 yang 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*.

Sebelum *evolutionary programming* 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 *Evolutionary programming* yaitu inisialisasi (*initialization*), statistik (*statistics*), mutasi (*mutation*), dan kompetisi (*competition*) untuk menciptakan himpunan titik-titik solusi. Untuk memeriksa hasil optimasi, kita membutuhkan fungsi *fitness* yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses induk harus digunakan untuk reproduksi, mutasi dan kompetisi untuk menciptakan keturunan (*offspring*).

3.2.2. *Evolutionary Strategy*^{[6][7][8]}

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

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

Sebelum *evolutionary strategy* 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 *evolutionary strategy* yaitu proses seleksi (*selection*), *recombination*, mutasi (*mutation*), dan kompetisi (*competition*) untuk menciptakan himpunan titik-titik solusi. Untuk memeriksa hasil optimasi, kita membutuhkan fungsi *fitness* yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, mutasi dan kompetisi untuk menciptakan keturunan (*offspring*).

3.3. Parameter *Evolutionary Programming* dan *Evolutionary strategy* ^[1]

Terdapat beberapa parameter yang digunakan dalam *evolutionary programming* dan *evolutionary strategy*. Parameter tersebut digunakan untuk melihat kompleksitas dari *evolutionary programming* dan *evolutionary strategy*. Parameter yang digunakan tersebut adalah :

3.3.1. Jumlah Generasi (*MAXGEN*)^[1]

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

3.3.2. Ukuran Populasi (*POPSIZE*)^[1]

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

3.3.3. Probabilitas Mutasi (*Pm*)^[1]

Mutasi digunakan untuk meningkatkan variasi populasi digunakan untuk menentukan tingkat mutasi yang terjadi, karena frekuensi terjadinya mutasi tersebut menjadi $Pm \times POPSIZE \times N$, dimana N adalah panjang struktur/gen dalam satu individu. Probabilitas mutasi yang rendah akan menyebabkan gen-gen yang berpotensi tidak dicoba. Sebaliknya tingkat mutasi yang tinggi akan menyebabkan keturunan akan semakin mirip dengan induknya.

3.3.4. Panjang Kromosom^[1]

Panjang kromosom berbeda-beda sesuai dengan pemodelan 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. 4. Mekanisme *Evolutionary Programming* dan *Evolutionary strategy*^[1]

3.4.1. Pengkodean (*initialization*)^[1]

Langkah pertama kali yang dilakukan dalam penggunaan *evolutionary programming* dan *evolutionary strategy* adalah melakukan pengkodean atau representasi terhadap permasalahan yang akan dilakukan.

Secara umum *evolutionary programming* dan *evolutionary strategy* 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 *evolutionary programming* dan *evolutionary strategy* untuk awal melakukan pencarian.

3.4.2. Fungsi Evaluasi (Fungsi *Fitness*)^[1]

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

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

3.4.3. Seleksi (*selection*)^[10]

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

3.4.3.1. *Roulette Wheel Selection*^[10]

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.

3.4.3.2. *Rank Selection*^[10]

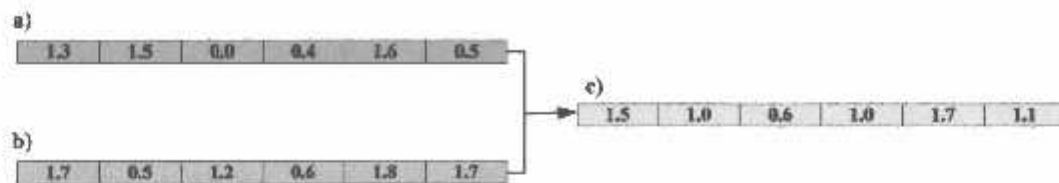
Apabila fitness yang dimiliki oleh suatu kromosom dalam populasi berbeda terlalu jauh dari kromosom lainnya maka hal ini dapat menjadi permasalahan. Misalnya bila kromosom terbaik mempunyai *fitness* yang menyebabkan besarnya tempat yang dimilikinya dalam *roulette wheel* sebesar 90% maka kromosom-kromosom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

Rank selection pertama kali merangking populasi dan kemudian setiap kromosom diberi nilai fitness baru berdasarkan hasil rangking tersebut. Yang pertama akan mempunyai *fitness* 1, yang kedua akan mempunyai fitness 2 dan seterusnya sampai yang terakhir akan mempunyai *fitness* N. Dengan demikian semua kromosom akan mempunyai peluang untuk diseleksi..

3.4.4. Rekombinasi (*Recombination*)^[10]

Khusus dalam *evolutionary strategy* setelah melakukan pengkodean dan evaluasi, individual dipilih secara acak menjadi *parents*. Setelah melakukan seleksi *parent* kemudian melakukan *Intermediate recombination*. dimana vektor

dari dua orang tua (*parent*) dirata-ratakan bersama-sama, unsur demi unsur, untuk membentuk suatu keturunan yang baru.



Gambar 3.1^[10]

Intermediate recombination dari Induk (*parent*) a) dan b) menjadi Anak (*offspring*) c)

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

3.4.5. Mutasi (*Mutation*)^[10]

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* dan

evolutionary strategy menggunakan operator *Gaussian mutation*, dimana setiap individu akan terpilih secara acak untuk mengalami mutasi berdasarkan nomor acak *Gaussian* untuk menciptakan individu baru (*offspring*).



Gambar 3.2.^[10]

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

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

3.4.6. Kompetisi (*Competition*)^[11]

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

3.5. Formulasi Masalah Optimasi Daya Reaktif^[1]

Permasalahan perencanaan optimasi daya reaktif adalah untuk meminimalkan biaya operasional dan biaya instalasi pembangkitan daya reaktif, sebagai berikut:

$$C_Q = C_P + C_I \dots\dots\dots (3.9.)$$

$$C_P = \sum_{i \in N_l} d_l C_F^i \dots\dots\dots (3.10.)$$

Dimana:

C_I = Biaya daya reaktif pembangkitan.

C_F^i = Biaya operasi daya nyata pada tingkatan l beban

d_l = Jangka waktu tingkatan l beban

N_l = Banyaknya tingkatan beban

Dengan subyek:

$$\underline{P}_{gi} \leq P_{gi} \leq \bar{P}_{gi} \quad (\text{Daya nyata generator})$$

$$\underline{Q}_{gi} \leq Q_{gi} \leq \bar{Q}_{gi} \quad (\text{Daya reaktif generator})$$

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

Tegangan bus beban dan daya reaktif generator merupakan variabel keadaan, yang dapat dibatasi dengan menambahkannya sebagai bentuk penalti kuadrat ke fungsi obyektif sebagai berikut:

$$C_P = \sum_{i \in N_g} d_l \left\{ C_F^i + \sum_{i \in N_l} \lambda_{v_i} (V_i - Sat(V_i))^2 + \sum_{i \in N_g} \lambda_{q_i} (Q_{gi} - Sat(Q_{gi}))^2 \right\} \dots (3.11.)$$

Dimana; $sat(x)$ adalah fungsi saturasi sebagai berikut:

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

3.6. Adaptasi *Evolutionary Programming* dan *Evolutionary Strategy* ke Masalah Optimasi Daya Reaktif⁽³⁾

Evolutionary programming dan *evolutionary strategy* adalah suatu pada mekanisme seleksi alam. Individu dari sebuah populasi dikodekan secara real, Populasi pertama dibangkitkan secara random. Untuk metode *evolutionary programming* generasi baru dibuat dengan mengaplikasikan operator-operator berikut terhadap sebuah populasi yaitu ; inisialisasi, mutasi dan kompetisi. Sedangkan untuk metode *evolutionary strategy* generasi baru dibuat dengan mengaplikasikan operator berikut terhadap sebuah populasi yaitu ; inisialisasi, rekombinasi, mutasi dan kompetisi dimana dari kedua metode ini tingkat pengoptimalan setiap calon solusi atau individu diukur dengan ketepatannya tergantung dari fungsi tujuan (*objective function*) permasalahan.

3.6.1. Representasi Solusi⁽¹⁾

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

3.6.2. Inisialisasi (*Initialization*)^[1]

Populasi awal diambil secara acak dari daerah variabel kontrol dari batasan atas sampai batasan bawah. Nilai *fitness* f_i diperoleh menurut fungsi obyektif.

3.6.3. Statistik (*Statistics*)^[1]

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

3.6.4. Mutasi (*Mutation*)^[1]

3.6.4.1. Mutasi *Evolutionary Programming*^[1]

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

$$P_{i+m,j} = P_{i,j} + N(0, \beta (\bar{\chi}_j - \underline{\chi}_j) \frac{f_i}{f_{\max}}), j = 1, 2, \dots, n \dots \dots \dots (3.12.)$$

Dimana :

- n = Banyaknya variabel keputusan di dalam setiap individu.
- $P_{i,j}$ = Elemen j^{th} unsur dari I^{th}
- $N(\mu, \sigma^2)$ = Variabel acak dengan menggunakan μ dan σ^2
- f_{\max} = Nilai *fitness* terbesar dari statistik, $\bar{\chi}_j$ dan $\underline{\chi}_j$
- β = Skala mutasi, $0 < \beta \leq 1$

3.6.4.2. Mutasi *Evolutionary strategy*^[1]

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

$$P_{i+m,j} = P_{i,j} + N(0, \beta \nabla_{dev}), \quad j = 1, 2, \dots, n, \dots \dots \dots (3.13.)$$

Dimana:

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

β = skala mutasi ($0 < \beta < 1$)

∇_{dev} = Variabel deviasi

3.6.5. Kompetisi (*Competition*)^[1]

Di dalam proses kompetisi, fitness individu dari populasi yang berukuran $2m$ disortir didalam order turunan. Dan m individu yang pertama dijaga sebagai orang tua untuk proses mutasi selanjutnya. Nilai berat W_i dari individu yang ke i dihitung menurut kompetisi berikut ini:

$$W_i = \sum_{i=1}^N W_{i,j} \dots \dots \dots (3.14.)$$

Dimana N adalah nomor kompetisi generasi secara acak. $W_{i,j}$ untuk yang manapun, 0 untuk kalah atau 1 untuk menang sebagai individu ke i yang bersaing dengan individu ke r yang dipilih secara acak di dalam populasi yang dikombinasikan itu. Nilai $W_{i,j}$ disampaikan sesuai persamaan berikut:

$$W_{i,j} = \begin{cases} 1 & \text{jika } U_i < \frac{f_r}{f_r + f_i} \\ 0 & \text{jika sebaliknya} \end{cases} \dots \dots \dots (3.15.)$$

Dimana:

f_r = Fitnes dari individu ke r yang dipilih secara acak

f_i = Fitnes dari individu ke i

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

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

3.6.6. Konvergensi (*Convergence*)^[1]

Jika kondisi konvergen tidak dijumpai, proses mutasi dan kompetisi akan berjalan lagi. Nomor generasi maksimum dapat digunakan untuk kondisi konvergen. Kriteria lain, seperti rasio dari *fitnes* rata-rata dan maksimum dari populasi dihitung berulang-ulang sampai :

$$\{f_{avg} / f_{max}\} \geq \delta \dots\dots\dots(3.17.)$$

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

BAB IV HASIL DAN ANALISA HASIL

4.1. Program Komputer Optimasi Daya Reaktif Menggunakan Metode *Evolutionary Programming* dan *Evolutionary Strategy*

Program komputer dalam skripsi ini dijalankan dengan menggunakan bahasa pemrograman Borland Delphi versi 7.0 dengan sistem operasi Windows XP profesional dan diaplikasikan pada komputer AMD Athlon 2.4 GHz dengan memori 256 Mb.

4.1.1. Algoritma Program

Algoritma program dasar permasalahan dilakukan dengan langkah-langkah sebagai berikut :

1. Masukkan inputan data pembangkitan $\{P_{\max}, P_{\min}, \text{konstanta biaya}\}$, data beban yang meliputi $\{\text{tegangan (V), sudut fasa tegangan } (\delta), \text{ daya aktif (P), daya reaktif (Q)}\}$, dan data impedansi saluran.
2. Melakukan proses *load flow* metode *newton raphson* dan proses metode *evolutionary programming* dan *evolutionary strategy*.
3. Apakah hasil perhitungan sudah optimal.
4. Jika “Tidak” kembali ke langkah 2.
5. Jika “Ya” maka lanjutkan kelangkah 6.
6. Hitung total biaya pembangkitan.
7. Berhenti.

Algoritma program optimasi menggunakan metode *evolutionary programming* dan *evolutionary strategy*

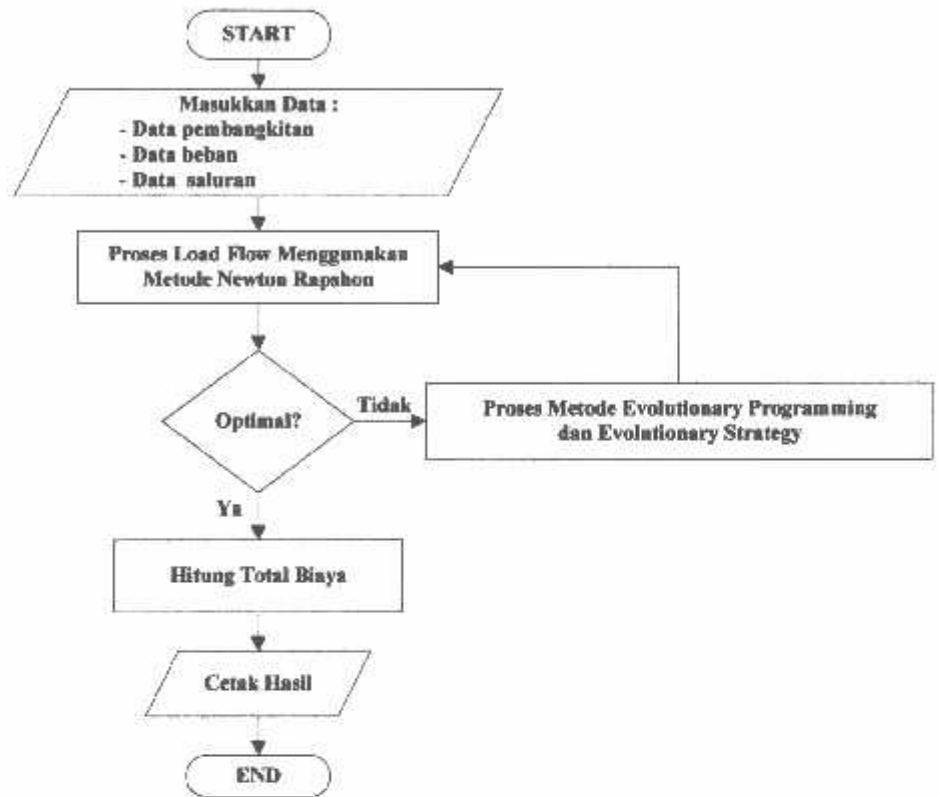
1. Inputkan data pembangkitan $\{P_{\text{mak}}, P_{\text{min}}, \text{konstanta biaya}\}$, data beban yang meliputi $\{\text{tegangan (V)}, \text{sudut fasa tegangan } (\delta), \text{ daya aktif (P)}, \text{ daya reaktif (Q)}\}$, dan data impedansi saluran.
 2. Menentukan parameter inputan *evolutionary programming* dan *evolutionary strategy* yang meliputi jumlah populasi, maksimum generasi, nilai kemungkinan mutasi, dan panjang kromosom tiap-tiap individu.
 3. Generasi = 0, Populasi = 0.
 4. Melakukan inisialisasi calon solusi.
 5. Menghitung *fitness* dari kromosom tiap-tiap individu.
 6. Melakukan proses statistik.
 7. Melakukan proses seleksi.
 8. Melakukan proses mutasi *evolutionary programming*.
 9. Menghitung *fitness offspring* dari *evolutionary programming*.
 10. Melakukan proses rekombinasi *evolutionary strategy*.
 11. Melakukan proses mutasi *evolutionary strategy*.
 12. Menghitung *fitness offspring* dari *evolutionary strategy*.
 13. Apakah *offspring evolutionary programming* lebih baik dari *offspring evolutionary strategy*.
 14. Jika “Tidak” maka lanjutkan ke langkah 16.
 15. Jika “Ya” maka nilai *offspring evolutionary programming* yang dipakai dan lanjutkan ke langkah 19.
-

16. Apakah *offspring evolutionary strategy* lebih baik dari *offspring evolutionary programming*
 17. Jika “Tidak” maka lanjutkan ke langkah 6.
 18. Jika “Ya “ maka nilai *offspring evolutionary strategy* yang dipakai dan lanjutkan ke langkah 18.
 19. Apakah *offspring* yang diinginkan sudah terpenuhi (max pop).
 20. Jika “ Tidak “ maka populasi = $pop + 1$, kembali ke langkah 7.
 21. Jika “Ya “ maka lanjutkan ke langkah 22.
 22. Melakukan proses kompetisi.
 23. Melakukan proses statistik.
 24. Apakah generasi yang diinginkan sudah terpenuhi (max Gen).
 25. Jika “ Tidak “ maka generasi = $gen + 1$, kembali ke langkah 7.
 26. Jika “ Ya “ cetak hasil dan perhitungan berhenti.
-

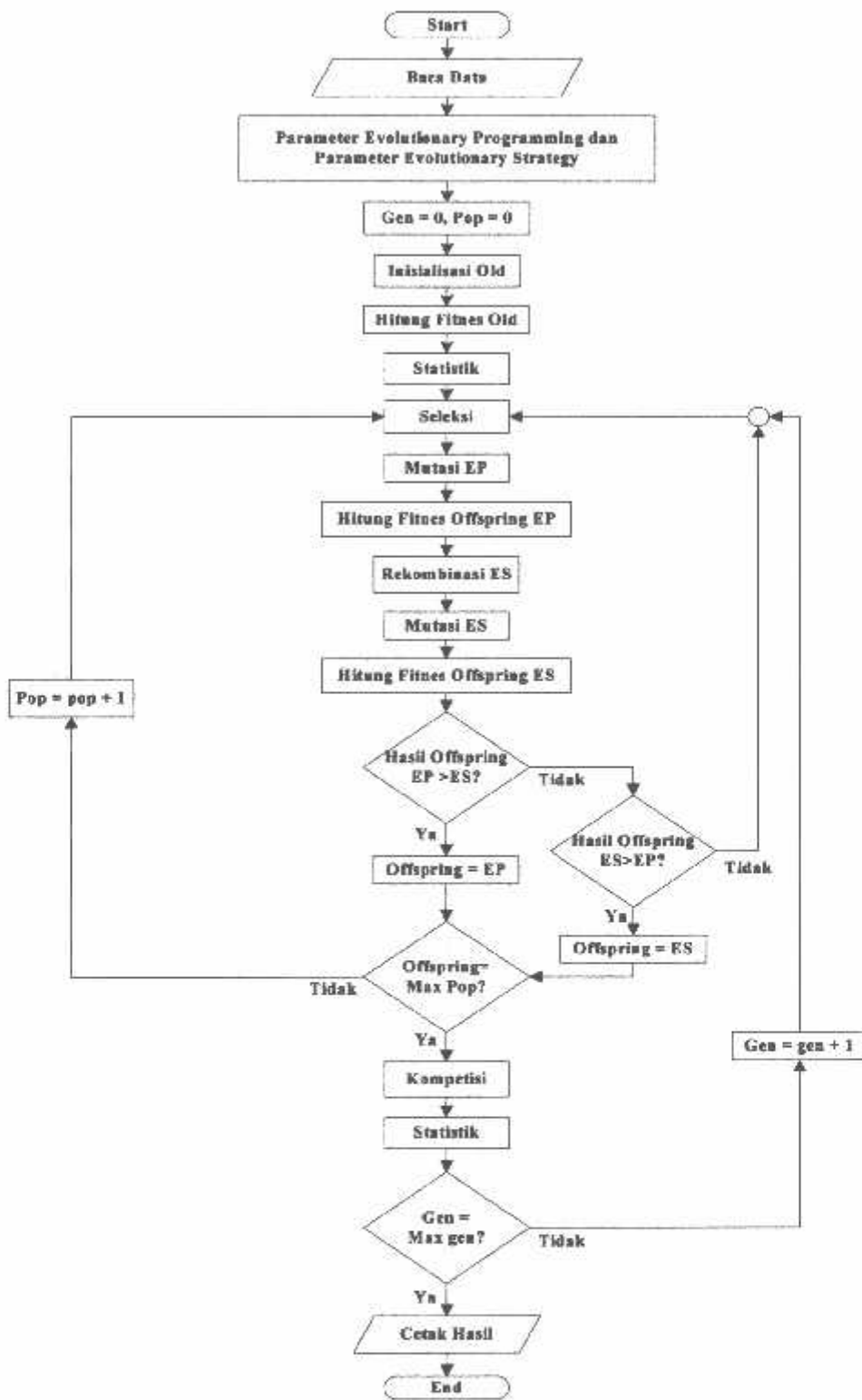
Sedangkan untuk algoritma aliran daya dengan metode *Newton Rapshon* adalah sebagai berikut :

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

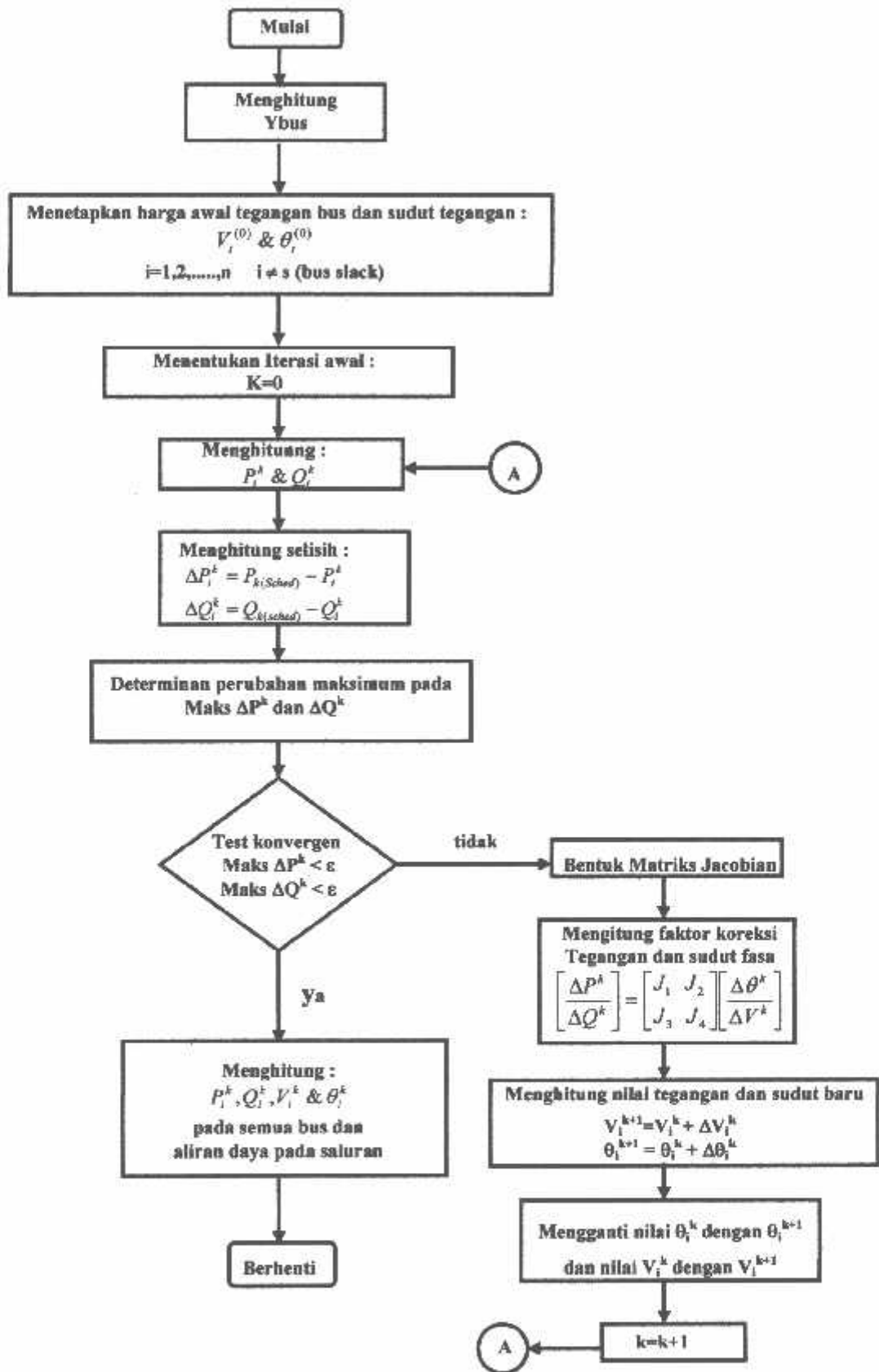
4.1.2. Flowchart Program



Gambar 4.1.
Flowchart Program Dasar Permasalahan



Gambar 4.2
Flowchart Program Optimasi Menggunakan Metode Evolutionary Programming dan Evolutionary Strategy



Gambar 4.3.
Flowchart Perhitungan Aliran Daya Menggunakan Metode Newton Rapshon

4.2. Validasi Data IEEE 30 Untuk Menyelesaikan Permasalahan Optimasi Daya Reaktif Secara Ekonomis Menggunakan Metode *Evolutionary Programming* dan *Evolutionary Strategy*

Tabel 4.1.^[11]
Data Generator dan Koefisien Biaya IEEE 30

| No. Bus | a2 | a1 | a0 | Pmin (MW) | Pmax (MW) | Qmin (MVAR) | Qmax (MVAR) |
|---------|---------|------|----|-----------|-----------|-------------|-------------|
| 1 | 0,00375 | 2 | 0 | 50 | 200 | -20 | 150 |
| 2 | 0,0175 | 1,75 | 0 | 20 | 80 | -20 | 60 |
| 5 | 0,0625 | 1 | 0 | 15 | 50 | -15 | 62 |
| 8 | 0,00834 | 3,25 | 0 | 10 | 35 | -15 | 50 |
| 11 | 0,025 | 3 | 0 | 10 | 30 | -10 | 40 |
| 13 | 0,025 | 3 | 0 | 12 | 40 | -15 | 45 |

Tabel 4.2.^[10]
Data Bus IEEE 30

| No. Bus | Tegangan | | Pembangkitan | | Pembebanan | |
|---------|----------------|-------------|--------------|----------------|------------|----------------|
| | Magnitude (pu) | Sudut (deg) | Aktif (MW) | Reaktif (MVAR) | Aktif (MW) | Reaktif (MVAR) |
| 1 | 1,05 | 0 | 0 | 0 | 0 | 0 |
| 2 | 1,0338 | 0 | 57,56 | 0 | 27,7 | 12,7 |
| 3 | 1 | 0 | 0 | 0 | 2,4 | 1,2 |
| 4 | 1 | 0 | 0 | 0 | 7,6 | 1,6 |
| 5 | 1,058 | 0 | 24,56 | 0 | 94,2 | 19 |
| 6 | 1 | 0 | 0 | 0 | 0 | 0 |
| 7 | 1 | 0 | 0 | 0 | 22,8 | 10,9 |
| 8 | 1,023 | 0 | 35 | 0 | 30 | 30 |
| 9 | 1 | 0 | 0 | 0 | 0 | 0 |
| 10 | 1 | 0 | 0 | 0 | 5,8 | 2 |
| 11 | 1,0913 | 0 | 17,93 | 0 | 0 | 0 |
| 12 | 1 | 0 | 0 | 0 | 11,2 | 7,5 |
| 13 | 1,0883 | 0 | 16,91 | 0 | 0 | 0 |
| 14 | 1 | 0 | 0 | 0 | 6,2 | 1,6 |
| 15 | 1 | 0 | 0 | 0 | 8,2 | 2,5 |
| 16 | 1 | 0 | 0 | 0 | 3,5 | 1,8 |
| 17 | 1 | 0 | 0 | 0 | 9 | 5,8 |
| 18 | 1 | 0 | 0 | 0 | 3,2 | 0,9 |
| 19 | 1 | 0 | 0 | 0 | 9,5 | 3,4 |

| | | | | | | |
|----|---|---|---|---|------|------|
| 20 | 1 | 0 | 0 | 0 | 2,2 | 0,7 |
| 21 | 1 | 0 | 0 | 0 | 17,5 | 11,2 |
| 22 | 1 | 0 | 0 | 0 | 0 | 0 |
| 23 | 1 | 0 | 0 | 0 | 3,2 | 1,6 |
| 24 | 1 | 0 | 0 | 0 | 8,7 | 6,7 |
| 25 | 1 | 0 | 0 | 0 | 0 | 0 |
| 26 | 1 | 0 | 0 | 0 | 3,5 | 2,3 |
| 27 | 1 | 0 | 0 | 0 | 0 | 0 |
| 28 | 1 | 0 | 0 | 0 | 0 | 0 |
| 29 | 1 | 0 | 0 | 0 | 2,4 | 0,9 |
| 30 | 1 | 0 | 0 | 0 | 10,6 | 1,9 |

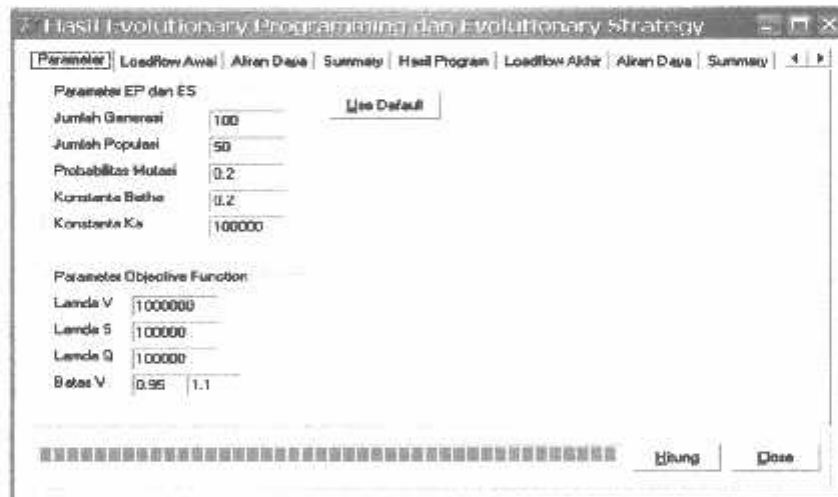
Tabel 4.3.^[10]
Data Saluran Impedansi IEEE 30

| No. Saluran | Hubungan 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 | 0,057 | 0,1737 | 0,0184 |
| 4 | 3-4 | 0,0132 | 0,0379 | 0,0042 |
| 5 | 3-5 | 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 | 0,046 | 0,0102 | 0,0102 |
| 9 | 6-7 | 0,0267 | 0,082 | 0,0085 |
| 10 | 6-8 | 0,012 | 0,042 | 0,0045 |
| 11 | 6-9 | 0,000 | 0,208 | 0,0000 |
| 12 | 6-10 | 0,000 | 0,556 | 0,0000 |
| 13 | 9-11 | 0,000 | 0,208 | 0,0000 |
| 14 | 9-10 | 0,000 | 0,11 | 0,0000 |
| 15 | 4-12 | 0,000 | 0,256 | 0,0000 |
| 16 | 12-13 | 0,000 | 0,14 | 0,0000 |
| 17 | 12-14 | 0,1231 | 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 | 0,221 | 0,1997 | 0,0000 |
| 21 | 16-17 | 0,0824 | 0,1932 | 0,0000 |
| 22 | 15-18 | 0,107 | 0,2185 | 0,0000 |
| 23 | 18-19 | 0,0639 | 0,1292 | 0,0000 |
| 24 | 19-20 | 0,034 | 0,068 | 0,0000 |
| 25 | 10-20 | 0,0936 | 0,209 | 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 | 0,1 | 0,202 | 0,0000 |
| 31 | 22-24 | 0,115 | 0,179 | 0,0000 |
| 32 | 23-24 | 0,132 | 0,27 | 0,0000 |
| 33 | 24-25 | 0,1885 | 0,3292 | 0,0000 |
| 34 | 25-26 | 0,2544 | 0,38 | 0,0000 |
| 35 | 25-27 | 0,1093 | 0,2087 | 0,0000 |
| 36 | 28-27 | 0 | 0,396 | 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 | 0,2 | 0,0210 |
| 41 | 6-28 | 0,0169 | 0,0599 | 0,0065 |

4.2.1. Hasil Validasi IEEE 30 Dengan Menggunakan Metode *Evolutionary Programming* dan *Evolutionary Strategy*

1. Tampilan Parameter Validasi



Gambar 4.4.
Tampilan Parameter Validasi IEEE 30

Hasil Evolutionary Programming dan Evolutionary Strategy

| Parameter | Loadflow Awal | Akhir Daya | Summary | Hasil Program | Loadflow Akhir | Akhir Daya | Summary | |
|-----------|---------------|------------|---------|---------------|----------------|------------|-----------|----------|
| Bus | absV (pu) | sudV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Sups (pu) | Type Bus |
| 1 | 1.10000 | 0.00000 | 167.123 | 5.961 | 0.000 | 0.000 | 0.000 | 1 |
| 2 | 1.08531 | -0.05478 | 47.290 | 46.986 | 27.700 | 12.700 | 0.000 | 2 |
| 3 | 1.08165 | -0.08210 | 0.000 | 0.000 | 2.400 | 1.200 | 0.000 | 3 |
| 4 | 1.04037 | -0.09639 | 0.000 | 0.000 | 7.600 | 1.600 | 0.000 | 3 |
| 5 | 1.07487 | -0.14148 | 50.000 | 51.826 | 54.200 | 18.000 | 0.000 | 2 |
| 6 | 1.03271 | -0.11628 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 3 |
| 7 | 1.04240 | -0.13465 | 0.000 | 0.000 | 22.900 | 10.900 | 0.000 | 3 |
| 8 | 1.02185 | -0.12225 | 10.000 | 5.144 | 30.000 | 30.000 | 0.000 | 2 |
| 9 | 1.02229 | -0.15734 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 3 |
| 10 | 0.99302 | -0.19130 | 0.000 | 0.000 | 5.800 | 2.000 | 0.000 | 3 |
| 11 | 1.07305 | -0.13740 | 10.525 | 25.804 | 0.000 | 0.000 | 0.000 | 2 |
| 12 | 0.98513 | -0.17118 | 0.000 | 0.000 | 11.200 | 7.500 | 0.000 | 3 |
| 13 | 0.97356 | -0.15201 | 13.133 | -7.520 | 0.000 | 0.000 | 0.000 | 2 |
| 14 | 0.97206 | -0.18812 | 0.000 | 0.000 | 6.200 | 1.600 | 0.000 | 3 |
| 15 | 0.98990 | -0.19215 | 0.000 | 0.000 | 8.200 | 2.500 | 0.000 | 3 |

Hitung Done

Gambar 4.5.
Hasil Load Flow

Hasil Evolutionary Programming dan Evolutionary Strategy

| Parameter | Loadflow Awal | Akhir Daya | Summary | Hasil Program | Loadflow Akhir | Akhir Daya | Summary |
|------------------------------|---------------|------------|---------|---------------|----------------|------------|---------|
| Hasil Setting Program | | | | | | | |
| No | Bus | absV (pu) | Pg (MW) | Qg (MVA) | | | |
| 1 | 1 | 1.10000 | 167.123 | 0.00 | | | |
| 2 | 2 | 1.08531 | 47.29 | 0.00 | | | |
| 3 | 5 | 1.07487 | 50.00 | 0.00 | | | |
| 4 | 8 | 1.02185 | 10.00 | 0.00 | | | |
| 5 | 11 | 1.07305 | 10.53 | 0.00 | | | |
| 6 | 13 | 0.97356 | 13.13 | 0.00 | | | |

| Rekapitulasi Biaya | |
|--------------------|-----------|
| Sebelum Optimasi | 788 \$/hr |
| Setelah Optimasi | 705 \$/hr |
| Setelah Optimasi | 67 \$/hr |

Hitung Done

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

Tabel 4.4.
Perbandingan Hasil Data Referensi Jurnal Dengan Data Optimasi

| No. Unit | No. Bus | Data Referensi | | Data Optimasi | |
|----------|---------|---------------------|-----------------------|---------------------|-----------------------|
| | | P _{Gen} MW | Q _{Gen} MVAR | P _{Gen} MW | Q _{Gen} MVAR |
| 1 | 1 | 159,826 | 5,502 | 167,123 | 5,561 |
| 2 | 2 | 53,235 | 46,915 | 47,29 | 46,986 |
| 3 | 5 | 50,031 | 52,003 | 50,00 | 51,826 |
| 4 | 8 | 11,069 | 5,237 | 10,00 | 5,144 |
| 5 | 11 | 11,002 | 25,795 | 10,53 | 25,804 |
| 6 | 13 | 12,897 | -7,932 | 13,13 | -7,290 |

Setelah dilakukan proses optimasi maka diperoleh total biaya pembangkitan dari data IEEE 30 sebesar 705,00 \$/h sedangkan dari data referensi [15] diperoleh total biaya pembangkitan sebesar 707,16 \$/h. Dari proses validasi di atas maka didapatkan *error* sebesar 0,003063.

4.3. Data Pembangkitan Termal Pada Sub Sistem Paiton-Bali

Skripsi ini akan membahas pembangkit termal yang berada pada sub sistem Paiton-Bali. Pembangkit termal 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 yang digunakan dalam perhitungan berdasarkan data dari PT. Indonesia Power dan PT. PJB.

Tabel 4.5.^[12]
Parameter Unit Pembangkit Thermal

| Nama Pembangkit | a2 | a1 | a0 | Pmin (MW) | Pmax (MW) |
|------------------|-------------|-----------|-------------|-----------|-----------|
| PLTU Paiton 1&2 | 276,246 | 58384,905 | 17350044,31 | 150 | 700 |
| PLTG Gilimanuk | 188,355772 | 214074,12 | 11705797,8 | 50 | 133,8 |
| PLTD Pesanggaran | 1913,573616 | 223579,60 | 11909964,8 | 20 | 75 |
| PLTG Pesanggaran | 15248,932 | 709162,36 | 18684292,8 | 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 :

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

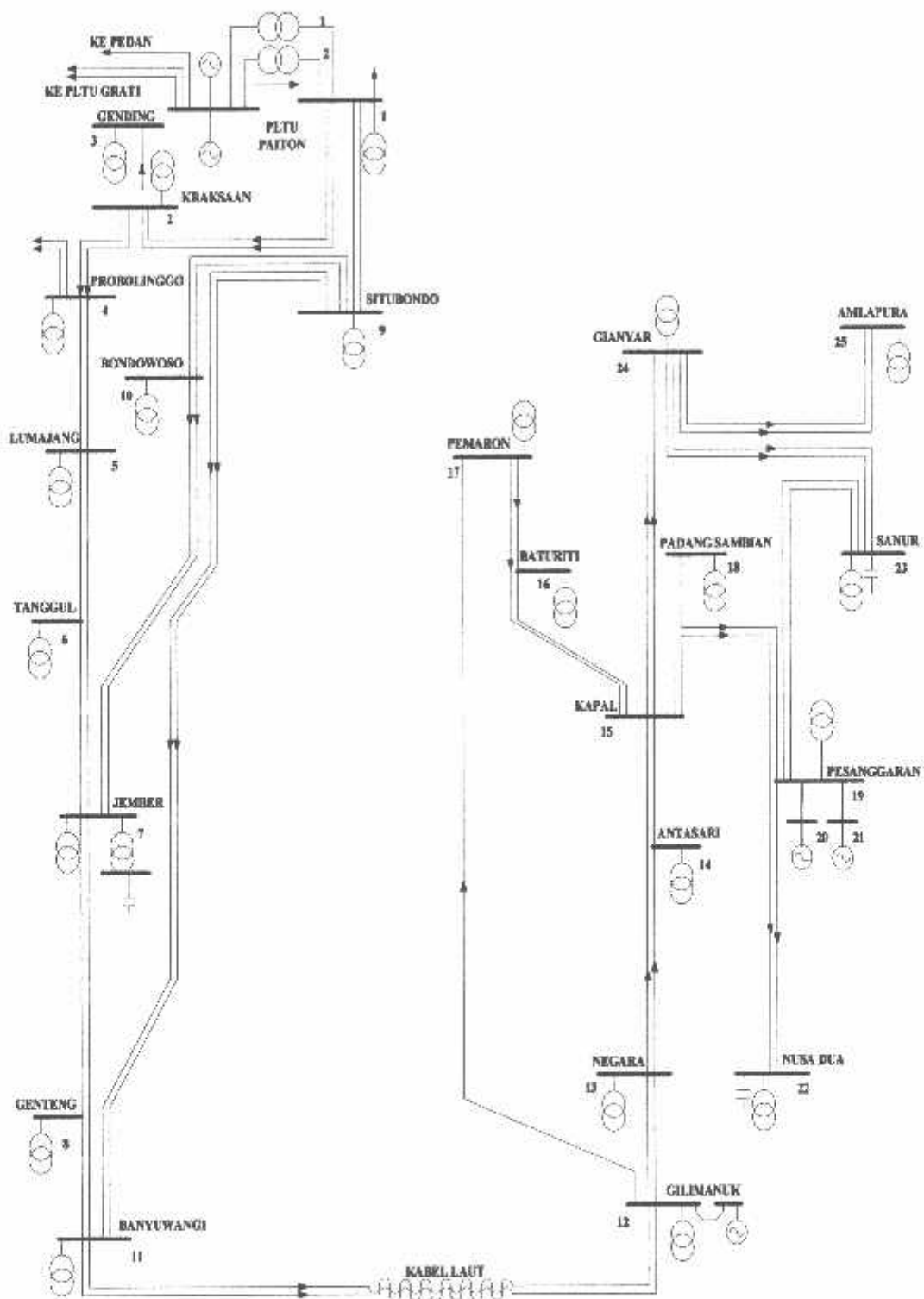
$$F_1(G_1) = 276,246 P^2 + 58384,905 P + 17350044,31$$

Sedangkan untuk biaya pengoperasian daya reaktif dari setiap pembangkit, dalam skripsi ini berdasarkan referensi maka diset sebesar US \$6,77 Mvar perjam^[15]. Kurs dollar yang dipakai sebesar Rp.9.800,- jadi biaya pengoperasian daya reaktif dari masing-masing pembangkit sebesar Rp 66.346 MVar perjam.

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-Bali

| No. | Nama Pembangkit | Persamaan Biaya Bahan Bakar (Rp/Jam) | Persamaan biaya Daya Reaktif (Rp/jam) |
|-----|-----------------|---|---------------------------------------|
| 1 | PLTU Paiton 1&2 | $F = 276,246 P^2 + 58384,905 P + 17350044,31$ | $C_r(Q_r) = 66346 (Q_r)$ |
| 2 | PLTG Gilimanuk | $F = 188,355772 P^2 + 214074,12 P + 11705797,8$ | $C_r(Q_r) = 66346 (Q_r)$ |
| 3 | PLTDPesanggaran | $F = 1913,573616 P^2 + 223579,6 P + 11909964,8$ | $C_r(Q_r) = 66346 (Q_r)$ |
| 4 | PLTGPesanggaran | $F = 15248,932 P^2 + 709162,36 P + 18684292,8$ | $C_r(Q_r) = 66346 (Q_r)$ |



Sumber : PT. PLN (persero) P3B Jawa-Bali

Gambar 4.7.
Diagram Segaris Jaringan Sistem Transmisi Listrik 150 kV Paiton-Bali

Tabel 4.7.
Penomoran Bus 150 kV Sub Sistem Paiton-Bali

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

4.4. Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton-Bali

Berdasarkan data pada referensi [14] maka dilakukan pengolahan data untuk mengubah dasar tegangan ke dalam pu. Serta perhitungan P_{load} dan Q_{load} .

Pada bus nomor 1, maka :

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

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

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

$$\text{Tegangan (pu)} = \frac{155}{150} = 1,033 \text{ pu}$$

$$P_{\text{beban}} = 11,7 + 26,4 = 38,1 \text{ MW}$$

$$Q_{\text{beban}} = 5,7 \text{ MVAR}$$

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

Tabel 4.8.^[13]

**Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton dan Bali
Hari Rabu, 30 Maret 2005 Pukul : 19:30 WIB**

| No. | Nama Bus | Tegangan | Pgenerator | Qgenerator | Pload | Qload | Tipe |
|-----|-------------|----------|------------|------------|-------|--------|------|
| | | (pu) | (MW) | (MVAR) | (MW) | (MVAR) | |
| 1 | PAITON | 1,033 | - | - | 38,1 | 5,7 | 1 |
| 2 | KRAKSAN | - | - | - | 16,5 | 6,5 | 3 |
| 3 | GENDING | - | - | - | 17,5 | 6,7 | 3 |
| 4 | PROBOLINGGO | - | - | - | 39,3 | 19,8 | 3 |
| 5 | LUMAJANG | - | - | - | 39,8 | 17,5 | 3 |
| 6 | TANGGUL | - | - | - | 23,4 | 9,3 | 3 |
| 7 | JEMBER | - | - | 47 | 58,9 | 30,3 | 3 |
| 8 | GENTENG | - | - | - | 39,8 | 21,8 | 3 |
| 9 | SITUBONDO | - | - | - | 24,4 | 5,8 | 3 |
| 10 | BONDOWOSO | - | - | - | 17,8 | 7,4 | 3 |

| | | | | | | | |
|----|-------------|-------|------|----|------|------|---|
| 11 | BANYUWANGI | - | - | - | 37,8 | 14,8 | 3 |
| 12 | GILIMANUK | 0,992 | 99,9 | - | 5,7 | 2 | 2 |
| 13 | NEGARA | - | - | - | 11,7 | 4,6 | 3 |
| 14 | ANTASARI | - | - | - | 6 | 2,8 | 3 |
| 15 | KAPAL | - | - | 50 | 69,5 | 23,5 | 3 |
| 16 | BATURITI | - | - | - | 4 | 0,5 | 3 |
| 17 | PEMARON | - | - | - | 24,8 | 8,8 | 3 |
| 18 | PDG SAMBIAN | - | - | - | 32,8 | 13,3 | 3 |
| 19 | PESANGGARAN | - | - | - | 76,4 | 28,9 | 3 |
| 20 | PLTD PSGRAN | 0,936 | 31,4 | - | - | - | 2 |
| 21 | PLTG PSGRAN | 0,936 | 84,6 | - | - | - | 2 |
| 22 | NUSADUA | - | - | 25 | 51,4 | 19,3 | 3 |
| 23 | SANUR | - | - | 25 | 60,3 | 17,7 | 3 |
| 24 | GIANYAR | - | - | - | 33,7 | 9,6 | 3 |
| 25 | AMLAPURA | - | - | - | 14,4 | 7 | 3 |

Keterangan: 1 = bus slack
 2 = bus generator
 3 = bus beban

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

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

Berdasarkan data pada referensi [14], 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.^[14]
Data Saluran Transmisi 150 kV Sub Sistem Paiton-Bali

| No. | Hubungan Bus | Jumlah Sirkuit | R (pu) | X (pu) | B (pu) |
|-----|--------------|----------------|--------|--------|--------|
| 1 | 1 – 2 | 2 | 0,0053 | 0,0179 | 0,0065 |
| 2 | 1 – 9 | 2 | 0,0048 | 0,0346 | 0,0251 |
| 3 | 2 – 3 | 1 | 0,01 | 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 | - |
| 16 | 12 – 13 | 2 | 0,0116 | 0,0034 | 0,0124 |
| 17 | 12 – 17 | 1 | 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 – 16 | 1 | 0,0362 | 0,0693 | 0,0240 |
| 22 | 15 – 17 | 1 | 0,0556 | 0,1054 | 0,0369 |
| 23 | 15 – 18 | 1 | 0,0105 | 0,0304 | 0,0112 |
| 24 | 15 – 19 | 1 | 0,0105 | 0,0304 | 0,0112 |
| 25 | 15 – 22 | 1 | 0,0293 | 0,0561 | 0,0195 |
| 26 | 15 – 24 | 2 | 0,0059 | 0,0170 | 0,0063 |
| 27 | 16 – 17 | 1 | 0,0194 | 0,0371 | 0,0129 |
| 28 | 18 – 19 | 1 | 0,0045 | 0,0130 | 0,0048 |
| 29 | 19 – 20 | 1 | 0,0020 | 0,0044 | - |
| 30 | 19 – 21 | 1 | 0,0020 | 0,0056 | - |
| 31 | 19 – 22 | 1 | 0,0127 | 0,0243 | 0,0084 |
| 32 | 19 – 23 | 2 | 0,0037 | 0,0070 | 0,0025 |
| 33 | 23 – 24 | 2 | 0,0078 | 0,0149 | 0,0052 |
| 34 | 24 – 25 | 2 | 0,0103 | 0,0298 | 0,0110 |

4.6. Prosedur Pelaksanaan Program Perhitungan

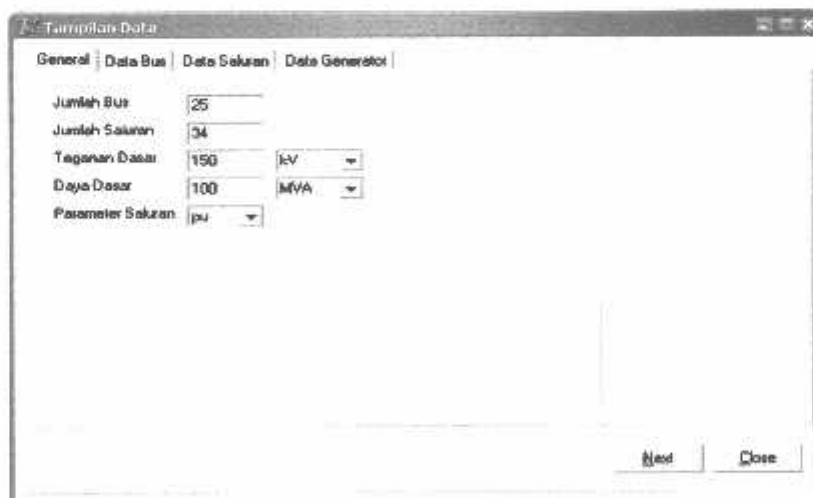
Prosedur menjalankan program perhitungan dengan menggunakan bahasa pemrograman Borland Delphi versi 7.0 dapat dilakukan sebagai berikut :

1. Menu utama dari program, tekan tombol *Open File* untuk membuka data yang sudah tersimpan



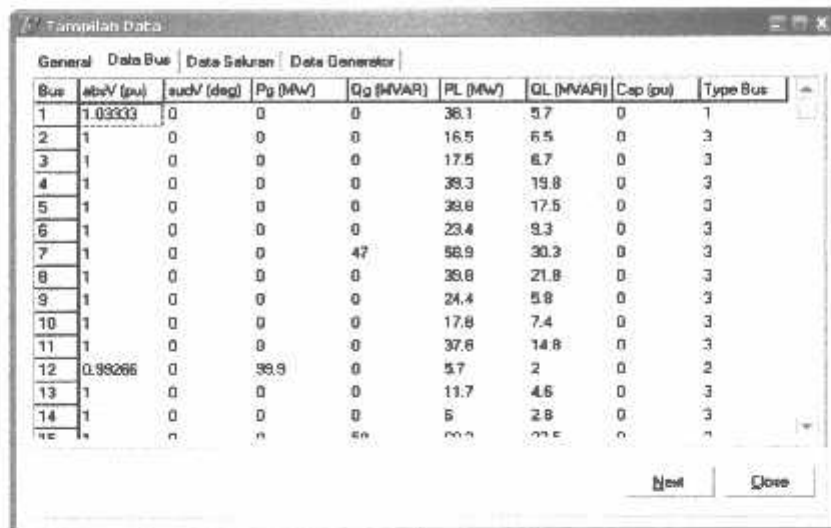
Gambar 4.8.
Menu Utama Program

2. Tampilan Data



Gambar 4.9.
Tampilan Data

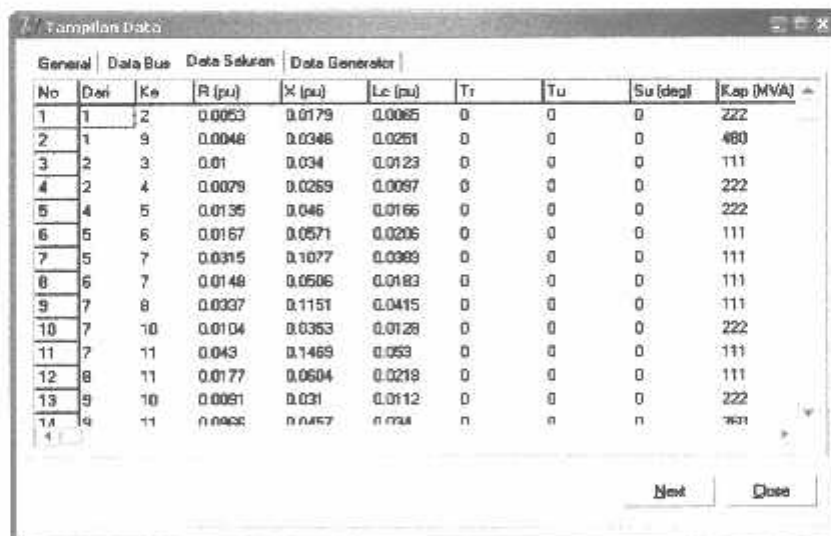
3. Kemudian tekan tombol *Data Bus*



| Bus | absV (pu) | audV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Cap (pu) | Type Bus |
|-----|-----------|------------|---------|-----------|---------|-----------|----------|----------|
| 1 | 1.03333 | 0 | 0 | 0 | 38.1 | 5.7 | 0 | 1 |
| 2 | 1 | 0 | 0 | 0 | 16.5 | 6.5 | 0 | 3 |
| 3 | 1 | 0 | 0 | 0 | 17.5 | 6.7 | 0 | 3 |
| 4 | 1 | 0 | 0 | 0 | 39.3 | 19.8 | 0 | 3 |
| 5 | 1 | 0 | 0 | 0 | 39.8 | 17.5 | 0 | 3 |
| 6 | 1 | 0 | 0 | 0 | 23.4 | 9.3 | 0 | 3 |
| 7 | 1 | 0 | 0 | 47 | 59.9 | 30.3 | 0 | 3 |
| 8 | 1 | 0 | 0 | 0 | 39.8 | 21.8 | 0 | 3 |
| 9 | 1 | 0 | 0 | 0 | 24.4 | 5.8 | 0 | 3 |
| 10 | 1 | 0 | 0 | 0 | 17.8 | 7.4 | 0 | 3 |
| 11 | 1 | 0 | 0 | 0 | 37.8 | 14.8 | 0 | 3 |
| 12 | 0.99266 | 0 | 99.9 | 0 | 5.7 | 2 | 0 | 2 |
| 13 | 1 | 0 | 0 | 0 | 11.7 | 4.6 | 0 | 3 |
| 14 | 1 | 0 | 0 | 0 | 5 | 2.8 | 0 | 3 |
| 15 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 3 |

Gambar 4.10.
Tampilan Data Bus

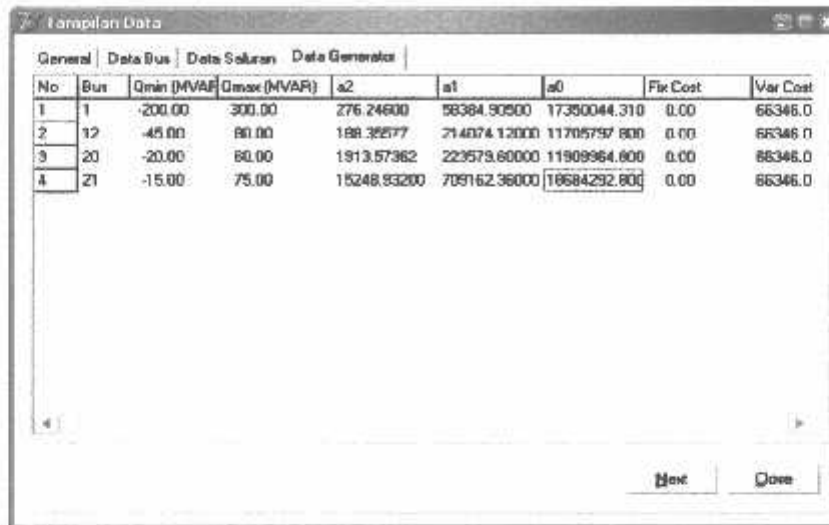
4. Kemudian tekan tombol *Data Saluran*



| No | Da | Ke | R (pu) | X (pu) | Lc (pu) | Tl | 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 | 0.0219 | 0 | 0 | 0 | 111 |
| 13 | 9 | 10 | 0.0091 | 0.031 | 0.0112 | 0 | 0 | 0 | 222 |
| 14 | 9 | 11 | 0.0093 | 0.0357 | 0.0134 | 0 | 0 | 0 | 222 |

Gambar 4.11.
Tampilan Data Saluran

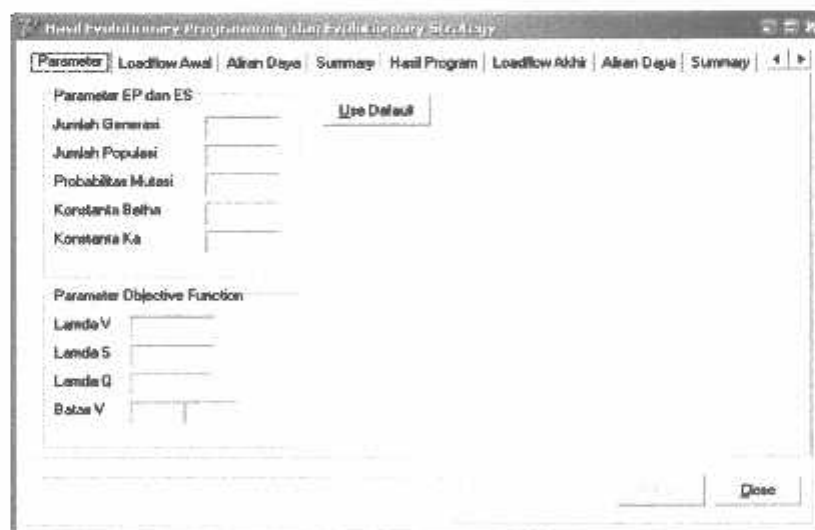
5. Kemudian tekan *Data Generator*



| No | Bus | Qmin (MVAR) | Qmax (MVAR) | a2 | a1 | a0 | Fix Cost | Var Cost |
|----|-----|-------------|-------------|-------------|--------------|--------------|----------|----------|
| 1 | 1 | -200.00 | 300.00 | 276.24600 | 58384.90500 | 17390044.310 | 0.00 | 66346.0 |
| 2 | 12 | -45.00 | 80.00 | 188.32577 | 214074.12000 | 11705797.800 | 0.00 | 66346.0 |
| 3 | 20 | -20.00 | 60.00 | 1913.57362 | 223573.60000 | 11909964.600 | 0.00 | 66346.0 |
| 4 | 21 | -15.00 | 75.00 | 15248.93200 | 709162.36000 | 18684292.800 | 0.00 | 66346.0 |

Gambar 4.12.
Tampilan Data Generator

6. Tekan tombol *Parameter* untuk melihat parameter-parameter *Evolutionary Programming* dan *Evolutionary Strategy*



Detail Evolutionary Programming dan Evolutionary Strategy

Parameter Loadflow Awal Aiken Daya Summary Hasil Program Loadflow Akhir Aiken Daya Summary

Parameter EP dan ES

Jumlah Generasi Use Default

Jumlah Populasi

Probabilitas Mutasi

Konstanta Belta

Konstanta Ka

Parameter Objective Function

Lambda V

Lambda S

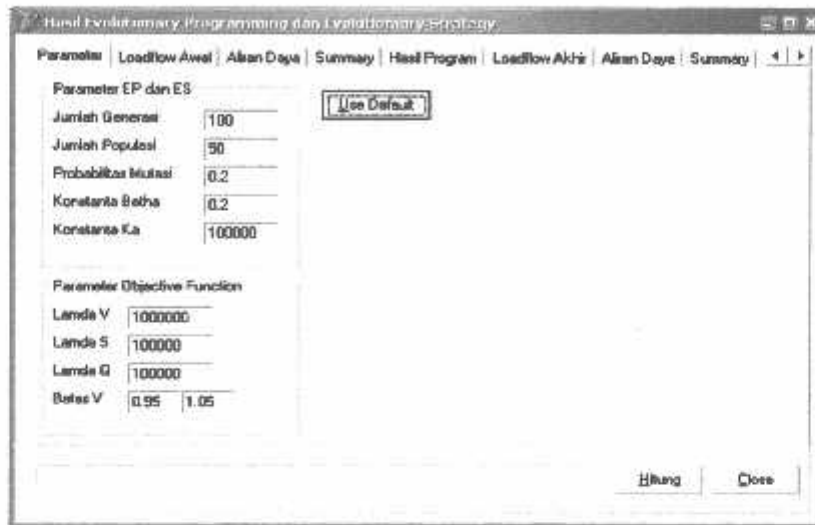
Lambda Q

Batas V

Close

Gambar 4.13.
Tampilan Parameter

7. Tekan *Used Default* untuk mengetahui nilai-nilai parameter *Evolutionary Programming* dan *Evolutionary Strategy*, dan tekan hitung.



Gambar 4.14.
Tampilan *Parameter* dengan menekan tombol *Used Default*

8. Tekan tombol *Loadflow Awal* untuk melihat hasil perhitungan *Load Flow* pada kondisi awal (sebelum optimasi)

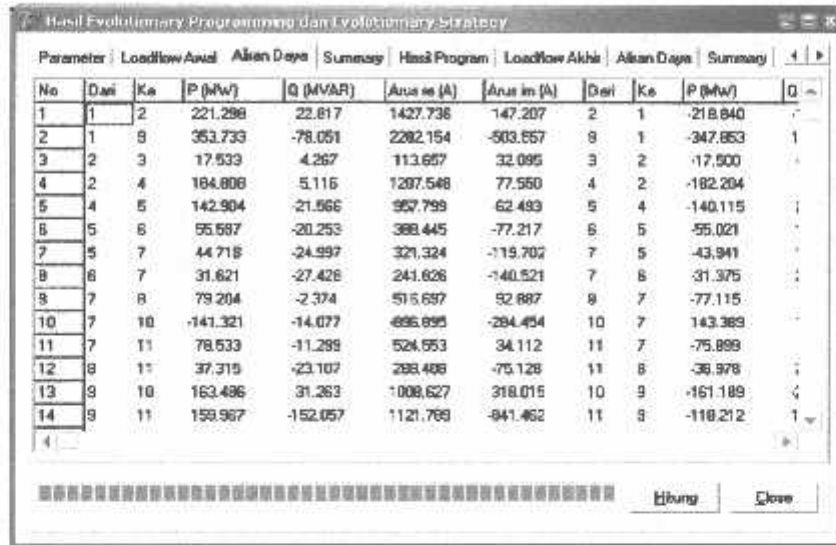
The screenshot shows the same software window, but with the "Loadflow Awal" tab selected. It displays a table of bus data for 15 buses.

| Bur | abv (pu) | sudV (deg) | Pg (MW) | Gg (MVAR) | PL (MW) | QL (MVAR) | Supr (pu) | Type Bus |
|-----|----------|------------|---------|-----------|---------|-----------|-----------|----------|
| 1 | 1.03333 | 0.00000 | 673.131 | 49.534 | 38.100 | 5.700 | 0.000 | 1 |
| 2 | 1.01858 | -0.03646 | 0.000 | 0.000 | 16.500 | 6.500 | 0.000 | 3 |
| 3 | 1.01503 | -0.04169 | 0.000 | 0.000 | 17.500 | 6.700 | 0.000 | 3 |
| 4 | 1.00390 | -0.08463 | 0.000 | 0.000 | 39.300 | 19.600 | 0.000 | 3 |
| 5 | 0.99603 | -0.15311 | 0.000 | 0.000 | 39.800 | 17.500 | 0.000 | 3 |
| 6 | 0.99776 | -0.18913 | 0.000 | 0.000 | 23.400 | 9.300 | 0.000 | 3 |
| 7 | 1.00626 | -0.20784 | 0.000 | 47.000 | 59.900 | 30.300 | 0.000 | 3 |
| 8 | 0.98177 | -0.29962 | 0.000 | 0.000 | 39.800 | 21.800 | 0.000 | 3 |
| 9 | 1.04925 | -0.11648 | 0.000 | 0.000 | 24.400 | 5.600 | 0.000 | 3 |
| 10 | 1.02647 | -0.16081 | 0.000 | 0.000 | 17.900 | 7.400 | 0.000 | 3 |
| 11 | 0.98832 | -0.32669 | 0.000 | 0.000 | 37.800 | 14.600 | 0.000 | 3 |
| 12 | 0.99266 | -0.33700 | 99.900 | 210.171 | 5.700 | 2.000 | 0.000 | 2 |
| 13 | 0.98888 | -0.34517 | 0.000 | 0.000 | 11.700 | 4.600 | 0.000 | 3 |
| 14 | 0.94722 | -0.42934 | 0.000 | 0.000 | 6.000 | 2.900 | 0.000 | 3 |
| 15 | 0.93892 | -0.47038 | 0.000 | 50.000 | 69.300 | 23.500 | 0.000 | 3 |

Buttons "Hitung" and "Close" are at the bottom right.

Gambar 4.15.
Tampilan *Load Flow* pada kondisi awal (Sebelum Optimasi)

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



| No | Dari | Ke | P (MW) | Q (MVAR) | Arus ke (A) | Arus in (A) | Dari | Ke | P (MW) | Q |
|----|------|----|----------|----------|-------------|-------------|------|----|----------|---|
| 1 | 1 | 2 | 221.298 | 22.617 | 1427.738 | 147.207 | 2 | 1 | -218.840 | - |
| 2 | 1 | 8 | 353.733 | -78.051 | 2282.154 | -503.657 | 8 | 1 | -347.863 | - |
| 3 | 2 | 3 | 17.533 | 4.267 | 113.657 | 32.095 | 3 | 2 | -17.500 | - |
| 4 | 2 | 4 | 184.808 | 5.116 | 1207.548 | 77.550 | 4 | 2 | -182.204 | - |
| 5 | 4 | 5 | 142.904 | -21.566 | 957.799 | 62.493 | 5 | 4 | -140.115 | - |
| 6 | 5 | 6 | 55.587 | -20.253 | 388.445 | -77.217 | 6 | 5 | -55.021 | - |
| 7 | 5 | 7 | 44.718 | -24.997 | 321.324 | -119.702 | 7 | 5 | -43.941 | - |
| 8 | 6 | 7 | 31.621 | -27.428 | 241.626 | -140.521 | 7 | 6 | -31.375 | - |
| 9 | 7 | 8 | 79.204 | -2.374 | 515.657 | 52.887 | 8 | 7 | -77.115 | - |
| 10 | 7 | 10 | -141.321 | -14.077 | -896.695 | -284.454 | 10 | 7 | 143.389 | - |
| 11 | 7 | 11 | 78.533 | -11.298 | 524.553 | 34.112 | 11 | 7 | -75.899 | - |
| 12 | 8 | 11 | 37.315 | -23.107 | 238.408 | -75.128 | 11 | 8 | -36.978 | - |
| 13 | 9 | 10 | 163.486 | 31.263 | 1008.627 | 318.015 | 10 | 9 | -161.189 | - |
| 14 | 9 | 11 | 159.967 | -152.057 | 1121.769 | -841.462 | 11 | 9 | -118.212 | - |

Gambar 4.16.
Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi)

10. Tekan tombol *Summary* untuk melihat hasil akhir jumlah pembangkitan, pembebanan, rugi-rugi (sebelum optimasi).



| Summary Loadflow | | |
|---------------------|--------------------|-----|
| Jumlah Pembangkitan | 829.031+ j 329.493 | MVA |
| Jumlah Pembebanan | 743.806+ j 283.600 | MVA |
| Jumlah Rugi-Rugi | 85.231+ j 45.883 | MVA |
| Iterasi | 4 | |
| Waktu Hitung | 0:0:0.15 | |

Gambar 4.17.
Tampilan Summary (Sebelum Optimasi)

11. Tekan tombol *Hasil Program* untuk mengetahui hasil perhitungan daya aktif (P), daya reaktif (Q) dan rekapitulasi biaya.

The screenshot shows the 'Hasil Program' tab with the following data:

| No | Bus | absV (pu) | Pg (MW) | Qg (MVA) |
|----|-----|-----------|---------|----------|
| 1 | 1 | 1.05000 | 602.576 | 0.00 |
| 2 | 7 | 0.00000 | 0.00 | 47.00 |
| 3 | 12 | 0.98052 | 133.80 | 0.00 |
| 4 | 15 | 0.00000 | 0.00 | 44.10 |
| 5 | 20 | 0.96069 | 75.00 | 0.00 |
| 6 | 21 | 0.95892 | 15.00 | 0.00 |
| 7 | 22 | 0.00000 | 0.00 | 25.00 |
| 8 | 23 | 0.00000 | 0.00 | 25.00 |

| Rekapitulasi Biaya | | |
|--------------------|-------------|---------|
| Sebelum Optimal | 412,711,577 | Rp./jam |
| Setelah Optimal | 281,045,483 | Rp./jam |
| Selisi Optimal | 131,666,094 | Rp./jam |

Gambar 4.18.
Tampilan Hasil Program

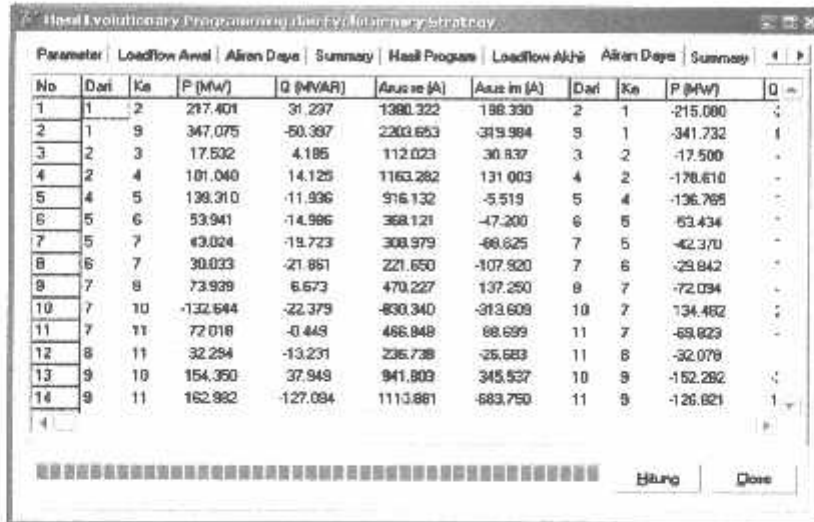
12. Kemudian tekan tombol *Loadflow Akhir*, untuk melihat hasil *Load Flow* pada kondisi akhir (setelah optimasi)

The screenshot shows the 'Loadflow Akhir' tab with the following data:

| Bus | absV (pu) | angV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Supr (pu) | Type Bus |
|-----|-----------|------------|---------|-----------|---------|-----------|-----------|----------|
| 1 | 1.05000 | 0.00000 | 602.576 | -13.480 | 36.100 | 5.700 | 0.000 | 1 |
| 2 | 1.03419 | -0.03428 | 0.000 | 0.000 | 15.500 | 5.500 | 0.000 | 3 |
| 3 | 1.03070 | -0.03936 | 0.000 | 0.000 | 17.500 | 5.700 | 0.000 | 3 |
| 4 | 1.01745 | -0.07945 | 0.000 | 0.000 | 30.300 | 10.600 | 0.000 | 3 |
| 5 | 1.00565 | -0.14347 | 0.000 | 0.000 | 39.800 | 17.500 | 0.000 | 3 |
| 6 | 1.00485 | -0.17609 | 0.000 | 0.000 | 23.400 | 9.300 | 0.000 | 3 |
| 7 | 1.01037 | -0.19399 | 0.000 | 47.000 | 58.900 | 30.300 | 0.000 | 3 |
| 8 | 0.97661 | -0.27660 | 0.000 | 0.000 | 39.800 | 21.800 | 0.000 | 3 |
| 9 | 1.05628 | -0.11056 | 0.000 | 0.000 | 24.400 | 5.800 | 0.000 | 3 |
| 10 | 1.03233 | -0.15118 | 0.000 | 0.000 | 17.800 | 7.400 | 0.000 | 3 |
| 11 | 0.97790 | -0.29910 | 0.000 | 0.000 | 37.800 | 14.800 | 0.000 | 3 |
| 12 | 0.98052 | -0.30872 | 133.800 | 97.823 | 5.700 | 2.000 | 0.000 | 2 |
| 13 | 0.95531 | -0.32364 | 0.000 | 0.000 | 11.700 | 4.600 | 0.000 | 3 |
| 14 | 0.95127 | -0.42344 | 0.000 | 0.000 | 5.000 | 2.800 | 0.000 | 3 |
| 15 | 0.95317 | -0.47288 | 0.000 | 44.163 | 69.300 | 23.500 | 0.000 | 3 |

Gambar 4.19.
Tampilan Load Flow pada kondisi akhir (Setelah Optimasi)

13. Kemudian tekan tombol aliran daya untuk melihat kondisi aliran daya pada kondisi akhir (setelah optimasi).



| Parameter | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary | | | |
|-----------|---------------|-------------|----------|---------------|----------------|-------------|---------|----|----------|----------|
| No | Dari | Ke | P (MW) | Q (MVAR) | Ausur (A) | Ausur (A) | Dari | Ke | P (MW) | Q (MVAR) |
| 1 | 1 | 2 | 217.401 | 31.297 | 1380.322 | 188.330 | 2 | 1 | -215.000 | - |
| 2 | 1 | 9 | 347.075 | -50.397 | 2203.653 | -319.984 | 9 | 1 | -341.732 | - |
| 3 | 2 | 3 | 17.532 | 4.185 | 112.023 | -30.837 | 3 | 2 | -17.500 | - |
| 4 | 2 | 4 | 101.040 | 14.125 | 1163.282 | 131.003 | 4 | 2 | -178.610 | - |
| 5 | 4 | 5 | 139.310 | -11.936 | 916.132 | -5.519 | 5 | 4 | -136.765 | - |
| 6 | 5 | 6 | 53.941 | -14.966 | 368.121 | -47.200 | 6 | 5 | 53.434 | - |
| 7 | 5 | 7 | 43.024 | -19.723 | 308.979 | -68.625 | 7 | 5 | -42.370 | - |
| 8 | 6 | 7 | 30.023 | -21.861 | 221.650 | -107.820 | 7 | 6 | -29.842 | - |
| 9 | 7 | 8 | 73.939 | 6.673 | 470.227 | 137.250 | 8 | 7 | -72.034 | - |
| 10 | 7 | 10 | -132.644 | -22.379 | -830.340 | -313.608 | 10 | 7 | 134.462 | - |
| 11 | 7 | 11 | 72.018 | -0.448 | 466.948 | 68.699 | 11 | 7 | -69.623 | - |
| 12 | 8 | 11 | 32.294 | -13.231 | 236.738 | -25.683 | 11 | 8 | -32.078 | - |
| 13 | 9 | 10 | 154.350 | 37.949 | 941.809 | 345.537 | 10 | 9 | -152.282 | - |
| 14 | 9 | 11 | 162.982 | -127.094 | 1113.861 | -683.750 | 11 | 9 | -126.821 | - |

Gambar 4.20.
Tampilan hasil aliran daya pada kondisi akhir (Setelah Optimasi)

14. Kemudian tekan tombol *summary*, untuk melihat hasil akhir jumlah pembangkitan, pembebanan, rugi-rugi (setelah optimasi)



| Parameter | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary |
|-------------------------|---------------|-------------|---------|---------------|----------------|-------------|---------|
| Summary Loadflow | | | | | | | |
| Jumlah Pembangkitan | 826.376+ | 326.481 | MVA | | | | |
| Jumlah Pembebanan | 743.600+ | 283.600 | MVA | | | | |
| Jumlah Rugi-Rugi | 82.576+ | 42.861 | MVA | | | | |
| Iterasi | 3 | | | | | | |
| Waktu Hitung | 0:1:4.125 | | | | | | |

Gambar 4.21.
Tampilan *summary* (Setelah Optimasi)

4.7. Hasil dan Analisis Hasil Perhitungan Optimasi Daya Reaktif Menggunakan Metode *Evolutionary Programming* dan *Evolutionary Strategy* Pada Saluran Transmisi 150 kV Sub Sistem Paiton-Bali

4.7.1. Hasil Perhitungan Sebelum Optimasi

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

| No. | Bus | Tegangan | Sudut | Pembangkitan | | Pembebanan | |
|-----|-----|----------|----------|--------------|---------|------------|---------|
| | | | | P(MW) | Q(MVAR) | P(MW) | Q(MVAR) |
| 1 | 1 | 1,03333 | 0 | 613,131 | -49,534 | 38,1 | 5,7 |
| 2 | 2 | 1,01858 | -0,03646 | 0 | 0 | 16,5 | 6,5 |
| 3 | 3 | 1,01503 | -0,04169 | 0 | 0 | 17,5 | 6,7 |
| 4 | 4 | 1,00380 | -0,08463 | 0 | 0 | 39,3 | 19,8 |
| 5 | 5 | 0,99603 | -0,15311 | 0 | 0 | 39,8 | 17,5 |
| 6 | 6 | 0,99776 | -0,18813 | 0 | 0 | 23,4 | 9,3 |
| 7 | 7 | 1,00625 | -0,20784 | 0 | 47 | 58,9 | 30,3 |
| 8 | 8 | 0,98177 | -0,29962 | 0 | 0 | 39,8 | 21,8 |
| 9 | 9 | 1,04925 | -0,11648 | 0 | 0 | 24,4 | 5,8 |
| 10 | 10 | 1,02647 | -0,16081 | 0 | 0 | 17,8 | 7,4 |
| 11 | 11 | 0,98832 | -0,32669 | 0 | 0 | 37,8 | 14,8 |
| 12 | 12 | 0,99266 | -0,33700 | 99,9 | 210,171 | 5,7 | 2 |
| 13 | 13 | 0,96868 | -0,34517 | 0 | 0 | 11,7 | 4,6 |
| 14 | 14 | 0,94722 | -0,42834 | 0 | 0 | 6 | 2,8 |
| 15 | 15 | 0,93892 | -0,47038 | 0 | 0 | 69,3 | 23,5 |
| 16 | 16 | 0,94905 | -0,45358 | 0 | 50 | 4 | 0,5 |
| 17 | 17 | 0,95432 | -0,44249 | 0 | 0 | 24,8 | 8,8 |
| 18 | 18 | 0,93338 | -0,48081 | 0 | 0 | 32,8 | 13,3 |
| 19 | 19 | 0,93426 | -0,48102 | 0 | 0 | 76,4 | 28,9 |
| 20 | 20 | 0,93600 | -0,47996 | 31,4 | 22,755 | 0 | 0 |
| 21 | 21 | 0,93600 | -0,47558 | 84,6 | -0,899 | 0 | 0 |
| 22 | 22 | 0,93220 | -0,48861 | 0 | 25 | 51,4 | 19,3 |
| 23 | 23 | 0,93270 | -0,48483 | 0 | 25 | 60,3 | 17,7 |
| 24 | 24 | 0,93320 | -0,48193 | 0 | 0 | 33,7 | 9,6 |
| 25 | 25 | 0,92966 | -0,48616 | 0 | 0 | 14,4 | 7 |

Tabel 4.11.
Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi

| No. | Saluran | | Daya | | Saluran | | Daya | |
|-----|---------|----|----------|----------|---------|----|----------|---------|
| | Dari | Ke | P(MW) | Q(MVAR) | Dari | Ke | P(MW) | Q(MVAR) |
| 1 | 1 | 2 | 221,298 | 22,817 | 2 | 1 | -218,840 | -15,883 |
| 2 | 1 | 9 | 353,733 | -78,051 | 9 | 1 | -347,853 | 114,995 |
| 3 | 2 | 3 | 17,533 | 4,267 | 3 | 2 | -17,500 | -6,700 |
| 4 | 2 | 4 | 184,808 | 5,116 | 4 | 2 | -182,204 | 1,766 |
| 5 | 4 | 5 | 142,904 | -21,566 | 5 | 4 | -140,115 | 27,750 |
| 6 | 5 | 6 | 55,597 | -20,253 | 6 | 5 | -55,021 | 18,129 |
| 7 | 5 | 7 | 44,718 | -24,997 | 7 | 5 | -43,941 | 19,855 |
| 8 | 6 | 7 | 31,621 | -27,428 | 7 | 6 | -31,375 | 24,595 |
| 9 | 7 | 8 | 79,204 | -2,374 | 8 | 7 | -77,115 | 1,307 |
| 10 | 7 | 10 | -141,321 | -14,077 | 10 | 7 | 143,389 | 18,452 |
| 11 | 7 | 11 | 78,533 | -11,299 | 11 | 7 | -75,899 | 9,755 |
| 12 | 8 | 11 | 37,315 | -23,107 | 11 | 8 | -36,978 | 20,025 |
| 13 | 9 | 10 | 163,486 | 31,263 | 10 | 9 | -161,189 | -25,852 |
| 14 | 9 | 11 | 159,967 | -152,057 | 11 | 9 | -118,212 | 164,746 |
| 15 | 11 | 12 | 193,289 | -209,327 | 12 | 11 | -191,959 | 212,235 |
| 16 | 12 | 13 | 207,489 | -8,199 | 13 | 12 | -202,415 | 7,301 |
| 17 | 12 | 17 | 78,670 | 4,135 | 17 | 12 | -76,130 | -5,290 |
| 18 | 13 | 14 | 96,427 | -5,354 | 14 | 13 | -93,749 | 7,832 |
| 19 | 13 | 15 | 94,288 | -6,547 | 15 | 13 | -90,382 | 8,653 |
| 20 | 14 | 15 | 87,749 | -10,632 | 15 | 14 | -86,517 | 11,513 |
| 21 | 15 | 16 | -22,528 | -3,888 | 16 | 15 | 22,738 | 0,012 |
| 22 | 15 | 17 | -24,072 | -3,941 | 17 | 15 | 24,438 | -1,980 |
| 23 | 15 | 18 | 32,198 | 5,189 | 18 | 15 | -32,070 | -6,782 |
| 24 | 15 | 19 | 31,913 | 2,561 | 19 | 15 | -31,790 | -4,171 |
| 25 | 15 | 22 | 27,068 | -4,337 | 22 | 15 | -26,823 | 1,394 |
| 26 | 15 | 24 | 63,021 | 9,525 | 24 | 15 | -62,749 | -9,843 |
| 27 | 16 | 17 | -26,738 | -0,512 | 17 | 16 | 26,892 | -1,530 |
| 28 | 18 | 19 | -0,730 | -6,518 | 19 | 18 | 0,732 | 5,687 |
| 29 | 19 | 20 | -31,366 | -22,680 | 20 | 19 | 31,400 | 22,755 |
| 30 | 19 | 21 | -84,437 | 1,356 | 21 | 19 | 84,600 | -0,899 |
| 31 | 19 | 22 | 24,669 | -5,593 | 22 | 19 | -24,577 | 4,306 |
| 32 | 19 | 23 | 45,791 | -3,500 | 23 | 19 | -45,701 | 3,233 |
| 33 | 23 | 24 | -14,599 | 4,067 | 24 | 23 | 14,620 | -4,932 |
| 34 | 24 | 25 | 14,429 | 5,175 | 25 | 24 | -14,400 | -7,000 |

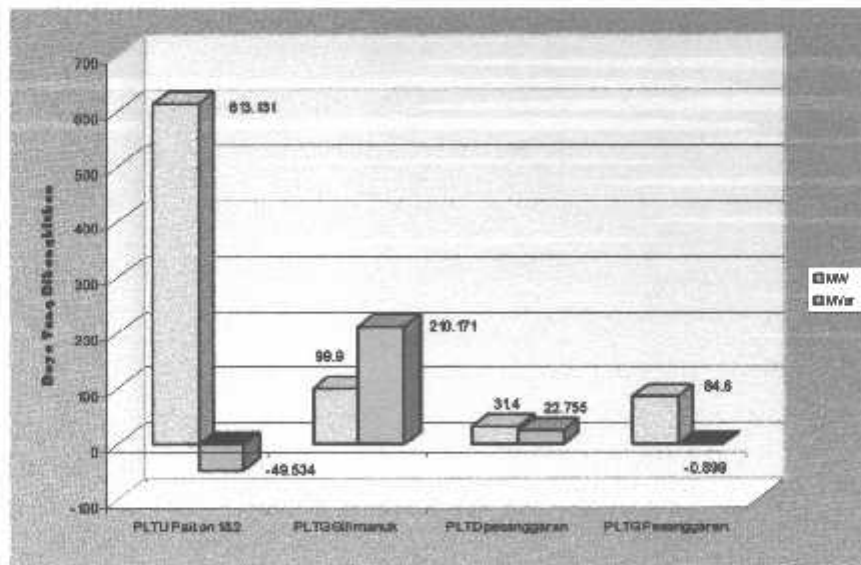
Tabel 4.12.
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum Optimasi

| Pembangkitan | | Pembebanan | | Rugi-rugi | | Waktu Komputasi |
|--------------|----------------|------------|----------------|------------|----------------|-----------------|
| Aktif (MW) | Reaktif (MVAR) | Aktif (MW) | Reaktif (MVAR) | Aktif (MW) | Reaktif (MVAR) | |
| 829,031 | 329,493 | 743,800 | 283,600 | 85,231 | 45,893 | 0:0:0:15 |

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

| No. | Nama Pembangkit | Daya Yang Dibangkitkan | | Biaya Operasi per-Jam |
|--------------------|-------------------|------------------------|---------|---------------------------|
| | | (MW) | (MVAR) | |
| 1 | PLTU Paiton 1 & 2 | 613,131 | -49,534 | Rp. 153.710.372,90 |
| 2 | PLTG Gilimanuk | 99,9 | 210,171 | Rp. 48.915.600,04 |
| 3 | PLTD Pesanggaran | 31,4 | 22,755 | Rp. 22.326.774,51 |
| 4 | PLTG Pesanggaran | 84,6 | -0,899 | Rp. 187.758.829,60 |
| Total Biaya | | | | Rp. 412.711.577,00 |

Grafik 4.1
Hasil Perhitungan Daya Yang Dibangkitkan Sebelum Optimasi



4.7.2. Hasil Perhitungan Setelah Optimasi

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

| No. | Bus | Tegangan | Sudut | Pembangkitan | | Pembebanan | |
|-----|-----|----------|----------|--------------|---------|------------|---------|
| | | | | P(MW) | Q(MVAR) | P(MW) | Q(MVAR) |
| 1 | 1 | 1,05000 | 0 | 602,576 | -13,460 | 38,1 | 5,7 |
| 2 | 2 | 1,03416 | -0,03428 | 0 | 0 | 16,5 | 6,5 |
| 3 | 3 | 1,03070 | -0,03936 | 0 | 0 | 17,5 | 6,7 |
| 4 | 4 | 1,01745 | -0,07945 | 0 | 0 | 39,3 | 19,8 |
| 5 | 5 | 1,00565 | -0,14347 | 0 | 0 | 39,8 | 17,5 |
| 6 | 6 | 1,00455 | -0,17609 | 0 | 0 | 23,4 | 9,3 |
| 7 | 7 | 1,01037 | -0,19399 | 0 | 47 | 58,9 | 30,3 |
| 8 | 8 | 0,97661 | -0,27660 | 0 | 0 | 39,8 | 21,8 |
| 9 | 9 | 1,05628 | -0,11056 | 0 | 0 | 24,4 | 5,8 |
| 10 | 10 | 1,03233 | -0,15118 | 0 | 0 | 17,8 | 7,4 |
| 11 | 11 | 0,97790 | -0,29910 | 0 | 0 | 37,8 | 1,8 |
| 12 | 12 | 0,98052 | -0,30872 | 133,8 | 97,823 | 5,7 | 2 |
| 13 | 13 | 0,95531 | -0,32364 | 0 | 0 | 11,7 | 4,6 |
| 14 | 14 | 0,95127 | -0,42344 | 0 | 0 | 6 | 2,8 |
| 15 | 15 | 0,95317 | -0,47298 | 0 | 0 | 69,3 | 23,5 |
| 16 | 16 | 0,95827 | -0,44968 | 0 | 44,163 | 4 | 0,5 |
| 17 | 17 | 0,96093 | -0,43508 | 0 | 0 | 24,8 | 8,8 |
| 18 | 18 | 0,95308 | -0,48776 | 0 | 0 | 32,8 | 13,3 |
| 19 | 19 | 0,95625 | -0,48997 | 0 | 0 | 76,4 | 28,9 |
| 20 | 20 | 0,96059 | -0,48770 | 75 | 60,593 | 0 | 0 |
| 21 | 21 | 0,95892 | -0,48994 | 15 | 40,343 | 0 | 0 |
| 22 | 22 | 0,95192 | -0,49523 | 0 | 25 | 51,4 | 19,3 |
| 23 | 23 | 0,95338 | -0,49229 | 0 | 25 | 60,3 | 17,7 |
| 24 | 24 | 0,95100 | -0,48668 | 0 | 0 | 33,7 | 9,6 |
| 25 | 25 | 0,95754 | -0,49075 | 0 | 0 | 14,4 | 7 |

Tabel 4.15.
Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi

| No. | Saluran | | Daya | | Saluran | | Daya | |
|-----|---------|----|----------|----------|---------|----|----------|---------|
| | Dari | Ke | P(MW) | Q(MVAR) | Dari | Ke | P(MW) | Q(MVAR) |
| 1 | 1 | 2 | 217,401 | 31,237 | 2 | 1 | -215,080 | -24,809 |
| 2 | 1 | 9 | 347,075 | -50,397 | 9 | 1 | -341,732 | 83,346 |
| 3 | 2 | 3 | 17,532 | 4,185 | 3 | 2 | -17,500 | -6,7 |
| 4 | 2 | 4 | 181,048 | 14,125 | 4 | 2 | -178,610 | -7,864 |
| 5 | 4 | 5 | 139,310 | -11,936 | 5 | 4 | -136,765 | 17,209 |
| 6 | 5 | 6 | 53,941 | -14,986 | 6 | 5 | -53,434 | 12,561 |
| 7 | 5 | 7 | 43,024 | -19,723 | 7 | 5 | -42,370 | 14,054 |
| 8 | 6 | 7 | 30,033 | -21,861 | 7 | 6 | -29,842 | 18,799 |
| 9 | 7 | 8 | 73,939 | 6,673 | 8 | 7 | -72,094 | -8,569 |
| 10 | 7 | 10 | -132,644 | -22,379 | 10 | 7 | 134,482 | 25,946 |
| 11 | 7 | 11 | 72,018 | 0,449 | 11 | 7 | -69,823 | -2,531 |
| 12 | 8 | 11 | 32,294 | -13,231 | 11 | 8 | -32,078 | 9,806 |
| 13 | 9 | 10 | 154,350 | 37,949 | 10 | 9 | -152,282 | -33,345 |
| 14 | 9 | 11 | 162,982 | -127,094 | 11 | 9 | -126,821 | 137,156 |
| 15 | 11 | 12 | 190,920 | -159,232 | 12 | 11 | -189,886 | 161,494 |
| 16 | 12 | 13 | 229,595 | -54,338 | 13 | 12 | -222,894 | 53,979 |
| 17 | 12 | 17 | 88,392 | -11,333 | 17 | 12 | -85,132 | 12,607 |
| 18 | 13 | 14 | 106,649 | -28,698 | 14 | 13 | -103,083 | 33,805 |
| 19 | 13 | 15 | 104,545 | -29,880 | 15 | 13 | -99,308 | 37,061 |
| 20 | 14 | 15 | 97,083 | -36,605 | 15 | 14 | -95,409 | 38,712 |
| 21 | 15 | 16 | -26,859 | 5,198 | 16 | 15 | 27,168 | -8,991 |
| 22 | 15 | 17 | -28,400 | 5,233 | 17 | 15 | 28,939 | -10,971 |
| 23 | 15 | 18 | 39,644 | -14,111 | 18 | 15 | -39,442 | 12,660 |
| 24 | 15 | 19 | 42,654 | -24,985 | 19 | 15 | -42,377 | 23,744 |
| 25 | 15 | 22 | 29,305 | -14,559 | 22 | 15 | -28,976 | 11,651 |
| 26 | 15 | 24 | 69,074 | -11,886 | 24 | 15 | -68,755 | 11,660 |
| 27 | 16 | 17 | -31,168 | 8,491 | 17 | 16 | 31,393 | -10,436 |
| 28 | 18 | 19 | 6,642 | -25,960 | 19 | 18 | -6,608 | 25,184 |
| 29 | 19 | 20 | -74,799 | -60,150 | 20 | 19 | 75 | 60,593 |
| 30 | 19 | 21 | -14,960 | -40,230 | 21 | 19 | 15 | 40,343 |
| 31 | 19 | 22 | 22,499 | 4,564 | 22 | 19 | -22,424 | -5,951 |
| 32 | 19 | 23 | 39,845 | 17,987 | 23 | 19 | -39,767 | -18,296 |
| 33 | 23 | 24 | -20,533 | 25,596 | 24 | 23 | 20,627 | -26,359 |
| 34 | 24 | 25 | 14,428 | 5,098 | 25 | 24 | -14,400 | -7 |

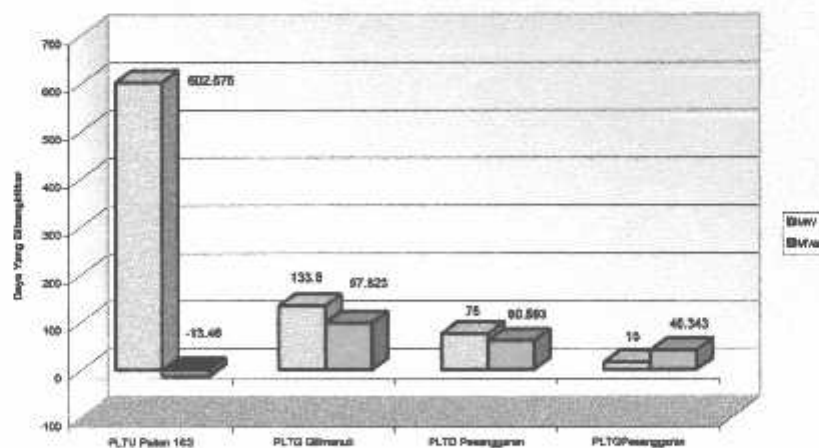
Tabel 4.16.
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi

| Pembangkitan | | Pembebanan | | Rugi-rugi | | Waktu Komputasi |
|--------------|----------------|------------|----------------|------------|----------------|-----------------|
| Aktif (MW) | Reaktif (MVAR) | Aktif (MW) | Reaktif (MVAR) | Aktif (MW) | Reaktif (MVAR) | |
| 826,376 | 326,461 | 743,800 | 283,600 | 82,576 | 42,861 | 0:1:23:594 |

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

| No | Nama Pembangkit | Daya Yang Dibangkitkan | | Biaya Operasi per-Jam |
|--------------------|-------------------|------------------------|---------|---------------------------|
| | | (MW) | (MVAR) | |
| 1 | PLTU Paiton 1 & 2 | 602,576 | -13,460 | Rp. 151.942.651,10 |
| 2 | PLTG Gilimanuk | 133,8 | 97,823 | Rp. 50.211.107,72 |
| 3 | PLTD Pesanggaran | 75 | 60,593 | Rp. 43.462.389,57 |
| 4 | PLTG Pesanggaran | 15 | 40,343 | Rp. 35.429.334,58 |
| Total Biaya | | | | Rp. 281.045.483,00 |

Grafik 4.2.
Hasil Perhitungan Daya Yang Dibangkitkan Setelah Optimasi



4.8. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Daya Reaktif Menggunakan Metode *Evolutionary Programming* dan *Evolutionary Strategy*

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

4.8.1. Tingkat Optimum Biaya Pembangkitan

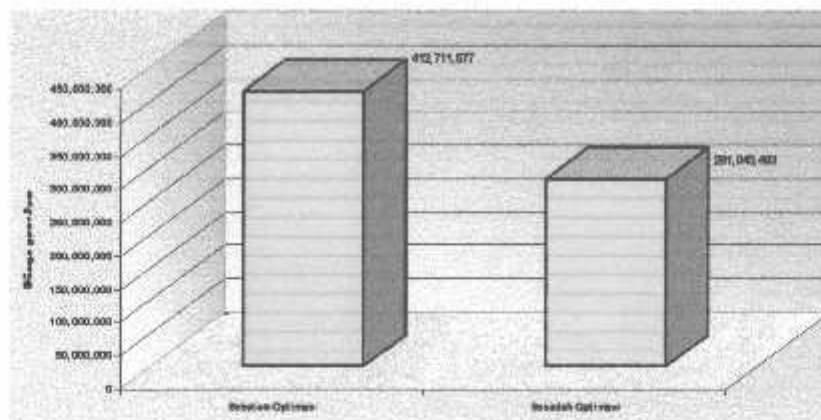
Berikut ini adalah tabel yang berisi perbandingan biaya optimum pembangkitan antara hasil perhitungan sebelum dan setelah optimasi.

Tabel 4.18.
Perbandingan Tingkat Optimum Biaya Pembangkitan

| No | Nama Pembangkit | Sebelum Optimasi | Setelah Optimasi |
|----|--------------------|--------------------|--------------------|
| 1 | PLTU Paiton 1 & 2 | Rp. 153.710.372,90 | Rp. 151.942.651,10 |
| 2 | PLTG Gilimanuk | Rp. 48.915.600,04 | Rp. 50.211.107,72 |
| 3 | PLTD Pesanggaran | Rp. 22.326.774,51 | Rp. 43.462.389,57 |
| 4 | PLTG Pesanggaran | Rp. 187.758.829,60 | Rp. 35.429.334,58 |
| | Total Biaya | Rp. 412.711.577,00 | Rp. 281.045.483,00 |

Dari tabel tingkat optimum diatas dapat dilihat bahwa besarnya biaya produksi pembangkitan sebelum optimasi adalah Rp.412.711.577,- per-jam sedangkan besarnya biaya produksi pembangkitan setelah optimasi adalah Rp.281.045.483,- per-jam Jadi selisih antara biaya produksi pembangkitan sebelum dengan setelah optimasi adalah Rp.131.666.094,- per-jam atau terjadi optimasi biaya sebesar 31,8980 %.

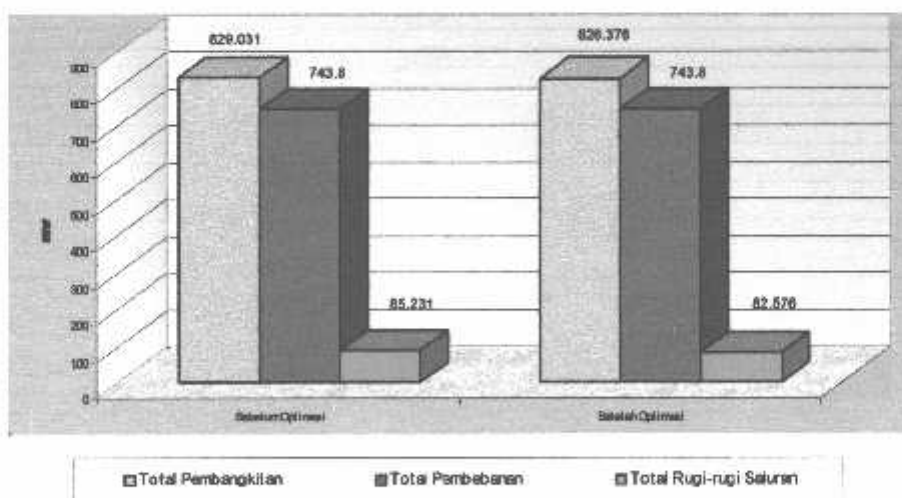
Grafik 4.3.
Perbandingan Biaya Pembangkitan Sebelum dan Setelah Optimasi



Tabel 4.19.
**Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran
 Sebelum dan Setelah Optimasi**

| Pembangkitan | | Pembebanan | | Rugi-rugi | |
|--------------|--------------|--------------|--------------|--------------|--------------|
| Sebelum (MW) | Setelah (MW) | Sebelum (MW) | Setelah (MW) | Sebelum (MW) | Setelah (MW) |
| 829.031 | 826.376 | 743.800 | 743.800 | 85.231 | 82.576 |

Grafik 4.4.
**Perbandingan Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran
 Sebelum dan Setelah Optimasi**



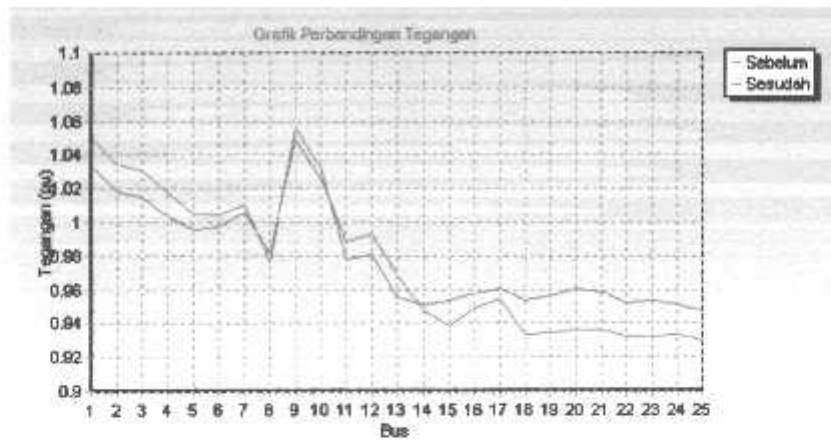
4.8.2. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

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

Tabel 4.20.
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

| No. | Sebelum Optimasi | | Setelah Optimasi | |
|-----|------------------|----------------------|------------------|----------------------|
| | Tegangan (pu) | Sudut Tegangan (deg) | Tegangan (pu) | Sudut Tegangan (deg) |
| 1 | 1,03333 | 0 | 1,05000 | 0 |
| 2 | 1,01858 | -0,03646 | 1,03416 | -0,03428 |
| 3 | 1,01503 | -0,04169 | 1,03070 | -0,03936 |
| 4 | 1,00380 | -0,08463 | 1,01745 | -0,07945 |
| 5 | 0,99603 | -0,15311 | 1,00565 | -0,14347 |
| 6 | 0,99776 | -0,18813 | 1,00455 | -0,17609 |
| 7 | 1,00625 | -0,20784 | 1,01037 | -0,19399 |
| 8 | 0,98177 | -0,29962 | 0,97661 | -0,27660 |
| 9 | 1,04925 | -0,11648 | 1,05628 | -0,11056 |
| 10 | 1,02647 | -0,16081 | 1,03233 | -0,15118 |
| 11 | 0,98832 | -0,32669 | 0,97790 | -0,29910 |
| 12 | 0,99266 | -0,33700 | 0,98052 | -0,30872 |
| 13 | 0,96868 | -0,34517 | 0,95531 | -0,32364 |
| 14 | 0,94722 | -0,42834 | 0,95127 | -0,42344 |
| 15 | 0,93892 | -0,47038 | 0,95317 | -0,47298 |
| 16 | 0,94905 | -0,45358 | 0,95827 | -0,44968 |
| 17 | 0,95432 | -0,44249 | 0,96093 | -0,43508 |
| 18 | 0,93338 | -0,48081 | 0,95308 | -0,48776 |
| 19 | 0,93426 | -0,48102 | 0,95625 | -0,48997 |
| 20 | 0,93600 | -0,47996 | 0,96059 | -0,48770 |
| 21 | 0,93600 | -0,47558 | 0,95892 | -0,48994 |
| 22 | 0,93220 | -0,48861 | 0,95192 | -0,49523 |
| 23 | 0,93270 | -0,48483 | 0,95338 | -0,49229 |
| 24 | 0,93320 | -0,48193 | 0,95100 | -0,48668 |
| 25 | 0,92966 | -0,48616 | 0,95754 | -0,49075 |

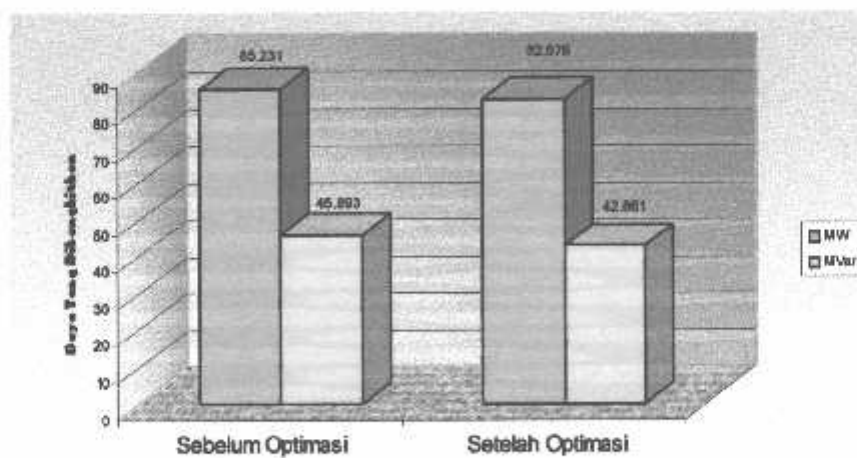
Grafik 4.5.
Perbandingan Profil Tegangan



4.8.3. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran

Untuk rugi-rugi daya pada saluran sebelum dioptimasi adalah 85,231 MW dan 45,893 MVAR, sedangkan rugi-rugi daya setelah dioptimasi adalah 82,576 MW dan 42,861 MVAR. Dari hasil perhitungan selisih rugi-rugi daya turun sebesar 2,655 MW dan 3,032 MVAR.

Grafik 4.6.
Perbandingan Rugi-rugi Daya



BAB V

KESIMPULAN DAN SARAN

5.1. Kesimpulan

Setelah dilakukan analisa perhitungan optimasi daya reaktif dengan menggunakan metode *evolutionary programming* dan *evolutionary strategy* maka dapat diambil kesimpulan sebagai berikut :

1. Dari hasil perhitungan *load flow* profil tegangan rata-rata tiap bus sebelum optimasi adalah 0,97339636 pu sedangkan setelah optimasi adalah 0,983926 pu sehingga rata-rata tiap bus mengalami perbaikan 1,0704%.
2. Dengan optimasi daya reaktif menggunakan metode *evolutionary programming* dan *evolutionary strategy* rugi-rugi daya dalam sistem turun dari 85,231 MW menjadi 82,576 MW dan 45,893 MVAR menjadi 42,861 MVAR terjadi penurunan rugi-rugi daya sebesar 2,655 MW dan 3,032 MVAR.
3. Biaya pembangkitan sebelum optimasi adalah Rp.402.419.746,- per-jam dan setelah optimasi adalah Rp.281.045.483,- per-jam. Jadi selisih antara biaya produksi pembangkitan sebelum dengan setelah optimasi adalah Rp.131.666.094,- per-jam atau terjadi optimasi biaya sebesar 31,8980 %.

5.2. Saran

Penggunaan metode *evolutionary programming* dan *evolutionary strategy* untuk menyelesaikan permasalahan optimasi daya reaktif masih perlu dikembangkan lagi terhadap sistem jaringan yang lebih luas dan penggunaan waktu komputasi yang lebih singkat sehingga nantinya dapat digunakan lebih efektif dan efisien dalam pemecahan permasalahan optimasi biaya pembangkitan.

DAFTAR PUSTAKA

- [1]. Kwang. Y. Lee, and Frank F. Yang, "*Optimal Reactive Power Planning Using Evolutionary Algorithms: A Comparative Study for Evolutionary Programming, Evolutionary Strategy, Genetic Algorithm, and Linier Programming*," IEEE Trans on Power System, Vol 13, No 1, February 1998.
 - [2]. Wood, Allan J and B. F. Wollenberg, "*Power Generation, Operation, and Control*", John Willey & Sons, Inc, 1996.
 - [3]. Stevenson, William D, Jr, "*Analisa System Tenaga Listrik*", Penerbit Erlangga edisi keempat, 1996.
 - [4]. Zuhail, "*Dasar Teknik Tenaga Listrik dan Elektronika Daya*", PT. Gramedia Utama, Jakarta, 1995.
 - [5]. Stagg, Glenn W. and Ahmed H. El-Abiad "*Computer Methods In Power System Analysis*".
 - [6]. Lamont. John W and Fu Jian, "*Cost Analysis Of Reactive Power Support*," IEEE Trans on Power System, Vol 14, No 2, August 1999
 - [7]. <http://zooland.alife.org/hhg2ec/>, "*What's Evolutionary Programming (EP) dan What's Evolutionary Strategy (ES)?*".
 - [8]. Luke, Brian T., "*Overview of Evolutionary Programming Methods*", Learning From The Web.net.
 - [9]. Spears, William M., "*An Overview of Evolutionary Computation*".
 - [10]. Hussain, Talib s., "*An Introduction to Evolutionary Computation*", Department of Computing and Information Science Queen's University, Kingston.
 - [11]. Pai, M. A., "*Computer Techniques in Power System Analysis*", Tata Mc Graw-Hill Publishing Company Limited.
 - [12]. Syafena, Jatri, Skripsi, "*Economic Dispatch Dengan Optimasi Daya Aktif dan Reaktif Pada Pembangkit Thermal di Area IV*", 2002.
 - [13]. Data "*Load Flow Region Jawa Timur dan Bali*", Sub Sistem 150 kV Paiton dan Bali Hari Rabu, 30 Maret 2005 Pukul; 19:30 WIB, PT. PLN (Persero) P3B Region Jawa dan Bali, Waru.
 - [14]. Data Saluran Transmisi 150 kV Sub Sistem Paiton dan Bali.
 - [15]. Lamont. John W and Fu Jian, "*Cost Analysis Of Reactive Power Support*," IEEE Trans on Power System, Vol. 14, No. 2, August. 1999.
-

LAMPIRAN



PERMOHONAN PERSETUJUAN SKRIPSI

Yang bertanda tangan dibawah ini

Nama : AGUSTIAN KHALIMAH
 NIM : 19012058
 Semester : 1
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-1
 Konsentrasi : Teknik Energi Listrik
 Alamat : Jl. SANDI ULU 50 MALANG

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

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

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

Yang diteliti kebenarannya data tersebut adalah
Rekording Teknik Elektro

Malang, ... 15 ... 2005
Pemohon

Disetujui
Ketua Jurusan Teknik Elektro

H. E. Yudi Limpratonu, MT
NIP. Y. 1939590274

Mengetahui
Dosen Wali

H. E. Yudi Limpratonu, MT

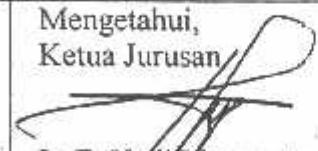
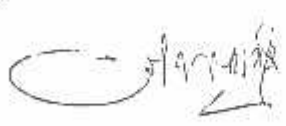
Catatan :

- Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI, agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1
1. IPK 3,00 / 3,33
 2. 6 praktik
 3. 11 seminar
- Si sudah penuhi



**LEMBAR PENGAJUAN JUDUL SKRIPSI
JURUSAN TEKNIK ELEKTRO S-1**

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika *)

| | | | | |
|--|---|--|---|------------|
| 1 | Nama Mahasiswa : | LUKMAN FAKIM | Nim : | 00-12058 |
| 2 | Waktu pengajuan | Tanggal : 21 | Bulan : 05 | Tahun : 05 |
| Spesifikasi judul (berilah tanda silang) | | | | |
| 3 | a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri | e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya | | |
| 4 | Konsultasikan judul sesuai materi bidang ilmu kepada Dosen *) : | | Mengetahui, Ketua Jurusan  Ir. F. Yudi Limpraptono, MT Nip. Y. 1039500274 | |
| 5 | Judul yang diajukan mahasiswa : | OPTIMAL POWER FLOW (OPF) MENGGUNAKAN METODE EVOLUTIONARY PROGRAMMING (EP) DAN EVOLUTIONARY STRATEGY (ES) PADA SALURAN TRANSMISI 150 KV SUBSISTEM PATIUN-BALI | | |
| 6 | Perubahan Judul yang disetujui Dosen sesuai materi bidang ilmu | | | |
| Catatan : | | | | |
| 7 | Persetujuan Judul Skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu | | Disetujui, 6 April 2005 Dosen  | |

Perhatian

1. Formulir Pengajuan ini harap dikembalikan kepada jurusan paling lambat satu minggu setelah disetujui kelompok dosen keahlian dengan dilampirkan proposal skripsi beserta persyaratan skripsi sesuai form S-1
2. Keterangan : *) coret yang tidak perlu
*) dilingkari a, b, c, atau g sesuai bidang keahlian

INSTITUT TEKNOLOGI NASIONAL
Jl. Bendungan Sigura-gura No. 2
MALANG

Lampiran : 1 (satu) berkas
Pembimbing Skripsi

Kepada : Yth. Bapak/Ibu Ir. Choiri
Dosen Institut Teknologi Nasional
MALANG

Yang bertanda tangan di bawah ini :

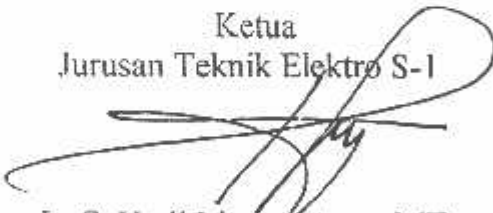
Nama : LUKMAN KHAKIM
Nim : 00-12-058
Jurusan : Teknik Elektro S-1
Konsentrasi : Energi Listrik

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing Utama / Pendamping *) , untuk penyusunan Skripsi dengan judul (proposal terlampir) : OPTIMAL POWER FLOW(OPF) MENGGUNAKAN METODE EVOLUTIONARY PROGRAMMING (EP) DAN EVOLUTIONARY STRATEGY (ES) PADA SALURAN TRANSMISI 150KV SUB SISTEM PATON-BALI

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.
Demikian permohonan kami dan atas kesediaan Bapak/Ibu kami ucapkan terima kasih.

Malang, April 2003

Ketua
Jurusan Teknik Elektro S-1


Ir. F. Yudi Limpraptono, MT
Nip. 131 991 182

Hormat kami,


LUKMAN KHAKIM

*) coret yang tidak perlu

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : LUKMAN KHAKIM
Nim : 00-12058
Semester : 10
Jurusan : Teknik Elektro S-1
Konsentrasi : Energi Listrik

Dengan ini Menyatakan bersedia / tidak bersedia *) Membimbing Skripsi dari mahasiswa tersebut, dengan judul : OPTIMAL POWER FLOW (OPF) MENGGUNAKAN METODE EVOLUTIONARY PROGRAMMING DAN EVOLUTIONARY STRATEGY (ES) PADA SALURAN TRANSMISI 150KV SUB-SISTEM DATON - BALI

Demikian surat Pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang, April 2005

Kami yang Membuat pernyataan,


Chairi

NIP. 130903042.

Catatan :

Setelah disetujui agar formulir ini
Diserahkan mahasiswa/i yang bersangkutan
Kepada Jurusan untuk diproses lebih lanjut.

*) Coret yang tidak perlu



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

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

PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

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

Malang, 27 Mei. 2005.

Nomor : ITN-446/I.SKP /2/'05
Lampiran : satu lembar
Perihal : **BIMBINGAN SKRIPSI**

Kepada : Yth. Sdr. **Ir. H. CHOIRI**
Dosen Institut Teknologi Nasional
di -
Malang

Dengan Hormat,
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi melalui seminar proposal yang telah dilakukan untuk mahasiswa :

Nama : LUKMAN KHAKIM
Nim : 0012058
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro
Konsentrasi : T. Energi Listrik (S-1)

Dengan ini pembimbingan skripsi tersebut kami serahkan sepenuhnya kepada saudara/I selama masa waktu **6 (enam) bulan** terhitung mulai tanggal:

23 Mei 2005 s/d 23 Nop. 2005

Adapun tugas tersebut merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik, Jurusan Teknik Elektro.
Demikian atas perhatian serta kerjasamanya yang baik kami ucapkan terima kasih



Ketua
Jurusan Teknik Elektro S-1

[Signature]
Ir. F. Yudi Limpraptono, MT
Nip. 1039500274

Tinjauan :

1. Mahasiswa yang bersangkutan
2. Arsip.

Form. S-4a



FORMULIR BIMBINGAN SKRIPSI

Nama : LUKMAN KHAKIM
 Nim : 00.12.058
 Masa Bimbingan : 23 Mei 2005 s/d 23 November 2005
 Judul Skripsi : OPTIMASI DAYA REAKTIF SECARA EKONOMIS
 MENGGUNAKAN METODE EVOLUTIONARY
 PROGRAMMING DAN EVOLUTIONARY STRATEGY PADA
 SALURAN TRANSMISI 150 KV SUB SISTEM PATTON-BALI

| No. | Tanggal | Uraian | Parap Pembimbing |
|-----|---------|-------------------------------|------------------|
| 1. | 22-5-05 | BAB. I. Cek batasan masalah. | JA |
| 2. | 25-6-05 | BAB. II. Kalimat mengenai gk. | JA |
| 3. | 10-7-05 | Konsultasi. BAB. <u>III</u> | JA |
| 4. | 22-7-05 | Konsultasi. BAB. <u>IV</u> | JA |
| 5. | 11-8-05 | Konsultasi. BAB. <u>V</u> | JA |
| 6. | | Ace | |
| 7. | | | |
| 8. | | | |
| 9. | | | |
| 10. | | | |

Malang 11-8. 2005
 Dosen Pembimbing.

Ir. H. CHOIRI

Form. S-4b



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

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

PT. BNI (PERSERO) MALANG
 BANK NIAGA MALANG

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

Nomor : ITN-1391/III.TA/2/2005
 Lampiran : -
 Perihal : Survey / Permintaan Data

Malang, 1 Juni 2005

Kepada : Yth Pimpinan
 PT. PLN (Persero) P3B Jawa Bali
 Region Jawa Timur dan Bali
 Jl. Suningrat No.45 Taman 61257
 di-
 Sidoarjo

Bersama ini dengan hormat kami mohon kebijaksanaan Bapak agar mahasiswa kami dari Fakultas Teknologi Industri, Jurusan Teknik Elektro S-1 Konsentrasi Teknik Energi Listrik dapat diijinkan untuk melaksanakan survey di PT. PLN (Persero) P3B Jawa Bali untuk mendapatkan data-data guna penyusunan Tugas Akhir dengan Judul :

OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN METODE EVOLUTIONARY PROGRAMMING DAN EVOLUTIONARY STRATEGY PADA SALURAN TRANSMISI 150 KV SUB SISTEM PAITON-BALI

Mahasiswa tersebut adalah :

1. LUKMAN KHAKIM

Nim : 00 12 058

Adapun lama Survey adalah : 2 (Dua) Minggu
 Demikian agar maklum dan atas perhatian serta bantuannya kami ucapkan terima kasih.



Dekan
 Fakultas Teknologi Industri
 U.b. Wakil Dekan I

Ir. I Made Wartana, MT
 Nip. 131 991 182

Tembusan disampaikan kepada Yth :

1. Ketua Jurusan
2. Arsip



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

Jl. Suningrat No. 45 Taman Sidoarjo 61257

Telepon : (031) 7882113, 7882114
Facsimile : (031) 7882578, 7881024

Kotak Pos : 4119 SBS
Bank : Bank Mandiri

22 JUN 2005

Nomor : 160/330/RJT/2005.
Surat Sdr. No. : ITN-1391/III.TA/2/2005.
Lampiran : 1 (satu) lampiran.
Perihal : Ijin Survey/ Pengambilan Data.

Kepada

Yth. Dekan Fakultas Teknik,
Institut Teknologi Nasional Malang
Di
MALANG

Menunjuk surat Saudara nomor : ITN-1391/III.TA/2/2005 tanggal 01 Juni 2005 perihal : Survey/ Permintaan Data. dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada Mahasiswa Saudara, bernama :

• LUKMAN KHAKIM Nim : 00.12.058.

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 Enjiniring.

Demikian harap maklum dan terima kasih atas perhatian saudara.

a.n. MANAGER
MANAGER BIDANG ENJINIRING.



Tembusan Yth. :

1. M.SDM PLN P3B.
2. Sdr. Lukman Khakim.

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : LUKMAN KHAKIM
Pria/ Wanita : PRIA
Tempat / Tanggal lahir : TULUNGABUNG / 17 APRIL 1982
Alamat / no telepon : Jl. KARANG BESUKI BLOK II A . MALANG
(0341) 560 262
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 Enjiniring 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, 8 Juni 05
Yang membuat pernyataan



LUKMAN KHAKIM

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : LUKMAN KHAKIM
Pria/ Wanita : PRIA
Tempat / Tanggal lahir : TULUNGAGUNG / 17 APRIL 1982
Alamat / no telepon : Jl. KARANG BESUKI BLOK IIA - MALANG
(0341) 560 292
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 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 Enjiniring 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, 8 Juni 05
Yang membuat pernyataan

PT PLN (Persero) P3B
6000
Tgl
LUKMAN KHAKIM



LEMBAR BIMBINGAN SKRIPSI

1. Nama : LUKMAN KHAKIM
2. NIM : 00.12.058
3. Jurusan : Teknik Elektro S-1
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY* PADA SALURAN TRANSMISI 150kV SUB SISTEM PAITON-BALI
6. Tanggal Mengajukan Skripsi : 5 Mei 2005
7. Tanggal Menyelesaikan Skripsi : 23 September 2005
8. Dosen Pembimbing : Ir. H. CHOIRI
9. Telah Dievaluasikan Dengan Nilai : 85 (A)

Mengetahui,
Ketua Jurusan Teknik Elektro S-1

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

Diperiksa dan Disetujui,
Dosen Pembimbing

Ir. H. CHOIRI
NIP. 130 703 042



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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama Mahasiswa : LUKMAN KHAKIM
N.I.M. : 00.12.058
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON-BALI

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

Hari : Kamis
Tanggal : 6 Otober 2005
Dengan Nilai : 77,2 (B+) *8*



(Ir. Mochtar Asroni, MSME)
Ketua Majelis Penguji

Panitia Ujian Skripsi

(Ir. F. Yudi Limpraptono, MT)
Sekretaris Majelis Penguji

(Ir. Soemarwanto)
Penguji Pertama

Anggota Penguji

(Ir. M Abdul Hamid, MT)
Penguji Kedua



PERSETUJUAN PERBAIKAN SKRIPSI

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

Hari : Kamis
Tanggal : 6 Oktober 2005

Telah dilakukan perbaikan skripsi oleh :

1. Nama : LUKMAN KHAKIM
2. NIM : 00.12.058
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI DAYA REAKTIF SECARA EKONOMIS MENGGUNAKAN METODE *EVOLUTIONARY PROGRAMMING* DAN *EVOLUTIONARY STRATEGY* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON-BALI

Perbaikan meliputi :

| No | Materi Perbaikan | Paraf |
|----|--|-------|
| 1. | Gambar Tabel, Grafik, Persamaan, Cuplikan Grafik belum ada sumber supaya diberi sumber | |
| 2. | Tambahkan teori pengaturan pembangkitan daya reaktif pada generator ataupun dengan kapasitor | |
| 3. | Judul serasikan dengan analisa | |

Ir. Soemarwanto
Penguji Pertama

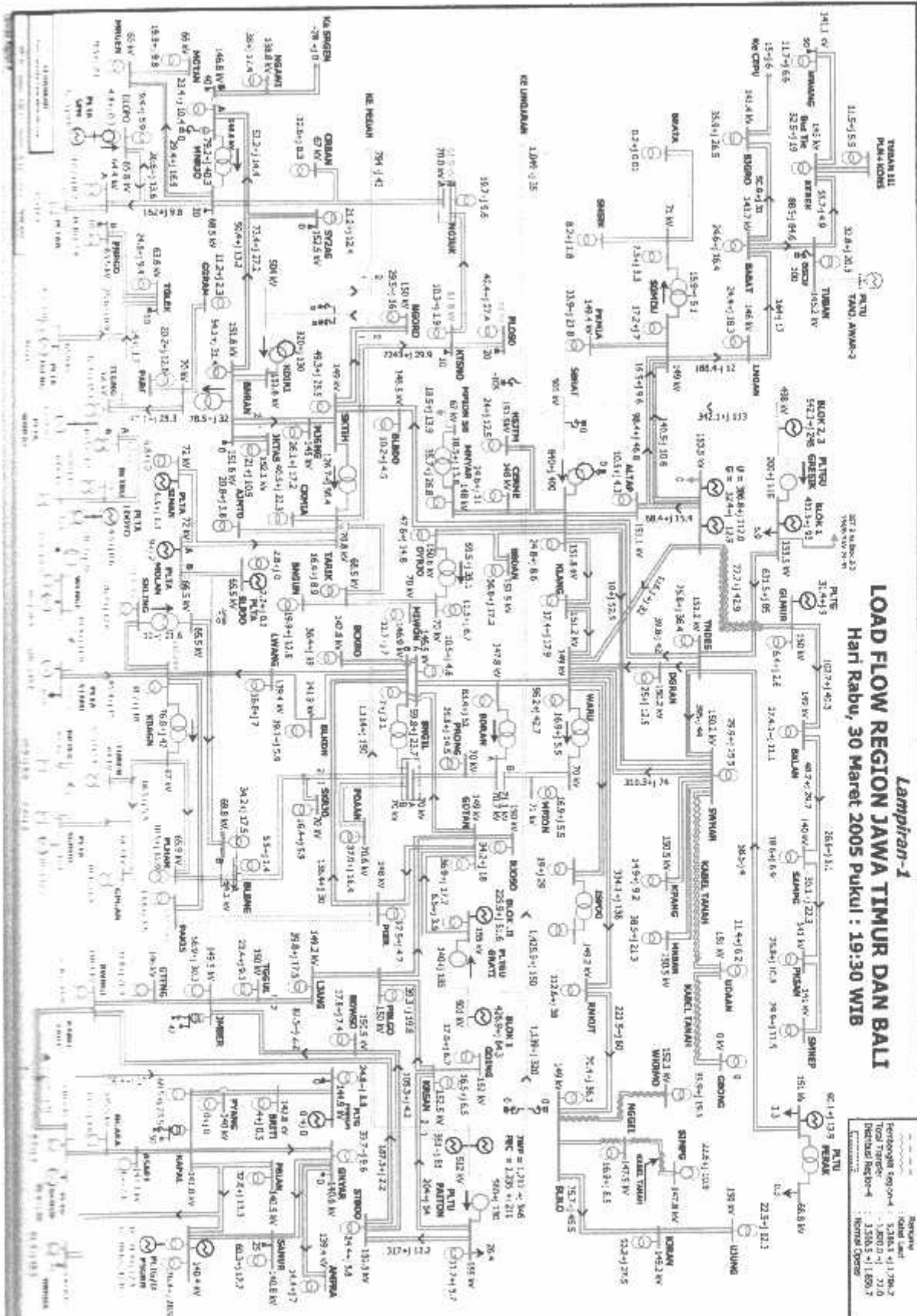
Anggota Penguji

Ir. M Abdul Hamid, MT
Penguji Kedua

Dosen Pembimbing

Ir. H. Choiri

Lampiran-1
LOAD FLOW REGION JAWA TIMUR DAN BALI
Hari Rabu, 30 Maret 2005 Pukul : 19.30 WIB



DATA KALKULASI EKSTRAK TRANSMISI UPT & SRB

| Jenis Stasi | Dat | KC | Rel | Teg. (kV) | Jarak (km) | Lintas | Perumahan | Jenis | Titik | 1sp. 21 | 2sp. 21 | 3sp. 21 | 4sp. 21 | 5sp. 21 | 6sp. 21 | 7sp. 21 | 8sp. 21 | 9sp. 21 | 10sp. 21 | 11sp. 21 | 12sp. 21 | 13sp. 21 | 14sp. 21 | 15sp. 21 | 16sp. 21 | 17sp. 21 | 18sp. 21 | 19sp. 21 | 20sp. 21 | 21sp. 21 | 22sp. 21 | 23sp. 21 | 24sp. 21 | 25sp. 21 | 26sp. 21 | 27sp. 21 | 28sp. 21 | 29sp. 21 | 30sp. 21 | 31sp. 21 | 32sp. 21 | 33sp. 21 | 34sp. 21 | 35sp. 21 | 36sp. 21 | 37sp. 21 | 38sp. 21 | 39sp. 21 | 40sp. 21 | 41sp. 21 | 42sp. 21 | 43sp. 21 | 44sp. 21 | 45sp. 21 | 46sp. 21 | 47sp. 21 | 48sp. 21 | 49sp. 21 | 50sp. 21 | 51sp. 21 | 52sp. 21 | 53sp. 21 | 54sp. 21 | 55sp. 21 | 56sp. 21 | 57sp. 21 | 58sp. 21 | 59sp. 21 | 60sp. 21 | 61sp. 21 | 62sp. 21 | 63sp. 21 | 64sp. 21 | 65sp. 21 | 66sp. 21 | 67sp. 21 | 68sp. 21 | 69sp. 21 | 70sp. 21 | 71sp. 21 | 72sp. 21 | 73sp. 21 | 74sp. 21 | 75sp. 21 | 76sp. 21 | 77sp. 21 | 78sp. 21 | 79sp. 21 | 80sp. 21 | 81sp. 21 | 82sp. 21 | 83sp. 21 | 84sp. 21 | 85sp. 21 | 86sp. 21 | 87sp. 21 | 88sp. 21 | 89sp. 21 | 90sp. 21 | 91sp. 21 | 92sp. 21 | 93sp. 21 | 94sp. 21 | 95sp. 21 | 96sp. 21 | 97sp. 21 | 98sp. 21 | 99sp. 21 | 100sp. 21 |
|-------------|-----|----|-----|-----------|------------|--------|-----------|-------|-------|---------|---------|---------|---------|---------|---------|---------|---------|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------|
|-------------|-----|----|-----|-----------|------------|--------|-----------|-------|-------|---------|---------|---------|---------|---------|---------|---------|---------|---------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------|

6. UPT Probolinggo

| | | | | | | | | | | | | | | | | | | | |
|---|--------|--------|---|-----|--------|--------------|-------------------------|------|------|------|------|------|--------|--------|--------|--------|--------|--------|----------|
| 1 | PITONS | KRSANS | 2 | 150 | 20.204 | ACSR AW | 330 mm | 150 | 1070 | 1000 | 1250 | 1750 | 21172 | 9.4003 | 2.9530 | 0.0105 | 0.0359 | 0.0030 | 192.8824 |
| 1 | PITONS | STBDOS | 1 | 150 | 55.433 | ACSR ZEBRA | 2 x 415 mm ² | 1500 | 1000 | 2000 | 1250 | 1250 | 0.0381 | 6.2807 | 4.0230 | 0.0095 | 0.0692 | 0.0002 | 415.8903 |
| 1 | PITONS | STBDOS | 2 | 150 | 55.433 | ACSR ZEBRA | 2 x 435 mm ² | 1500 | 2000 | 2000 | 1250 | 1250 | 0.0387 | 6.2807 | 4.0230 | 0.0095 | 0.0692 | 0.0002 | 415.8903 |
| 1 | BNGIL4 | GDTANS | 1 | 70 | 4.452 | ACSR OSTRICH | 300 MCM | 400 | | | | | | | | | | | 53.3456 |
| 1 | BNGIL4 | GDTANS | 2 | 70 | 4.452 | ACSR OSTRICH | 300 MCM | 400 | | | | | | | | | | | 48.5897 |
| 1 | BNGIL4 | POAAMA | 1 | 70 | 9.330 | ACSR OSTRICH | 300 MCM | 440 | 250 | 400 | | | 0.2180 | 0.3850 | 2.972 | 0.0437 | 0.0762 | 0.0014 | 53.3456 |
| 1 | BNGIL4 | POAAMA | 2 | 70 | 9.330 | ACSR OSTRICH | 300 MCM | 440 | 250 | 400 | 800 | 800 | 0.2180 | 0.3850 | 2.972 | 0.0437 | 0.0762 | 0.0014 | 53.3456 |
| 1 | BNGIL4 | BORAMA | 1 | 70 | 11.700 | ACSR PIPER | 300 MCM | 440 | 400 | 400 | 400 | 400 | 0.2180 | 0.3850 | 2.972 | 0.0521 | 0.0919 | 0.0017 | 53.3456 |
| 1 | BNGIL4 | PRONG4 | 1 | 70 | 11.700 | ACSR PIPER | 300 MCM | 440 | 400 | 400 | 400 | 400 | 0.2180 | 0.3850 | 2.972 | 0.0521 | 0.0919 | 0.0017 | 53.3456 |
| 1 | BNGIL4 | BLBNG4 | 1 | 70 | 40.000 | ACSR PIGEON | 300 AWG | 300 | 200 | 200 | 800 | 800 | 0.3051 | 0.3584 | 3.202 | 0.2492 | 0.2926 | 0.0053 | 36.3720 |
| 1 | BNGIL4 | BLBNG4 | 2 | 70 | 40.000 | ACSR PIGEON | 300 AWG | 300 | 200 | 200 | 400 | 800 | 0.3051 | 0.3584 | 3.202 | 0.2492 | 0.2926 | 0.0053 | 36.3720 |

Modifikasi TL Bay Grati - Pier 1 sirkuit menjadi (Grati - Gdian - Pier) dan Probolinggo - Gondangwetan 1 sirkuit menjadi (Pbligo - Grati - Gdian)

| | | | | | | | | | | | | | | | | | | | |
|---|--------|--------|---|-----|--------|-------|---------------------|------|------|------|------|------|--------|--------|--------|--------|--------|--------|----------|
| 1 | GRATIS | GDTANS | 1 | 150 | 50.743 | TACSR | 410 mm ² | 2800 | 1000 | 1000 | 1250 | 2000 | 0.0387 | 0.2807 | 4.0230 | 0.0018 | 0.0134 | 0.0097 | 727.4400 |
| 1 | GRATIS | GDTANS | 2 | 150 | 50.743 | TACSR | 410 mm ² | 2800 | 1000 | 1000 | 1250 | 2000 | 0.0387 | 0.2807 | 4.0230 | 0.0018 | 0.0134 | 0.0097 | 727.4400 |
| 1 | GDTANS | PIERS | 1 | 150 | 21.183 | TACSR | 410 mm ² | 2800 | 1000 | 1000 | 1250 | 2000 | 0.0387 | 0.2807 | 4.0230 | 0.0036 | 0.0264 | 0.0192 | 727.4400 |
| 1 | GDTANS | PIERS | 2 | 150 | 21.183 | TACSR | 410 mm ² | 2800 | 1000 | 1000 | 1250 | 2000 | 0.0387 | 0.2807 | 4.0230 | 0.0036 | 0.0264 | 0.0192 | 727.4400 |

| Unit (MTR) | Code | Qty | Unit | Uprate (kg/kV) | Uprate (km) | Unit | Remarks | Uprate (kg/kV) | Uprate (km) | Unit | Remarks | Uprate (kg/kV) | Uprate (km) | Unit | Remarks | Uprate (kg/kV) | Uprate (km) | Unit | Remarks |
|------------|------|-----|------|----------------|-------------|------|---------|----------------|-------------|------|---------|----------------|-------------|------|---------|----------------|-------------|------|---------|
|------------|------|-----|------|----------------|-------------|------|---------|----------------|-------------|------|---------|----------------|-------------|------|---------|----------------|-------------|------|---------|

2. UPT Jember

| | | | | | | | | | | | | | | | | | | |
|---|---------|---------|----------|-----|--------|-----------------|------------|------|------|------|------|--------|--------|--------|--------|--------|----------|----------|
| 1 | PLRG05 | LAANG5 | 1 | 150 | 51.652 | ACSR DOVE | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | PLRG05 | LAANG5 | 2 | 150 | 51.652 | ACSR DOVE | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | LAANG5 | TGGUL5 | 1 | 150 | 32.080 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | LAANG5 | JMIBERS | 1 | 150 | 50.530 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | TNGUL5 | JMIBERS | 1 | 150 | 26.450 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | JMIBERS | BWNGIS | 1 | 150 | 87.585 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | JMIBERS | GTENG5 | 1 | 150 | 64.708 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | BOWS05 | JMIBERS | 1 | 150 | 39.682 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | BOWS05 | JMIBERS | 2 | 150 | 39.682 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | STBDO5 | PTONG5 | 1 | 150 | 55.433 | ACSR ZEBRA | 2 x 435 mm | 1620 | 2000 | 2000 | 1250 | 0.0387 | 0.2807 | 4.0230 | 0.0095 | 0.0502 | 420.8760 | |
| 1 | STBDO5 | PTONG5 | 2 | 150 | 55.433 | ACSR ZEBRA | 2 x 435 mm | 1620 | 2000 | 2000 | 1250 | 0.0387 | 0.2807 | 4.0230 | 0.0095 | 0.0502 | 420.8760 | |
| 1 | STBDO5 | BOWS05 | 1 | 150 | 34.766 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | STBDO5 | BOWS05 | 2 | 150 | 34.766 | ACSR AW | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | STBDO5 | BWNGIS | 1 | 150 | 74.270 | ACSR DOVE | 2 x 340 mm | 1200 | 300 | 900 | 1600 | 1570 | 0.2560 | 0.2713 | 4.0740 | 0.0192 | 0.0680 | 314.7660 |
| 1 | STBDO5 | BWNGIS | 2 | 150 | 74.270 | ACSR DOVE | 2 x 340 mm | 1200 | 300 | 900 | 1600 | 1570 | 0.2560 | 0.2713 | 4.0740 | 0.0192 | 0.0680 | 314.7660 |
| 1 | GTENG5 | BWNGIS | 1 | 150 | 33.950 | ACSR DOVE | 330 mm | 140 | 900 | 900 | 900 | 0.1172 | 0.2503 | 2.8530 | 0.0269 | 0.0332 | 192.2520 | |
| 1 | BWNGIS | KTPNG5 | 1 | 150 | 7.990 | ACSR HAWK | 240 MM4 | 655 | 500 | 500 | 800 | 0.1370 | 0.3555 | 2.8820 | 0.0049 | 0.0141 | 170.1690 | |
| 1 | BWNGIS | KTP-GLM | 1(11) | 150 | 4.471 | SUBMARINE CABLE | 330 mm | 500 | 500 | 900 | 900 | 0.1513 | 0.3482 | 2.8820 | 0.0032 | 0.0088 | 149.5000 | |
| 1 | BWNGIS | KTP-GLM | 1(banda) | 150 | 4.879 | DSS | 300 mm | 465 | 600 | 800 | 800 | 0.1330 | 0.3556 | 2.8820 | 0.0049 | 0.0141 | 170.1690 | |
| 1 | BWNGIS | KTPNG5 | 2 | 150 | 7.990 | ACSR HAWK | 477 MM | 555 | 600 | 800 | 800 | 0.1643 | 0.3548 | 2.8820 | 0.0052 | 0.0095 | 129.9060 | |
| 1 | BWNGIS | KTP-GLM | 1(10) | 150 | 4.388 | SUBMARINE CABLE | 300 mm | 500 | 600 | 600 | 600 | 0.1643 | 0.3548 | 2.8820 | 0.0052 | 0.0095 | 129.9060 | |
| 1 | BWNGIS | KTP-GLM | 1(banda) | 150 | 4.260 | DCS | 300 mm | 465 | 600 | 800 | 800 | 0.1643 | 0.3548 | 2.8820 | 0.0052 | 0.0095 | 129.9060 | |

11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

| Unit | Detail | No. | No. Surat | Leg (W) | Jenis (J) | Unit | Proporitas | Jenis | Item | Item | Kem. 1 | Kem. 2 | Kem. 3 | Kem. 4 | Kem. 5 | Dimensi | Jumlah | Wt. Smp | Wt. Smp | Wt. Smp | Wt. Smp | Wt. Smp | Wt. Smp | Kem. 1 | Kem. 2 | Kem. 3 | Kem. 4 | Kem. 5 | Kem. 6 | Kem. 7 | Kem. 8 |
|------|--------|-----|-----------|---------|-----------|------|------------|-------|------|------|--------|--------|--------|--------|--------|---------|--------|---------|---------|---------|---------|---------|---------|--------|--------|--------|--------|--------|--------|--------|--------|
|------|--------|-----|-----------|---------|-----------|------|------------|-------|------|------|--------|--------|--------|--------|--------|---------|--------|---------|---------|---------|---------|---------|---------|--------|--------|--------|--------|--------|--------|--------|--------|

Sub Region Bali

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--------|---------|-----|--|--------|---------------|--------|-----|-----|-----|-----|-----|-----|-----|-----|--------|--------|--------|--------|--------|--------|----------|--|--|--|--|--|--|--|--|--|--|--|
| 1 | KAPALS | PEGRANS | 130 | | 11.430 | ACSR PARTIDGE | 135 mm | 400 | 300 | 200 | 500 | 600 | 600 | 600 | 600 | 0.2135 | 0.4084 | 2.7570 | 0.0107 | 0.0223 | 0.0094 | 103.9200 | | | | | | | | | | | |
| 1 | BRITIS | PMRONS | 150 | | 20.470 | ACSR PARTIDGE | 135 mm | 400 | 300 | 200 | 600 | 600 | 600 | 600 | 600 | 0.2115 | 0.4054 | 2.7570 | 0.0154 | 0.0371 | 0.0128 | 103.9200 | | | | | | | | | | | |

Pengembangan Proyek

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--------|---------|-----|--|--------|---------------|--------|-----|------|------|------|------|------|------|------|--------|--------|--------|--------|--------|--------|----------|--|--|--|--|--|--|--|--|--|--|--|
| 1 | KAPALS | PEGRANS | 130 | | 11.720 | ACSR HAWK | 477 mm | 600 | 500 | 500 | 600 | 600 | 600 | 600 | 600 | 0.1370 | 0.5585 | 7.8270 | 0.0105 | 0.0304 | 0.0112 | 103.9200 | | | | | | | | | | | |
| 1 | PIBANS | PSGRNS | 130 | | 1.400 | ACSR HAWK | 477 mm | 645 | 1000 | 1000 | 800 | 800 | 800 | 800 | 800 | 0.1370 | 0.5385 | 7.8270 | 0.0245 | 0.0130 | 0.0048 | 167.5770 | | | | | | | | | | | |
| 1 | KAPALS | UBUDS | 130 | | 18.730 | ACSR PARTIDGE | 135 mm | 400 | 600 | 500 | 1200 | 1200 | 1200 | 1200 | 1200 | 0.2135 | 0.4084 | 2.7970 | 0.0158 | 0.0304 | 0.0105 | 103.9200 | | | | | | | | | | | |
| 1 | UBUDS | BRITIS | 150 | | 25.440 | ACSR PARTIDGE | 135 mm | 400 | 600 | 800 | 1200 | 1200 | 1200 | 1200 | 1200 | 0.2135 | 0.4084 | 2.7970 | 0.0179 | 0.0334 | 0.0185 | 103.9200 | | | | | | | | | | | |

Informasi

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--------|--------|-----|--|--------|---------------|--------|-----|-----|-----|------|------|------|------|------|--------|--------|--------|--------|--------|--------|----------|--|--|--|--|--|--|--|--|--|--|--|
| 1 | KAPALS | UBUDS | 130 | | 22.500 | ACSR PARTIDGE | 135 mm | 400 | 600 | 800 | 1200 | 1200 | 1200 | 1200 | 1200 | 0.2135 | 0.4084 | 2.7970 | 0.0214 | 0.0408 | 0.0142 | 103.9200 | | | | | | | | | | | |
| 1 | UBUDS | BRITIS | 150 | | 25.670 | ACSR PARTIDGE | 135 mm | 400 | 600 | 900 | 1200 | 1200 | 1200 | 1200 | 1200 | 0.2135 | 0.4084 | 2.7970 | 0.0244 | 0.0466 | 0.0162 | 103.9200 | | | | | | | | | | | |

Rencana Up-rating

| | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|---|--------|--------|-----|--|--------|-------|---------------------|-----|-----|-----|-----|-----|-----|-----|-----|--------|--------|--------|--------|--------|--------|----------|--|--|--|--|--|--|--|--|--|--|--|--|--|
| 1 | KAPALS | UBUDS | 130 | | 22.500 | TACSR | 160 mm ² | 734 | 600 | 400 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0220 | 0.0390 | 0.0150 | 190.6532 | | | | | | | | | | | | | |
| 1 | UBUDS | BRITIS | 150 | | 25.670 | TACSR | 160 mm ² | 734 | 600 | 400 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0251 | 0.0445 | 0.0171 | 190.6532 | | | | | | | | | | | | | |
| 1 | BRITIS | PMRONS | 130 | | 20.450 | TACSR | 160 mm ² | 734 | 600 | 200 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0200 | 0.0354 | 0.0136 | 190.6532 | | | | | | | | | | | | | |
| 1 | UBUDS | PMRONS | 150 | | 20.430 | TACSR | 160 mm ² | 734 | 600 | 200 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0200 | 0.0354 | 0.0136 | 190.6532 | | | | | | | | | | | | | |
| 1 | KAPALS | PMRONS | 150 | | 26.580 | TACSR | 160 mm ² | 734 | 600 | 400 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0273 | 0.0515 | 0.0290 | 190.6532 | | | | | | | | | | | | | |
| 1 | KAPALS | BRITIS | 150 | | 28.170 | TACSR | 160 mm ² | 734 | 600 | 800 | 830 | 830 | 830 | 830 | 830 | 0.2200 | 0.3900 | 2.9600 | 0.0373 | 0.0662 | 0.0254 | 190.6532 | | | | | | | | | | | | | |

```

unit uEvoPro;

interface

uses uUtils,uRandom,uObjFunc,uParamTS;

type
  TIndividu1=record
    chrom:dArr1;
    fitness:double;
  end;

  TPopulasi1=array of TIndividu1;

  TIndividu2=record
    chrom:dArr2;
    fitness:double;
  end;

  TPopulasi2=array of TIndividu2;

  TEvoPro=class
  private
    FMaxGen,FPopSize,FLength:integer;
    FBetha:double;
    FRandom:TRandomu;
    function getMin:dArr1;
    function getAvg:dArr1;
    function getMax:dArr1;
  protected
    FMin,FMax,FAvg:dArr1;
  public
    constructor Create(const rMaxGen,rPopSize,rLength:integer;
      const rBetha:double);
    destructor Destroy;override;
    property MaxGen:integer read FMaxGen write FMaxGen;
    property PopSize:integer read FPopSize write FPopSize;
    property Length:integer read FLength write FLength;
    property Betha:double read FBetha write FBetha;
    property Min:dArr1 read getMin;
    property Avg:dArr1 read getAvg;
    property Max:dArr1 read getMax;
  end;

  TEvoPro1=class(TEvoPro)
  private

```

```

FBatas:TBatasArr1;
FBestIndi:TIndividu1;
FParent,FChild:TPopulasi1;
FMin1,FAvg1,FMax1,FKa,FPmutasi:double;
function getBatas:TBatasArr1;
procedure setBatas(const rBatas:TBatasArr1);
function getIndividu(const rIndi:TIndividu1):TIndividu1;
procedure SwapIndi(var rIndi1,rIndi2:TIndividu1);
function FindFitnessMax:double;
function FindIndiMax:TIndividu1;
procedure InitParent;
procedure Statistik;
procedure Generasi;
procedure Kompetisi;
procedure doHitung;
function getBestChrom:dArr1;
public
constructor Create(const rMaxGen,rPopSize,rLength:integer;
                  const rBetha,rKa,rPmutasi:double;
                  const rBatas:TBatasArr1);
property Batas:TBatasArr1 read getBatas write setBatas;
property Ka:double read FKa write FKa;
property BestChrom:dArr1 read getBestChrom;
end;

```

implementation

```

{ TEvoPro }
//constructor
constructor TEvoPro.Create(const rMaxGen,rPopSize,rLength:integer;
                          const rBetha:double);
begin
  inherited Create;
  FMaxGen:=rMaxGen;
  FPopSize:=rPopSize;
  FLength:=rLength;
  FBetha:=rBetha;
  FRandom:=TRandomu.Create;
end;

//destructor
destructor TEvoPro.Destroy;
begin
  try
    FRandom.Free;
  finally

```

```

SetLength(FBatas,Length);
for i:=0 to Length-1 do
begin
  FBatas[i].min:=rBatas[i].min;
  FBatas[i].max:=rBatas[i].max;
end;
end;

function TEvoPro1.getBatas:TBatasArr1;
var i:integer;
begin
  SetLength(result,Length);
  for i:=0 to Length-1 do
  begin
    result[i].min:=FBatas[i].min;
    result[i].max:=FBatas[i].max;
  end;
end;

procedure TEvoPro1.setBatas(const rBatas:TBatasArr1);
var i:integer;
begin
  SetLength(FBatas,Length);
  for i:=0 to Length-1 do
  begin
    FBatas[i].min:=rBatas[i].min;
    FBatas[i].max:=rBatas[i].max;
  end;
end;

//data processing
function TEvoPro1.getIndividu(const rIndi:TIndividu1):TIndividu1;
var i:integer;
begin
  SetLength(result.chrom,Length);
  for i:=0 to Length-1 do
  begin
    result.chrom[i]:=rIndi.chrom[i];
  end;
  result.fitness:=rIndi.fitness;
end;

procedure TEvoPro1.SwapIndi(var rIndi1,rIndi2:TIndividu1);
var tmp:TIndividu1;
begin
  tmp:=getIndividu(rIndi1);

```

```

rIndi1:=getIndividu(rIndi2);
rIndi2:=getIndividu(tmp);
end;

function TEvoPro1.FindFitnessMax:double;
var i:integer;
begin
result:=FParent[0].fitness;
for i:=0 to PopSize-1 do
begin
if result<FParent[i].fitness then
begin
result:=FParent[i].fitness;
end;
end;
end;

function TEvoPro1.FindIndiMax:TIndividu1;
var i:integer;
begin
result:=getIndividu(FParent[0]);
for i:=1 to PopSize-1 do
begin
if result.fitness<FParent[i].fitness then
begin
result:=getIndividu(FParent[i]);
end;
end;
end;

procedure TEvoPro1.InitParent;
var i,j:integer;
begin
SetLength(FParent,FPopSize);
SetLength(FChild,FPopSize);
SetLength(FMin,FMaxGen);
SetLength(FAvg,FMaxGen);
SetLength(FMax,FMaxGen);
for i:=0 to PopSize-1 do
begin
SetLength(FParent[i].chrom,Length);
SetLength(FChild[i].chrom,Length);
for j:=0 to Length-1 do
begin
FParent[i].chrom[j]:=FRandom.NextDouble(FBatas[j].min,FBatas[j].max);
end;
end;

```

```

    FParent[i].fitness:=FKa/gObjFunc.doHitung(FParent[i].chrom);
end;
end;

```

```

procedure TEvoPro1.Statistik;
var i:integer;
    sumFitness:double;
begin
    sumFitness:=0;
    FMin1:=FParent[0].fitness;
    FMax1:=FParent[0].fitness;
    for i:=1 to PopSize-1 do
    begin
        sumFitness:=sumFitness+FParent[i].fitness;
        if FMin1>FParent[i].fitness then
        begin
            FMin1:=FParent[i].fitness;
        end;
        if FMax1<FParent[i].fitness then
        begin
            FMax1:=FParent[i].fitness;
        end;
    end;
    FAvg1:=sumFitness/FPopSize;
end;

```

```

procedure TEvoPro1.Generasi;
var i,j:integer;
    Fmax,fitEP,fitES:double;
    tho:double;
    chromEP,chromES:dArr1;
begin
    Fmax:=FindFitnessMax;
    SetLength(chromEP,Length);
    SetLength(chromES,Length);
    for i:=0 to PopSize-1 do
    begin
        for j:=0 to Length-1 do
        begin
            tho:=(FBatas[j].max-FBatas[j].min)*((Fmax-FParent[i].fitness)/
                Fmax+Betha);
            chromEP[j]:=FParent[i].chrom[j]+FRandom.NextGaussian(0,sqr(tho));
            if chromEP[j]>FBatas[j].max then
            begin
                chromEP[j]:=FBatas[j].max;
            end;
        end;
    end;
end;

```

```

    if chromEP[j]<FBatas[j].min then
    begin
        chromEP[j]:=FBatas[j].min;
    end;
end;
fitEP:=FKa/gObjFunc.doHitung(chromEP);
for j:=0 to Length-1 do
begin
    chromES[j]:=FParent[i].chrom[j]+FRandom.NextGaussian(0,FPmutasi);
    if chromES[j]>FBatas[j].max then
    begin
        chromES[j]:=FBatas[j].max;
    end;
    if chromES[j]<FBatas[j].min then
    begin
        chromES[j]:=FBatas[j].min;
    end;
end;
fitES:=FKa/gObjFunc.doHitung(chromES);
if fitEP>fitES then
begin
    for j:=0 to Length-1 do
    begin
        FChild[i].chrom[j]:=chromEP[j];
    end;
    FChild[i].fitness:=fitEP;
end
else
begin
    for j:=0 to Length-1 do
    begin
        FChild[i].chrom[j]:=chromES[j];
    end;
    FChild[i].fitness:=fitES;
end;
end;
end;

procedure TEvoPro1.Kompetisi;
var i,j,Ntmp,sa:integer;
    tmp:TPopulasi1;
    sort:iArr1;
begin
    Ntmp:=2*PopSize;
    SetLength(tmp,Ntmp);
    for i:=0 to PopSize-1 do

```

```

begin
  tmp[i]:=getIndividu(FParent[i]);
  tmp[PopSize+i]:=getIndividu(FChild[i]);
end;
SetLength(sort,Ntmp);
for i:=0 to Ntmp-1 do
begin
  sort[i]:=0;
end;
for i:=0 to Ntmp-1 do
begin
  for j:=0 to Ntmp-2 do
  begin
    repeat
      sa:=FRandom.NextInt(0,(Ntmp-1));
    until sa<>i;
    if tmp[i].fitness>tmp[sa].fitness then
    begin
      inc(sort[i]);
    end;
  end;
end;
for i:=0 to Ntmp-2 do
begin
  for j:=i to Ntmp-1 do
  begin
    if sort[i]<sort[j] then
    begin
      Swap(sort[i],sort[j]);
      SwapIndi(tmp[i],tmp[j]);
    end;
  end;
end;
for i:=0 to PopSize-1 do
begin
  FParent[i]:=getIndividu(tmp[i]);
end;
end;

```

```

procedure TEvoPro1.doHitung;
var gen:integer;
    TempIndi:TIndividu1;
begin
  InitParent;
  Statistik;
  FBestIndi:=FindIndiMax;

```

```

gen:=1;
repeat
  Generasi;
  Kompetisi;
  Statistik;
  TempIndi:=FindIndiMax;
  if FBestIndi.fitness<TempIndi.fitness then
  begin
    FBestIndi:=GetIndividu(TempIndi);
  end;
  FMin[gen-1]:=FMin1;
  FAvg[gen-1]:=FAvg1;
  FMax[gen-1]:=FMax1;
  frmParamTS.pblIterasi.StepBy(1);
  inc(gen);
until (gen>MaxGen);
end;

//data output
function TEvoPro1.getBestChrom:dArr1;
var i:integer;
begin
  doHitung;
  SetLength(result,Length);
  for i:=0 to Length-1 do
  begin
    result[i]:=FBestIndi.chrom[i];
  end;
end;

end.

```

```

unit uObjFunc;

interface

uses uUtils,uNewtonPolar,uLoadflow;

type
  TObjFunc=class
  private
    FLamdaV,FLamdaQ,FLamdaS:double;
    FBatasV:TBatas;
    FBatasChrom:TBatasArr1;
    function getBatasV:TBatas;
    procedure setBatasV(const rBatasV:TBatas);
    function DecodeChromBin(const rChrom:bArr2):dArr1;
    function DecodeChrom(const rChrom:dArr1):dArr2;
    function FindPinV(const rLBus:dArr2):integer;
    function FindPinKapBranch(const rLBranch:dArr2):double;
    function FindPinKapQgen(const rLBus:dArr2):double;
    function FindCostGen(const rNo:integer;
      const rP,rQ:double):double;
  public
    constructor Create(const rLamdaV,rLamdaQ,rLamdaS:double;
      const rBatasV:TBatas);
    function FindLength:integer;
    function FindBatasChrom(var rBatasV:TBatas):TBatasArr1;
    function doHitung(const rChrom:dArr1):double;
    function DecodeChromAkhir(const rChrom:dArr1):dArr2;
    function FindTotalCostGen(const rLBus:dArr2):double;
    procedure doHitungAkhir(const rChrom:dArr1;
      var rLBus,rLBranch:dArr2;
      var rGeneral:TGeneral;
      var rCostTotal:double);
    property LamdaV:double read FLamdaV write FLamdaV;
    property LamdaQ:double read FLamdaQ write FLamdaQ;
    property LamdaS:double read FLamdaS write FLamdaS;
    property BatasV:TBatas read getBatasV write setBatasV;
  end;

var gObjFunc:TObjFunc;

implementation

//constructor
constructor TObjFunc.Create(const rLamdaV,rLamdaQ,rLamdaS:double;
  const rBatasV:TBatas);

```

```

begin
  inherited Create;
  FLamdaV:=rLamdaV;
  FLamdaQ:=rLamdaQ;
  FLamdaS:=rLamdaS;
  FBatasV.min:=rBatasV.min;
  FBatasV.max:=rBatasV.max;
  FBatasChrom:=FindBatasChrom(FBatasV);
end;

//data accessing
function TObjFunc.getBatasV:TBatas;
begin
  result.min:=FBatasV.min;
  result.max:=FBatasV.max;
end;

procedure TObjFunc.setBatasV(const rBatasV:TBatas);
begin
  FBatasV.min:=rBatasV.min;
  FBatasV.max:=rBatasV.max;
end;

//data processing
function TObjFunc.DecodeChromBin(const rChrom:bArr2):dArr1;
var i,j,Param,Length:integer;
    chrom:bArr2;
    val:dArr1;
begin
  Param:=high(rChrom)+1;
  Length:=high(rChrom[0])+1;
  SetLength(chrom,Param+1,Length+1);
  for i:=1 to Param do
    begin
      for j:=1 to Length do
        begin
          chrom[i,j]:=rChrom[i-1,j-1];
        end;
      end;
    val:=DecodeBinToFloat2(chrom);
    SetLength(result,Param);
    for i:=0 to Param-1 do
      begin
        result[i]:=GetBatasToReal(val[i+1],FBatasChrom[i].min,
          FBatasChrom[i].max);
      end;
    end;
end;

```

```
end;
```

```
function TObjFunc.FindLength:integer;
```

```
var i:integer;
```

```
begin
```

```
  result:=0;
```

```
  for i:=0 to high(gBus) do
```

```
  begin
```

```
    if gBus[i,7]=1 then
```

```
    begin
```

```
      inc(result);
```

```
    end
```

```
    else if gBus[i,7]=2 then
```

```
    begin
```

```
      inc(result);
```

```
      inc(result);
```

```
    end
```

```
    else if gBus[i,3]<>0 then
```

```
    begin
```

```
      inc(result);
```

```
    end;
```

```
  end;
```

```
end;
```

```
function TObjFunc.FindBatasChrom(var rBatasV:TBatas):TBatasArr1;
```

```
var i,sa,Ngen:integer;
```

```
begin
```

```
  sa:=FindLength;
```

```
  SetLength(result,sa);
```

```
  sa:=0;
```

```
  Ngen:=0;
```

```
  for i:=0 to high(gBus) do
```

```
  begin
```

```
    if gBus[i,7]=1 then
```

```
    begin
```

```
      result[sa].min:=rBatasV.min;
```

```
      result[sa].max:=rBatasV.max;
```

```
      inc(sa);
```

```
      inc(Ngen);
```

```
    end
```

```
    else if gBus[i,7]=2 then
```

```
    begin
```

```
      result[sa].min:=rBatasV.min;
```

```
      result[sa].max:=rBatasV.max;
```

```
      inc(sa);
```

```
      result[sa].min:=gGen[Ngen,8];
```

```

    result[sa].max:=gGen[Ngen,9];
    inc(sa);
    inc(Ngen);
end
else if gBus[i,3]<>0 then
begin
    result[sa].min:=0;
    result[sa].max:=gBus[i,3];
    inc(sa);
end;
end;
end;

```

```

function TObjFunc.DecodeChrom(const rChrom:dArr1):dArr2;
var i,j,sa:integer;
begin
    SetLength(result,high(gBus)+1,high(gBus[0])+1);
    for i:=0 to high(gBus) do
    begin
        for j:=0 to high(gBus[0]) do
        begin
            result[i,j]:=gBus[i,j];
        end;
    end;
    sa:=0;
    for i:=0 to high(gBus) do
    begin
        if gBus[i,7]=1 then
        begin
            result[i,0]:=rChrom[sa];
            inc(sa);
        end
        else if gBus[i,7]=2 then
        begin
            result[i,0]:=rChrom[sa];
            inc(sa);
            result[i,2]:=rChrom[sa];
            inc(sa);
        end
        else if gBus[i,3]<>0 then
        begin
            result[i,3]:=rChrom[sa];
            inc(sa);
        end;
    end;
end;
end;

```

```

function TObjFunc.FindPinV(const rLBus:dArr2):integer;
var i:integer;
begin
  result:=0;
  for i:=0 to high(rLBus) do
  begin
    if rLBus[i,0]>FBatasV.max then
    begin
      inc(result);
    end;
    if rLBus[i,0]<FBatasV.min then
    begin
      inc(result);
    end;
  end;
end;

```

```

function TObjFunc.FindPinKapBranch(const rLBranch:dArr2):double;
var i:integer;
    absAlir:double;
begin
  result:=0;
  for i:=0 to high(rLBranch) do
  begin
    if rLBranch[i,9]>0 then
    begin
      absAlir:=sqrt(sqrt(rLBranch[i,9])+sqrt(rLBranch[i,10]));
      if absAlir>rLBranch[i,8] then
      begin
        result:=result+abs(absAlir-rLBranch[i,8]);
      end;
    end;
  end;
  else
  begin
    absAlir:=sqrt(sqrt(rLBranch[i,11])+sqrt(rLBranch[i,12]));
    if absAlir>rLBranch[i,8] then
    begin
      result:=result+abs(absAlir-rLBranch[i,8]);
    end;
  end;
end;
end;

```

```

function TObjFunc.FindPinKapQgen(const rLBus:dArr2):double;
var i,sa:integer;

```

```

begin
  result:=0;
  sa:=0;
  for i:=0 to high(rLBus) do
  begin
    if rLBus[i,7]<>3 then
    begin
      if rLBus[i,3]>gGen[sa,2] then
      begin
        result:=result+rLBus[i,3]-gGen[sa,2];
      end;
      if rLBus[i,3]<gGen[sa,1] then
      begin
        result:=result+abs(gGen[sa,1])-abs(rLBus[sa,3]);
      end;
      inc(sa);
    end;
  end;
end;

function TObjFunc.FindCostGen(const rNo:integer;
  const rP,rQ:double):double;
var CostP,CostQ:double;
begin
  CostP:=0;
  if rP<>0 then
  begin
    CostP:=gGen[rNo,3]*sqr(rP)+gGen[rNo,4]*rP+gGen[rNo,5];
  end;
  CostQ:=rQ*gGen[rNo,7];
  result:=CostP+CostQ+gGen[rNo,6];
end;

function TObjFunc.FindTotalCostGen(const rLBus:dArr2):double;
var i,sa:integer;
begin
  result:=0;
  sa:=0;
  for i:=0 to high(rLBus) do
  begin
    if rLBus[i,7]<>3 then
    begin
      result:=result+FindCostGen(sa,rLBus[i,2],rLBus[i,3]);
      inc(sa);
    end;
  end;
end;

```

end;

```
function TObjFunc.doHitung(const rChrom:dArr1):double;
var LBus:dArr2;
    pinV:integer;
    pinS,PinQ,CostTotal:double;
begin
    LBus:=DecodeChrom(rChrom);
    NewtonPolar(LBus,gBranch,gGencral);
    pinV:=FindPinV(LBus);
    pinS:=FindPinKapBranch(gBranch);
    pinQ:=FindPinKapQgen(LBus);
    CostTotal:=FindTotalCostGen(LBus);
    result:=CostTotal+FLamdaV*pinV+FLamdaS*pinS+FLamdaS*pinQ;
end;
```

```
function TObjFunc.DecodeChromAkhir(const rChrom:dArr1):dArr2;
var i,sa,ne:integer;
begin
    ne:=0;
    for i:=0 to high(gBus) do
    begin
        if gBus[i,7]=1 then
        begin
            inc(ne);
        end
        else if gBus[i,7]=2 then
        begin
            inc(ne);
        end
        else if gBus[i,3]<>0 then
        begin
            inc(ne);
        end;
    end;
    SetLength(result,ne,4);
    sa:=0;
    ne:=0;
    for i:=0 to high(gBus) do
    begin
        if gBus[i,7]=1 then
        begin
            result[ne,0]:=i;
            result[ne,1]:=rChrom[sa];
            inc(sa);
            inc(ne);
        end;
    end;
end;
```

```

end
else if gBus[i,7]=2 then
begin
  result[ne,0]:=i;
  result[ne,1]:=rChrom[sa];
  inc(sa);
  result[ne,2]:=rChrom[sa];
  inc(sa);
  inc(ne);
end
else if gBus[i,3]<>0 then
begin
  result[ne,0]:=i;
  result[ne,3]:=rChrom[sa];
  inc(sa);
  inc(ne);
end;
end;
end;

procedure TObjFunc.doHitungAkhir(const rChrom:dArr1;
  var rLBus,rLBranch:dArr2;
  var rGeneral:TGeneral;
  var rCostTotal:double);
var i,j:integer;
begin
  rLBus:=DecodeChrom(rChrom);
  SetLength(rLBranch,high(gBranch)+1,high(gBranch[0])+1);
  for i:=0 to high(gBranch) do
  begin
    for j:=0 to high(gBranch[0]) do
    begin
      rLBranch[i,j]:=gBranch[i,j];
    end;
  end;
  end;
  rGeneral.MaxIterasi:=gGeneral.MaxIterasi;
  rGeneral.Vbase:=gGeneral.Vbase;
  rGeneral.VKonst:=gGeneral.VKonst;
  rGeneral.Pbase:=gGeneral.Pbase;
  rGeneral.PKonst:=gGeneral.PKonst;
  rGeneral.Toleransi:=gGeneral.Toleransi;
  rGeneral.Param:=gGeneral.Param;
  NewtonPolar(rLBus,rLBranch,rGeneral);
  rCostTotal:=FindTotalCostGen(rLBus);
end;
end.

```

```

unit uRandom;

interface

type
  TRandomu = class
  private
    FNextGaussian:double;
    procedure GetGaussian(var dv1,dv2:double);
  public
    constructor Create;
    function NextInt(const dmax:integer):integer;overload;
    function NextInt(const dmin,dmax:integer):integer;overload;
    function NextDouble:double;overload;
    function NextDouble(const dmin,dmax:double):double;overload;
    function NextBoolean:boolean;overload;
    function NextBoolean(const dflip:double):boolean;overload;
    function NextGaussian:double;overload;
    function NextGaussian(const dmean,dvariance:double):double;overload;
    function NextCauchy(const dmean,dvariance:double):double;
  end;

implementation

constructor TRandomu.Create;
begin
  inherited Create;
  FNextGaussian:=0;
end;

function TRandomu.NextInt(const dmax:integer):integer;
begin
  result:=round(random*dmax);
end;

function TRandomu.NextInt(const dmin,dmax:integer):integer;
begin
  result:=round(dmin+(dmax-dmin)*random);
end;

function TRandomu.NextDouble:double;
begin
  result:=random;
end;

function TRandomu.NextDouble(const dmin,dmax:double):double;

```

```

begin
  result:=dmin+(dmax-dmin)*random;
end;

function TRandomu.NextBoolean:boolean;
begin
  result:=false;
  if random<=0.5 then
  begin
    result:=true;
  end;
end;

function TRandomu.NextBoolean(const dflip:double):boolean;
begin
  result:=false;
  if random<=dflip then
  begin
    result:=true;
  end;
end;

procedure TRandomu.GetGaussian(var dv1,dv2:double);
var s,multiplier:double;
begin
  repeat
    dv1:=2*random-1;
    dv2:=2*random-1;
    s:=dv1*dv1+dv2*dv2;
  until s<=1;
  multiplier:=sqrt((-2*Ln(s))/s);
  dv1:=dv1*multiplier;
  dv2:=dv2*multiplier;
end;

function TRandomu.NextGaussian:double;
begin
  if FNextGaussian<>0 then
  begin
    result:=FNextGaussian;
    FNextGaussian:=0;
  end
  else
  begin
    GetGaussian(result,FNextGaussian);
  end;
end;

```

end;

```
function TRandomu.NextGaussian(const dmean,
    dvariance:double):double;
var gauss:double;
begin
    if FNextGaussian <> 0 then
        begin
            result:=dmean+sqrt(dvariance)*FNextGaussian;
            FNextGaussian:=0;
        end
    else
        begin
            GetGaussian(gauss,FNextGaussian);
            result:=dmean+sqrt(dvariance)*gauss;
        end;
    end;
end;
```

```
function TRandomu.NextCauchy(const dmean,dvariance:double):double;
var v1,v2:double;
begin
    v1:=NextGaussian(dmean,dvariance);
    v2:=NextGaussian(dmean,dvariance);
    if abs(v2)>0.001 then
        begin
            result:=v1/v2;
        end
    else
        begin
            result:=0;
        end;
    end;
end;
```

end.

Optimal Reactive Power Planning Using Evolutionary Algorithms: A Comparative Study for Evolutionary Programming, Evolutionary Strategy, Genetic Algorithm, and Linear Programming

Kwang. Y. Lee, Senior Member, IEEE, and Frank F. Yang
Department of Electrical Engineering
The Pennsylvania State University
University Park, PA 16802

Abstract: This paper presents a comparative study for three evolutionary algorithms (EAs) to the Optimal Reactive Power Planning (ORPP) problem: evolutionary programming, evolutionary strategy, and genetic algorithm. The ORPP problem is decomposed into P- and Q-optimization modules, and each module is optimized by the EAs in an iterative manner to obtain the global solution. The EA methods for the ORPP problem are evaluated against the IEEE 30-bus system as a common testbed, and the results are compared against each other and with those of linear programming.

Keywords: Optimal reactive power planning, evolutionary algorithms, genetic algorithm.

I. INTRODUCTION

In general, the problem of optimal reactive power planning (ORPP) can be defined as to determine the amount and location of shunt reactive power compensation devices needed for minimum cost while keeping an adequate voltage profile. The ORPP is one of the most challenging problems since both objective functions, the operation cost and the investment cost of new reactive power sources, should be minimized simultaneously. The ORPP is a large-scale nonlinear optimization problem with a large number of variables and uncertain parameters. Various mathematical optimization algorithms have been developed for the ORPP, which in most cases, use nonlinear [1], linear [2], or mixed integer programming [3], and decomposition methods [4-7]. However, these conventional techniques are known to converge to a local optimal solution rather than the global one for problems such as ORPP which have many local minima.

PE-858-PWRS-1-04-1997 A paper recommended and approved by the IEEE Power System Analysis, Computing and Economics Committee of the IEEE Power Engineering Society for publication in the IEEE Transactions on Power Systems. Manuscript submitted January 2, 1997; made available for printing April 11, 1997.

Recently, evolutionary algorithms (EAs) [8-15] have been used for optimization; in particular both the genetic algorithm and evolutionary programming have been used in the ORPP problem. The EA is a powerful optimization technique analogous to the natural selection process in genetics. Theoretically, this technique converges to the global optimum solution with probability one. It is useful especially when other optimization methods fail in finding the optimal solution. Evolutionary algorithm is an inherently parallel process. Recent advances in distributed processing architectures could result in dramatically reduced execution times, and it is now possible to do a large amount of computation in order to obtain the global instead of a local optimal solution.

This paper investigates the applicability of the following three different EAs in the ORPP problem: evolutionary programming (EP), evolutionary strategy (ES), and genetic algorithm (GA). Rather than the usual approach of loss minimization [11], the fuel cost minimization approach [4-7,10] is adopted as a direct measure of operation cost since the loss minimization does not guarantee the optimal operation with minimum fuel. The ORPP problem is then decomposed into the real power (P) and the reactive power (Q) optimization problem. The P-optimization is to minimize the operation cost by adjusting real power generation; while the Q-optimization is to adjust reactive power generation, transformer tap-settings, and the amount of Var source investment. The EA methods are evaluated against the IEEE 30-bus system [4-7] as a common testbed for comparison with each other and with linear programming.

II. EVOLUTIONARY ALGORITHMS

The EAs, including Evolutionary Programming (EP), Evolutionary Strategy (ES), and Genetic Algorithm (GA), are artificial intelligence methods for optimization based on the mechanics of natural selection, such as mutation, recombination, reproduction, selection, etc. Mutation randomly perturbs a candidate solution; recombination randomly mixes their parts to form a novel solution; reproduction replicates the most successful solutions found in a population; whereas selection purges poor solutions from a population. Starting from an initial generation of

candidate solutions, this process produces advanced generations with candidates that are successively better suited to their environment.

These methods share many similarities. The EP is introduced first, and followed by BS and GA.

A. Evolutionary Programming [11]

1) *Initialization*: The initial population of control variables is selected randomly from the set of uniformly distributed control variables ranging over their upper and lower limits. The fitness score f_i is obtained according to the objective function and the environment.

2) *Statistics*: The maximum fitness f_{max} , minimum fitness f_{min} , the sum of fitness Σf , and average fitness f_{avg} of this generation are calculated.

3) *Mutation*: Each selected parent, for example P_i , is mutated and added to its population following the rule:

$$P_{i+m,j} = P_{i,j} + N(0, \beta (\bar{x}_j - \underline{x}_j) \frac{f_i}{f_{max}}), \quad j=1, 2, \dots, n, \quad (1)$$

where n is the number of decision variables in an individual, $P_{i,j}$ denotes the j^{th} element of the i^{th} individual; $N(\mu, \sigma^2)$ represents a Gaussian random variable with mean μ and variance σ^2 ; f_{max} is the maximum fitness of the old generation which is obtained in *statistics*; \bar{x}_j and \underline{x}_j are, respectively, maximum and minimum limits of the j^{th} element; and β is the mutation scale, $0 < \beta \leq 1$, that could be adaptively decreased during generations.

If any mutated value exceeds its limit, it will be given the limit value. The mutation process (1) allows an individual with larger fitness to produce more offspring for the next generation [11].

4) *Competition*: Several individuals (k) which have the best fitness are kept as the parents for the next generation. Other individuals in the combined population of size $(2m - k)$ have to compete with each other to get their chances for the next generation. A weight value W_i of the i^{th} individual is calculated by the following competition:

$$W_i = \sum_{t=1}^N W_{i,t}, \quad (2)$$

where N is the competition number generated randomly; $W_{i,t}$ is either 0 for loss or 1 for win as the i^{th} individual competes with a randomly selected (r^{th}) individual in the combined population. The value of $W_{i,t}$ is given in the following equation:

$$W_{i,t} = \begin{cases} 1 & \text{if } U_i < \frac{f_r}{f_r + f_i} \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where f_r is the fitness of the randomly selected r^{th} individual, and the f_i is the fitness of the i^{th} individual; and U_i is randomly selected from a uniformly distributed set, $U(0,1)$. When all $2m$ individuals, get their competition weights, they will be ranked in a descending order according to their corresponding value W_i . The first m individuals are selected along with their corresponding fitness f_i to be the bases for the next generation. The maximum, minimum and the average fitness and the sum of fitness of current generation are then calculated in the *statistics*.

5) *Convergence test*: If the convergence condition is not met, the *mutation* and the *competition* processes will run again. The maximum generation number can be used for convergence condition. Other criteria, such as the ratio of the average and the maximum fitness of the population is computed and generations are repeated until

$$\{f_{avg} / f_{max}\} \geq \delta \quad (4)$$

where δ should be very close to 1, which represents the degree of satisfaction. If the convergence has reached a given accuracy, an optimal solution has been found for an optimization problem.

B. Evolutionary Strategy [9,12]

The evolutionary strategy is very similar to the evolutionary programming, and the difference is as follows:

In the *mutation* process, each selected parent, for example P_i , is mutated and added to its population following the rule,

$$P_{i+m,j} = P_{i,j} + N(0, \beta \nabla_{dev}), \quad j=1, 2, \dots, n, \quad (5)$$

where n is the number of decision variables, ∇_{dev} is fixed, and its value depends on the size of decision variables.

In the *competition* process, the fitness of individuals of population size $2m$ are sorted in a descending order. The first m individuals are kept as the parents for the next *mutation* process.

C. Genetic Algorithm [8,9,10-15]

Genetic algorithm (GA) emphasizes models of DNA selection as observed in nature, such as crossover and mutation, and are applied to abstracted chromosomes. This will be easily realized by a string representation, which costs additional encoding and decoding time. This is contrast in to BS and EP, which emphasize mutational transformations that maintain behavioral linkage between each parent and its offspring. The GA used in this paper is very similar to the algorithm that can be found in the standard literature on the topic [8,9,10-15], also known as the *simple genetic algorithm*. We use the three-operator GA with only minor deviations from the original.

1) *Initial population generation:* In this scheme, an initial population of binary strings is created randomly. Each of these strings represents one feasible solution to the search problem, i.e., a point in the search space or a domain satisfying constraints.

2) *Fitness evaluation:* Next the solution strings are converted into their decimal equivalents and each candidate solution is tested in its environment. The fitness of each candidate is evaluated through some appropriate measure, such as the inverse of the cost function:

$$f = 1/(\alpha + C), \quad (6)$$

where C is the cost function to be minimized and α is the fitness function parameter.

The algorithm is driven towards maximizing this fitness measure. After the fitness of the entire population has been determined, it must be determined whether or not the termination criterion has been satisfied. This criterion can be any number of things. One possibility is to stop the algorithm at some finite number of generations and designate the result as the best fit from the population. Another possibility is to test if the average fitness of the population exceeds some fraction of the best fit in the population. If the criterion is not satisfied then we continue with the three genetic operations of *reproduction, crossover, and mutation*.

3) *Selection and reproduction:* Fitness-proportionate reproduction is effected through the simulated spin of a weighted roulette wheel. The roulette wheel is biased with the fitness of each of the solution candidates. The wheel is spun N times where N is the number of strings in the population. This operation yields a new population of strings that reflect the fitness of the previous generation's fit candidates.

4) *Crossover:* The next operation, crossover, is performed on two strings at a time that are selected from the population at random. Crossover involves choosing a random position in the two strings and swapping the bits that occur after this position. In one generation the crossover operation is performed on a specified percentage of the population. Crossover can occur at a single position (single crossover), or at number of different positions (multiple crossover). Crossover can also be performed in two different means: Tail-tail and head-tail crossovers [10,13]. The tail-tail crossover is the usual crossover, where the tail ends of the two strings are swapped. In the head-tail crossover, on the other hand, the tail end of one string becomes the head of another string and vice-versa. The tail-tail crossover tends to change the less significant bits; while the head-tail crossover gives more chance of changes by changing the more significant bits. The two crossover methods can be changed during iterations; the head-tail crossover can be

used in the earlier generations and then switched to tail-tail crossover in the later generations for fine tuning.

5) *Mutation:* The final genetic operator in the algorithm is mutation. Mutation is performed sparingly, typically after every 100-1000 bit transfers from crossover, and it involves selecting a string at random as well as a bit position at random and changing it from a 1 to a 0 or vice-versa. It is used to escape from a local minimum. After mutation, the new generation is complete and the procedure begins again with the *fitness evaluation* of the population.

III. PROBLEM FORMULATION

The optimal reactive power planning (ORPP) problem is to determine the optimal investment of Var sources over a planning horizon [5]. The cost function to be minimized is the sum of the operation cost and the investment cost. The long-term ORPP is often decomposed into a three-level hierarchical optimization problem using the maximum principle, the Bender's decomposition, and the P-Q decomposition methods [7]. To highlight the use of EAs however, this paper considers a short-term ORPP, where the investment is to be performed only once [10,11].

A. Objective Functions

The operation cost is often assumed to have single quadratic cost functions. In reality, the cost function has discontinuities corresponding to change of fuels [12]. Therefore, it is more appropriate to represent the cost function with piecewise quadratic functions. When using piecewise quadratic cost functions, the operation cost is defined as follows:

$$C_p = \sum_{i \in N_g} C_i(P_i) \quad (7)$$

$$C_i = \begin{cases} a_{i1} + b_{i1}P_i + c_{i1}P_i^2 & \text{if } \underline{P}_i \leq P_i < P_{i1} \\ a_{i2} + b_{i2}P_i + c_{i2}P_i^2 & \text{if } P_{i1} \leq P_i < P_{i2} \\ \dots & \dots \\ \dots & \dots \\ a_{im} + b_{im}P_i + c_{im}P_i^2 & \text{if } P_{i,m-1} \leq P_i < \bar{P}_i \end{cases}$$

where

N_g : the set of generators,

$C_i(P_i)$: cost of the i^{th} generator,

a_{ij}, b_{ij}, c_{ij} : cost coefficients of the i^{th} generator at the j^{th} power level,

P_i : the generated power of the i^{th} generator [MW],

$\underline{P}_i, \bar{P}_i$: minimum and maximum real power generation of the i^{th} generator.

Since EAs use the objective function directly, rather than its derivatives, more realistic cost functions can be used for the ORPP problem. Most papers consider the transmission loss in the objective function. However, the minimization of loss does not guarantee the minimization of the operation cost unless all units have the same efficiency. Therefore the fuel cost has been used for both real and reactive power dispatch [4-7].

The investment cost is simply the installation cost of the Var sources:

$$C_I = \sum_{i \in N_C} \{C_{fi} + C_{ci} [\bar{Q}_{ci}]\}, \quad (8)$$

where N_C is the set of compensators, C_{fi} and C_{ci} are, respectively, the fixed and the unit costs for investment, and \bar{Q}_{ci} is the amount of Var source investment in discrete steps. This cost function assumes the capacitive compensators; however, the reactive compensator can also be included by replacing \bar{Q}_{ci} with \underline{Q}_{ci} .

B. P-Q Decomposition

The ORPP problem is decomposed into two subproblems, the Q-optimization module, and the P-optimization module.

1) *The P-module:* In this module, the objective is to minimize the sum of the operation cost:

$$C_P = \sum_{i \in N_l} d_i C_P^i, \quad (9)$$

where C_P^i is the operation cost (6) for load level i , d_i is the duration of load level i , and N_l is the set of load levels. The minimization is with respect to the real power generations P_g for each load level subject to the real and reactive power balance of the power system which can be solved by calling the *load flow* program, and the following inequality constraints:

$$\begin{aligned} \underline{P}_{gi} &\leq P_{gi} \leq \bar{P}_{gi} \\ \underline{Q}_{gi} &\leq Q_{gi} \leq \bar{Q}_{gi} \\ \underline{T}_k &\leq T_k \leq \bar{T}_k \\ \underline{V}_i &\leq V_i \leq \bar{V}_i \\ \underline{Q}_{ci} &\leq Q_{ci} \leq \bar{Q}_{ci} \end{aligned} \quad (10)$$

where P_g and Q_g are generator real and reactive powers, respectively, T are transformer tap-settings, V are bus voltages, and Q_c are the reactive power output of compensators.

The real power generations P_g are the optimization or decision variables for the P-module, which are self-constrained. Since the generator bus voltages, transformer tap-settings and capacitor investments are the optimization

variables in the Q-module, they are fixed in the P-module, and thus the constraints are automatically satisfied. The load bus voltages and the generator reactive powers are state variables, which can be constrained by augmenting them as the quadratic penalty terms to the objective function. The P-optimization module is therefore changed to minimize the following generalized objective function:

$$C_P = \sum_{i \in N_l} d_i \left\{ C_P^i + \sum_{i \in N_l} \lambda_{vi} (V_i - \text{Sat}(V_i))^2 + \sum_{i \in N_g} \lambda_{gi} (Q_{gi} - \text{Sat}(Q_{gi}))^2 \right\}, \quad (11)$$

where N_l is the set of load buses and λ_{vi} and λ_{gi} are the penalty weights and $\text{Sat}(x)$ is the saturation function defined by

$$\text{Sat}(x) = \begin{cases} \underline{x} & \text{if } x < \underline{x} \\ x & \text{if } \underline{x} \leq x \leq \bar{x} \\ \bar{x} & \text{if } x > \bar{x} \end{cases} \quad (12)$$

2) *The Q-module:* In this module, the objective function is to minimize the sum of the operation cost and the investment cost,

$$C_Q = C_P + C_I, \quad (13)$$

with respect to transformer tap-settings T , generator bus voltages and the Var source investment \bar{Q}_c ; subject to the real and reactive power balance equations which can be solved by calling the *load flow* program, and the inequality constraints (10).

The transformer tap-settings, generator bus voltages and the Var source investment are the optimization or decision variables for the Q-module, which are self-constrained. Since the real power generations are obtained from the last P-module, they should satisfy the constraint. The voltages of load buses and the reactive powers of generators are state variables, which may be out of limits, and penalties are also added to the objective function. The formulation of the penalty parts are the same as the procedure in the P-module. Then the Q-module is to minimize the generalized cost function as follows:

$$C_Q = C_I + \sum_{i \in N_l} d_i \left\{ C_P^i + \sum_{i \in N_l} \lambda_{vi} (V_i - \text{Sat}(V_i))^2 + \sum_{i \in N_g} \lambda_{gi} (Q_{gi} - \text{Sat}(Q_{gi}))^2 \right\} \quad (14)$$

The P-Q decomposition is now complete. The optimization variables for the P-optimization subproblem are generator real power outputs, and those for the Q-optimization subproblem are generator voltage magnitudes, Var source investments, and transformer tap-settings. The

security constraints are the operating limits of these control variables, line flows, and the state variables.

The P- and Q- modules were solved sequentially to obtain local optimal values of the optimization variables for each module. The computation diagram of the ORPP problem is shown in Fig. 1.

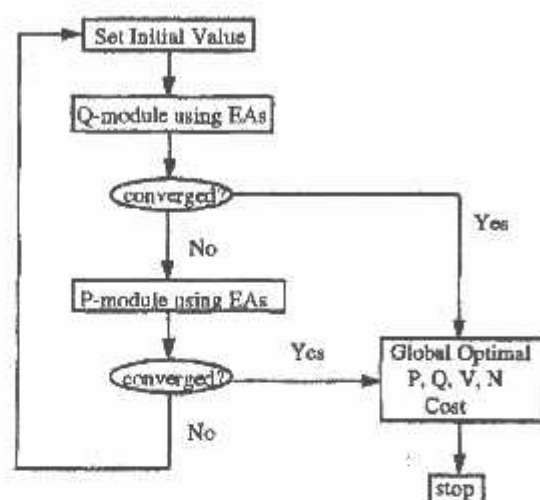


Fig. 1. Flow chart for ORPP by using EAs.

IV. SIMULATION RESULTS

A comparative study is performed for the three evolutionary algorithms (EAs) and the linear programming (LP) by solving the optimal reactive power planning (ORPP) problem for the IEEE 30-bus system as a testbed [4-7]. String representations and simulation parameters for EAs are discussed, and the modification of fuel costs with piecewise quadratic cost functions and the simulation results and the convergence speeds of EAs are given.

1) String representation is an important factor in solving the ORPP problem using the GA. In order to accommodate different decision variables, i.e., the investment or operation variables, the following representation method is used.

A string consists of substrings; the number of substrings is equal to the number of decision or control variables. In the P-optimization problem, it is the total number of generators and each substring represents the generator real power output. In the Q-optimization problem, on the other hand, the decision variables are generator voltage magnitudes, the maximum Var limits of compensators, and transformer tap settings. The simulation parameters for the three EAs are given in Table I and Table II.

TABLE I. SIMULATION PARAMETERS IN EP, ES, EP+ES AND GA

| method | FS | EP | EP+ES | GA |
|----------------------|------|----|------------------|-----|
| coefficients | | | | |
| number of generation | 50 | 60 | 100 | 100 |
| number of population | 80 | 40 | 50 | 50 |
| standard deviation | 0.03 | NA | EP(NA), ES(0.03) | NA |

NA: Not Applicable

TABLE II. SIMULATION PARAMETERS IN GA

| parameters | value | GA |
|-------------------------------------|-------|----------------------|
| mutation rate | | 0.01 |
| crossover rate | | 0.75 |
| abandoning rate | | 0.9 |
| maximum generations | | 100 |
| population size | | 50 |
| control variables | | 6 |
| parameter resolution | | 8 bits per substring |
| chromosome length | | 30 |
| fitness function parameter α | | 0.01 |

2) The fuel cost coefficients of the piecewise quadratic cost functions are given in Table III, which are generated from the original single quadratic cost functions [4]. Small units (at buses 5, 8, 11, and 13) are represented by two segments, while large units (at buses 1 and 3) are by three segments of fuel curves.

As shown in the P- and Q-optimization modules, the formulation is general to include a number of different load levels to optimally balance the savings in the operation cost against the cost of capital investment. However for simplicity, only the peak load condition [4] is considered for the comparative study. This scenario, obviously, will give an optimistic savings in the operation cost, and therefore could suggest over-investment.

TABLE III
COST COEFFICIENTS OF PIECEWISE QUADRATIC FUNCTION

| U | GENERATION | | | | | P | cost coefficients | | |
|---|------------|-----|-----|-----|---|-------|-------------------|---------|---|
| | Min | P1 | P2 | Max | F | | a | b | c |
| 1 | 50 | 100 | 190 | 200 | 1 | 0.000 | 1.900 | 0.00355 | |
| | | 1 | 2 | 3 | 2 | 0.000 | 2.000 | 0.00375 | |
| | | | | | 3 | 0.000 | 2.200 | 0.00415 | |
| 2 | 20 | 35 | 50 | 80 | 1 | 0.000 | 1.700 | 0.01700 | |
| | | 1 | 2 | 3 | 2 | 0.000 | 1.750 | 0.01750 | |
| | | | | | 3 | 0.000 | 2.050 | 0.02350 | |
| 3 | 15 | 30 | 50 | | 1 | 0.000 | 1.000 | 0.06250 | |
| | | 1 | 2 | | 2 | 0.000 | 1.200 | 0.08250 | |
| 4 | 10 | 25 | 35 | | 1 | 0.000 | 3.250 | 0.00834 | |
| | | 1 | 2 | | 2 | 0.000 | 3.650 | 0.01234 | |
| 5 | 10 | 20 | 30 | | 1 | 0.000 | 3.000 | 0.02500 | |
| | | 1 | 2 | | 2 | 0.000 | 3.300 | 0.03500 | |
| 6 | 12 | 24 | 40 | | 1 | 0.000 | 3.000 | 0.02500 | |
| | | 1 | 2 | | 2 | 0.000 | 3.300 | 0.03500 | |

3) The initial load flow, with generator voltages set to 1.0 p.u., showed that the voltage magnitudes at the load buses 14-30 were all below the lower operating limit of 0.95 p.u., and the operation cost was as high as 903.31 \$/hr. First for optimal power flow, the P-optimization module and the Q-optimization module without the investment were run by using both LP [6] and ES. Real power distribution, voltage magnitudes and transformer tap-settings were optimized to minimize the operation cost. The results of LP and ES showed that, all the control variables and state variables were within their hard limits, but for some load buses, such as buses 18-26, 29, and 30, the voltages were very close to their lower limits. Thus, more reactive power from other Var sources were needed to improve the voltage profile. As we compared the optimal power flow results, ES gave slightly higher voltage profile at slightly lower operation cost and power loss.

4) For the ORPP problem, the candidate buses for reactive power compensation are 15, 17, 20, 23, 24 and 29. All are shunt capacitors with the investment cost of 0.02 \$/Mvar per unit hour. The fixed cost for installation is neglected for simplicity. However, the site-dependent fixed cost can be incorporated in additional substrings in EAs; the number of additional substrings being equal to the number of candidate buses. The investments are made in discrete steps: 2.5, 5, 7.5, ..., 30 Mvars, which become the upper limits \bar{Q}_{ci} in the operational constraints (10). In both LP and EAs, when the constraints are violated in any iteration, additional steps of the investments are made in the candidate buses in the Q-optimization module and the iteration is repeated.

TABLE IV. OPTIMAL OPERATION WITH INVESTMENT
(A) REAL POWER AND REACTIVE POWER DISTRIBUTION

| Variable | GA | ES | EP | EP+ES |
|----------|----------|----------|----------|----------|
| P_1 | 155.002 | 156.777 | 158.216 | 159.826 |
| P_2 | 55.235 | 56.027 | 59.256 | 53.235 |
| P_3 | 49.857 | 49.941 | 49.693 | 50.031 |
| P_4 | 12.608 | 11.975 | 12.025 | 11.069 |
| P_{11} | 12.665 | 13.049 | 12.834 | 11.002 |
| P_{12} | 12.607 | 12.212 | 12.223 | 12.897 |
| Q_1 | 17.500 | -16.836 | -17.402 | 5.502 |
| Q_2 | 28.013 | 58.117 | 27.832 | 46.915 |
| Q_3 | -1.826 | 18.774 | 36.727 | 52.003 |
| Q_4 | 47.154 | 24.528 | 47.565 | 5.237 |
| Q_{11} | 4.001 | 23.921 | 14.415 | 25.795 |
| Q_{12} | 28.358 | 30.655 | 28.307 | -7.932 |
| Loss | 7.5+j3.2 | 7.5+j0.3 | 8.2+j0.5 | 8.5+j1.3 |

(B) BUS VOLTAGES

| Variable(p.u.) | GA | ES | EP | EP+ES |
|----------------|--------|-------|-------|-------|
| V_1 | 1.0903 | 1.084 | 1.084 | 1.083 |
| V_2 | 1.0615 | 1.064 | 1.063 | 1.062 |
| V_3 | 1.0508 | 1.050 | 1.050 | 1.049 |
| V_4 | 1.0425 | 1.043 | 1.042 | 1.042 |
| V_5 | 1.0289 | 1.032 | 1.030 | 1.030 |
| V_6 | 1.0340 | 1.037 | 1.036 | 1.035 |

| | | | | |
|----------|--------|-------|-------|-------|
| V_7 | 1.0234 | 1.027 | 1.025 | 1.025 |
| V_8 | 1.0294 | 1.038 | 1.033 | 1.033 |
| V_9 | 1.0006 | 1.028 | 1.019 | 1.022 |
| V_{10} | 1.0032 | 1.014 | 1.010 | 1.013 |
| V_{11} | 1.0584 | 1.055 | 1.056 | 1.056 |
| V_{12} | 1.0111 | 1.019 | 1.018 | 1.019 |
| V_{13} | 1.0380 | 1.032 | 1.035 | 1.035 |
| V_{14} | 1.0000 | 1.010 | 1.008 | 1.010 |
| V_{15} | 0.9992 | 1.011 | 1.009 | 1.011 |
| V_{16} | 1.0013 | 1.011 | 1.009 | 1.011 |
| V_{17} | 0.9988 | 1.011 | 1.007 | 1.010 |
| V_{18} | 0.9901 | 1.003 | 1.001 | 1.004 |
| V_{19} | 0.9879 | 1.002 | 1.000 | 1.003 |
| V_{20} | 0.9924 | 1.007 | 1.005 | 1.008 |
| V_{21} | 0.9958 | 1.006 | 1.002 | 1.005 |
| V_{22} | 0.9970 | 1.007 | 1.002 | 1.006 |
| V_{23} | 0.9981 | 1.010 | 1.006 | 1.009 |
| V_{24} | 0.9972 | 1.002 | 0.997 | 1.001 |
| V_{25} | 1.0114 | 1.006 | 0.996 | 1.002 |
| V_{26} | 0.9926 | 0.998 | 0.978 | 0.984 |
| V_{27} | 1.0289 | 1.018 | 1.005 | 1.012 |
| V_{28} | 1.0264 | 1.033 | 1.032 | 1.031 |
| V_{29} | 1.0183 | 1.014 | 1.002 | 1.009 |
| V_{30} | 1.0030 | 0.995 | 0.983 | 0.990 |

(C) TAP SETTINGS, CAPACITIVE VARS, AND COSTS

| Variable | GA | ES | EP | EP+ES |
|--------------------|--------|---------|--------|--------|
| N_{11} | 1.0155 | 1.0290 | 1.0223 | 1.0165 |
| N_{12} | 0.9940 | 1.0337 | 1.0199 | 1.0261 |
| N_{13} | 1.0320 | 1.0298 | 1.0286 | 1.0277 |
| N_{24} | 0.9930 | 1.0274 | 1.0176 | 1.0192 |
| Q_{c15} (Mvar) | 3.05 | 6.5218 | 4.7253 | 4.7541 |
| Q_{c17} (Mvar) | 2.89 | 6.4387 | 4.4545 | 4.4418 |
| Q_{c20} (Mvar) | 2.79 | 6.8332 | 5.0405 | 5.0644 |
| Q_{c21} (Mvar) | 4.73 | 6.6059 | 4.7453 | 4.7540 |
| Q_{c23} (Mvar) | 2.96 | 6.4141 | 4.4747 | 4.4762 |
| Q_{c24} (Mvar) | 7.47 | 6.3411 | 4.2969 | 4.2865 |
| Q_{c29} (Mvar) | 3.09 | 6.6167 | 4.7567 | 4.7358 |
| Gen. cost (\$/hr) | 706.87 | 706.125 | 706.58 | 706.41 |
| Inv. cost (\$/hr) | 0.75 | 1.05 | 0.75 | 0.75 |
| Total cost (\$/hr) | 707.62 | 707.175 | 707.33 | 707.16 |

5) The results of the ORPP by using LP and EAs are given in Table IV. The results show that all the control variables and state variables, such as voltage magnitudes, transformer tap-settings, real power and reactive power generations are within their hard limits. The results of ES, EP, and EP+ES are very close and comparable with those of LP, and the average differences of voltage magnitudes and real power generations are within 5/1000. The power loss and the operation cost of LP is slightly higher than those of EAs. It is noted that ES showed the least operation cost at the expense of the highest investment cost; while the combined EP and ES (EP+ES) showed the least total cost (sum of the generation and the investment costs). Table IV(C) showed that two of the four tap-settings are reversed in direction in LP; while they all are in the same direction in all EAs. This resulted in the fairly even investment of Var sources in all seven candidate buses for EAs, while it is not the case for LP. It should be noted that the capacitive Vars, Q_{ci} , shown in the table are the operational values after the

investment, which are in general less than the maximum limits, \bar{Q}_{ci} , given by the investment in discrete steps.

The total cost of each P- and Q-modules in each iteration of ORPP are shown in the Table V. The iteration process is alternating as Q-P-Q-P-, etc. It is shown in the table that only 4 or 5 iterations of P- and Q-modules are required for convergence.

TABLE V. TOTAL COST DURING ITERATIONS OF ES FOR ORPP

| Iter | 1 | 2 | 3 | 4 | 5 |
|------|--------|--------|--------|--------|--------|
| Q | 866.07 | 707.12 | 707.00 | 707.19 | 706.90 |
| P | 707.56 | 706.95 | 707.30 | 706.90 | 706.93 |

Each P- or Q-module converges in at least 30 generations. The CPU time of VAX for each generation is about 1.85 seconds, and in the ORPP problem, the CPU time for EAs to reach the global minimum is therefore at least $1.85 \times 30 \times 5$ seconds. However, for LP, it needs about 13 iterations to reach a local minimum, and the total CPU time is only around 7 seconds in VAX. Although the EAs take longer than LP, the formulation is straight forward and can handle arbitrary cost functions which may not be convex; and moreover, they can find the global minimum, while LP can only find a local minimum.

6) Within the EA family, the characteristics of ES, EP and GA were compared. For fair comparison, the same conditions are set for all decision variables of the Q-module, and only the P-module was run. Sample results are given in Table VI. The optimal real power distribution and the operation cost are very close for all methods. However, the number of generations to converge are different for different methods, and also different for different runs in the same method. Thus these programs were run in batch file for 5 times, and the average of the best fitness in each generation as calculated, which are shown in Fig. 2 and Fig. 3. In the P+ES method, at first the EP process is called, and the best fitness of these generations decreases rapidly; then after 10 iterations, ES process is called to continue the calculation. It is shown that, the ES and ES+EP methods need nearly the same number of generations (on average) to converge, while the EP and GA methods need almost twice as much time to converge.

TABLE VI. COMPARISON OF EA'S FOR P-OPTIMIZATION

| Method | EP | ES | EP+ES | GA |
|-------------------------|--------|--------|--------|--------|
| P1[MW] | 157.92 | 155.93 | 158.13 | 154.92 |
| P2[MW] | 60.587 | 57.887 | 54.664 | 56.891 |
| P3[MW] | 48.975 | 48.996 | 49.996 | 48.879 |
| P4[MW] | 19.467 | 20.467 | 18.775 | 17.086 |
| P5[MW] | 13.613 | 14.335 | 12.703 | 13.703 |
| P6[MW] | 11.745 | 11.776 | 11.753 | 11.328 |
| Loss[MW] | 8.995 | 9.013 | 8.985 | 8.944 |
| Total cost | 708.87 | 708.86 | 708.87 | 708.64 |
| Generations to converge | 60 | 37 | 36 | 63 |

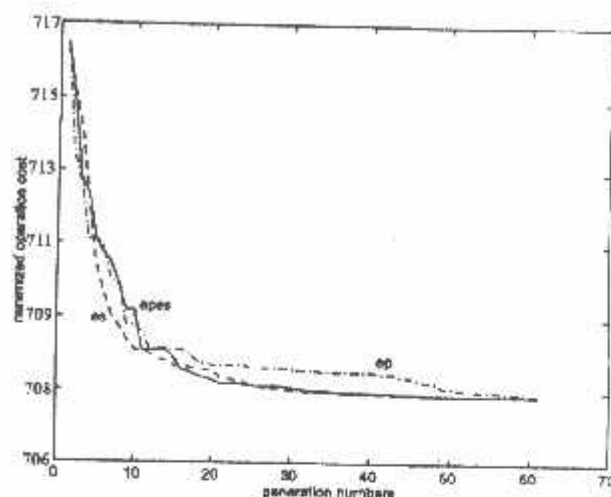


Fig. 2. Convergence comparison for EAs in the P-module

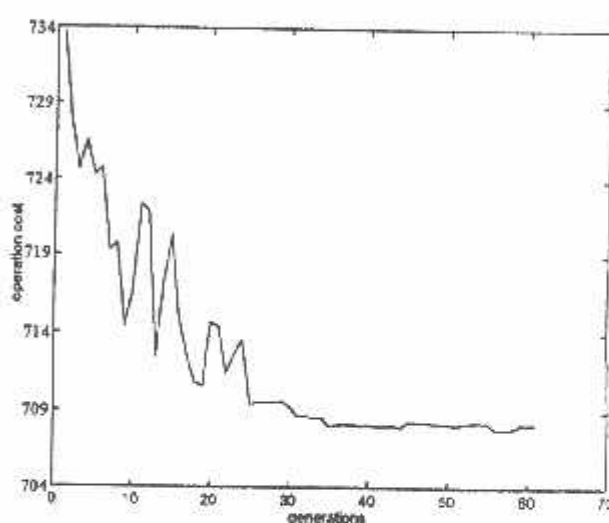


Fig. 3. Convergence characteristics of GA in the P-module

V. CONCLUSIONS

In this paper, the ORPP problem was solved by minimizing the total cost which includes the operation cost and the investment cost. The IEEE 30-bus system with piecewise quadratic cost functions is selected as a testbed. The results of ORPP by using different EA methods are almost identical. When the results are compared with the LP's, the EAs seem to be better; the total cost and the power loss are slightly lower, while all the hard limits are satisfied. Moreover, the P- and Q-modules of EAs can be easily formulated for general piecewise cost functions, not necessarily convex, while for LP, it is quite difficult. The

characteristics of EAs are also compared. The ES needs less generations to converge in either P- or Q-modules, but it has a higher probability to fall into a local minimum. The EP needs more generations to converge, however, it is less likely to fall into a local minimum. When the EP is combined with ES, it only needs nearly the same number of generations to converge as ES, but with the improved robustness in finding the global minimum.

VI. ACKNOWLEDGMENT

This work is supported in part by the Allengeny Power Company under grant "Optimal Reactive Power Planning" and the National Science Foundation under grant "Research and Curriculum Development for Power Plant Intelligent Distributed Control" (EID-9212132).

VII. REFERENCES

- [1] R. Billington and S. S. Sachdev, "Optimum network VAR planning by nonlinear programming", *IEEE Trans. on Power Appar. and Syst.*, Vol. PAS-92, pp. 1217-1225, July/August 1973.
- [2] G. T. Heydt and W. M. Grady, "Optimal Var siting using linear load flow formulation", *IEEE Trans. on Power Appar. and Syst.*, pp. 1214-1222, Vol. PAS-102, No. 5, May 1983.
- [3] K. Aoki, M. Fan, and A. Nishikori, "Optimal var planning by approximation method for recursive mixed integer linear planning", *IEEE Trans. on Power Syst.*, Vol. PWR-3, No. 4, pp. 1741-1747, November 1988.
- [4] K. Y. Lee, Y. M. Park, and J. L. Ortiz, "A united approach to optimal real and reactive power dispatch", *IEEE Trans. on Power Appar. and Syst.*, Vol. PAS-104, pp. 1147-1153, May 1985.
- [5] K. Y. Lee, J. L. Ortiz, Y. M. Park, and L. G. Pond, "An optimization technique for reactive power planning of subtransmission network under normal operation", *IEEE Trans. on Power Syst.*, Vol. PWR-1, pp. 153-159, May 1986.
- [6] M. K. Mangoli, K. Y. Lee, and Y. M. Park, "Operational real and reactive power control using linear programming", *Electric Power Systems Research*, Vol. 26, pp. 1-10, 1993.
- [7] M. K. Mangoli, K. Y. Lee, and Y. M. Park, "Optimal long-term reactive power planning using decomposition techniques", *Electric Power System Research*, Vol. 26, pp. 41-52, 1993.
- [8] D. E. Goldberg, "Genetic algorithms in search, optimization, and machine learning", Addison-Wesley, 1989.
- [9] D. B. Fogal, "An introduction to simulated evolutionary optimization", *IEEE Trans. on Neural Network*, Vol. 5, No. 1, pp. 3-14, January 1994.
- [10] K. Y. Lee, X. Bai, and Y. M. Park, "Optimization method for reactive power planning by using a modified simple genetic algorithm", *IEEE Trans. on Power Systems*, Vol. 10, No. 4, pp. 1843-1850, November 1995.
- [11] L. L. Lai and J. T. Ma, "Application of evolutionary programming to reactive power planning-comparison with nonlinear programming approach", *IEEE/PES Winter Meeting, #96 WM 248-5 PWR*, January 1996.
- [12] J. H. Park, S. O. Yang, H. S. Lee, and Y. M. Park, "Economic load dispatch using evolutionary algorithm", *Proc. of International Conference on Intelligent Systems Applications to Power Systems, Orlando, FL*, pp. 441-445, January 1996.
- [13] R. Dimeo and K. Y. Lee, "Boiler-turbine control system design using a genetic algorithm", *IEEE Trans. on Energy Conversion*, Vol. 10, pp. 752-759, December 1995.
- [14] V. Miranda, J. V. Ranito, and L. M. Proenica, "Genetic algorithm in optimal multistage distribution network planning", *IEEE PES Winter Meeting, #94 WM 229-5-PWR*.
- [15] K. Iba, "Reactive power optimization by genetic algorithm", *IEEE Trans. on Power Systems*, Vol. 9, No. 2, pp. 685-692, May 1994.

Kwang Y. Lee was born in Pusan, Korea on March 6, 1942. He received the B. S. degree in electrical engineering from Seoul National University, Seoul, Korea, in 1964, the M.S. degree in electrical engineering from North Dakota State University, Fargo, ND, in 1967, and the Ph.D. degree in systems science from Michigan State University, East Lansing, MI, in 1971. He has been on the faculties of Michigan State, Oregon State, University of Houston, and the Pennsylvania State University, University Park, PA, where he is currently a Professor of Electrical Engineering. His current research interests include control theory, intelligence systems, and their applications to power systems control, operation and planning.

Dr. Lee is currently the Director of Power Systems Control Laboratory and co-director of Intelligent Distributed Controls Research Laboratory at Penn State. He is a Senior Member of IEEE.

Frank F. Yang was born in Taixing, China on March 13, 1969. He received the B. S. degree in electrical engineering from Tsinghua University, Beijing, China, in 1992, the M.S. degree in electrical engineering from Pennsylvania State University, PA, in 1996.

Mr. Yang's interests include control theory, power systems, computer architecture, computer network, and computational intelligence.