

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



**OPTIMISASI *MULTI-OBJECTIVE VOLT-VAR* MENGGUNAKAN
METODE *PARETO FRONT - GA* PADA SALURAN TRANSMISI
150 KV SUB SISTEM PAITON-BALI**

SKRIPSI

Disusun Oleh :

PENDHI PURWO PRASETYO

01.12.125

SEPTEMBER 2006

REPUBLICAN PARTY NATIONAL COMMITTEE
1100 PENNSYLVANIA AVENUE, N.W.
WASHINGTON, D.C. 20004
TEL: (202) 638-1234

REPUBLICAN PARTY NATIONAL COMMITTEE
1100 PENNSYLVANIA AVENUE, N.W.
WASHINGTON, D.C. 20004
TEL: (202) 638-1234

REPUBLICAN PARTY

1100 PENNSYLVANIA AVENUE, N.W.

WASHINGTON, D.C. 20004

TEL: (202) 638-1234

REPUBLICAN PARTY

LEMBAR PERSETUJUAN

**OPTIMISASI *MULTI – OBJECTIVE VOLT VAR*
MENGUNAKAN METODE *PARETO FRONT – GA*
PADA SALURAN TRANSMISI 150 kV SUB SISTEM
PAITON - BALI**

SKRIPSI

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

Disusun Oleh :

PENDHI PURWO PRASETYO

01.12.125

Mengetahui,

Ketua Jurusan Teknik Elektro

Diperiksa dan disetujui,

Dosen Pembimbing



Ir. F. YUDI LIMPRAPTONO, MT

NIP.Y. 1039500274



Ir. H CHOIRI

NIP.131703042

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

ABSTRAKSI

OPTIMISASI *MULTI – OBJECTIVE VOLT VAR* MENGGUNAKAN METODE *PARETO FRONT – GA* PADA SALURAN TRANSMISI 150 KV SUB SISTEM PAITON BALI

(Pendhi Purwo P, 01.12.125, Teknik Elektro Energi Listrik S – 1, ITN 2006)
(Dosen Pembimbing : Ir. H Choiri)

Kata Kunci : *Sistem Transmisi, Multi – Objective Volt Var, Penempatan Kapasitor, Pengurangan Rugi – rugi Daya, Pareto Front, Genetika Algorithm.*

Perlunya optimisasi pada sumber daya reaktif untuk saluran transmisi telah lama diakui. Jika sumber daya investasi terbatas, bermacam-macam solusi (level tegangan, jumlah suplay daya reaktif dan seterusnya) dapat berdampak pada jumlah sasaran (rugi-rugi transmisi, factor daya, profile tegangan dan sebagainya). Seiring dengan meningkatnya beban –beban induktif, maka daya reaktif yang ada pada jaringan akan semakin besar yang selanjutnya akan memperbesar komponen rugi – rugi daya, disamping itu dapat memperburuk kondisi tegangan. Untuk mereduksi rugi daya solusinya dengan pemasangan kapasitor.

Dalam skripsi yang saya ajukan ini merupakan salah satu solusi dalam penempatan kapasitor yang lebih optimal dari kondisi sebelumnya yaitu menggunakan metode *Pareto Front – Genetika Algorithm*. Serta digunakan perhitungan aliran daya *Newton Raphson* untuk mengetahui perubahan hasil dari pemasangan kapasitor.

Dengan menggunakan data jaringan pada saluran transmisi 150 kV sub system Paiton – Bali, di peroleh hasil pemasangan kapasitor yaitu pada bus #5 dengan kapasitas 40 MVar, bus #6 dengan kapasitas 34 MVar, bus #14 dengan kapasitas 15 MVar, dan bus #23 dengan kapasitas 48Mvar dan diperoleh pengurangan rugi daya sebesar $2,784 + j 4,245$ MVA. Biaya instalasi pemasangan kapasitor sebesar 1.370.000 US \$.

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadiran Allah SWT yang telah memberikan rahmat dan hidayahNya, sehingga penulis dapat menyelesaikan skripsi ini dengan sebaik-baiknya.

Skripsi yang berjudul **“OPTIMISASI *MULTI-OBJECTIVE VOLT-VAR* MENGGUNAKAN METODE *PARETO FRONT - GA* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON-BALI “** tersusun atas bantuan berbagai pihak. Untuk itu penulis menghaturkan terima kasih yang mendalam kepada:

1. Bapak DR. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME, selaku Dekan FTI ITN Malang.
3. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro S-1 Institut Teknologi Nasional Malang.
4. Bapak Ir. H Choiri, selaku Dosen Pembimbing yang telah memberi motifasi didalam penyelesaian skripsi ini.
5. Bapak Ir.M. Abdul Hamid, MT dan Bapak Bambang Prio H. ST, MT, selaku dosen penguji ujian skripsi.
6. Bapak Saptono, Ibu Endang , Ibu Sri dan seluruh staf PT. PLN (Persero) P3B Jawa-Bali atas kerja samanya.
7. Keluarga Besar penulis, terutama Bapak dan Ibu, terima kasih atas semua ketulusan yang diberikan.

8. Rekan Mahasiswa Teknik Elektro Energi Listrik yang telah membantu atas terselesaikannya skripsi ini.

Dengan bantuan dari semua pihak, penulis berusaha dengan sebaik-baiknya, namun penulis menyadari bahwa kekurangan akan selalu ada dan masih jauh dari kesempurnaan. Maka penulis sangat membutuhkan saran dan kritik dari semua pihak.

Akhir kata, semoga Skripsi bermanfaat bagi penulis sendiri khususnya dan para pembaca pada umumnya.

Malang, September 2006

Penulis

DAFTAR ISI

Halaman :

HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	xi
DAFTAR GRAFIK	xiii
DAFTAR TABEL	xiv
BAB I PENDAHULUAN	1
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan.....	3
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian	3
1.6. Sistematika Pembahasan	4
1.7. Kontribusi Penelitian.....	5
BAB II SISTEM TENAGA LISTRIK	6
2.1. Sistem Tenaga Listrik.....	6
2.2. Saluran Transmisi	7
2.2.1. Saluran Transmisi Pendek	8

2.2.2. Saluran Transmisi Menengah.....	8
2.2.3. Saluran Transmisi Panjang.....	10
2.3. Daya Dalam Sistem Tenaga Listrik.....	11
2.3.1. Daya Aktif (<i>Active Power</i>).....	11
2.3.2. Daya Reaktif (<i>Reactive Power</i>).....	11
2.3.3. Daya Semu (<i>Apparent Power</i>).....	12
2.4. Sistem Per- Unit.....	13
2.5. Mengubah Dasar Sistem Per-Unit.....	14
2.6. Kapasitor Shunt.....	15
2.7. Pengurangan Rugi-rugi Daya dengan Kapasitor Shunt.....	16
2.8. Perbaikan Tegangan.....	18
2.9. Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem.....	18
2.10. Perhitungan Pengaruh Perbaikan Faktor Daya.....	21
2.11. Penentuan Rating Kapasitor untuk Perbaikan Faktor Daya Beban.....	22
BAB III OPTIMASI <i>MULTI</i> – <i>OBJECTIVE</i> VOLT VAR	
MENGGUNAKAN METODE <i>PARETO FRONT</i> – <i>GA</i>	23
3.1. Analisa Aliran Daya.....	23
3.1.1. Klasifikasi Bus.....	23
3.1.2. Metode Newton Raphson.....	25
3.2. Genetika Algoritma.....	26
3.2.1. Istilah-istilah dalam <i>Genetic Algorithm</i>	29
3.2.2. Parameter <i>Genetic Algorithm</i>	29
3.2.3. Proses <i>Genetic Algorithm</i>	32

3.2.4. Elitism.....	35
3.2.5. Crossover (Pindah Silang).....	36
3.2.6. Mutation (Mutasi).....	38
3.3. Konsep Optimisasi <i>Multi-Objective</i> Volt Var Menggunakan Metode <i>Pareto Front – GA</i>	39
BAB IV OPTIMISASI <i>MULTI – OBJECTIVE VOLT VAR</i> MENGUNAKAN <i>PARETO FRONT - GA</i> PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON – BALI	42
4.1. Program Komputer Optimisasi <i>Multi-Objective</i> Volt Var menggunakan <i>Pareto Front-GA</i> pada saluran transmisi 150 kv sub sistem Paiton – Bali.....	42
4.2. Algoritma Program.....	42
4.2.1. Algoritma Aliran Daya Newton Raphson.....	42
4.2.2. Algoritma Pemecahan Masalah.....	43
4.2.3. Algoritma Program Optimisasi <i>Multi-Objective Volt Var</i> Menggunakan Metode <i>Pareto Front – GA</i>	44
4.2.4. Algoritma Program Fitness (<i>Pareto Front</i>).....	45
4.3. Flowchart Program.....	46
4.3.1. Flowchart Aliran Daya Newton Raphson.....	46
4.3.2. Flowchart Algoritma Dasar.....	47
4.3.2.1.Flowchart Algoritma Pemecahan Masalah.....	47
4.3.2.2.Flowchart Algoritma Program Optimisasi <i>Multi-Objective</i> <i>Volt Var</i> Menggunakan Metode <i>Pareto Front – GA</i>	48
4.3.2.3.Flowchart Algoritma Program Fitness (<i>Pareto Front</i>).....	49

4.4.	Sistem Transmisi 150 kV Sub Sistem Paiton-Bali	50
4.4.1.	Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton-Bali	53
4.4.2.	Data Saluran Transmisi 150 kV Sub Sistem Paiton-Bali	54
4.5.	Prosedur Pelaksanaan Program Perhitungan	56
4.6.	Hasil Perhitungan Optimasi <i>Multi-Objective Volt Var</i> Menggunakan Metode Pareto Front – GA Pada Saluran Transmisi 150 kV Sub Sistem Paiton dan Bali	63
4.6.1.	Hasil Perhitungan Kondisi Awal	63
4.6.2.	Hasil Perhitungan Setelah Optimasi	66
4.7.	Hasil Analisa Penempatan dan Kapasitas Kapasitor saerta Biaya Instalasi Kapasitor Menggunakan Metode <i>Pareto Front – GA</i>	69
4.7.1.	Penempatan Kapasitor dan Kapasitas Kapasitor	69
4.7.2.	Biaya Instalasi Kapasitor	69
4.8.	Analisa Hasil Perhitungan Kondisi Awal dan Kondisi Akhir Optimisasi <i>Multi-Objective Volt Var</i> Menggunakan Metode <i>Pareto Front – GA</i>	70
4.8.1.	Perbandinfgan Total Daya Aktif Pembangkitan, Pembebanan dan Rugi-rugi Saluran	70
4.8.2.	Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus	71
4.8.3.	Perbandingan Aliran Daya Tiap Bus	72

4.8.4. Perbandingan Tingkat Rugi Daya Aktif dan Daya Reaktif Pada Saluran Transmisi.....	74
BAB V KESIMPULAN	75
5.1. Kesimpulan.....	75
5.2. Saran – saran.....	76
DAFTAR PUSTAKA	77
LAMPIRAN	

DAFTAR GAMBAR

	Halaman:
Gambar 2.1. Rangkaian Ekvivalen 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. Saluran Primer Dengan Beban Terpusat	16
Gambar 2.7. Vektor Diagram Sebelum Dan Sesudah Pemasangan Kapasitor Shunt Pada Jaringan	18
Gambar 2.8. Perbaikan Faktor Daya	19
Gambar 2.9. Diagram Fasor dan Sudut Faktor Daya	21
Gambar 3.1. Pengkodean Representasi	32
Gambar 3.2. <i>Roulette – Wheel</i>	34
Gambar 3.3. Pembentukan Nex Generation Dalam Algoritma Genetika	36
Gambar 3.4. Ilustrasi Operator Dengan <i>One Point Crossover</i>	37
Gambar 3.5. Ilustrasi Operator Dengan <i>Two Point Crossover</i>	37
Gambar 3.6. Ilustasi Operator Dengan <i>Uniform Crossover</i>	37
Gambar 3.7. Ilustasi Operator Mutasi Untuk Representasi Biner.....	38
Gambar 3.8. Ilustasi Operator Mutasi Untuk Representasi Integer	39
Gambar 3.9 Optimisasi Pareto, Solusi Non Dominan dan Dominan.....	40
Gambar 4.1. <i>Flowchart</i> Aliran Daya <i>Newton Raphson</i>	46

Gambar 4.2.	<i>Flowchart</i> Algoritma Pemecahan masalah.....	47
Gambar 4.3.	<i>Flowchart</i> Algoritma Pemecahan Multi – Objective Volt Var Dengan Algoritma Genetika.....	48
Gambar 4.4.	<i>Flowchart</i> Algoritma Program Fitness (<i>Pareto Front</i>)	49
Gambar 4.5.	Diagram Segaris Jaringan Sistem Transmisi 150 Kv Paiton – Bali .	51
Gambar 4.6.	Menu Utama Program	56
Gambar 4.7.	Tampilan Data	56
Gambar 4.8.	Tampilan Data Bus.....	57
Gambar 4.9.	Tampilan Data Saluran	57
Gambar 4.10.	Tampilan Data Generator	58
Gambar 4.11.	Tampilan <i>Loadflow</i> Awal.....	58
Gambar 4.12.	Tampilan Aliran Daya Hasil Perhitungan <i>Loadflow</i> Awal	59
Gambar 4.13.	Tampilan <i>Summary</i> Pada Kondisi Awal	59
Gambar 4.14.	Tampilan Parameter <i>Genetica Algorithm</i>	60
Gambar 4.15.	Tampilan <i>Loadflow</i> Akhir	60
Gambar 4.16.	Tampilan Aliran Daya Hasil Perhitungan <i>Loadflow</i> Akhir.....	61
Gambar 4.17.	Tampilan <i>Summary</i> Pada Kondisi Akhir.....	61
Gambar 4.18.	Tampilan Perbandingan Tegangan Tiap Bus	62
Gambar 4.19.	Tampilan Hasil Penempatan Kapasitor	62

DAFTAR GRAFIK

Halaman:

Grafik 4.1. Hasil Perhitungan Daya yang Dibangkitkan Tiap Unit Pembangkit Pada Kondisi Awal	65
Grafik 4.2. Hasil Perhitungan Daya yang Dibangkitkan Tiap Unit Pembangkit Pada Kondisi Awal	68
Grafik 4.3. Perbandingan Total Daya Aktif Pembangkitan, Pembebanan dan Rugi-rugi saluran Pada Kondisi Awal dan Akhir	70
Grafik 4.4. Perbandingan Profil Tegangan	72
Grafik 4.5. Perbandingan Rugi Daya	74

DAFTAR TABEL

Halaman:

Tabel 4.1.	Penomoran Bus 150 kV Sub Sistem Paiton - Bali	52
Tabel 4.2.	Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton – Bali	53
Tabel 4.3.	Data Saluran Transmisi 150 kV Sub Sistem Paiton – Bali	55
Tabel 4.4.	Hasil Perhitungan Tegangan, Sudut Teganagan, Pembangkitan dan Pembebanan Pada Kondisi Awal.....	63
Tabel 4.5.	Hasil Perhitungan Aliran Daya Antar Saluran Pada Kondisi Awal ...	64
Tabel 4.6.	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Awal.....	65
Tabel 4.7.	Hasil Perhitungan Daya Yang Dibangkitkan Pada Kondisi Awal	65
Tabel 4.8.	Hasil Perhitungan Tegangan, Sudut Teganagan, Pembangkitan dan Pembebanan Pada Kondisi Akhir.....	66
Tabel 4.9.	Hasil Perhitungan Aliran Daya Antar Saluran Pada Kondisi Akhir ..	67
Tabel 4.10.	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Akhir	68
Tabel 4.11.	Hasil Perhitungan Daya Yang Dibangkitkan Pada Kondisi Akhir ...	68
Tabel 4.12.	Penempatan dan Kapasitas Kapasitor	69
Tabel 4.13.	Perbandingan Total Daya Aktif, Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Awal dan Kondisi Akhir	70
Tabel 4.14.	Perbandingan Tingkat Tegangan dan Sudut Teganagan Pada Tiap Bus.....	71
Tabel 4.15.	Perbandingan Aliran Daya Pada TiapBus.....	72

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dengan meningkatnya usaha di sektor industri dan meningkatnya taraf hidup masyarakat maka kebutuhan akan energi listrik semakin meningkat pula, sehingga diperlukan penyediaan energi listrik yang baik dan keandalan penyaluran tegangan. Bertambahnya industri – industri menyebabkan peran penggunaan alat – alat listrik akan semakin luas, misalnya motor – motor listrik, trafo, AC, lampu TL, dan lain – lain. Beban industri sangat membutuhkan daya reaktif induktif. Dengan meningkatnya beban – beban induktif, maka daya reaktif yang ada di jaringan transmisi akan semakin besar, sehingga memperbesar komponen rugi – rugi daya dan dapat memperburuk profil tegangan pada jaringan transmisi.

Alokasi daya reaktif yang tepat dalam sistem bisa menghasilkan kondisi optimum, yaitu kondisi dengan rugi-rugi transmisi yang minimum dengan biaya investasi yang murah dan mampu memperbaiki profil tegangan sistem. Permasalahan besarnya rugi-rugi daya di saluran menjadi perhatian utama khususnya pada saluran transmisi 150 KV.

Permasalahan tersebut telah menjadi perhatian utama khususnya pada saluran transmisi. Alternatif yang sering dipakai untuk memperbaiki kondisi jaringan adalah dengan pemasangan kapasitor.

Pada skripsi ini digunakan metode *Pereto Front-GA* , yang dapat menjadi salah satu alternatif untuk penempatan kapasitor (VAR) pada saluran transmisi 150 kV Paiton – Bali. Skripsi ini akan membahas penentuan letak dan kapasitas

dari kapasitor berdasarkan dua fungsi sasaran yang ingin dicapai yaitu keuntungan dari sisi teknis dan ekonomis, sehingga dihasilkan kualitas sistem tenaga listrik yang baik.

1.2. Rumusan Masalah

Berdasarkan latar belakang di atas, maka dapat dirumuskan permasalahan yaitu :

- Dimana penempatan dan kapasitas kapasitor serta biaya instalasi pemasangan kapasitor pada saluran transmisi 150 kV menggunakan metode *Pareto Front-GA*.
- Berapa profil tegangan pada setiap bus setelah dilakukan pemasangan kapasitor.
- Berapa rugi-rugi daya pada saluran transmisi 150 kV setelah penempatan kapasitor menggunakan metode *Pareto Front-GA*.

Berdasarkan pada deskripsi permasalahan dan latar belakang tersebut diatas maka skripsi ini diberi judul :

**OPTIMISASI *MULTI-OBJECTIVE VOLT-VAR* MENGGUNAKAN
METODE *PARETO FRONT - GA* PADA SALURAN TRANSMISI
150 KV SUB SISTEM PAITON-BALI**

1.3. Tujuan

Berdasarkan permasalahan yang telah dikemukakan diatas maka skripsi ini bertujuan untuk :

- Menganalisis penempatan dan kapasitas kapasitor, biaya instalasi pemasangan kapasitor, profil tegangan serta rugi daya setelah penempatan kapasitor pada saluran transmisi 150 kV menggunakan metode *Pareto Front-GA*.

1.4. Batasan Masalah

Dalam skripsi ini akan dilakukan analisis mengenai penempatan dan kapasitas kapasitor menggunakan metode *Pareto Front-GA* pada saluran 150 kV sub sistem Paiton-Bali. Agar pembahasan mengarah sesuai tujuan, maka pembahasan dalam proposal skripsi ini dibatasi oleh hal sebagai berikut :

- Analisis perhitungan penempatan kapasitor dan kapasitasnya pada saluran 150 kV sub sistem Paiton-Bali menggunakan metode *Pareto Front-GA*.
- Tidak membahas analisa aliran daya *Newton – Raphson* secara mendetail.
- Tidak membahas kontrol dan proteksi yang digunakan pada kapasitor tersebut.
- Tidak membahas penyebab gangguan.

1.5. Metodologi Penelitian

Metode yang digunakan dalam pembahasan proposal ini dilakukan dengan langkah – langkah sebagai berikut :

- **Studi Literatur**

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

- Pengumpulan data lapangan yang dipakai dalam objek penelitian yakni data impedansi saluran transmisi, data pembebanan, data tap kapasitor serta data kapasitas kapasitor yang akan digunakan pada saluran 150 kV sub sistem Paiton - Bali.
- Analisa perhitungan tegangan, sudut fasa tegangan, aliran daya pada tiap saluran, biaya pembangkitan dan biaya instalasi dengan menggunakan metode *Pareto Front-GA* yang disimulasikan dengan komputer dengan bahasa pemrograman *Borland Delphi* versi 7.0.
- Pengambilan kesimpulan dari hasil analisis.

1.6. Sistematika Pembahasan

Penulisan laporan skripsi ini terdiri atas beberapa bab dan sub bab yang tersusun dengan sistematika sebagai berikut :

BAB I PENDAHULUAN

Berisikan masalah umum mengenai latar belakang penulisan, rumusan masalah, tujuan penulisan skripsi, pembatasan masalah, metodologi pembahasan, sistematika penulisan dan kontribusi penelitian.

BAB II TINJAUAN PUSTAKA

Berisi antara lain tentang pengenalan sistem tenaga listrik, kapasitor shunt, perbaikan tegangan, perbaikan faktor daya dan kenaikan kapasitas sistem, pengaruh perbaikan faktor daya .

BAB III OPTIMASI *MULTI – OBJECTIVE* VOLT Var MENGGUNAKAN METODE *PARETO FRONT - GA*

Berisi tentang aliran daya, aplikasi metode *Pareto Front-GA*.

BAB IV OPTIMISASI *MULTI – OBJECTIVE* VOLT Var MENGGUNAKAN *PARETO FRONT-GA* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON – BALI.

Berisi tentang Program Komputer, Algoritma program dan hasil perhitungan serta analisa data dengan metode *Pareto Front-GA*.

BAB V PENUTUP

Berisi tentang kesimpulan akhir dan saran.

1.7. Kontribusi Penelitian

Kontribusi dari skripsi ini adalah penggunaan dari metode *Pareto Front-GA* dalam menganalisa penempatan dan kapasitas kapasitor yang digunakan pada saluran transmisi 150 kV sub sistem Paiton – Bali sehingga dapat menekan rugi-rugi daya dalam sistem, dan dapat diterapkan oleh pemerintah khususnya PT. PLN (Persero), sehingga dapat diterapkan bagi PLN sebagai perusahaan penyediaan energi listrik di Indonesia.

BAB II

SISTEM TENAGA LISTRIK

2.1. Sistem Tenaga Listrik^[2]

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.

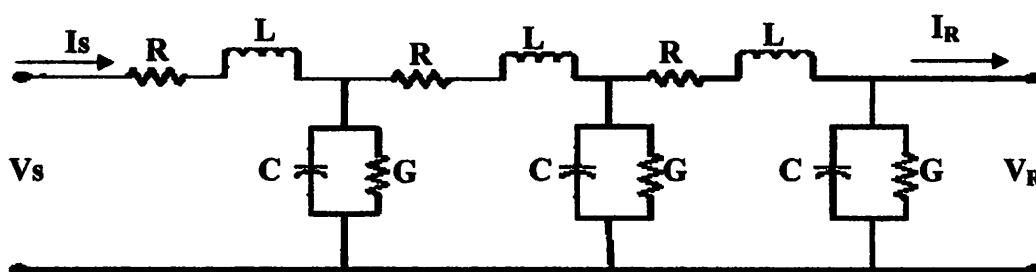
Saluran transmisi mempunyai empat parameter yang mempengaruhi kemampuannya dalam menyalurkan daya listrik. Keempat parameter tersebut yaitu : resistansi (R), induktansi (L), kapasitansi (C), serta konduktansi (G).

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 menunjukkan rangkaian yang setara dengan saluran transmisi.



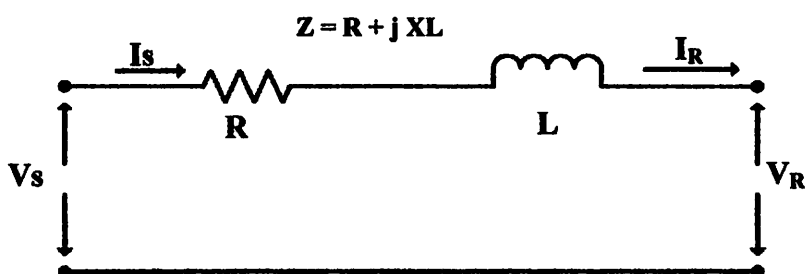
Gambar 2.1.
Rangkaian Setara Saluran Transmisi^[5]

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

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

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

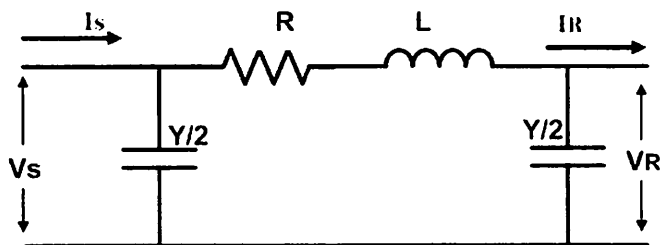
Bila kondisi tegangan pada ujung penerima diketahui, maka tegangan pada ujung sisi kirim adalah :

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

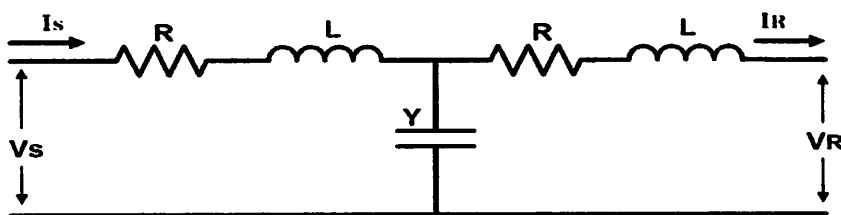
2.2.2. Saluran Transmisi Menengah^[3]

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.



(a) Rangkaian Ekivalen π



(b) Rangkaian Ekivalen T

Gambar 2.3.
Rangkaian Setara Saluran Transmisi Menengah

Untuk rangkaian π berlaku :

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

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

Untuk rangkaian T berlaku :

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

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

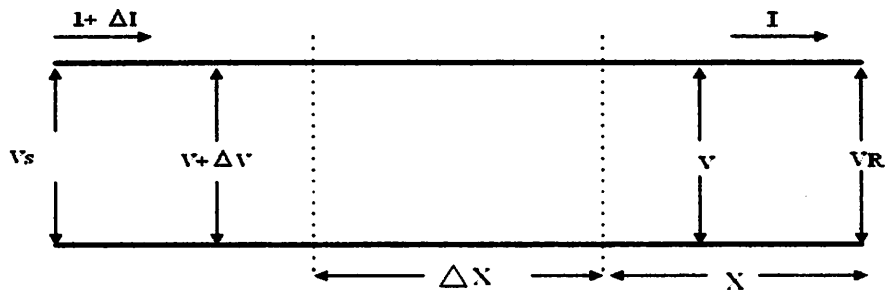
Dimana :

V_s dan I_s = Tegangan dan Arus sisi kirim

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

2.2.3. Saluran Transmisi Panjang^[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

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

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

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

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^[2]

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 yanga 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.

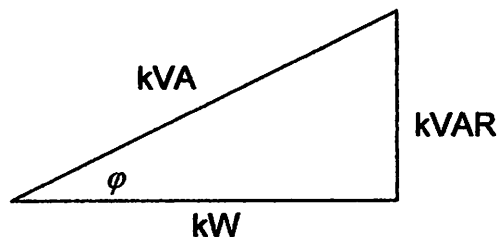
Daya reaktif induktif adalah daya listrik yang dibutuhkan untuk menghasilkan medan magnet yang dibutuhkan oleh alat-alat seperti motor induksi, transformator dan sebagainya.

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^[2]

Dimana :

$\text{Cos } \varphi$ = 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 fasa

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 :

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

2.5 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:

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

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:

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

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

2.6 Kapasitor Shunt

Kapasitor shunt adalah kapasitor yang dihubungkan secara paralel dengan saluran transmisi untuk kompensasi daya reaktif yang dilengkapi alat penghubung dan pemutus secara otomatis. Hal ini bertujuan agar kapasitor tersebut dapat terhubung dengan sistem pada saat – saat tertentu sesuai dengan besarnya kVar yang dibutuhkan, sehingga pada saat beban induktif pada titik terendah sistem tidak terjadi kelebihan daya reaktif kapasitif yang dihasilkan oleh kapasitor daya. Dengan memperhatikan besarnya variasi beban dalam sistem transmisi, maka kapasitor digunakan untuk mengurangi jatuh tegangan di sepanjang saluran dan memperbaiki faktor daya.

2.7 Pengurangan Rugi – rugi Daya Dengan Kapasitor Shunt^[6]

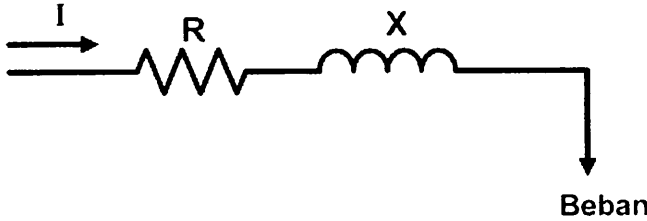
Rugi-rugi saluran per fasa dari saluran 3 fasa seimbang dengan beban terpusat seperti pada gambar 2.6. adalah $I^2 (R-jX)$ atau dapat dibedakan menjadi :

- Rugi daya aktif (I^2R) = $(I_R^2 + I_X^2) R$
- Rugi daya reaktif ($I^2 X$) = $(I_R^2 + I_X^2) X$

dimana :

I_R = komponen arus aktif

I_X = komponen arus reaktif



**Gambar 2.6^[6]
Saluran Primer Dengan Beban Terpusat**

Rugi-rugi saluran daya (I^2R) dapat dibagi menjadi dua komponen yaitu komponen arus aktif dan komponen arus reaktif. Rugi-rugi karena komponen arus aktif tidak akan mempengaruhi penempatan kapasitor *shunt* pada saluran, hal ini dapat dijelaskan sebagai berikut :

Diasumsikan bahwa rugi-rugi daya (I^2R) disebabkan oleh arus saluran I (*lagging*) yang mengalir pada resistansi R , sehingga :

$$I^2R = (I \cos \phi)^2 R + (I \sin \phi)^2 R \dots \dots \dots (2.24)$$

Setelah dipasang kapasitor shunt dengan arus I_c , didapat arus saluran baru I_1 , dan rugi-rugi daya $I_1^2 R$ sebagai berikut :

$$I_1^2 R = (I \cos \phi)^2 R + (I \sin \phi - I_c)^2 R \dots \dots \dots (2.25)$$

Sehingga pengurangan rugi-rugi daya sebagai akibat pemasangan kapasitor didapat :

$$\begin{aligned}\Delta p_{ls} &= I^2 R - I_1^2 R \\ &= (I \cos \phi)^2 R + (I \sin \phi)^2 R - (I \sin \phi - I_c)^2 R \\ &= 2(I \sin \phi) I_c R - I_c^2 R \dots \dots \dots (2.26)\end{aligned}$$

maka hanya komponen arus reaktif ($I \sin \theta$) saja yang berpengaruh terhadap pengurangan rugi daya $I^2 R$ akibat pemasangan kapasitor *shunt* pada saluran transmisi. Sehingga pengurangan rugi daya saluran 3 fasa adalah :

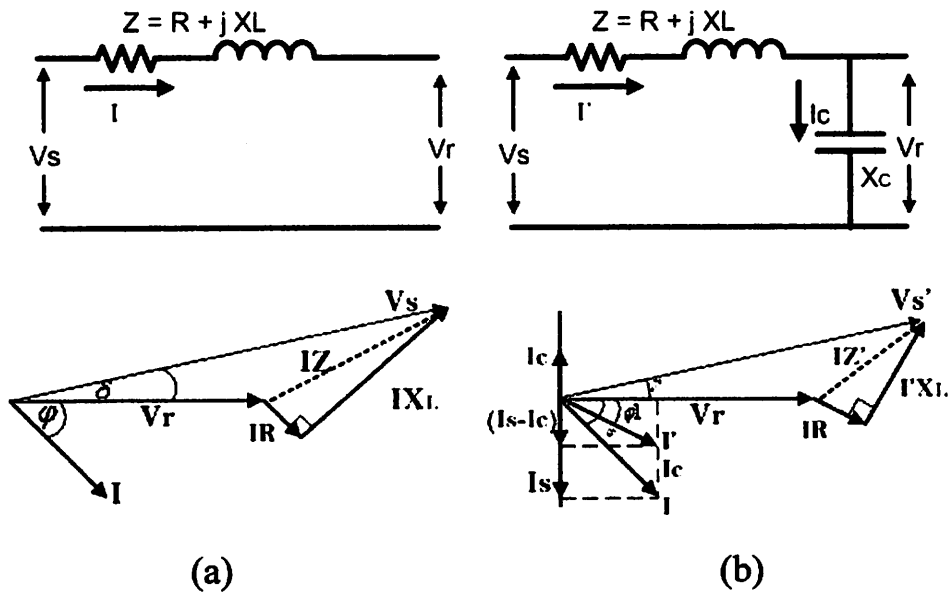
$$3R [2(I \sin \theta) I_c - I_c^2] \text{ Watt} \dots \dots \dots (2.27)$$

Bila suatu kapasitor $ckVar$ dipasang pada beban, maka faktor daya menjadi :

$$\cos \phi_1 = \frac{kW}{KVA_1} \dots \dots \dots (2.28)$$

2.8. Perbaikan Tegangan^[6]

Pemakaian kapasitor shunt dalam sistem tenaga listrik selain untuk perbaikan faktor daya juga bertujuan untuk menaikkan tegangan. Dan vektor dapat dilihat pada gambar 2.7 sebagai berikut :



Gambar 2.7^[6]
Vektor Diagram sebelum (a) dan Sesudah(b) pemasangan kapasitor shunt pada jaringan

Kerugian tegangan disebabkan arus beban I sebelum kapasitor dipasang :

$$\delta V = I_R R + I_X X_L \dots \dots \dots (2.29)$$

Kerugian tegangan setelah kapasitor dipasang :

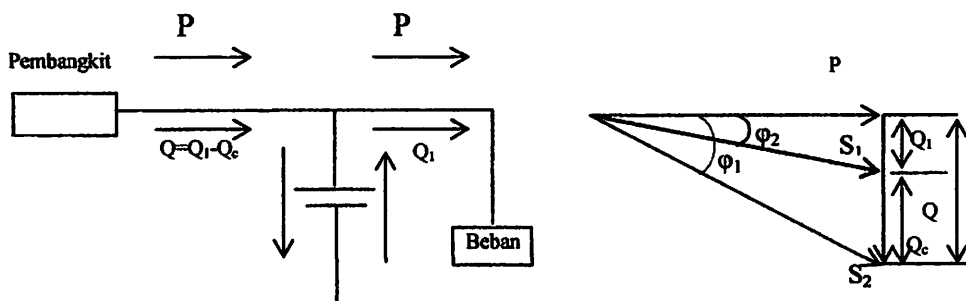
$$\delta V = I_R R + I_X X_L - I_C X_C \dots \dots \dots (2.30)$$

2.9. Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem^[6]

Manfaat terbesar yang diperoleh perbaikan faktor daya berasal dari pengurangan daya reaktif dalam sistem. Hal ini menghasilkan pengurangan biaya

pemakaian daya yang lebih rendah, kenaikan kapasitas sistem, perbaikan tegangan dan pengurangan rugi-rugi dalam sistem. Satu-satunya jalan untuk memperbaiki faktor daya adalah mengurangi daya reaktif di jaringan, jika komponen arus reaktif dapat dikurangi, maka total arus akan berkurang sedangkan komponen daya aktif tidak berubah, maka faktor daya akan lebih besar sebagai akibat berkurangnya daya reaktif. Faktor daya akan mencapai 100% jika komponen daya reaktif sama dengan nol (0).

Dengan menambah kapasitor, daya reaktif Q akan berkurang, gambar 2.8 menunjukkan perbaikan faktor daya pada sistem, kapasitor mensuplai daya reaktif ke beban.



Gambar 2.8^[6]
Perbaikan Faktor Daya

Diasumsikan bahwa beban disuplai oleh daya nyata P, daya reaktif Q dan daya semu S, pada faktor daya $\cos \phi$ tertinggal (*lagging*).

$$\cos \phi = \frac{P}{S} \dots\dots\dots(2.31)$$

Bila suatu kapasitor Q_c kVAR dipasang pada beban, faktor daya dapat diperbaiki dari $\cos \varphi_1$ menjadi $\cos \varphi_2$ dimana :

$$\cos \varphi = \frac{P}{S}$$

$$\cos \varphi = \frac{P}{\sqrt{P^2 + Q^2}}$$

$$\cos \varphi = \frac{P}{\sqrt{P^2 + (Q_1 - Q_c)^2}} \dots\dots\dots(2.32)$$

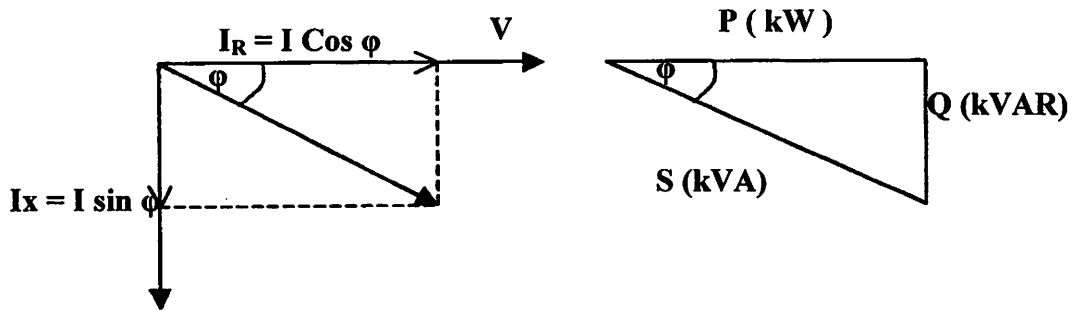
Sehingga daya semu dan daya reaktif berkurang dari S_1 (kVA) ke S_2 (kVA) dari Q_1 (kVAR) ke Q_2 (kVAR) sehingga kapasitas beban akan meningkat. Dengan demikian dapat diambil kesimpulan bahwa presentase pengurangan rugi-rugi daya jaringan dapat dihitung menggunakan rumus berikut :

$$\% \text{Rugidaya} = 100 \left(\frac{\text{Faktor daya mula - mula} (\cos \varphi_1)}{\text{Faktor daya baru} (\cos \varphi_2)} \right) \dots\dots\dots(2.33)$$

$$\% \text{Daya} = 100 \left(1 - \left(\frac{\text{Faktor daya mula - mula} (\cos \varphi_1)}{\text{Faktor daya baru} (\cos \varphi_2)} \right)^2 \right) \dots\dots\dots(2.34)$$

2.10. Perhitungan Pengaruh Perbaikan Faktor Daya^[6]

Diagram fasor dari dua komponen arus nyata itu arus aktif dan reaktif dapat dilihat pada gambar 2.9 berikut :



Gambar 2.9^[6]
Diagram Fasor Dan Sudut Faktor Daya

Penjumlahan secara vektor dari arus aktif dan reaktif menghasilkan arus total yang dapat dinyatakan dengan persamaan :

$$\begin{aligned}
 I = \text{ arus semu} &= \sqrt{(\text{ arus aktif})^2 + (\text{ arus reaktif})^2} \\
 &= \sqrt{(I \cos \phi)^2 + (I \sin \phi)^2} \dots\dots\dots(2.35)
 \end{aligned}$$

pada suatu tegangan V daya aktif, daya reaktif dan daya nyata adalah sebanding dengan arus, dimana hubungannya dapat dinyatakan sebagai berikut :

$$\begin{aligned}
 \text{Daya semu (kVA)} &= \sqrt{(\text{Daya aktif})^2 + (\text{Daya reaktif})^2} \\
 (VI) &= \sqrt{(VI \cos \phi)^2 + (VI \sin \phi)^2} \dots\dots\dots(2.36)
 \end{aligned}$$

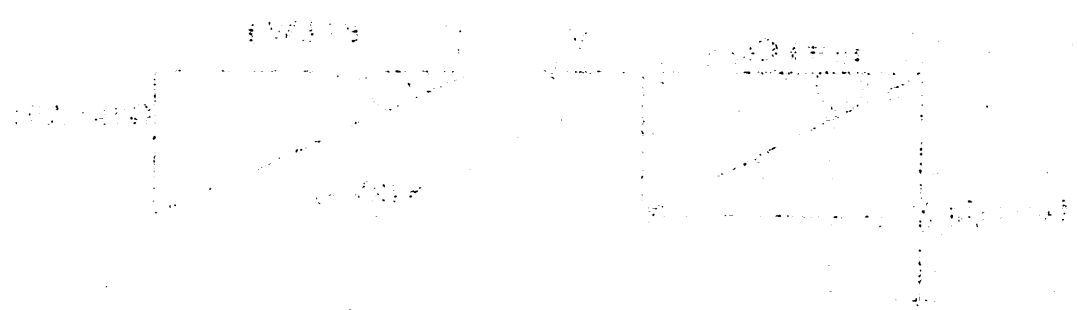
$$\text{Faktor Daya} = \frac{\text{Daya aktif}}{\text{Daya semu}} = \frac{\text{kW}}{\text{kVA}}$$

$$\text{Daya Aktif} = \text{Daya semu} \times \text{Faktor daya}$$

$$\text{kW} = \text{kVA} \times \text{Faktor daya}$$

$$\text{kW} = \text{kVA} \cos \phi$$

1. The first part of the paper is devoted to the study of the properties of the function $f(x)$ defined by the equation $f(x) = \int_0^x f(t) dt$. It is shown that $f(x)$ is a constant function.



2. In the second part, we consider the function $f(x) = \int_0^x f(t) dt + x$. It is shown that $f(x) = x$ is the only solution of this equation.

3. The third part of the paper is devoted to the study of the properties of the function $f(x) = \int_0^x f(t) dt + x^2$. It is shown that $f(x) = x^2$ is the only solution of this equation.

4. In the fourth part, we consider the function $f(x) = \int_0^x f(t) dt + x^3$. It is shown that $f(x) = x^3$ is the only solution of this equation.

5. The fifth part of the paper is devoted to the study of the properties of the function $f(x) = \int_0^x f(t) dt + x^4$. It is shown that $f(x) = x^4$ is the only solution of this equation.

6. In the sixth part, we consider the function $f(x) = \int_0^x f(t) dt + x^5$. It is shown that $f(x) = x^5$ is the only solution of this equation.

7. The seventh part of the paper is devoted to the study of the properties of the function $f(x) = \int_0^x f(t) dt + x^6$. It is shown that $f(x) = x^6$ is the only solution of this equation.

8. In the eighth part, we consider the function $f(x) = \int_0^x f(t) dt + x^7$. It is shown that $f(x) = x^7$ is the only solution of this equation.

9. The ninth part of the paper is devoted to the study of the properties of the function $f(x) = \int_0^x f(t) dt + x^8$. It is shown that $f(x) = x^8$ is the only solution of this equation.

10. In the tenth part, we consider the function $f(x) = \int_0^x f(t) dt + x^9$. It is shown that $f(x) = x^9$ is the only solution of this equation.

2.11. Penentuan Rating Kapasitor untuk Perbaikan Faktor Daya Beban^[6]

Dari hubungan fasor diagram daya aktif dan daya reaktif dapat ditulis beberapa persamaan matematis sebagai berikut :

$$\cos \varphi = \frac{\text{daya aktif}}{\text{Daya semu}} = \frac{(kW)}{(kVA)} \dots\dots\dots(2.37)$$

$$\sin \varphi = \frac{\text{Daya reaktif}}{\text{Daya semu}} = \frac{(kVAR)}{(kVA)} \dots\dots\dots(2.38)$$

$$\tan \varphi = \frac{\text{Daya reaktif}}{\text{Daya aktif}} = \frac{(kVAR)}{(kW)} \dots\dots\dots(2.39)$$

Karena komponen daya aktif biasanya konstan, dan daya semu serta komponen daya reaktif berubah sesuai dengan faktor daya, maka persamaan ini dapat ditulis sebagai berikut :

$$\begin{aligned} \text{Daya reaktif pada faktor daya mula-mula} &= \text{Daya aktif} \times \tan \varphi_1 \\ &= (kW) \times \tan \varphi_1 \end{aligned}$$

$$\begin{aligned} \text{Daya reaktif pada faktor daya baru} &= \text{Daya aktif} \times \tan \varphi_2 \\ &= (kW) \times \tan \varphi_2 \end{aligned}$$

Dengan φ_1 = sudut dari faktor daya mula-mula

φ_2 = sudut dari faktor daya yang telah diperbaiki

Rating kapasitor yang dibutuhkan perbaikan faktor daya :

$$\text{Daya reaktif (kVAR)} = \text{Daya aktif} \times \tan (\varphi_1 - \varphi_2)$$

$$(kVAR) = (kW) \times \tan (\varphi_1 - \varphi_2)$$

Untuk penyederhanaan $\tan(\varphi_1 - \varphi_2)$ sering ditulis $\Delta \tan$, yang merupakan suatu faktor pengali untuk menentukan daya reaktif.

$$\text{Daya reaktif (kVAR)} = \text{Daya aktif} \times \Delta \tan$$

$$(kVAR) = (kW) \times \Delta \tan$$

BAB III
OPTIMASI *MULTI – OBJECTIVE* VOLT VAR MENGGUNAKAN
METODE *PARETO FRONT-GA*

3.1. Analisa Aliran Daya^{[2][3]}

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. Pada 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 mempelajari analisa aliran daya ini adalah:

1. Untuk mengetahui keadaan tegangan pada setiap bus dari sistem jaringan.
2. Untuk mengetahui besarnya daya yang mengalir pada setiap cabang dari stuktur jaringan.
3. Untuk mengetahui besar rugi-rugi daya aktif dan daya reaktif pada setiap cabang jaringan.

3.1.1. Klasifikasi Bus^{[2][3]}

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

Melihat parameter diatas setiap bus dapat diklasifikasikan menjadi 3 bagian:

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

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)

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

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, oleh karena itu fasor tegangan dari bus dipakai sebagai referensi.

Daya total yang mengalir pada setiap bus dituliskan sebagai berikut:

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

Atau

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

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

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

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

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

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

3.1.2 Metode Newton Rapshon

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 $Q_{k, \text{ calc}}$). Selisih daya yang diterapkan dan perhitungan (ΔP_k dan ΔQ_k) dihitung dengan persamaan :

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

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

Selisih daya diatas digunakan untuk menghitung nilai perubahan parameter tegangan bus, yaitu $\Delta |V_k|$ dan $\Delta \delta_k$, 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 berikut:

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

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

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 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. Genetic Algorithms ^{[7][8]}

Genetic Algorithms merupakan metode pemecahan masalah yang meniru prinsip *evolusi* alam. Yang bisa digunakan untuk memecahkan suatu pencarian

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

Seperti halnya alam, algoritma ini menyelesaikan permasalahan-permasalahan dengan menemukan *kromosom-kromosom* yang baik, dengan memanipulasi materi dan sifat (*gene*) *kromosom-kromosom*. Algoritma ini tidak mengetahui tipe permasalahan yang akan diselesaikan. Hanya informasi yang telah diberikan dari *evaluasi* berupa nilai fitness setiap kromosom dengan nilai fitness terbaik yang bertahan hidup dan selalu diproduksi.

Sebelum *Genetic Algorithms* dijalankan, maka sebuah kode yang sesuai (representasi) untuk suatu permasalahan harus dirancang. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk *kromosom / string* yang terdiri dari komponen terkecil genetik yaitu *gen*. *Genetic Algorithms* bekerja dengan populasi *string* dengan melakukan proses pencarian nilai optimal secara paralel, dan juga melakukan proses rekombinasi antar individu dengan menggunakan operator genetika.

Operator-operator genetika tersebut meliputi:

Reproduksi: merupakan proses pembentukan individu lama (induk) menjadi individu baru.

Crossover : merupakan pertukaran silang gen-gen antar individu-individu yang berpasangan.

Mutasi : merupakan perubahan gen dalam suatu individu itu sendiri.

Pemakaian bilangan seperti *integer*, *floating point* dan abjad sebagai allele (nilai gen), memungkinkan penerapan operator genetika untuk menciptakan himpunan titik solusi. Untuk memeriksa hasil optimasi, kita membutuhkan fungsi *fitness* yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, pindah silang, dan mutasi untuk menciptakan keturunan (*offspring*). Jika *Genetic Algorithms* didesain dengan baik, populasi akan mengalami konvergensi dan akan mendapatkan sebuah solusi optimum.

Genetic Algorithms memiliki empat dasar kerja yaitu:

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

Algoritma genetika bekerja dengan suatu *set string* sebagai calon solusi dengan jumlah yang konstan. Populasi ini kemudian berkembang dari satu generasi ke generasi baru melalui aplikasi operasi genetika. Setiap langkah iterasi

yang disebut generasi individu dalam populasi saat itu, akan dievaluasi dan diseleksi untuk menentukan populasi pada generasi selanjutnya.

3.2.1 Istilah – Istilah Dalam *Genetic Algorithms*

Genetic Algorithms merupakan mekanisme genetika yang ada pada proses alami dan sistem buatan. Istilah-istilah yang digunakan adalah gabungan dari dua disiplin ilmu, yaitu ilmu biologi dan komputer. Istilah-istilah yang digunakan dalam *Genetic Algorithms* sebagai berikut:

Kromosom : individu berupa segmen string yang sudah ditentukan.

Gen : bagian dari string.

Loci : posisi gen.

Alele : nilai yang dimasukkan dalam gen.

Phenotype : string yang merupakan solusi terakhir.

Genotype : sejumlah string hasil perkawinan yang berpotensi sebagai solusi.

3.2.2. Parameter *Genetic Algorithms*

Terdapat beberapa parameter yang digunakan dalam *Genetic Algorithms*. Parameter tersebut digunakan untuk melihat kompleksitas dari *Genetic Algorithms*. Parameter yang digunakan tersebut adalah:

➤ **Jumlah Generasi (Max Gen)**

Merupakan jumlah perulangan (iterasi) dilakukannya rekombinasi dan seleksi. Jumlah generasi ini memengaruhi kestabilan output dan lama

iterasi (waktu proses *Genetic Algorithms*). 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 lokal optimum.

➤ **Ukuran Populasi (Pop Size)**

Ukuran populasi mempengaruhi kinerja dan efektifitas dari *Genetic Algorithms*. Jika ukuran populasi kecil maka populasi tidak menyediakan cukup materi untuk mencakup ruang permasalahan, sehingga pada umumnya kinerja *Genetic Algorithms* menjadi buruk. Dalam hal ini dibutuhkan ruang yang lebih besar untuk mempresentasikan keseluruhan ruang permasalahan. Selain itu penggunaan populasi yang besar dapat mencegah terjadinya konvergensi pada wilayah local. Zbigniew Michalewces (1996) berpendapat banyak aplikasi *Genetic Algorithms* mempergunakan populasi pada range 50 – 100.

➤ **Probabilitas Crossover (Pc)**

Probabilitas crossover ini digunakan untuk mengendalikan frekuensi operator *crossover*. Dalam hal ini, dalam populasi terdapat $P_c \times Pop\ size$ struktur (individu) yang melakukan pindah silang. Semakin besar nilai *probabilitas crossover* maka semakin cepat struktur baru yang diperkenalkan dalam populasi. Namun jika *probabilitas crossover* terlalu besar maka struktur dengan nilai fungsi obyektif yang baik dapat hilang

dengan lebih cepat dari seleksi. Sebaliknya jika probabilitas terlalu kecil akan menghalangi proses pencarian dalam proses *Genetic Algorithms*. Zbigniew Michalewics (1996) berpendapat banyak aplikasi *Genetic Algorithms* mempergunakan angka probabilitas *crossover* pada range 0,65-1.

➤ **Probabilitas Mutasi (P_m)**

Mutasi digunakan untuk meningkatkan variasi populasi digunakan untuk meningkatkan tingkat mutasi yang terjadi, karena frekuensi terjadinya mutasi tersebut menjadi $P_m \times \text{Pop Size} \times N$, dimana N adalah panjang struktur / gen dalam satu individu. Dalam *Genetic Algorithms*, mutasi menjalankan aturan penting yaitu:

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

Zbigniew Michalewics (1996) berpendapat banyak aplikasi *Genetic Algorithms* mempergunakan angka probabilitas mutasi pada range 0,001 – 0,01.

➤ **Panjang Kromosom ($NVAR$)**

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

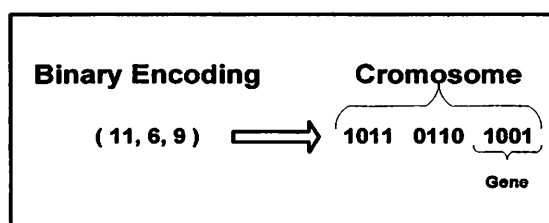
3.2.3. Proses *Genetic Algorithms*

Sangat perlu untuk mengetahui proses dalam *Genetic Algorithms*. Dibawah ini akan diuraikan mengenai hal itu, dimana uraian ini merupakan penjabaran dari *Genetic Algorithms* seperti penjelasan sebagai berikut :

A. Pengkodean atau Representasi

Langkah pertama kali yang dilakukan dalam penggunaan *Genetic Algorithms* adalah melakukan pengkodean atau representasi terhadap permasalahan yang akan dilakukan seperti pada gambar 3.1. Secara umum *Genetic Algorithms* dibentuk oleh serangkaian *kromosom* yang ditandai dengan x_i ($i = 1, 2, 3, \dots, N$). Setiap elemen dalam *kromosom* ini adalah *variable string* yang disebut *gen*, berisi nilai-nilai atau *allele*. Variabel-variabel ini dapat dinyatakan dalam bentuk bilangan biner, bilangan real (*floating point*), iteger, abjad.

Selanjutnya beberapa *kromosom* dibentuk dan berkumpul membentuk populasi. Populasi inilah populasi awal bagi *Genetic Algorithms* untuk awal melakukan pencarian.



Gambar 3.1
Pengkodean Representasi

B. Fitness Function (Fungsi Evaluasi)

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

kromosom tertentu, fungsi objektif atau fungsi evaluasi akan mengevaluasi status masing-masing *kromosom*. Setiap *gen* x_i ($i = 1, 2, 3, \dots, N$) dipergunakan untuk menghitung $f_k(x)$ ($k = 1, 2, \dots, \text{Pop Size}$).

Permasalahan klasik dalam *Genetic Algorithms* adalah berapa *kromosom* dengan nilai *fitness* yang tinggi (tetapi bukan nilai optimum), mendominasi populasi dan mengakibatkan *Genetic Algorithms* konvergen pada local optimum. Ketika mencapai konvergen, kemampuan *Genetic Algorithms* untuk mencari solusi yang lebih baik menghilang.

Tukar silang antara *kromosom* induk hampir identik. Dalam hal ini hanya operasi mutasi yang mampu menghasilkan *kromosom* yang relatif baru, dan merupakan cara untuk menghindari kromosom tertentu mendominasi populasi.

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

C. Seleksi

Pada *Genetic Algorithms* terdapat proses seleksi yaitu proses pemilihan *kromosom* yang akan di *crossover*kan dengan *kromosom* dari individu lain. Masalah yang paling mendasar pada proses ini adalah bagaimana proses penyelesaiannya. Menurut teori Darwin proses seleksi individu adalah : *individu yang terbaik akan tetap hidup dan menghasilkan keturunan*. Pada proses seleksi

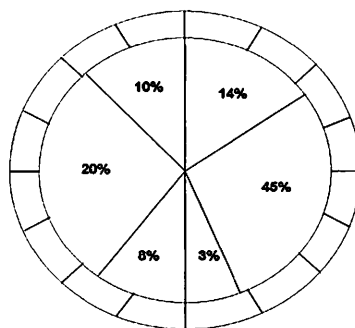
ini dapat menggunakan metode seperti *roulette wheel selection*, *rank selection* dan sebagainya.

- *Roulette wheel selection*

Dimana setiap individu memiliki harga fitness sehingga didapatkan probabilitas individu $(f(t)/\sum f(t))$ tersebut dicopykan pada populasi baru. Untuk individu yang memiliki probabilitas 20% untuk jumlah populasi 10, maka kemungkinan individual tersebut dapat terpilih sebanyak dua kali. Ilustrasi kerja operator ini dapat digambarkan seperti pada gambar 3.2.

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 0 hingga total *fitness* lebih besar atau sama dengan nilai k lalu ambil *kromosom* tersebut.



Gambar 3.2
Roulette – Wheel

Sumber : Mitsuo Gen, Runwei Cheng, " *Genetic Algorithm And Engineering Design* " (John Wiley & Son, Inc, 1994) p-7

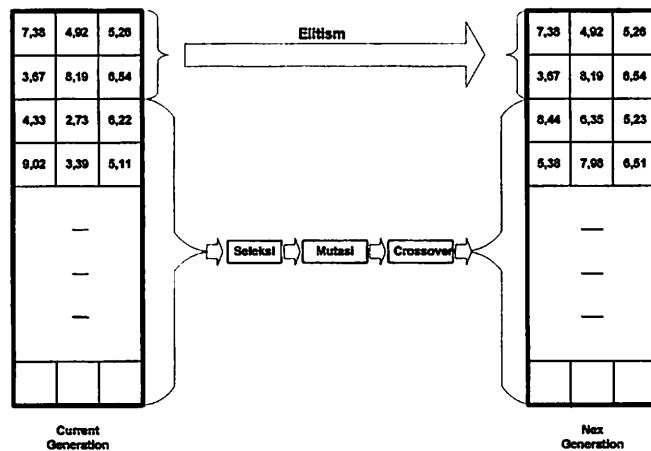
- **Rank Selection**

Apabila *fitness* yang dimiliki oleh *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 mempunyai *fitness* 2 dan seterusnya sampai yang terakhir mempunyai *fitness* N. dengan demikian semua *kromosom* akan mempunyai peluang untuk diseleksi.

3.2.4. Elitism

Selama membuat populasi baru dengan *crossover* dan *mutasi*, kemungkinan akan terjadi kehilangan *kromosom* terbaik (*best / few best*). *Elitism* adalah metode yang pertama kali meng-copy-kan *kromosom* terbaik (*best / few best*) ke dalam populasi baru. Sisanya dikerjakan secara biasa, yaitu melalui seleksi, *crossover*, dan *mutasi*. *Elitism* dapat secara tepat meningkatkan performansi dari *Genetic Algorithms*, karena *elitism* menghindarkan hilangnya solusi terbaik (*best / few best*) yang telah ditemukan. Ilustrasi kerja operator ini dapat digambarkan seperti pada gambar 3.3.



Gambar 3.3

Pembentukan Nex Generation Dalam Algoritma Genetika

Sumber : Mitsuo Gen, Runwei Cheng, " Genetic Algorithm And Engineering Design "

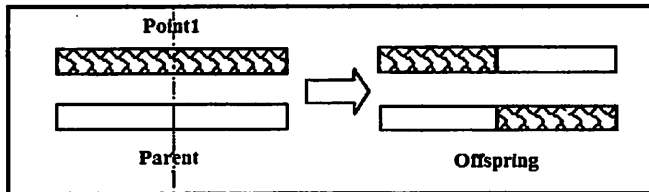
(John Wiley & Son, Inc, 1994) p-7

3.2.5. Crossover (Pindah Silang)

Fungsi dari *crossover* adalah menghasilkan kromosom anak dari kombinasi materi-materi gen dua *kromosom* induk. Cara kerjanya dengan membangkitkan sebuah nilai random r_k , dimana $k = 1, 2, \dots, Pop\ Size$. *Probabilitas Crossover* (P_c), ditentukan dan digunakan untuk mengendalikan frekuensi operator *Crossover*. Apabila nilai $r_k < P_c$ maka kromosom ke - k terpilih untuk mengalami *Crossover*. *Crossover* yang paling sederhana *One Point Crossover*. Posisi titik persilangan (*point*) ditentukan secara random pada range satu sampai panjang kromosom.

Kemudian nilai *offspring* diambil dari dua parent tersebut dengan batas titik persilangan tersebut. Ilustrasi kerja operator ini seperti pada gambar 3.3. Kemudian ditingkatkan lagi dengan menggunakan *two point crossover*. Penentuan posisi titik persilangan sama seperti *one point crossover* sebelumnya. Pemilihan

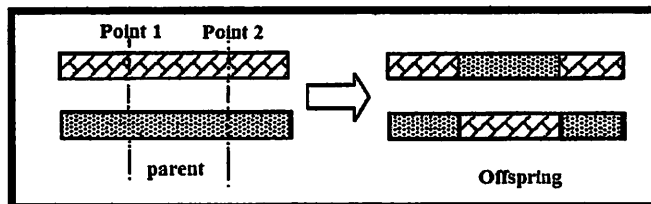
secara random dilakukan 2 kali. Kemudian nilai *offspring* diambil dari dua *parent* tersebut dengan batas dua titik persilangan tersebut. Ilustrasi kerja operator ini dapat digambarkan seperti pada gambar 3-4 dan gambar 3.5.



Gambar 3.4

Ilustrasi Operator Dengan One Point Crossover

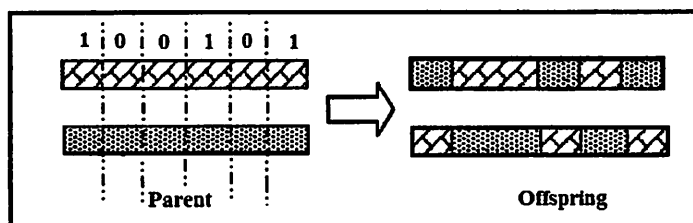
Sumber : Mitsuo Gen, Runwei Cheng, " *Genetic Algorithm And Engineering Design* "
(John Wiley & Son, Inc, 1994) p-7



Gambar 3.5

Ilustrasi Operator Dengan Two Point Crossover

Sumber : Mitsuo Gen, Runwei Cheng, " *Genetic Algorithm And Engineering Design* "
(John Wiley & Son, Inc, 1994) p-7



Gambar 3.6

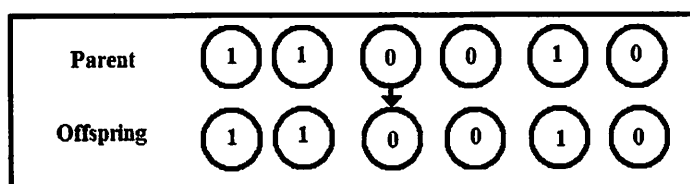
Ilustrasi Operator Dengan Uniform Crossover

Sumber : Mitsuo Gen, Runwei Cheng, " *Genetic Algorithm And Engineering Design* "
(John Wiley & Son, Inc, 1994) p-7

Untuk *uniform crossover* dibangkitkan suatu nilai random 0 dan 1 sepanjang jumlah *kromosom* untuk nilai loci. Jika nilai yang dibangkitkan mempunyai nilai 1 maka *allele parent 2* dan *offspring 2*. Untuk loci tersebut diambil *allele parent 1* dan *offspring 2* untuk loci tersebut diambil dari *allele parent 2*. Ilustrasi kerja operator ini dapat digambarkan seperti pada gambar 3.6.

3.2.6. Mutation (Mutasi)

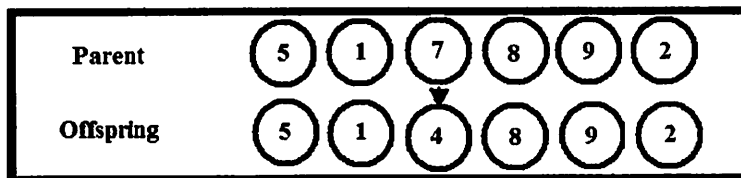
Operator mutasi digunakan untuk memodifikasi satu atau lebih nilai gen dalam satu individu. Cara kerjanya dengan membangkitkan sebuah nilai random r_k dimana $k = 1, 2, \dots, NVAR$ (panjang *kromosom*). Probabilitas mutasi (P_m) ditentukan dan digunakan untuk mengendalikan frekuensi operator mutasi. Apabila nilai random $r_k < P_m$ maka gen ke - k kromosom tersebut terpilih untuk mengalami mutasi. Untuk bentuk biner atau selain representasi *integer* atau *floating point* seperti gambar 3.7 dan 3.8, proses mutasi terjadi apabila nilai $r_k < P_m$ memenuhi, maka gen ke - k digantikan oleh suatu nilai random yang dibangkitkan pada range tertentu sesuai dengan pembentukan posisi awal.



Gambar 3-6.

Ilustrasi operator mutasi untuk representasi string biner

Sumber : Mitsuo Gen, Runwei Cheng, "*Genetic Algorithm And Engineering Design*", (John Wiley & Son, Inc., 1994) p-7



Gambar 3.7

Ilustrasi Operator Mutasi Untuk Representasi Integer

Sumber : Mitsuo Gen, Runwei Cheng, " *Genetic Algorithm And Engineering Design* "
(John Wiley & Son, Inc, 1994) p-7

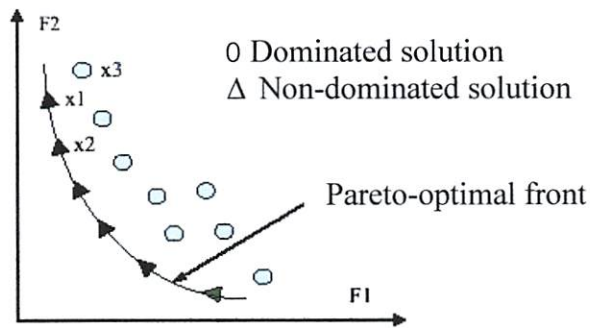
Fungsi dari operator mutasi adalah untuk menghindari agar solusi permasalahan yang diperoleh bukan merupakan solusi optimum lokal. Seperti halnya pada operator *Crossover*, tipe dan implementasi dari operator mutasi bergantung pada jenis pengkodean dan permasalahan yang dihadapi. Seberapa sering mutasi dilakukan dinyatakan dengan suatu probabilitas mutasi (P_m). Mutasi dikerjakan dengan cara melakukan perubahan pada elemen tersebut.

3.3. Konsep Optimisasi Multiobjective Volt Var Menggunakan Metode Pareto Front - GA^[1]

Optimasi *multi-objective* didasarkan pada bangunan *Pareto Front* (*Vilfredo Pareto 1896*) sebagai fungsi evaluasi didalam proses *GA*. fungsi evaluasi akan mengevaluasi status masing-masing *kromosom*. Setiap *gen* dipergunakan untuk menghitung $F_i(x)$ ($k = 1, 2, \dots, Pop\ Size$).

Solusi dikatakan Pareto-optimal (memiliki Pareto-Optimal front, atau sekumpulan solusi), jika dengan perubahannya tidak satupun fungsi sasaran dapat diperbaiki tanpa mengurangi seluruh lainnya. Semua solusi yang membentuk

Pareto-Optimal front dikatakan sebagai nondominasi (oleh solusi lainnya). Konsep dari Pareto-Optimal front, non dominasi dan solusi dominasi selanjutnya dijelaskan dalam gambar 3.9. Aksis dari gambar 3.9 (F1 dan F2) adalah dua fungsi sasaran. Kemungkinan solusi untuk minimisasi ditampilkan oleh bidang F1-F2. Solusi yang ditandai dengan segitiga disebut dengan solusi nondominasi.



Gambar 3.9^[1]

Optimisasi Pareto, solusi non-dominan dan dominan

Sebuah solusi x didominasi jika terdapat sebuah solusi y di mana seluruh fungsi sasaran F_i menjadi :

$$F_i(x) \leq F_i(y) \text{ dimana } i \in \{ 1, 2, \dots, n \}$$

Jika solusi tidak didominasi oleh solusi seseorang lainnya, kita menyebut ini solusi non dominasi (Pareto-Optimal). Jika operator dominasi adalah '>':

$$x_1 > x_3 \text{ dan } x_2 > x_3 \text{ (} x_3 \text{ dikatakan dominan)}$$

$$x_1 \not> x_2 \text{ dan } x_2 \not> x_1 \text{ (} x_1, x_2 \text{ dikatakan non-dominan)}$$

Memecahkan Pareto-Optimal front mengasumsikan lebih dari satu sasaran. Dua sasaran yang dipilih adalah (1) investasi dan (2) kerugian feeder. Dengan demikian dapat direformulasikan sebagai berikut:

$$\text{Minimisasi L: } P_{\text{loss}} = P_{\text{loss}}(\delta, V, Q_c) \dots\dots\dots (3.9)$$

$$\text{Minimisasi: investasi} = \sum_{i=1}^2 Q_{ci} \cdot P_T \dots\dots\dots (3.10)$$

Menjadi subyek terhadap : $G(\delta, V, Q_c) = 0$;

$$Q_{ci} = k Q_0 \quad k = 0, 1, 2, \dots$$

Dimana :

- F : vektor sasaran
- G : himpunan persamaan aliran daya
- δ : vektor dari sudut tegangan
- V : vektor dari magnitude tegangan
- Q_c : vektor dukungan reaktif
- Q_{ci} : dukungan reaktif yang diaplikasikan kepada bus i
- Q_{c0} : langkah dukungan reaktif incremental
- P_T : Biaya dukungan Reaktif (10\$ /kVAR)

BAB IV

OPTIMISASI *MULTI – OBJECTIVE VOLT VAR* MENGGUNAKAN *PARETO FRONT - GA* PADA SALURAN TRANSMISI 150 kV SUB SISTEM PAITON – BALI

4.1. Program Komputer Optimisasi *Multi – objective* Volt Var menggunakan *Pareto Front-GA* pada saluran transmisi 150 kv sub sistem Paiton – Bali.

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

Program komputer dalam skripsi ini dijalankan dengan menggunakan bahasa pemrograman Borland Delphi versi 7.0 dan diaplikasikan pada komputer Pentium 4. dengan Prosesor 1,8A GHz, dengan Memori 256 Mb.

4.2. Algoritma Program

4.2.1. Algoritma Aliran Daya Newton Raphson

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.

4.2.2. Algoritma Pemecahan Masalah

1. Masukan Data Beban P, Q, Data Saluran R, X, Daya Dasar (P) dan Tegangan Dasar (V) dan Data Kapasitansi Kapasitor
2. Melakukan proses Aliran Daya (*Newton Raphson*)
3. Mengecek apakah ada pelanggaran tegangan
 - a. ‘Ya’ lanjutkan ke langkah 4
 - b. ‘Tidak’ Langsung ke langkah 5
4. Melakukan Sub Routine Algoritma *Pareto Front - GA*
5. Cetak hasil
6. Stop

4.2.3. Algoritma Program Optimisasi *Multi-Objective Volt Var* Menggunakan Metode *Pareto Front – GA*

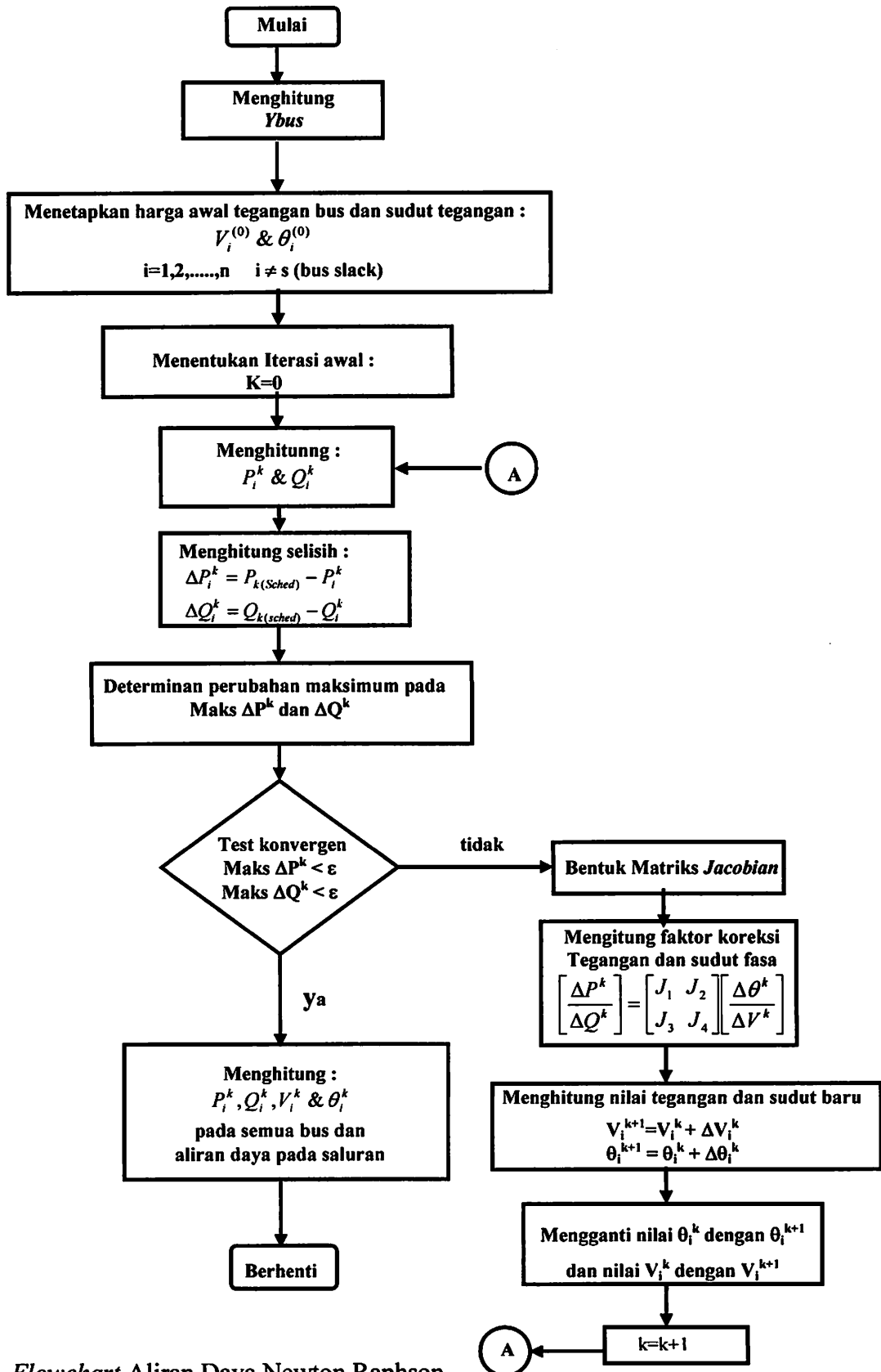
1. Masukan Data Beban P, Q, Data Saluran R, X, Daya Dasar (P), Tegangan Dasar (V), dan Data Kapasitas Kapasitor
2. Menentukan parameter algoritma genetika yaitu jumlah populasi (Pop size), maksimum generasi(Max Gen), nilai kemungkinan crossover(Pc), nilai kemungkinan mutasi (Pm) dan panjang kromosom tiap individu(NVAR).
3. Melakukan proses inialisasi parent.
4. Menghitung Fitness Parent (Pareto Front).
5. Melakukan proses statistik
6. Melakukan proses seleksi
7. Melakukan proses crossover
8. Melakukan proses mutasi
9. Apakah offspring sudah mencapai maks populasi.
 - a. Ya, Lanjutkan ke langkah 10,
 - b. Tidak, Kembali ke langkah 6.
10. Menghitung Elitism
11. Apakah Maks Gen sudah terpenuhi
 - a. Ya, Maka langkah selanjutnya
 - b. Tidak, hitung GEN+1 kembali ke langkah 5.
12. Cetak Hasil
13. Stop

4.2.4. Algoritma Program *Fitness* (*ParetoFront*)

1. Baca Data Kromosom
2. Mengubah Data Beban dari Data Kromosom
3. Melakukan proses Load Flow
4. Melakukan proses perhitungan rugi – rugi daya
5. Melakukan proses perhitungan biaya
6. Melakukan proses solusi Pareto Front
7. Kembali ke GA

4.3. Flowchart Program

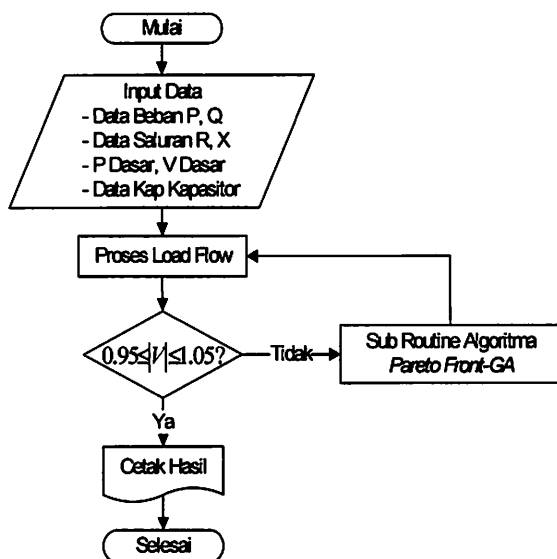
4.3.1. Flowchart Aliran Daya Newton Raphson



Gambar 4.1 Flowchart Aliran Daya Newton Raphson

4.3.2. Flowchart Algoritma Dasar

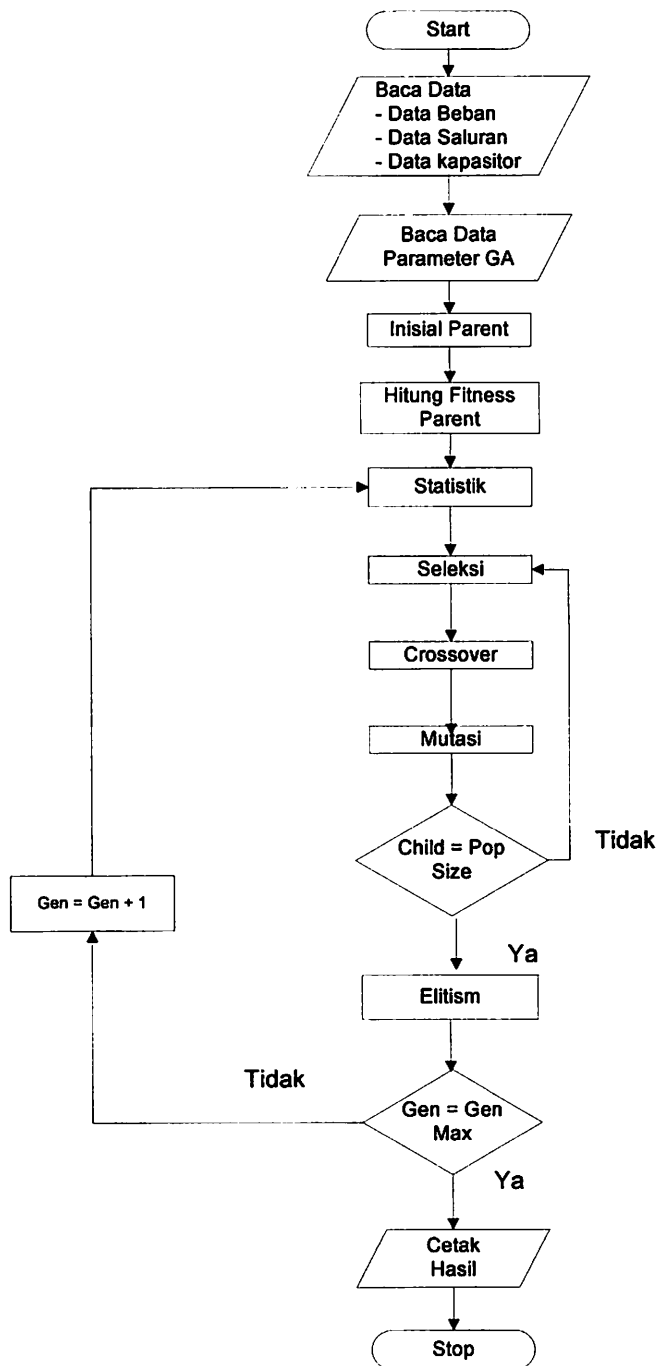
4.3.2.1. Flowchart Algoritma Pemecahan Masalah



Gambar 4.2
Flowchart Algoritma Pemecahan Masalah

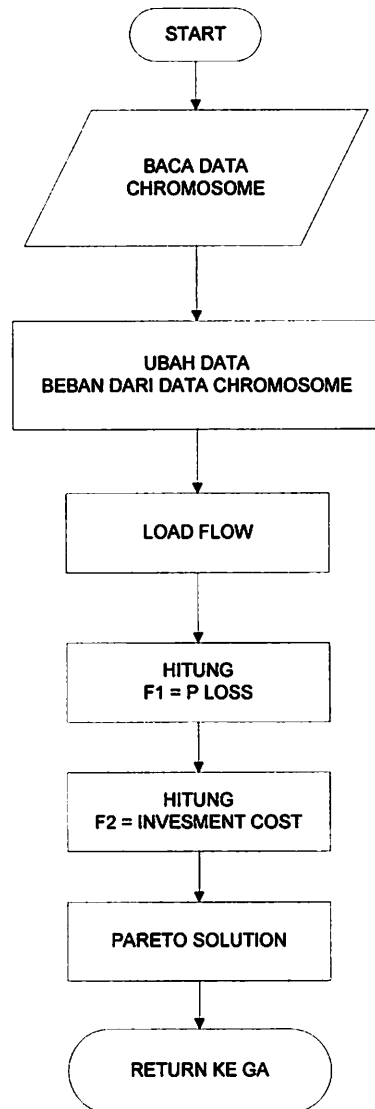
4.3.2.2. Flowchart Algoritma Program Optimisasi *Multi-Objective Volt Var*

Menggunakan Metode *Pareto Front – GA*



Gambar 4.3
Flowchart Algoritma Program Pemecahan *Multi-Objective Volt Var*
Dengan Algoritma Genetika

4.3.2.3. Flowchart Algoritma Program Fitness (Pareto Front)



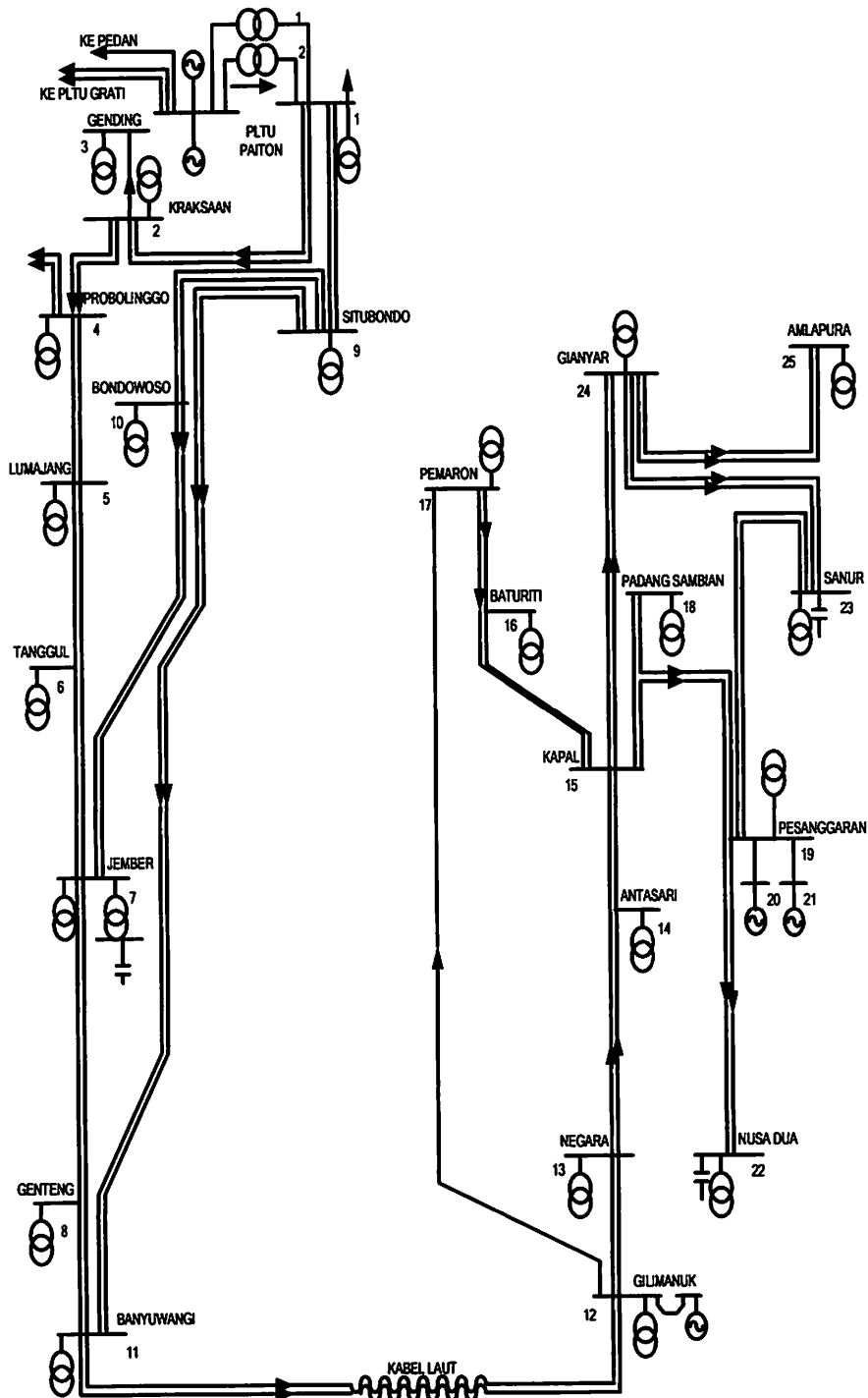
Gambar 4.4
Flowchart Algoritma Program Fitness (Pareto Front)

4.4. Sistem Transmisi 150 kV Sub Sistem Paiton – Bali

Sistem transmisi 150 kV sub sistem Paiton-Bali terdiri dari 25 bus dengan 34 saluran. Gambar 4.5 menunjukkan diagram segaris jaringan transmisi 150 kV Paiton-Bali. Untuk mengetahui penomoran bus dari sistem transmisi 150 kV Paiton Bali dapat dilihat pada tabel 4.1.

Data yang akan diambil dari saluran transmisi sub sistem Paiton-Bali adalah data pembebanan, data saluran dan data kapasitas kapasitor yang telah terpasang pada saluran tersebut. Kemudian dari data yang didapat dilakukan analisa aliran daya menggunakan metode Newton Raphson untuk mengetahui keadaan tegangan pada setiap bus dari sistem jaringan, besarnya daya yang mengalir pada setiap cabang dari struktur jaringan serta besar rugi-rugi daya aktif dan daya reaktif pada setiap cabang jaringan. Agar memudahkan perhitungan maka digunakan system per-unit (pu), dimana dasar yang digunakan:

1. Tegangan Dasar : 150 kV
2. Daya Dasar : 100 MVA



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

Gambar 4.5. Diagram Segaris Jaringan Sistem Transmisi 150 kV Paiton-Bali

Tabel 4.1.
Penomoran Bus 150 kV Sub Sistem Paiton-Bali

No. Bus	Nama Bus
1	PAITON
2	KRAKSAN
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	PDG SAMBIAN
19	PESANGGARAN
20	PLTD PESANGGARAN
21	PLTG PESANGGARAN
22	NUSADUA
23	SANUR
24	GIANYAR
25	AMLAPURA

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

Berdasarkan pada referensi [3] maka dilakukan pengolahan data untuk mengubah dasar tegangan ke dalam pu. Serta perhitungan P_{load} dan Q_{load} .

Pada bus nomor 1, maka:

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

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

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

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

$$P_{\text{beban}} = 11,7 + 27,6 = 39,30 \text{ MW}$$

$$Q_{\text{beban}} = 7,9 \text{ MVAR}$$

Selanjutnya dilakukan perhitungan P_{load} dan Q_{load} , sehingga didapatkan hasil yang ditunjukkan pada tabel 4.2.

Tabel 4.2
Data Pembangkitan dan Pembebanan 150 kV Sub Sistem Paiton - Bali

No.	Nama Bus	Tegangan	Pgenerator	Qgenerator	Pload	Qload	Tipe
		(pu)	(MW)	(MVAR)	(MW)	(MVAR)	
1	PAITON	1,033	-	-	39.3	7.9	1
2	KRAKSAN	-	-	-	14	6	3
3	GENDING	-	-	-	18	8.5	3
4	PROBOLINGGO	-	-	-	36.1	16.5	3
5	LUMAJANG	-	-	-	39	16.6	3
6	TANGGUL	-	-	-	22.5	9.7	3
7	JEMBER	-	-	50	62.2	30	3
8	GENTENG	-	-	-	41	21.3	3
9	SITUBONDO	-	-	-	23	5.9	3
10	BONDOWOSO	-	-	-	17.2	7	3

11	BANYUWANGI	-	-	-	37,8	18,7	3
12	GILIMANUK	0,992	99,9	-	5,4	2,7	2
13	NEGARA	-	-	-	11,9	4,4	3
14	ANTASARI	-	-	-	6,2	2,9	3
15	KAPAL	-	-	-	63,6	22,8	3
16	BATURITI	-	-	-	4,1	0,8	3
17	PEMARON	-	-	-	23,6	4,3	3
18	PDG SAMBIAN	-	-	-	16,4	7	3
19	PESANGGARAN	-	-	-	73,3	28,9	3
20	PLTD PSGRAN	0,95	32,5	-	0	0	2
21	PLTG PSGRAN	0,95	67,3	-	0	0	2
22	NUSADUA	-	-	25	45,5	13,6	3
23	SANUR	-	-	25	50,3	13,5	3
24	GIANYAR	-	-	-	35,4	7,9	3
25	AMLAPURA	-	-	-	14,2	3,3	3

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

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

Berdasarkan data, 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 data saluran, sehingga didapatkan hasil data yang ditunjukkan pada tabel 4.3.

Tabel 4.3
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

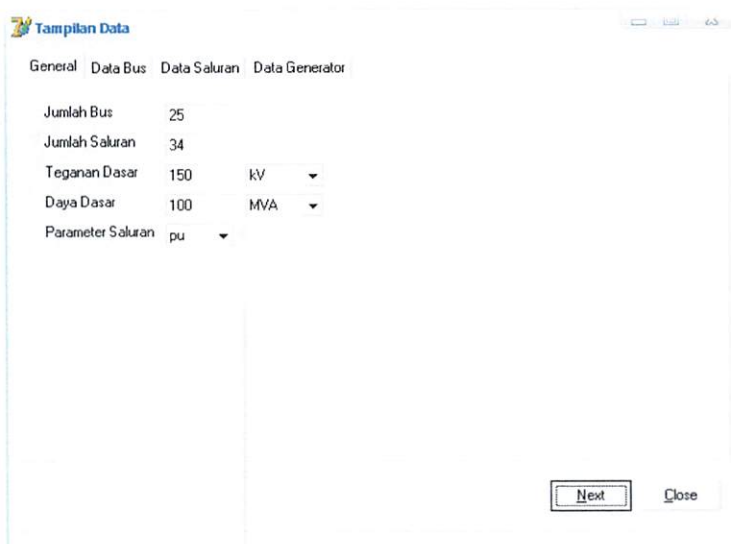
4.5. Prosedur Pelaksanaan Program Perhitungan

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

1. Menu utama program, tekan tombol *Open File* untuk membuka data yang sudah tersimpan, setelah menekan tombol *Open File* maka akan keluar tampilan data seperti pada gambar 4.7.

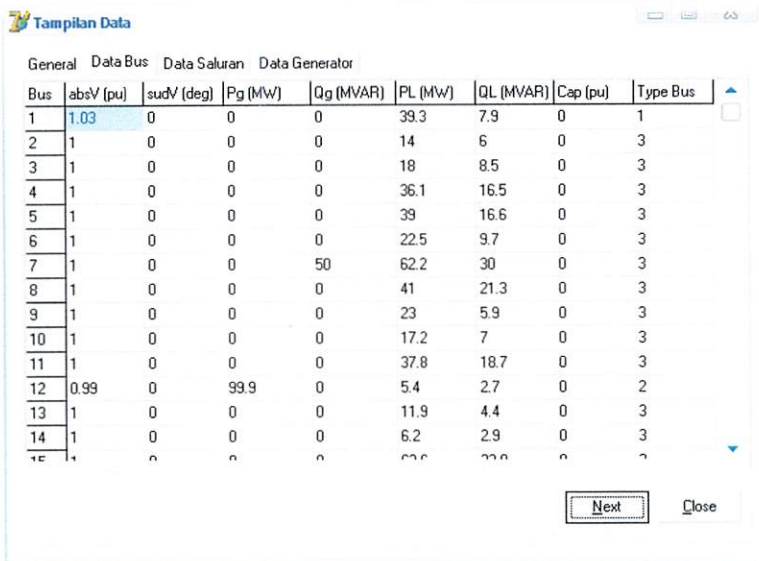


Gambar 4.6 Menu Utama Program



Gambar 4.7 Tampilan Data

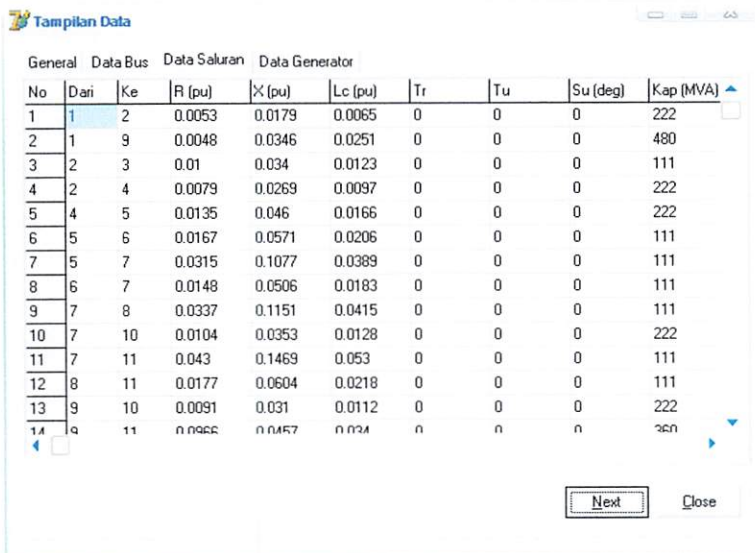
2. Kemudian tekan tombol *Data Bus*



Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Cap (pu)	Type Bus
1	1.03	0	0	0	39.3	7.9	0	1
2	1	0	0	0	14	6	0	3
3	1	0	0	0	18	8.5	0	3
4	1	0	0	0	36.1	16.5	0	3
5	1	0	0	0	39	16.6	0	3
6	1	0	0	0	22.5	9.7	0	3
7	1	0	0	50	62.2	30	0	3
8	1	0	0	0	41	21.3	0	3
9	1	0	0	0	23	5.9	0	3
10	1	0	0	0	17.2	7	0	3
11	1	0	0	0	37.8	18.7	0	3
12	0.99	0	99.9	0	5.4	2.7	0	2
13	1	0	0	0	11.9	4.4	0	3
14	1	0	0	0	6.2	2.9	0	3
15	1	0	0	0	6.2	2.9	0	3

Gambar 4.8 Tampilan Data Bus

3. Kemudian tekan tombol *Data Saluran*



No	Dari	Ke	R (pu)	X (pu)	Lc (pu)	Tr	Tu	Su (deg)	Kap (MVA)
1	1	2	0.0053	0.0179	0.0065	0	0	0	222
2	1	9	0.0048	0.0346	0.0251	0	0	0	480
3	2	3	0.01	0.034	0.0123	0	0	0	111
4	2	4	0.0079	0.0269	0.0097	0	0	0	222
5	4	5	0.0135	0.046	0.0166	0	0	0	222
6	5	6	0.0167	0.0571	0.0206	0	0	0	111
7	5	7	0.0315	0.1077	0.0389	0	0	0	111
8	6	7	0.0148	0.0506	0.0183	0	0	0	111
9	7	8	0.0337	0.1151	0.0415	0	0	0	111
10	7	10	0.0104	0.0353	0.0128	0	0	0	222
11	7	11	0.043	0.1469	0.053	0	0	0	111
12	8	11	0.0177	0.0604	0.0218	0	0	0	111
13	9	10	0.0091	0.031	0.0112	0	0	0	222
14	9	11	0.0055	0.0457	0.0134	0	0	0	360

Gambar 4.9 Tampilan Data Saluran

4. Kemudian tekan *Data Generator*

No	Bus	Qmin (MVAR)	Qmax (MVAR)	a2	a1	a0	Fix Cost	Var Cost	Pmin
1	1	-200.00	300.00	6.18000	1306.15000	388144.168	0.73	5.10	150.0
2	12	-45.00	80.00	1.40690	1599.00000	87435.0000	1.03	7.20	50.0
3	20	-20.00	60.00	14.29320	1670.00000	88960.0000	1.03	7.20	21.0
4	21	-15.00	75.00	113.90000	5297.00000	139560.000	1.03	7.20	15.0

Gambar 4.10 Tampilan Data Generator

5. Tekan tombol *Loadflow* untuk melihat hasil perhitungan *Loadflow* awal

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.03000	0.00000	570.206	-44.079	39.300	7.900	0.000	1
2	1.01638	-1.96246	0.000	0.000	14.000	6.000	0.000	3
3	1.01216	-2.26299	0.000	0.000	18.000	8.500	0.000	3
4	1.00323	-4.55311	0.000	0.000	36.100	16.500	0.000	3
5	0.99580	-8.23062	0.000	0.000	39.000	16.600	0.000	3
6	0.99713	-10.08402	0.000	0.000	22.500	9.700	0.000	3
7	1.00523	-11.10936	0.000	50.000	62.200	30.000	0.000	3
8	0.98045	-15.76323	0.000	0.000	41.000	21.300	0.000	3
9	1.04420	-6.17049	0.000	0.000	23.000	5.900	0.000	3
10	1.02352	-8.57626	0.000	0.000	17.200	7.000	0.000	3
11	0.98616	-16.94211	0.000	0.000	37.800	18.700	0.000	3
12	0.99000	-17.45425	99.900	169.608	5.400	2.700	0.000	2
13	0.96961	-17.94412	0.000	0.000	11.900	4.400	0.000	3
14	0.95263	-22.28190	0.000	0.000	6.200	2.900	0.000	3
15	0.94681	-24.43812	0.000	0.000	63.600	22.800	0.000	3

Gambar 4.11 Tampilan *Loadflow* awal

6. Tekan Aliran Daya untuk mengetahui aliran daya pada tiap bus

No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	2	206.398	19.978	1335.911	129.311	2	1	-204.249	-
2	1	9	324.508	-71.957	2100.375	-465.743	9	1	-319.526	1
3	2	3	18.037	6.094	116.868	44.001	3	2	-18.000	-
4	2	4	172.212	1.986	1128.465	51.698	4	2	-169.943	-
5	4	5	133.843	-20.261	897.294	-63.609	5	4	-131.394	-
6	5	6	51.428	-18.499	358.487	-73.283	6	5	-50.937	-
7	5	7	40.966	-23.390	293.850	-115.719	7	5	-40.311	-
8	6	7	28.437	-25.787	217.379	-136.456	7	6	-28.232	-
9	7	8	70.588	-0.393	459.867	87.645	8	7	-68.922	-
10	7	10	-131.873	-11.666	-843.284	-244.432	10	7	133.674	-
11	7	11	67.628	-8.603	451.098	30.432	11	7	-65.678	-
12	8	11	27.922	-19.202	218.185	-74.079	11	8	-27.724	-
13	9	10	152.890	26.616	952.197	273.862	10	9	-150.874	-
14	9	11	143.636	-134.983	1004.357	-758.231	11	9	-110.090	1

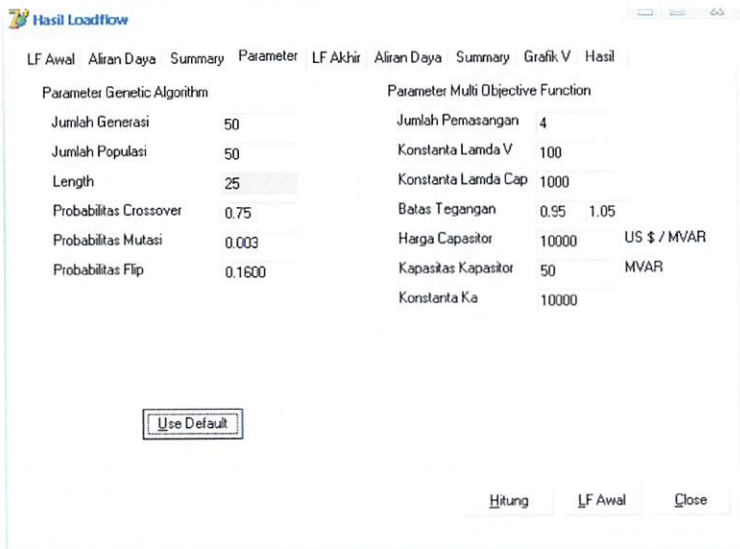
Gambar 4.12 Tampilan Aliran Daya Hasil Perhitungan *Loadflow* Awal

7. Tekan tombol *Summary* untuk melihat hasil total pembangkitan, pembebanan dan rugi daya pada kondisi awal.

Summary Loadflow	
Jumlah Pembangkitan	769.606+ j277.424 MVA
Jumlah Pembebanan	700.000+ j260.200 MVA
Jumlah Rugi-Rugi	69.606+ j17.224 MVA
Iterasi	4
Waktu Hitung	0:0:0 (jam : menit : detik : mdetik)

Gambar 4.13 Tampilan *Summary* pada kondisi awal

8. Tekan tombol *Parameter* untuk melihat Parameter dari *Genetica Algorithm*, lalu tekan tombol *use default*, kemudian tekan *Hitung*



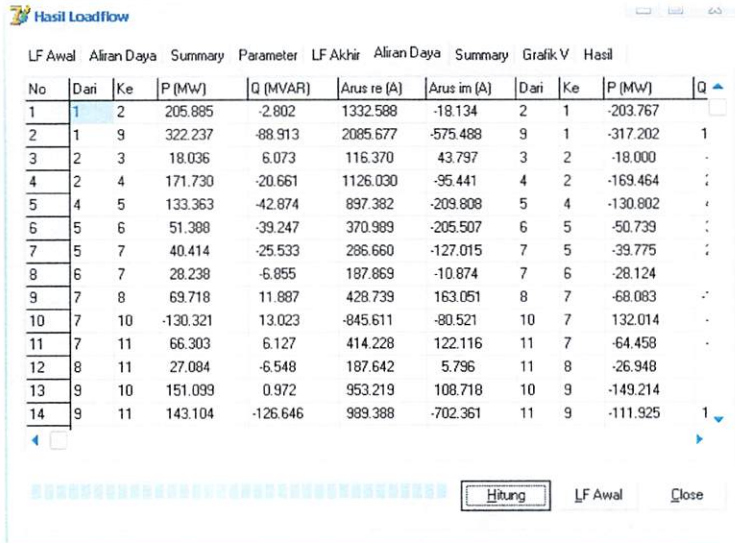
Gambar 4.14 Tampilan Parameter *Genetica Algorithm*

9. Tekan tombol *Loadflow akhir* untuk melihat hasil perhitungan *loadflow*

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.03000	0.00000	567.422	-83.815	39.300	7.900	0.000	1
2	1.02040	-2.01557	0.000	0.000	14.000	6.000	0.000	3
3	1.01620	-2.31378	0.000	0.000	18.000	8.500	0.000	3
4	1.01337	-4.66218	0.000	0.000	36.100	16.500	0.000	3
5	1.01644	-8.38643	0.000	0.000	39.000	16.600	0.000	3
6	1.02944	-10.33296	0.000	0.000	22.500	-30.300	0.000	3
7	1.02790	-11.14606	0.000	50.000	62.200	-4.000	0.000	3
8	0.98950	-15.36135	0.000	0.000	41.000	21.300	0.000	3
9	1.04998	-6.13794	0.000	0.000	23.000	5.900	0.000	3
10	1.03718	-8.59253	0.000	0.000	17.200	7.000	0.000	3
11	0.98750	-16.36642	0.000	0.000	37.800	18.700	0.000	3
12	0.99000	-16.84212	99.900	121.349	5.400	2.700	0.000	2
13	0.96889	-17.38258	0.000	0.000	11.900	4.400	0.000	3
14	0.95600	-21.76045	0.000	0.000	6.200	2.900	0.000	3
15	0.95181	-23.92779	0.000	0.000	63.600	7.800	0.000	3

Gambar 4.15 Tampilan *Loadflow akhir*

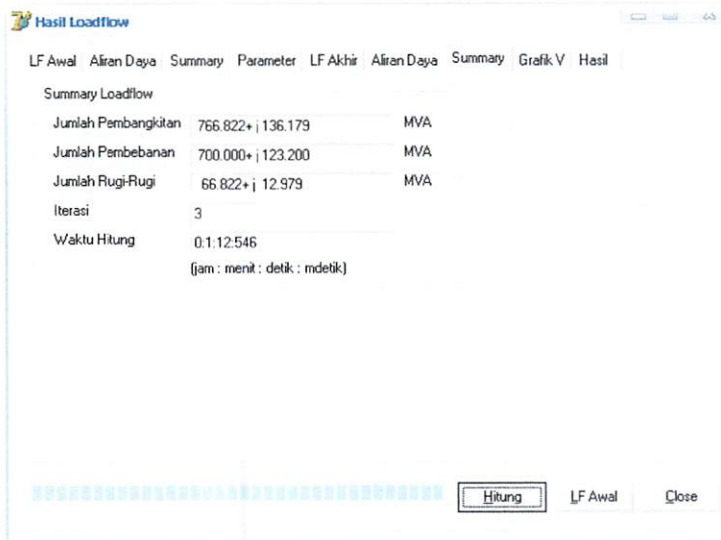
10. Tekan tombol Aliran Daya untuk mengetahui aliran daya pada tiap bus



No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	2	205.885	-2.802	1332.588	-18.134	2	1	-203.767	
2	1	9	322.237	-88.913	2085.677	-575.488	9	1	-317.202	1
3	2	3	18.036	6.073	116.370	43.797	3	2	-18.000	
4	2	4	171.730	-20.661	1126.030	-95.441	4	2	-169.464	:
5	4	5	133.363	-42.874	897.382	-209.808	5	4	-130.802	:
6	5	6	51.388	-39.247	370.989	-205.507	6	5	-50.739	:
7	5	7	40.414	-25.533	286.660	-127.015	7	5	-39.775	:
8	6	7	28.238	-6.855	187.869	-10.874	7	6	-28.124	
9	7	8	69.718	11.887	428.739	163.051	8	7	-68.083	·
10	7	10	-130.321	13.023	-845.611	-80.521	10	7	132.014	
11	7	11	66.303	6.127	414.228	122.116	11	7	-64.458	
12	8	11	27.084	-6.548	187.642	5.796	11	8	-26.948	
13	9	10	151.099	0.972	953.219	108.718	10	9	-149.214	
14	9	11	143.104	-126.646	989.388	-702.361	11	9	-111.925	1

Gambar 4.16 Tampilan Aliran Daya Tiap Bus Pada Kondisi Akhir

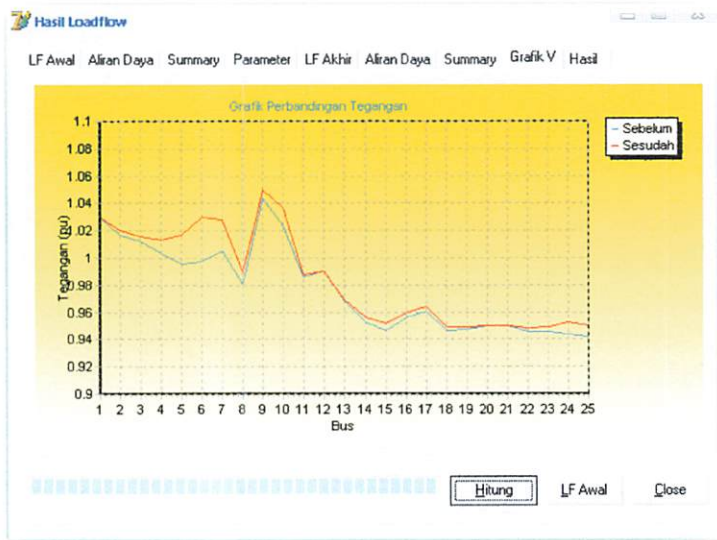
11. Kemudian tekan tombol *Summary*, untuk melihat hasil jumlah pembangkitan, pembebanan dan rugi daya pada kondisi akhir



Summary Loadflow	
Jumlah Pembangkitan	766.822+ j 136.179 MVA
Jumlah Pembebanan	700.000+ j 123.200 MVA
Jumlah Rugi-Rugi	66.822+ j 12.979 MVA
Iterasi	3
Waktu Hitung	0.1.12.546 (jam : menit : detik : mdetik)

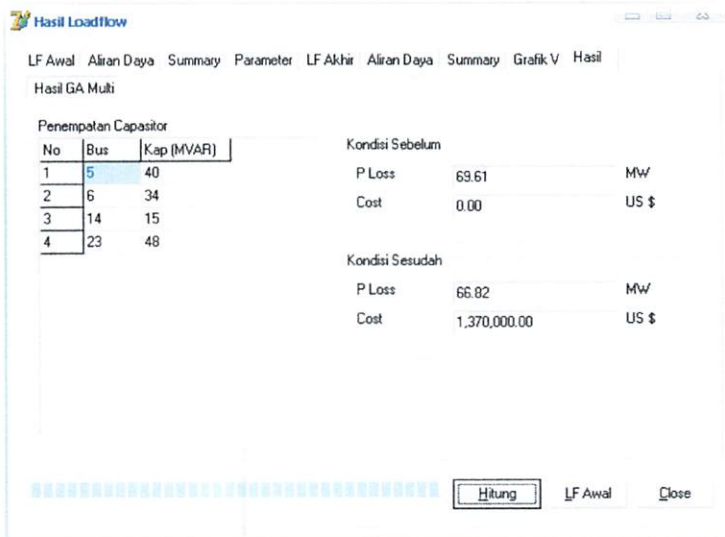
Gambar 4.17 Tampilan *Summary* pada kondisi akhir

12. Kemudian tekan tombol grafik V untuk melihat perbandingan tegangan pada tiap bus.



Gambar 4.18 Tampilan Perbandingan Tegangan Tiap Bus

13. Kemudian tekan tombol *hasil*, untuk melihat hasil penempatan kapasitor pada bus



Gambar 4.19 Tampilan Hasil Penempatan Kapasitor

4.6. Hasil Perhitungan Optimasi *Multi – Objective Volt Var* Menggunakan Metode *Pareto Front – GA* Pada Saluran Transmisi 150 kV Sub Sistem Paiton dan Bali

Hasil analisa optimasi *Multi-Objektif Volt Var* dibagi menjadi dua bagian yaitu hasil perhitungan kondisi awal dan hasil perhitungan setelah optimasi.

4.6.1. Hasil Perhitungan Kondisi Awal

Tabel 4.4
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Pada Kondisi Awal

No.	Bus	Tegangan	Sudut	Pembangkitan		Pembebanan	
				P(MW)	Q(MVAR)	P(MW)	Q(MVAR)
1	1	1.03	0	570.206	-44.079	39.3	7.9
2	2	1.01638	-1.96246	0	0	14	6
3	3	1.01216	-2.26299	0	0	18	8.5
4	4	1.00323	-4.55311	0	0	36.1	16.5
5	5	0.9958	-8.23062	0	0	39	16.6
6	6	0.99713	-10.08402	0	0	22.5	9.7
7	7	1.00523	-11.10936	0	50	62.2	30
8	8	0.98045	-15.76323	0	0	41	21.3
9	9	1.0442	-6.17049	0	0	23	5.9
10	10	1.02352	-8.57626	0	0	17.2	7
11	11	0.98616	-16.94211	0	0	37,8	18.7
12	12	0.99	-17.45425	99.9	169.608	5.4	2.7
13	13	0.96861	-17.94412	0	0	11.9	4.4
14	14	0.95263	-22.2819	0	0	6.2	2.9
15	15	0.94681	-24.43812	0	0	63.6	22.8
16	16	0.95601	-23.6128	0	0	4.1	0.8
17	17	0.96088	-23.05147	0	0	23.6	4.3
18	18	0.94631	-24.97594	0	0	16.4	7
19	19	0.94765	-25.08555	0	0	73.3	28,9
20	20	0.95	-25.04123	32.2	36.017	0	0
21	21	0.95	-24.8659	67.3	15.877	0	0
22	22	0.94557	-25.46215	0	25	45.5	13.6
23	23	0.94584	-25.2637	0	25	50.3	13.5
24	24	0.94425	-25.10259	0	0	35.4	7.9
25	25	0.94195	-25.35976	0	0	14.2	3.3

Tabel 4.5
 Hasil Perhitungan Aliran Daya Antar Saluran Pada Kondisi Awal

No.	Saluran		Daya		Saluran		Daya	
	Dari	Ke	P(MW)	Q(MVAR)	Dari	Ke	P(MW)	Q(MVAR)
1	1	2	206.398	19.978	2	1	-204.249	-13.408
2	1	9	324.508	-71.957	9	1	-319.526	105.205
3	2	3	18.037	6.094	3	2	-18	-7.24
4	2	4	172.212	1.986	4	2	-169.943	4.737
5	4	5	133.849	-20.261	5	4	-131.394	26.936
6	5	6	51.428	-18.499	6	5	-50.937	18.135
7	5	7	40.966	-23.39	7	5	-40.311	21.27
8	6	7	28.437	-25.787	7	6	-28.232	24.672
9	7	8	70.588	-0.393	8	7	-68.922	1.891
10	7	10	-131.873	-11.666	10	7	133.674	16.485
11	7	11	67.628	-8.603	11	7	-65.678	9.912
12	8	11	27.922	-19.202	11	8	-27.724	17.78
13	9	10	152.89	26.616	10	9	-150.874	-20.971
14	9	11	143.636	-134.983	11	9	-110.874	147.146
15	11	12	165.692	-182.957	12	11	-164.689	185.15
16	12	13	187.481	-16.94	13	12	-183.291	16.953
17	12	17	71.709	-1.301	17	12	-69.61	3.293
18	13	14	86.792	-9.493	14	13	-84.611	13.116
19	13	15	84.599	-10.696	15	13	-81.437	15.741
20	14	15	78.411	-13.403	15	14	-77.426	14.882
21	15	16	-19.866	-4.205	16	15	20.048	2.363
22	15	17	-21.452	-4.378	17	15	21.738	1.613
23	15	18	25.237	-8.033	18	15	-25.157	7.261
24	15	19	29.04	-13.485	19	15	-28.923	12.82
25	15	22	23.371	-11.607	22	15	-23.161	10.262
26	15	24	58.954	-6.426	24	15	-58.723	6.527
27	16	17	-24.148	-0.969	17	16	24.272	0.027
28	18	19	8.757	-13.259	19	18	-8.745	12.864
29	19	20	-32.148	-35.903	20	19	32.2	36.017
30	19	21	67.194	-15.58	21	19	67.3	15.877
31	19	22	22.412	-4.248	22	19	-22.339	3.633
32	19	23	41.298	2.585	23	19	-41.227	-2.676
33	23	24	-9.073	14.4	24	23	9.099	-14.814
34	24	25	14.224	1.413	25	24	-14.2	-2.324

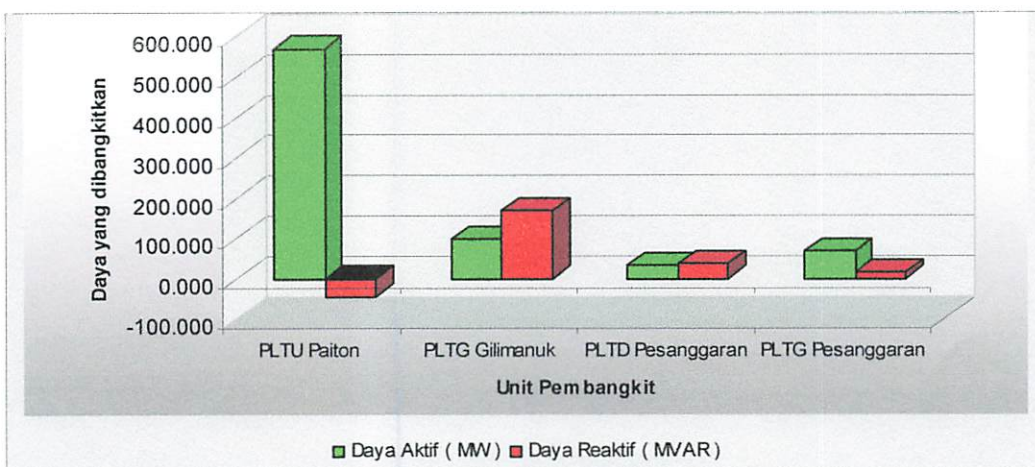
Tabel 4.6
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Awal

Pembangkitan		Pembebanan		Rugi-rugi		Waktu Komputasi
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	
769.606	277.424	700	260.2	69.606	17.224	0:0:0:0

Tabel 4.7
Hasil Perhitungan Daya Yang Dibangkitkan Pada Kondisi Awal

No.	Nama Pembangkit	Daya Yang Dibangkitkan	
		(MW)	(MVAR)
1	PLTU Paiton	570.206	-44.079
2	PLTG Gilimanuk	99.9	169.608
3	PLTD Pesanggaran	32.2	36.017
4	PLTG Pesanggaran	67.3	15.877

Grafik 4.1
Hasil Perhitungan Daya Yang Dibangkitkan Tiap Unit Pembangkit Pada Kondisi Awal



4.6.2. Hasil Perhitungan Setelah Optimasi

Tabel 4.8
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan Dan Pembebanan
Pada Kondisi Akhir

No.	Bus	Tegangan	Sudut	Pembangkitan		Pembebanan	
				P(MW)	Q(MVAR)	P(MW)	Q(MVAR)
1	1	1.03	0	567.422	-83.815	39.3	7.9
2	2	1.0204	-2.01557	0	0	14	6
3	3	1.0162	-2.31378	0	0	18	8.5
4	4	1.01337	-4.66218	0	0	36.1	16.5
5	5	1.01644	-8.38643	0	0	39	16.6
6	6	1.02944	-10.33296	0	0	22.5	9.7
7	7	1.0279	-11.14606	0	50	62.2	30
8	8	0.9895	-15.36135	0	0	41	21.3
9	9	1.04998	-6.13794	0	0	23	5.9
10	10	1.03718	-8.59253	0	0	17.2	7
11	11	0.9875	-16.36642	0	0	37,8	18.7
12	12	0.99	-16.84212	99.9	121.349	5.4	2.7
13	13	0.96889	-17.38258	0	0	11.9	4.4
14	14	0.956	-21.76405	0	0	6.2	2.9
15	15	0.95181	-23.92779	0	0	63.6	22.8
16	16	0.96008	-23.07619	0	0	4.1	0.8
17	17	0.96444	-22.50092	0	0	23.6	4.3
18	18	0.94878	-24.39038	0	0	16.4	7
19	19	0.94904	-24.46902	0	0	73.3	28,9
20	20	0.95	-24.38685	32.2	6.185	0	0
21	21	0.95	-24.21993	67.3	-7.54	0	0
22	22	0.94806	-24.87436	0	25	45.5	13.6
23	23	0.94943	-24.70642	0	25	50.3	13.5
24	24	0.95255	-24.67352	0	0	35.4	7.9
25	25	0.95027	-24.92634	0	0	14.2	3.3

Tabel 4.9
 Hasil Perhitungan Aliran Daya Antar Saluran Pada Kondisi Akhir

No.	Saluran		Daya		Saluran		Daya	
	Dari	Ke	P(MW)	Q(MVAR)	Dari	Ke	P(MW)	Q(MVAR)
1	1	2	205.885	-2.802	2	1	-203.767	9.625
2	1	9	322.237	-88.913	9	1	-317.202	122.541
3	2	3	18.036	6.073	3	2	-18	-7.23
4	2	4	171.73	-20.661	4	2	-169.464	27.37
5	4	5	133.363	-42.874	5	4	-130.802	49.895
6	5	6	51.388	-39.247	6	5	-50.739	39.34
7	5	7	40.414	-25.533	7	5	-39.34	23.699
8	6	7	28.238	-6.855	7	6	-28.124	5.308
9	7	8	69.718	11.887	8	7	-68.083	-10.688
10	7	10	-130.321	13.023	10	7	132.014	-8.632
11	7	11	66.303	6.127	11	7	-64.458	-5.424
12	8	11	27.084	-6.548	11	8	-26.984	4.878
13	9	10	151.009	0.972	10	9	-149.214	4.214
14	9	11	143.104	-126.646	11	9	-111.925	137.684
15	11	12	165.532	-145.193	12	11	-164.736	146.933
16	12	13	187.298	-24.136	13	12	-183.083	24.336
17	12	17	71.938	-3.968	17	12	-69.831	5.987
18	13	14	86.685	-13.188	14	13	-84.492	16.843
19	13	15	84.498	-14.384	15	13	-81.318	19.478
20	14	15	78.292	-17.111	15	14	-77.301	18.599
21	15	16	-19.999	-2.942	16	15	20.16	1.074
22	15	17	-21.562	-3.106	17	15	21.847	0.304
23	15	18	24.384	0.146	18	15	-24.315	-0.961
24	15	19	27.798	-1.797	19	15	-27.708	1.042
25	15	22	23.582	-7.492	22	15	-23.392	6.09
26	15	24	60.816	-25.342	24	15	-60.535	25.58
27	16	17	-24.26	0.338	17	16	24.384	-1.289
28	18	19	7.915	-5.031	19	18	-7.911	4.611
29	19	20	-32.176	-6.133	20	19	32.2	6.185
30	19	21	-67.198	7.825	21	19	67.3	-7.54
31	19	22	22.186	-8.426	22	19	-22.108	7.818
32	19	23	39.509	-26.379	23	19	-39.416	26.328
33	23	24	-10.884	-14.603	24	23	10.911	14.187
34	24	25	14.224	1.377	25	24	-14.2	-2.307

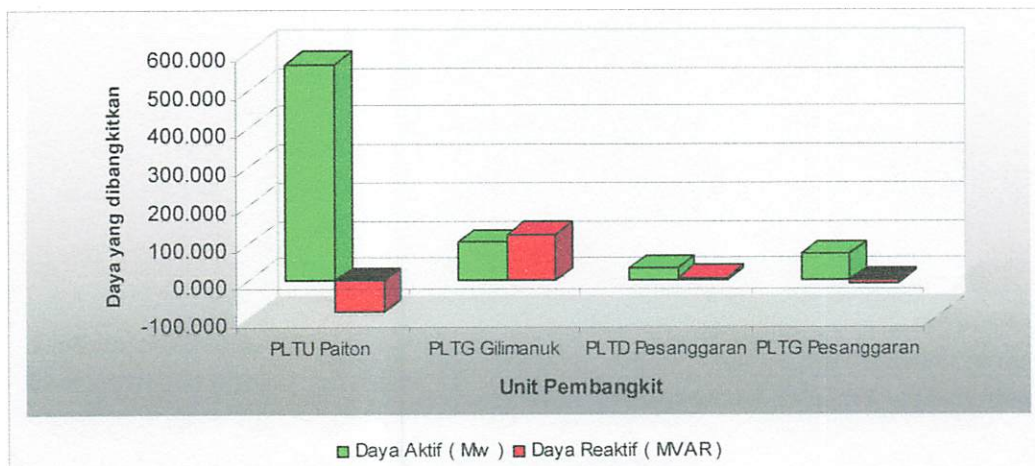
Tabel 4.10
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Akhir

Pembangkitan		Pembebanan		Rugi-rugi		Waktu Komputasi
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	
766.822	136.179	700	123.2	66.822	12.979	0:1:12:546

Tabel 4.11
Hasil Perhitungan Daya Yang Dibangkitkan Pada Kondisi Akhir

No	Nama Pembangkit	Daya Yang Dibangkitkan	
		(MW)	(MVAR)
1	PLTU Paiton	567.422	-83.815
2	PLTG Gilimanuk	99.9	121.349
3	PLTD Pesanggaran	32.2	6.185
4	PLTG Pesanggaran	67.3	-7.54

Grafik 4.2
Hasil Perhitungan Daya Yang Dibangkitkan Tiap Unit Pembangkit Pada Kondisi Akhir



4.7. Hasil Analisa Penempatan dan Kapasitas Kapasitor serta Biaya Instalasi Kapasitor Menggunakan Metode *Pareto Front – GA*

4.7.1. Penempatan Kapasitor dan Kapasitas Kapasitor

Tabel 4.12
Penempatan dan Kapasitas Kapasitor

No.	Bus	Kapasitor
		(Mvar)
1	5	40
2	6	34
3	14	15
4	23	48

Setelah dilakukan hasil perhitungan untuk penempatan kapasitor menggunakan metode *Pareto Front – GA* , maka bus yang perlu dipasang kapasitor adalah bus #5 dengan kapasitas 40 MVar, bus #6 dengan kapasitas 34 MVar, bus #14 dengan kapasitas 15 MVar, dan bus #23 dengan kapasitas 48MVar.

4.7.2. Biaya Instalasi Kapasitor

Jumlah biaya sebelum penempatan kapasitor diasumsikan nol atau belum ada biaya pemasangan kapasitor, sedangkan biaya instalasinya diasumsikan 10 US \$/ KVAR^[1]. Setelah diketahui penempatan dan kapasitas kapasitor menggunakan metode *Pareto front - GA* maka biaya instalasinya adalah jumlah total kapasitas kapasitor x 10000 US \$/MVAR = 137 MVAR x 10000 US\$/MVAR = 1.370.000 US \$.

4.8. Analisa Hasil Perhitungan Kondisi Awal dan Kondisi Akhir Optimisasi

Multi – Objective Volt Var Menggunakan Metode Pareto Front - GA

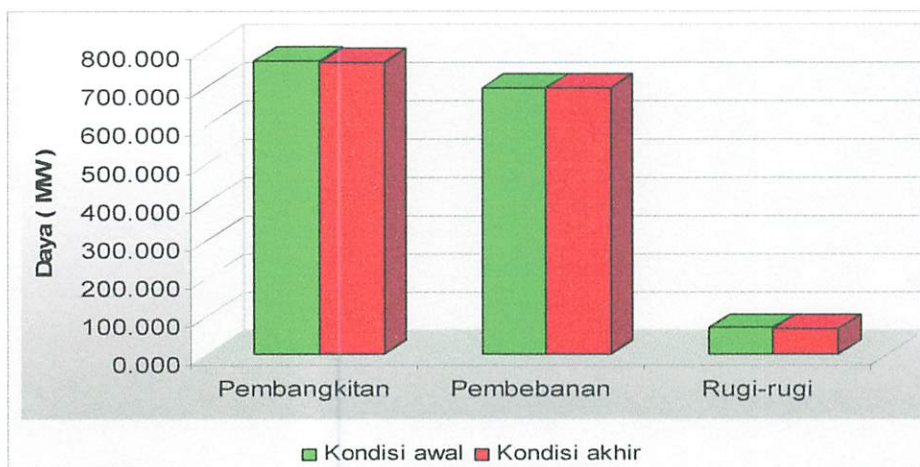
4.8.1. Perbandingan Total Daya Aktif Pembangkitan, Pembebanan dan Rugi-rugi Saluran

Tabel 4.13 dan grafik 4.3 adalah tabel yang berisi perbandingan total daya aktif pembangkitan, pembebanan dan rugi saluran transmisi antara hasil perhitungan pada kondisi awal dan kondisi akhir setelah penempatan kapasitor:

Tabel 4.13
Perbandingan Total Daya Aktif Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Awal dan Kondisi Akhir

Pembangkitan		Pembebanan		Rugi-rugi	
Kondisi Awal (MW)	Kondisi Akhir (MW)	Kondisi Awal (MW)	Kondisi Akhir (MW)	Kondisi Awal (MW)	Kondisi Akhir (MW)
769.606	766.822	700	700	69.606	66.822

Grafik 4.3.
Perbandingan Total Daya Aktif Pembangkitan, Pembebanan dan Rugi-rugi Saluran Pada Kondisi Awal dan Kondisi Akhir



4.8.2. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

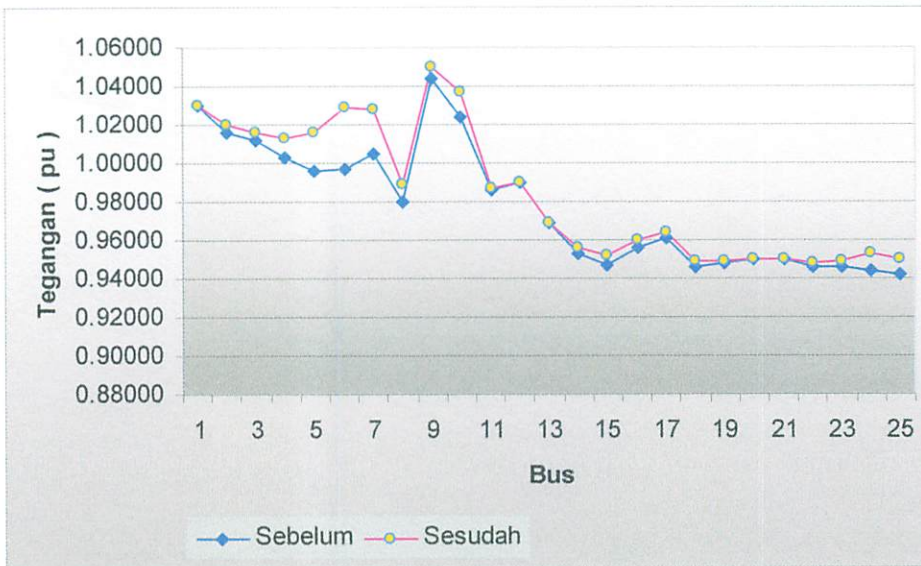
Tabel 4.14
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

Bus	Kondisi Awal		Kondisi Akhir	
	Tegangan	Sudut	Tegangan	Sudut
1	1.03	0	1.03	0
2	1.01638	-1.96246	1.0204	-2.01557
3	1.01216	-2.26299	1.0162	-2.31378
4	1.00323	-4.55311	1.01337	-4.66218
5	0.9958	-8.23062	1.01644	-8.38643
6	0.99713	-10.08402	1.02944	-10.33296
7	1.00523	-11.10936	1.0279	-11.14606
8	0.98045	-15.76323	0.9895	-15.36135
9	1.0442	-6.17049	1.04998	-6.13794
10	1.02352	-8.57626	1.03718	-8.59253
11	0.98616	-16.94211	0.9875	-16.36642
12	0.99	-17.45425	0.99	-16.84212
13	0.96861	-17.94412	0.96889	-17.38258
14	0.95263	-22.2819	0.956	-21.76405
15	0.94681	-24.43812	0.95181	-23.92779
16	0.95601	-23.6128	0.96008	-23.07619
17	0.96088	-23.05147	0.96444	-22.50092
18	0.94631	-24.97594	0.94878	-24.39038
19	0.94765	-25.08555	0.94904	-24.46902
20	0.95	-25.04123	0.95	-24.38685
21	0.95	-24.8659	0.95	-24.21993
22	0.94557	-25.46215	0.94806	-24.87436
23	0.94584	-25.2637	0.94943	-24.70642
24	0.94425	-25.10259	0.95255	-24.67352
25	0.94195	-25.35976	0.95027	-24.92634

Setelah dilakukan penempatan kapasitor, maka terjadi kenaikan profil tegangan pada beberapa bus yaitu: bus #5, tegangannya naik dari 0,99580 p.u menjadi 1,01644 p.u; bus #6, tegangannya naik dari 0,99713 p.u menjadi 1,02944 p.u; bus # 10 tegangannya naik dari 1,02352 p.u menjadi 1,03718 p.u; bus #15, tegangannya naik dari 0,94681 p.u menjadi 0,95181 p.u; bus #16, tegangannya

naik dari 0,95601 p.u menjadi 0,96008 p.u; bus #24, tegangannya naik dari 0.94425 p.u menjadi 0.95255 p.u; bus #25, tegangannya naik dari 0,94195 p.u menjadi 0,95027 p.u.

Grafik 4.4.
Perbandingan Profil Tegangan



4.8.3. Perbandingan Aliran Daya Pada Tiap Bus

Tabel 4.15
Perbandingan Aliran Daya Pada Tiap Bus

No.	Saluran		Kondisi Awal		Kondisi Akhir	
			Daya		Daya	
	Dari	Ke	P(MW)	Q(MVAR)	P(MW)	Q(MVAR)
1	1	2	206.398	19.978	205.885	-2.802
2	1	9	324.508	-71.957	322.237	-88.913
3	2	1	-204.249	-13.408	-203.767	-9.265
4	2	3	18.037	6.094	18.036	6.073
5	2	4	172.212	1.986	171.73	-20.661
6	3	2	-18	-7.24	-18	7.237
7	4	2	-169.943	4.737	-169.464	27.37
8	4	5	133.843	-20.261	133.363	-42.874
9	5	4	-131.394	26.936	-130.802	49.895
10	5	6	51.428	-18.499	51.388	-39.247
11	5	7	40.966	-23.39	40.404	-25.533

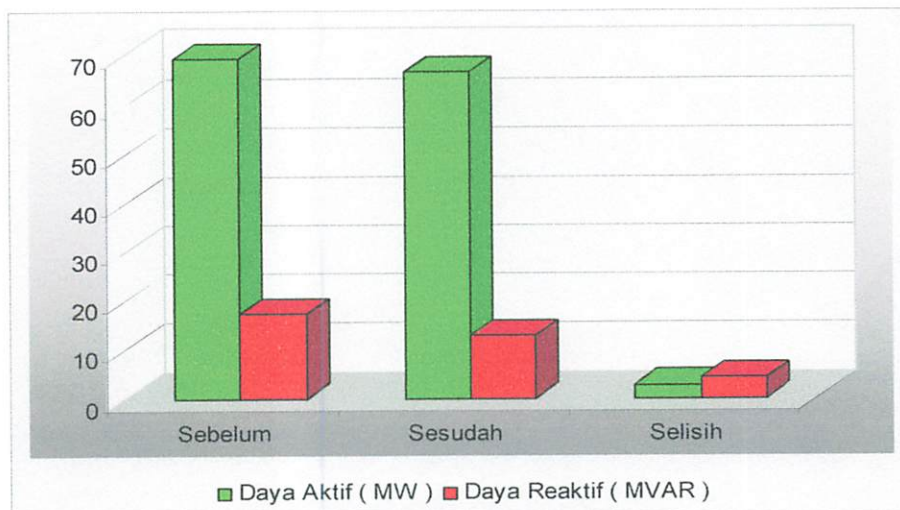
12	6	5	-50.937	18.135	-50.739	39.34
13	6	7	28.437	-25.787	28.238	-6.855
14	7	5	-40.311	21.27	-39.755	23.699
15	7	6	-28.232	24.672	-28.124	5.308
16	7	8	70.588	-0.393	69.718	11.887
17	7	10	-131.873	-11.666	-130.321	13.023
18	7	11	67.628	-8.603	66.303	6.127
19	8	7	-68.922	1.891	-68.083	-10.688
20	8	11	27.922	-19.202	27.084	-6.584
21	9	1	-319.526	105.205	-317.302	122.541
22	9	10	152.89	26.616	151.099	0.972
23	9	11	143.636	-134.983	145.104	-126.646
24	10	7	133.674	16.485	132.014	-8.623
25	10	9	-150.874	-20.971	-149.214	4.214
26	11	7	-65.678	9.912	-64.458	-5.424
27	11	8	-27.724	17.78	-26.948	4.878
28	11	9	-110.09	147.146	-111.925	137.648
29	11	12	165.692	-182.957	165.532	-145.193
30	12	11	-164.689	185.15	-164.736	146.933
31	12	13	187.481	-16.94	187.298	-24.316
32	12	17	71.709	-1.301	71.938	-3.968
33	13	12	-183.291	16.953	-183.083	24.336
34	13	14	86.792	-9.493	86.685	-13.188
35	13	15	84.599	-10.696	84.498	-14.384
36	14	13	-84.611	13.116	-84.492	16.843
37	14	15	78.411	-13.403	78.292	-17.111
38	15	13	-81.437	15.741	-81.318	19.478
39	15	14	-77.426	14.882	-77.301	18.599
40	15	16	-19.866	-4.205	-19.999	-0.942
41	15	17	-21.452	-4.378	-21.562	-3.106
42	15	18	25.237	-8.033	24.384	0.146
43	15	19	29.04	-13.485	27.798	-1.797
44	15	22	23.371	-11.607	23.582	-0.492
45	15	24	58.954	-6.426	60.816	-25.342
46	16	15	20.048	2.363	20.16	1.074
47	16	17	-24.148	-0.969	-24.26	0.338
48	17	12	-69.61	3.293	-69.831	5.987
49	17	15	21.738	1.613	21.847	0.304
50	17	16	24.272	0.027	24.384	-1.289
51	18	15	-25.157	7.261	-24.315	-0.961
52	18	19	8.757	-13.259	7.915	-5.031
53	19	15	-28.923	12.82	-27.708	1.042
54	19	18	-8.745	12.864	-7.911	4.611
55	19	20	-32.148	-35.903	32.2	6.185

56	19	21	67.194	-15.58	-67.198	7.825
57	19	22	22.412	-4.248	22.186	-8.426
58	19	23	41.298	2.585	39.509	-26.379
59	20	19	32.2	36.017	32.2	6.185
60	21	19	67.3	15.877	67.3	-7.54
61	22	15	-23.161	10.262	-23.392	6.09
62	22	19	-22.339	3.633	-22.108	7.818
63	23	19	-41.227	-2.676	-39.416	26.328
64	23	24	-9.073	14.4	-10.884	-14.603
65	24	15	-58.723	6.527	-60.535	25.58
66	24	23	9.099	-14.814	10.911	14.187
67	24	25	14.224	1.413	14.224	1.377
68	25	24	-14.2	-2.324	-14.2	-2.307

4.8.4. Perbandingan Tingkat Rugi Daya Aktif dan Daya Reaktif Pada Saluran Transmisi

Perbandingan rugi daya ditunjukkan grafik 4.6, dimana rugi daya pada saluran pada kondisi awal adalah $69,606 + j 17,224$ MVA, sedangkan rugi daya pada kondisi akhir setelah pemasangan kapasitor adalah $66,822 + j 12,979$ MVA. Dari hasil perhitungan, selisih rugi daya turun menjadi $2,784 + j 4,245$ MVA.

Grafik 4.5
Perbandingan Rugi Daya





BAB V KESIMPULAN

5.1. Kesimpulan

Setelah dilakukan analisa mengenai penentuan letak dan kapasitas kapasitor pada jaringan transmisi 150 kV dengan menggunakan metode *Pareto Front – GA*, maka dapat diambil kesimpulan:

1. Berdasarkan hasil perhitungan dengan metode metode *Pareto Front – GA*, letak kapasitor yang optimal terletak pada bus 5, 6, 14 dan 23. serta kapasitas kapasitor masing-masing adalah 40, 34, 15 dan 48 MVAR. Dan biaya instalasinya adalah 1.370.000 US \$.
2. Dari hasil perhitungan aliran daya pada saluran transmisi 150 kV sub system Paiton Bali Setelah dilakukan penempatan kapasitor, maka terjadi kenaikan profil tegangan pada beberapa bus yaitu: bus #5, tegangannya naik dari 0,99580 p.u menjadi 1,01644 p.u; bus #6, tegangannya naik dari 0,99713 p.u menjadi 1,02944 p.u; bus # 10 tegangannya naik dari 1,02352 p.u menjadi 1,03718 p.u; bus #15, tegangannya naik dari 0,94681 p.u menjadi 0,95181 p.u; bus #16, tegangannya naik dari 0,95601 p.u menjadi 0,96008 p.u; bus #24, tegangannya naik dari 0.94425 p.u menjadi 0.95255 p.u; bus #25, tegangannya naik dari 0,94195 p.u menjadi 0,95027 p.u.
3. Besar rugi daya mengalami penurunan yaitu rugi daya saluran pada kondisi awal adalah $69,606 + j 17,224$ MVA, sedangkan rugi daya pada kondisi akhir setelah pemasangan kapasitor adalah $66,822 + j 12,979$ MVA. Dari hasil perhitungan, selisih rugi daya turun menjadi $2,784 + j 4,245$ MVA.

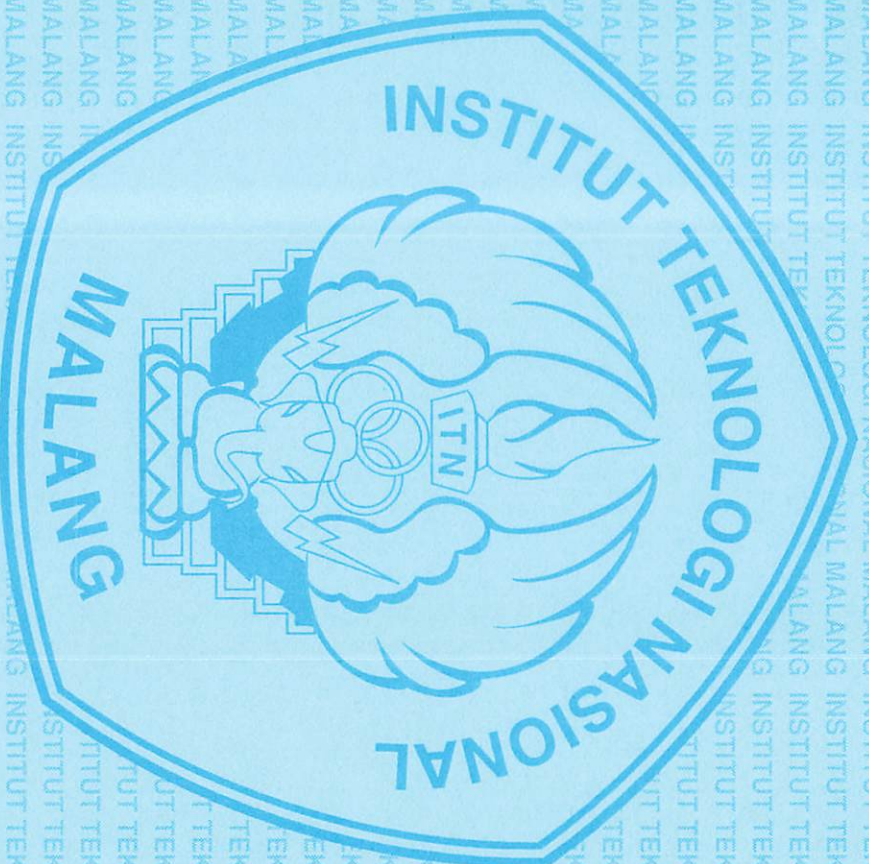
5.2. Saran – saran

Meningkatnya kebutuhan masyarakat akan energi listrik selalu bertambah, sehingga diperlukan penyediaan energi listrik yang baik dan keandalan sistem dalam penyalurannya kepada masyarakat harus baik juga. Berbagai cara telah dilakukan oleh PLN guna memenuhi kebutuhan itu.

Salah satu metode yang bermanfaat dalam mengoptimalkan suplay daya dari pembangkit sampai ke beban telah disampaikan penulis melalui skripsi ini.

DAFTAR PUSTAKA

- [1]. M.Begovic Miroslav, Radibratovic Branislav, C.Lambert Frank, (2004) School of Electrical and Computer Engineering, Georgia Institute of Technology Atlanta, " ***On Multiobjective Volt-VAR Optimization in Power Systems***", Proceedings of the 37th Hawaii International Conference on System Sciences
- [2]. Marsudi Djiteng, Ir, (1990), " ***Operasi Sistem Tenaga Listrik*** ", Balai Penerbit dan Humas Institut Sains dan Teknologi Nasional, Jakarta.
- [3]. Stevenson, William D, Jr, (1996), " ***Analisa System Tenaga Listrik***", Penerbit Erlangga edisi keempat.
- [4]. Wood, Allan J. and B. F. Wollenberg.(1996) , " ***Power Generation, Operation, and Control***", John Willey & Sons, Inc.
- [5]. Wartana I Made Ir, (1993) ,"Penyaluran Tenaga Listrik" Buku Ajar, Nopember.
- [6]. Bisri Hasan,(1990), " ***Sistem Distribusi*** ".Balai Penerbit dan Humas ISTN.
- [7]. Kusumadewi Sri, (2003), " ***Artificial Intellegence (Teknik dan Aplikasinya)***", Graha Ilmu, Yogyakarta,.
- [8]. Gen Mitsuo, Cheng Runwei, (1994), " ***Genetic Algorithm And Engineering Design***", (John Wiley & Son, Inc.)
- [9]. Baran B, Vallejos J, Ramos, Fernandez U, (2001), " ***Reactive Power Compensation using a Multi-objective Evolutionary Algorithm***" IEEE Porto Power Tech Conference, Porto, Portugal September.





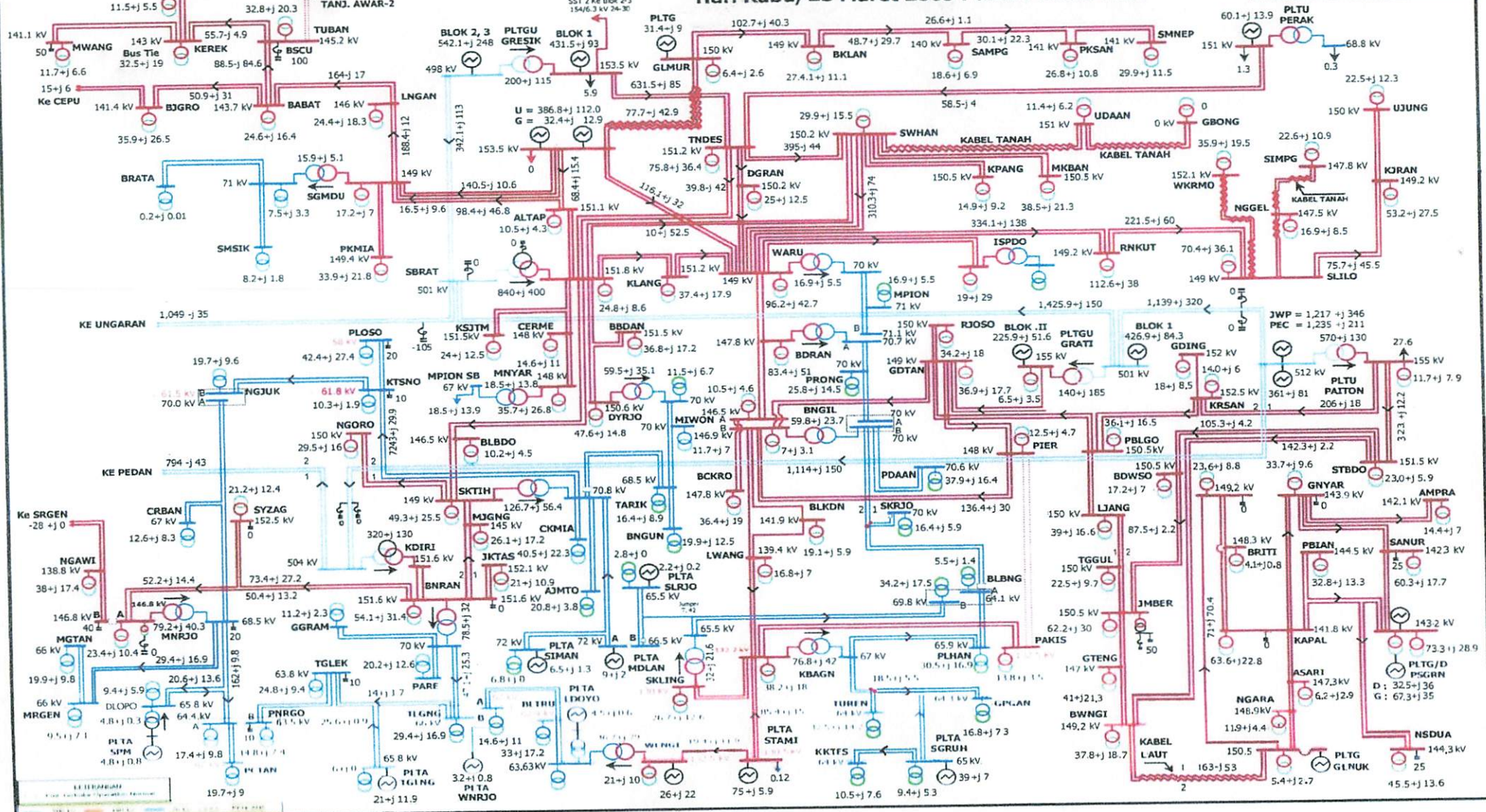
LAMPIRAN

I

- ✓ **LOAD FLOW REGION JAWA TIMUR DAN BALI**
- ✓ **DATA KARAKTERISTIK TRANSMISI UPT & SRB**
- ✓ **LISTING PROGRAM**

Lampiran-1 LOAD FLOW REGION JAWA TIMUR DAN BALI Hari Rabu, 23 Maret 2005 Pukul : 19:00 WIB

-----	: Rencana
-----	: Kabel Laut
-----	: Pembangkit Region-4
-----	: Total Transfer
-----	: Distribusi Region-4
-----	: Normal Operasi



DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jml Sirtit	Darl	Ke	No. Sirtit	Teg (kV)	Jarak (km)	Data Penghantar		I Nom (amp)	Tap CT Darl	Tap WT Darl	R (Ohm/km)	X (Ohm/km)	B (pF/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z act (Ohm)	Z brns (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Days (MVA)	
						Jenis	(2)																	
				(1)	(2)										(7)-(2)(5)	(5)-(2)(5)	(5)-(2)(5)	(3)akar(5)-(7)	(10)-(9)-(1)100	(11)-(9)-(10)	(12)-(9)-(10)	(13)-(7)-(10)	(14)-(9)(10)0,1975	(15)-(11)(10)1,7321000

6. UPT Probolinggo

1	PITON7	GRAT17	1	500	55.433	GANNET	4x392,8	2800	2000	2000	0,0251	0,2808	4,0490	1,3914	15,5856	224,4482	15,6276	2,500,000	0,0063	0,0006	0,0062	0,0045	2,424,8000
1	PITON7	GRAT17	2	500	55.433	GANNET	4x392,8	2800	2000	2000	0,0251	0,2808	4,0490	1,3914	15,5856	224,4482	15,6276	2,500,000	0,0063	0,0006	0,0062	0,0045	2,424,8000
1	BNGLS	GDTANS	1	150	16,805	ACSR, AW	330 mm ²	740	1000	1250	0,1172	0,4003	2,8530	1,9695	6,7270	47,9447	7,0094	225,0000	0,0312	0,0088	0,0299	0,1018	192,2520
1	BNGLS	GDTANS	2	150	16,805	ACSR, AW	330 mm ²	740	1000	1250	0,1172	0,4003	2,8530	1,9695	6,7270	47,9447	7,0094	225,0000	0,0312	0,0088	0,0299	0,1018	192,2520
1	BNGLS	BDRANS	1	150	8,800	ACSR, AW	330 mm ²	740	1000	1000	0,1172	0,4003	2,8530	1,486	3,9229	27,9594	4,0876	225,0000	0,0182	0,0051	0,0174	0,0063	192,2520
1	BNGLS	WARJUS	1	150	9,800	ACSR, AW	330 mm ²	740	1000	1000	0,1172	0,4003	2,8530	1,486	3,9229	27,9594	4,0876	225,0000	0,0182	0,0051	0,0174	0,0063	192,2520
1	BNGLS	PIERS	1	150	5,150	ACSR, ZEBRA	2 x 435 mm ²	1620	2000	1000	0,3870	0,2807	4,0203	1,9931	1,4456	20,7045	2,4621	225,0000	0,0109	0,0089	0,0064	0,0047	420,8760
1	BNGLS	PIERS	2	150	5,150	ACSR, ZEBRA	2 x 435 mm ²	1620	2000	1000	0,3870	0,2807	4,0203	1,9931	1,4456	20,7045	2,4621	225,0000	0,0109	0,0089	0,0064	0,0047	420,8760
1	BNGLS	BCKROS	1	150	6,200	ACSR, ZEBRA	2 x 435 mm ²	1620	1000	1250	0,3870	0,2807	4,0203	2,3994	1,7403	24,9259	2,9641	225,0000	0,0132	0,0107	0,0077	0,0056	420,8760
1	BNGLS	BCKROS	2	150	6,200	ACSR, ZEBRA	2 x 435 mm ²	1620	1000	1250	0,3870	0,2807	4,0203	2,3994	1,7403	24,9259	2,9641	225,0000	0,0132	0,0107	0,0077	0,0056	420,8760
1	BNGLS	PNMANG	1	150	27,770	ACSR, AW	330 mm ²	740	1000	1600	0,1172	0,4003	2,8530	3,2546	11,1163	79,2278	11,5630	225,0000	0,0515	0,0145	0,0494	0,0178	192,2520
1	BNGLS	LWANGS	1	150	34,660	ACSR, AW	330 mm ²	740	1000	1600	0,1172	0,4003	2,8530	4,0645	13,8824	98,9420	14,4652	225,0000	0,0643	0,0181	0,0617	0,0223	192,2520
1	GDTANS	RJOSOS	1	150	10,487	ACSR, AW	330 mm ²	740	1000	800	0,1172	0,4003	2,8530	1,2291	4,1979	28,9194	4,3742	225,0000	0,0194	0,0055	0,0187	0,0087	192,2520
1	GDTANS	RJOSOS	2	150	10,487	ACSR, AW	330 mm ²	740	1000	800	0,1172	0,4003	2,8530	1,2291	4,1979	28,9194	4,3742	225,0000	0,0194	0,0055	0,0187	0,0087	192,2520
1	PIERS	GRAT15	1	150	31,935	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	12,3588	8,9642	128,3983	15,2875	225,0000	0,0679	0,0549	0,0398	0,0289	420,8760
1	PIERS	GRAT15	2	150	31,935	TACSR	410 mm ²	1620	1000	1000	0,3870	0,2807	4,0203	12,3588	8,9642	128,3983	15,2875	225,0000	0,0679	0,0549	0,0398	0,0289	420,8760
1	PBLGOS	GDTANS	1	150	33,827	ACSR, AW	330 mm ²	740	800	1000	0,1172	0,4003	2,8530	3,9645	13,5409	96,5084	14,1094	225,0000	0,0627	0,0176	0,0602	0,0217	192,2520
1	PBLGOS	GDTANS	2	150	33,827	ACSR, AW	330 mm ²	740	800	1000	0,1172	0,4003	2,8530	3,9645	13,5409	96,5084	14,1094	225,0000	0,0627	0,0176	0,0602	0,0217	192,2520
1	PBLGOS	LJANGS	1	150	51,692	ACSR, AW	330 mm ²	740	600	800	0,1172	0,4003	2,8530	6,0663	20,6923	147,4773	21,5610	225,0000	0,0958	0,0269	0,0920	0,0332	192,2520
1	PBLGOS	LJANGS	2	150	51,692	ACSR, AW	330 mm ²	740	600	800	0,1172	0,4003	2,8530	6,0663	20,6923	147,4773	21,5610	225,0000	0,0958	0,0269	0,0920	0,0332	192,2520
1	KRSANS	PBLGOS	1	150	30,239	ACSR, AW	330 mm ²	740	1000	1000	0,1172	0,4003	2,8530	3,5440	12,1047	86,2719	12,6128	225,0000	0,0561	0,0158	0,0538	0,0194	192,2520
1	KRSANS	PBLGOS	2	150	30,239	ACSR, AW	330 mm ²	740	1000	1000	0,1172	0,4003	2,8530	3,5440	12,1047	86,2719	12,6128	225,0000	0,0561	0,0158	0,0538	0,0194	192,2520
1	KRSANS	GDINGS	1	150	18,137	ACSR, AW	330 mm ²	740	1000	400	0,1172	0,4003	2,8530	2,2429	7,6605	54,5979	7,9621	225,0000	0,0355	0,0100	0,0340	0,0123	192,2520
1	KRSANS	GDINGS	2	150	18,137	ACSR, AW	330 mm ²	740	1000	400	0,1172	0,4003	2,8530	2,2429	7,6605	54,5979	7,9621	225,0000	0,0355	0,0100	0,0340	0,0123	192,2520
1	PITONS	KRSANS	1	150	20,204	ACSR, AW	330 mm ²	740	1000	1000	0,1172	0,4003	2,8530	2,3879	8,0877	57,6420	8,4272	225,0000	0,0375	0,0105	0,0359	0,0130	192,2520

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jml Skrut	Darl	Ke	No.		Data Pengantar		Jenis	lNom (amp)	Teg CT Darl	Ke	Teg WT Darl	Ke	R (Ohm/km)	X (Ohm/km)	B (MVA/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z ket (Ohm)	Z bus (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Days (MVA)	
			Skrut	Teg (kV)	Jarak (km)	Jenis																				
				(1)	(2)								(3)	(4)	(5)	(6) (X^2+R^2)	(7) (X^2+R^2)	(8) (X^2+R^2)	(9) $Z^2 = R^2 + X^2$	$100 \times (1-R^2) / Z^2$	$(10) = (R/X) \times 100$	$(11) = (X/R) \times 100$	$(12) = (R/Z) \times 100$	$(13) = (X/Z) \times 100$	$57.735 \times (R/X) \times 100$	$1000 \times (1/R) \times (1-X^2)$

6. UPT Probolinggo

1	PTONS	KRSANS	2	150	20.204	ACSR, ANV	330 mm ²	740	1000	1000	1250	1250	0.1172	0.4003	2.8530	2.3579	8.0877	57.6420	8.4272	225.0000	0.0375	0.0105	0.0359	0.0130	192.2520
1	PTONS	STB005	1	150	55.433	ACSR, ZEBRA	2 x 435 mm ²	1600	2000	2000	1250	1250	0.3870	0.2807	4.0203	21.4526	15.5600	222.6573	26.5015	225.0000	0.1178	0.0953	0.0892	0.0502	415.6600
1	PTONS	STB005	2	150	55.433	ACSR, ZEBRA	2 x 435 mm ²	1800	2000	2000	1250	1250	0.3870	0.2807	4.0203	21.4526	15.5600	222.6573	26.5015	225.0000	0.1178	0.0953	0.0892	0.0502	415.6600
1	BNGIL4	GOTAM4	1	70	4.452	ACSR, OSTRIK	300 MCM	440	400	400	800	800	0.2175	0.3871	2.9560	0.9883	1.7234	13.1601	1.9788	49.0000	0.0403	0.0198	0.0352	0.0008	48.6960
1	BNGIL4	GOTAM4	2	70	4.452	ACSR, OSTRIK	300 MCM	400	250	400	800	800	0.2175	0.3871	2.9560	3.7549	3.7549	28.6732	4.3070	49.0000	0.0879	0.0431	0.0766	0.0014	53.3456
1	BNGIL4	PDAAM4	1	70	9.700	ACSR, OSTRIK	300 MCM	440	400	400	800	800	0.2175	0.3871	2.9560	2.1098	3.7549	28.6732	4.3070	49.0000	0.0879	0.0431	0.0766	0.0014	53.3456
1	BNGIL4	PDAAM4	2	70	9.700	ACSR, OSTRIK	300 MCM	440	400	400	800	800	0.2175	0.3871	2.9560	2.1098	3.7549	28.6732	4.3070	49.0000	0.0879	0.0431	0.0766	0.0014	53.3456
1	BNGIL4	BDRAM4	1	70	11.700	ACSR, PIPER	300 MCM	440	400	400	800	800	0.2180	0.3859	2.9850	2.5506	4.5150	34.6905	5.1857	49.0000	0.1058	0.0521	0.0921	0.0017	53.3456
1	BNGIL4	PRONC4	1	70	11.700	ACSR, PIPER	300 MCM	440	400	400	800	800	0.2180	0.3859	2.9850	2.5506	4.5150	34.6905	5.1857	49.0000	0.1058	0.0521	0.0921	0.0017	53.3456
1	BNGIL4	BLBNC4	1	70	40.000	ACSR, PIGEON	30 AWG	300	200	200	800	800	0.3866	0.3594	3.2020	15.4640	14.3360	128.0800	21.0869	49.0000	0.4303	0.3156	0.2926	0.0063	36.3720
1	BNGIL4	BLBNC4	2	70	40.000	ACSR, PIGEON	30 AWG	300	200	200	800	800	0.3866	0.3594	3.2020	15.4640	14.3360	128.0800	21.0869	49.0000	0.4303	0.3156	0.2926	0.0063	36.3720
Modifikasi TL By Graf-p/Per 1 skrit menjadi (Graf-Gdian-Per) dan Probolinggo - Gondangweitan 1 skrit menjadi (Plygo-Graf-Gdian)																									
1	GRATIS	GOTANS	1	150	10.743	TACSR	410 mm ²	2800	1000	1000	1250	2000	0.3870	0.2807	4.0203	4.1575	3.0156	43.1901	5.1360	225.0000	0.0228	0.0185	0.0134	0.0097	727.4400
1	GRATIS	GOTANS	2	150	10.743	TACSR	410 mm ²	2800	1000	1000	1250	2000	0.3870	0.2807	4.0203	4.1575	3.0156	43.1901	5.1360	225.0000	0.0228	0.0185	0.0134	0.0097	727.4400
1	GOTANS	PIERS	1	150	21.183	TACSR	410 mm ²	2800	1000	1000	1250	2000	0.3870	0.2807	4.0203	8.1978	5.9461	65.1620	10.1272	225.0000	0.0450	0.0384	0.0284	0.0192	727.4400
1	GOTANS	PIERS	2	150	21.183	TACSR	410 mm ²	2800	1000	1000	1250	2000	0.3870	0.2807	4.0203	8.1978	5.9461	65.1620	10.1272	225.0000	0.0450	0.0384	0.0284	0.0192	727.4400

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jml Shtk	Darl	Ko	No. Shtk	Data Perangkat		I Nom (Amp)	Tap CT Darl Ko	Tap WT Darl Ko	R (Ohm/km)	X (Ohm/km)	B (MVA/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z act (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Daya (MVA)
				Jenis	Jarak (Km)																
			(1)		(2)				(3)	(4)	(5)	$(\sqrt{3}) \times (6)$	$(\sqrt{3}) \times (7)$	$(\sqrt{3}) \times (8)$	$(9) \times \text{akar}(10)$	$(10) \times (11)$	$(11) \times (12)$	$(12) \times (13)$	$(13) \times (14)$	$(14) \times (15) \times 100$	

7. UPT Jember

1	PRLG05	LWANG5	1	150	51,692	ACSR DOVE	330 mm ²	740	600	800	800	0,1172	0,4003	2,6530	6,0563	20,6923	147,4773	21,5610	225,0000	0,0958	0,0289	0,0920	0,0332	192,2520
1	PRLG05	LWANG5	2	150	51,692	ACSR DOVE	330 mm ²	740	600	800	800	0,1172	0,4003	2,6530	6,0563	20,6923	147,4773	21,5610	225,0000	0,0958	0,0289	0,0920	0,0332	192,2520
1	LWANG5	TGGLU5	1	150	32,080	ACSR AW	330 mm ²	740	800	1250	1250	0,1172	0,4003	2,6530	3,7598	91,5242	13,3807	225,0000	0,0595	0,0167	0,0571	0,0205	192,2520	
1	LWANG5	JMBERS	1	150	60,530	ACSR AW	330 mm ²	740	800	1250	800	0,1172	0,4003	2,6530	7,0941	24,2302	172,6921	25,2473	225,0000	0,1122	0,0315	0,1077	0,0389	192,2520
1	TGGLU5	JMBERS	1	150	28,450	ACSR AW	330 mm ²	740	1000	800	800	0,1172	0,4003	2,6530	3,3343	11,3885	81,1679	11,8566	225,0000	0,0527	0,0148	0,0506	0,0183	192,2520
1	JMBERS	BWNGS5	1	150	92,595	ACSR AW	330 mm ²	740	600	800	1250	0,1172	0,4003	2,6530	9,6801	33,0628	235,6435	34,4507	225,0000	0,1200	0,0430	0,1488	0,0350	192,2520
1	JMBERS	JMBERS	1	150	64,708	ACSR AW	330 mm ²	740	600	800	800	0,1172	0,4003	2,6530	7,9338	25,9026	184,6119	28,9900	225,0000	0,0735	0,0207	0,0708	0,0255	192,2520
1	BWNGS5	JMBERS	1	150	39,682	ACSR AW	330 mm ²	740	800	800	1600	0,1172	0,4003	2,6530	4,6484	15,8767	113,1557	16,5432	225,0000	0,0735	0,0207	0,0708	0,0255	192,2520
1	BWNGS5	PITONS	2	150	55,433	ACSR ZEBRA	2 x 435 mm ²	1620	2000	2000	1250	0,3870	0,2807	4,0203	21,4526	15,5800	222,8573	28,5015	225,0000	0,1178	0,0353	0,0692	0,0302	420,8760
1	STB005	PITONS	1	150	55,433	ACSR ZEBRA	2 x 435 mm ²	1620	2000	2000	1250	0,3870	0,2807	4,0203	21,4526	15,5800	222,8573	28,5015	225,0000	0,1178	0,0353	0,0692	0,0302	420,8760
1	STB005	BDWSOS	2	150	34,786	ACSR AW	330 mm ²	740	800	800	800	0,1172	0,4003	2,6530	4,0769	13,9248	99,2445	14,5094	225,0000	0,0645	0,0181	0,0618	0,0223	192,2520
1	STB005	BDWSOS	1	150	34,786	ACSR AW	330 mm ²	740	800	800	1600	0,1172	0,4003	2,6530	4,0769	13,9248	99,2445	14,5094	225,0000	0,0645	0,0181	0,0618	0,0223	192,2520
1	STB005	BDWSOS	2	150	74,200	ACSR DOVE	2 x 340 mm ²	1200	800	800	1600	0,0588	0,2815	4,0740	4,3481	20,8973	302,2908	21,3351	225,0000	0,0948	0,0193	0,0928	0,0360	311,7500
1	STB005	BWNGS5	1	150	74,200	ACSR DOVE	2 x 340 mm ²	1200	800	800	1600	0,0588	0,2815	4,0740	4,3481	20,8973	302,2908	21,3351	225,0000	0,0948	0,0193	0,0928	0,0360	311,7500
1	STB005	BWNGS5	2	150	33,950	ACSR DOVE	330 mm ²	740	1000	600	1250	0,1172	0,4003	2,6530	3,9789	13,5902	96,8594	14,1607	225,0000	0,0629	0,0177	0,0604	0,0218	192,2520
1	STB005	BWNGS5	1	150	33,950	ACSR DOVE	330 mm ²	740	1000	600	1250	0,1172	0,4003	2,6530	3,9789	13,5902	96,8594	14,1607	225,0000	0,0629	0,0177	0,0604	0,0218	192,2520
1	BWNGS5	KTPNG5	1	150	7,990	ACSR HAWK	477 MM	655	600	800	800	0,1370	0,3966	2,8900	1,0946	3,1688	23,0112	1,7157	225,0000	0,0149	0,0032	0,0089	0,0052	129,9000
1	BWNGS5	KTP-GLM	1	150	4,829	OSS	300 mm ²	465	600	800	800	0,1370	0,3966	2,8900	1,0946	3,1688	23,0112	1,7157	225,0000	0,0149	0,0032	0,0089	0,0052	129,9000
1	BWNGS5	KTP-GLM	2	150	7,990	ACSR HAWK	477 MM	655	600	800	800	0,1370	0,3966	2,8900	1,0946	3,1688	23,0112	1,7157	225,0000	0,0149	0,0032	0,0089	0,0052	129,9000
1	BWNGS5	KTP-GLM	1	150	4,389	OSS	300 mm ²	500	600	800	800	0,1643	0,3546	2,8900	0,7211	1,5563	23,0112	1,7153	225,0000	0,0076	#DNV01	0,0141	0,0052	129,9000
1	BWNGS5	KTP-GLM	1	150	4,260	SUBMARINE CABLE	300 mm ²	465	600	800	800	0,1643	0,3546	2,8900	0,7211	1,5563	23,0112	1,7153	225,0000	0,0076	#DNV01	0,0141	0,0052	129,9000

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jml Stakt	Dati	Ke	No. Shttt	Teg (V)	Jarak (Km)	Dati - Penghantar		I Nom (Amp)	Teg CT Dati	Ke	Teg HT Dati	Ke	R (Ohm/km)	X (Ohm/km)	B (MVA/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z act (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Daya (MVA)		
						Jenis	Jenis																				
				(1)	(2)								(3)	(4)	(5)	$(\frac{R}{2})^2 + (\frac{X}{2})^2$	$\frac{R}{2}$	$\frac{X}{2}$	$\frac{B}{2}$	$(9) = \frac{Z_{act}^2}{Z_{base}}$	$\frac{Z_{base}}{100}$	$\frac{(10)}{100}$	$\frac{(11)}{100}$	$\frac{(12)}{100}$	$\frac{(13)}{100}$	$(14) = \frac{B}{V^2} \times 100$	$(15) = (1) \times (A) \times 1,732 / 1000$

Sub Region Ball

1	GLUKUS	CH GLUKUK	1	150	1,720	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	0,2356	0,6822	4,9536	0,7217	225,0000	0,0032	0,0010	0,0030	0,0011	167,5710
1	GLUKUS	CH GLUKUK	2	150	1,720	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	0,2356	0,6822	4,9536	0,7217	225,0000	0,0032	0,0010	0,0030	0,0011	167,5710
1	CH GLUKUK	CH KTRNG	1	150	4,471	OFC	3 x 300 mm ²	462	800	800	800	800	0,1613	0,3482	0,7212	1,5568	-	1,7157	225,0000	0,0076	0,0032	0,0031	0,0069	120,0276	
1	CH GLUKUK	CH KTRNG	2	150	4,389	OFC	3 x 300 mm ²	462	800	800	800	800	0,1613	0,3548	0,7079	1,5572	-	1,7106	225,0000	0,0076	0,0032	0,0031	0,0069	120,0276	
1	CH KTRNG	BHNGS	1	150	7,990	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	1,0946	3,1688	23,0112	3,3526	225,0000	0,0149	0,0049	0,0141	0,0052	167,5710
1	CH KTRNG	BHNGS	2	150	7,990	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	1,0946	3,1688	23,0112	3,3526	225,0000	0,0149	0,0049	0,0141	0,0052	167,5710
1	GLUKUS	NGARAS	1	150	38,040	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	5,2115	15,0867	109,5552	15,9614	225,0000	0,0709	0,0232	0,0671	0,0247	167,5710
1	GLUKUS	NGARAS	2	150	38,040	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	5,2115	15,0867	109,5552	15,9614	225,0000	0,0709	0,0232	0,0671	0,0247	167,5710
1	NGARAS	PMRONS	1	150	75,900	ACSR, HAWK	2 x 477 MCM	1280	800	800	1250	1250	0,1183	0,3995	2,9380	8,9790	29,5631	222,9942	30,8965	225,0000	0,1373	0,0099	0,1314	0,0392	335,1420
1	NGARAS	PMRONS	2	150	75,900	ACSR, HAWK	2 x 477 MCM	1280	800	800	1250	1250	0,1183	0,3995	2,9380	8,9790	29,5631	222,9942	30,8965	225,0000	0,1373	0,0099	0,1314	0,0392	335,1420
1	NGARAS	ASARIS	1	150	67,710	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	6,0828	17,8690	127,9608	18,6300	225,0000	0,0828	0,0270	0,0783	0,0288	167,5710
1	NGARAS	ASARIS	2	150	67,710	ACSR, HAWK	477 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	6,0828	17,8690	127,9608	18,6300	225,0000	0,0828	0,0270	0,0783	0,0288	167,5710
1	KAPALAS	ANSRIS	1	150	23,310	ACSR, HAWK	477 MCM	645	800	800	830	830	0,2136	0,4048	2,7970	12,5127	163,8483	28,8120	225,0000	0,1192	0,0556	0,1054	0,0399	103,9200	
1	KAPALAS	ANSRIS	2	150	23,310	ACSR, HAWK	477 MCM	645	800	800	830	830	0,2136	0,4048	2,7970	12,5127	163,8483	28,8120	225,0000	0,1192	0,0556	0,1054	0,0399	103,9200	
1	KAPALAS	PMRONS	1	150	58,590	ACSR, HAWK	135 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	2,6318	7,6187	55,3248	8,0604	225,0000	0,0633	0,0293	0,0991	0,0125	167,5710
1	KAPALAS	PMRONS	2	150	58,590	ACSR, HAWK	135 MCM	645	800	800	830	830	0,1370	0,3966	2,8800	2,6318	7,6187	55,3248	8,0604	225,0000	0,0633	0,0293	0,0991	0,0125	167,5710
1	KAPALAS	GNVARS	1	150	19,210	ACSR, HAWK	477 MCM	645	400	400	400	400	0,2136	0,4080	2,7970	6,8204	12,6113	85,4971	14,2350	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	KAPALAS	GNVARS	2	150	19,210	ACSR, HAWK	477 MCM	645	400	400	400	400	0,2136	0,4080	2,7970	6,8204	12,6113	85,4971	14,2350	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	KAPALAS	NSDUAS	1	150	30,910	ACSR, PARTRIDGE	135 mm ²	645	800	800	1250	1250	0,1370	0,3970	2,8800	2,3691	6,8363	49,5936	7,2319	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	KAPALAS	NSDUAS	2	150	30,910	ACSR, PARTRIDGE	135 mm ²	645	800	800	1250	1250	0,1370	0,3970	2,8800	2,3691	6,8363	49,5936	7,2319	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	KAPALAS	PNGANS	1	150	17,220	TACSR	135 mm ²	400	800	800	1250	1250	0,1370	0,4084	2,7970	8,1531	15,5886	106,7615	17,5920	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	KAPALAS	PNGANS	2	150	17,220	TACSR	135 mm ²	400	800	800	1250	1250	0,1370	0,4084	2,7970	8,1531	15,5886	106,7615	17,5920	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	GNVARS	BRITIS	1	150	38,170	TACSR	135 mm ²	400	800	800	1250	1250	0,1370	0,4084	2,7970	8,1531	15,5886	106,7615	17,5920	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	GNVARS	BRITIS	2	150	38,170	TACSR	135 mm ²	400	800	800	1250	1250	0,1370	0,4084	2,7970	8,1531	15,5886	106,7615	17,5920	225,0000	0,0782	0,0321	0,0705	0,0112	167,5710
1	GNVARS	SAUDURS	1	150	16,900	ACSR, PARTRIDGE	477 MCM	400	200	1000	830	830	0,2136	0,4080	2,7970	3,5244	6,7320	48,1695	7,5988	225,0000	0,0338	0,0157	0,0299	0,0104	103,9200
1	GNVARS	SAUDURS	2	150	16,900	ACSR, PARTRIDGE	477 MCM	400	200	1000	830	830	0,2136	0,4080	2,7970	3,5244	6,7320	48,1695	7,5988	225,0000	0,0338	0,0157	0,0299	0,0104	103,9200
1	GNVARS	AMPPLAS	1	150	33,760	ACSR, HAWK	477 MCM	645	400	1000	830	830	0,1370	0,3966	2,8800	4,6251	13,3892	97,2288	14,1656	225,0000	0,0630	0,0206	0,0595	0,0219	167,5710
1	GNVARS	AMPPLAS	2	150	33,760	ACSR, HAWK	477 MCM	645	400	1000	830	830	0,1370	0,3966	2,8800	4,6251	13,3892	97,2288	14,1656	225,0000	0,0630	0,0206	0,0595	0,0219	167,5710
1	SAUDURS	AMPPLAS	1	150	33,760	ACSR, HAWK	477 MCM	400	400	1000	830	830	0,2136	0,4080	2,7970	1,6533	3,1679	21,6488	3,5646	225,0000	0,0159	0,0073	0,0140	0,0048	103,9200
1	SAUDURS	AMPPLAS	2	150	33,760	ACSR, HAWK	477 MCM	400	400	1000	830	830	0,2136	0,4080	2,7970	1,6533	3,1679	21,6488	3,5646	225,0000	0,0159	0,0073	0,0140	0,0048	103,9200
1	SAUDURS	PGRANS	1	150	7,740	ACSR, PARTRIDGE	135 mm ²	400	400	1000	830	830	0,2136	0,4080	2,7970	1,6554	3,1620	21,6788	3,5691	225,0000	0,0159	0,0074	0,0141	0,0048	103,9200
1	SAUDURS	PGRANS	2	150	7,740	ACSR, HAWK	477 MCM	400	1000	1000	830	830	0,2136	0,4080	2,7970	1,6554	3,1620	21,6788	3,5691	225,0000	0,0159	0,0074	0,0141	0,0048	103,9200

DATA KARAKTERISTIK TRANSMISI UPT & SRB

Jmlh Strukt	Dati	Ko	No. Strukt	Data Penggandar		I Nom (Amp)	Tap CT Dari	Tap WT Dari	R (Ohm/km)	X (Ohm/km)	B (Mg/km)	R (Ohm)	X (Ohm)	B (Ohm)	Z act (Ohm)	Z base (Ohm)	Z pu (Ohm)	R pu (Ohm)	X pu (Ohm)	B pu (Ohm)	Days (MVA)	
				Teg (kV)	Jarak (Km)																	Jenis
																						(1)-(1) x 1,732/1000
																						(1)-(1) x 1,732/1000

Sub Region Ball

1	NSDUAS	PMRONS	1	150	13.410	ACSR. PARTRIDGE	135 mm ²	400	200	830	830	0.2136	0.4084	2.7970	2.8944	5.4766	37.5078	6.1805	225.0000	0.0275	0.0127	0.0243	0.0084	103.9200
1	BRITIS	PMRONS	1	150	20.430	ACSR. PARTRIDGE	135 mm ²	400	800	830	830	0.2136	0.4084	2.7970	4.3938	8.3436	57.1427	9.4159	225.0000	0.0418	0.0194	0.0371	0.0128	103.9200
Pengembangan																								
1	KAPALS	PBIANS	1	150	17.220	ACSR. HAWK	477 mm ²	645	1000	830	830	0.1370	0.3966	2.8820	2.3591	7.0258	48.6280	7.4113	225.0000	0.0329	0.0165	0.0312	0.0112	167.5710
1	PBIANS	PSGRNS	1	150	7.400	ACSR. HAWK	477 mm ²	645	1000	830	830	0.1370	0.3966	2.8820	1.0138	3.0192	21.3268	3.1949	225.0000	0.0142	0.0045	0.0134	0.0048	167.5710
1	KAPALS	UBUDS	1	150	16.730	ACSR. PARTRIDGE	135 mm ²	400	800	1250	1250	0.2136	0.4080	2.7970	3.5735	6.6351	46.7938	7.5362	225.0000	0.0335	0.0159	0.0295	0.0105	103.9200
1	UBUDS	BRITIS	1	150	28.440	ACSR. PARTRIDGE	135 mm ²	400	800	1250	1250	0.2136	0.4080	2.7970	6.2884	11.8759	82.3437	13.2816	225.0000	0.0589	0.0279	0.0519	0.0185	103.9200
Informasi																								
1	KAPALS	UBUDS	1	150	22.500	ACSR. PARTRIDGE	135 mm ²	400	800	1250	1250	0.2136	0.4080	2.7970	4.8060	8.1800	62.9325	10.3620	225.0000	0.0461	0.0214	0.0408	0.0142	103.9200
1	UBUDS	BRITIS	1	150	25.670	ACSR. PARTRIDGE	135 mm ²	400	800	1250	1250	0.2136	0.4080	2.7970	5.4831	10.4734	71.7990	11.8218	225.0000	0.0525	0.0244	0.0465	0.0162	103.9200
Rencana Up-rating																								
1	KAPALS	UBUDS	1	150	22.500	TACSR	160 mm ²	734	400	830	830	0.2200	0.3900	2.9600	4.9500	8.7750	66.6000	10.0749	225.0000	0.0448	0.0220	0.0390	0.0150	190.6932
1	UBUDS	BRITIS	1	150	25.670	TACSR	160 mm ²	734	400	830	830	0.2200	0.3900	2.9600	5.6474	10.0113	75.9832	11.4943	225.0000	0.0511	0.0251	0.0445	0.0171	190.6932
1	BRITIS	PMRONS	1	150	20.430	TACSR	160 mm ²	734	800	200	830	0.2200	0.3900	2.9600	4.4946	7.9577	60.4728	9.1480	225.0000	0.0407	0.0200	0.0354	0.0138	190.6932
1	BRITIS	PMRONS	1	150	20.430	TACSR	160 mm ²	734	800	200	830	0.2200	0.3900	2.9600	4.4948	7.9577	60.4728	9.1480	225.0000	0.0407	0.0200	0.0354	0.0138	190.6932
1	KAPALS	PMRONS	1	150	58.580	TACSR	160 mm ²	734	400	830	830	0.2200	0.3900	2.9600	12.8876	22.8462	173.3968	28.2305	225.0000	0.1166	0.0573	0.1015	0.0390	190.6932
1	KAPALS	BRITIS	1	150	38.170	TACSR	160 mm ²	734	800	800	1250	0.2200	0.3900	2.9600	8.3974	14.8863	112.9832	17.0915	225.0000	0.0760	0.0373	0.0662	0.0240	190.6932

```

unit uNewtonRaphson;

interface

uses uUtils, uComplex, uLoadflow, uMatrix;

procedure NewtonRaphson(var rBus:TBusArr1;
    var rBranch:TBranchArr1;
    var rParamLF:TParamLF); overload;
procedure NewtonRaphson(var rParamLF:TParamLF;
    var rV, rSg, rSL:CxArr1;
    var rCap:dArr1;
    var rTypBus:iArr1;
    var rZ, rTp, rAlir, rArus:CxArr2;
    var rLc, rTr:dArr2); overload;

implementation

function MismatchDaya(var rNbus, rNgen:integer;
    var rV, rSg, rSL:CxArr1;
    var rTyp:iArr1;
    var rY:CxArr2):dArr1;
var i, j, Ns, Np, Nq:integer;
    sumP, sumQ:double;
begin
    Ns:=rNbus-1+rNbus-rNgen-1;
    SetLength(result, Ns);
    Np:=-1;
    Nq:=rNbus-2;
    for i:=0 to rNbus-1 do
    begin
        if rTyp[i]<>1 then
        begin
            inc(Np);
            sumP:=0.0;
            for j:=0 to rNbus-1 do
            begin
                //sumP:=sumP+Ui*Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
                sumP:=sumP+rV[i].real*rV[j].real*(rY[i, j].real*
                    cos(rV[i].imag-rV[j].imag)+rY[i, j].imag*
                    sin(rV[i].imag-rV[j].imag));
            end;
            result[Np]:=rSg[i].real-rSL[i].real-sumP;
        end;
        if rTyp[i]=3 then
        begin
            inc(Nq);
            sumQ:=0.0;
            for j:=0 to rNbus-1 do
            begin
                //sumQ:=sumQ+Ui*Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
                sumQ:=sumQ+rV[i].real*rV[j].real*(rY[i, j].real*

```



```

                sin(rV[i].imag-rV[j].imag)-rY[i,j].imag*
                cos(rV[i].imag-rV[j].imag));
        end;
        result[Nq]:=rSg[i].imag-rSL[i].imag-sumQ;
    end;
end;
end;
end;

function Jaqobian(var rNbus,rNgen:integer;
    var rV:CxArr1;
    var rTyp:iArr1;
    var rY:CxArr2):dArr2;
var i,j,k,row,col:integer;
    sum,Pj,Qj:double;
begin
    row:=rNbus-1+rNbus-rNgen-1;
    SetLength(result,row,row);
    //Pembentukan Jaqobian H dP/d0
    row:=-1;
    for i:=0 to rNbus-1 do
    begin
        if rTyp[i]<>1 then
        begin
            inc(row);
            col:=-1;
            for j:=0 to rNbus-1 do
            begin
                if rTyp[j]<>1 then
                begin
                    inc(col);
                    if j=i then
                    begin
                        sum:=0.0;
                        for k:=0 to rNbus-1 do
                        begin
                            //sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);
                            sum:=sum+((rY[j,k].real*sin(rV[j].imag-rV[k].imag)-
                                rY[j,k].imag*cos(rV[j].imag-rV[k].imag))*
                                rV[k].real);
                        end;
                        //Qj:=sum*Uj;
                        Qj:=sum*rV[j].real;
                        //result[row,col]:=-Qj-Bij*sqr(Ui);
                        result[row,col]:=-Qj-rY[i,j].imag*sqr(rV[i].real);
                    end
                    else
                    begin
                        //result[row,col]:=Ui*Uj*(Gij*sin(di-dj)-Bij*cos(di-
dj));
                        result[row,col]:=rV[i].real*rV[j].real*(rY[i,j].real*
                            sin(rV[i].imag-rV[j].imag)-
rY[i,j].imag*

```

```

                                cos(rV[i].imag-rV[j].imag));
        end;
    end;
end;
end;
end;
//Pembentukan Matrik N dP/dV
row:=-1;
for i:=0 to rNbus-1 do
begin
    if rTyp[i]<>1 then
    begin
        inc(row);
        col:=rNbus-2;
        for j:=0 to rNbus-1 do
        begin
            if rTyp[j]=3 then
            begin
                inc(col);
                if j=i then
                begin
                    sum:=0.0;
                    for k:=0 to rNbus-1 do
                    begin
                        //sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
                        sum:=sum+((rY[j,k].real*cos(rV[j].imag-rV[k].imag)+
                            rY[j,k].imag*sin(rV[j].imag-rV[k].imag))*
                            rV[k].real);
                    end;
                    //Pj:=sum*Uj;
                    Pj:=sum*rV[j].real;
                    //result[row,col]:=Pj+Gij*Ui;
                    result[row,col]:=Pj+rY[i,j].real*sqr(rV[i].real);
                end
            else
            begin
                //result[row,col]:=Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
                result[row,col]:=rV[i].real*rV[j].real*(rY[i,j].real*
                    cos(rV[i].imag-
rV[j].imag)+rY[i,j].imag*
                                sin(rV[i].imag-rV[j].imag));
            end;
        end;
    end;
end;
end;
end;
//Pembentukan Jaqobian M dQ/d0
row:=rNbus-2;
for i:=0 to rNbus-1 do
begin
    if rTyp[i]=3 then
    begin

```

```

inc(row);
col:=-1;
for j:=0 to rNbus-1 do
begin
  if rTyp[j]<>1 then
  begin
    inc(col);
    if j=i then
    begin
      sum:=0;
      for k:=0 to rNbus-1 do
      begin
        //sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
        sum:=sum+((rY[j,k].real*cos(rV[j].imag-rV[k].imag)+
          rY[j,k].imag*sin(rV[j].imag-rV[k].imag))*
          rV[k].real);
      end;
      //Pj:=sum*Ui;
      Pj:=sum*rV[i].real;
      //result[row,col]:=Pj-Gij*sqr(Ui);
      result[row,col]:=Pj-rY[i,j].real*sqr(rV[i].real);
    end
  else
  begin
    //result[row,col]:=-Ui*Uj*(Gij*cos(di-dj)+Bij*sin(di-
dj));
    result[row,col]:=-rV[i].real*rV[j].real*(rY[i,j].real*
      cos(rV[i].imag-
rV[j].imag)+rY[i,j].imag*
      sin(rV[i].imag-rV[j].imag));
  end;
end;
end;
end;
end;
end;
//Pembentukan Jaqobian L dQ/dV
row:=rNbus-2;
for i:=0 to rNbus-1 do
begin
  if rTyp[i]=3 then
  begin
    inc(row);
    col:=rNbus-2;
    for j:=0 to rNbus-1 do
    begin
      if rTyp[j]=3 then
      begin
        inc(col);
        if j=i then
        begin
          sum:=0.0;
          for k:=0 to rNbus-1 do

```

```

begin
  //sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);
  sum:=sum+((rY[j,k].real*sin(rV[j].imag-rV[k].imag)-
            rY[j,k].imag*cos(rV[j].imag-rV[k].imag))*
            rV[k].real);
end;
//Qj:=sum*Ui;
Qj:=sum*rV[i].real;
//result[row,col]:=Qj-Bij*Ui;
result[row,col]:=Qj-rY[i,j].imag*sqr(rV[i].real);
end
else
begin
  //result[row,col]:=Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
  result[row,col]:=rV[i].real*rV[j].real*(rY[i,j].real*
            sin(rV[i].imag-rV[j].imag)-
rY[i,j].imag*
            cos(rV[i].imag-rV[j].imag));
end;
end;
end;
end;
end;
end;
end;

```

```

procedure UpdateTegangan(var rNbus:integer;
  var rdS:dArr1;
  var rJaq:dArr2;
  var rTyp:iArr1;
  var rV:CxArr1);
var i,Np,Nq:integer;
  dV:double;
  YE:dArr1;
begin
  YE:=EllGauss(rJaq,rdS);
  Np:=-1;
  Nq:=rNbus-2;
  for i:=0 to rNbus-1 do
  begin
    if rTyp[i]<>1 then
    begin
      inc(Np);
      rV[i].imag:=rV[i].imag+YE[Np];
    end;
    if rTyp[i]=3 then
    begin
      inc(Nq);
      dV:=YE[Nq]*rV[i].real;
      rV[i].real:=rV[i].real+dV;
      //rV[i].real:=rV[i].real+YE[Nq];
    end;
  end;
end;

```

```

end;

procedure NewtonRaphson(var rBus:TBusArr1;
    var rBranch:TBranchArr1;
    var rParamLF:TParamLF);
var i,Nbus,Nsal,Ngen:integer;
    max:double;
    V,Sg,SL:CxArr1;
    Cap,dS:dArr1;
    TypBus:iArr1;
    Z,Tp,Y,Alir,Arus:CxArr2;
    Lc,Tr,mJaq:dArr2;
begin
    DecodeCommDataToLFData(rBus,Nbus,Nsal,V,Sg,SL,Cap,TypBus,
        rBranch,Z,Tp,Lc,Tr);
    DecodeData(rParamLF,Nbus,V,Sg,SL,Cap,TypBus,Z,Tp,Lc,Tr);
    Ngen:=FindSumGen(Nbus,TypBus);
    Admitansi(Nbus,Z,Tp,Lc,Tr,Cap,Y);
    rParamLF.Iterasi:=0;
    for i:=0 to rParamLF.MaxIterasi-1 do
        begin
            dS:=MismatchDaya(Nbus,Ngen,V,Sg,SL,TypBus,Y);
            max:=MaxDataArray(dS);
            if max<=rParamLF.Toleransi then break;
            mJaq:=Jacobian(Nbus,Ngen,V,TypBus,Y);
            UpdateTegangan(Nbus,dS,mJaq,TypBus,V);
            inc(rParamLF.Iterasi);
        end;
        V:=PolarToRec(V);
        AliranDaya(Nbus,V,Y,Lc,Alir);
        ArusBranch(Nbus,V,Lc,Y,Arus);
        DayaGen(Nbus,V,SL,Y,TypBus,Sg);
        DayaSlack(Nbus,Alir,TypBus,SL,Sg);
        V:=RecToPolar(V);
        UpdateAkhir(Nbus,Nsal,rParamLF,V,Sg,SL,Alir,Arus,rBus,rBranch);
    end;

procedure NewtonRaphson(var rParamLF:TParamLF;
    var rV,rSg,rSL:CxArr1;
    var rCap:dArr1;
    var rTypBus:iArr1;
    var rZ,rTp,rAlir,rArus:CxArr2;
    var rLc,rTr:dArr2);overload;
var i,Nbus,Ngen:integer;
    max:double;
    dS:dArr1;
    Y:CxArr2;
    mJaq:dArr2;
begin
    Nbus:=high(rV)+1;
    DecodeData(rParamLF,Nbus,rV,rSg,rSL,rCap,rTypBus,rZ,rTp,rLc,rTr);
    Ngen:=FindSumGen(Nbus,rTypBus);

```

```

Admitansi (Nbus, rZ, rTp, rLc, rTr, rCap, Y);
rParamLF.Iterasi:=0;
for i:=0 to rParamLF.MaxIterasi-1 do
begin
  dS:=MismatchDaya (Nbus, Ngen, rV, rSg, rSL, rTypBus, Y);
  max:=MaxDataArray (dS);
  if max<=rParamLF.Toleransi then break;
  mJaq:=Jacobian (Nbus, Ngen, rV, rTypBus, Y);
  UpdateTegangan (Nbus, dS, mJaq, rTypBus, rV);
  inc (rParamLF.Iterasi);
end;
rV:=PolarToRec (rV);
AliranDaya (Nbus, rV, Y, rLc, rAlir);
ArusBranch (Nbus, rV, rLc, Y, rArus);
DayaGen (Nbus, rV, rSL, Y, rTypBus, rSg);
DayaSlack (Nbus, rAlir, rTypBus, rSL, rSg);
rV:=RecToPolar (rV);
UpdateAkhir (Nbus, rParamLF, rSg, rSL, rAlir, rArus);
end;

end.

```

```

unit uInputLF;

interface

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

type
  TfrmInputLF = class(TForm)
    PageControl1: TPageControl;
    TabSheet1: TTabSheet;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    edtNbus: TEdit;
    edtNsal: TEdit;
    edtVbase: TEdit;
    edtPbase: TEdit;
    cmbParam: TComboBox;
    cmbVKonst: TComboBox;
    cmbPkonst: TComboBox;
    TabSheet2: TTabSheet;
    fgBus: TStringGrid;
    TabSheet3: TTabSheet;
    fgBranch: TStringGrid;
    TabSheet4: TTabSheet;
    fgGen: TStringGrid;
    Panel1: TPanel;
    btnClose: TButton;
    btnNext: TButton;
    SaveDialog1: TSaveDialog;
    procedure btnCloseClick(Sender: TObject);
    procedure FormCreate(Sender: TObject);
    procedure edtNbusChange(Sender: TObject);
    procedure edtNsalChange(Sender: TObject);
    procedure cmbPkonstChange(Sender: TObject);
    procedure cmbParamChange(Sender: TObject);
    procedure btnNextClick(Sender: TObject);
  private
    { Private declarations }
  protected
    procedure ShowHasil;virtual;abstract;
  public
    { Public declarations }
  end;

var
  frmInputLF: TfrmInputLF;

implementation

{$R *.dfm}

procedure TfrmInputLF.btnCloseClick(Sender: TObject);
begin
  Close;

```

end;

```
procedure TfrmInputLF.FormCreate(Sender: TObject);
begin
```

```
    edtNbus.Text:='1';
    edtNsal.Text:='1';
    edtVbase.Text:='150';
    edtPbase.Text:='100';
    cmbVkonst.Text:='kV';
    cmbPkonst.Text:='MVA';
    cmbParam.Text:='pu';
    fgBus.Cells[0,0]:='Bus';
    fgBus.Cells[1,0]:='absV (pu)';
    fgBus.Cells[2,0]:='sudV (deg)';
    fgBus.Cells[3,0]:='Pg (MW)';
    fgBus.Cells[4,0]:='Qg (MVAR)';
    fgBus.Cells[5,0]:='PL (MW)';
    fgBus.Cells[6,0]:='QL (MVAR)';
    fgBus.Cells[7,0]:='Cap (pu)';
    fgBus.Cells[8,0]:='Type Bus';
    fgBranch.Cells[0,0] :='No';
    fgBranch.Cells[1,0] :='Dari';
    fgBranch.Cells[2,0] :='Ke';
    fgBranch.Cells[3,0] :='R (pu)';
    fgBranch.Cells[4,0] :='X (pu)';
    fgBranch.Cells[5,0] :='Lc (pu)';
    fgBranch.Cells[6,0] :='Tr';
    fgBranch.Cells[7,0] :='Tu';
    fgBranch.Cells[8,0] :='Su (deg)';
    fgBranch.Cells[9,0] :='Kap (MVA)';
    fgBranch.Cells[10,0]:='Length (m)';
    fgBranch.Cells[11,0]:='Diameter';
    fgGen.Cells[0,0]:='No';
    fgGen.Cells[1,0]:='Bus';
    fgGen.Cells[2,0]:='Qmin (MVAR)';
    fgGen.Cells[3,0]:='Qmax (MVAR)';
    fgGen.Cells[4,0]:='a2';
    fgGen.Cells[5,0]:='a1';
    fgGen.Cells[6,0]:='a0';
    fgGen.Cells[7,0]:='Fix Cost';
    fgGen.Cells[8,0]:='Var Cost';
    fgGen.Cells[9,0]:='Pmin (MW)';
    fgGen.Cells[10,0]:='Pmax (MW)';
    fgGen.Cells[11,0]:='Nama';
```

end;

```
procedure TfrmInputLF.edtNbusChange(Sender: TObject);
var i:integer;
```

```
begin
    if edtNbus.Text='' then
    begin
        fgBus.RowCount:=2;
    end
    else
    begin
        try
            fgBus.RowCount:=StrToInt(edtNbus.Text)+1;
            for i:=1 to StrToInt(edtNbus.Text) do
                begin
```



```

        fgBus.Cells[0,i]:=IntToStr(i);
        fgBus.Cells[1,i]:='1';
        fgBus.Cells[2,i]:='0';
        fgBus.Cells[3,i]:='0';
        fgBus.Cells[4,i]:='0';
        fgBus.Cells[5,i]:='0';
        fgBus.Cells[6,i]:='0';
        fgBus.Cells[7,i]:='0';
        fgBus.Cells[8,i]:='3';
    end;
    fgBus.Cells[8,1]:='1';
except
    raise Exception.Create('Tolong Masukkan angka!');
end;
end;
end;

procedure TfrmInputLF.edtNsalChange(Sender: TObject);
var i:integer;
begin
    if edtNsal.Text='' then
        begin
            fgBranch.RowCount:=2;
        end
    else
        begin
            try
                fgBranch.RowCount:=StrToInt(edtNsal.Text)+1;
                for i:=1 to StrToInt(edtNsal.Text) do
                    begin
                        fgBranch.Cells[0,i]:=IntToStr(i);
                        fgBranch.Cells[3,i]:='0';
                        fgBranch.Cells[4,i]:='0';
                        fgBranch.Cells[5,i]:='0';
                        fgBranch.Cells[6,i]:='0';
                        fgBranch.Cells[7,i]:='0';
                        fgBranch.Cells[8,i]:='0';
                        fgBranch.Cells[9,i]:='10000';
                        fgBranch.Cells[10,i]:='0';
                    end;
                except
                    raise Exception.Create('Tolong Masukkan angka!');
                end;
            end;
        end;
end;

procedure TfrmInputLF.cmbPKonstChange(Sender: TObject);
begin
    if cmbPKonst.Text='VA' then
        begin
            fgBus.Cells[3,0]:='Pg (W)';
            fgBus.Cells[4,0]:='Qg (VAR)';
            fgBus.Cells[5,0]:='PL (W)';
            fgBus.Cells[6,0]:='QL (VAR)';
            fgBranch.Cells[9,0] :='Kap (VA)';
            fgGen.Cells[9,0]:='Pmin (W)';
            fgGen.Cells[10,0]:='Pmax (W)';
        end
    else if cmbPKonst.Text='kVA' then

```

```

begin
  fgBus.Cells[3,0]:='Pg (kW)';
  fgBus.Cells[4,0]:='Qg (kVAR)';
  fgBus.Cells[5,0]:='PL (kW)';
  fgBus.Cells[6,0]:='QL (kVAR)';
  fgBranch.Cells[9,0] :='Kap (VA)';
  fgGen.Cells[9,0]:='Pmin (kW)';
  fgGen.Cells[10,0]:='Pmax (kW)';
end
else if cmbPKonst.Text='MVA' then
begin
  fgBus.Cells[3,0]:='Pg (MW)';
  fgBus.Cells[4,0]:='Qg (MVAR)';
  fgBus.Cells[5,0]:='PL (MW)';
  fgBus.Cells[6,0]:='QL (MVAR)';
  fgBranch.Cells[9,0] :='Kap (VA)';
  fgGen.Cells[9,0]:='Pmin (MW)';
  fgGen.Cells[10,0]:='Pmax (MW)';
end;
end;

procedure TfrmInputLF.cmbParamChange(Sender: TObject);
begin
  if cmbParam.Text='pu' then
  begin
    fgBranch.Cells[3,0]:='R (pu)';
    fgBranch.Cells[4,0]:='X (pu)';
    fgBranch.Cells[5,0]:='Lc (pu)';
    fgBus.Cells[7,0]:='Cap (pu)';
  end
  else if cmbParam.Text='ohm' then
  begin
    fgBranch.Cells[3,0]:='R (ohm)';
    fgBranch.Cells[4,0]:='X (ohm)';
    fgBranch.Cells[5,0]:='Lc (ohm)';
    fgBus.Cells[7,0]:='Cap (ohm)';
  end;
end;

procedure TfrmInputLF.btnNextClick(Sender: TObject);
var NamaFile,Nama:string;
    input:TextFile;
    NCable,Nbus,Nsal,Ngen,i,typ,dari,ke,param:integer;
    R,X,Lc,Tr,Tu,Su,Cap,CapSal,phi,Pmin,Pmax,Harga,length:double;
    VKonst,PKonst,Vbase,Pbase,absV,sudV,Pg,Qg,PL,QL:double;
    jam,menit,detik,mdetik:word;
    mulai,selesai,selang:TDateTime;
    load:string;
begin
  if btnNext.Caption='&Save' then
  begin
    Nbus:=StrToInt(edtNbus.Text);
    Nsal:=StrToInt(edtNsal.Text);
    Vbase:=StrToFloat(edtVbase.Text);
    VKonst:=1;
    if cmbVKonst.Text='V' then
    begin
      VKonst:=1;
    end
  end

```

```

else if cmbVKonst.Text='kV' then
begin
    VKonst:=1000;
end
else if cmbVKonst.Text='MV' then
begin
    VKonst:=1000000;
end;
Pbase:=StrToFloat(edtPbase.Text);
PKonst:=1;
if cmbPKonst.Text='VA' then
begin
    PKonst:=1;
end
else if cmbPKonst.Text='kVA' then
begin
    PKonst:=1000;
end
else if cmbPKonst.Text='MVA' then
begin
    PKonst:=1000000;
end;
Param:=1;
if cmbParam.Text='pu' then
begin
    Param:=1;
end
else if cmbParam.Text='ohm' then
begin
    Param:=2;
end;
try
    if SaveDialog1.Execute then
    begin
        NamaFile:=SaveDialog1.FileName;
        AssignFile(input,NamaFile+'.txt');
        Rewrite(input);
        Writeln(input,Nbus);
        Writeln(input,Nsal);
        Writeln(input,Vbase:6:2);
        Writeln(input,VKonst:7:0);
        Writeln(input,Pbase:6:2);
        Writeln(input,PKonst:7:0);
        Writeln(input,param);
        phi:=4*arctan(1);
        for i:=1 to Nbus do
        begin
            absV:=StrToFloat (fgBus.Cells[1,i]);
            sudV:=StrToFloat (fgBus.Cells[2,i])*180/phi;
            Pg:=StrToFloat (fgBus.Cells[3,i]);
            Qg:=StrToFloat (fgBus.Cells[4,i]);
            PL:=StrToFloat (fgBus.Cells[5,i]);
            QL:=StrToFloat (fgBus.Cells[6,i]);
            Cap:=StrToFloat (fgBus.Cells[7,i]);
            Typ:=StrToInt (fgBus.Cells[8,i]);
            Writeln(input,absV:7:5,' ',sudV:7:5,' ',Pg:9:3,' ',Qg:9:3,
            ' ',PL:9:3,' ',QL:9:3,' ',Cap:7:5,' ',Typ);
        end;
        for i:=1 to Nsal do

```

```

begin
  dari:=StrToInt (fgBranch.Cells[1,i]);
  ke:=StrToInt (fgBranch.Cells[2,i]);
  R:=StrToFloat (fgBranch.Cells[3,i]);
  X:=StrToFloat (fgBranch.Cells[4,i]);
  Lc:=StrToFloat (fgBranch.Cells[5,i]);
  Tr:=StrToFloat (fgBranch.Cells[6,i]);
  Tu:=StrToFloat (fgBranch.Cells[7,i]);
  Su:=StrToFloat (fgBranch.Cells[8,i]);
  CapSal:=StrToFloat (fgBranch.Cells[9,i]);
  //Length:=StrToFloat (fgBranch.Cells[21,i]);
  Writeln(input,dari,' ',ke,' ',R:7:5,' ',X:7:5,' ',
  Lc:7:5,' ',Tr:7:5,' ',Tu:7:5,' ',Su:7:5,' ',
  CapSal:7:2);
end;
Ngen:=0;
load:='3';
for i:=1 to Nbus do
begin
  if fgBus.Cells[8,i]<>load then
  begin
    inc(Ngen);
  end;
end;
Writeln(input,Ngen);
for i:=1 to Ngen do
begin
  dari:=StrToInt (fgGen.Cells[1,i]);
  R:=StrToFloat (fgGen.Cells[2,i]);
  X:=StrToFloat (fgGen.Cells[3,i]);
  Lc:=StrToFloat (fgGen.Cells[4,i]);
  Tr:=StrToFloat (fgGen.Cells[5,i]);
  Tu:=StrToFloat (fgGen.Cells[6,i]);
  Su:=StrToFloat (fgGen.Cells[7,i]);
  CapSal:=StrToFloat (fgGen.Cells[8,i]);
  Pmin:=StrToFloat (fgGen.Cells[9,i]);
  Pmax:=StrToFloat (fgGen.Cells[10,i]);
  Writeln(input,dari,' ',R:7:2,' ',X:7:2,' ',Lc:7:5,' ',
  Tr:7:5,' ',Tu:7:5,' ',Su:7:2,' ',CapSal:7:2,' ',
  Pmin:7:5,' ',Pmax:7:2);
end;
end;
CloseFile(input);
MessageDlg('File berhasil disimpan!',mtInformation,[mbOK],0);
except
  MessageDlg('Tolong dicek angka-angkanya kembali!',
  mtWarning,[mbOK],0);
end;
end
else if btnNext.Caption='&Next' then
begin
  ShowHasil;
end;
end;
end;
end.

```

```

unit uLoadflow;

interface

uses uUtils, uComplex;

type
  TParamBranch=(pbOhm, pbPu) ;

  TParamLF=record
    MaxIterasi, Iterasi:byte;
    Vbase, Vkonst, Pbase, Pkonst, Zbase, Ibase, Toleransi:double;
    ParamBranch:TParamBranch;
    SumGen, SumLoad, SumLoss:TCmplx;
  end;

  TBus=record
    absV, sudV, Pgen, Qgen, PL, QL, Cap:double;
    typeBus:byte;
  end;

  TBusArr1=array of TBus;

  TBranch=record
    dari, ke:integer;
    R, X, Lc, Tr, Tu, Su, KapSal, Length, JenisCable:double;
    Sij, Sji, Aij, Aji:TCmplx;
  end;

  TBranchArr1=array of TBranch;

  TGenLF=record
    nama:string;
    bus:integer;
    Qmin, Qmax, a2, a1, a0, FixCost, VarCost, Pmin, Pmax:double;
  end;

  TGenLFArr1=array of TGenLF;

  TSwith=record
    dari, ke:integer;
  end;

  TSwithArr1=array of TSwith;

function CalcCostGenLF(const rGen:TGenLF;
  const rPgen:double):double;
procedure DecodeCommDataToLFData(const rBus:TBusArr1;
  var rNbus, rNsal:integer;
  var rV, rSg, rSL:CxArr1;
  var rCap:dArr1;
  var rTypBus:iArr1;
  const rBranch:TBranchArr1;
  var rZ, rTp:CxArr2;
  var rLc, rTr:dArr2);
procedure DecodeData(var rParamLF:TParamLF;
  var rNbus:integer;

```

```

    var rV, rSg, rSL:CxArr1;
    var rCap:dArr1;
    var rTypBus:iArr1;
    var rZ, rTp:CxArr2;
    var rLc, rTr:dArr2);
function FindSumGen(const rNbus:integer;
const rTypBus:iArr1):integer;
procedure Admitansi(const rNbus:integer;
const rZ, rTp:CxArr2;
const rLc, rTr:dArr2;
const rCap:dArr1;
var rY:CxArr2);
function RecToPolar(const rData:CxArr1):CxArr1;
function PolarToRec(const rData:CxArr1):CxArr1;
function MaxDataArray(const rData:dArr1):double;
procedure AliranDaya(const rNbus:integer;const rV:CxArr1;
const rY:CxArr2;
const rLc:dArr2;
var rAlirS:CxArr2);
procedure DayaGen(const rNbus:integer;
const rV, rSL:CxArr1;
const rY:CxArr2;
const rTyp:iArr1;
var rSg:CxArr1);
procedure DayaSlack(const rNbus:integer;
const rAlirS:CxArr2;
const rTyp:iArr1;
const rSL:CxArr1;
var rSg:CxArr1);
procedure ArusBranch(const rNbus:integer;
const rV:CxArr1;
const rLc:dArr2;
const rY:CxArr2;
var rArus:CxArr2);
procedure UpdateAkhir(const rNbus, rNsal:integer;
var rParamLF:TParamLF;
const rV, rSg, rSL:CxArr1;
const rAlir, rArus:CxArr2;
var rBus:TBusArr1;
var rBranch:TBranchArr1);overload;
procedure UpdateAkhir(const rNbus:integer;
var rParamLF:TParamLF;
var rSg, rSL:CxArr1;
var rAlir, rArus:CxArr2);overload;
function FindVarControl(var rBus:TBusArr1):iArr1;overload;
function FindVarControl(var rSg:CxArr1;
var rTypeBus:iArr1):iArr1;overload;
function FindBatasControl(var rBus:TBusArr1;
var rParamLF:TParamLF):TBatasArr1;

var gBus:TBusArr1;
gBranch:TBranchArr1;
gParamLF:TParamLF;
gGenLF:TGenLFArr1;

```

implementation

```

function CalcCostGenLF(const rGen:TGenLF;
    const rPgen:double):double;
begin
    result:=0;
    if rPgen>0 then
    begin
        result:=rGen.a2*sqr(rPgen)+rGen.a1*rPgen+rGen.a0;
    end;
end;

procedure DecodeCommDataToLFData(const rBus:TBusArr1;
    var rNbus,rNsal:integer;
    var rV,rSg,rSL:CxArr1;
    var rCap:dArr1;
    var rTypBus:iArr1;
    const rBranch:TBranchArr1;
    var rZ,rTp:CxArr2;
    var rLc,rTr:dArr2);
var i,dari,ke:integer;
begin
    rNbus:=high(rBus)+1;
    rNsal:=high(rBranch)+1;
    SetLength(rV,rNbus);
    SetLength(rSg,rNbus);
    SetLength(rSL,rNbus);
    SetLength(rCap,rNbus);
    SetLength(rTypBus,rNbus);
    for i:=0 to rNbus-1 do
    begin
        rV[i].real:=rBus[i].absV;
        rV[i].imag:=rBus[i].sudV;
        rSg[i].real:=rBus[i].Pgen;
        rSg[i].imag:=rBus[i].Qgen;
        rSL[i].real:=rBus[i].PL;
        rSL[i].imag:=rBus[i].QL;
        rCap[i]:=rBus[i].Cap;
        rTypBus[i]:=rBus[i].typeBus;
    end;
    SetLength(rZ,rNbus,rNbus);
    SetLength(rTp,rNbus,rNbus);
    SetLength(rLc,rNbus,rNbus);
    SetLength(rTr,rNbus,rNbus);
    for i:=0 to rNsal-1 do
    begin
        dari:=rBranch[i].dari-1;
        ke:=rBranch[i].ke-1;
        rZ[dari,ke].real:=rBranch[i].R;
        rZ[dari,ke].imag:=rBranch[i].X;
        rLc[dari,ke]:=rBranch[i].Lc;
        rTr[dari,ke]:=rBranch[i].Tr;
        rTp[dari,ke].real:=rBranch[i].Tu;
        rTp[dari,ke].imag:=rBranch[i].Su;
    end;
end;

procedure DecodeData(var rParamLF:TParamLF;
    var rNbus:integer;

```

```

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

function FindSumGen(const rNbus:integer;
  const rTypBus:iArr1):integer;
var i:integer;
begin
  result:=0;
  for i:=0 to rNbus-1 do
  begin
    if rTypBus[i]=2 then
    begin
      inc(result);
    end;
  end;
end;

```



```

    end;
end;
end;

procedure Admitansi(const rNbus:integer;
    const rZ,rTp:CxArr2;
    const rLc,rTr:dArr2;
    const rCap:dArr1;
    var rY:CxArr2);
var i,j,k:integer;
    a,b:double;
    Lc:dArr2;
    CY:CxArr2;
    sum,Za,Ca:TCmplx;
begin
    SetLength(CY,rNbus,rNbus);
    SetLength(Lc,rNbus,rNbus);
    sum:=Cmplx(1.0,0.0);
    for i:=0 to rNbus-1 do
    begin
        for j:=0 to rNbus-1 do
        begin
            if rZ[i,j].imag<>0 then
            begin
                if rTr[i,j]<>0 then
                begin
                    a:=rTr[i,j];
                    CY[i,j]:=Divide(sum,rZ[i,j]);
                    Lc[i,j]:=(1/sqr(a)-1/a)*CY[i,j].imag;
                    Lc[j,i]:=(1-1/a)*CY[i,j].imag;
                    CY[i,j]:=Divide(CY[i,j],a);
                    CY[j,i]:=Cmplx(CY[i,j]);
                end
            else
            begin
                CY[i,j]:=Divide(sum,rZ[i,j]);
                CY[j,i]:=Cmplx(CY[i,j]);
                Lc[i,j]:=rLc[i,j];
                Lc[j,i]:=rLc[i,j];
            end;
        end;
    end;
end;
SetLength(rY,rNbus,rNbus);
for i:=0 to rNbus-1 do
begin
    for j:=0 to rNbus-1 do
    begin
        if j=i then
        begin
            rY[i,j]:=Cmplx(0.0,0.0);
            for k:=0 to rNbus-1 do
            begin
                rY[i,j].real:=rY[i,j].real+CY[i,k].real;
                rY[i,j].imag:=rY[i,j].imag+CY[i,k].imag+Lc[i,k];
            end;
        end
    end
end
end

```

```

else
begin
  rY[i,j].real:=-CY[i,j].real;
  rY[i,j].imag:=-CY[i,j].imag;
end;
end;
for i:=0 to rNbus-1 do
begin
  for j:=0 to rNbus-1 do
  begin
    if rTp[i,j].real<>0 then
    begin
      a:=rTp[i,j].real*cos(rTp[i,j].imag);
      b:=rTp[i,j].real*sin(rTp[i,j].imag);
      rY[i,i].real:=rY[i,i].real-CY[i,j].real;
      rY[i,i].imag:=rY[i,i].imag-CY[i,j].imag-rLc[i,j];
      rY[i,i].real:=rY[i,i].real+CY[i,j].real/(sqr(a)+sqr(b));
      rY[i,i].imag:=rY[i,i].imag+CY[i,j].imag/(sqr(a)+sqr(b))+rLc[i,j];
      Za:=Cmplx(CY[i,j]);
      Za:=Negative(Za);
      Ca:=Cmplx(a,b);
      sum:=Divide(Za,Ca);
      rY[j,i]:=Cmplx(sum);
      Ca:=Conjugate(Ca);
      sum:=Divide(Za,Ca);
      rY[i,j]:=Cmplx(sum);
      {sUpfc:=TComplex.Create(aUpfc[i,j].tap*cos(aUpfc[i,j].sudut),
        aUpfc[i,j].tap*sin(aUpfc[i,j].sudut));
      CLc:=TComplex.Create(0.0,aLc[i,j]);
      result[i,i]:=result[i,i]-Cx[i,j]-CLc;
      result[i,i]:=result[i,i]+Cx[i,j]/sqr(sUpfc.Abs)+CLc;
      result[i,j]:=-Cx[i,j]/conj(sUpfc);
      result[j,i]:=-Cx[i,j]/sUpfc;
      CLc.Free;
      sUpfc.Free;}
    end;
  end;
end;
for i:=0 to rNbus-1 do
begin
  if rCap[i]<>0 then
  begin
    rY[i,i].imag:=rY[i,i].imag+rCap[i];
  end;
end;
end;

function RecToPolar(const rData:CxArr1):CxArr1;
var i,NData:integer;
    abs,sud:double;
begin
  NData:=high(rData)+1;
  SetLength(result,NData);
  for i:=0 to NData-1 do
  begin
    abs:=getAbs(rData[i]);

```

```

        sud:=getAngleRad(rData[i]);
        result[i]:=Cmplx(abs, sud);
    end;
end;

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

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

procedure AliranDaya(const rNbus:integer;const rV:CxArr1;
    const rY:CxArr2;
    const rLc:dArr2;
    var rAlirS:CxArr2);
var i,j:integer;
    Lca,tmp1,tmp2:TCmplx;
begin
    SetLength(rAlirS,rNbus,rNbus);
    for i:=0 to rNbus-1 do
    begin
        for j:=0 to rNbus-1 do
        begin
            rAlirS[i,j].real:=0.0;
            rAlirS[i,j].imag:=0.0;
            if j<>i then
            begin
                if rY[i,j].imag<>0 then
                begin
                    Lca:=Cmplx(0.0,rLc[i,j]);
                    tmp1:=Multiply(Multiply(Conjugate(rV[i]),Subtract(rV[i],rV[j])),
                        Negative(rY[i,j]));
                    tmp2:=Multiply(Multiply(Conjugate(rV[i]),rV[i]),Lca);
                    rAlirS[i,j]:=Conjugate(Add(tmp1,tmp2));
                end;
            end;
        end;
    end;
end;

```

```

        //result[i,j]:=conj(aV[i])*(aV[i]-aV[j])*(-aY[i,j])+
        //          conj(aV[i])*aV[i]*dLc;
        //result[i,j]:=conj(result[i,j]);
    end;
end;
end;
end;
end;

procedure DayaGen(const rNbus:integer;
    const rV,rSL:CxArr1;
    const rY:CxArr2;
    const rTyp:iArr1;
    var rSg:CxArr1);
var i,j:integer;
    sum:double;
begin
    for i:=0 to rNbus-1 do
        begin
            sum:=0.0;
            if rTyp[i]=2 then
                begin
                    for j:=0 to rNbus-1 do
                        begin
                            //sum:=sum+(Ej*Gij+Fj*-Bij)-Ei*(Fj*Gij-Ej*-Bij);
                            sum:=sum+(rV[i].imag*(rV[j].real*rY[i,j].real+rV[j].imag*-
rY[i,j].imag)-
                                rV[i].real*(rV[j].imag*rY[i,j].real-rV[j].real*-
rY[i,j].imag));
                            end;
                            rSg[i].imag:=sum+rSL[i].imag;
                        end;
                    end;
                end;
            end;
        end;
end;

procedure DayaSlack(const rNbus:integer;
    const rAlirS:CxArr2;
    const rTyp:iArr1;
    const rSL:CxArr1;
    var rSg:CxArr1);
var i,j:integer;
    sumP,sumQ:double;
begin
    for i:=0 to rNbus-1 do
        begin
            if rTyp[i]=1 then
                begin
                    sumP:=0.0;
                    sumQ:=0.0;
                    for j:=0 to rNbus-1 do
                        begin
                            if rAlirS[i,j].imag<>0 then
                                begin
                                    sumP:=sumP+rAlirS[i,j].real;
                                    sumQ:=sumQ+rAlirS[i,j].imag;
                                end;
                            end;
                        end;
                    end;
                end;
            end;
        end;
end;

```

```

        rSg[i].real:=sumP+rSL[i].real;
        rSg[i].imag:=sumQ+rSL[i].imag;
    end;
end;
end;

procedure ArusBranch(const rNbus:integer;
    const rV:CxArr1;
    const rLc:dArr2;
    const rY:CxArr2;
    var rArus:CxArr2);
var i,j:integer;
    xLc,tmp1,tmp2:TCmplx;
begin
    SetLength(rArus,rNbus,rNbus);
    SetLength(rArus,rNbus,rNbus);
    for i:=0 to rNbus-1 do
    begin
        for j:=0 to rNbus-1 do
        begin
            if rY[i,j].imag<>0 then
            begin
                xLc:=Cmplx(0,rLc[i,j]);
                tmp1:=Multiply(Subtract(rV[i],rV[j]),Negative(rY[i,j]));
                tmp2:=Multiply(rV[i],xLc);
                rArus[i,j]:=Conjugate(Add(tmp1,tmp2));
                //result[i,j]:=(aV[i]-aV[j])*(-aY[i,j])+aV[i]*xLc;
                //result[i,j]:=Conj(result[i,j]);
            end
            else
            begin
                rArus[i,j].real:=0.0;
                rArus[i,j].imag:=0.0;
            end;
        end;
    end;
end;
end;

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

```

```

    rParamLF.SumLoad:=Add(rParamLF.SumLoad,rSL[i]);
end;
rParamLF.SumGen:=Multiply(rParamLF.SumGen,rParamLF.Pbase);
rParamLF.SumLoad:=Multiply(rParamLF.SumLoad,rParamLF.Pbase);
rParamLF.SumLoss:=Subtract(rParamLF.SumGen,rParamLF.SumLoad);
for i:=0 to rNsal-1 do
begin
    dari:=rBranch[i].dari-1;
    ke:=rBranch[i].ke-1;
    rBranch[i].Sij:=Multiply(Cmplx(rAlir[dari,ke]),rParamLF.Pbase);
    rBranch[i].Sji:=Multiply(Cmplx(rAlir[ke,dari]),rParamLF.Pbase);
    rBranch[i].Aij:=Multiply(Cmplx(rArus[dari,ke]),rParamLF.Ibase);
    rBranch[i].Aji:=Multiply(Cmplx(rArus[ke,dari]),rParamLF.Ibase);
end;
end;

procedure UpdateAakhir(const rNbus:integer;
    var rParamLF:TParamLF;
    var rSg,rSL:CxArr1;
    var rAlir,rArus:CxArr2);
var i,j:integer;
begin
    rParamLF.SumGen:=Cmplx(0.0,0.0);
    rParamLF.SumLoad:=Cmplx(0.0,0.0);
    for i:=0 to rNbus-1 do
    begin
        rSg[i].real:=rSg[i].real*rParamLF.Pbase;
        rSg[i].imag:=rSg[i].imag*rParamLF.Pbase;
        rSL[i].real:=rSL[i].real*rParamLF.Pbase;
        rSL[i].imag:=rSL[i].imag*rParamLF.Pbase;
        rParamLF.SumGen:=Add(rParamLF.SumGen,rSg[i]);
        rParamLF.SumLoad:=Add(rParamLF.SumLoad,rSL[i]);
    end;
    rParamLF.SumLoss:=Subtract(rParamLF.SumGen,rParamLF.SumLoad);
    for i:=0 to rNbus-1 do
    begin
        for j:=0 to rNbus-1 do
        begin
            if rAlir[i,j].imag<>0 then
            begin
                rAlir[i,j]:=Multiply(rAlir[i,j],rParamLF.Pbase);
                rArus[i,j]:=Multiply(rArus[i,j],rParamLF.Ibase);
            end;
        end;
    end;
end;

function FindVarControl(var rBus:TBusArr1):iArr1;
var i,Nbus,sa:integer;
begin
    Nbus:=high(rBus)+1;
    sa:=0;
    for i:=0 to Nbus-1 do
    begin
        if rBus[i].typeBus<>3 then
        begin
            inc(sa);
        end;
    end;
end;

```

```

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

function FindVarControl(var rSg:CxArr1;
    var rTypeBus:iArr1):iArr1;
var i,Nbus,sa:integer;
begin
    Nbus:=high(rSg)+1;
    sa:=0;
    for i:=0 to Nbus-1 do
    begin
        if rTypeBus[i]<>3 then
        begin
            inc(sa);
        end;
    end;
    for i:=0 to Nbus-1 do
    begin
        if rTypeBus[i]=3 then
        begin
            if rSg[i].imag<>0 then
            begin
                inc(sa);
            end;
        end;
    end;
end;
end;

```

```

end;
SetLength(result, sa);
sa:=0;
for i:=0 to Nbus-1 do
begin
  if rTypeBus[i]<>3 then
  begin
    result[sa]:=i;
    inc(sa);
  end;
end;
for i:=0 to Nbus-1 do
begin
  if rTypeBus[i]=3 then
  begin
    if rSg[i].imag<>0 then
    begin
      result[sa]:=i;
      inc(sa);
    end;
  end;
end;
end;

function FindBatasControl(var rBus:TBusArr1;
  var rParamLF:TParamLF):TBatasArr1;
var i,Nbus,sa:integer;
begin
  Nbus:=high(rBus)+1;
  sa:=0;
  for i:=0 to Nbus-1 do
  begin
    if rBus[i].typeBus<>3 then
    begin
      inc(sa);
    end;
  end;
  for i:=0 to Nbus-1 do
  begin
    if rBus[i].typeBus=3 then
    begin
      if rBus[i].Qgen<>0 then
      begin
        inc(sa);
      end;
    end;
  end;
  SetLength(result, sa);
  sa:=0;
  for i:=0 to Nbus-1 do
  begin
    if rBus[i].typeBus<>3 then
    begin
      result[sa].min:=0.95;
      result[sa].max:=1.05;
      inc(sa);
    end;
  end;
end;

```



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

```

unit uObjFunc;

interface

uses uUtils, uLoadflow, uNewtonRaphson, SysUtils, uTypDatGA;

type
  TObjFunc=class
  private
    FNCap, FCountCap:integer;
    FCostCap, FKa:double;
    FBatasV:TBatas;
    FLamdaV, FLamdaCap:double;
    function getPinV(const rLBus:TBusArr1):double;
    function getNCap(const rChrom:TChromCap):integer;
    function UpdateDataBeban(const rChrom:TChromCap):TBusArr1;
    function HitungCostCap(const rChrom:TChromCap):double;
  public
    constructor Create(const rCostCap,rKa,rLamdaV,rLamdaCap:double;
                      const rCountCap,rNCap:integer;
                      const rBatasV:TBatas);
    function doHitung(const rChrom:TChromCap):double;
    procedure doHitungAwal(var rPloss,rCostCap:double);
    procedure doHitungAkhir(const rChrom:TChromCap;
                           var rLBus:TBusArr1;
                           var rPloss,rCostCap:double;
                           var rPosCap:dArr2);
    property NCap:integer read FNCap write FNCap;
    property CostCap:double read FCostCap write FCostCap;
  end;

var gObjFunc:TObjFunc;

implementation

{ TObjFunc }

constructor TObjFunc.Create(const
rCostCap,rKa,rLamdaV,rLamdaCap:double;
                           const rCountCap,rNCap:integer;
                           const rBatasV:TBatas);
begin
  inherited Create;
  FCostCap:=rCostCap;
  FKa:=rKa;
  FLamdaV:=rLamdaV;
  FLamdaCap:=rLamdaCap;
  FCountCap:=rCountCap;
  FNCap:=rNCap;
  FBatasV.min:=rBatasV.min;
  FBatasV.max:=rBatasV.max;
end;

//data processing
function TObjFunc.getPinV(const rLBus:TBusArr1):double;
var i,Nbus:integer;
begin

```

```

Nbus:=high(rLBus)+1;
result:=0;
for i:=0 to Nbus-1 do
begin
  if rLBus[i].absV>FBatasV.max then
  begin
    result:=result+(rLBus[i].absV-FBatasV.max);
  end;
  if rLBus[i].absV<FBatasV.min then
  begin
    result:=result+(FBatasV.min-rLBus[i].absV);
  end;
end;
end;

function TObjFunc.getNcap(const rChrom:TChromCap):integer;
var i,Nbus,Count:integer;
begin
  Count:=0;
  Nbus:=high(gBus)+1;
  for i:=0 to Nbus-1 do
  begin
    if rChrom[i+1].bus=true then
    begin
      inc(Count);
    end;
  end;
  result:=0;
  if Count>FCountCap then
  begin
    result:=Count-FCountCap;
  end;
end;

function TObjFunc.UpdateDataBeban(const rChrom:TChromCap):TBusArr1;
var i,Nbus:integer;
begin
  Nbus:=high(gBus)+1;
  SetLength(result,Nbus);
  for i:=0 to Nbus-1 do
  begin
    result[i].absV:=gBus[i].absV;
    result[i].sudV:=gBus[i].sudV;
    result[i].Pgen:=gBus[i].Pgen;
    result[i].Qgen:=gBus[i].Qgen;
    result[i].PL:=gBus[i].PL;
    result[i].QL:=gBus[i].QL;
    result[i].Cap:=gBus[i].Cap;
    result[i].typeBus:=gBus[i].typeBus;
    if rChrom[i+1].bus=true then
    begin
      result[i].QL:=result[i].QL-rChrom[i+1].cap;
    end;
  end;
end;

function TObjFunc.HitungCostCap(const rChrom:TChromCap):double;

```

```

var i,Nbus:integer;
begin
  Nbus:=high(gBus)+1;
  result:=0;
  for i:=0 to Nbus-1 do
  begin
    if rChrom[i+1].bus=true then
    begin
      result:=result+(rChrom[i+1].cap*FCostCap);
    end;
  end;
end;

function TObjFunc.doHitung(const rChrom:TChromCap):double;
var Ploss, CostCap, pinV, a, b:double;
    Count:integer;
    LBus:TBusArr1;
begin
  LBus:=UpdateDataBeban(rChrom);
  Count:=getNCap(rChrom);
  NewtonRaphson(LBus, gBranch, gParamLF);
  Ploss:=gParamLF.SumLoss.real;
  CostCap:=HitungCostCap(rChrom);
  pinV:=getPinV(LBus);
  a:=1;
  b:=0.000001;
  if gParamLF.Iterasi=gParamLF.MaxIterasi then
  begin
    result:=0;
  end
  else
  begin
    result:=FKa/(a*Ploss+b*CostCap+FLamdaV*pinV+FLamdaCap*Count);
  end;
end;

procedure TObjFunc.doHitungAwal(var rPloss, rCostCap:double);
begin
  NewtonRaphson(gBus, gBranch, gParamLF);
  rPloss:=gParamLF.SumLoss.real;
  rCostCap:=0;
end;

procedure TObjFunc.doHitungAkhir(const rChrom:TChromCap;
  var rLBus:TBusArr1;
  var rPloss, rCostCap:double;
  var rPosCap:dArr2);
var Ploss, CostCap:double;
    i, sa, Nbus:integer;
begin
  Nbus:=high(gBus)+1;
  rLBus:=UpdateDataBeban(rChrom);
  NewtonRaphson(rLBus, gBranch, gParamLF);
  rPloss:=gParamLF.SumLoss.real;
  rCostCap:=HitungCostCap(rChrom);
  sa:=0;
  for i:=0 to Nbus-1 do

```

```
begin
  if rChrom[i+1].bus=true then
    begin
      inc(sa);
    end;
  end;
  SetLength(rPosCap, sa, 2);
  sa:=0;
  for i:=0 to Nbus-1 do
    begin
      if rChrom[i+1].bus=true then
        begin
          rPosCap[sa,0]:=i;
          rPosCap[sa,1]:=rChrom[i+1].cap;
          inc(sa);
        end;
      end;
    end;
  end;
end.
```

```

unit uHasil;

interface

uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms,
  Dialogs, StdCtrls, ExtCtrls, TeEngine, Series, TeeProcs, Chart,
  Grids,
  ComCtrls;

type
  TfrmHasil = class(TForm)
    PageControl1: TPageControl;
    TabSheet3: TTabSheet;
    TabSheet4: TTabSheet;
    TabSheet9: TTabSheet;
    Panell1: TPanel;
    btnClose: TButton;
    btnLFAwal: TButton;
    GroupBox6: TGroupBox;
    Label8: TLabel;
    Label9: TLabel;
    Label10: TLabel;
    Label11: TLabel;
    Label12: TLabel;
    lblGen: TLabel;
    lblLoad: TLabel;
    lblLoss: TLabel;
    edtSumGen: TEdit;
    edtSumLoad: TEdit;
    edtSumLoss: TEdit;
    edtIterasi: TEdit;
    edtTime: TEdit;
    btnHitung: TButton;
    TabSheet8: TTabSheet;
    fgBranch: TStringGrid;
    fgBus: TStringGrid;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    TabSheet5: TTabSheet;
    TabSheet6: TTabSheet;
    Chart1: TChart;
    Series1: TLineSeries;
    Series2: TLineSeries;
    fgBus2: TStringGrid;
    fgBranch2: TStringGrid;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    lblGen2: TLabel;
    lblLoad2: TLabel;
    lblLoss2: TLabel;
    edtSumGen2: TEdit;
  end;

```

```
edtSumLoad2: TEdit;
edtSumLoss2: TEdit;
edtIterasi2: TEdit;
edtTime2: TEdit;
GroupBox2: TGroupBox;
GroupBox3: TGroupBox;
pbGen: TProgressBar;
TabSheet7: TTabSheet;
PageControl2: TPageControl;
TabSheet10: TTabSheet;
fgHasilSA: TStringGrid;
Label28: TLabel;
Label6: TLabel;
Label7: TLabel;
Label13: TLabel;
Label14: TLabel;
Label15: TLabel;
edtPmutat: TEdit;
edtPcross: TEdit;
edtLength: TEdit;
edtPopSize: TEdit;
edtMaxGen: TEdit;
Label16: TLabel;
Label19: TLabel;
Label20: TLabel;
Label23: TLabel;
edtCountCap: TEdit;
edtLamdaV: TEdit;
edtLamdaCap: TEdit;
edtVmin: TEdit;
edtVmax: TEdit;
Label25: TLabel;
Label26: TLabel;
GroupBox4: TGroupBox;
Label27: TLabel;
edtCostCap1: TEdit;
Label34: TLabel;
edtPflip: TEdit;
Label21: TLabel;
Button1: TButton;
Label24: TLabel;
edtCostCap: TEdit;
Label32: TLabel;
edtNCap: TEdit;
Label35: TLabel;
Label33: TLabel;
edtKa: TEdit;
Label17: TLabel;
edtPloss1: TEdit;
Label18: TLabel;
GroupBox5: TGroupBox;
Label29: TLabel;
Label30: TLabel;
Label36: TLabel;
Label37: TLabel;
edtCostCap2: TEdit;
edtPloss2: TEdit;
```

```

Label22: TLabel;
procedure btnCloseClick(Sender: TObject);
procedure btnLFAwalClick(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure Button1Click(Sender: TObject);
procedure FormCreate(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmHasil: TfrmHasil;

implementation

uses uUtils, uLoadflow, uNewtonRaphson, uComplex, uObjFunc, uIGA,
  uTypDatGA;

{$R *.dfm}

procedure TfrmHasil.btnCloseClick(Sender: TObject);
begin
  Close;
end;

procedure TfrmHasil.btnLFAwalClick(Sender: TObject);
var i, Nbus, Nsal, ia, ja: integer;
    V, Sg, SL: CxArr1;
    Cap: dArr1;
    Lc, Tr: dArr2;
    TypBus: iArr1;
    Z, Tp, Alir, Arus: CxArr2;
    mulai, selesai, selang: TDateTime;
    jam, menit, detik, mdetik: word;
begin
  DecodeCommDataToLFData(gBus, Nbus, Nsal, V, Sg, SL, Cap, TypBus,
    gBranch, Z, Tp, Lc, Tr);
  mulai:=time;
  NewtonRaphson(gParamLF, V, Sg, SL, Cap, TypBus, Z, Tp, Alir, Arus,
    Lc, Tr);
  selesai:=time;
  selang:=selesai-mulai;
  Series1.Clear;
  for i:=0 to high(gBus) do
  begin
    fgBus.Cells[0, i+1]:=IntToStr(i+1);
    fgBus.Cells[1, i+1]:=RealToStr(V[i].real, 5);
    fgBus.Cells[2, i+1]:=RealToStr(V[i].imag*
      ANGLE_OF_DEGREES/ANGLE_OF_RADIAN, 5);
    fgBus.Cells[3, i+1]:=RealToStr(Sg[i].real, 3);
    fgBus.Cells[4, i+1]:=RealToStr(Sg[i].imag, 3);
    fgBus.Cells[5, i+1]:=RealToStr(SL[i].real, 3);
    fgBus.Cells[6, i+1]:=RealToStr(SL[i].imag, 3);
    fgBus.Cells[7, i+1]:=RealToStr(Cap[i], 3);
    fgBus.Cells[8, i+1]:=IntToStr(TypBus[i]);
  end;
end;

```



```

pbGen.Max:=MaxGen;
PopSize:=StrToInt (edtPopSize.Text);
Length:=StrToInt (edtLength.Text);
Pcross:=StrToFloat (edtPcross.Text);
Pmutat:=StrToFloat (edtPmutat.Text);
Pflip:=StrToFloat (edtPflip.Text);
Lengtha:=StrToInt (edtLength.Text);
NCap:=StrToInt (edtNCap.Text);
iga:=TIGA.Create (MaxGen, PopSize, Length, NCap, Pcross, Pmutat, Pflip);
chromHasil:=iga.ChromHasil;
gObjFunc.doHitungAwal (Ploss1, CostCap1);
gObjFunc.doHitungAkhir (chromHasil, LBus, Ploss2, CostCap2, HasilCap);
//=====
Nbus:=high (LBus)+1;
Nsal:=high (gBranch)+1;
DecodeCommDataToLFData (LBus, Nbus, Nsal, V, Sg, SL, Cap, TypBus,
gBranch, Z, Tp, Lc, Tr);
NewtonRaphson (gParamLF, V, Sg, SL, Cap, TypBus, Z, Tp, Alir, Arus,
Lc, Tr);
Series2.Clear;
for i:=0 to high (gBus) do
begin
fgBus2.Cells[0,i+1]:=IntToStr (i+1);
fgBus2.Cells[1,i+1]:=RealToStr (V[i].real, 5);
fgBus2.Cells[2,i+1]:=RealToStr (V[i].imag*
ANGLE_OF_DEGREES/ANGLE_OF_RADIAN, 5);
fgBus2.Cells[3,i+1]:=RealToStr (Sg[i].real, 3);
fgBus2.Cells[4,i+1]:=RealToStr (Sg[i].imag, 3);
fgBus2.Cells[5,i+1]:=RealToStr (SL[i].real, 3);
fgBus2.Cells[6,i+1]:=RealToStr (SL[i].imag, 3);
fgBus2.Cells[7,i+1]:=RealToStr (Cap[i], 3);
fgBus2.Cells[8,i+1]:=IntToStr (TypBus[i]);
Series2.Add (V[i].real, IntToStr (i+1));
end;
for i:=0 to high (gBranch) do
begin
ia:=gBranch[i].dari-1;
ja:=gBranch[i].ke-1;
fgBranch2.Cells[0,i+1]:=IntToStr (i+1);
fgBranch2.Cells[1,i+1]:=IntToStr (gBranch[i].dari);
fgBranch2.Cells[2,i+1]:=IntToStr (gBranch[i].ke);
fgBranch2.Cells[3,i+1]:=RealToStr (Alir[ia,ja].real, 3);
fgBranch2.Cells[4,i+1]:=RealToStr (Alir[ia,ja].imag, 3);
fgBranch2.Cells[5,i+1]:=RealToStr (Arus[ia,ja].real, 3);
fgBranch2.Cells[6,i+1]:=RealToStr (Arus[ia,ja].imag, 3);
fgBranch2.Cells[7,i+1]:=IntToStr (gBranch[i].ke);
fgBranch2.Cells[8,i+1]:=IntToStr (gBranch[i].dari);
fgBranch2.Cells[9,i+1]:=RealToStr (Alir[ja,ia].real, 3);
fgBranch2.Cells[10,i+1]:=RealToStr (Alir[ja,ia].imag, 3);
fgBranch2.Cells[11,i+1]:=RealToStr (Arus[ja,ia].real, 3);
fgBranch2.Cells[12,i+1]:=RealToStr (Arus[ja,ia].imag, 3);
end;
edtSumGen2.Text:=toStringJ (gParamLF.SumGen, 3);
edtSumLoad2.Text:=toStringJ (gParamLF.SumLoad, 3);
edtSumLoss2.Text:=toStringJ (gParamLF.SumLoss, 3);
edtIterasi2.Text:=IntToStr (3);
selesai:=time;

```

```

selang:=selesai-mulai;
DecodeTime (selang, jam, menit, detik, mdetik);
edtTime2.Text:=IntToStr(jam)+' '+IntToStr(menit)+' '+
IntToStr(detik)+' '+IntToStr(mdetik);
fgHasilSA.RowCount:=high(HasilCap)+2;
for i:=0 to high(HasilCap) do
begin
    fgHasilSA.Cells[0,i+1]:=IntToStr(i+1);
    fgHasilSA.Cells[1,i+1]:=FloatToStr(HasilCap[i,0]);
    fgHasilSA.Cells[2,i+1]:=FloatToStr(HasilCap[i,1]);
end;
edtPloss1.Text:=FormatFloat('#,##0.00',Ploss1);
edtPloss2.Text:=FormatFloat('#,##0.00',Ploss2);
edtCostCap1.Text:=FormatFloat('#,##0.00',CostCap1);
edtCostCap2.Text:=FormatFloat('#,##0.00',CostCap2);
//=====
gObjFunc.Free;
iga.Free;
end;

procedure TfrmHasil.Button1Click(Sender: TObject);
var per:double;
begin
    edtMaxGen.Text:='50';
    edtPopSize.Text:='50';
    edtCountCap.Text:='4';
    edtLength.Text:=IntToStr(high(gBus)+1);
    edtPcross.Text:='0.75';
    edtPmutat.Text:='0.003';
    per:=StrToInt(edtCountCap.Text)/StrToInt(edtLength.Text);
    edtPflip.Text:=FormatFloat('#,##0.0000',per);
    edtNCap.Text:='50';
    edtCostCap.Text:='10000';
    edtLamdaV.Text:='100';
    edtLamdaCap.Text:='1000';
    edtVmin.Text:='0.95';
    edtVmax.Text:='1.05';
    edtKa.Text:='10000';
    btnHitung.Enabled:=true;
end;

procedure TfrmHasil.FormCreate(Sender: TObject);
begin
    fgHasilSA.Cells[0,0]:='No';
    fgHasilSA.Cells[1,0]:='Bus';
    fgHasilSA.Cells[2,0]:='Kap (MVAR)';
end;

end.

```

LAMPIRAN

II

- ✓ FORMULIR SKRIPSI
- ✓ SURAT SURVEY





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

LEMBAR BIMBINGAN SKRIPSI

Nama : PENDHI PURWO PRASETYO

NIM : 01.12.125

NIRM : 0032021102125

Jurusan : TEKNIK ELEKTRO S-1

Konsentrasi : TEKNIK ENERGI LISTRIK

Judul Skripsi :

**OPTIMISASI *MULTI - OBJECTIVE VOLT VAR* MENGGUNAKAN
METODE *PARETO FRONT – GA* PADA SALURAN TRANSMISI
150 kV SUB SISTEM PAITON - BALI**

Tanggal Mengajukan Skripsi : 5 Maret 2006

Tanggal Menyelesaikan Skripsi : 21 September 2006

Dosen Pembimbing : Ir. H. Choiri

Telah dievaluasi dengan nilai : 80 (Delapan Puluh)

Malang, September 2006

Mengetahui
Ketua Jurusan Teknik Elektro S-1

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

Disetujui
Dosen Pembimbing

Ir. H. Choiri
NIP.130703042



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 : PENDHI PURWO PRASETYO
N.I.M. : 01.12.125
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : OPTIMISASI *MULTI – OBJECTIVE VOLT VAR*
MENGUNAKAN METODE *PARETO FRONT – GA*
PADA SALURAN TRANSMISI 150 kV SUB SISTEM
PAITON - BALI

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

Hari : Kamis
Tanggal : 21 September 2006
Dengan Nilai : 74,3 (B+)

Panitia Ujian Skripsi



Ketua Majelis Penguji

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

Sekretaris Majelis Penguji

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

Anggota Penguji

Penguji Pertama

(Ir. M. Abdul Hamid, MT)
NIP. 1018800188

Penguji Kedua

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



PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : **PENDHI PURWO PRASETYO**
 NIM : **0112125**
 Semester : **ix (sembilan)**
 Fakultas : **Teknologi Industri**
 Jurusan : **Teknik Elektro S-1**
 Konsentrasi : **Teknik Energi Listrik**
 Alamat :

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

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

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

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

Telah diteliti kebenaran data tersebut diatas
 Recording Teknik Elektro

[Handwritten signature]
 (.....)

Malang,2005
 Pemohon

[Handwritten signature]
 (.....)
 PENDHI P. P

Disetujui
 Ketua Jurusan Teknik Elektro

[Handwritten signature]

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

Mengetahui
 Dosen Wali

[Handwritten signature]
 (.....)

Catatan :

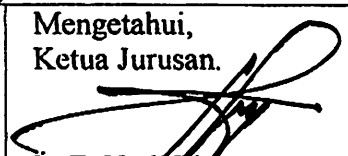
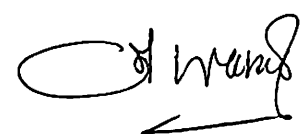
Bagi mahasiswa yang telah memenuhi persyaratan mengambil SKRIPSI agar membuat proposal dan mendapat persetujuan dari Ketua Jurusan/Sekretaris Jurusan T. Elektro S-1

1. IPK 3.75 / 2.76
 2. 136
 3. - 7 praktik yg baru
- Praktik III, V belum program.
 Seminar



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

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika *)

1	Nama Mahasiswa : <u>PENPHI PURWO P</u>		Nim : <u>0112125</u>	
2	Waktu pengajuan	Tanggal : <u>5</u>	Bulan : <u>9</u>	Tahun : <u>06</u>
Spesifikasi judul (berilah tanda silang)				
3	<input checked="" type="checkbox"/> a. Sistem Tenaga Elektrik <input type="checkbox"/> b. Energi & Konversi Energi <input type="checkbox"/> c. Tegangan Tinggi & Pengukuran <input type="checkbox"/> d. Sistem Kendali Industri		<input type="checkbox"/> e. Elektronika & Komponen <input type="checkbox"/> f. Elektronika Digital & Komputer <input type="checkbox"/> g. Elektronika Komunikasi <input type="checkbox"/> h. lainnya	
4	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen *) : <u>Ir. I Made Wantane, MT</u>		Mengetahui, Ketua Jurusan.  Ir. F. Yudi Limpraptono, MT Nip. Y. 1079500274	
5	Judul yang diajukan mahasiswa :	<u>Optimisasi Multi Objective Volt-Var Menggunakan Kombinasi Pareto front - GA pada Saluran Transmisi 150 KV Sub sistem Palton-Bali</u>		
6	Perubahan Judul yang disetujui Dosen sesuai materi bidang ilmu		
7	Catatan : <u>Rumus? yg di jml ke masukan ke proposal</u> <u>Buat Algoritma utam. Kombinasi Pareto front - GA</u> <u>Sepernah Flow chart</u>		Disetujui, 200..... Dosen 	
	Persetujuan Judul Skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu			

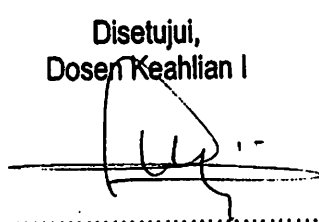
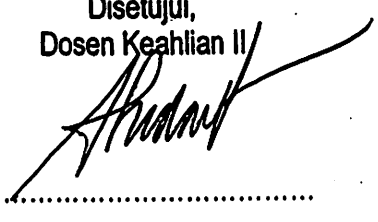
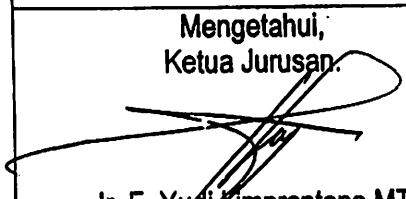
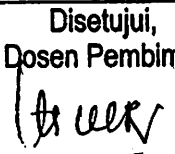
Perhatian :

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



BERITA ACARA SEMINAR PROPOSAL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika*)

1.	Nama Mahasiswa: <u>PENDHI PURWO PRASETYO</u>	Nim: <u>0112125</u>		
2.	Keterangan	Tanggal	Waktu	Tempat
	Pelaksanaan	<u>30/06/2006</u>	<u>13.00 - selesai</u>	Ruang:
3.	Spesifikasi Judul (berilah tanda silang)**)			
	a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri	e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya		
4.	Judul Proposal yang diseminarkan Mahasiswa	<u>OPTIMISASI MULTI-OBJECTIVE VOLT-VAR MENGGUNAKAN METODE PARETO FRONT-GA PADA SAURAN TRANSMISI 150 KV SUB SISTEM PAITON BALI</u>		
5.	Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian 2		
6.	Catatan:			
	Catatan:			
7.	Persetujuan Judul Skripsi			
	Disetujui, Dosen Keahlian I 	Disetujui, Dosen Keahlian II 		
	Mengetahui, Ketua Jurusan 	Disetujui, Calon Dosen Pembimbing ybs 		
	<u>Ir. F. Yudi Limpraptono, MT</u> NIP. P. 1039500274 Ir. H. Choiri		

Perhatian:

1. Keterangan: *) Coret yang tidak perlu

**) dilingkari a, b, c, atau g sesuai bidang keahlian



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

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

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

PT. BNI (PERSERO) MALANG
 BANK NIAGA MALANG

Malang, 19 Juli 2006

Nomor : ITN-1546/I.TA/2/06
 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 : PENDHI PURWO P
 Nim : 0112125
 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:

1 Juli 2006 s/d 1 Jan. 2007

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

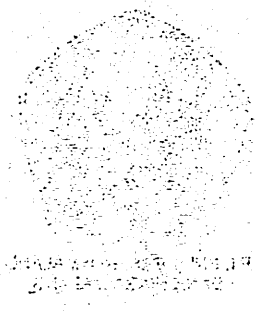
Ir. F. Yudi Limpraptono, MT
 Nip. Y. 1039500274

Tindakan :

1. Mahasiswa yang bersangkutan
2. Arsip.

Form. S-4a

STATE OF TEXAS
COUNTY OF DALLAS
DEPARTMENT OF HEALTH SERVICES
PUBLIC HEALTH DIVISION



NOTICE OF THE DEPARTMENT OF HEALTH SERVICES, PUBLIC HEALTH DIVISION, THAT THE FOLLOWING PERSONS ARE REGISTERED AS NURSES IN THE STATE OF TEXAS:

NAME

EXPIRES

CLASSIFICATION

DATE OF BIRTH

SEX

HEIGHT

WEIGHT

EDUCATION

EXPERIENCE

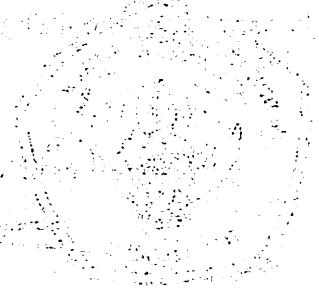
HAIR

12-8-85

0

HAIR COLOR

0



HAIR COLOR

12-8-85

0

0

0



FORMULIR BIMBINGAN SKRIPSI

Nama : Pendhi Purwo Prasetyo
Nim : 01.12.125
Masa Bimbingan : 1 Juli 2006 s/d 1 Januari 2007
Judul Skripsi : OPTIMISASI MULTI-OBJECTIVE VOLT VAR
MENGUNAKAN METODE PARETO FRONT - GA
PADA SALURAN TRANSMISI 150 KV SUB SISTEM
PAITON - BALI

No.	Tanggal	Uraian	Parap Pembimbing
1.	25-7-06	Konsultasi BAB I, II (revisi No. 96 dan ketik).	JH
2.	2-8-06	Konsultasi BAB I, II, III (revisi)	JH
3.	10-8-06	Konsultasi BAB I, II, III (Aee)	JH
4.	14-8-06	Konsultasi BAB IV (Aee)	JH
5.	28-8-06	Konsultasi BAB V (Aee)	JH
6.	28-8-06	Konsultasi makalah seminar (Aee)	JH
7.		Acc. & harmoniskan	
8.			
9.			
10.			

Malang 28-8-2006
Dosen Pembimbing,

Ir. H. Choiri
NIP. 130703042



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

Jalan SunIngrat No. 45 Taman Sidoarjo 61257

Telepon : (031) 7882113, 7882114

Kotak Pos : 4119 SBS

Facsimile : (031) 7882578, 7881024

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

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

Nomor : 219 / 330/ RJTB/ 2006
Surat Sdr. No. : ITN-1801/III.TA/2/06
Lampiran : 1 (satu) lampiran.
Perihal : Ijin Survey/ Pengambilan Data.

18 SEP 2006

Kepada

Yth. Dekan Fakultas Teknologi Industri.
INSTITUT TEKNOLOGI NASIONAL MALANG
Di
M A L A N G

Menunjuk surat Saudara nomor : ITN-1801/III.TA/2/06 tanggal 28 Juni 2006 perihal : Survey/ Permintaan Data, dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada Mahasiswa Saudara, bernama :

• PENDHI PURWO PRASETYO

NIM : 01.12.125

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

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

Demikian harap maklum dan terima kasih atas perhatian saudara.

MANAGER BIDANG SDM & ADMINISTRASI,



Tembusan Yth. :

0. M.SDMO PLN P3B.
0. M.OPHAR PLN P3B RJTB.
0. AsMan Transmisi & Gardu Induk
1. Sdr. Pendhi Purwo Prasetyo

SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : PENDHI PURWO P
Pria/ Wanita : Pria
Tempat / Tanggal lahir : NGANJUK , 02 MEI 1980
Alamat / No. telepon : Jl Tarsisan Surabaya 59 Malang
Pekerjaan : Mahasiswa ITN Malang

Dengan ini saya menerangkan bahwa :

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

Surabaya,

Yang membuat pernyataan

