

SKRIPSI

REKONFIGURASI DENGAN PENDEKATAN MULTI - OBJEKTIF BERBASIS *FUZZY* MENGGUNAKAN METODE *ENHANCED GENETIC ALGORITHM* PADA JARINGAN DISTRIBUSI 20 kV DI G.I. BLIMBING



Disusun oleh :
ARIK DWI TRISTANTO
Nim : 01.12.065

MILIK
PERPUSTAKAAN
ITN MALANG

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

SEPTEMBER 2007

LEMBAR PERSETUJUAN

**REKONFIGURASI DENGAN PENDEKATAN MULTI -
OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE
ENHANCED GENETIC ALGORITHM PADA JARINGAN
DISTRIBUSI 20 kV DI G.I. BLIMBING**

SKRIPSI

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

Disusun Oleh :

ARIK DWI TRISTANTO

01.12.065

Disetujui :

Dosen Pembimbing I

Dosen Pembimbing II


Ir. H. TAUFIK HIDAYAT, MT.
NIP.P. 1618700151


IRRINE BUDI S., ST., MT.
NIP. 132314400



Mengetahui,

Ketua Jurusan Teknik Elektro


Ir. E. YUDI LIMPRAPTONO, MT.
NIP. P. 1039500274

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

KATA PENGANTAR

Puji Syukur kepada Tuhan Yang Maha Esa, karena berkat rahmat dan karunia-Nya penyusun dapat menyelesaikan skripsi yang berjudul

REKONFIGURASI DENGAN PENDEKATAN MULTI-OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE ENHANCED GENETIC ALGORITHM PADA JARINGAN DISTRIBUSI 20 kV DI G.I. BLIMBING

Skripsi ini merupakan salah satu persyaratan untuk menempuh ujian tingkat sarjana S-1 di jurusan Teknik Elektro Konsentrasi Teknik Energi Listrik, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.

Pada kesempatan ini penulis menyampaikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir.F.Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro S-1 Institut Teknologi Nasional Malang.
4. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Sekretaris Jurusan Teknik Elektro S-1 Institut Teknologi Nasional Malang.
5. Bapak Ir. Taufik Hidayat, MT, selaku Dosen Pembimbing I.
6. Ibu Irrine Budi Sulistiawati, ST, MT, selaku Dosen Pembimbing II.
7. Bapak dan Ibu Dosen Teknik Elektro Energi Listrik, Institut Teknologi Nasional Malang.

8. Orang Tua yang telah memberikan do'a dan dukungan.
9. Rekan-rekan Mahasiswa Elektro dan semua pihak yang telah membantu dalam menyelesaikan skripsi ini.

Penyusun menyadari bahwa skripsi ini masih jauh dari sempurna dan masih banyak kekurangan-kekurangan baik isi maupun penyajian, hal ini tidak lain karena keterbatasan ilmu yang penulis miliki, oleh karena itu penulis mengharapkan kritik dan saran yang sifatnya membangun guna kesempurnaan skripsi ini.

Pada akhirnya penulis berharap semoga laporan skripsi ini dapat bermanfaat bagi penulis dan bagi yang membutuhkannya.

Malang, 2007

Penulis

DAFTAR ISI

LEMBAR PERSETUJUAN	i
ABSTRAKSI	ii
KATA PENGHANTAR	iii
DAFTAR ISI	v
DAFTAR GAMBAR	viii
DAFTAR TABEL	ix
DAFTAR GRAFIK	x
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 Pembahasan	4
1.6. Sistematika Penulisan	5
1.7. Kontribusi Penelitian.....	6
BAB II SISTEM JARINGAN DISTRIBUSI PRIMER DAN	
FAKTOR DAYA	7
2.1. Sistem Distribusi	7
2.2. Struktur Jaringan Distribusi Tenaga Listrik	8
2.2.1. Sistem Jaringan Distribusi Radial	9
2.2.1.1. Sistem Radial Pohon	10
2.2.1.2. Sistem Radial dengan Tie dan Switch Pemisah	11
2.3.1.3. Sistem Radial dengan Beban Terpusat.....	12
2.3.1.4. Sistem Radial dengan Pembagian Phase Area	12
2.2.2. Struktur Jaringan Loop	13
2.2.3. Struktur Jaringan Mesh	14
2.3. Daya Sistem Distribusi	15
2.3.1. Daya Semu	15
2.3.2. Daya Aktif	16

2.3.3.	Daya Reaktif	16
2.4.	Faktor Daya	17
2.5.	Rugi- Rugi Daya Saluran Distribusi	18
2.6.	Logika Fuzzy	18
2.7.	Algoritma Genetika	19

**BABIII METODE *ENHANCED GENETIC ALGORITHM* UNTUK
MENENTUKAN KOMBINASI SWITCH PADA JARINGAN
DISTRIBUSI PRIMER 20 KV** 22

3.1.	Analisa Aliran Daya	22
3.1.1.	Tujuan.....	22
3.1.2.	Pendekatan Studi Aliran Daya	22
3.2.	Sistem Per- unit	23
3.3.	Klasifikasi Bus	24
3.3.1.	Bus Beban atau Load Bus	25
3.3.2.	Bus Generator (<i>Bus Generator</i>)	25
3.3.3.	Bus Referensi (<i>Slack Bus</i>)	25
3.4.	Metode Newton Rhapson.....	26
3.4.1.	Algoritma Aliran Daya Newton Rhapson	28
3.4.2.	Diagram Alir Aliran Daya Newton Rhapson	30
3.5.	Metode Rekonfigurasi Jaringan.....	31
3.6.	Rumusan Multi- Objektif Fuzzy.....	32
3.6.1.	MF untuk kehilangan Daya	33
3.6.2.	MF untuk Tegangan Bus Bar	33
3.6.3.	MF untuk Arus Cabang	34
3.6.4.	MF untuk Jumlah Switcing	36
3.6.5.	Persamaan Aliran Daya Yang Di Sederhanakan	37
3.7.	Metode <i>Anhanced Genetic Algorithm</i>	38
3.7.1.	Fungsi- Fungsi Coding dan Decoding.....	39
3.7.2.	Initialisasi	39
3.7.3.	Fungsi Kecocokan	39
3.7.4.	Seleksi dan Elitisme	40

3.7.5.	Perlintasan Multiple- Point.....	40
3.7.1.	Mutasi.....	41
3.8.	Profil di G.I. Blimbing	42
3.9.	Data Umum Pada Trafo II G.I. Blimbini.....	43
BAB IV ANALISA REKONFIGURASI DENGAN PROGRAM		
MENGGUNAKAN METODE ENHANCED GENETIC		
ALGORITHM PADA JARINGAN DISTRIBUSI PRIMER 20 KV DI		
G.I. BLIMBING..... 45		
4.1.	Program Metode Rekonfigurasi	45
4.1.1	Algoritma Program.....	45
4.1.2.	Flow Chart Program.....	47
4.1.3.	Algoritma Metode Penyelesaian Masalah.....	48
4.1.4.	Flow Chart <i>Enhanced Genetic Algorithm</i>	49
4.2.	Analisa Aliran Daya Menggunakan Metode Newton Rhapson .	50
4.3.	Analisa Rekonfigurasi Jaringan Menggunakan	
	Metode <i>Enhanced Genetic Algorithm</i>	57
4.4.	Analisa Perbandingan Sebelum dan Setelah Rekonfigurasi	
	Menggunakan Metode <i>Enhanced Genetic Algorithm</i>	62
4.4.1.	Analisa Simulasi Tegangan dan Sudut Fasa Tegangan	62
4.4.2.	Perbandingan Tingkat Rugi- Rugi Daya Pada Saluran	64
BAB V KESIMPULAN DAN SARAN 66		
5.1	Kesimpulan.....	66
DAFTAR PUSTAKA 67		
LAMPIRAN		

DAFTAR GAMBAR

Gambar :	Halaman :
2-1 Diagram Satu Garis Sistem Tenaga Listrik.....	7
2-2 Jaringan Distribusi Radial	9
2-3 Jaringan Distribusi Radial dengan Tic dan Switch Pemisah	11
2-4 Jaringan Distribusi dengan Beban Terpusat.....	12
2-5 Jaringan Distribusi Radial dengan Phase Area.....	13
2-6 Jaringan Loop	14
2-7 Jaringan Mesh	15
2-8 Hubungan Segitiga Daya.....	17
3-1 Diagram Alir Aliran Daya Newton Rhapson.....	30
3-2 Skematik Tipe NC Switch Dan Tipe NO Switch.....	31
3-7 Diagram Satu Garis dari Sistem Distribusi Radial.....	38
3-9 Diagram Satu Garis Dari G.I. Blimbing.....	42
4-1 Flowchart Program	46
4-2 Flowchart <i>Anhanced Genetic Algorilhm</i>	49
4-3 Tampilan Menu Utama Program.....	50
4-4 Tampilan Input Data General	50
4-5 Tampilan Aliran Daya (awal) Newton Raphson	53
4-6 Konfigurasi Jaringan Radial dari Single Line Diagram G.I. Blimbing Malang	55
4-7 Hasil Program Rugi- Rugi Daya Seclum Rekonfigurasi.....	56
4-8 Konfigurasi Jaringan Mesh	57
4-9 Tampilan Program Parameter EGA.....	58
4-10 Tampilan Aliran Daya Setelah Menggunakan EGA	58
4-11 Tampilan Hasil Program Rugi- Rugi Daya Setelah Menggunakan EGA.....	60
4-12 Tampilan Program Simulasi Kombinasi Switch Setelah Rekonfigurasi.....	61

DAFTAR TABEL

Tabel :	Halaman :
3-1 Data Penyulang Mawar	43
3-2 Data Penyulang Asahan.....	43
3-3 Data Penyulang Wendit.....	44
3-4 Data Penyulang Glintung	44
4-1 Daya dan Impedansi Saluran Jaringan Distribusi G.I. Blimbing.....	52
4-2 Hasil Program Aliran Daya Sebelum Rekonfigurasi	53
4-3 Hasil Program Rugi- Rugi Daya Sebelum Rekonfigurasi.....	56
4-4 Hasil Program Aliran Daya Setelah Menggunakan EGA	59
4-5 Hasil Program Rugi- Rugi Daya Setelah Menggunakan EGA.....	60
4-6 Hasil Kombinasi Optimal Switch.....	61
4-7 Perbandingan Tingkat Tegangan Dan Sudut Tegangan Pada Tiap Bus.....	63
4-8 Tabel Perbandingan Rugi- Rugi Daya Sebelum dan Setelah Rekonfigurasi.....	65

DAFTAR GRAFIK

Gambar :	Halaman :
4-1 Hasil Program Rugi- Rugi Daya Sebelum Rekonfigurasi.....	56
4-2 Hasil Program Rugi- Rugi Daya Setelah Menggunakan EGA.....	60
4-3 Perbandingan Tegangan Sebelum dan Setelah Rekonfigurasi.....	64
4-4 Perbandingan Rugi- Rugi Daya Sebelum dan Setelah Rekonfigurasi.....	65

BAB I

PENDAHULUAN

1.1. Latar Belakang

Pembangunan dan teknologi yang berkembang pesat memberikan dampak yang cukup besar pada kebutuhan dan pola konsumsi masyarakat akan energi listrik. Keadaan tersebut berpengaruh pada pola dan sistem penyediaan energi listrik, terutama pada penyedia tenaga dan pengatur pendistribusian energi listrik agar didapat suatu sistem yang baik.

Dari pengertian diatas *produsen* energi listrik harus benar-benar memperhatikan dan mengusahakan penyaluran energi listrik yang baik kepada konsumen, mulai dari sistem transmisi tenaga sampai dengan sistem distribusi.

Didalam suatu penyaluran tenaga listrik, sistem distribusi menduduki tempat yang penting, karena akan berhubungan langsung dengan pemakai atau konsumen energi listrik. Keberhasilan di dalam sistem distribusi dapat terjamin bila dipenuhinya persyaratan mengenai pertumbuhan beban (pada kondisi operasi normal).

Kelemahan sistem distribusi yang hampir semuanya menggunakan sistem radial sulitnya memenuhi aspek teknis, yakni lokasi beban dan variasi dari kerapatan beban yang menyebabkan tingginya rugi daya dari sistem distribusi telah dilakukan oleh beberapa peneliti dengan memformulasikan permasalahan sebagai persoalan rekonfigurasi jaringan distribusi.

Penyelesaian persoalan rekonfigurasi jaringan distribusi bertujuan untuk meminimalisasi rugi daya jaringan sehingga fungsi obyektif mempresentasikan menghitung rugi daya total dari sistem distribusi^[1].

Berbagai permasalahan tidak hanya untuk memenuhi tujuan yang realistis dan batasan-batasan rekonfigurasi akan tetapi, juga untuk melakukan pendekatan-pendekatan matematis yang cermat dalam proses pemecahan masalah. Teori fuzzy memberikan suatu bingkai kerja yang baik untuk memadukan pendekatan matematis ke dalam rumusan rekonfigurasi yang lebih realistis, dengan menggunakan suatu komputasi yang efisien. Sebuah pendekatan yang mengoptimalisasi fungsi objektif tunggal itu tidak sesuai dengan masalah ini. Pemilihan fuzzy karena di ambil untuk simulasi memperhitungkan objektif yang berganda (banyak) dan untuk mendapatkan keputusan yang maksimal. EGA (*Enhanced Genetic Algorithm*) merupakan metode yang efisien untuk memecahkan masalah optimalisasi kombinasi skala besar, sebab kemampuannya untuk mencari solusi yang hampir merata^[1].

1.2. Rumusan Masalah

Bagaimana cara mengatasi masalah rugi-rugi daya dan rugi tegangan pada sistem distribusi radial yang besar dan kompleks. Serta dengan bertambah luasnya beban pada jaringan distribusi. Pada analisis ini akan dikaji pengaturan status dari *Normally Open Switch* (NO) dan *Normally Close Switch* (NC) pada penyulang jaringan distribusi sehingga dapat mengoptimalkan rugi-rugi daya. Analisa permasalahan menggunakan metode *Enhanced Genetic Algorithm* yang disimulasikan dengan program komputer *Borland Delphi* versi 7.0.

1.3. Tujuan

Tujuan menganalisa sistem jaringan distribusi radial adalah untuk mengurangi rugi-rugi daya, rugi tegangan dan untuk mendapatkan kombinasi switch yang paling optimal.

1.4. Batasan Masalah

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

1. Hanya membahas rekonfigurasi jaringan distribusi untuk meminimalisasi rugi-rugi daya dan memperbaiki rugi tegangan.
2. Beban diasumsikan sebagai beban tiga fasa seimbang.
3. Tidak membahas jenis dan penyebab terjadinya gangguan.
4. Tidak membahas pengontrolan switching yang dilakukan.
5. Power faktor diasumsikan 0,85.
6. Data dan acuan diambil dari PT. PLN (Persero) Distribusi Jawa Timur pada trafo II G.I. Blimbing-Malang.

1.5. Metodologi

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

1. Studi Literatur

Yaitu kajian pustaka dengan mempelajari teori-teori yang terkait melalui literatur yang ada, yang berhubungan dengan permasalahan pada sistem distribusi radial.

2. Pengumpulan data

Pengumpulan data lapangan yang dipakai dalam objek penelitian yakni pada PT. PLN (Persero) Distribusi Jawa Timur pada trafo II G.I. Blimbing-Malang.

- Data kuantitatif yaitu data yang dapat dihitung atau data yang berbentuk angka-angka.
 - Data kualitatif yaitu data yang berbentuk diagram. Dalam hal ini adalah single line diagram penyulang.
3. Melakukan analisa dengan menggunakan bahasa program Delphi 7.0.
- a) Analisa Aliran Daya pada penyulang di trfo II G.I. Blimbing sistem 20 kV menggunakan *Metode Newton Raphson*.
 - b) Analisa penentuan kombinasi switch menggunakan metode *Enhanced Genetic Algorithm (EGA)*.
 - c) Analisa pengaruh penentuan kombinasi switch terhadap sistem distribusi radial diantaranya perbaikan rugi daya, rugi tegangan dan menganalisa aliran daya menggunakan Metode Newton Raphson.

1.6. Sistematika Penulisan

Untuk mendapat arah yang tepat mengenai hal-hal yang akan dibahas maka dalam skripsi ini disusun sebagai berikut :

BAB I : PENDAHULUAN

Merupakan pendahuluan yang meliputi latar belakang yang melandasi skripsi yang dibahas, rumusan masalah, tujuan yang ingin dicapai,

batasan masalah, metodologi, sistematika penulisan dan kontribusi penelitian.

BAB II : SISTEM JARINGAN DISTRIBUSI PRIMER DAN FAKTOR DAYA

Disini akan menguraikan mengenai sistem jaringan distribusi, pengertian jatuh tegangan, pengaruh rendahnya faktor daya, penjelasan teori fuzzy dan algoritma genetika.

BAB III : METODE *ENHANCED GENETIC ALGORITHM* UNTUK MENENTUKAN KOMBINASI SWITCH PADA JARINGAN DISTRIBUSI PRIMER 20 KV

Pada bab ini akan dibahas sedikit metode aliran daya Newton Raphson, teori tentang Metode *Enhanced Genetic Algorithm* (EGA) serta profil di G.I. Blimbing.

BAB IV : ANALISA REKONFIGURASI DENGAN PROGRAM MENGGUNAKAN METODE *ENHANCED GENETIC ALGORITHM* PADA JARINGAN DISTRIBUSI PRIMER 20 KV TRAF0 II DI G.I. BLIMBING

Pada bab ini akan dibahas penentuan kombinasi switch yang akan dianalisis menggunakan program komputer dengan menggunakan pendekatan Metode *Enhanced Genetic Algorithm* (EGA) serta perbandingan profil tegangan sebelum dan setelah rekonfigurasi.

BAB V : KESIMPULAN

Merupakan bab terakhir yang merupakan intisari dan hasil pembahasan berisikan kesimpulan.

1.7. Kontribusi penelitian

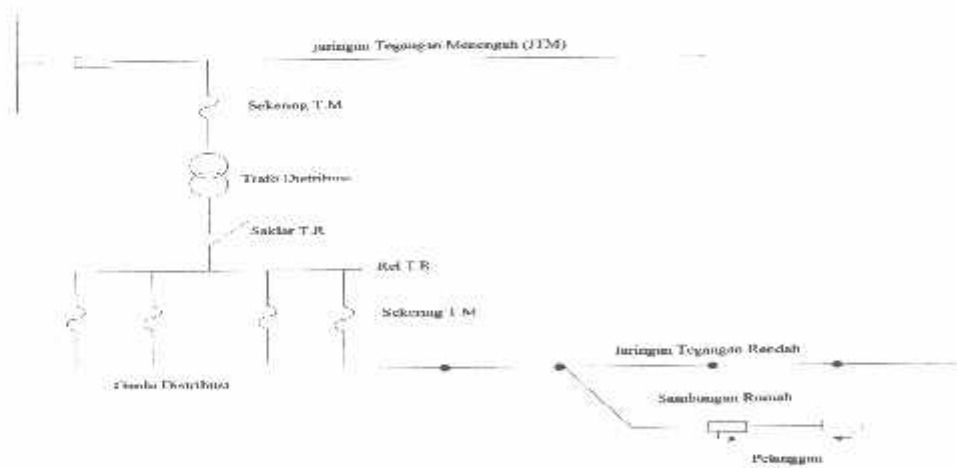
Dalam skripsi ini diharapkan dapat memberikan informasi mengenai analisa penyelesaian persoalan rekonfigurasi jaringan sistem distribusi radial untuk meminimalisasi rugi daya dari jaringan menggunakan *Enhanced genetic algorithm*. Diharap dengan metode, penyelesaian *rekonfigurasi* jaringan memperoleh hasil yang paling *optimal* dan akhirnya dapat diaplikasikan pada instansi-instansi yang terkait dengan sistem penyaluran energi listrik.

BAB II
SISTEM JARINGAN DISTRIBUSI PRIMER DAN
FAKTOR DAYA

2.1. Sistem Distribusi^[2]

Sistem tenaga listrik merupakan sistem yang terbentuk oleh hubungan-hubungan peralatan dan komponen-komponen tenaga listrik seperti generator, transformator, jaringan tenaga listrik beserta bebannya. Peranan utama dari sistem tenaga listrik adalah menyalurkan energi yang dibangkitkan oleh generator ke konsumen-konsumen yang membutuhkan energi listrik tersebut.

Dari gambar 2-1 dapat di lihat bahwa sistem distribusi merupakan bagian paling akhir dari penyaluran sistem tenaga listrik. Sehingga dalam pelaksanaanya sistem distribusi yang berfungsi dan berhubungan langsung dengan beban atau konsumen yang membutuhkan energi listrik.



Gambar 2-1
Diagram Satu Garis Sistem Tenaga Listrik^[2]

Berdasarkan sistem diatas, fungsi dari masing-masing sub sistem dapat diperjelas sebagai berikut :

- a) Pembangkitan berperan sebagai sumberdaya tenaga listrik dan disebut juga sebagai prodaktor energi listrik.
- b) Sistem transmisi berfungsi sebagai penyalur daya listrik secara besar-besaran dari pembangkit kebagian distribusi konsumen.
Dilihat dari sistem transmisi sistem distribusi dapat dianggap sebagai beban sistem transmisi, sistem distribusi dapat dianggap sebagai beban sistem transmisi.
- c) Sistem distribusi berperan sebagai distribusi energi ke konsumen-konsumen yang membutuhkan energi listrik tersebut.

Jaringan setelah keluar dari GI biasa disebut jaringan distribusi. Jaringan distribusi dapat diklasifikasikan menjadi dua bagian sistem yaitu sistem distribusi primer atau sistem distribusi tegangan menengah dan sistem distribusi sekunder atau sistem distribusi tegangan rendah. Dan berdasarkan tingkat penyaluran tegangan, klasifikasi sistem distribusi di bagi menjadi dua yaitu sistem satu fasa dengan dua kawat maupun sistem tiga fasa dengan empat kawat.

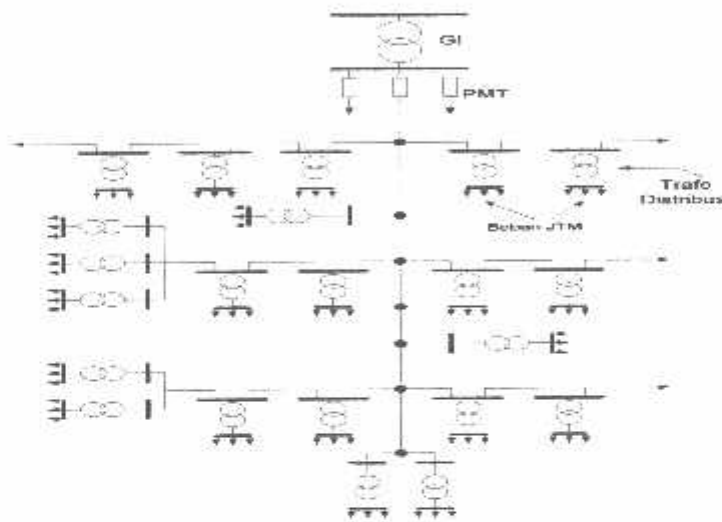
2.2. Struktur Jaringan Distribusi Tenaga Listrik^[2]

Ada beberapa bentuk jaringan yang umum dipergunakan untuk menyalurkan dan mendistribusikan tenaga listrik yaitu :

1. Struktur jaringan distribusi radial.
 2. Struktur jaringan distribusi rangkaian tertutup (*loop*).
 3. Struktur jaringan distribusi *mesh*.
-

2.2.1. Struktur Jaringan Distribusi Radial^[2]

Bentuk jaringan radial ini merupakan bentuk dasar yang paling sederhana dan paling banyak digunakan. Pada struktur radial ini, tidak ada alternatif pasokan, oleh sebab itu tingkat keandalannya relatif rendah. Sehingga untuk menambah keandalan struktur radial ini di upayakan pasokan dayanya tidak satu arah, walaupun pada pengoperasiannya dilaksanakan secara radial. Sistem ini terdiri dari saluran utama (*tunk line*) dan saluran cabang (*lateral*) seperti pada gambar 2-2.



Gambar 2-2
Contoh Jaringan Distribusi Radial^[2]

Pelayanan tenaga listrik untuk suatu daerah beban tertentu dilaksanakan dengan memasang transformator pada sembarang titik, yang jaraknya sedekat mungkin dengan daerah beban yang dilayani. Untuk daerah beban yang menyimpang jauh dari saluran utama maupun saluran cabang, maka akan ditarik lagi saluran tambahan yang dicahangkan pada saluran tersebut.

Kelemahan yang dimiliki oleh sistem radial adalah jatuh tegangan yang cukup besar dan bila terjadi gangguan pada sistem akan dapat mengakibatkan

jatuhnya sebagian atau bahkan keseluruhan beban sistem. Dengan demikian bila salah satu saluran cabang mengalami gangguan maka seluruh beban yang ada di saluran tersebut akan mengalami pemadaman total.

Jaringan distribusi radial mempunyai beberapa keuntungan sebagai berikut :

1. Bentuknya sederhana dibandingkan dengan bentuk yang lain.
2. Biaya investasinya relatif lebih murah, karena saluran menuju ke tiap beban hanya tersedia satu jalur.

Jaringan radial ini dalam perkembangannya mengalami beberapa bentuk *modifikasi* sehingga dikenal beberapa macam jaringan distribusi radial , yaitu :

1. Sistem radial pohon.
2. Sistem radial dengan *Tie* dan *Switch* Pemisah.
3. Sistem radial dengan pusat beban.
4. Sistem radial dengan pembagian Daerah Phasa (*phase area*).

2.2.1.1. Sistem Radial Pohon

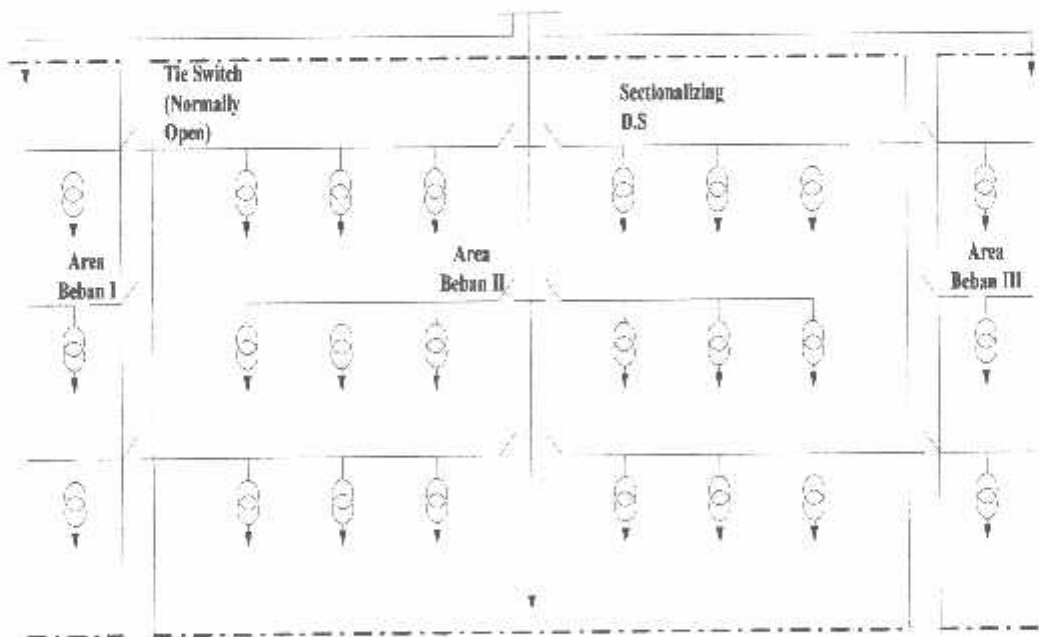
Sistem radial pohon ini merupakan bentuk yang paling dasar dari sistem jaringan radial. Penyulang utama (*main feeder*) ditarik dari suatu Gardu Induk sesuai dengan kebutuhan kemudian dicabangkan melalui penyulang cabang (*lateral feeder*), selanjutnya dicabangkan lagi melalui anak cabang (*sub lateral feeder*). Ukuran dari masing-masing saluran tergantung dari kerapatan arus yang ditanggung. *Main feeder* merupakan saluran yang dialiri arus terbesar, selanjutnya arus mengecil pada tiap cabang tergantung dari besarnya beban.

2.2.1.2. Sistem Radial dengan *Tie* dan Switch Pemisah

Sistem ini merupakan pengembangan dari sistem radial pohon, untuk meningkatkan keandalan sistem saat terjadinya gangguan maka *feeder* yang terganggu akan dilokalisir sedangkan area yang semula dilayani feeder tersebut pelayanannya dialihkan pada feeder yang sehat atau yang tidak terganggu.

Spesifikasi dari jaringan radial ini adalah :

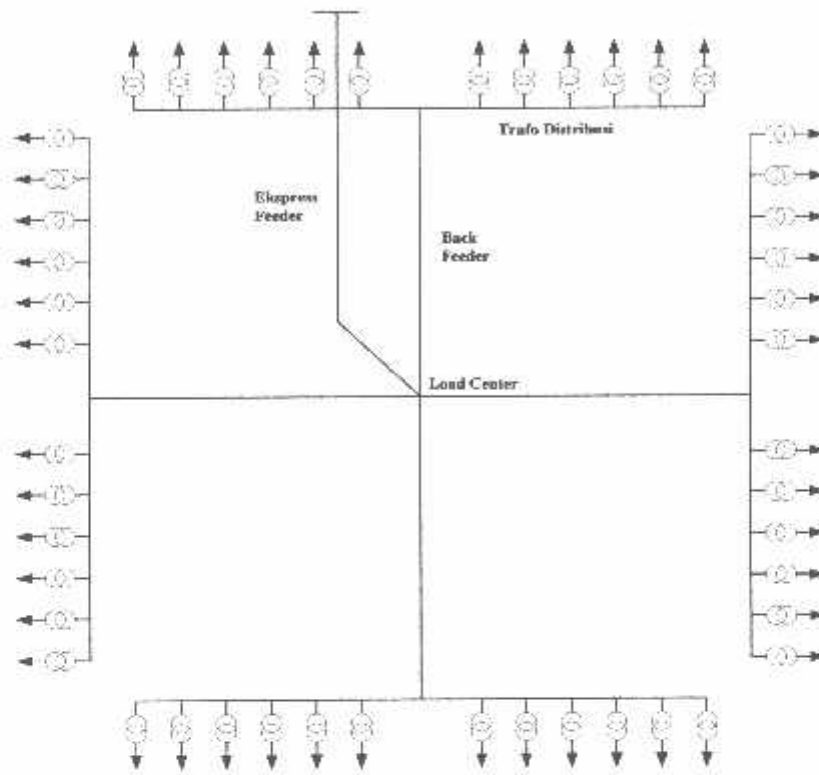
1. Bentuknya sederhana.
2. Biaya investasinya relatif murah.
3. Kualitas pelayanan dayanya relatif jelek karena rugi tegangan dan rugi daya yang terjadi pada saluran relatif besar.
4. Kontinuitas pelayanan dayanya tidak terjamin karena antara titik sumber dan titik beban hanya ada satu alternatif saluran sehingga bila saluran tersebut mengalami gangguan, maka seluruh rangkaian mengalami *black out* secara total.



Gambar 2-3
Contoh Jaringan Distribusi Radial dengan Tie dan Switch Pemisah

2.2.1.3. Sistem Radial dengan Beban Terpusat

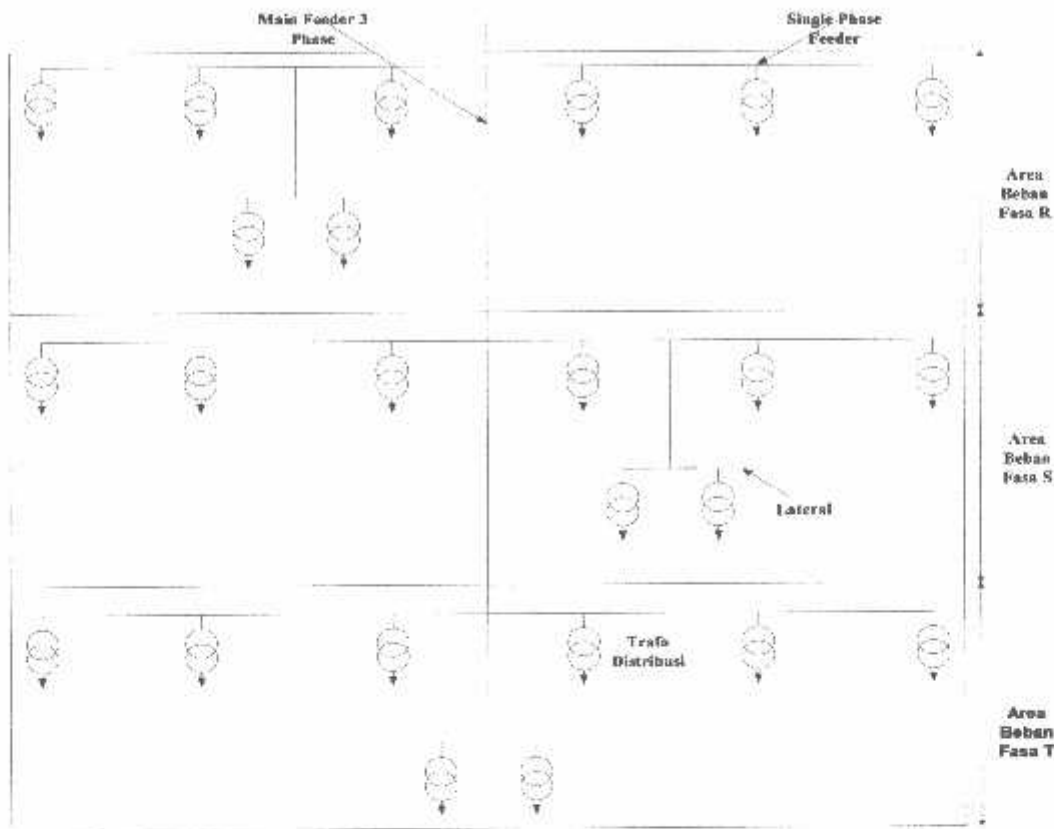
Bentuk dari system ini mensuplai daya dengan menggunakan *main feeder* yang disebut *express feeder* langsung ke pusat beban dan dari titik pusat beban ini disebut dengan menggunakan *back feeder* secara radial.



Gambar 2-4
Contoh Jaringan Distribusi Radial dengan Beban Terpusat

2.2.1.4. Sistem Radial dengan Pembagian Fasa Area

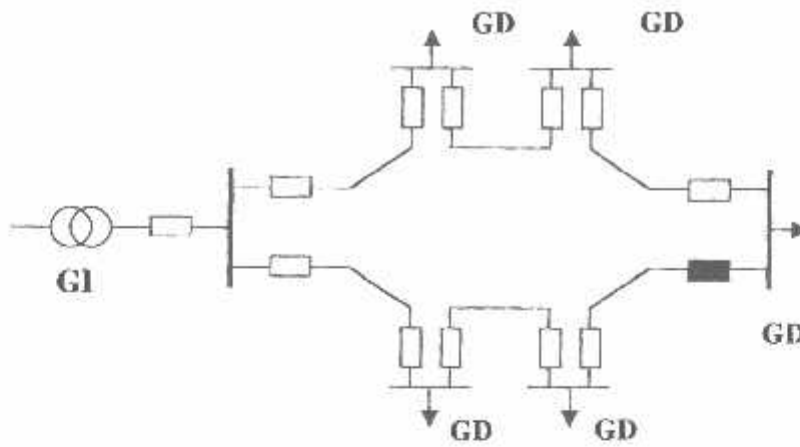
Pada bentuk ini masing-masing fasa dari jaringan bertugas melayani daerah beban yang berlainan. Bentuk ini akan dapat menimbulkan kondisi system tiga fasa yang tidak seimbang (*simetris*), bila digunakan pada daerah beban yang baru dan belum mantap pembagian bebannya.



Gambar 2-5
Contoh Jaringan Distribusi dengan Phasa Area


2.2.2. Struktur Jaringan *Loop*^[2]

Struktur jaringan *loop* merupakan gabungan dari 2 (dua) sistem jaringan radial, dimana pada ujung kedua jaringan dipasang sebuah pemutus (CB) atau pemisah (DS). Saat terjadi gangguan pada jaringan sekunder dapat diisolir, maka pemisah atau pemutus ditutup sehingga aliran daya listrik ke bagian yang tidak terkena gangguan tidak berhenti. Tetapi pada saat terjadi gangguan pada jaringan primernya, maka pemutus beban yang ada di GI akan membuka dan ini menyebabkan semua gardu distribusi akan mengalami pemadaman. Dalam kondisi normal, struktur jaringan *loop* ini merupakan dua struktur jaringan radial. Struktur jaringan ini mempunyai keandalan yang relatif lebih baik, sehingga biaya pembangunannya relatif lebih mahal.



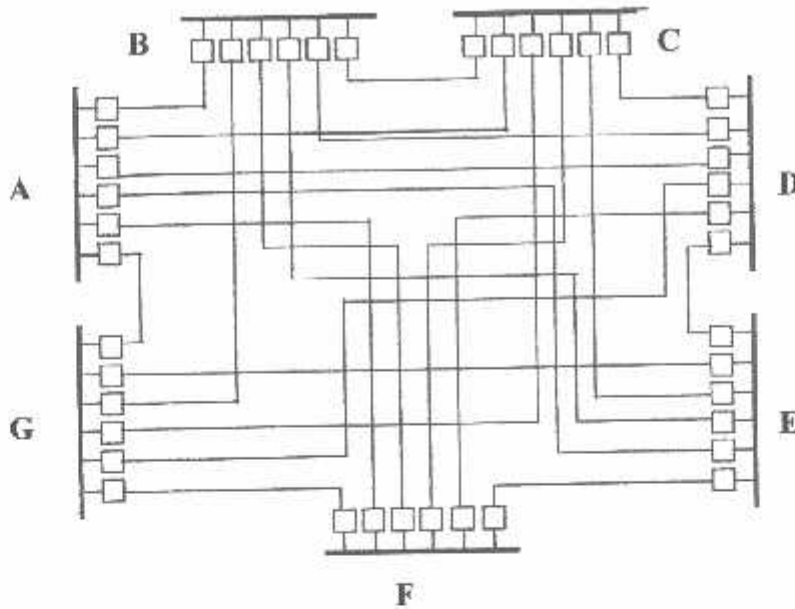
Gambar 2-6
Contoh Struktur Jaringan Loop

Keterangan :  = *Normally Open*

 = *Normally Close*

2.2.3. Struktur Jaringan Mesh^[2]

Struktur jaringan ini merupakan kombinasi antara struktur jaringan *radial* dengan struktur jaringan *loop*. Titik beban memiliki lebih banyak alternatif penyulang, sehingga bila salah satu penyulang terganggu maka dengan segera dapat digantikan oleh penyulang yang lain. Dengan demikian kontinuitas penyaluran daya dan kualitas pelayanan sangat diutamakan. Gambar 2-7 menunjukkan bentuk dari struktur jaringan *Mesh*.



Gambar 2-7
Contoh Struktur Jaringan Mesh
A,B,C,D,E, dan F adalah penyulang

2.3. Daya Sistem Distribusi^{[3] [4]}

Daya merupakan banyaknya perubahan tenaga terhadap waktu dalam besaran tegangan dan arus. Satuan daya adalah watt. Daya yang diserap oleh suatu beban pada setiap saat adalah hasil kali jatuh tegangan sesaat diantara beban dalam volt dengan arus sesaat yang mengalir dalam beban tersebut dalam ampere.

2.3.1. Daya Semu (*Apparent Power*)^{[3] [4]}

Daya semu untuk sistem fasa tunggal, sirkuit dan kawat adalah perkalian skalar arus efektif dan beda tegangan efektifnya.

Jadi daya semu S dinyatakan oleh persamaan :

$$S = [V][I] \dots\dots\dots (2.1)$$

Untuk sistem fasa-tiga daya semunya adalah :

$$S = \sqrt{3}[V_1][I_1] \dots\dots\dots (2.2)$$

dimana V_1 = Tegangan Jala dan I_1 = Arus Jala

2.3.2. Daya Aktif (*Active Power*)^{[3] [4]}

Secara umum daya aktif dinyatakan oleh persamaan :

$$P = [V][I] \cos \varphi \dots\dots\dots (2.3)$$

dimana : V dan I nilai efektifnya

P adalah daya rata-rata yang juga disebut daya aktif

Bila beban fasa-tiganya seimbang maka :

$$P = \sqrt{3}[V_{jala}][I_{jala}] \cos \varphi \dots\dots\dots (2.4)$$

dimana : V_{jala} = tegangan jala efektif dan I_{jala} = arus jala efektif

2.3.3. Daya Reaktif (*Reaktif Power*)^{[3] [4]}

Suku kedua dari persamaan (2.4) dimana suku ini mengandung $\sin \varphi$ yang nilainya berganti-ganti antara positif dan negatif dan nilai rata-ratanya nol. Komponen sesaat dari P ini disebut daya reaktif sesaat dan ini menunjukkan bahwa aliran dayanya bolak-balik menuju beban dan meninggalkan beban. Nilai maksimum daya yang berayun ini dinyatakan dengan Q, yang disebut daya reaktif.

Jadi daya reaktif adalah :

$$Q = [V][I] \sin \varphi \dots\dots\dots (2.5)$$

Bila beban fasa-tiganya seimbang maka :

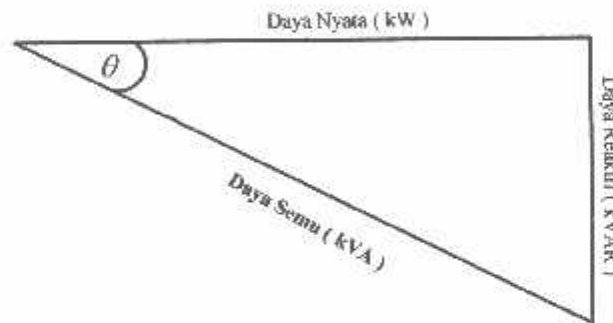
$$Q = \sqrt{3}[V_{jala}][I_{jala}] \sin \varphi \dots\dots\dots (2.6)$$

2.4. Faktor Daya^{[3] [4]}

Setiap pemakaian daya reaktif akan menyebabkan turunnya faktor daya yang kemudian menyebabkan memburuknya karakteristik kerja peralatan-peralatan sistem pada umumnya, baik dari segi teknik operasional maupun dari segi ekonomisnya. Faktor Daya adalah perbandingan antara daya nyata dan daya semu.

$$\text{Faktor Daya} = \frac{\text{Daya Nyata (kW)}}{\text{Daya Semu (kVA)}} \dots\dots\dots (2.7)$$

Untuk daya semu sendiri dibentuk oleh dua komponen daya nyata (kW) dan komponen daya reaktif (kVAR). Hubungan ini dapat digambarkan sebagai berikut :



Gambar 2-8
Hubungan Segitiga Daya

Faktor Daya

$$\text{Cos } \phi = \frac{P}{S} = \frac{P}{\sqrt{P^2 + Q^2}} = \frac{\text{kW}}{\text{kVA}} \dots\dots\dots (2.8)$$

$$\text{kW} = \text{kVA} \cos \theta \dots\dots\dots (2.9)$$

$$\text{Kvar} = \text{Kva} \sin \theta = \text{kW} \tan \theta \dots\dots\dots (2.10)$$

$$\tan \theta = \frac{\text{kVAR}}{\text{kW}} \dots\dots\dots (2.11)$$

Dimana : $\cos \theta$ = Faktor daya
 P = Daya Nyata (kW)
 S = Daya Semu (kVA)
 Q = Daya Reaktif (kVAR)

Faktor daya akan mendahului (*leading*) jika arus mendahului tegangan dan akan tertinggal (*lagging*) bila arus terbelakang terhadap tegangan.

2.5. Rugi-Rugi Daya Saluran Distribusi^{[3] [4]}

Rugi daya adalah besar daya yang hilang dalam penyaluran daya elektrik. Rugi daya ini terdiri dari rugi daya aktif dan rugi daya reaktif. Rugi-rugi ini dapat terjadi pada komponen-komponen umum pada sistem tenaga listrik seperti rugi pada penyulang utama, peralatan saluran dan rugi pada trafo distribusi.

2.6. Logika Fuzzy^[5]

Fuzzy logic digagas oleh Prof Lotfi Zadeh dari universitas barkeley pada tahun 1965. Menemukan bahwa hukum benar atau salah dari logika Boolean tidak memperhitungkan beragam kondisi yang nyata. Pada dasarnya merupakan logika berpikir yang mengijinkan nilai antara yang didefinisikan di antara nilai logika Boolean konvensional seperti benar/salah, putih/hitam, ya/tidak dan sebagainya. Fuzzy logic mempunyai cara berpikir seperti manusia, yang mengenali misalnya hangat panas, sangat panas, dan sebagainya. Untuk menghitung gradasi yang tak terbatas jumlahnya antara benar atau salah. Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam ruang output.

Logika fuzzy memiliki 2 atribut, yaitu:

- Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA, TUA.
- Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variable, seperti: 40, 25, 52 dan sebagainya.

Himpunan fuzzy merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variable fuzzy. Dan fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya yang memiliki interval antara 0 sampai 1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi.

2.7. Algoritma Genetika^[5]

GA pertama kali di kembangkan oleh jhon Holland 1962. Teori algoritma genetika didasari oleh teori Darwin. Landasan Algoritma Genetika terinspirasi dari mekanisme alam, dimana individu yang lebih kuat memiliki kemungkinan untuk menjadi pemenang dan mempunyai kesempatan hidup yang lebih besar didalam lingkungan yang kompetitif.

Sebelum EGA dijalankan, maka sebuah kode yang sesuai (representasi) untuk persoalan harus dirancang. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/string yang terdiri dari komponen genetik terkecil yaitu gen. Sebuah fungsi pengkodean (*coding*) menerjemahkan variabel-variabel keputusan ke dalam string-string biner, dan sebuah fungsi *decoding*

menerjemahkan string-string biner kembali ke dalam variabel-variabel keputusan. EGA menjalankan operasi-operasinya menurut penunjukan string-string ini, lantas sebuah fungsi kecocokan mengevaluasi kecocokan string-string itu.

Inisialisasi (*initialisation*), Sebuah populasi string-string awal secara acak dibangkitkan dari kondisi-kondisi awal dari sistem yang sedang dipelajari. Skema ini dimaksudkan untuk memberikan suatu basis yang ekstensif dimana EGA memulai pencariannya. Setting-setting parameter juga diberikan dalam tahap inisialisasi.

Fungsi kecocokan (*fitness function*) Individu-individu berkembang menurut kecocokan mereka dengan lingkungan. Kecocokan itu didefinisikan sebagai gambaran jasa non-negatif dengan serangkaian variabel-variabel keputusan. Untuk sebuah masalah maksimal, makin besar fungsi objektif harga nilai individu mengimplikasikan kecocokan yang lebih besar.

Seleksi (*selection*) Kriteria untuk memilih individu-individu didasarkan pada nilai kecocokan mereka. Individu-individu dengan nilai kecocokan yang lebih besar, yakni solusi-solusi yang lebih baik terhadap permasalahan, menerima juga jumlah kopian yang lebih besar di dalam himpunan pasangan (*mating pool*).

Elitisme (*elitism*) adalah perangkat yang efektif untuk menyimpan solusi-solusi awal dengan memastikan survival dari string-string terbaik di dalam masing-masing generasi. Perlintasan multiple point (*multiple-point crossover*) Individu-individu di dalam himpunan pasangan secara acak diambil dalam pasangan-pasangan, setelah himpunan itu dikonstruksikan (himpun).

Mutasi (*mutation*) Mutasi mendiversifikasi pencarian dan mencegah konvergensi premature yang menghantar pada individual yang nyaris sama di

dalam sebuah populasi setelah beberapa kali pembangkitan. Untuk masing-masing bit dari turunan, mutasi mengubah sebuah 1 menjadi sebuah 0, atau sebaliknya, menurut probabilitas mutasi tetap. Probabilitas mutasi harus cukup kecil tidak hanya untuk memastikan bahwa perlintasan merupakan perangkat utama untuk menciptakan turunan yang baru, akan tetapi agar EGA tidak tereduksi menjadi pencarian acak.

BAB III

METODE *ENHANCED GENETIC ALGORITHM* UNTUK MENENTUKAN KOMBINASI SWITCH PADA JARINGAN DISTRIBUSI PRIMER 20 KV

Dalam analisa metode rekonfigurasi diperlukan suatu proses aliran daya untuk mengetahui tegangan pada bus beban dan rugi-rugi daya pada saluran. Oleh karena itu pembahasan selanjutnya sebelum membahas proses metode rekonfigurasi adalah analisis aliran daya menggunakan Metode *Newton Raphson*.

3.1. Analisa Aliran Daya^[4]

3.1.1. Tujuan

Tujuan mempelajari analisa aliran daya ini adalah.

- a) Untuk mengetahui kondisi tegangan pada setiap bus dari sistem jaringan.
- b) Untuk mengetahui besarnya daya yang mengalir pada setiap cabang dari struktur jaringan.
- c) Untuk mengetahui besar rugi-rugi daya aktif dan daya reaktif pada setiap cabang jaringan.

3.1.2. Pendekatan Studi Aliran Daya^[4]

Di dalam pengoperasian sistem tenaga listrik, parameter-parameter listrik yang perlu diperhatikan sehubungan dengan analisa aliran daya adalah besarnya magnitude tegangan $|V|$, sudut fasa tegangan θ , daya nyata P dan daya reaktif Q . Daya nyata P mempunyai ketergantungan yang kuat dengan besarnya magnitude

tegangan $|V|$. Bila P dan Q berubah, maka θ dan $|V|$ berubah pula demikian sebaliknya.

Didalam analisa aliran daya terdapat 3 jenis variabel :

1. Variabel bebas misalnya $|V|$ dan Q pada bus generator.
2. Variabel tidak bebas, misalnya P dan $|V|$ pada bus generator.
3. Variabel yang tidak dapat diatur, misalnya kebutuhan konsumen.

3.2. Sistem Per-Unit^[4]

Untuk memudahkan perhitungan – perhitungan dalam sistem tenaga listrik digunakan dalam sistem p.u. (per-unit) yang didefinisikan sebagai perbandingan harga yang sebenarnya dengan harga dasar (*base value*), sehingga dapat dirumuskan sebagai berikut :

$$\text{Besaran per-unit} = \frac{\text{besaran sebenarnya}}{\text{besaran dasar dengan kuantitas yang sama}} \dots\dots\dots (3.1)$$

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

- Untuk sistem 1 fasa :

$$\text{Arus dasar} = \frac{\text{Dasar } KVA_{1\phi}}{\text{Tegangan dasar, } kV_{LN}} \dots\dots\dots (3.2)$$

$$\text{Impedansi dasar} = \frac{\text{Tegangan dasar, } V_{LN}}{\text{Arus dasar}} \dots\dots\dots (3.3)$$

$$\text{Impedansi dasar} = \frac{(\text{Tegangan dasar, } kV_{LN})^2 \times 1000}{\text{Dasar } KVA_{1\phi}} \dots\dots\dots (3.4)$$

$$\text{Impedansi dasar} = \frac{(\text{Tegangan dasar, } kV_{LN})^2}{\text{Dasar } MVA_{1\phi}} \dots\dots\dots (3.5)$$

- Untuk sistem 3 fasa :

$$\text{Arus dasar} = \frac{\text{Dasar } KVA_{3\phi}}{\sqrt{3} \times \text{Tegangan dasar, } kV_{LL}} \dots\dots\dots (3.6)$$

$$\text{Impedansi dasar} = \frac{(\text{Tegangan dasar, } kV_{LL})^2 \times 1000}{\text{Dasar } KVA_{3\phi}} \dots\dots\dots (3.7)$$

$$\text{Impedansi dasar} = \frac{(\text{Tegangan dasar, } kV_{LL})^2}{\text{Dasar } MVA_{3\phi}} \dots\dots\dots (3.8)$$

3.3. Klasifikasi Bus

Pada setiap simpul (rel atau bus) terdapat parameter-parameter sebagai berikut :

1. Daya nyata dinyatakan dengan P satuannya Megawatt (MW)
2. Daya reaktif dinyatakan dengan Q satuannya Mega Volt Ampere Reaktif (MVAR)
3. Besar (magnitude) tegangan mempunyai simbol $|V|$ dengan satuan KiloVolt (kV)
4. Sudut phasa tegangan mempunyai simbol δ dengan satuan derajat

Parameter 1 dan 2 menyatakan daya yang dibangkitkan oleh generator yang mengalir ke bus. Jika pada bus terdapat beban, maka daya tersebut menyatakan selisih antara daya yang dibangkitkan dengan daya beban.

Dalam analisis aliran daya, pada setiap busnya perlu diketahui 2 parameter dari keseluruhan 4 parameter yang diperhitungkan. Dengan melihat kedua parameter yang diketahui, setiap bus dalam suatu sistem dapat diklasifikasikan menjadi tiga yaitu :

1. Bus beban atau load bus
2. Bus generator
3. Bus referensi atau *slack bus*

3.3.1. Bus Beban atau load Bus

Pada bus ini terhubung beban-beban yang permintaan daya aktif dan daya reaktif jelas diketahui, sedangkan tegangan $|V|$ dan sudut fasa tegangan δ merupakan dua besaran yang akan dihitung nilainya.

3.3.2. Bus Generator (*generator bus*)

Bus ini terdapat generator-generator yang nilai tegangan dan daya aktifnya diketahui, sementara daya reaktif dan sudut fasa δ dihitung.

3.3.3. Bus Referensi (*slack bus*)

Pada saat bus ini nilai daya aktif dan daya reaktif dibiarkan mengambang atau tidak diketahui, hal ini dikarenakan daya yang dikirim kepada sistem oleh generator tidak dapat dipastikan terlebih dahulu. Besarnya daya aktif dan daya reaktif selain ditentukan oleh besarnya beban juga ditentukan oleh besarnya daya yang hilang atau rugi-rugi pada saluran, nilainya hanya dapat ditentukan pada akhir perhitungan. Pada bus ini nilai tegangan masing-masing telah ditetapkan, yaitu sebesar 1 pu dan 0 derajat.

3.4. Metode Newton Raphson^[4]

Secara matematis persamaan aliran daya Newton Raphson dapat diselesaikan dengan menggunakan koordinat rektangular, koordinat polar atau bentuk hibrid (gabungan antara bentuk kompleks dengan bentuk polar). Dalam pembahasan skripsi ini menggunakan bentuk polar.

Hubungan antara arus simpul I_p dengan tegangan V_q pada suatu jaringan dengan n simpul dapat dituliskan :

$$I_p = \sum_{q=1}^n Y_{pq} V_{pq} \dots\dots\dots(3.9)$$

Injeksi daya pada simpul p adalah :

$$S_p = P_p - jQ_p = V_p^* \cdot I_p \dots\dots\dots(3.10)$$

$$= V_p^* \sum_{q=1}^n Y_{pq} V_{pq} \dots\dots\dots(3.11)$$

Dalam penyelesaian aliran daya dengan Newton Raphson bentuk persamaan aliran daya yang dipilih adalah *polar*, dimana tegangan dinyatakan dalam bentuk *polar*, yaitu :

$$V_p^* = |V_p| e^{-j\delta_p}$$

$$V_q = |V_q| e^{j\delta_q}$$

$$Y_{pq}^* = |Y_{pq}| e^{-j\theta_{pq}}$$

Maka persamaan dapat ditulis :

$$P_p - jQ_p = \sum_{q=1}^n |V_p V_q Y_{pq}| e^{j(\delta_p - \delta_q + \theta_{pq})} \dots\dots\dots(3.12)$$

Dengan memisahkan bagian riil dan bagian imajiner maka diperoleh :

$$P_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \dots \dots \dots (3.13)$$

$$Q_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \dots \dots \dots (3.14)$$

Kedua persamaan diatas akan menghasilkan suatu kumpulan persamaan serempak (*simultan*) yang tidak linier untuk setiap simpul sistem tenaga listrik. Untuk mengetahui magnitudo tegangan (V) dan sudut fasa (δ) disetiap simpul dapat diselesaikan dengan menggunakan persamaan (3.13) dan (3.14) yang dilinierkan dengan metode Newton Raphson yang dapat dilihat dari persamaan dibawah ini :

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |V| \end{bmatrix} \dots \dots \dots (3.15)$$

Dimana :

ΔP = Selisih injeksi bersih daya nyata dengan penjumlahan aliran daya nyata tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan *iterasi* ke-k

ΔQ = Selisih injeksi bersih daya reaktif dengan penjumlahan aliran daya reaktif tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan *iterasi* ke-k

$\Delta \delta$ = Vektor koreksi sudut fasa tegangan

$\Delta |V|$ = Vektor koreksi magnitudo tegangan

H, L, M, N merupakan elemen-elemen off diagonal dari sub matriks *Jacobian* yang dibentuk dengan mendefinisikan persamaan (3.13) dan (3.14), dimana :

$$H_{pq} = \frac{\partial P_p}{\partial \delta_q} \qquad N_{pq} = \frac{\partial P_p}{\partial |V_q|}$$

$$M_{pq} = \frac{\partial Q_p}{\partial \delta_q} \qquad L_{pq} = \frac{\partial Q_p}{\partial |V_q|}$$

Persamaan diselesaikan untuk menghitung vektor koreksi magnitudo tegangan $\Delta(|V|)$ dan sudut fasa tegangan ($\Delta\delta$) yang baru. Sehingga diperoleh harga magnitudo tegangan dan sudut fasa yang baru, yaitu :

$$|V|^{k+1} = |V|^k + \Delta|V|^k$$

$$\delta^{k+1} = \delta^k + \Delta\delta^k$$

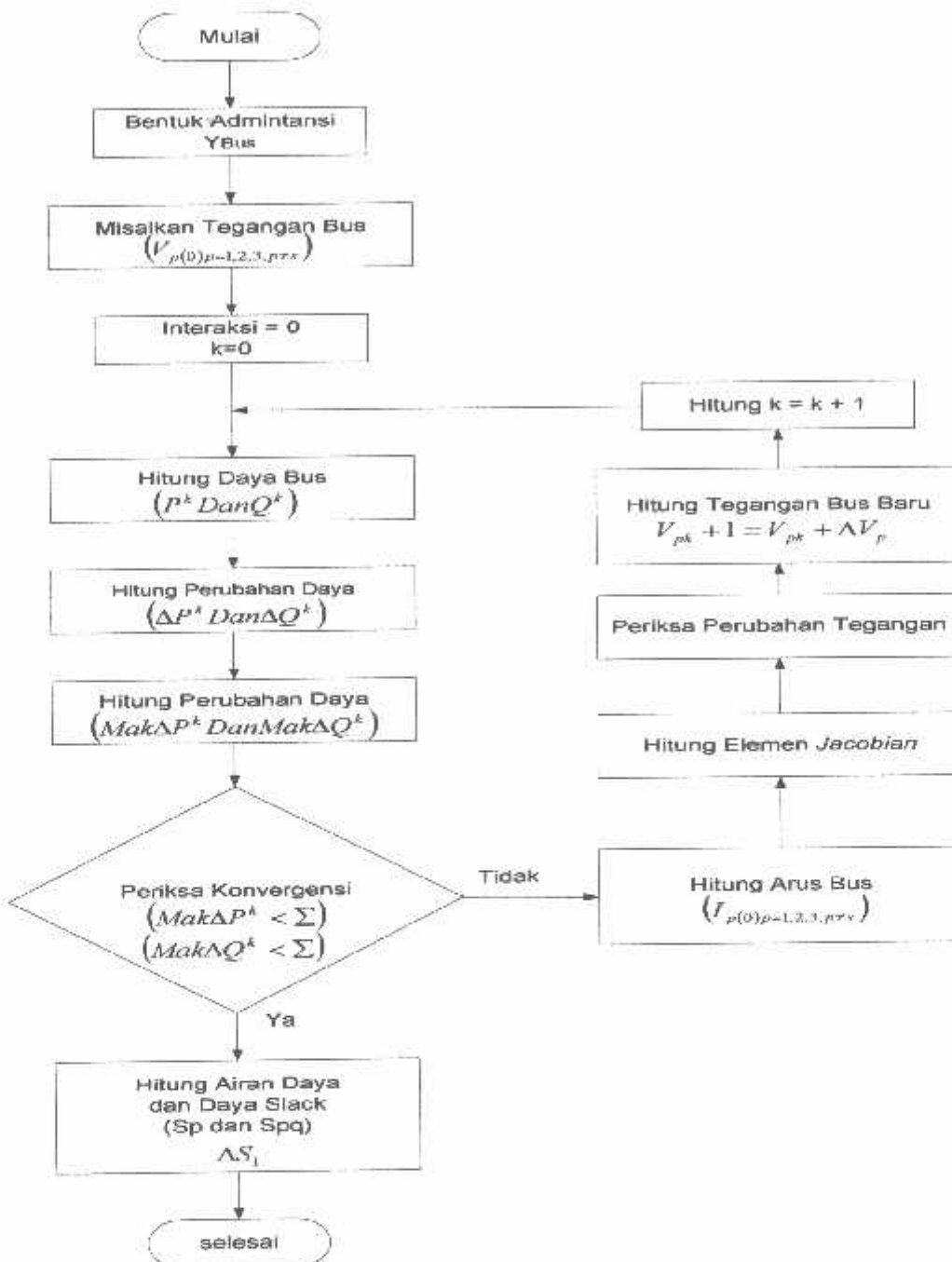
proses perhitungan akan berulang sampai selisih daya nyata dan daya reaktif antara yang dijadwalkan dengan yang dihitung, yaitu ΔP dan ΔQ untuk semua simpul mendekati nilai toleransi atau proses perhitungan iterasi mencapai *konvergen*.

3.4.1. Algoritma Aliran Daya Newton Raphson

1. Tentukan nilai $P_{p(\text{ditetapkan})}$ dan $Q_{p(\text{ditetapkan})}$ yang mengalir kedalam sistem pada setiap rel untuk nilai yang ditentukan atau perkiraan dari besar dan sudut tegangan untuk iterasi pertama atau tegangan yang ditentukan paling akhir untuk iterasi berikutnya.
2. Hitung ΔP pada setiap rel.
3. Hitung nilai-nilai matriks Jaqobian dengan menggunakan nilai-nilai perkiraan atau yang ditentukan dari besar dan sudut tegangan dalam persamaan untuk turunan parsial yang ditentukan dengan differensiasi persamaan (3.13) dan (3.14).
4. Balikkan Jacobian itu dan hitung koreksi-koreksi tegangan $\Delta\delta_q$ dan $\Delta|V_q|$ pada nilai sebelumnya.

5. Hitung nilai baru dari δ_q dan $|V_q|$ dengan menambah $\Delta\delta_p$ dan $\Delta|V_q|$ pada nilai sebelumnya.
 6. Kembali kelangkah pertama dan ulangi proses itu dengan menggunakan nilai untuk besar dan sudut tegangan yang ditentukan paling akhir sehingga semua nilai ΔP dan ΔQ atau semua nilai $\Delta\delta$ dan $\Delta|V|$ lebih kecil dari suatu indeks ketetapan yang dipilih.
-

3.4.2. Diagram Alir Aliran Daya Newton Raphson



Gambar 3-1
Diagram Alir Aliran Daya Newton Raphson

3.5. Metode Rekonfigurasi Jaringan^[2]

Proses menata konfigurasi awal dari jaringan (*initial condition*) menjadi suatu konfigurasi optimum akhir (*optimal condition*), sehingga dari konfigurasi yang terakhir ini diperoleh kerugian daya sistem distribusi yang paling kecil (aliran daya yang paling optimum).

Meskipun jaringan distribusi dioperasikan menggunakan sistem radial, namun besarnya sistem tersebut dikombinasikan ke dalam beberapa tipe. Tipe yang paling umum adalah model jaringan distribusi radial dengan *tie* dan *switch* pemisah.

Rekonfigurasi jaringan distribusi dapat dilakukan dengan mengatur kondisi dari *switch-switch* ini dalam keadaan *on* atau *off*. Switch yang terdapat dalam jaringan distribusi terdapat dua macam :

1. *Normally Closed Switch (NC Switch)* yang dalam keadaan operasi normal, posisi kontakannya selalu tertutup (*on*).
2. *Normally Open Switch (NO Switch)* yang dalam keadaan operasi normal, posisi kontakannya selalu terbuka (*off*).



Gambar 3-2
Diagram Skematik Tipe (a) NC switch dan (b) NO switch

Jadi jelaslah bahwa sebenarnya rekonfigurasi jaringan dalam sistem distribusi tenaga listrik adalah proses mengganti struktur *topologi* dari jaringan distribusi dengan mengubah status *open / closed* dari NO / NC switch sehingga dari proses ini bisa didapatkan aliran daya yang optimal pada sistem distribusi yang menyebabkan kerugian daya jadi minimum dengan proses akhir sistem kembali radial.

3.6. Rumusan *Multi-Objektif Fuzzy*^[1]

Dari rumusan masalah yang telah di sebutkan di atas adalah yang pertama kali di *fuzzy*kan dan di padukan ke dalam fungsi objektif, pemenuhan secara *fuzzy* M melalui faktor-faktor pembebanan yang tepat sebagaimana yang ditunjukkan di bawah ini

$$\text{Max } M = w_1 \mu_p + w_2 \sum_{i=1}^{N_{bs}} \mu_{v_i} + w_3 \sum_{i=1}^{N_{bh}} \mu_{I_i} + w_4 \mu_s \dots \dots \dots (3.16)$$

Dimana:

μ_p = MV untuk meminimalkan kehilangan daya total

$\sum_{i=1}^{N_{bs}} \mu_{v_i}$ = MV untuk meminimalkan pelanggaran tegangan

$\sum_{i=1}^{N_{bh}} \mu_{I_i}$ = MV untuk meminimalkan pelanggaran arus

μ_s = MV dari minimalisasi jumlah switching

N_{bs} = jumlah busbar

N_{bh} = jumlah cabang

$w_i, i = 1,2,3,4$ = merupakan faktor pembebanan

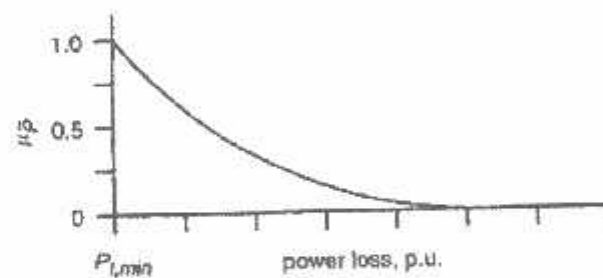
Fungsi-fungsi keanggotaan (MF) digunakan untuk mendeskripsikan empat objektif-objektif yang dinyatakan sebagai berikut :

3.6.1. MF Untuk Kehilangan Daya

Rekonfigurasi jaringan digunakan untuk mencoba meminimalkan kehilangan daya, sementara tidak ada pelanggaran batasan tegangan. Konfigurasi dengan rugi daya yang besar di tunjukkan dalam MV yang rendah. Fungsi eksponensial kondisi ini, ia diseleksi untuk MF dari rugi daya yang ditunjukkan dalam persamaan berikut :

$$\mu_{\bar{p}} = \exp(-(P_{loss} - P_{l,min}) / P_{l,min}) \dots\dots\dots(3.17)$$

Dengan $\mu_{\bar{p}}$ dinyatakan dalam MV yaitu rugi daya P_{loss} dan besar nilai rugi daya $P_{l,min}$ adalah minimum rugi daya yang di peroleh dengan meminimalkan satu objektif (single objective) dari rugi daya.



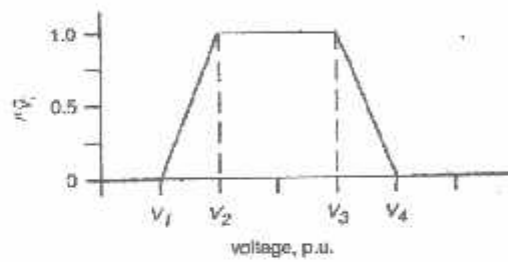
Gambar 3-3
Fungsi Keanggotaan Dari Rugi Daya

3.6.2. MF Untuk Tegangan Busbar

Untuk menseleksi trapezium MF guna mencegah terjadinya kekurangan dan kelebihan tegangan di masing-masing busbar di uraikan sebagai berikut :

$$\mu_{V_i} = \begin{cases} 0, & V < V_1 \\ \frac{V - V_1}{V_2 - V_1}, & V_1 \leq V < V_2 \\ 1.0, & V_2 \leq V < V_3 \\ \frac{V_4 - V}{V_4 - V_3}, & V_3 \leq V < V_4 \\ 0, & V_4 \leq V \end{cases} \dots\dots\dots(3.18)$$

Gambar 3-4 mengetengahkan MF untuk tegangan dimana μ_{V_i} dalam MV untuk tegangan V_i di busbar i ; V_1, V_2, V_3 , dan V_4 , yang merupakan tegangan-
tegangan yang sudah di spesifikasi.

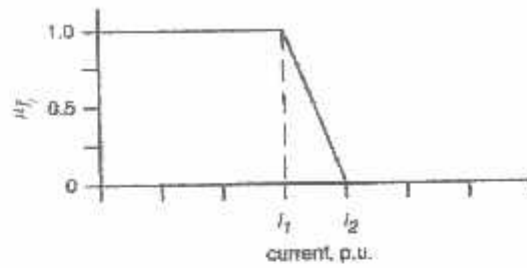


Gambar 3-4
Fungsi Keanggotaan Dari Tegangan Busbar

3.6.3. MF Untuk Arus Cabang

MF untuk arus cabang diuraikan dalam rumusan berikut :

$$\mu_{I_i} = \begin{cases} 1.0, & I < I_1 \\ \frac{I_2 - I}{I_2 - I_1}, & I_1 \leq I < I_2 \\ 0, & I_2 \leq I \end{cases} \dots\dots\dots(3.19)$$



Gambar 3-5
Fungsi Keanggotaan Dari Arus Cabang

Gambar 3-5 menguraikan MF untuk arus cabang, dimana μ_{I_i} merepresentasikan MV arus I_i , arus dalam cabang i ; I_1 dan I_2 merupakan besar nilai arus keluaran. Indeks beban arus rata-rata (CLI_{ave}) dan indeks beban arus maksimum (CLI_{max}) mengevaluasi arus pembebanannya dari konfigurasi jaringan dan dinyatakan dalam rumus sebagai berikut :

$$CLI_{ave} = \frac{1}{N} \sum_{i=1}^{N_{bh}} \frac{I_{f,i}}{I_{c,i}} \dots\dots\dots(3.20)$$

$$CLI_{max} = \text{Max} \left\{ \frac{I_{f,i}}{I_{c,i}}, i = 1, 2, \dots, N_{bh} \right\} \dots\dots\dots(3.21)$$

Dengan :

$I_{f,i}$ = menyatakan arus pada cabang i

$I_{c,i}$ = kapasitas dari cabang jalur i

N_{bh} = jumlah cabang

Indeks pembebanan arus minimum sama dengan nol, menunjukkan tidak ada arus melalui cabang ini CLI_{ave} terendah mengimplikasikan derajat tertinggi dari beban setimbang. Arus cabang $I_{f,i}$ dapat di hitung dengan aliran daya.

3.6.4. MF Untuk Jumlah Switching

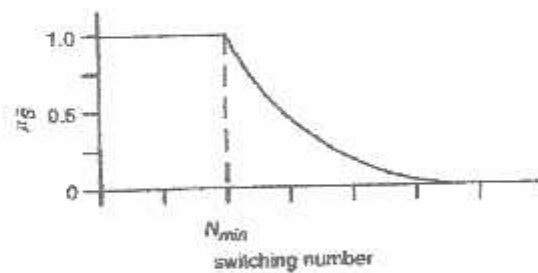
Untuk mengurangi operasi kerja, termasuk rentang umur switch, jumlah switching juga merupakan pertimbangan penting dalam rekonfigurasi dan harus dijaga seminimal mungkin. Sebuah konfigurasi dengan angka switch yang besar ditandai dengan MV terendah. Karena fungsi eksponensial memenuhi syarat ini, maka ia digunakan untuk MF dari jumlah switching, sebagaimana ditunjukkan gambar 3-6, dan dituliskan sebagai berikut :

$$\mu_S = \begin{cases} 1.0, & N_S \leq N_{\min} \\ \exp(-(N_S - N_{\min})/N_{\min}), & N_{\min} < N_S \end{cases} \dots\dots\dots(3.22)$$

Dimana :

μ_S = menyatakan MV dari jumlah switching

N_S dan N_{\min} = merupakan limit dari jumlah switching yang sudah di seleksi



Gambar 3-6
Fungsi Keanggotaan Dari Switching Number

Dimana μ_S menyatakan MV dari jumlah switching N_S dan N_{\min} merupakan limit jumlah switching yang sudah diseleksi sebelumnya.

3.6.5. Persamaan Aliran Daya Yang Disederhanakan

Dari gambar 5 jaringan radial, jalur impedansi antara cabang i dan cabang $i+1$ direpresentasikan sebagai $z_i = r_i + jx_i$ dan model beban diasumsikan sebagai resapan daya konstan, $S_L = P_L + jQ_L$. Dengan mengasumsikan bahwa rugi pada saluran antar cabang itu lebih kecil daripada bentuk daya cabang (*branch power terms*) maka persamaan aliran daya yang dipergunakan :

$$P_{i+1} = P_i - P_{L,i+1} = \sum_{k=i+2}^{N_{bb}} Q_{L,k} \dots\dots\dots(3.23)$$

$$Q_{i+1} = Q_i - Q_{L,i+1} = \sum_{k=i+2}^{N_{bb}} Q_{L,k} \dots\dots\dots(3.24)$$

$$V_{i+1}^2 = V_i^2 - 2 (r_i P_i + x_i Q_i) \dots\dots\dots(3.25)$$

Dimana :

P_{i+1} = daya nyata

Q_{i+1} = daya reactive

V_{i+1} = tegangan busbar

$i+1$ = merupakan nilai-nilai pada cabang

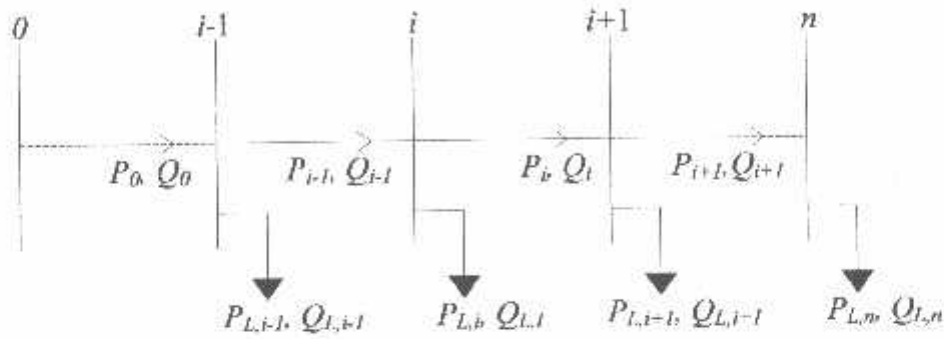
Setelah determinan menentukan besar nilai P_i , Q_i , dan V_i untuk cabang $0, 1, \dots, i$, P_b , Q_b , dan V_i dapat digunakan untuk menghitung arus cabang :

$$I_{f,i} = \sqrt{(P_i^2 + Q_i^2) / V_i^2} \dots\dots\dots(3.26)$$

Untuk kehilangan daya total pada sebuah system distribusi dapat dikalkulasikan dengan cara menjumlah kehilangan daya pada setiap cabang melalui persamaan berikut :

$$P_{loss} = \sum_{i=1}^{N_{bb}} r_i (P_i^2 + Q_i^2) / V_i^2 \dots\dots\dots(3.27)$$

Formulasi multi-objektif (I) dimaksimalkan dengan mempergunakan pendekatan berbasis Enhanced Genetic Algorithm (EGA).



Gambar 3-7
Diagram Satu Garis Dari Sistem Distribusi Radial

3.7. Metode *Enhanced Genetic Algorithm*^[1]

SGA merupakan mekanisme pencarian stokastik berdasarkan prinsip Darwin mengenai evolusi alamiah. Proses evolusioner menyebabkan solusi-solusi awal bergerak menuju titik optimal dari berbagai generasi. Seleksi roulette-wheel yang dipergunakan dalam SGA memiliki potensial pengulangan kembali dimana anggota populasi terbaik bisa jadi gagal menghasilkan keturunan (offspring) di dalam generasi selanjutnya, dan mungkin menyebabkan kesalahan stokastik. Elitisme dan strategi perkalian crossover dilekatkan dalam EGA untuk memperbaiki dan meningkatkan SGA.

3.7.1. Fungsi-Fungsi Coding dan Decoding

Sebuah skema bit-bit biner disebut sebuah string atau sebuah individual di dalam EGA yang cocok dengan serangkaian gen, yakni sebuah kromosom, dalam genetika alamiah. Sebuah fungsi coding menerjemahkan variabel-variabel keputusan ke dalam string-string biner, dan sebuah fungsi decoding menerjemahkan string-string biner kembali ke dalam variabel-variabel keputusan. EGA menjalankan operasi-operasinya menurut penunjukan string-string ini, lantas sebuah fungsi kecocokan mengevaluasi kecocokan string-string itu.

3.7.2. Inisialisasi

Sebuah populasi string-string awal secara acak dibangkitkan dari kondisi-kondisi awal dari sistem yang sedang dipelajari. Skema ini dimaksudkan untuk memberikan suatu basis yang ekstensif dimana EGA memulai pencariannya. Setting-setting parameter juga diberikan dalam tahap inisialisasi.

3.7.3. Fungsi Kecocokan

Individu-individu berkembang menurut kecocokan mereka dengan lingkungan. Kecocokan itu didefinisikan sebagai gambaran jasa non-negatif dengan serangkaian variabel-variabel keputusan. Untuk sebuah masalah maksimal, makin besar fungsi objektif harga nilai individu mengimplikasikan kecocokan yang lebih besar.

3.7.4. Seleksi dan Elitisme

Kriteria untuk memilih individu-individu didasarkan pada nilai kecocokan mereka. Individu-individu dengan nilai kecocokan yang lebih besar, yakni solusi-solusi yang lebih baik terhadap permasalahan, menerima juga jumlah kopian yang lebih besar di dalam himpunan pasangan (*mating pool*). Sebagai contoh, asumsikan bahwa ada m individu masing-masing dengan nilai kecocokan $f_i - 1, 2, \dots, m$. Lantas, di dalam himpunan pasang, jumlah individual (n_i) dengan kecocokan f_i itu proporsional dengan besar nilai yang dihitung dengan rumus

$$n_i = \frac{f_i}{\left(\sum_{j=1}^m f_j \right)} \dots \dots \dots (3.28)$$

Elitisme adalah perangkat yang efektif untuk menyimpan solusi-solusi awal dengan memastikan survival dari string-string terbaik di dalam masing-masing pembangkitan. Elitisme menempatkan string terbaik dari pembangkitan yang lama ke dalam pembangkitan baru untuk meningkatkan (memperbaiki) performa konvergensi SGA.

3.7.5. Perlintasan Multiple-Point

Individu-individu di dalam himpunan pasangan secara acak diambil dalam pasangan-pasangan, setelah himpunan itu dikonstruksikan. Operasi perlintasan digunakan di dalam SGA. merupakan sebuah perlintasan *single-point*, lantaran satu lokasi perlintasan dipilih. Perlintasan sederhana ini dapat digeneralisasi kepada perlintasan *multiple-point* dimana jumlah poin perlintasan N_e dispesifikasi. Dua string ini lantas dibagi dari indukan dipertukarkan satu sama lain untuk menghasilkan turunan mereka. Perlintasan *multiple-point* dapat

memecahkan masalah-masalah tertentu dari kombinasi-kombinasi sifat (fitur) yang dikodekan pada kromosom-kromosom yang tidak bisa dipecahkan perlintasan *single-point*.

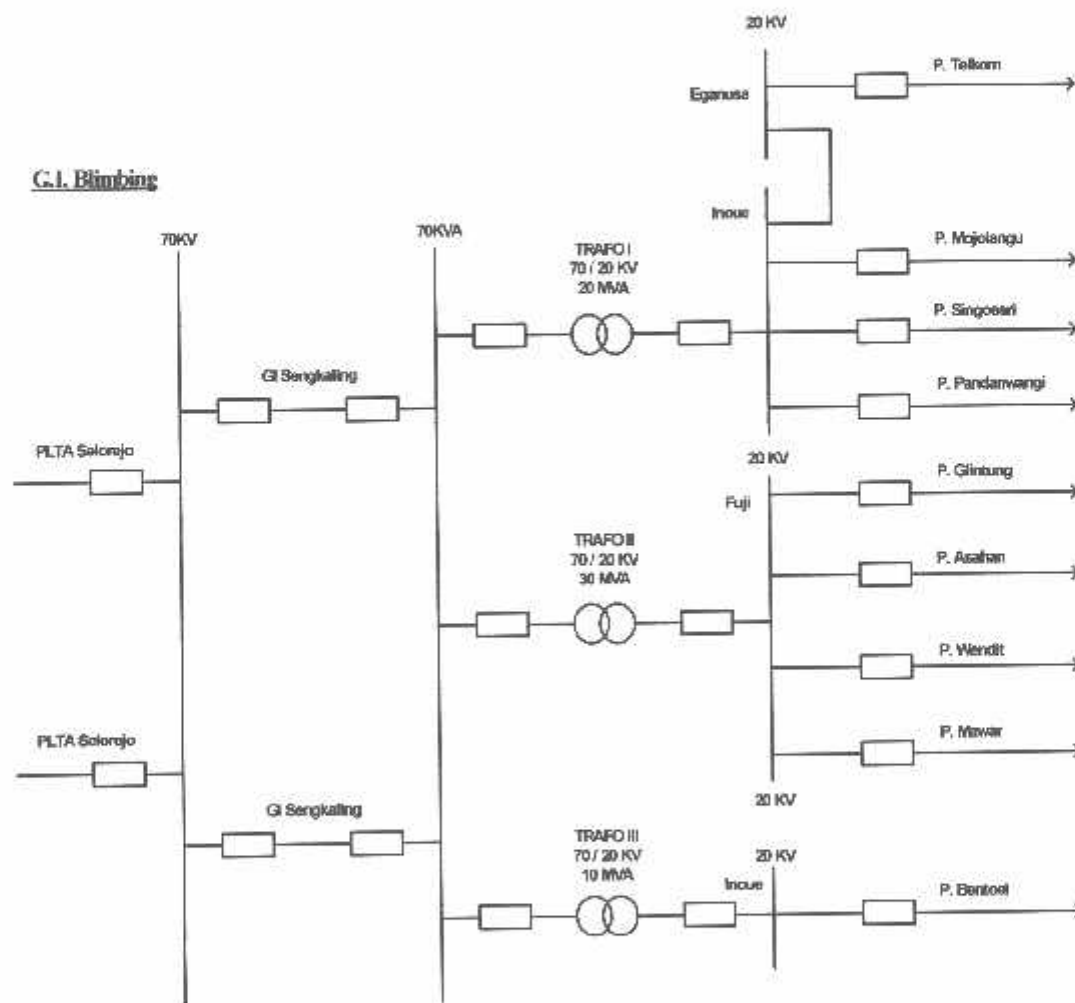
3.7.6. Mutasi

Mutasi mendiversifikasi pencarian dan mencegah konvergensi premature yang menghantar pada individual yang nyaris sama di dalam sebuah populasi setelah beberapa kali pembangkitan. Untuk masing-masing bit dari turunan, mutasi mengubah sebuah 1 menjadi sebuah 0, atau sebaliknya, menurut probabilitas mutasi tetap. Probabilitas mutasi harus cukup kecil tidak hanya untuk memastikan bahwa perlintasan merupakan perangkat utama untuk menciptakan turunan yang baru, akan tetapi agar EGA tidak tereduksi menjadi pencarian acak.

Namun demikian, probabilitas mutasi yang terlalu kecil tidak dapat menghindarkan konvergensi premature. Perlintasan dan mutasi itu diulangi hingga himpunan pasangan (*mating pool*) kosong. Pada saat pembangkitan yang baru itu dicapai, maka proses yang berkembang mengulangi operasi-operasi seleksi genetic, perlintasan dan mutasi, untuk memunculkan individu-individu baru. Proses itu berhenti ketika kecocokan rata-rata berada di dalam persentase kecocokan string terbaik, atau ketika hitungan pembangkitan total tertentu sudah tercapai.

3.8. Profil Di G.I. Blimbing

Berkembangnya teknologi dan bertambahnya populasi penduduk di Malang, memaksa penyedia energi listrik yaitu Perusahaan Listrik Negara (PLN) untuk selalu memberikan yang terbaik bagi pemakai energi listrik (konsumen). Karena kemajuan teknologi, mendorong manusia untuk menggunakan teknologi tersebut. Sehingga penyedia energi listrik harus mengantisipasi apabila beban yang dipergunakan konsumen semakin meningkat. Pada gambar 3-9 merupakan single line dari G.I. Blimbing.



Gambar 3-9
Single line diagram dari G.I. Blimbing

3.9. Data Umum Pada Trafo II G.I. Blimbing

Tabel 3-1
Data penyulang Mawar

NO. Trafo	Nominal (kVA)	Beban Trafo (kVA)	Persen Beban (%)	P (MW)	Q (MVAR)
186	150	88.22	58.81	74.99	46.47
270	150	125.62	83.75	106.78	66.17
369	150	123.20	82.13	104.72	64.90
370	150	91.08	60.72	77.42	47.98
371	150	111.45	74.3	94.73	58.71
424	250	198.31	79.33	168.56	104.47
425	100	40.82	40.82	34.70	21.50
455	250	138.25	55.30	117.51	72.83
542	150	81.06	54.04	68.90	42.70
588	160	110.15	68.84	93.63	58.03

Data tabel selengkapnya dapat dilihat pada lampiran A-1.

Tabel 3-2
Data Penyulang Asahan

NO. Trafo	Nominal (kVA)	Beban Trafo (kVA)	Persen Beban (%)	P (MW)	Q (MVAR)
39	200	133	66.5	113.05	70.06
88	250	187.5	75	159.38	98.77
124	200	70.06	35.03	59.55	36.91
125	150	39.27	26.18	33.38	20.69
128	100	27.61	27.61	23.47	14.54
267	100	61.6	61.60	52.36	32.45
445	630	120	75	102.00	63.21
495	160	17.25	10.78	14.66	9.09
621	100	20.1	12.6	17.09	10.59
622	100	75	75	63.75	39.51

Data tabel selengkapnya dapat dilihat pada lampiran A-1.

Tabel 3-3
Data Penyulang Wendit

NO. Trafo	Nominal (kVA)	Beban Trafo (kVA)	Persen Beban (%)	P (MW)	Q (MVAR)
224	160	72.68	45.43	61.78	38.29
215	200	92.4	46.20	78.54	48.67
338	200	93.68	46.84	79.63	49.35
577	250	208.95	83.58	177.61	110.07
28	160	79.17	49.48	67.29	41.71
203	160	33.08	20.67	28.12	17.43
204	160	101.9	63.7	86.62	53.68
222	630	766.32	74.4	651.37	403.68
293	160	38.06	23.78	32.35	20.05
502	160	11.45	7.16	9.73	6.03

Data tabel selengkapnya dapat dilihat pada lampiran A-1.

Tabel 3-4
Data Penyulang Glitung

NO. Trafo	Nominal (kVA)	Beban Trafo (kVA)	Persen Beban (%)	P (MW)	Q (MVAR)
93	250	109.72	109.72	93.26	57.80
141	200	77.28	77.28	65.69	40.71
196	200	81.62	81.62	69.38	43.00
575	160	51.51	51.51	43.78	27.13
16	250	160.54	160.54	136.46	84.57
51	250	171.46	171.46	145.74	90.32
148	250	97.94	97.94	83.25	51.59
210	400	164	164	139.40	86.39
281	160	108.55	108.55	92.27	57.18
536	160	105.29	105.29	89.50	55.46

Data tabel selengkapnya dapat dilihat pada lampiran A-1.

BAB IV

ANALISA REKONFIGURASI DENGAN PROGRAM MENGGUNAKAN METODE *ENHANCED GENETIC ALGORITHM* PADA JARINGAN DISTRIBUSI PRIMER 20 KV TRAF0 II DI G.I. BLIMBING

4.1. Program Metode Rekonfigurasi Jaringan

Untuk pemecahan masalah rekonfigurasi jaringan distribusi digunakan bantuan program komputer. Program komputer ini menggunakan *Borland Delphi versi 7.0* yang memiliki bahasa pemrograman, sehingga relatif mudah untuk dipelajari dan digunakan untuk mempercepat proses perhitungan yang membutuhkan ketelitian tinggi dan sering melibatkan iterasi yang membutuhkan waktu yang lama bila dikerjakan manual. Program dalam skripsi ini dijalankan dengan diaplikasikan pada komputer menggunakan system operasi Windows XP, PC Intel P-4 2.40 GHz, memori 128 Mb.

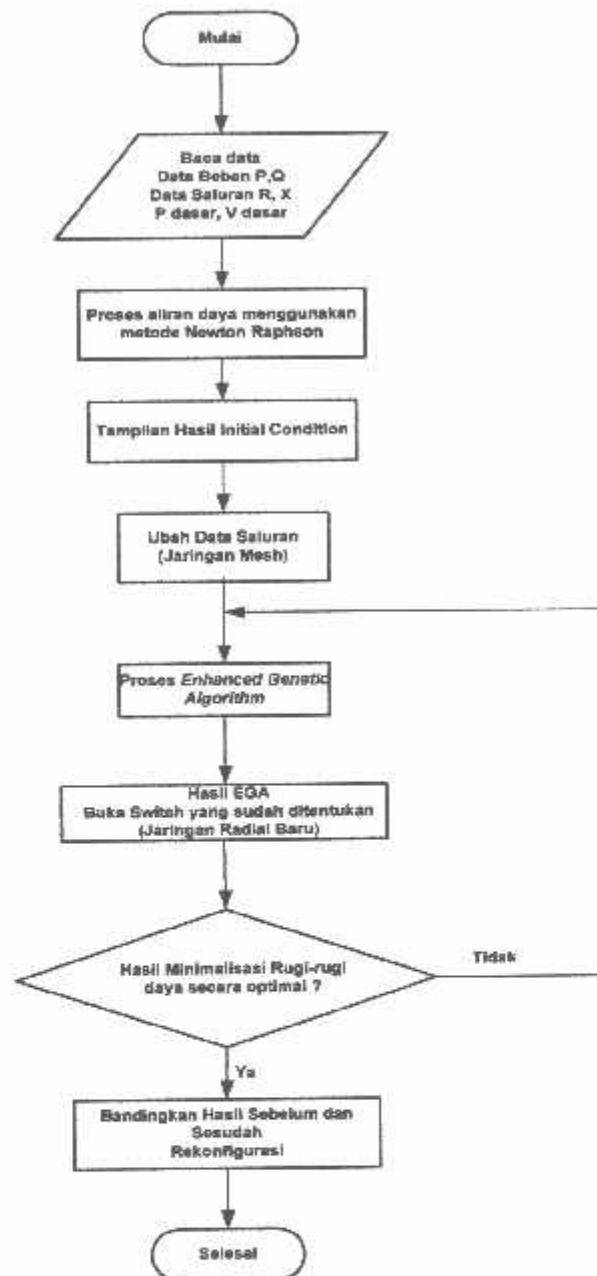
4.1.1. Algoritma Program

Urutan langkah-langkah pada program komputer yang digunakan dapat dilihat pada algoritma program berikut :

1. Masukkan data-data jaringan distribusi PLN.
2. Menjalankan proses *load flow* menggunakan metode *Newton Rhapson*.
3. Tampilkan hasil *initial condition*.
4. Menutup semua *tie switch* pada jaringan dan mengubah menjadi jaringan Mesh.

5. Menjalankan proses dengan metode *Enhanced Genetic Algorithm*.
 6. Hasil EGA kombinasi *switch* yang sudah ditentukan (jaringan radial baru).
 7. Menampilkan hasil minimalisasi rugi-rugi EGA.
 8. Bandingkan hasil sebelum rekonfigurasi dan sesudah rekonfigurasi.
 9. Selesai.
-

4.1.2. Flowchart Program



Gambar 4-1
Diagram Alir Program

4.1.3. Algoritma Metode Penyelesaian Masalah

Algoritma program penyelesaian masalah rekonfigurasi jaringan distribusi dengan pendekatan multi-objektif berbasis fuzzy dengan menggunakan metode Enhanced Genetic Algorithm.

1. Memasukan inputan data beban yang meliputi tegangan (V), daya aktif (P), daya reaktif (Q), dan data konfigurasi sistem
2. Memasukkan variabel-variabel status on atau off dari switch-switch yang dirubah ke biner, untuk ditentukan didalam rekonfigurasi
3. Open = 0, Close = 1
4. Melakukan proses Inisialisasi dan menentukan parameter
5. Membuat populasi baru yang meliputi:
 - Melakukan proses seleksi
 - Melakukan proses elitism
 - Melakukan proses crossover
 - Melakukan proses mutasi
6. Melakukan proses pengkodean dan memeriksa jaringan radial
7. Melakukan proses fungsi kecocokan untuk di evaluasi

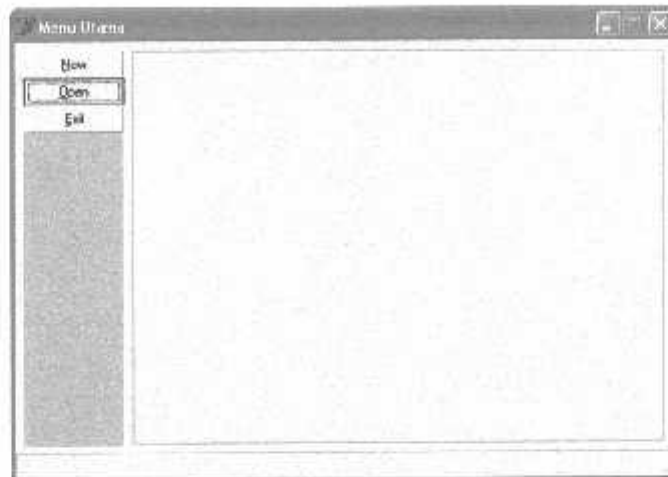
$$f = M = w_1 \mu_p + w_2 \sum_{i=1}^{Nbh} \mu_{vi} + w_3 \sum_{i=1}^{Nbh} \mu_{di} + w_4 \mu_s$$
8. Perintah berhenti setelah proses evaluasi terpenuhi
9. Jika "Tidak" maka kembali kelangkah 5
10. Jika "Ya" maka mencetak hasil keluaran dan selesai

4.1.4. Flow Chart *Enhanced Genetic Algorithm*

Gambar 4-2
Diagram Alir *Enhanced Genetic Algorithm*

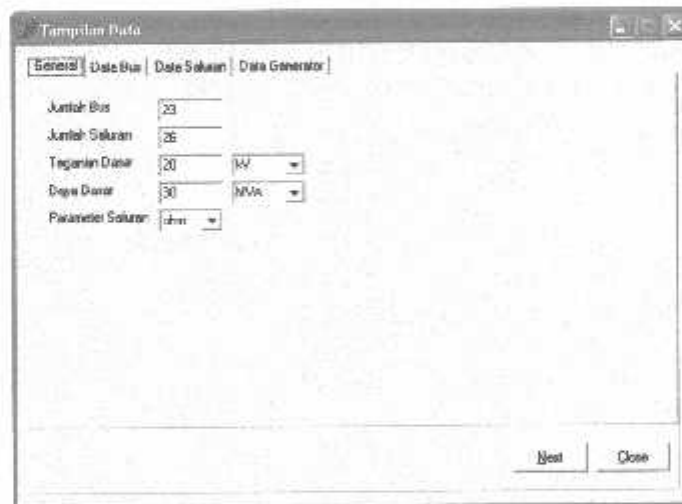
4.2. Analisa Aliran Daya Menggunakan Metode *Newton Rhapson*

Untuk menjalankan program buka menu utama



Gambar 4-3
Menu Utama Program

Setelah itu tekan Open untuk memasukkan file yang sudah tersimpan dan akan muncul data *general* (umum) dari GI Blimbing.



Gambar 4-4
Inputan Program Data General GI Blimbing

Dalam skripsi ini data yang digunakan adalah penyulang-penyulang yang keluar dari Trafo II GI Blimbing Malang. Alasan pemilihan data tersebut karena output dari trafo II GI Blimbing mensuplai daerah yang kepadatan bebannya sangat bervariasi.

Data teknis dari Trafo II G.I Blimbing Malang adalah sebagai berikut :

Jumlah Bus	: 23
Jumlah Saluran	: 26
Tegangan Dasar	: 20 kV
Daya Dasar	: 30 MVA
Penghantar SUTM 20 kV	: AAAC 150 mm ²
Impedansi Saluran	: 0,2162 + j 0,3305 Ω / km
	: 0,3949 < 5681°
ϕ	: 0,85

Data beban untuk masing-masing penyulang dapat dilihat pada lampiran dalam tabel. Dari lampiran, daya P (MW) dan Q (MVAR) dari bus penyulang dapat dihitung nilai P (MW), Q (MVAR), R (Ω) dan X (Ω) sebagai berikut :

$$\begin{aligned}
 P &= 1772,97 \times \cos 0,85 &= 1772,97 \times 0,85 = 1507,82 \text{ KW} \\
 Q &= 1772,97 \times \sin 0,85 &= 1772,97 \times 0,52 = 933,97 \text{ KVAR} \\
 R \text{ dan } X &= 5,616 \text{ Km} \times Z \text{ saluran} &= 5,616 \times 0,3949 < 56,81^\circ \\
 & &= 2,22 < 56,81^\circ \\
 & &= 1,214 + j 1.856
 \end{aligned}$$

Dengan perhitungan yang sama maka masing-masing bus akan diperoleh hasil seperti pada tabel 4-1 berikut :

Tabel 4-1
Daya dan Impedansi Saluran Jaringan Distribusi
G.I Blimbing Malang

BUS PANGKAL	BUS UJUNG	P BUS UJUNG (MW)	Q BUS UJUNG (MVAR)	JARAK (KM)	R (Ω)	X (Ω)
1	2	3	4	5	6	7
1	2	1.50702	0.93397	5,616	1,2142	1,8560
2	3	1.02550	0.63555	4,085	0,8832	1,3501
3	4	1.37303	0.85093	5,040	1,0896	1,6657
4	5	0.77738	0.48177	3,897	0,8425	1,2880
5	6	0.17770	0.11013	0,677	0,1464	0,2238
6	7	0.61197	0.37927	1,678	0,3628	0,5546
1	8	0.27211	0.16864	1,381	0,2986	0,4564
8	9	0.68660	0.42551	1,194	0,2581	0,3946
9	10	0.29704	0.18409	1,016	0,2197	0,3358
10	11	0.62638	0.38820	1,492	0,3226	0,4931
1	12	0.63868	0.39582	2,474	0,5349	0,8177
12	13	2.37357	1.47100	3,443	0,7444	1,1379
12	14	0.14518	0.08997	1,646	0,3559	0,5440
14	15	0.48876	0.30291	1,183	0,2558	0,3910
14	16	1.54462	0.95727	2,745	0,5935	0,9072
16	17	0.04781	0.02963	0,131	0,0283	0,0433
1	18	0.06178	0.03829	1,782	0,3853	0,5890
18	19	0.33578	0.20809	3,031	0,6553	1,0017
18	20	2.67903	1.66031	7,639	1,6516	2,5247
19	21	0.23324	0.14455	1,549	0,3349	0,5119
21	22	0.63915	0.39611	1,476	0,3191	0,4878
21	23	0.94831	0.58771	4,689	1,0138	1,5497
4	11	-	-	0,207	0,0448	0,0684
9	13	-	-	0,173	0,0374	0,0572
15	19	-	-	1,114	0,2408	0,3682
5	16	-	-	0,262	0,0566	0,0866

Keterangan : Bus 1 merupakan slack bus, bus 2 sampai dengan 23 merupakan bus beban.

Dari data pada tabel 4.1 (data jaringan distribusi) dimasukkan kedalam *software* rekonfigurasi jaringan, maka hasil dari perhitungan aliran daya menggunakan metode *Newton Raphson* sebelum rekonfigurasi dapat dilihat pada gambar 4-5.

Bus	Vr (pu)	Sudut (deg)	Pg (MW)	Qg (MVAR)	Pl (MW)	Ql (MVAR)	Saps (pu)	Type Bus
1	1.00000	0.00000	17.918	11.464	0.000	0.000	0.000	1
2	0.96530	-0.89585	0.000	0.000	1.507	0.934	0.000	3
3	0.94690	-1.39428	0.000	0.000	1.025	0.636	0.000	3
4	0.93001	-1.86741	0.000	0.000	1.373	0.851	0.000	3
5	0.92304	-2.06727	0.000	0.000	0.777	0.482	0.000	3
6	0.92243	-2.08504	0.000	0.000	0.178	0.110	0.000	3
7	0.92126	-2.11901	0.000	0.000	0.612	0.379	0.000	3
8	0.99728	-0.07461	0.000	0.000	0.272	0.169	0.000	3
9	0.99528	-0.13023	0.000	0.000	0.687	0.426	0.000	3
10	0.99431	-0.015786	0.000	0.000	0.297	0.154	0.000	3
11	0.99332	-0.018447	0.000	0.000	0.626	0.388	0.000	3
12	0.98602	-0.037014	0.000	0.000	0.639	0.396	0.000	3
13	0.97721	-0.060893	0.000	0.000	2.374	1.471	0.000	3
14	0.98207	-0.04669	0.000	0.000	0.145	0.090	0.000	3
15	0.98145	-0.049359	0.000	0.000	0.499	0.303	0.000	3

Gambar 4-5
Hasil Program Load Flow Awal

Dengan memasukkan data dari tabel 4-1 ke dalam program simulasi, maka akan diperoleh hasil seperti yang terdapat pada tabel 4-2 berikut ini :

Tabel 4-2
Hasil Program Aliran Daya Sebelum Rekonfigurasi

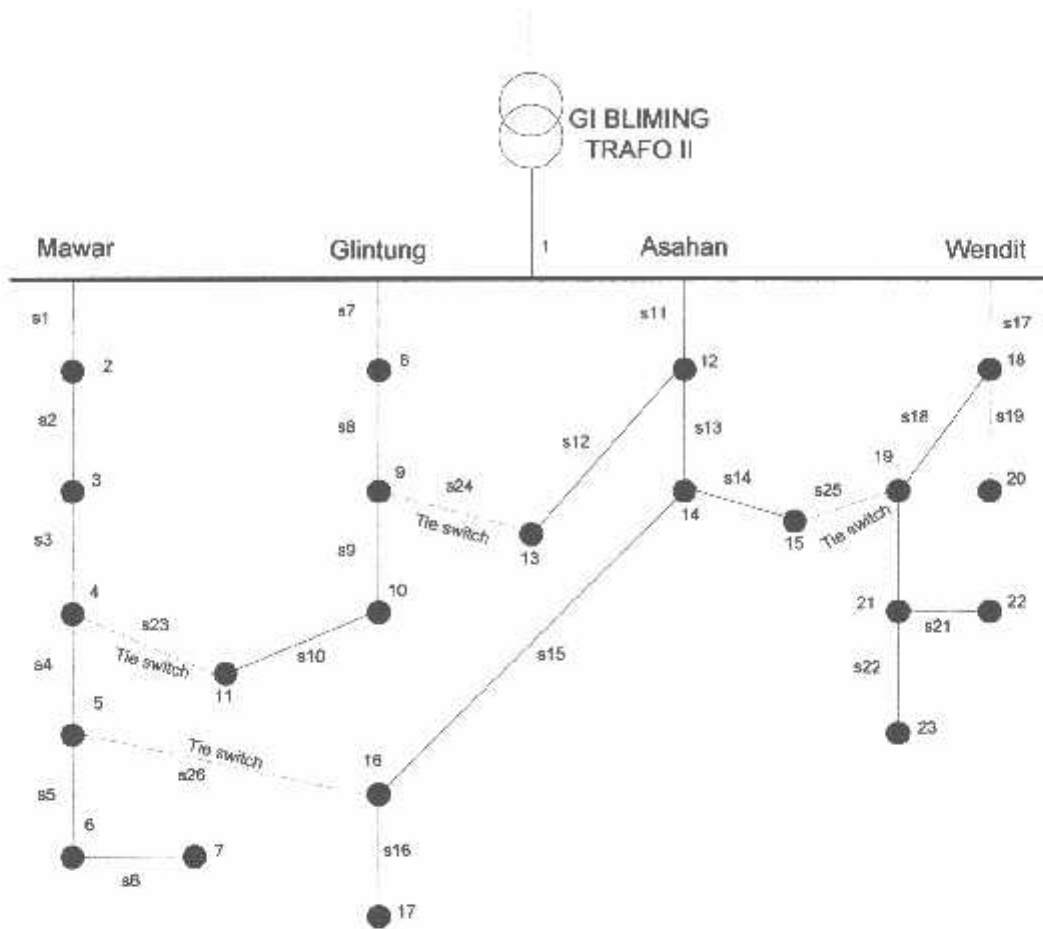
BUS	TEGANGAN		DAYA PEMBANGKITAN		DAYA PEMBEBANAN	
	V (pu)	Sudut ($^{\circ}$)	MW	MVAR	MW	MVAR
1	1.00000	0.00000	17.918	11.464	0.000	0.000
2	0.96530	-0.89585	0.000	0.000	1.507	0.934
3	0.94690	-1.39428	0.000	0.000	1.025	0.636
4	0.93001	-1.86741	0.000	0.000	1.373	0.851
5	0.92304	-2.06727	0.000	0.000	0.777	0.482
6	0.92243	-2.08504	0.000	0.000	0.178	0.110
7	0.92126	-2.11901	0.000	0.000	0.612	0.379
8	0.99728	-0.07461	0.000	0.000	0.272	0.169
9	0.99528	-0.13023	0.000	0.000	0.687	0.426
10	0.99431	-0.015786	0.000	0.000	0.297	0.154
11	0.99332	-0.018447	0.000	0.000	0.626	0.388
12	0.98602	-0.037014	0.000	0.000	0.639	0.396
13	0.97721	-0.060893	0.000	0.000	2.374	1.471
14	0.98207	-0.04669	0.000	0.000	0.145	0.090
15	0.98145	-0.049359	0.000	0.000	0.499	0.303
16	0.97736	-0.060494	0.000	0.000	1.545	0.957
17	0.97735	-0.060512	0.000	0.000	0.048	0.030
18	0.99051	-0.024800	0.000	0.000	0.062	0.038

19	0.98347	-0.043673	0.000	0.000	0.336	0.208
20	0.96821	-0.084870	0.000	0.000	2.679	1.660
21	0.98043	-0.051896	0.000	0.000	0.233	1.145
22	0.97942	-0.054661	0.000	0.000	0.639	0.396
23	0.97563	-0.064968	0.000	0.000	0.948	0.588

Untuk mengetahui kondisi awal jaringan (*initial condition*) yang terdapat pada gambar 4-6 dilakukan setelah proses *load flow* menggunakan metode *Newton Raphson*. Dan output Trafo II GI Blimbing terdiri :

1. Penyulang Mawar.
2. Penyulang Glintung.
3. Penyulang Asahan.
4. Penyulang Wendit.

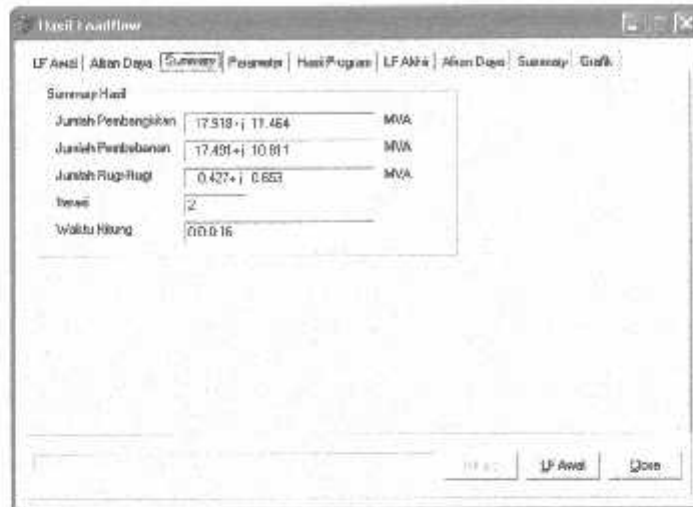
Gambar 4-6 merupakan gambar jaringan distribusi radial sebelum dilakukan rekonfigurasi yang terdiri dari empat *feeder* dengan dua puluh enam *switch NC (normally closed)* dan empat *tie switch NO (normally open)*, yaitu s23, s24, s25 dan s26. Dimana bus beban merupakan penyusunan dari trafo-trafo yang ada dalam saluran pada masing-masing penyulang yang dibatasi dengan adanya *switch*.



Gambar 4-6
Konfigurasi Jaringan Radial dari Single Line diagram G.I Blimbing Malang (Kondisi Awal)

- Ket : ● = Bus beban
 — = Switch NC (Normally Closed)
 - - - = tie switch NO (Normally Open)

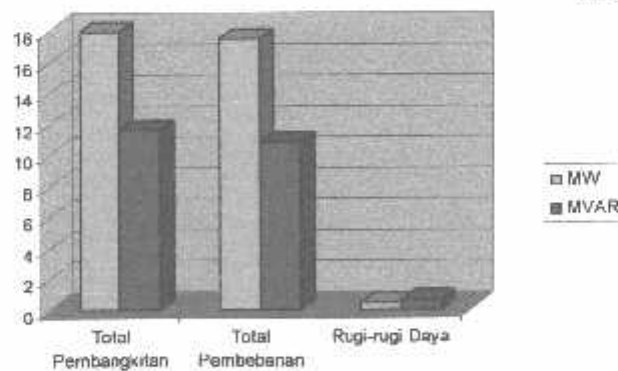
Untuk mengetahui kondisi awal total daya, total pembebanan dan total rugi-rugi daya yang terdapat pada GI tekan menu summary.



Gambar 4-7
Hasil Program Rugi-rugi Daya Sebelum Rekonfigurasi

Tabel 4-3
Hasil Program Rugi-rugi Daya Sebelum Rekonfigurasi

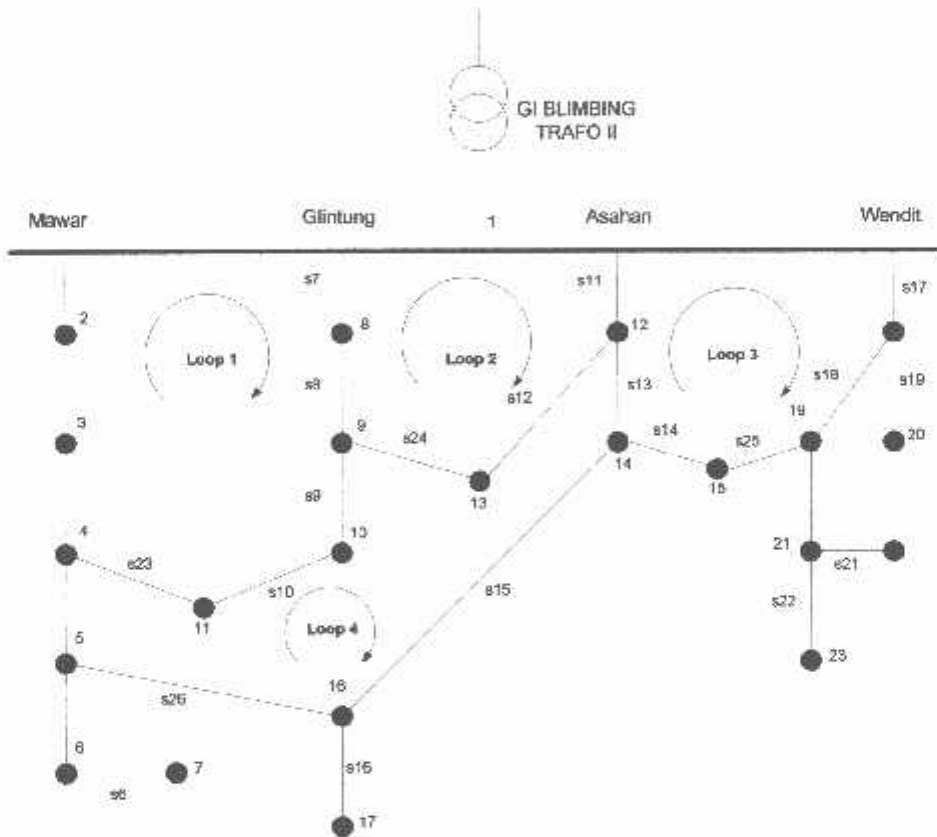
TOTAL PEMBANGKITAN		TOTAL PEMBEBANAN		TOTAL RUGI-RUGI DAYA	
P (MW)	Q (MVAR)	P (MW)	Q (MVAR)	P (MW)	Q (MVAR)
17.918	11.464	17.491	10.811	0.427	0.653



Grafik 4-1
Hasil Program Rugi-rugi Daya Sebelum Rekonfigurasi

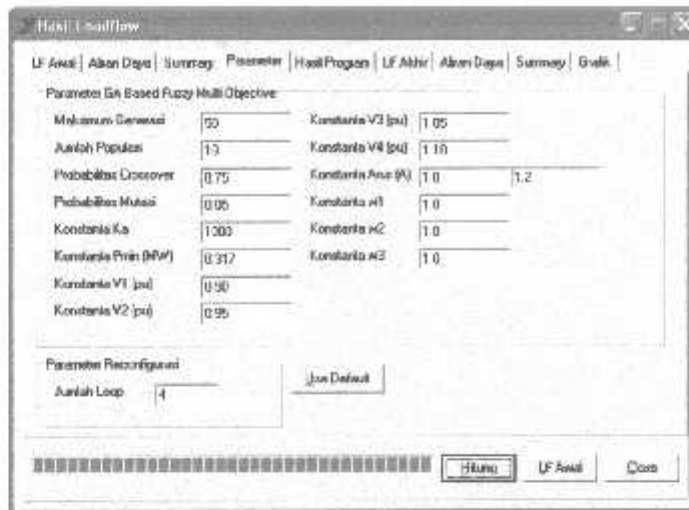
4.3. Analisa Rekonfigurasi Jaringan Menggunakan Metode *Enhanced Genetic Algorithm*

Dalam analisa rekonfigurasi jaringan semua *tie switch* (*normally open*) yang saling interkoneksi pada beberapa bus beban (*load bus*) baik dalam satu *feeder* maupun berlainan *feeder* ditutup (*normally closed*) sehingga jaringan radial seperti pada gambar 4-6 berubah menjadi jaringan mesh seperti terlihat seperti gambar 4-8, dari penutupan *tie switch* terdapat 4 *loop*.



Gambar 4-8
Konfigurasi Jaringan *mesh*

Memasukkan parameter *Enhanced Genetic Algorithm* yang akan diolah oleh program simulasi rekonfigurasi EGA untuk memperoleh hasil yang diinginkan seperti pada gambar 4-9



Gambar 4-9
Program Parameter EGA

Dengan memasukkan parameter pada program simulasi di lakukan untuk menghitung aliran daya menggunakan metode *Enhanced Genetic Algorithm* lalu tekan *hitung* maka load flow setelah menggunakan metode EGA dapat di ketahui dengan menekan menu *LF Akhir* yang dapat di lihat pada gambar 4-10.

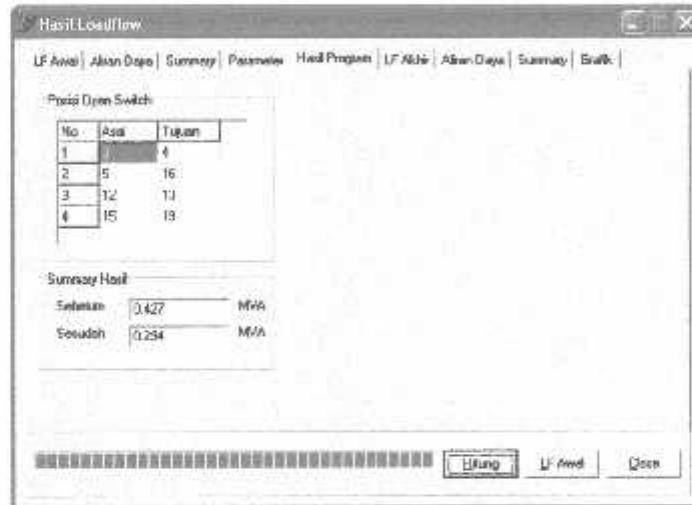
Gambar 4-10
Hasil Program Loadflow akhir Setelah Menggunakan EGA

Tabel 4-4
Hasil Program Aliran Daya Setelah Menggunakan EGA

BUS	TEGANGAN		DAYA PEMBANGKITAN		DAYA PEMBEBANAN	
	V (pu)	Sudut ($^{\circ}$)	MW	MVAR	MW	MVAR
1	1.00000	0.00000	17.769	11.235	0.000	0.000
2	0.98474	-0.40628	0.000	0.000	1.507	0.934
3	0.98024	-0.52828	0.000	0.000	1.025	0.636
4	0.96958	-0.81787	0.000	0.000	1.373	0.851
5	0.96289	-1.00199	0.000	0.000	0.777	0.482
6	0.96231	-1.01825	0.000	0.000	0.178	0.110
7	0.96118	-1.04952	0.000	0.000	0.612	0.379
8	0.98924	-0.28401	0.000	0.000	0.277	0.169
9	0.98030	-0.52513	0.000	0.000	0.687	0.426
10	0.97605	-0.64157	0.000	0.000	0.297	0.154
11	0.97024	-0.79968	0.000	0.000	0.626	0.388
12	0.99242	-0.20114	0.000	0.000	0.639	0.396
13	0.97986	-0.53718	0.000	0.000	2.374	1.471
14	0.98850	-0.30631	0.000	0.000	0.145	0.090
15	0.98788	-0.32299	0.000	0.000	0.489	0.303
16	0.98382	-0.43289	0.000	0.000	1.545	0.957
17	0.98381	-0.43307	0.000	0.000	0.048	0.030
18	0.99051	-0.24800	0.000	0.000	0.062	0.038
19	0.98347	-0.43673	0.000	0.000	0.336	0.208
20	0.96821	-0.84870	0.000	0.000	2.679	1.660
21	0.98043	-0.51896	0.000	0.000	0.233	0.145
22	0.97942	-0.54661	0.000	0.000	0.693	0.396
23	0.97563	-0.64968	0.000	0.000	0.948	0.588

Hasil minimalisasi rugi-rugi daya yang diperoleh merupakan hasil terbaik dari minimalisasi rugi-rugi daya yang ada. Setelah melakukan beberapa training dengan mengubah - ubah parameter pada kolom parameter EGA seperti pada gambar 4-11.

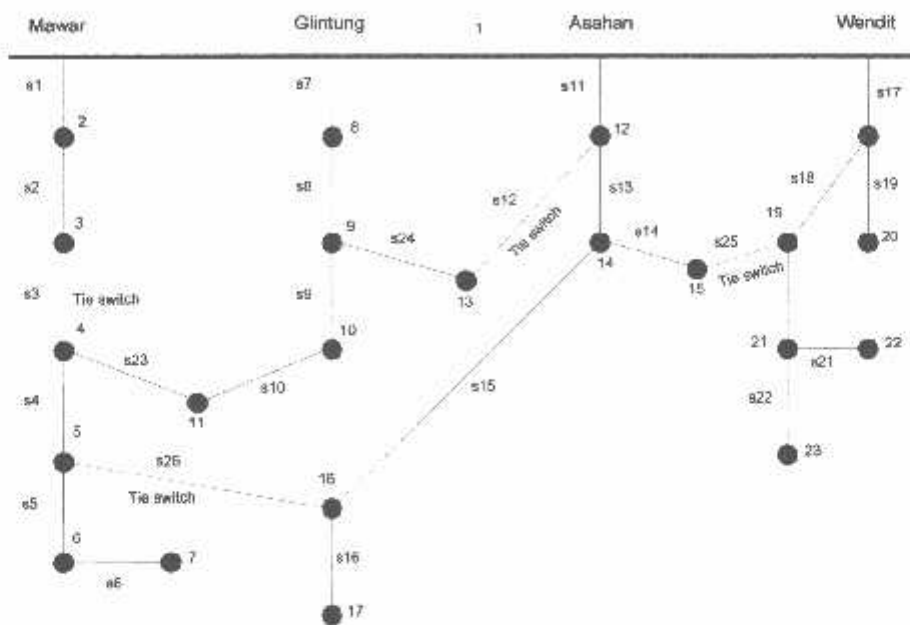
Kombinasi optimal *switching* didapat dengan durasi perhitungan sekitar 00:02 detik. Dan mencari konfigurasi jaringan yang baru dengan metode sesuai dengan hasil rekonfigurasi pada tabel 4-6 didapat kondisi *switch 3*, *switch 12*, *switch 25*, dan *switch 26* dalam status terbuka (*Normally Open*), sehingga jaringan *mesh* yang telah ada akan berubah menjadi konfigurasi jaringan radial *open loop* baru seperti pada gambar 4-13.



Gambar 4-12
Hasil simulasi kombinasi switch setelah rekonfigurasi

Tabel 4-6
Hasil Kombinasi Optimal *Switch*

No.	Sebelum Rekonfigurasi			Setelah Rekonfigurasi		
	Bus Pangkal	Bus Ujung	Posisi <i>switch open</i>	Bus Pangkal	Bus Ujung	Posisi <i>switch open</i>
1	4	11	s23	3	4	s3
2	9	13	s24	5	16	s26
3	15	19	s25	12	13	s12
4	5	16	s26	15	19	s25



Gambar 4-13
Konfigurasi Jaringan Radial Setelah Rekonfigurasi

Konfigurasi jaringan radial *open loop* setelah dilakukan rekonfigurasi tersebut merupakan konfigurasi jaringan radial *open loop* terbaik dari berbagai kemungkinan konfigurasi jaringan yang ada dengan menggunakan parameter *Enhanced Genetic Algorithm* yang berbeda dengan parameter yang telah digunakan sebelumnya.

4.4. Analisis Perbandingan Sebelum dan Setelah Rekonfigurasi

Menggunakan Metode *Enhanced Genetic Algorithm*

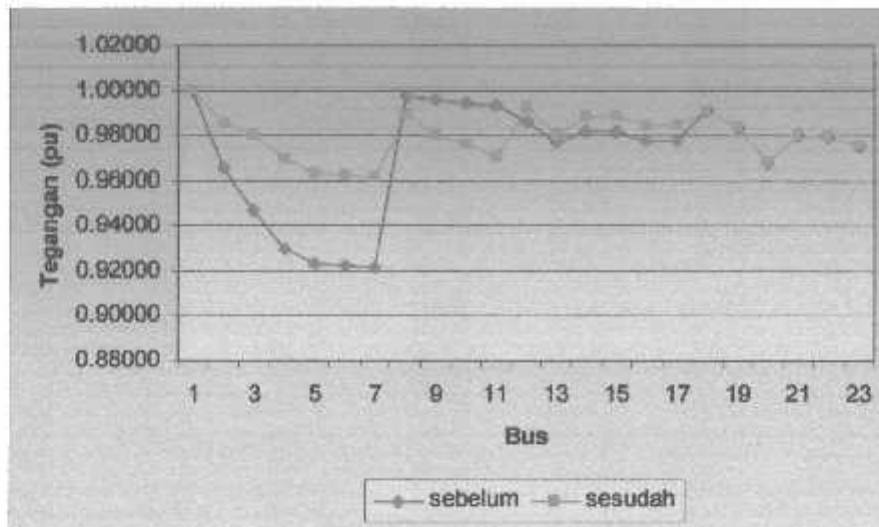
4.4.1. Analisis Simulasi Tegangan dan Sudut Fasa Tegangan

Pada proses perhitungan, menggunakan data pada tabel hasil perhitungan daya dan impedansi saluran jaringan distribusi G.I Blimbing Malang. Dimana nilai dasar yang digunakan adalah 20 kV untuk tegangan dan 30 MVA untuk

daya. Dengan mengacu pada 4 penyulang GI Blimbing Malang diantaranya Penyulang Mawar, Asahan, Wendit dan Glintung. Berikut ini adalah tabel 4-7 perbandingan tingkat tegangan dan sudut tegangan pada tiap bus.

Tabel 4-7
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

BUS	TEGANGAN Sebelum Rekonfigurasi		TEGANGAN Setelah Rekonfigurasi	
	V (pu)	Sudut ($^{\circ}$)	V (pu)	Sudut ($^{\circ}$)
1	1.00000	0.00000	1.00000	0.00000
2	0.96530	-0.89585	0.98474	-0.40628
3	0.94690	-1.39428	0.98024	-0.52828
4	0.93001	-1.86741	0.96958	-0.81787
5	0.92304	-2.06727	0.96289	-1.00199
6	0.92243	-2.08504	0.96231	-1.01825
7	0.92126	-2.11901	0.96118	-1.04952
8	0.99728	-0.07461	0.98924	-0.28401
9	0.99528	-0.13023	0.98030	-0.52513
10	0.99431	-0.015786	0.97605	-0.64157
11	0.99332	-0.018447	0.97024	-0.79968
12	0.98602	-0.037014	0.99242	-0.20114
13	0.97721	-0.060893	0.97986	-0.53718
14	0.98207	-0.04669	0.98850	-0.30631
15	0.98145	-0.049359	0.98788	-0.32299
16	0.97736	-0.060494	0.98382	-0.43289
17	0.97735	-0.060512	0.98381	-0.43307
18	0.99051	-0.024800	0.99051	-0.24800
19	0.98347	-0.043673	0.98347	-0.43673
20	0.96821	-0.084870	0.96821	-0.84870
21	0.98043	-0.051896	0.98043	-0.51896
22	0.97942	-0.054661	0.97942	-0.54661
23	0.97563	-0.064968	0.97563	-0.64968



Grafik 4-3

Grafik Perbandingan Tegangan Sebelum dan Setelah Rekonfigurasi

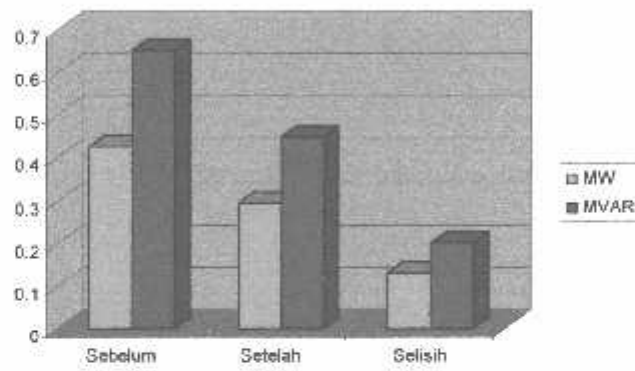
Dari hasil grafik 4-3, perbandingan setelah rekonfigurasi dapat kita ketahui bahwa nilai tegangan terendah pada saat sebelum rekonfigurasi sebesar 0.92126 pu yang terdapat pada bus 7 sedangkan setelah dilakukan rekonfigurasi nilai tegangan terendah sebesar 0.96118 pu yang terdapat pada bus 7 atau dengan kata lain setelah rekonfigurasi nilai tegangan mengalami kenaikan.

4.4.2. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran

Total rugi-rugi daya saluran sebelum rekonfigurasi adalah 0.427 MW dan 0.653 MVAR, sedangkan total rugi-rugi daya saluran setelah rekonfigurasi adalah 0.294 MW dan 0.450 MVAR. Dari hasil perhitungan dapat dilihat bahwa selisih total rugi-rugi daya turun sebesar 0.133 MW dan 0.203 MVAR. Sehingga rugi-rugi daya dapat ditekan sebesar 31,15 % yang terdapat pada tabel 4-8.

Tabel 4-8.
Tabel Perbandingan Rugi-rugi Daya Sebelum dan Setelah Rekonfigurasi

Total Rugi-rugi Daya Sebelum Rekonfigurasi		Total Rugi-rugi Daya Sesudah Rekonfigurasi		Selisih Rugi-rugi Daya Setelah Rekonfigurasi	
MW	MVAR	MW	MVAR	MW	MVAR
0.427	0.653	0.294	0.450	0.133	0.203



Grafik 4-4
Perbandingan Rugi-rugi Daya Sebelum dan Setelah Rekonfigurasi

BAB V

KESIMPULAN

5.1. Kesimpulan

Dari hasil analisis dapat di peroleh beberapa kesimpulan sebagai berikut :

1. Total rugi-rugi daya turun dari 0.427 MW dan 0.653 MVAR menjadi 0.294 MW dan 0.450 MVAR terjadi optimasi sebesar 0.133 MW dan 0.203 MVAR.
2. Nilai tegangan terendah pada saat sebelum rekonfigurasi sebesar 0.92126 pu yang terdapat pada bus 7 sedangkan setelah dilakukan rekonfigurasi nilai tegangan terendah sebesar 0.96118 pu yang terdapat pada bus 7 atau dengan kata lain setelah dilakukan rekonfigurasi, nilai tegangan mengalami kenaikan tegangan pada tiap-tiap bus.
3. Berdasarkan hasil program rekonfigurasi jaringan dengan menggunakan metode EGA, penentuan kombinasi switchnya adalah pada kondisi switch s3 dan switch s12 dalam status terbuka (NO). Dan kombinasi switch dalam status tertutup (NC) adalah switch s23 dan switch s24.

5.2. Saran

Penggunaan metode *Enhanced Genetic Algorithm* (EGA) untuk menyelesaikan permasalahan rekonfigurasi jaringan masih perlu dikembangkan lagi terhadap sistem jaringan yang lebih luas dan penggunaan waktu *komputasi* yang lebih cepat sehingga nantinya dapat digunakan lebih efektif dan efisien dalam pemecahan permasalahan rekonfigurasi jaringan.

DAFTAR PUSTAKA

- [1] Y.C. Huang, (2002) “ *Enhanced Genetic Algorithm-Based Fuzzy Multi Objective Approach to Distribution Network Reconfiguration* ” Proc. Inst. Elect. Eng., Vol. 149, no.5, pp.615-620.
 - [2] Djiteng Marsudi, (1990) ,” *Operasi Tenaga Listrik* ”, Balai Penerbit dan humas ISTN. Bumi Srengseng Indah P. Minggu Jakarta Selatan.
 - [3] Hasan Basri, Ir, (1996) “ *Sistem Distribusi Tenaga Listrik* ”. Balai Penerbit dan Humas IST Bumi Srengseng Indah P, Minggu, Jakarta Selatan.
 - [4] William D.Stevenson, Jr, (1996) “ *Analisis Sistem Tenaga Listrik* ”, edisi keempat, Erlangga, Jakarta.
 - [5] Dewi, Sri Kusuma, (2003) ”*Artificial Intellgence (Teknik dan Aplikasinya)*” Graha Ilmu, Yogyakarta.
-

LAMPIRAN

I

- ↳ LISTING PROGRAM
 - ↳ DATA SALURAN
 - ↳ SINGLE LINE
 - ↳ JOURNAL
-

```
program ReconEGA;
```

```
uses
```

```
  Forms,  
  uAbout in 'uAbout.pas' {frmAbout},  
  uHasil in 'uHasil.pas' {frmHasil},  
  uInputLFChild in 'uInputLFChild.pas',  
  uMenu in 'uMenu.pas' {frmMenu},  
  uVarGlobal in 'uVarGlobal.pas',  
  uFitness in 'Shared\uFitness.pas',  
  uGenetic in 'Shared\uGenetic.pas',  
  uRecursive in 'Shared\uRecursive.pas';
```

```
{$R *.res}
```

```
begin
```

```
  Application.Initialize;  
  Application.CreateForm(TfrmMenu, frmMenu);  
  Application.CreateForm(TfrmHasil, frmHasil);  
  Application.CreateForm(TfrmAbout, frmAbout);  
  Application.Run;
```

```
end.
```

```
unit uAbout;

interface

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

type
  TfrmAbout = class(TForm)
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmAbout: TfrmAbout;

implementation

{$R *.dfm}

end.
```

```

unit uFitness;

interface

uses uUtils, uComplex, uLoadflow, uNewtonRaphson, uRecursive,
    SysUtils, Classes, uLasil, uMatrix;

type
    TIndividu=record
        chrom:iArr1;
        fitness:double;
    end;

    TPopulasi=array of TIndividu;

    TFitness=class
    private
        FNbus, FNsal, FNLoop:integer;
        FPlossMin, FV1, FV2, FV3, FV4, FI1, FI2, Fw1, Fw2, Fw3:double;
        FVb:TBatas;
        FV, FSg, FSL:CxArr1;
        FCap:dArr1;
        FTypBus:iArr1;
        FZ, FTp:CxArr2;
        FLC, FTz, FCapSal:dArr2;
        FSwith:TSwithArr1;
        FBestIndi:TIndividu;
        function getIndividu(const rIndi:TIndividu):TIndividu;
        function CekStopRecomb(const rData:iArr1):boolean;
        function isNumeric(const rStr:Char):boolean;
        function isSameData(const rStr1, rStr2:string):boolean;
        function isSameDataList(var rList:TStringList;
            const rStr:string):boolean;
        function CalcVPinality(const rV:CxArr1):integer;
        function getChrom:iArr1;
        function MFPowerloss(const rPloss:double):double;
        function MFBusV(const rV:CxArr1):double;
        function MFARus(const rArus:CxArr2):double;
        function CalcFitness(const rData:iArr1):double;overload;
    public
        constructor Create(const rNLoop:integer);overload;
        constructor Create(const rNLoop:integer;
            const rPlossMin, rV1, rV2, rV3, rV4,
            rI1, rI2, rW1, rW2, rW3:double);overload;
        procedure CreateCombinasi(var rList:TStringList;
            var rCount:integer);
        function CalcFitness(const rData:bArr1):double;overload;
        function CalcFitnessFuzzy(const rData:bArr1):double;
        function GetSwith(const rData:iArr1):TSwithArr1;
        function DecodeStrToData(const rStr:string):iArr1;
        function DecodeDataToStr(const rData:iArr1):string;
        property BestChrom:iArr1 read getChrom;
    end;

var gFit:TFitness;

implementation

constructor TFitness.Create(const rNLoop:integer);
var i, dari, ke:integer;
begin
    inherited Create;
    FNLoop:=rNLoop;
    DecodeCommDataToLFData(gBus, FNbus, FNsal, FV, FSg, FSL, FCap, FTypBus,

```

```

gBranch, FZ, FTp, FLc, FTr);
SetLength(FCapSal, FNbus, FNbus);
for i:=0 to FNsal-1 do
begin
  dari:=gBranch[i].dari-1;
  ke:=gBranch[i].ke-1;
  FCapSal[dari, ke]:=gBranch[i].KapSal;
end;
FSwith:=TnitSwith(FZ);
FVb.min:=0.95;
FVb.max:=1.05;
end;

constructor TFitness.Create(const rNLoop:integer;
  const rPlossMin, rV1, rV2, rV3, rV4, rI1, rI2, rw1, rw2, rw3:double);
var i, dari, ke:integer;
begin
  inherited Create;
  FNLoop:=rNLoop;
  FPlossMin:=rPlossMin;
  FV1:=rV1;
  FV2:=rV2;
  FV3:=rV3;
  FV4:=rV4;
  FI1:=rI1;
  FI2:=rI2;
  Fw1:=rw1;
  Fw2:=rw2;
  Fw3:=rw3;
  DecodeCommDataToTData(gBus, FNbus, FNsal, FV, FSg, FSL, FCap, FTypBus,
  gBranch, FZ, FTp, FLc, FTr);
  SetLength(FCapSal, FNbus, FNbus);
  for i:=0 to FNsal-1 do
  begin
    dari:=gBranch[i].dari-1;
    ke:=gBranch[i].ke-1;
    FCapSal[dari, ke]:=gBranch[i].KapSal;
    FCapSal[ke, dari]:=gBranch[i].KapSal;
  end;
  FSwith:=TnitSwith(FZ);
  FVb.min:=0.95;
  FVb.max:=1.05;
end;

function TFitness.getIndividu(const rIndi:TIndividu):TIndividu;
var i:integer;
begin
  SetLength(result.chrom, FNLoop);
  for i:=0 to FNLoop-1 do
  begin
    result.chrom[i]:=rIndi.chrom[i];
  end;
  result.fitness:=rIndi.fitness;
end;

function TFitness.GetSwith(const rData:iArr1):TSwithArr1;
var i, NData:integer;
begin
  NData:=high(rData)+1;
  SetLength(result, NData);
  for i:=0 to NData-1 do
  begin
    if (rData[i]<0) or (rData[i]>=FNsal) then
    begin

```

```

        raise Exception.Create('Melebihi Indeks Matrik!');
    end;
    result[i].dari:=Fswith[rData[i]].dari;
    result[i].ke:=Fswith[rData[i]].ke;
end;
end;

function TFitness.CekStopRecomb(const rData:iArr1):boolean;
var i:integer;
begin
    result:=true;
    for i:=0 to FNLoop-1 do
    begin
        if rData[i]<>(FNsal-1) then
        begin
            result:=false;
            break;
        end;
    end;
end;

function TFitness.isNumeric(const rStr:Char):boolean;
begin
    result:=false;
    if rStr in ['1','2','3','4','5','6','7','8','9','0'] then
    begin
        result:=true;
    end;
end;

function TFitness.DecodeStrToData(const rStr:string):iArr1;
var i,sa:integer;
    st,str:string;
begin
    str:=trin(rStr);
    SetLength(result,4);
    sa:=0;
    for i:=1 to length(str)+1 do
    begin
        if isNumeric(str[i])=true then
        begin
            st:=st+str[i];
        end
        else
        begin
            result[sa]:=StrToInt(st);
            inc(sa);
            st:='';
        end;
    end;
end;

function TFitness.DecodeDataToStr(const rData:iArr1):string;
var i,NData:integer;
begin
    NData:=high(rData)+1;
    result:='';
    for i:=0 to NData-1 do
    begin
        result:=result+IntToStr(rData[i])+' ';
    end;
end;

function TFitness.isSameData(const rStr1,rStr2:string):boolean;

```

```

var Data1,Data2:iArr1;
    i,j,Count:integer;
begin
    Data1:=DecodeStrToData(rStr1);
    Data2:=DecodeStrToData(rStr2);
    result:=false;
    Count:=0;
    for i:=0 to high(Data1) do
    begin
        for j:=0 to high(Data2) do
        begin
            if Data1[i]=Data2[j] then
            begin
                inc(Count);
                break;
            end;
        end;
    end;
    if Count=high(Data1)+1 then result:=true;
end;

function TFitness.isSameDataList(var rList:TStringList;
    const rstr:string):boolean;
var i:integer;
begin
    result:=false;
    for i:=0 to rList.Count-1 do
    begin
        if isSameData(rStr,rList.Strings[i])=true then
        begin
            result:=true;
            break;
        end;
    end;
end;

function TFitness.CalcVPInalty(const rV:CxArr1):integer;
var i:integer;
begin
    result:=0;
    for i:=0 to FNbus-1 do
    begin
        if rV[i].real>1.05 then inc(result);
        if rV[i].real<0.95 then inc(result);
    end;
end;

function TFitness.MFPowerLoss(const rPloss:double):double;
begin
    result:=exp(-(rPloss-FPlossMin)/FPlossMin);
end;

function TFitness.MFBusV(const rV:CxArr1):double;
var i:integer;
begin
    result:=0;
    for i:=0 to FNbus-1 do
    begin
        if rV[i].real<FV1 then
        begin
            result:=result+0;
        end
        else if (rV[i].real>=FV1) and (rV[i].real<FV2) then
        begin

```

```

    result:=result+(rV[i].real-FV1)/(FV2-FV1);
end
else if (rV[i].real>=FV2) and (rV[i].real<FV3) then
begin
    result:=result+1.0;
end
else if rV[i].real>FV4 then
begin
    result:=result-0;
end;
end;
result:=result/FNbus;
end;

```

```

function TFitness.MFARus(const rArus:CxArr2):double;
var i,j,sa:integer;
    absArus:double;
begin
    result:=0;
    sa:=0;
    for i:=0 to FNbus-1 do
    begin
        for j:=0 to FNbus-1 do
        begin
            if rArus[i,j].imag<>0 then
            begin
                sa:=sa+1;
                absArus:=sqrt(sqr(rArus[i,j].real)+sqr(rArus[i,j].imag))/
                    FCapSa1[i,j];
                if absArus<FI1 then
                begin
                    result:=result+1.0;
                end
                else if (absArus>=FI1) and (absArus<FI2) then
                begin
                    result:=result+(FI2-absArus)/(FI2-FI1);
                end
                else if absArus>=FI2 then
                begin
                    result:=result+0;
                end;
            end;
        end;
    end;
    result:=result/sa;
end;

```

```

function TFitness.CalcFitness(const rData:iArr1):double;
var Vpin:integer;
    V,Sg:CxArr1;
    Zb,Alir,Arus:CxArr2;
    sw:TSwitchArr1;
begin
    sw:=GetSwitch(rData);
    Zb:=BuatZBaru(FZ,sw);
    V:=CopyMatrix(FV);
    Sg:=CopyMatrix(FSg);
    NewtonRaphson(gParamLF,V,Sg,FSI,FCap,FTypBus,
    Zb,FTp,Alir,Arus,Flc,FTtr);
    Vpin:=CalcVPinality(FV);
    result:=-10*1000-1000*FNbus;
    if gParamLF.Iterasi<gParamLF.MaxIterasi then
    begin
        result:=gParamLF.SumLoss.real-1000*Vpin;
    end;
end;

```

```

end;
end;

function TFitness.CalcFitness(const rData:bArr1):double;
var Vpin:integer;
    V,Sg:CxArr1;
    Zb,Alir,Arus:CxArr2;
    TesRadial:boolean;
begin
    Zb:=BuatZBaru(FZ,rData);
    TesRadial:=TesJaringan(Zb);
    if TesRadial=true then
    begin
        V:=CopyMatrix(FV);
        Sg:=CopyMatrix(FSg);
        NewtonRaphson(gParamLF,V,Sg,FSL,FCap,FTypBus,
            Zb,FTp,Alir,Arus,FLc,FTtr);
        Vpin:=CalcVPinalty(FV);
        result:=gParamLF.SumLoss.real+1000*Vpin;
    end
    else
    begin
        result:=(high(FV)+5)*1000;
    end;
end;

function TFitness.CalcFitnessFuzzy(const rData:bArr1):double;
var V,Sg:CxArr1;
    Zb,Alir,Arus:CxArr2;
    TesRadial:boolean;
    uPloss,uV,uI:double;
begin
    Zb:=BuatZBaru(FZ,rData);
    TesRadial:=TesJaringan(Zb);
    if TesRadial=true then
    begin
        V:=CopyMatrix(FV);
        Sg:=CopyMatrix(FSg);
        NewtonRaphson(gParamLF,V,Sg,FSL,FCap,FTypBus,
            Zb,FTp,Alir,Arus,FLc,FTtr);
        if gParamLF.Iterasi<15 then
        begin
            uPloss:=-MFPowerLoss(gParamLF.SumLoss.real);
            uV:=MFBusV(V);
            uI:=MFArus(Arus);
            result:=Fw1*uPloss+Fw2*uV+Fw3*uI;
        end
        else
        begin
            result:=0.00001;
        end;
    end
    else
    begin
        result:=-0.00001;
    end;
end;

procedure TFitness.CreateCombinasi(var rList:TStringList;
    var rCount:integer);
var i:integer;
    CekRadial,CekCombinasi:boolean;
    str:string;
    sw:TSwitchArr1;

```

```

    Zb:CxArr2;
    Data:IArr1;
    CekIndi:TIndividu;
begin
    SetLength(Data, FNLoop);
    for i:=0 to FNLoop-1 do
    begin
        Data[i]:=0;
    end;
    rCount:=0;
    rList.Clear;
    repeat
        GenCombinasi(FNsal, Data, CekCombinasi);
        if CekCombinasi=true then
        begin
            sw:=GetSwith(Data);
            Zb:=BuatZBaru(FZ, sw);
            CekRadial:=TesJaringan(Zb);
            if (CekCombinasi=true) and (CekRadial=true) then
            begin
                str:=DecodeDataToStr(Data);
                if isSameDataList(rList, str)=false then
                begin
                    if rCount=0 then
                    begin
                        FBestIndi.chrom:=CopyMatrix(Data);
                        FBestIndi.fitness:=CalcFitness(Data);
                    end
                    else
                    begin
                        CekIndi.chrom:=CopyMatrix(Data);
                        CekIndi.fitness:=CalcFitness(Data);
                        if CekIndi.fitness<FBestIndi.fitness then
                        begin
                            FBestIndi:=getIndividu(CekIndi);
                        end;
                    end;
                    rList.Add(str);
                    inc(rCount);
                end;
            end;
        end;
        frmHasil.pbGen.StepBy(1);
    until CekStopRecomb(Data)=true;
end;

function TFitness.getChrom:IArr1;
var i:integer;
begin
    SetLength(result, FNLoop);
    for i:=0 to FNLoop-1 do
    begin
        result[i]:=FBestIndi.chrom[i];
    end;
end;

end.

```

```

unit uGenetic;

interface

uses uUtils, uFitness, uRandom, uHasil;

type
  TIndiBin1=record
    chrom:bArr1;
    fitness:double;
  end;

  TPopBin1=array of TIndiBin1;

  TGenetic=class
  private
    FMaxGen, FPopSize, FLength:integer;
    FCross, FPMutasi, FKa:double;
    function getMin:dArr1;
    function getAvg:dArr1;
    function getMax:dArr1;
  protected
    FMin, FAvg, FMax:dArr1;
    FRandom:TRandomu;
  public
    constructor Create(const rMaxGen, rPopSize, rLength:integer;
      const rCross, rPMutasi, rKa:double);
    destructor Destroy;override;
    property MaxGen:integer read FMaxGen write FMaxGen;
    property PopSize:integer read FPopSize write FPopSize;
    property Length:integer read FLength write FLength;
    property FCross:double read FCross write FCross;
    property PMutasi:double read FPMutasi write FPMutasi;
    property Ka:double read FKa write FKa;
    property Min:dArr1 read getMin;
    property Avg:dArr1 read getAvg;
    property Max:dArr1 read getMax;
  end;

  TGenRecon=class(TGenetic)
  private
    FNLoop:integer;
    FMin1, FAvg1, FMax1, FSumFitness:double;
    FParent, FChild:TPopBin1;
    FBestIndi:TIndiBin1;
    function getIndividu(const rIndi:TIndiBin1):TIndiBin1;
    function getBestChrom:bArr1;
    procedure RepairChrom(var rChrom:bArr1);
    procedure InitParent;
    procedure Statistik;
    function FindIndividuMax:TIndiBin1;
    function Seleksi:integer;
    function Mutasi(const rAllele:bcolean):bcolean;
    procedure Crossover(const rParent1, rParent2:bArr1;
      var rChild1, rChild2:bArr1);
    procedure Generasi;
    procedure Replikasi;
    procedure doHitung;
  public
    constructor Create(const rMaxGen, rPopSize, rLength, rNLoop:integer;
      const rCross, rPMutasi, rKa:double);
    function DecodeDataToIndi(const rData:iArr1;
      const rFitness:double):TIndiBin1;
    procedure DecodeIndiToData(const rIndi:TIndiBin1;

```

```

        var rData:iArr1;
        var rFitness:double);
    function getBestIndi:TIndiBin1;
    property BestChrom:bArr1 read getBestChrom;
end;

implementation

{ TGenetic }

//constructor
constructor TGenetic.Create(const rMaxGen,rPopSize,rLength:integer;
    const rPCross,rPMutasi,rKa:double);
begin
    inherited Create;
    FMaxGen:=rMaxGen;
    FPopSize:=rPopSize;
    FLength:=rLength;
    FPCross:=rPCross;
    FPMutasi:=rPMutasi;
    FKa:=rKa;
    SetLength(FMin,FMaxGen);
    SetLength(FAvg,FMaxGen);
    SetLength(FMax,FMaxGen);
    FRandom:=TRandomu.Create;
end;

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

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

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

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

```

```

    end;
end;

{ TGenRecon }

constructor TGenRecon.Create(const rMaxGen, rPopSize, rLength, rNloop:integer;
    const rFCross, rPMutasi, rKa:double);
begin
    inherited Create(rMaxGen, rPopSize, rLength, rFCross, rPMutasi, rKa);
    FNloop:=rNloop;
end;

function TGenRecon.getIndividu(const rIndi:TIndiBin1):TIndiBin1;
var i:integer;
begin
    SetLength(result.chrom, FLength);
    for i:=0 to FLength-1 do
    begin
        result.chrom[i]:=rIndi.chrom[i];
    end;
    result.fitness:=rIndi.fitness;
end;

function TGenRecon.getBestChrom:tArr1;
var i:integer;
begin
    SetLength(result, FLength);
    for i:=0 to FLength-1 do
    begin
        result[i]:=FBestIndu.chrom[i];
    end;
end;

procedure TGenRecon.RepairChrom(var rChrom:tArr1);
var i,sa,cu,no:integer;
begin
    sa:=0;
    for i:=0 to Length-1 do
    begin
        if rChrom[i]=true then
        begin
            inc(sa);
        end;
    end;
    if sa>FNloop then
    begin
        cu:=sa-FNloop;
        for i:=0 to cu-1 do
        begin
            repeat
                no:=FRandom.NextInt(0, Length-1);
            until rChrom[no]=true;
            rChrom[no]:=false;
        end;
    end
    else if sa<FNloop then
    begin
        cu:=FNloop-sa;
        for i:=0 to cu-1 do
        begin
            repeat
                no:=FRandom.NextInt(0, Length-1);
            until rChrom[no]=false;
            rChrom[no]:=true;
        end;
    end;
end;

```

```

    end;
  end;
end;

function TGenRecon.FindIndividuMax:TIndiBin1;
var i:integer;
begin
  result:=getIndividu(FParent[0]);
  for i:=1 to PopSize-1 do
  begin
    if result.fitness<FParent[i].fitness then
    begin
      result:=getIndividu(FParent[i]);
    end;
  end;
end;

procedure TGenRecon.InitParent;
var i,j,no:integer;
begin
  SetLength(FParent,PopSize);
  SetLength(FChild,PopSize);
  for i:=0 to PopSize-1 do
  begin
    SetLength(FParent[i].chrom,Length);
    SetLength(FChild[i].chrom,Length);
    for j:=0 to FNLoop-1 do
    begin
      repeat
        no:=FRandom.NextInt(0,Length-1);
      until FParent[i].chrom[no]=false;
      FParent[i].chrom[no]:=true;
    end;
    RepairChrom(FParent[i].chrom);
    //FParent[i].fitness:=Ka/gFit.CalcFitness(FParent[i].chrom);
    FParent[i].fitness:=gFit.CalcFitnessFuzzy(FParent[i].chrom);
  end;
  SetLength(FBestIndi.chrom,Length);
end;

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

function TGenRecon.Seleksi:integer;
var i:integer;
    sum,partsum:double;

```

```

begin
  i:=0;
  sum:=0;
  partsum:=FRandom.NextDouble*FSumFitness;
  repeat
    sum:=sum+FParent[i].fitness;
    inc(i);
  until (sum>partsum) or (i>=(PopSize-1));
  result:=i;
end;

function TGenRecon.Mutasi(const rAllele:boolean);boolean;
begin
  if FRandom.NextBoolean(PMutasi)=true then
  begin
    result:=not rAllele;
  end
  else
  begin
    result:=rAllele;
  end;
end;

procedure TGenRecon.Crossover(const rParent1,rParent2:bArr1;
  var rChild1,rChild2:bArr1);
var i,pos:integer;
begin
  if FRandom.NextBoolean(PCross)=true then
  begin
    pos:=FRandom.NextInt(0,(Length-2));
    for i:=0 to pos do
    begin
      rChild1[i]:=Mutasi(rParent1[i]);
      rChild2[i]:=Mutasi(rParent2[i]);
    end;
    for i:=pos+1 to Length-1 do
    begin
      rChild1[i]:=Mutasi(rParent2[i]);
      rChild2[i]:=Mutasi(rParent1[i]);
    end;
  end
  else
  begin
    for i:=0 to Length-1 do
    begin
      rChild1[i]:=Mutasi(rParent1[i]);
      rChild2[i]:=Mutasi(rParent2[i]);
    end;
  end;
end;

procedure TGenRecon.Generasi;
var i,matel,mate2:integer;
begin
  i:=0;
  repeat
    matel:=Seleksi;
    mate2:=Seleksi;
    Crossover(FParent[matel].chrom,FParent[mate2].chrom,
    FChild[i].chrom,FChild[i+1].chrom);
    RepairChrom(FChild[i].chrom);
    //FChild[i].fitness:=Ka/gFit.CalcFitness(FChild[i].chrom);
    FChild[i].fitness:=gFit.CalcFitnessFuzzy(FChild[i].chrom);
    RepairChrom(FChild[i+1].chrom);
  until

```

```

    //FChild[i+1].fitness:=Ka/gFit.CalcFitness(FChild[i+1].chrom);
    FChild[i+1].fitness:=gFit.CalcFitnessFuzzy(FChild[i+1].chrom);
    i:=i+2;
    frmHasil.pbGen.StepBy(1);
until i>=PopSize;
end;

procedure TGenRecon.Replikasi;
var i,pos:integer;
    tmpPop:TPopBin1;
begin
    SetLength(tmpPop,PopSize);
    for i:=0 to PopSize-1 do
    begin
        repeat
            pos:=FRandom.NextInt(0,(PopSize-1));
        until pos<>i;
        if FChild[i].fitness>FParent[pos].fitness then
        begin
            tmpPop[i]:=getIndividu(FChild[i]);
        end
        else
        begin
            tmpPop[i]:=getIndividu(FParent[pos]);
        end;
    end;
    for i:=0 to PopSize-1 do
    begin
        FParent[i]:=getIndividu(tmpPop[i]);
    end;
end;

procedure TGenRecon.doHitung;
var gen:integer;
    tmpIndi:TIndiBin1;
begin
    InitParent;
    Statistik;
    gen:=0;
    FBestIndi:=FindIndividuMax;
    repeat
        generasi;
        Replikasi;
        Statistik;
        tmpIndi:=FindIndividuMax;
        if FBestIndi.fitness<tmpIndi.fitness then
        begin
            FBestIndi:=getIndividu(tmpIndi);
        end;
        FMin[gen]:=FMin1;
        FAvg[gen]:=FAvg1;
        FMax[gen]:=FMax1;
        frmHasil.pbGen.StepBy(1);
        gen:=gen+1;
    until gen>=Maxgen;
end;

function TGenRecon.DecodeDataToIndi(const rData:iArr1;
    const rFitness:double):TIndiBin1;
var i:integer;
begin
    SetLength(result.chrom,Length);
    for i:=0 to high(rData) do
    begin

```

```
    result.chrom[rData[i]]:=true;
end;
result.fitness:=rFitness;
end;

procedure TGenRecon.DecodeIndiToData(const rIndi:TIndiBin1;
    var rData:iArr1;
    var rFitness:double);
var i,sa:integer;
begin
    SetLength(rData, FNloop);
    sa:=0;
    for i:=0 to Length-1 do
        begin
            if rIndi.chrom[i]=true then
                begin
                    rData[sa]:=i;
                    inc(sa);
                end;
            end;
        rFitness:=rIndi.fitness;
    end;

function TGenRecon.getBestIndi:TIndiBin1;
begin
    doHitung;
    result:=getIndividu(FBestIndi);
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)  
  TabSheet2: TTabSheet;  
  TabSheet3: TTabSheet;  
  TabSheet4: TTabSheet;  
  TabSheet9: TTabSheet;  
  Panel1: TPanel;  
  btnClose: TButton;  
  btnLFAwal: TButton;  
  TabSheet5: TTabSheet;  
  TabSheet6: TTabSheet;  
  TabSheet7: TTabSheet;  
  PageControl1: TPageControl;  
  btnHitung: TButton;  
  TabSheet8: TTabSheet;  
  fgBus: TStringGrid;  
  fgBranch: TStringGrid;  
  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;  
  GroupBox2: TGroupBox;  
  Label6: TLabel;  
  edtNLoop: TEdit;  
  GroupBox4: TGroupBox;  
  Label18: TLabel;  
  edtMaxGen: TEdit;  
  Label19: TLabel;  
  edtPopSize: TEdit;  
  Label20: TLabel;  
  edtPCross: TEdit;  
  Label21: TLabel;  
  edtPMutasi: TEdit;  
  Label23: TLabel;  
  edtKa: TEdit;  
  TabSheet10: TTabSheet;  
  Chart1: TChart;  
  Series1: TLineSeries;  
  Series2: TLineSeries;  
  GroupBox1: TGroupBox;  
  Label1: TLabel;  
  Label2: TLabel;  
  Label3: TLabel;  
  Label4: TLabel;  
  Label5: TLabel;
```

```

lblGen2: TLabel;
lblLoad2: TLabel;
lblLoss2: TLabel;
edtSumGen2: TEdit;
edtSumLoad2: TEdit;
edtSumLoss2: TEdit;
edtIterasi2: TEdit;
edtTime2: TEdit;
fgBus2: TStringGrid;
GroupBox5: TGroupBox;
fgOS: TStringGrid;
GroupBox3: TGroupBox;
Label16: TLabel;
Label17: TLabel;
lblLossP: TLabel;
lblLossN: TLabel;
edtSebelum: TEdit;
edtSesudah: TEdit;
btnUseDefault: TButton;
pbGen: TProgressBar;
fgBranch2: TStringGrid;
Label7: TLabel;
edtPlossMin: TEdit;
Label13: TLabel;
edtV1: TEdit;
edtV2: TEdit;
edtV3: TEdit;
edtV4: TEdit;
edtI1: TEdit;
edtI2: TEdit;
Label14: TLabel;
Label15: TLabel;
Label22: TLabel;
Label24: TLabel;
Label25: TLabel;
Label26: TLabel;
Label27: TLabel;
edtw1: TEdit;
edtw2: TEdit;
edtw3: TEdit;
procedure btnCloseClick(Sender: TObject);
procedure btnLFAwalClick(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure btnUseDefaultClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure edtNLoopChange(Sender: TObject);
private
  { Private declarations }
  function CariNoSaluran(const rText:string):integer;
public
  { Public declarations }
end;

var
  frmHasil: TfrmHasil;

implementation

uses uUtils, uLoadflow, uNewtonRaphson, uComplex, uTopology,
    uRecursive, uMatrix, uFitness, uVarGlobal, uGenetic;

{$R *.dfm}

var Nsal:integer;

```

```

CostSebelum, CostSesudah:double;

function TfrmHasil.CariNoSaluran(const rText:string):integer;
var i, count:integer;
begin
end;

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

procedure TfrmHasil.btnLFAwalClick(Sender: TObject);
var i, ia, ja, Nbus:integer;
    mulai, selesai, selang:TDateTime;
    jam, menit, detik, mdetik:word;
    V, Sg, SL:CxArr1;
    Cap:dArr1;
    TypBus:iArr1;
    Z, Tp, Alir, Arus:CxArr2;
    Lc, Tr:dArr2;
begin
  Nbus:=high(gBus)+1;
  DecodeCommDataToLFData(gBus, Nbus, Nsal, V, Sg, SL, Cap, TypBus, gBranch,
  Z, Tp, Lc, Tr);
  gZrec:=CopyMatrix(Z);
  if Nbus=23 then
  begin
    Z[3,10]:=Cmplx(0,0);
    Z[8,12]:=Cmplx(0,0);
    Z[14,18]:=Cmplx(0,0);
    Z[4,15]:=Cmplx(0,0);
  end
  else if Nbus=14 then
  begin
    Z[2,8]:=Cmplx(0,0);
    Z[7,11]:=Cmplx(0,0);
    Z[4,13]:=Cmplx(0,0);
  end;
  btnUseDefault.Enabled:=true;
  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 Nbus-1 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]);
    Series1.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;

```

```

fgBranch.Cells[0,i+1]:=IntToStr(i+1);
fgBranch.Cells[1,i+1]:=IntToStr(ia+1);
fgBranch.Cells[2,i+1]:=IntToStr(ja+1);
fgBranch.Cells[3,i+1]:=RealToStr(Alir[ia,ja].real,3);
fgBranch.Cells[4,i+1]:=RealToStr(Alir[ia,ja].imag,3);
fgBranch.Cells[5,i+1]:=RealToStr(Arus[ia,ja].real,3);
fgBranch.Cells[6,i+1]:=RealToStr(Arus[ia,ja].imag,3);
fgBranch.Cells[7,i+1]:=IntToStr(ja+1);
fgBranch.Cells[8,i+1]:=IntToStr(ia+1);
fgBranch.Cells[9,i+1]:=RealToStr(Alir[ja,ia].real,3);
fgBranch.Cells[10,i+1]:=RealToStr(Alir[ja,ia].imag,3);
fgBranch.Cells[11,i+1]:=RealToStr(Arus[ja,ia].real,3);
fgBranch.Cells[12,i+1]:=RealToStr(Arus[ja,ia].imag,3);
end;
edtSumGen.Text:=toStringJ(gParamLF.SumGen,3);
edtSumLoad.Text:=toStringJ(gParamLF.SumLoad,3);
edtSumLoss.Text:=toStringJ(gParamLF.SumLoss,3);
edtIterasi.Text:=IntToStr(gParamLF.Iterasi);
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime.Text:=IntToStr(jam)+':'+'+IntToStr(menit)+':'+'+
IntToStr(detik)+':'+'+IntToStr(mdetik);
CostSebelum:=-gParamLF.SumLoss.real;
end;

procedure TForm1.btnHitungClick(Sender: TObject);
var i,ia,ja,Nbus,Nsal,NLoop,MaxGen,PopSize,len:integer;
pcross,pmutasi,ka,fitness,PlossMin,V1,V2,V3,V4,I1,I2,w1,w2,w3:double;
jam,menit,detik,mdetik:word;
tesRadial:boolean;
mulai,selesai,selang:TDateTime;
Cap:dArr1;
Lc,Tr:dArr2;
BestIndi:TIndiBin1;
TypBus,DataHasil:iArr1;
V,Sg,SL:CxArr1;
Zb,Tp,Alir,Arus:CxArr2;
sw:TSwitchArr1;
gats:TGenRecon;
begin
//new code place here
mulai:=time;
NLoop:=StrToInt(edtNLoop.Text);
PlossMin:=StrToFloat(edtPlossMin.Text);
V1:=StrToFloat(edtV1.Text);
V2:=StrToFloat(edtV2.Text);
V3:=StrToFloat(edtV3.Text);
V4:=StrToFloat(edtV4.Text);
I1:=StrToFloat(edtI1.Text);
I2:=StrToFloat(edtI2.Text);
w1:=StrToFloat(edtw1.Text);
w2:=StrToFloat(edtw2.Text);
w3:=StrToFloat(edtw3.Text);
gFit:=TFitness.Create(NLoop,PlossMin,V1,V2,V3,V4,I1,I2,w1,w2,w3);
MaxGen:=StrToInt(edtMaxGen.Text);
pEGen.Max:=MaxGen;
PopSize:=StrToInt(edtPopSize.Text);
PCross:=StrToFloat(edtPCross.Text);
PMutasi:=StrToFloat(edtPMutasi.Text);
Ka:=StrToFloat(edtKa.Text);
len:=high(gBranch)+1;
gats:=TGenRecon.Create(MaxGen,PopSize,len,NLoop,
pcross,PMutasi,Ka);
BestIndi:=gats.getBestIndi;
gats.DecodeIndiToData(BestIndi,DataHasil,fitness);

```

```

sw:=gFit.GetSwitth(DataHasil);
DecodeCommDataTol.FData(gBus,Nbus,Nsal,V,Sg,SL,Cap,TypBus,gBranch,
Zb,Tp,Lc,Tr);
Zb:=BuatZBaru(gZrec,sw);
tesRadial:=TesJaringan(Zb);
if tesRadial=true then
begin
NewtonRaphson(gParamLF,V,Sg,SL,Cap,TypBus,Zb,Tp,Alir,Arus,Lc,Tr);
selesai:=time;
selang:=selesai-mulai;
Series2.Clear;
for i:=0 to Nbus-1 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;
Nsal:=high(gBranch)+1;
for i:=0 to Nsal-1 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(ia+1);
fgBranch2.Cells[2,i+1]:=IntToStr(ja+1);
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(ja+1);
fgBranch2.Cells[8,i+1]:=IntToStr(ia+1);
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:='3';
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime2.Text:=IntToStr(jam)+' ':'+IntToStr(menit)+' ':'+
IntToStr(detik)+' ':'+IntToStr(mdetik);
CostSesudah:=gParamLF.SumLoss.real;
edtSebelum.Text:=FormatFloat('#,##0.000',CostSebelum);
edtSesudah.Text:=FormatFloat('#,##0.000',CostSesudah);
for l:=0 to 3 do
begin
fgOS.Cells[0,l+1]:=IntToStr(i+1);
fgOS.Cells[1,l+1]:=IntToStr(sw[i].dari+1);
fgOS.Cells[2,l+1]:=IntToStr(sw[i].ke+1);
end;
end
else
begin
MessageDlg('Perhitungan gagal!',mtWarning,[mbOK],0);

```

```

end;
gFlt.Free;
gats.Free;
end;

procedure TFormHasil.btnUseDefaultClick(Sender: TObject);
var Nbus:integer;
begin
  Nbus:=high(gBus)+1;
  if Nbus=23 then
  begin
    edtNLoop.Text:='4';
  end
  else if Nbus=14 then
  begin
    edtNLoop.Text:='3';
  end
  else if Nbus=43 then
  begin
    edtNLoop.Text:='11';
  end;
  fgOS.RowCount:=StrToInt(edtNLoop.Text)+1;
  edtMaxGen.Text:='50';
  edtPopSize.Text:='10';
  edtPCross.Text:='0.75';
  edtPMutasi.Text:='0.05';
  edtKa.Text:='1000';
  edtPlossMin.Text:='0.317';
  edtV1.Text:='0.90';
  edtV2.Text:='0.95';
  edtV3.Text:='1.05';
  edtV4.Text:='1.10';
  edtI1.Text:='1.0';
  edtI2.Text:='1.2';
  edtw1.Text:='1.0';
  edtw2.Text:='1.0';
  edtw3.Text:='1.0';
  btnHitung.Enabled:=true;
end;

procedure TFormHasil.FormCreate(Sender: TObject);
begin
  fgOS.Cells[0,0]:='No';
  fgOS.Cells[1,0]:='Asal';
  fgOS.Cells[2,0]:='Tujuan';
end;

procedure TFormHasil.edtNLoopChange(Sender: TObject);
begin
  if edtNLoop.Text='' then
  begin
    fgOS.RowCount:=2;
  end
  else
  begin
    fgOS.RowCount:=StrToInt(edtNLoop.Text)+1;
  end;
end;

end.

```

```

unit uInputLFChild;

interface

uses uInputLF, SysUtils, uHasil, Forms;

type
  TfrmInputLFChild=class(TfrmInputLF)
  protected
    procedure ShowHasil;override;
  end;

var frmInput:TfrmInputLFChild;

implementation

uses uLoadflow;

procedure TfrmInputLFChild.ShowHasil;
begin
  try
    if frmHasil=nil then
      begin
        frmHasil:=TfrmHasil.Create(Application);
      end;
    frmHasil.fgBus.Cells[0,0]:='Bus';
    frmHasil.fgBus.Cells[1,0]:='absV (pu)';
    frmHasil.fgBus.Cells[2,0]:='sudV (deg)';
    frmHasil.fgBus.Cells[7,0]:='Supr (pu)';
    frmHasil.fgBus.Cells[8,0]:='Type Bus';
    frmHasil.fgBranch.Cells[0,0]:='No';
    frmHasil.fgBranch.Cells[1,0]:='Dari';
    frmHasil.fgBranch.Cells[2,0]:='Ke';
    frmHasil.fgBranch.Cells[5,0]:='Arus re (A)';
    frmHasil.fgBranch.Cells[6,0]:='Arus im (A)';
    frmHasil.fgBranch.Cells[7,0]:='Dari';
    frmHasil.fgBranch.Cells[8,0]:='Ke';
    frmHasil.fgBranch.Cells[11,0]:='Arus re (A)';
    frmHasil.fgBranch.Cells[12,0]:='Arus im (A)';
    frmHasil.fgBus2.Cells[0,0]:='Bus';
    frmHasil.fgBus2.Cells[1,0]:='absV (pu)';
    frmHasil.fgBus2.Cells[2,0]:='sudV (deg)';
    frmHasil.fgBus2.Cells[7,0]:='Supr (pu)';
    frmHasil.fgBus2.Cells[8,0]:='Type Bus';
    frmHasil.fgBranch2.Cells[0,0]:='No';
    frmHasil.fgBranch2.Cells[1,0]:='Dari';
    frmHasil.fgBranch2.Cells[2,0]:='Ke';
    frmHasil.fgBranch2.Cells[5,0]:='Arus re (A)';
    frmHasil.fgBranch2.Cells[6,0]:='Arus im (A)';
    frmHasil.fgBranch2.Cells[7,0]:='Dari';
    frmHasil.fgBranch2.Cells[8,0]:='Ke';
    frmHasil.fgBranch2.Cells[11,0]:='Arus re (A)';
    frmHasil.fgBranch2.Cells[12,0]:='Arus im (A)';
    if gParamLF.PKonst=1 then
      begin
        frmHasil.fgBus.Cells[3,0]:='Pg (W)';
        frmHasil.fgBus.Cells[4,0]:='Qg (VAR)';
        frmHasil.fgBus.Cells[5,0]:='PL (W)';
        frmHasil.fgBus.Cells[6,0]:='QL (VAR)';
        frmHasil.fgBranch.Cells[3,0]:='P (W)';
        frmHasil.fgBranch.Cells[4,0]:='Q (VAR)';
        frmHasil.fgBranch.Cells[9,0]:='P (W)';
        frmHasil.fgBranch.Cells[10,0]:='Q (VAR)';
        frmHasil.lblGen.Caption:='VA';
      end;
  end;
end;

```

```

frmHasil.lblLoad.Caption:='VA';
frmHasil.lblLoss.Caption:='VA';
frmHasil.fgBus2.Cells[3,0]:='Pg (W)';
frmHasil.fgBus2.Cells[4,0]:='Qg (VAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (W)';
frmHasil.fgBus2.Cells[6,0]:='QL (VAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (W)';
frmHasil.fgBranch2.Cells[4,0]:='Q (VAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (W)';
frmHasil.fgBranch2.Cells[10,0]:='Q (VAR)';
frmHasil.lblGen2.Caption:='VA';
frmHasil.lblLoad2.Caption:='VA';
frmHasil.lblLoss2.Caption:='VA';
frmHasil.lblLossP.Caption:='VA';
frmHasil.lblLossN.Caption:='VA';
end
else if gParamLF.PKonst=1000 then
begin
frmHasil.fgBus.Cells[3,0]:='Pg (kW)';
frmHasil.fgBus.Cells[4,0]:='Qg (kVAR)';
frmHasil.fgBus.Cells[5,0]:='PL (kW)';
frmHasil.fgBus.Cells[6,0]:='QL (kVAR)';
frmHasil.fgBranch.Cells[3,0]:='P (kW)';
frmHasil.fgBranch.Cells[4,0]:='Q (kVAR)';
frmHasil.fgBranch.Cells[9,0]:='P (kW)';
frmHasil.fgBranch.Cells[10,0]:='Q (kVAR)';
frmHasil.lblGen.Caption:='kVA';
frmHasil.lblLoad.Caption:='kVA';
frmHasil.lblLoss.Caption:='kVA';
frmHasil.fgBus2.Cells[3,0]:='Pg (kW)';
frmHasil.fgBus2.Cells[4,0]:='Qg (kVAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (kW)';
frmHasil.fgBus2.Cells[6,0]:='QL (kVAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (kW)';
frmHasil.fgBranch2.Cells[4,0]:='Q (kVAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (kW)';
frmHasil.fgBranch2.Cells[10,0]:='Q (kVAR)';
frmHasil.lblGen2.Caption:='kVA';
frmHasil.lblLoad2.Caption:='kVA';
frmHasil.lblLoss2.Caption:='kVA';
frmHasil.lblLossP.Caption:='kVA';
frmHasil.lblLossN.Caption:='kVA';
end
else if gParamLF.PKonst=1000000 then
begin
frmHasil.fgBus.Cells[3,0]:='Pg (MW)';
frmHasil.fgBus.Cells[4,0]:='Qg (MVAR)';
frmHasil.fgBus.Cells[5,0]:='PL (MW)';
frmHasil.fgBus.Cells[6,0]:='QL (MVAR)';
frmHasil.fgBranch.Cells[3,0]:='P (MW)';
frmHasil.fgBranch.Cells[4,0]:='Q (MVAR)';
frmHasil.fgBranch.Cells[9,0]:='P (MW)';
frmHasil.fgBranch.Cells[10,0]:='Q (MVAR)';
frmHasil.lblGen.Caption:='MVA';
frmHasil.lblLoad.Caption:='MVA';
frmHasil.lblLoss.Caption:='MVA';
frmHasil.fgBus2.Cells[3,0]:='Pg (MW)';
frmHasil.fgBus2.Cells[4,0]:='Qg (MVAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (MW)';
frmHasil.fgBus2.Cells[6,0]:='QL (MVAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (MW)';
frmHasil.fgBranch2.Cells[4,0]:='Q (MVAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (MW)';
frmHasil.fgBranch2.Cells[10,0]:='Q (MVAR)';

```

```
frmHasil.lblGen2.Caption:='MVA';
frmHasil.lblLoad2.Caption:='MVA';
frmHasil.lblLoss2.Caption:='MVA';
frmHasil.lblLossP.Caption:='-MVA';
frmHasil.lblLossN.Caption:='MVA';
end;
frmHasil.fgBus.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasil.fgBranch.RowCount:=StrToInt(edtNsal.Text)+1;
frmHasil.fgBus2.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasil.fgBranch2.RowCount:=StrToInt(edtNsal.Text)+1;
frmHasil.ShowModal;
finally
  frmHasil.Free;
end;
end;
end.
```

```

unit uMenu;

interface

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

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

var
  frmMenu: TfrmMenu;

implementation

uses uInputLEChild, uComplex, uUtils, uLoadflow;

{$R *.dfm}

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

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

```

```

Readln(output, Nbus);
Readln(output, Nsal);
Readln(output, Vbase);
Readln(output, VKonst);
Readln(output, Pbase);
Readln(output, PKonst);
Readln(output, param);
gParamLF.Vbase:=Vbase;
gParamLF.VKonst:=VKonst;
gParamLF.Pbase:=Pbase;
gParamLF.PKonst:=PKonst;
if Param=1 then
begin
  gParamLF.ParamBranch:=pbPu;
end
else if Param=2 then
begin
  gParamLF.ParamBranch:=pbOhm;
end;
gParamLF.MaxIterasi:=15;
gParamLF.Toleransi:=0.0001;
try
  frmInput:=TfrmInputLFChild.Create(Application);
  frmInput.edtNbus.Text:=IntToStr(Nbus);
  frmInput.edtNsal.Text:=IntToStr(Nsal);
  frmInput.edtVbase.Text:=FloatToStr(Vbase);
  if VKonst=1 then
  begin
    frmInput.cmbVKonst.Text:='V';
  end
  else if VKonst=1000 then
  begin
    frmInput.cmbVKonst.Text:='kV';
  end
  else if VKonst=1000000 then
  begin
    frmInput.cmbVKonst.Text:='MV';
  end;
  frmInput.edtPbase.Text:=FloatToStr(Pbase);
  if PKonst=1 then
  begin
    frmInput.cmbPKonst.Text:='VA';
    frmInput.fgBus.Cells[3,0]:='Pg (W)';
    frmInput.fgBus.Cells[4,0]:='Qg (VAR)';
    frmInput.fgBus.Cells[5,0]:='PL (W)';
    frmInput.fgBus.Cells[6,0]:='QL (VAR)';
    frmInput.fgBranch.Cells[9,0]:='Kap (VA)';
    frmInput.fgBranch.Cells[10,0]:='P (W)';
    frmInput.fgBranch.Cells[11,0]:='Q (VAR)';
    frmInput.fgBranch.Cells[16,0]:='P (W)';
    frmInput.fgBranch.Cells[17,0]:='Q (VAR)';
  end
  else if PKonst=1000 then
  begin
    frmInput.cmbPKonst.Text:='kVA';
    frmInput.fgBus.Cells[3,0]:='Pg (kW)';
    frmInput.fgBus.Cells[4,0]:='Qg (kVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (kW)';
    frmInput.fgBus.Cells[6,0]:='QL (kVAR)';
    frmInput.fgBranch.Cells[9,0]:='Kap (kVA)';
    frmInput.fgBranch.Cells[10,0]:='P (kW)';
    frmInput.fgBranch.Cells[11,0]:='Q (kVAR)';
    frmInput.fgBranch.Cells[16,0]:='P (kW)';
    frmInput.fgBranch.Cells[17,0]:='Q (kVAR)';
  end

```

```

end
else if PKonst=1000000 then
begin
  frmInput.cmbPKonst.Text:='MVA';
  frmInput.fgBus.Cells[3,0]:='Pg (MW)';
  frmInput.fgBus.Cells[4,0]:='Qg (MVAR)';
  frmInput.fgBus.Cells[5,0]:='PL (MW)';
  frmInput.fgBus.Cells[6,0]:='QL (MVAR)';
  frmInput.fgBranch.Cells[9,0]:='Kap (MVA)';
  frmInput.fgBranch.Cells[10,0]:='P (MW)';
  frmInput.fgBranch.Cells[11,0]:='Q (MVAR)';
  frmInput.fgBranch.Cells[16,0]:='F (MW)';
  frmInput.fgBranch.Cells[17,0]:='-Q (MVAR)';
end;
if param=1 then
begin
  frmInput.cmbParam.Text:='-pu';
  frmInput.fgBranch.Cells[3,0]:='R (pu)';
  frmInput.fgBranch.Cells[4,0]:='X (pu)';
  frmInput.fgBranch.Cells[5,0]:='Lc (pu)';
  frmInput.fgBus.Cells[7,0]:='Cap (pu)';
end
else if param=2 then
begin
  frmInput.cmbParam.Text:='ohm';
  frmInput.fgBranch.Cells[3,0]:='R (ohm)';
  frmInput.fgBranch.Cells[4,0]:='X (ohm)';
  frmInput.fgBranch.Cells[5,0]:='Lc (ohm)';
  frmInput.fgBus.Cells[7,0]:='Cap (ohm)';
end;
SetLength(gBus,Nbus);
for i:=0 to Nbus-1 do
begin
  Readln(output,absV,sudV,Pg,Qg,PL,QL,Cap,Typ);
  gBus[i].absV:=absV;
  gBus[i].sudV:=sudV;
  gBus[i].Pgen:=Pg;
  gBus[i].Qgen:=Qg;
  gBus[i].PL:=PL;
  gBus[i].QL:=-QL;
  gBus[i].Cap:=Cap;
  gBus[i].typeBus:=Typ;
  frmInput.fgBus.Cells[0,i+1]:=IntToStr(i+1);
  frmInput.fgBus.Cells[1,i+1]:=-FloatToStr(absV);
  frmInput.fgBus.Cells[2,i+1]:=-FloatToStr(sudV);
  frmInput.fgBus.Cells[3,i+1]:=FloatToStr(Pg);
  frmInput.fgBus.Cells[4,i+1]:=FloatToStr(Qg);
  frmInput.fgBus.Cells[5,i+1]:=FloatToStr(PL);
  frmInput.fgBus.Cells[6,i+1]:=FloatToStr(QL);
  frmInput.fgBus.Cells[7,i+1]:=FloatToStr(Cap);
  frmInput.fgBus.Cells[8,i+1]:=-IntToStr(typ);
end;
SetLength(gBranch,Nsal);
for i:=0 to Nsal-1 do
begin
  Readln(output,dari,ke,R,X,Lc,Tr,Tu,Su,CapSal);
  gBranch[i].dari:=dari;
  gBranch[i].ke:=ke;
  gBranch[i].R:=R;
  gBranch[i].X:=X;
  gBranch[i].Lc:=Lc;
  gBranch[i].Tr:=Tr;
  gBranch[i].Tu:=Tu;
  gBranch[i].Su:=Su;

```

```

gBranch[i].KapSal:=CapSal;
frmInput.fgBranch.Cells[0,i+1]:=IntToStr(i+1);
frmInput.fgBranch.Cells[1,i+1]:=IntToStr(dari);
frmInput.fgBranch.Cells[2,i+1]:=IntToStr(ka);
frmInput.fgBranch.Cells[3,i+1]:=FloatToStr(R);
frmInput.fgBranch.Cells[4,i+1]:=FloatToStr(X);
frmInput.fgBranch.Cells[5,i+1]:=FloatToStr(Lc);
frmInput.fgBranch.Cells[6,i+1]:=FloatToStr(Tr);
frmInput.fgBranch.Cells[7,i+1]:=FloatToStr(Tu);
frmInput.fgBranch.Cells[8,i+1]:=FloatToStr(Su);
frmInput.fgBranch.Cells[9,i+1]:=FloatToStr(CapSal);
end;
Readln(output,Ngen);
if Ngen<>0 then
begin
frmInput.fgGen.RowCount:=Ngen+1;
SetLength(gGenLF,Ngen);
for i:=0 to Ngen-1 do
begin
Readln(output,dari,R,X,Lc,Tr,Tu,Su,CapSal,Pmin,Pmax);
gGenLF[i].bus:=dari;
gGenLF[i].Qmin:=R;
gGenLF[i].Qmax:=X;
gGenLF[i].a2:=Lc;
gGenLF[i].a1:=Tr;
gGenLF[i].a0:=Tu;
gGenLF[i].FixCost:=-Su;
gGenLF[i].VarCost:=-CapSal;
gGenLF[i].Pmin:=Pmin;
gGenLF[i].Pmax:=Pmax;
frmInput.fgGen.Cells[0,i+1]:=IntToStr(i+1);
frmInput.fgGen.Cells[1,i+1]:=IntToStr(gGenLF[i].bus);
frmInput.fgGen.Cells[2,i+1]:=RealToStr(gGenLF[i].Qmin,2);
frmInput.fgGen.Cells[3,i+1]:=RealToStr(gGenLF[i].Qmax,2);
frmInput.fgGen.Cells[4,i+1]:=RealToStr(gGenLF[i].a2,5);
frmInput.fgGen.Cells[5,i+1]:=RealToStr(gGenLF[i].a1,5);
frmInput.fgGen.Cells[6,i+1]:=RealToStr(gGenLF[i].a0,5);
frmInput.fgGen.Cells[7,i+1]:=RealToStr(gGenLF[i].FixCost,2);
frmInput.fgGen.Cells[8,i+1]:=RealToStr(gGenLF[i].VarCost,2);
frmInput.fgGen.Cells[9,i+1]:=RealToStr(gGenLF[i].Pmin,2);
frmInput.fgGen.Cells[10,i+1]:=RealToStr(gGenLF[i].Pmax,2);
end;
end
else
begin
frmInput.fgGen.RowCount:=2;
end;
CloseFile(output);
frmInput.Caption:='Tampilan Data';
frmInput.btnNext.Caption:='&Next';
frmInput.ShowModal;
finally
frmInput.Free;
end;
end;
except
MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);
end;
end;

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

```

```
procedure TFormMenu.btnSettingClick(Sender: TObject);  
begin  
    Application.Terminate;  
end;  
  
end.
```

```

unit uRecursive;

interface

uses uUtils, uComplex, uLoadflow, SysUtils;

function isNumeric(const rStr:Char):boolean;
function DecodeStrToData(const rStr:string):iArr1;
function DecodeDataToStr(const rData:iArr1):string;
function InitSwit(const rZb:CxArr2):TSwitArr1;
function BuatZBaru(const rZ:CxArr2;
    const rSwit:TSwitArr1):CxArr2; overload;
function BuatZBaru(const rZ:CxArr2;
    const rChrom:bArr1):CxArr2; overload;
function TesJaringan(const rZ:CxArr2):boolean;
procedure GenCombinasi(const rNData:integer;
    var rData:iArr1;
    var rCek:boolean);

var gZrec:CxArr2;
    gSw:TSwitArr1;

implementation

var bus:bArr1;
    Njar:integer;
    findLoop:boolean;

function isNumeric(const rStr:Char):boolean;
begin
    result:=false;
    if rStr in ['1','2','3','4','5','6','7','8','9','0'] then
    begin
        result:=true;
    end;
end;

function DecodeStrToData(const rStr:string):iArr1;
var i,sa:integer;
    st,str:string;
begin
    str:=trim(rStr);
    SetLength(result,4);
    sa:=0;
    for i:=1 to length(str)+1 do
    begin
        if isNumeric(str[i])=true then
        begin
            st:=st+str[i];
        end
        else
        begin
            result[sa]:=StrToInt(st);
            inc(sa);
            st:='';
        end;
    end;
end;

function DecodeDataToStr(const rData:iArr1):string;
var i,NData:integer;
begin
    NData:=high(rData)+1;
    result:='';

```

```

    for i:=0 to NData-1 do
    begin
        result:=result+IntToStr(rData[i])+' ';
    end;
end;

function InitSwitH(const rZb:CxArr2):TSwitHArr1;
var i,j,sa,Nbus:integer;
begin
    Nbus:=high(rZb)+1;
    sa:=0;
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            if rZb[i,j].imag<>0 then
            begin
                inc(sa);
            end;
        end;
    end;
    SetLength(result,sa);
    sa:=0;
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            if rZb[i,j].imag<>0 then
            begin
                result[sa].dari:=i;
                result[sa].ke:=j;
                inc(sa);
            end;
        end;
    end;
end;

function BuatZBaru(const rZ:CxArr2;
    const rSwitH:TSwitHArr1):CxArr2;
var i,j,NSwitH,Nbus:integer;
begin
    Nbus:=high(rZ)+1;
    SetLength(result,Nbus,Nbus);
    for i:=0 to Nbus-1 do
    begin
        for j:=0 to Nbus-1 do
        begin
            result[i,j]:=Cmplx(rZ[i,j]);
        end;
    end;
    NSwitH:=high(rSwitH)+1;
    for i:=0 to NSwitH-1 do
    begin
        result[rSwitH[i].dari,rSwitH[i].ke].real:=0;
        result[rSwitH[i].dari,rSwitH[i].ke].imag:=0;
    end;
end;

function BuatZBaru(const rZ:CxArr2;
    const rChrom:bArr1):CxArr2;
var i,j,Nbus,sa:integer;
begin
    Nbus:=high(rZ)+1;
    SetLength(result,Nbus,Nbus);

```

```

sa:=0;
for i:=0 to Nbus-1 do
begin
  for j:=0 to Nbus-1 do
  begin
    if rZ[i,j].imag<>0 then
    begin
      if rChrom[sa]=true then
      begin
        result[i,j]:=Cmplx(0,0);
      end
      else
      begin
        result[i,j]:=Cmplx(rZ[i,j]);
      end;
      inc(sa);
    end;
  end;
end;
end;

function BuatMatrikJalur(const rZ:CxArr2):iArr2;
var i,j,Nbus:integer;
begin
  Nbus:=high(rZ)+1;
  SetLength(result,Nbus,Nbus);
  for i:=0 to Nbus-1 do
  begin
    for j:=0 to Nbus-1 do
    begin
      result[i,j]:=0;
    end;
  end;
  for i:=0 to Nbus-1 do
  begin
    for j:=0 to Nbus-1 do
    begin
      if rZ[i,j].imag<>0 then
      begin
        result[i,j]:=1;
        result[j,i]:=1;
      end;
    end;
  end;
end;

function TesJaringan(const rZ:CxArr2):boolean;
var i:integer;
    Jalur:iArr2;

procedure doRecursive(dari,ke:integer);
var i:integer;
begin
  if findLoop=true then
  begin
    exit;
  end;
  if bus[ke]=false then
  begin
    bus[ke]:=true;
  end
  else
  begin
    findLoop:=true;
  end;
end;

```

```

        exit;
    end;
    for i:=0 to Njar-1 do
    begin
        if Jalur[ke,i]=1 then
        begin
            if i<>dari then
            begin
                doRecursive(ke,i);
                if findLoop=true then
                begin
                    exit;
                end;
            end;
        end;
    end;
end;

begin
    Jalur:=BuatMatrikJalur(rZ);
    Njar:=high(Jalur)+1;
    SetLength(bus,Njar);
    for i:=0 to Njar-1 do
    begin
        bus[i]:=false;
    end;
    findLoop:=false;
    result:=true;
    doRecursive(0,0);
    if findLoop=true then
    begin
        result:=false;
        exit;
    end;
    for i:=0 to Njar-1 do
    begin
        if bus[i]=false then
        begin
            result:=false;
            exit;
        end;
    end;
end;

function CekCombinasi(const rData:iArr1):boolean;
var i,j:integer;
begin
    result:=true;
    for i:=0 to high(rData)-1 do
    begin
        for j:=i+1 to high(rData) do
        begin
            if rData[i]=rData[j] then
            begin
                result:=false;
                break;
            end;
        end;
    end;
end;

procedure GenCombinasi(const rNData:integer;
    var rData:iArr1;
    var rCek:boolean);

```

```
var l,NData,Ncom:integer;
begin
  NData:=rNData-1;
  Ncom:=high(rData)+1;
  for i:=Ncom-1 downto 0 do
  begin
    inc(rData[i]);
    if rData[i]<=NData then
    begin
      break;
    end
    else
    begin
      rData[i]:=0;
    end;
  end;
  rCek:=CekCombinasi(rData);
end;

end.
```

```
unit uVarGlobal;  
  
interface  
  
var statPC1,statPC2,statPC3,statPC4:boolean;  
    strPC1,strPC2,strPC3,strPC4:string;  
  
implementation  
  
end.
```



PT. PLH (PERSERO)
DISTRIBUSI JAWA TIMUR
AREA PELAYANAN & JARINGAN MALANG

DATA BEBAN GARDU INDUK BLIMBING

BULAN : FEBRUARI 2007

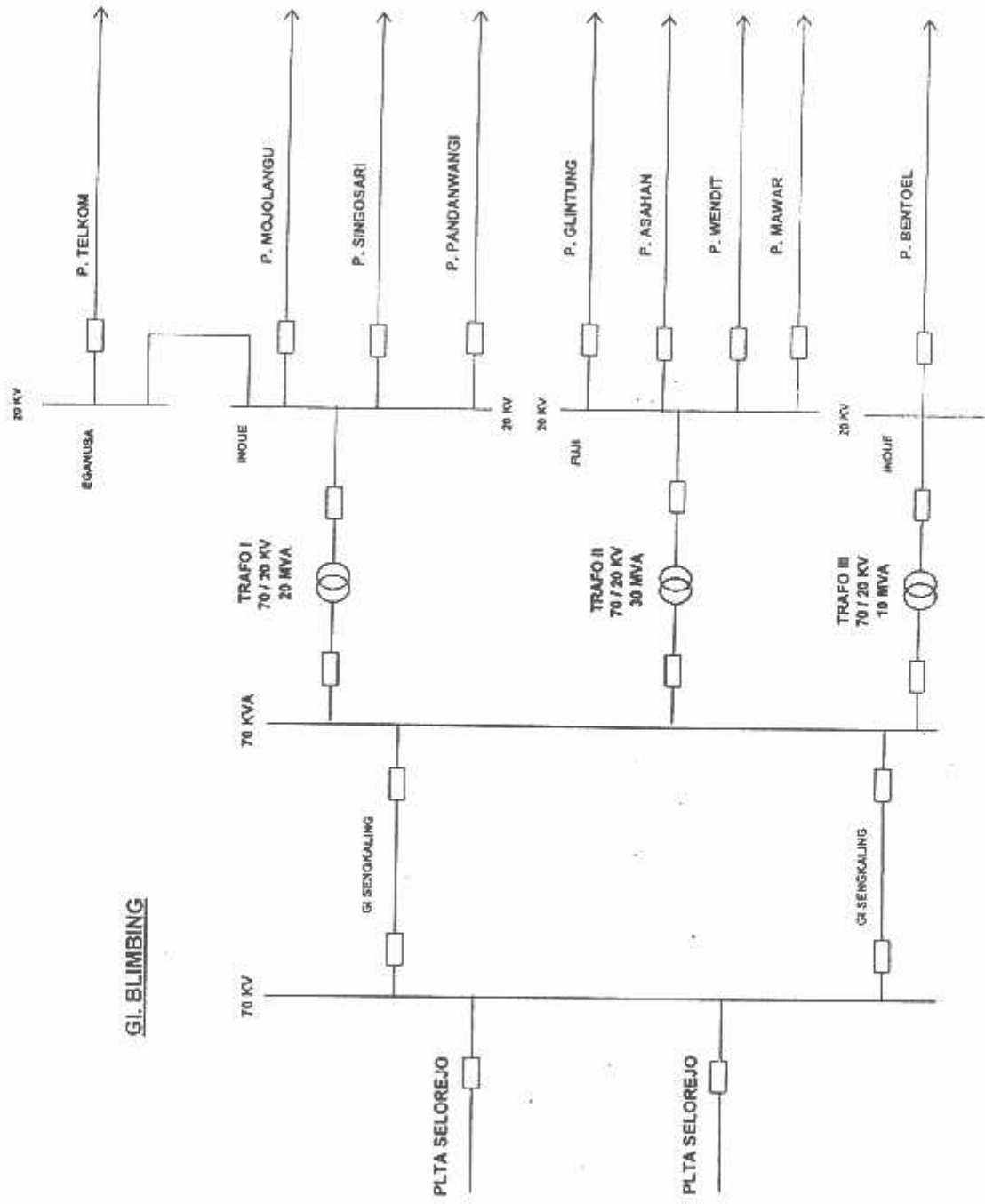
NO	GARDU INDUK MERK TRAFO PENYULANG	TEG	PANJANG PENYULANG		PANJANG JTM (kms)	KABEL INCOMING TRF & OUTGOING	I Nom (Amp)	I Set (A.np)	BEBAN		PROSENTASE INOM		KETERANGAN
			SKTM	SUTM					SIANG	MALAM	SIANG	MALAM	
1	G. BLIMBING	70/20 KV											
	Trafo I/20 MVA UNINDO						577	650	311	473	53,9	82,0	
1	P. MOJOLANGU	20	168	22.407	22.575	C V T	304	300	136	200	44,7	65,8	
2	P. SINGOSARI	20	912	78.960	79.872	C V T	304	300	154	245	50,7	80,6	
3	P. PANDANWANGI	20	251	11.304	11.565	C V T	304	300	20	21	6,6	6,9	
4	P. TELKOM	20	410	260	670	X L P E	400	320	4	5	1,0	1,3	
	Trafo II/30 MVA PASTI	70/20 KV					866	1.000	520	638	60,0	73,7	
5	P. MAWAR	20	838	36.870	37.708	C V T	304	300	186	260	61,2	85,5	
6	P. ASAHAN	20	963	26.819	29.772	C V T	304	300	170	163	55,9	53,6	
7	P. WENDIT	20	472	16.473	18.945	C V T	304	300	50	80	16,4	26,3	
8	P. GLUNTUNG	20	338	25.735	26.073	C V T	304	300	116	146	38,2	48,0	
	Trafo III/10 MVA TAKAOKA	70/20 KV					289	320	170	104	58,8	36,0	
9	P. BENTOEL	20	241	7.366	7.637	C V T	400	300	170	140	42,5	35,0	
	JUMLAH		4.583	230.224	234.807								

Malang, 23 Maret 2007

Mengenalai
Spv. Operasi Distribusi

RAMDANI AGUSTIYANSAH

GI. BLIMBING



- : TRANSFORMATOR
- : PEMUTUS
- : PERTULANG



PT. PLN (PERSERO)
DISTRIBUSI JATIM
APAJI MALANG

SINGLE LINE DIAGRAM
GARDU INDUK BLIMBING

NO. GAMBAR	DIREVISI	DISERTUJUKAN
00001	01	
NO. SURTAH	NO. SURTAH	NO. SURTAH



DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA PASAHAN"

TH. 2006 SEMESTER II

NO GRD	ALAMAT	DAYA (KVA)	TBL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			TEG PADA TRAFO					BEBAN (KVA)	PERSEN BEBAN	TEG UJUNG SUTR (VOLT)		STEMPEN TEGANGAN UJUNG SUTR									
					R	S	T	N	R	S	T	N	R	S	T	R	S	T	R	S	T			R	S		T	R	S	T	R	S	T		
1	JL DIPONEGORO	200	05-09-06	0.69	77	89	75	31											245	210	220	82	218	218	378	378	378	147.15	73.58	210	212	211	AIBAC1		
2	JL JAKSA AGUNG SUPRAPTO	200	07-09-06	0.72	61	87	55	33											176	258	220	88	233	233	404	404	404	152.38	76.19	217	216	214	C4DLA7		
3	JL MAHAKAM	100	09-09-06	0.62											99	103	66	71	226	226	226	391	391	391	391	391	391	60.57	60.57	224	224	224	D3A7		
4	JL TENAGA (PAB. RIK. PENAJARAS)	250	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00						
5	JL CILIWUNG	200	05-09-06	0.66					108	88	114	18	71	66	92	91	223	223	223	386	386	386	386	386	386	386	386	386	170.15	85.07	219	218	217	B3CE6D3	
6	JL KARYA TMR (KOYA NANAS)	200	07-09-06	0.82	55	55	59	5											98	107	105	11	226	226	226	391	391	391	70.08	35.03	224	223	223	A4B7	
7	JL KARYA TMR (PT. BENTOEL)	150	06-09-06	0.66	12	20	28	5											44	55	71	16	231	231	231	400	400	400	39.27	26.18	230	225	224	C4B5	
8	JL TENAGA	100	09-09-06	0.72											36	42	38	18										27.61	27.61	237	236	237	B4A1		
9	JL CILIWUNG II	250	07-09-06	0.65					151	205	218	51	72	49	20	39	14	14	6	237	268	252	96	238	238	238	412	412	180.17	72.07	236	236	236	B1A6B3	
10	JL CITARUM	200	05-09-06	0.67	61	82	66	15											108	127	138	46	225	225	225	390	390	390	97.43	48.71	220	218	219	C5B4	
11	JL CIJUNG	160	07-09-06	0.66	45	29	49	41											64	64	69	73	233	233	233	404	404	404	45.90	28.69	230	230	230	C3B6C1	
12	JL J A SUPRAPTO	75													0	0	0	0										0.00	0.00						
13	JL HASANMUJID	100	07-09-06	0.64					85	88	148	22							11	15	9	51	233	233	233	404	404	404	82.95	82.95	216	214	215	D1A3B5C2	
14	JL TENAGA UTARA	100	08-09-06	0.62					55	97	77	12							11	27	8	11	224	224	224	388	388	388	61.60	61.60	220	213	220	87A5	
15	JL J A SUPRAPTO (TIARA)	200	08-09-06	0.6	79	91	131	22											141	152	182	40	229	229	229	397	397	397	108.78	54.39	221	220	221	C1D1C6	
16	JL LETJEND. S. PARMAN (GRENDEL)	25	KHUSUS		0	0	0	0							0	0	0	0										0.00	0.00						
17	JL KARYA TIMUR	200	05-09-06	0.93	117	159	89	12											117	155	89	12	223	223	223	386	386	386	80.50	40.25	220	220	220	A7B3	
18	JL CILIWUNG	100	09-09-06	0.66	*				101	83	124	51																							
19	JL DR. WAHIDIN	150	08-09-06	0.57	151	183	180	22											13	16	42	35													
20	JL KARYA TIMUR	160	06-09-06	0.66											7	15	14	51											6.42	5.27	230	231	230	C5B2A8	
21	JL DIPONEGORO	150	09-09-06	0.83	92	87	63	11											92	87	63	11	235	235	235	407	407	407	56.87	37.91	227	227	227	A5DLA2	
22	JL CILIWUNG III	250	05-09-06	0.65	151	121	147	24											272	208	202	75	229	229	229	397	397	397	156.18	62.47	220	223	221	A4C3	
23	DS PATUK SALAM	100	09-09-06	0.68					79	47	47	31							49	75	47	42	235	235	235	407	407	407	80.84	80.84	231	232	230	C3B1C1	
24	JL MAHAKAM	75	06-09-06	0.68	33	29	14	51											33	29	14	51	225	225	225	390	390	390	17.10	22.80	222	221	220	A4B3A3D2	
25	JL PATUK SALAM (IND. KERAMIK)	160	12-09-06	0.72											23	23	19	22											15.21	9.51	220	219	221	D2C4B1	

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA PASAHAN"

TH. 2006 SEMESTER II

NO NO GRD	ALAMAT	DAYA (KVA)	TEL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			TEG PADA TRAFO			TEG - Ø - Ø			BEBAN (KVA)	PERSEN BEBAN	TEG UJUNG - SUTR (MOLT)			STEMPEN TEGANGAN UJUNG SUTR	
					R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T			R	S	T		R
26	445 JL TENAGA SELATAN (PT KARNEA)	630	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
27	495 JL TENAGA TIMUR 445	160	12-09-06	0.85				12	14	16	43			17	8	41	24	24	84	230	230	230	398	398	398	226	229	227	B6D4
28	528 JL J A SUPRAPTO (H. KARTIKA)	630	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
29	582 JL KARYA TIMUR (P. SPT VERONA)	160	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
30	621 JL STAPANG TENAGA	100	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
31	622 JL TENAGA BARAT	100	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
32	623 JL TENAGA UTARA (PR BENTOEL)	200	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
33	624 JL J A SUPRAPTO (H. RESENT)	630	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
34	654 JL INDUSTRI BARAT (P. PLASTIK)	100	08-09-06	0.62	35	34	45	22	7	7	41			42	41	52	63	225	225	225	225	225	390	390	390	220	221	220	A282A2
35	660 JL TENAGA (KAR. PODOJOYO)	160	06-09-06	0.66	5	5	6	31						5	5	8	31	237	237	237	237	237	410	410	410	217	220	219	A582A5
36	667 JL KARYA TIMUR	160	12-09-06	0.62	37	28	18	22						37	28	18	22	230	230	230	230	230	398	398	398	226	227	228	A283A5D2
37	668 JL TENAGA (PT. KARNEA)	250	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
38	694 JL S P SUBARNO (B. HOOK)	160	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
39	697 JL EMAS/DS. PANDEAN (PUSKOPAK)	200	12-09-06	0.66									251	236	237	31	225	225	225	225	225	390	390	390	218	220	219	D2A5B5	
40	726 JL TENAGA TIMUR	160	12-09-06	0.59				28	97	82	41			26	17	18	22	234	234	234	234	234	405	405	405	215	216	217	B3A2C5
41	734 JL TENAGA BARU I	100	12-09-06	0.75				55	47	45	55			55	47	45	55	237	237	237	237	237	410	410	410	220	221	219	B6A2
42	735 JL TENAGA BARU V	160	08-09-06	0.64	83	22	45	51					5	43	45	31	228	228	228	228	228	395	395	395	220	223	220	A785C5	
43	736 JL TAMAN TENAGA	200	11-09-06	0.66				139	66	93	31			47	67	58	22	228	228	228	228	228	395	395	395	219	225	218	B1D
44	739 JL TENAGA SELATAN (PT KARNEA)	100	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
45	752 JL KARYA TIMUR (DELTA MAS)	200	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
46	757 JL TENAGA (PAB PERMEN)	630	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
47	767 JL T. SURYO (KON. JHU HUN)	160	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
48	819 JL LETJEND. SUTOYO (MITRA II)	630	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				
49	818 JL MAHAKAM	100	12-09-06	0.67	65	66	28	32					65	66	28	32	228	228	228	228	228	395	395	395	225	224	226	A486C3D2	
50	834 JL TAMAN TENAGA (P. CAT)	160	KHUSUS		0	0	0	0	0	0				0	0	0	0	0	0	0	0	0	0	0	0				

445
621

667
778

445

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P ASAHAN "

TH.2006 SEMESTER II

NO GRD	ALAMAT	DAYA (KVA)	TEL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFO			TEG PADA TRAFO			BEBAN PERSEN BEBAN		BEBAN (KVA)		TEG UJUNG (VOLT)			STEMEN TEGANGAN UJUNG SUTR				
					R	S	T	N	R	S	T	N	R	S	T	N	R	S	T	R-0	S-0	T-0	R-0	S-0	T-0	R-0	S-0	T-0	R-0		S-0	T-0		
51	843 JL TAMAN TENAGA (FT. GRENDEL	2600	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
52	851 JL J. A. SUPRAPTO	160	11-09-06	0.85	115	42	51	31			22	23	28	72			137	65	79	83	226	226	326	326	391	391	391	219	221	223	A284C2			
53	865 JL TENAGA (PAB. PLASTIK)	200	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
54	875 JL TENAGA BARU	100	11-09-06						33	37	41	21						57	58	59	23	26	226	226	381	381	381	381	224	223	222	86A9		
55	899 JL TENAGA BARU II	1250	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
56	906 JL J. A. SUPRAPTO (DUNKIND)	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
57	910 JL J. A. SUPRAPTO (BPD JATIM)	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
58	913 JL TENAGA BARU (G. BETHANIS)	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
59	924 JL SUNANDAR P. SUDARMO	200			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
60	936 JL TENAGA BARU V	200	11-09-06	0.66	236	236	241	81									236	236	241	81	230	230	230	398	398	398	398	227	225	226	A381D5			
61	939 JL TENAGA BARU III	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
62	960 JL TENAGA BARU IV (PERCETAKA	160	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
63	1015 JL TENAGA	630	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
64	1034 JL SUNANDAR P. SUDARMO	100			0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
65	1081 JL DR CPTO	160	11-09-06	0.66					97	67	77	32					97	58	75	41	194	125	152	73	236	236	409	409	409	409	217	218	212	07
66	1082 JL TENAGA BARU IV	100	11-09-06	0.72	25	25	25	26									25	25	25	26	236	236	236	405	405	405	405	233	233	233	4983			
67	1099 JL CIWULAN	100	09-09-06	0.65					62	57	107	16					62	57	107	16	234	234	234	405	405	405	405	225	230	220	0482			



PT PLN (PERSERO)
DISTRIBUSI JAWA TIMUR
A.P. & J. MALANG

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P GLINTUNG"

TH. 2006 SEMESTER II

NO	NO BRD	ALAMAT	DAYA (KVA)	TGL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEB PADA TRAFU			TEB UJUNG SUTR (VOLT) R-Ø S-Ø T-Ø	STEBEN TEBANGAN UJUNG SUTR												
						R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T			R	S	T	R	S	T						
1	16	JL LETJEND. SUTOYO	250	15-09-06	0.68	190	92	115	49	18	13	12	7	69	25	51	12	64	48	52	22	281	178	230	90	233	233	233	404	404	404	160.54	64.21	219	220	22	A2DAA2B2
2	35	JL RETAWU	250	18-09-06	0.62					84	44	56	8				178	137	165	15	262	181	221	23	223	223	223	386	386	386	148.07	59.23	219	220	220	D1C2D3	
3	39	JL BOROBUDUR	200	19-09-06	0.65					158	170	166	51				58	85	57	41	216	255	223	92	223	223	223	386	386	386	154.76	77.38	218	217	215	B3C6	
4	40	JL LETJEND. SUTOYO	250	16-09-06	0.64	97	126	144	42					213	220	185	53				310	346	329	95	215	215	215	372	372	372	211.78	84.71	209	210	210	A3B1	
5	51	JL LETJEND. SUPARMAN	250	16-09-06	0.72	86	79	114	34					144	155	174	55				230	234	288	89	228	228	228	395	395	395	171.46	68.58	221	217	223	A9	
6	58	JL JAKARTA	150	15-09-06	0.66	57	58	53	15								35	75	51	54	92	133	104	69	231	231	231	400	400	400	76.00	50.67	228	227	225	A4D5	
7	59	JL KURUMI	160	18-09-06	0.85					67	116	141	72				41	28	28	32	108	144	169	104	225	225	225	390	390	390	94.73	59.20	219	219	220	B4C8D4	
8	75	JL PAHLAWAN TRIP	200	19-09-06	0.59												244	229	271	248	229	271	248	219	219	219	379	379	379	162.94	81.47	116	227	228	D3C3B4		
9	85	JL STUPANG IJEN	200	19-09-06	0.75					53	44	52	54				73	59	79	70	126	103	131	124	228	228	228	395	395	395	82.08	41.04	224	223	223	B6C1D8	
10	93	JL LETJEND. S. PARMAN	250	15-09-06	0.64					35	32	21	59	43	44	52	13	99	78	88	48	177	154	161	120	223	223	223	386	386	386	109.72	43.89	217	213	219	C2D4
11	107	JL KALURANG	100	16-09-06	0.66					3	6	14	8				54	53	83	23	57	59	67	31	246	246	246	426	426	426	45.02	45.02	205	200	204	D1C2D3A1	
12	114	JL SEMARANG (IKIP)	150	18-09-06	0.67	34	29	50	38												34	29	50	38	230	230	230	398	398	398	25.99	17.33	221	219	218	A3D5A3	
13	137	JL SEMARANG	150	16-09-06	0.85					30	102	36	62	90	92	97	63				120	194	133	125	239	239	239	414	414	414	106.83	71.22	228	229	230	B3C1D7	
14	138	JL BONDOWOSO	160	15-09-06	0.62												88	85	73	81	85	85	73	81	221	221	221	383	383	383	53.70	33.56	210	212	211	D4C2B4	
15	141	JL LETJEND. S. PARMAN	200	18-09-06	0.66	67	54	60	60					49	54	52	52				116	108	112	112	230	230	230	398	398	398	77.28	38.64	224	224	224	C5	
16	142	JL KALURANG BARAT	250	15-09-06	0.62	136	134	158	59								112	119	138	40	248	253	296	99	229	229	229	397	397	397	182.51	73.01	218	219	218	A4D11	
17	148	JL LETJEND. S. PARMAN	250	16-09-06	0.65	70	98	62	39					86	48	60	36				156	146	122	75	231	231	231	400	400	400	97.94	39.18	228	229	228	A7	
18	153	JL LETJEND. SUTOYO	160	19-09-06	0.60	87	89	69	82					13	5	5	8				100	94	74	89	218	218	218	378	378	378	58.42	36.52	217	216	216	A4B4A3	
19	156	JL BANDUNG (UNMUH)	100	18-09-06	0.68	9	12	37	30								4	6	18	16	13	18	55	46	223	223	223	386	386	386	19.18	19.18	222	221	221	A1D3	
20	165	JL KALURANG	200	15-09-06	0.66					188	221	146	65				26	19	18	22	216	240	164	87	244	244	244	423	423	423	151.28	75.64	221	220	221	B6A3B1	
21	166	JL GRESIK	160	19-09-06	0.68	47	32	15	31					3	16	2	7				50	48	17	38	224	224	224	388	388	388	25.76	16.10	205	205	210	A7D3C8	
22	173	JL SURABAYA (JASA TIRTA)	250	KRUSUS		0	0	0	0					0	0	0	0				0	0	0	0	0	0	0	0	0	0	0.00	0.00					
23	196	JL ACHMAD YANI	200	19-09-06	0.65	6	6	5	6	52	59	39	50	92	45	67	68				150	110	111	124	220	220	220	381	381	381	81.62	40.81	205	206	209	C7	
24	210	JL LETJEND. S. PARMAN (TELKOM)	400	KRUSUS		0	0	0	0					0	0	0	0				0	0	0	0	0	0	0	0	0	0	0.00	0.00					
25	211	JL LETJEND. SUTOYO	150	19-09-06	0.68	85	85	84	85					3	2	4	3				88	87	88	88	219	219	219	379	379	379	57.60	38.40	212	215	215	B9D6C3	
26	262	JL AMBARAWA (IKIP)	200	19-09-06	0.68					12	6	7	8	38	31	16	28				50	37	23	37	125	125	125	217	217	217	13.75	6.88	216	215	217	B4A2D1C1	



PT PLN (PERSERO)
DISTRIBUSI JAWA TIMUR
A P & J MALANG
MALANG

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P GLINTUNG"

TH. 2006 SEMESTER II

NO BRD	NO BRD	ALAMAT	DAYA (KVA)	TEL URUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEG PADA TRAFU			TEG B - B	BEBAN (KVA)	PERSEN BEBAN	TEG UJUNG SUK (VOLT)	SIFIRAN TEGANGAN UJUNG SUK							
						R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T						R	S	T				
27	281	JL BANTARAN	160	16-09-06	0.60				89	91	104	2				84	33	73	52	173	124	177	54	229	229	397	397	397	108.55	67.84	223	222	225	D4C3D1	
28	320	JL SEMPANG IJEN	190	15-09-06	0.60				66	45	62	58				65	83	128	92	131	128	190	150	216	216	382	382	382	96.98	64.66	230	209	209	B2C7	
29	398	JL LETJEND SUTOYO	160	19-09-06	0.66								57	22	19	33					57	22	221	221	383	383	383	21.65	13.54	210	211	210	C9R2		
30	432	JL AMBARAWA	200	15-09-06	0.62				16	2	2	15				61	52	36	30	77	54	38	45	224	224	388	388	388	37.86	18.93	219	117	132	B6A7B5C3	
31	437	JL KALI URANG BARAT	300	18-09-06	0.65				85	115	151	71				85	115	151	71	230	230	230	230	398	398	398	398	398	80.73	80.73	221	222	225	B2C5D5	
32	449	JL TERUSAN BOGOR (PAB. ES)	250	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00					
33	529	JL SEMPANG IJEN	160	15-09-06	0.68											25	21	27	2	25	21	27	2	229	229	397	397	397	16.72	10.45	220	227	225	D3C1D1A2	
34	536	JL BANTARAN	160	18-09-06	0.66				53	31	34	31				157	109	99	54	210	140	133	85	218	218	378	378	378	105.29	65.81	212	214	212	D2C2D7	
35	575	JL JEND. AHMAD YANI	160	18-09-06	0.68	13	29	35	18				60	29	65	49					73	58	100	67	223	223	386	386	386	51.51	32.20	214	217	218	C3D21
37	703	PRM ENDOKILO / RETAWU	280	19-09-06	0.68				88	93	72	84				64	84	68	72	152	177	140	156	219	219	379	379	379	102.71	41.08	218	228	219	B1D5	
38	724	JL BONDOWOSO	300	18-09-06	0.60				61	50	36	49				5	2	3	3	66	52	39	52	220	220	381	381	381	34.54	34.54	218	218	217	B12A2	
39	740	JL BONDOWOSO	160	19-09-06	0.68											72	74	48	65	74	74	48	65	220	220	381	381	381	42.68	26.68	219	218	219	D12	
40	762	JL KALURANG BRT (PERC. KGT)	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00					
41	860	JL BANDUNG	200	16-09-06	0.66											79	159	98	82	79	159	98	82	228	228	395	395	395	76.61	38.30	225	220	224	D5A3B1	
42	956	JL AHMAD YANI (ALFA)	200	18-09-06	0.67	163	164	156	40												163	164	156	40	219	219	379	379	379	105.78	52.89	214	215	214	A7B4A1
43	1043	JL AMBARAWA (JEP)	200	15-09-06	0.85	36	35	47	39												36	35	47	39	235	235	407	407	407	27.73	13.87	231	232	232	A6B3
44	1080	JL LETJEND S. PARMAN	160	16-09-06	0.83	5	2	3	2							128	112	93	59	133	114	96	61	227	227	393	393	393	77.86	48.66	215	217	221	D3C1D9	
45	1086	PRM VILLA PUNCAK TIDAR	200																		0	0	0	0	0	0	0	0.00	0.00						
46	1119	JL LET SUTOYO (HOTEL SANITI)	630	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00						
47	1125	JL LET SUTOYO (KANTOR PAJA)	340	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.00	0.00						
48	1133	JL KALURANG (SUNINGSARI IT)	200																		0	0	0	0	0	0	0	0.00	0.00						
49	1149	JL VETERAN DE RUMAH	160																		0	0	0	0	0	0	0	0.00	0.00						
50	1160	JL LETJEND SUTOYO	100	16-09-06	0	41	42	28	20							41	42	28	20	20	20	20	20	233	233	404	404	404	25.86	25.86	219	220	220	A5B4C6	

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P MAWAR "

TH.2006 SEMESTER II

NO	NO BRD	ALAMAT	DAYA (KVA)	TGL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEG PADA TRAFU			BEBAN (KVA)			PESEN BEBAN			TEG UJUNG			STAMPEN TEGANGAN UJUNG				
						R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T		R	S	T	R
1	15	JL BRIGJEND. SLAMET READI	250	02-09-06	0.60				158	139	124	22				98	105	79	23	256	244	203	45	220	220	220	381	381	381	154.66	61.86	203	204	202	B2C11		
2	21	JL MURIA	200	02-10-06	0.66	134	107	138	126				89	41	65	65				223	148	203	191	216	216	216	374	374	374	123.98	61.99	210	211	210	C6		
3	22	JL MAYJEND. PANJAITAN	150	03-10-06	0.85				24	37	20	27				98	120	138	119	202	178	183	188	218	218	218	386	386	386	122.73	81.82	210	211	211	D6		
4	32	JL KALURANG (SAMAJAN)	100	05-10-06	0.59	82	104	78	88				30	44	17	30				112	148	95	118	217	217	217	376	376	376	77.04	77.04	209	210	211	A6		
5	36	JL PUNCAK	250	03-10-06	0.75				189	187	156	177				163	127	193	161	352	314	349	338	206	206	206	357	357	357	209.09	83.64	202	201	220	B1C5		
6	38	JL JAKSA AGUNG SUPRAPTO	160	02-10-06	0.64	176	127	96	133										176	127	96	133	224	224	224	388	388	388	89.38	95.86	212	222	222	A2D3			
7	93	JL MAYJEND. PANJAITAN	150	04-10-06	0.66				31	35	4	23				22	151	201	125	53	186	205	148	213	213	213	369	369	369	94.57	63.05	200	200	200	D2C3D6		
8	100	JL MAYJEND. HARIO NO (UNITE	100	05-10-06	0.67								3	13	19	12	42	9	29	27	45	22	48	38	220	220	220	381	381	381	25.30	25.30	215	215	215	B5	
9	110	JL MAWAR	150	03-10-06	0.85	32	21	26	26				160	138	118	139				192	159	144	165	230	230	230	398	398	398	113.85	75.90	224	224	224	D4C2B4		
10	122	JL MAYJEND. HARIO NO	150	05-10-06	0.62				2	2	12	5				290	165	200	218	292	167	242	224	215	215	215	364	364	364	144.27	96.18	210	209	208	D2A5		
11	133	JL WZJAYA KUSUMA (EX. MAW	150	02-10-06	0.66	21	7	19	16				65	94	67	75				86	101	86	91	230	230	230	398	398	398	62.79	41.86	228	227	227	D2C5		
12	150	JL CIANJUR	100	04-10-06	0.62											114	78	122	105	114	78	122	105	214	214	214	371	371	371	67.20	67.20	208	208	208	D2A4		
13	186	JL BOROBUDUR	150	02-10-06	0.65				113	125	163	134				143	169	210	174	250	344	386	327	214	214	214	371	371	371	209.72	83.89	208	209	208	D8C6		
14	193	JL CENGEH	250	03-10-06	0.60				107	175	176	153				129	115	102	115	243	193	187	208	218	218	218	378	378	378	135.81	67.91	212	211	213	B9		
15	230	JL LETJEND. SUTCHO V	200	06-10-06	0.62				114	78	85	92				18	2	15	12	182	193	204	193	217	217	217	376	376	376	126.64	78.53	212	211	210	B4A5D4		
16	231	JL PARANG TRITIS	160	04-10-06	0.67				164	191	169	181				75	89	70	78	121	165	145	144	216	216	216	374	374	374	93.10	46.55	211	211	210	B5		
17	235	JL SUKAPURA	200	02-10-06	0.62	28	48	37	38	18	28	28								249	325	246	273	225	225	225	390	390	390	184.50	73.80	219	219	220	C3B7		
18	237	JL PANGGALAN	250	06-10-06	0.66	190	175	163	176				59	150	83	97																					
19	240	JL MAYJEND. PANJAITAN	150	03-10-06	0.62				134	22	42	66				62	77	59	66	196	99	101	132	218	218	218	378	378	378	86.33	53.96	212	213	213	B8		
20	244	JL MALABAR	300	04-10-06	0.65				49	75	37	54				8	8	33	16	57	83	70	70	220	220	220	381	381	381	46.20	46.20	218	218	216	C6		
21	253	JL PISANG KIPAS	150	03-10-06	0.66				74	87	119	93				129	110	66	102	203	197	185	195	216	216	216	374	374	374	126.36	84.24	211	210	212	D3B4A3		
22	265	JL PANJAITAN	150	06-10-06	0.68				78	86	93	86				91	21	16	43	169	107	109	128	216	216	216	374	374	374	83.16	85.44	208	209	210	A4C2		
23	270	JL BOROBUDUR BARAT	150	04-10-06	0.69				166	115	204	162				23	39	24	29	189	154	228	190	220	220	220	380	380	380	125.62	83.75	216	217	217	B2C8		
24	273	JL M. T. HARIO NO (UNITEBRA)	200	02-10-06	0.72	26	18	14	19				13	18	25	19				39	36	39	38	219	219	219	379	379	379	24.97	12.48	217	217	216	D3C2D1		
25	284	JL KALPATARU	150	04-10-06	0.74	112	177	81	123				12	25	46	28				124	202	127	151	214	214	214	371	371	371	96.94	64.63	211	210	211	D2A2B4		



PT PLN (PERSERO)
DISTRIBUSI JAWA TIMUR
A P & J MALANG

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P MAWAR "

TH.2006 SEMESTER II

NO	NO GRD	ALAMAT	DAYA (KVA)	TGL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEG PADA TRAFU			TES Ø - Ø			BEBAN (KVA)	PERSEN BEBAN	TEG UJUNG			STEMPEN TEGANGAN UJUNG				
						R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T			R	S	T		R	S	T	
26	290	D5 TEMBALANGAN	150	04-10-06	0.77				49	57	39	47				148	155	119	141	197	212	154	188	209	209	209	362	362	362	117.67	78.44	205	205	204	85	
27	305	JL KEDAWUNG	160	04-10-06	0.77				101	115	95	104				35	76	29	47	136	191	124	150	223	223	223	386	386	386	100.57	62.86	212	215	215	D3C2	
28	306	JL L S PARMAN /BAJANG RAITU	160	10-10-06	0.82				9	22	19	17				91	105	142	113	100	127	161	129	209	209	209	362	362	362	81.09	50.58	205	206	203	R7A3	
29	315	JL J A SUPRPTO II	150	02-10-06	0.37				22	28	29	26				178	151	149	159	200	179	178	186	221	221	221	383	383	383	123.10	82.06	210	211	210	D6B1	
30	323	JL BOGOR	150	05-10-06	0.66								54	107	48	70				82	183	74	113	214	214	214	371	371	371	72.55	48.36	211	212	210	A5D5	
31	335	JL LEMBANG	150	05-10-06	0.83								12	21	26	20				32	69	69	57	221	221	221	383	383	383	37.57	25.05	220	220	219	A3B3D2	
32	337	JL UNIBRA POLITEKNIK	1000	03-10-06																											0.00	0.00				
33	398	JL MAWAR SELATAN	150	04-10-06	0.68											95	58	75	63	83	106	87	226	226	226	391	391	391	59.21	39.47	224	223	223	A5B1		
34	361	JL MAYJEND. PANJAITAN	160	10-10-06																											0.00	0.00				
35	366	JL NUJSA INDAH	150	02-10-06	0.65								156	144	147	149																				
36	369	JL BANTARAN TERUSAN	150	03-10-06	0.65											117	120	123	120																	
37	370	PRM BTM / CAINDI MENDUT	150	03-10-06	0.65																															
38	390	D5 TEMBALANGAN / KALPATAR	150	04-10-06	0.93																															
39	408	JL KEMBANG TURI	200	10-10-06	0.66																															
40	409	JL SARANGAN	150	02-10-06	0.57																															
41	412	JL MAYJEND HARIONO (UNBRA)	160	05-10-06	0.66								11	30	15	19																				
42	424	PRM BTM / SIMP BOROBUDE	250	10-10-06	0.83											113	139	180	144																	
43	425	PRM BANTARAN INDAH	100	03-10-06	0.65											26	40	22	29																	
44	455	DK BUKIR SARI / TULLUS REJO	250	10-10-06	0.68											171	127	96	131																	
45	460	JL M. THARYONO (UNBRA)	1250	05-10-06																																
46	485	JL KEDAWUNG	200	07-10-06	0.65											142	172	113	142	142	172	113	142	218	218	218	378	378	378	93.09	46.54	212	215	215	D4B6	
47	520	JL LEBAY SARI	160	09-10-06	0.68											75	129	109	104																	
48	927	JL KALPATARU	150	02-10-06	0.62											116	41	96	84	162	57	123	114	217	217	217	376	376	376	74.21	49.48	210	211	210	B2A3	
49	542	JL CENGGER AVAM	150	07-10-06	0.65											97	64	145	99	137	88	152	126	215	215	215	372	372	372	81.06	54.04	209	209	210	D2B1A1	
50	543	JL KALPATARU	150	06-10-06	0.64											183	147	219	183	183	147	219	183	218	218	218	378	378	378	119.68	79.79	213	211	213	D5C4	



PT PLN (PERSERO)
DISTRIBUSI JAWA TIMUR
MIRNABALUR A P & J MALANG

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)

"UJ KOTA P MAWAR "

TH.2006 SEMESTER II

NO GRD	ALAMAT	DAYA (KVA)	TEL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEG PADA TRAFU			BEBAN (KVA)	PERSEN BEBIAN	TEG UJUNG			STEMPEN TEGANGAN UJUNG			
					R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T			R	S	T		R-O	S-O	T-O
51	565 JL MAYJEND PANJAITAN	100	10-10-06	0.72	11	13	6	10	219	219	11	13	6	10	219	219	11	13	6	10	219	219	379	379	379	6.57	217	217	218	A7	
52	570 JL MAYJEND HARIOHO	160	05-10-06	0.65							68	100	106	91	124	166	142	216	216	216	216	216	374	374	374	92.02	280	211	211	B8C3	
53	584 JL SEMBOJA	160	09-10-06	0.68							58	64	96	73	182	181	181	224	224	224	224	224	388	388	388	76.16	218	219	217	B4A3	
54	505 JL TEMBALANGAN	200	05-10-06	0.66							98	136	84	107	203	255	202	214	214	214	214	214	371	371	371	141.24	209	210	208	D11A2	
55	588 PRM BOROBUDUR	160	05-10-06	0.68							137	123	147	136	161	162	198	160	229	229	229	229	397	397	397	110.15	225	227	226	B9	
56	605 JL BANTARAN BARAT	100	07-10-06	0.62							19	25	15	20	88	82	75	82	214	214	214	214	371	371	371	52.43	211	211	210	B6A2	
57	615 JL SRI GADING	250	09-10-06	0.66							136	131	106	124	289	279	241	270	211	211	211	211	365	365	365	170.70	207	207	205	D3C2B4	
58	689 JL UNIBRA (FAK. EKONOMI)	160																													
59	706 JL B S READI (SMAK S. YOSEP	100																													
60	729 JL J. A. SUPRAPTO (B MANDIRI	160																													
61	737 JL CENGEH	250	06-10-06	0.68							80	37	45	54																	
62	741 JL MELATI	160	06-10-06	0.66																											
63	744 JL CANDI ARUNG	250	07-10-06	0.62							20	5	13	13																	
64	764 JL BOROBUDUR	200	07-10-06	0.65							166	158	155	160																	
65	779 JL CANDI SARI UTARA	250	07-10-06	0.60																											
66	777 JL MAYJEND PANJAITAN	250	09-10-06	0.68							182	184	207	191																	
67	857 JL KANTOR REKTORAT UNIBRA	200	09-10-06	0.62																											
68	920 JL SEMANGGI TIMUR	100	09-10-06	0.65																											
69	937 JL LAB BIOLOGI UNIBRA	160																													
70	972 JL SUKARNO-HATTA (UNIBRA	200																													
71	998 JL KENONGO / MALANG	160	10-10-06	0.66																											
72	1069 JL CANDI MENDUT	100	07-10-06	0.85							61	56	58	58																	
73	1071 JL GUNTUR	160	09-10-06	0.59							91	93	116	100																	
75	1084 JL MT HARYONO (PAR ES)	200	10-10-06	0.79																											
76	1095 JL SOEKARNO - HATTA	160	07-10-06	0.64							76	42	39	52																	



PT PLN (PERSERO)
DISTRIBUSI JAWA TIMUR
A.P & J. MALANG

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P MAWAR "

TH.2006 SEMESTER II

NO	NO GRD	ALAMAT	DAYA (KVA)	TEL UKUR	Cos f	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			FADA TRAF0			TEG PADA TRAF0			TEG @ . @			BEBAN (KVA)	PERSEN BEBAN	TEG UJUNG		STEMPEN TEGANGAN UJUNG				
						R	S	T	N	R	S	T	N	R	S	T	N	R	S	T	N	R	S	T	R-0	R-5			R-T	S-T		R-0	S-0	T-0	
77	1101	Jl. BUKIT SARI / TULLUS REJO	160	09-10-06	0.66	38	9	61	36											120	116	185	140	234	234	234	405	405	405	98.51	61.57	224	225	223	C6D9C2D10
78	1124	Jl. IKAN PAUS	200	09-10-06	0.67	73	91	24	63											73	91	24	63	217	217	217	376	376	376	40.80	20.40	214	213	214	A1B4
79	1157	Jl. SOEKARNO - HATTA	160																	0	0	0	0	0	0	0	0	0	0	0.00	0.00				
80	1158	Jl. BOROBUDUR	160																	0	0	0	0	0	0	0	0	0	0	0.00	0.00				

DATA PENGUKURAN BEBAN DAN TEGANGAN UJUNG (MALAM)
"UJ KOTA P WENDIT"

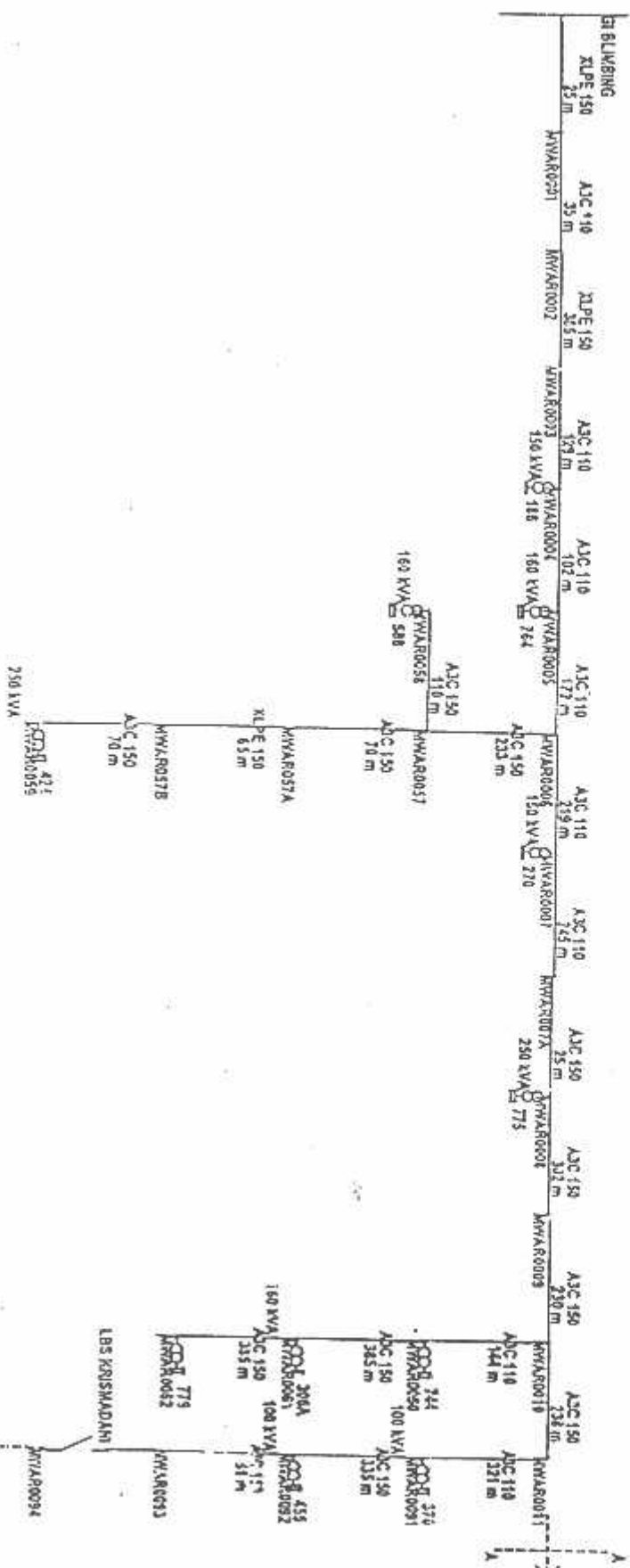
TH. 2006 SEMESTER II

NO NO	ALAMAT	DAYA (KVA)	TEL UKUR	Cos f	JURUSAN 'A'			JURUSAN 'B'			JURUSAN 'C'			JURUSAN 'D'			PADA TRAFU						TEG PADA TRAFU						BEBAN (KVA)	PERSEN BEBAN	TEGANGAN UTUNG			STEMBEN TEG ULUAG
					R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	N	S	T	R	S	T	R	S	T			R-0	S-0	T-0	
1	ZB JL RAYA WENDIT	160	17-10-06	0.68				139	133	83	118	223	223	223	386	386	386	79.17	49.48	219	219	219	217	83A9										
2	185 JL TELUK CENDRAWASIH	150	16-10-06	0.62				82	102	126	103							112.05	74.70	221	221	221	223	D8C6B1										
3	202 JL TELUK GRAJAKAN	250	16-10-06	0.65				52	12	30	31							36.74	14.70	217	217	217	217	D1C2D3										
4	203 JL RAYA WENDIT	160	17-10-06	0.64				42	17	19	26							33.08	20.67	218	218	218	215	B9C6										
5	215 DS PLAOSAN BARAT	200	16-10-06	0.72				51	106	99	85							92.40	46.20	214	214	214	211	D3C7										
6	222 JL POJAM / WENDIT	630	KHUSUS					0	0	0	0							0.00	0.00															
7	224 JL L.A SUPTO	160	16-10-06	0.85				43	118	64	75							72.68	45.43	212	212	212	215	B6A1										
8	254 JL TELUK BONE	150	KHUSUS					0	0	0	0							0.00	0.00															
9	293 DS MANGLAWAN	160	17-10-06	0.75				26	12	8								38.06	23.78	210	211	210	210	B7A3B4C4										
10	388 DS WONOSULAM	200	18-10-06	0.64				52	58	43	51							93.68	46.84	219	219	219	215	B2A4										
11	378 JL RABEN INTAN	200	18-10-06	0.66				245	220	204	223							155.36	77.68	205	210	206	206	D1C9B5C1										
12	428 JL TERUSAN GORONTALO	100	18-10-06	0.67				29	72	35	45							59.57	59.57	230	210	210	210	C2D1C6B4										
13	469 JL ARJOSARI (P36T)	630	KHUSUS					0	0	0	0							0.00	0.00															
14	502 DS MANGLAWAN	160	17-10-06	0.62														11.45	7.35	205	206	209	08											
15	503 DS MANGLAWAN	160																0.00	0.00															
16	537 PRM POND BLIMBING INDAH	200	18-10-06	0.62				56	63	67	62							129.65	64.82	208	206	204	D11											
17	584 PRM KOMP ARAYA (VILLA GRANDY)	160																0.00	0.00															
18	577 DS PLAOSAN TIMUR -	250	16-10-06	0.60				242	197	201	213							206.95	83.58	207	206	204	204	B16A4										
19	606 JL GRAHA TELAGA GOLF	160	19-10-06	0.68				2	1	2	2							19.85	12.40	216	215	217	217	A2D1C11										
20	608 JL MANGLAWAN	150	17-10-06	0.66				90	90	90	90							62.71	41.81	210	210	211	211	B2A8										
21	649 JL TIRTOMOYO / PAKIS	250	18-10-06	0.68				158	178	185	164							197.12	78.85	220	219	218	218	D1A2										
22	650 JL TIRTOMOYO / PAKIS 17/4	160	18-10-06	0.66				117	132	71	107							133.70	83.57	230	208	205	205	B7A4C3										
23	655 PRM ASABRI / MANGLAWAN	100	18-10-06	0.72				18	14	22	18							32.27	32.27	216	218	217	217	D8C7B1										
24	659 DS MANGLAWAN (P. PLASTIK P)	800	KHUSUS	0.66				0	0	0	0							0.00	0.00															
25	676 DS MANGLAWAN (C SETILAWAN	200	KHUSUS	0.64				35	33	26	31							35.97	17.99	214	215	214	214	D3C6										
26	681 PRM POND BLIMBING INDAH	150	19-10-06	0.66				55	32	46	44							55.44	36.96	207	207	208	208	B6C3B2										

DATA PENGUKURAN BEBAN DAN TEGANGAN UTJUNG (MALANG)
"UJ KOTA P WENDIT"

TH.2006 SEMESTER II

NO GRD	ALAMAT	DAYA (KVA)	TEL UKUR	Cos φ	JURUSAN "A"			JURUSAN "B"			JURUSAN "C"			JURUSAN "D"			PADA TRAFU			TEG PADA TRAFU			BEBAN (KVA)	PERSEN BEBAN	TEGANGAN UTJUNG			STEMEN TER. UTJUNG														
					R	S	T	R	S	T	R	S	T	R	S	T	R	S	T	R	S	T			R	S	T															
54	967	PMJ SAPTO RAYA (SAPTO RENEGGO	160	21-10-06	0,62				6	6	5	6							6	4	5	5	12	10	10	11	226	226	226	391	391	391	7,23	4,52	221	219	220	D8C2				
55	968	PMJ SAPTO RAYA (SAPTO RENEGGO	160	21-10-06	0,66				21	25	39	28							51	55	34	47	72	80	73	75	221	221	221	383	383	383	49,73	31,08	215	215	214	D6A3B6				
56	974	PMJ PONDOK BLIMBING DUDAH	160	21-10-06	0,62														47	63	59	56	47	63	59	56	220	220	220	381	381	381	37,18	23,24	216	214	214	D10				
57	991	JL MANGELAWAN (PAR. SENTENG	160	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
58	997	D5 BALU SAPTO RENEGGO	160	21-10-06	0,66	12	68	62	47														12	68	62	47	213	213	213	369	369	369	30,28	18,90	209	209	208	A9D3C2				
59	999	PMJ KOTA ABAYA BLIMBING	25	20-10-06	0,68	14	17	9	13														14	17	9	13	221	221	221	383	383	383	8,84	35,36	218	218	219	A12B3				
60	1000	PMJ KOTA ABAYA BLIMBING	200	20-10-06	0,65	102	99	75	92														117	125	104	115	220	220	220	381	381	381	76,12	38,06	218	215	216	A1D4C3D1				
61	1001	PMJ KOTA ABAYA BLIMBING	200	20-10-06	0,65				14	17	9	13							4	17	3	8	18	34	12	21	225	225	225	390	390	390	14,40	7,20	220	220	221	D3A4				
62	1002	PMJ KOTA ABAYA BLIMBING	200	20-10-06	0,68	5	2	1	3	12	10	4	9										17	12	5	11	226	226	226	391	391	391	7,68	3,84	225	225	228	B6A4				
63	1003	PMJ KOTA ABAYA BLIMBING	160	20-10-06	0,65																		9	2	4	5	227	227	227	393	393	393	3,41	2,13	224	224	225	C7D4				
64	1004	PMJ KOTA ABAYA BLIMBING	300	20-10-06	0,65	14	2	6	7														14	2	6	7	225	225	225	390	390	390	4,95	4,95	214	217	218	A9				
65	1005	PMJ P B I / PURI PALMA	200	01-11-06	0,65	9	4	3	5	26	14	24	21	22	12	22	19						57	30	49	46	226	226	226	391	391	391	30,74	15,37	218	217	219	B1D5				
67	1006	PMJ PONDOK BLIMBING DUDAH	160	21-10-06	0,64	106	132	135	124														106	132	135	124	224	224	224	388	388	388	83,55	52,22	220	219	219	A3B9				
68	1040	D5 MANGELAWAN (PAR. KAYU)	160	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
69	1041	JL RADEN INTAN (PAR. SEPATU)	100	KHUSUS		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
70	1054	JL TELUK TOWNIN	160	16-10-06	0,68	155	39	154	116														155	39	154	116	209	209	209	362	362	362	72,73	45,46	210	211	212	A1D11				
71	1055	PMJ VILLA GRANDY II ABAYA	200	01-11-06	0,66				2	8	4	5	5	5	5	5	2	2	1				9	15	10	10	226	226	226	391	391	391	7,68	3,84	220	219	218	C10B3				
72	1056	PMJ VILLA GRANDY II ABAYA	20	01-11-06	0,68	9	2	12	8														16	18	29	21	230	230	230	398	398	398	14,49	72,45	225	227	226	A3B7C1				
73	1066	JL TEL. GRAYAKAN	160	16-10-06	0,62				52	49	38	46											21	7	20	16	73	55	58	62	221	221	221	383	383	383	41,33	25,83	216	217	219	D6A4



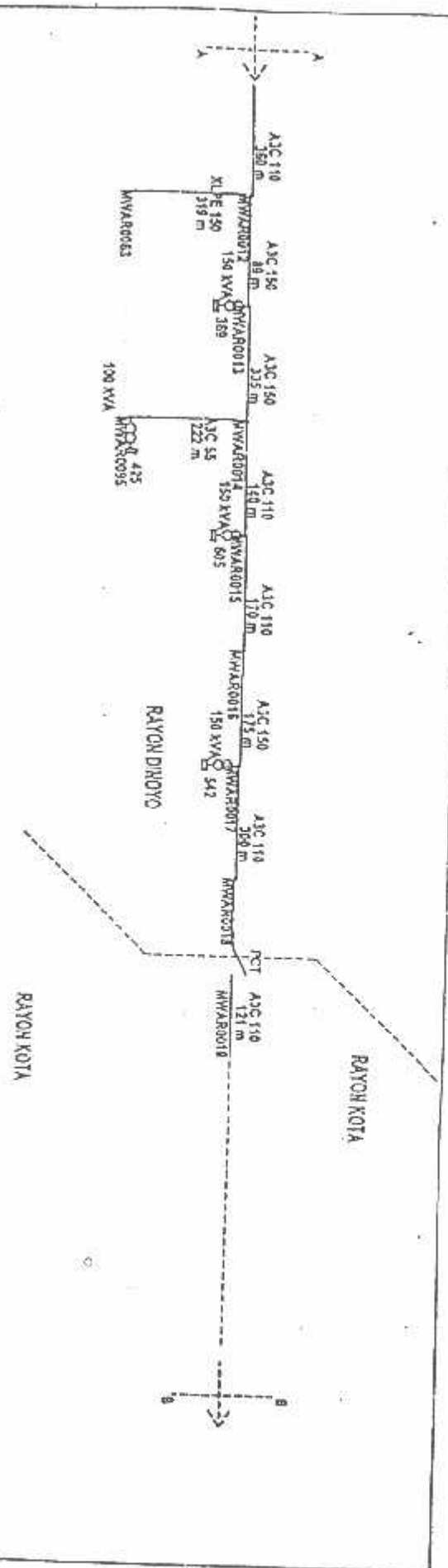
PT. PLN (PERSERO)
JASA ENJINIRING

Project Name: **PENURUNAN SUSUT KWH**
PT. PLN (PERSERO) DISTRIBUSI LAMPIR - CABANG MALANG

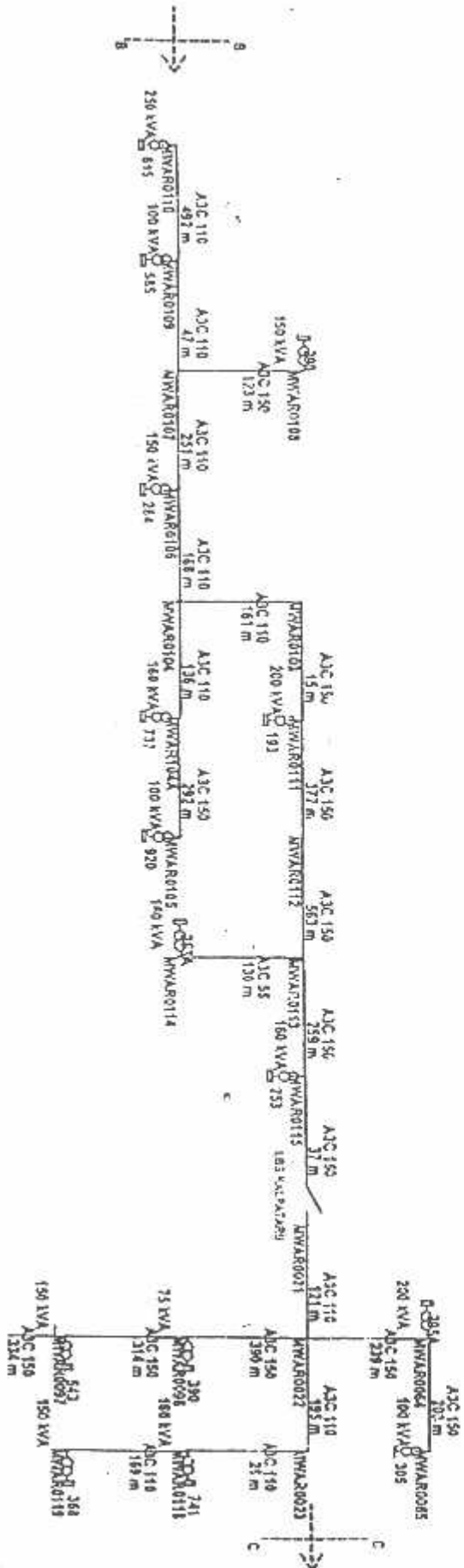
Task: **ANALISA EHNK SHAKTIKAWA (TITIK KAWA)**

Design By: _____
 Drawn By: _____
 Checked By: _____
 Approved By: _____
 Drawing No: _____

4102-3100-1.51.05.10004.016.01-0
 Rev 0



	
PT. PLN (PERSERO) JASA ENJINIRING	
Proyek / Lokasi PERMITSURAN SUBSTANSI PT. PLN (PERSERO) DISTRIBUSI ATM - CABANG MALANG	
Tahap SHEDULIRUP TEKNIK/AMPEL HUBUNG MANUFA	
Dibuat By (Name)	(Drawing No)
Disetujui By (Name)	(Date)
Disetujui By (Name)	(Date)
(Title)	(Scale)

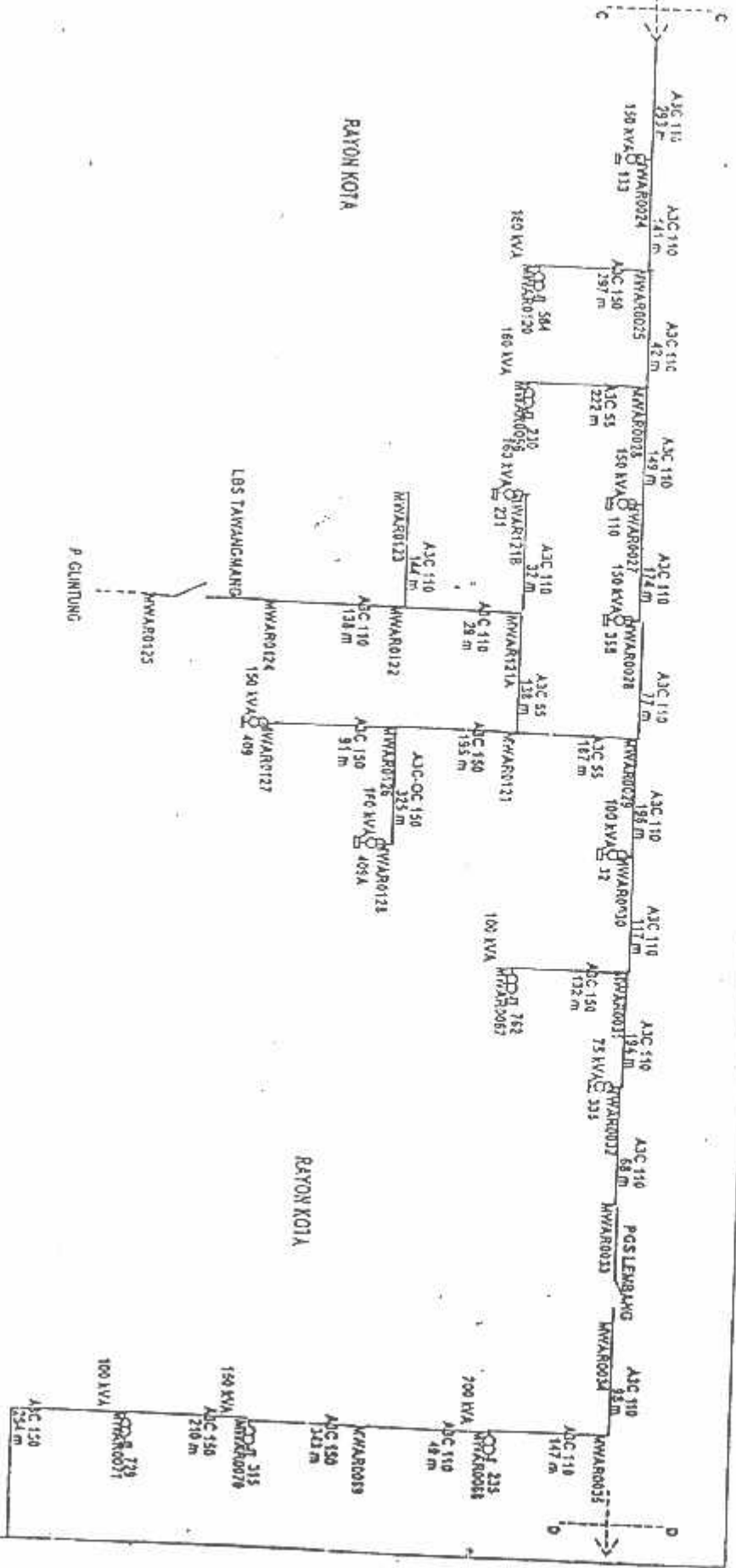


PT. PLN (PERSERO)
JASA ENGINEERING

Project Name: PERAIRAN AIRBUT MAI
PT. PUN (PERSERO) DISTRIBUSI JATIM - CADANG MALANG

TYPE: SINGLE LINE DIAGRAM PERULANG BARU

Designed By:	Checked By:
Approved By:	4180.3100.1.4-OS.MYAS.218.050
Created By:	Rev: 0
Date:	



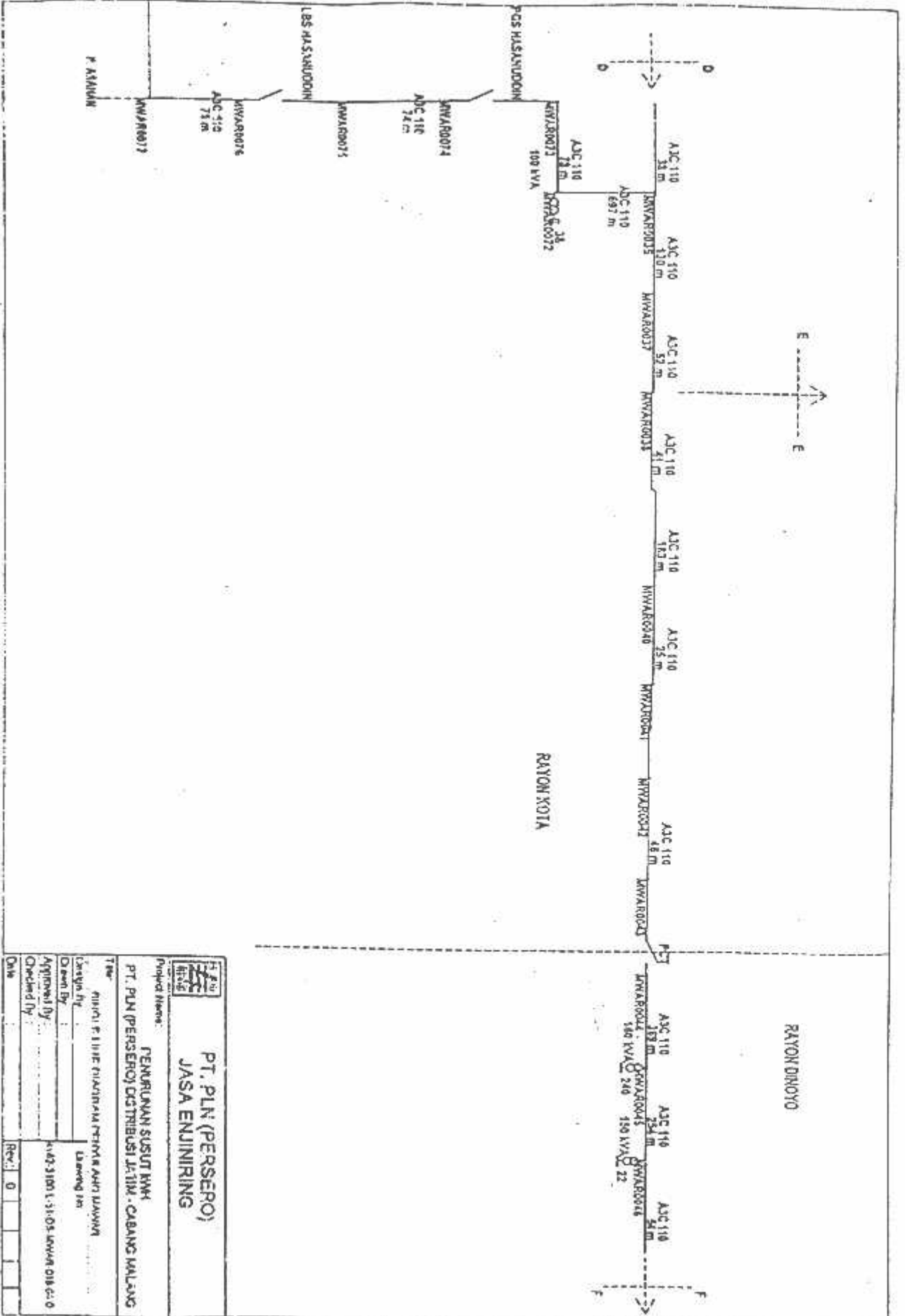
**PT. PLN (PERSERO)
JASA ENJINIRING**

Project Name
**PT. PLN (PERSERO) SUBSIDI BMT
 PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG KULAMO**

Drawn By
 Checked By
 Approved By
 Design No.:

Drawn By
 Checked By
 Approved By
 Design No.:

Rev: 0



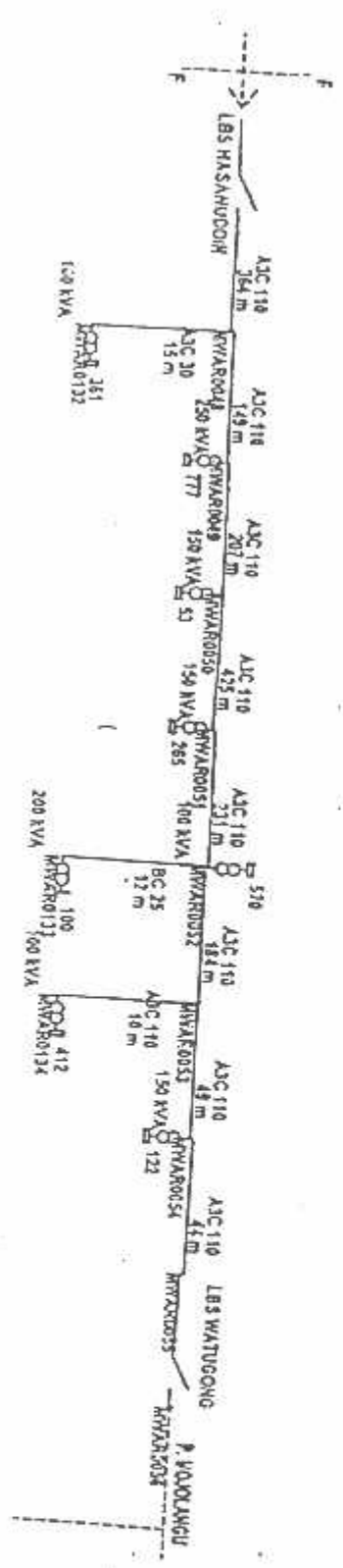



PT. PLN (PERSERO)
JASA ENJINIRING

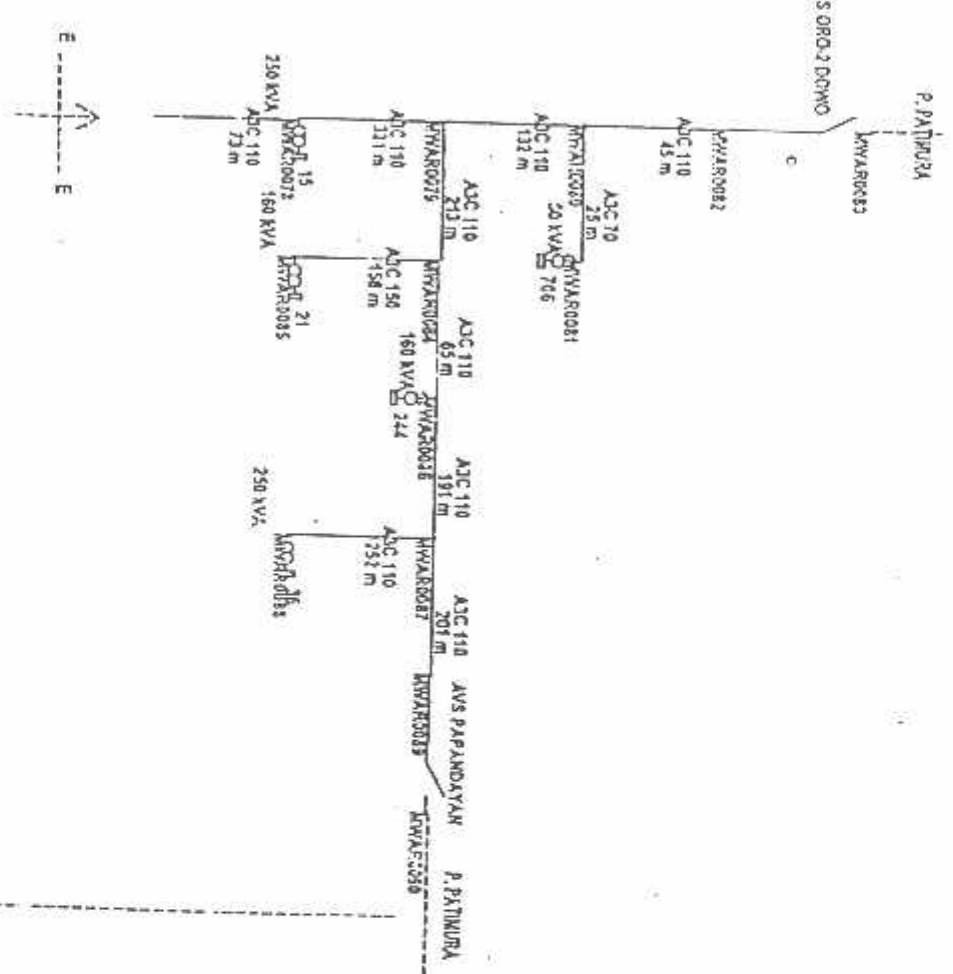
Project Name:
PERMURAHAN SUSUT RUMAH
PT. PLN (PERSERO) DISTRIK BUKIT BARU - CABANG MALANG

Title:
PERENCANAAN SISTEM TENAGA LISTRIK
 Design By: [Blank]
 Drawn By: [Blank]
 Approved By: [Blank]
 Checked By: [Blank]


Date: [Blank] Rev.: 0

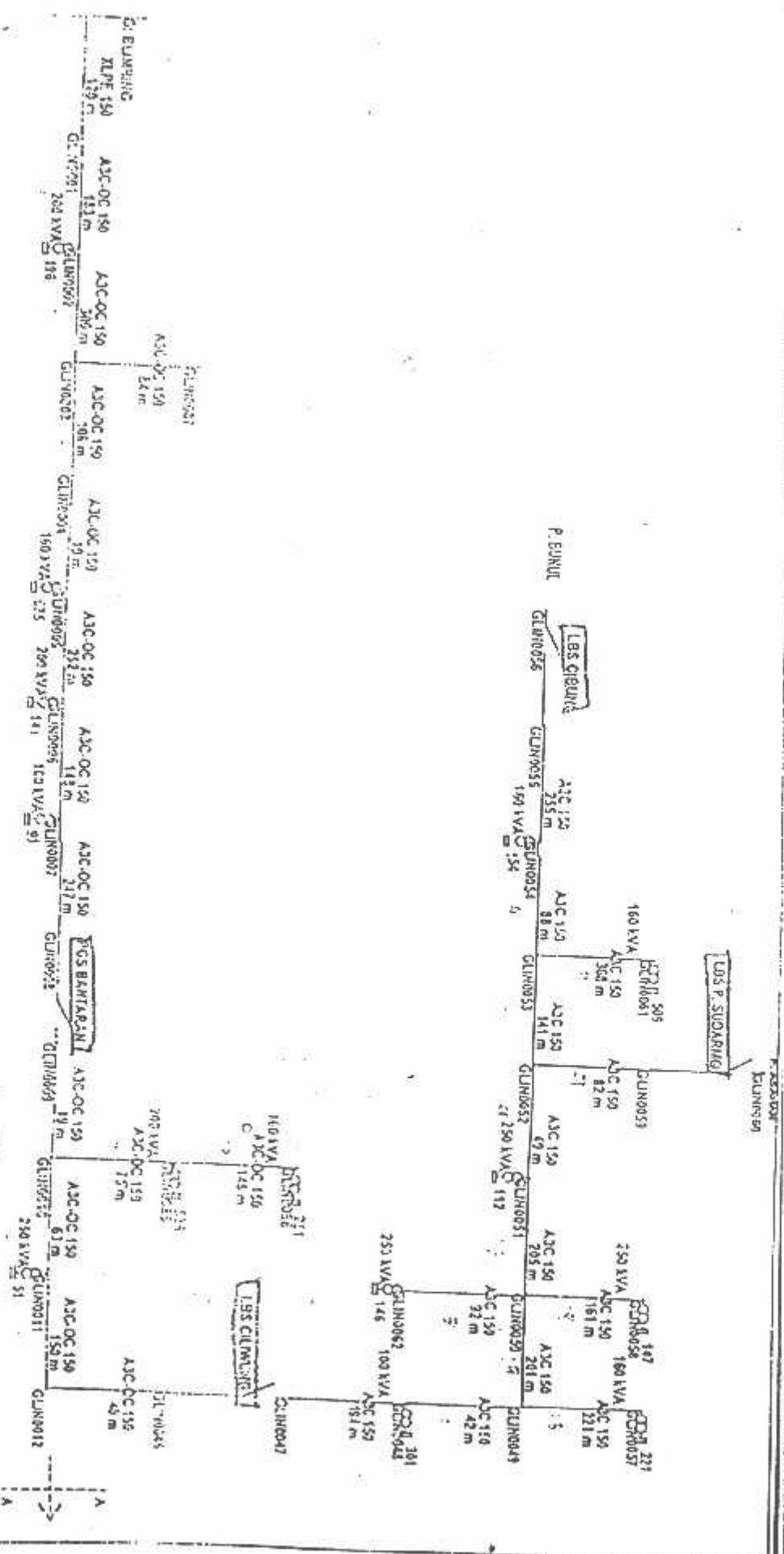


 PT. PLN (PERSERO) JASA ENJINIRING	
Project Name	
PENURUN SUSUT IAH	
PT. PLN (PERSERO) DISTRIKSI JATIM - CABANG MALANG	
Title	
SINGLE LINE DIAGRAM ITIWA AND MANGUN	
Drawn By	Drawing No.
Approved By	
Checked By	4183.31001.51-05.MAWAR.01602-0
Date	
Scale	



FAYOR DINOYO

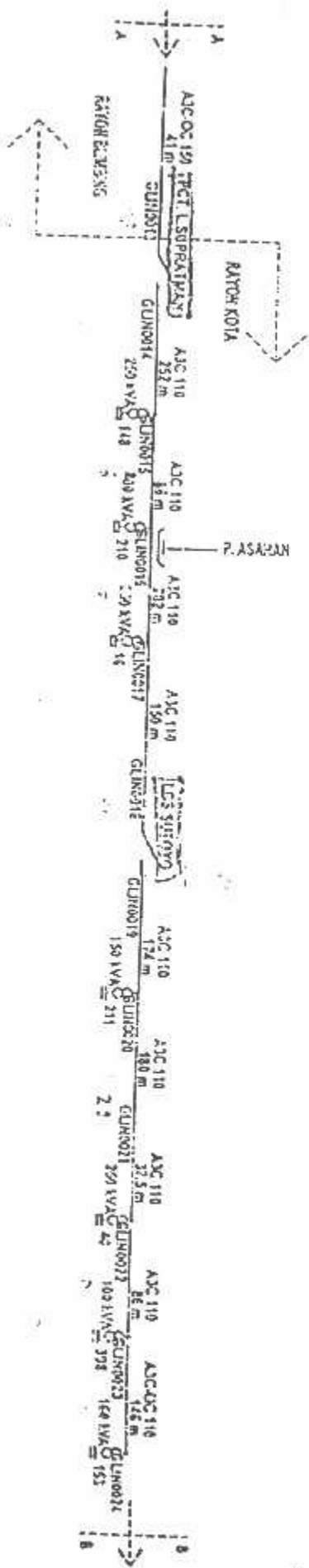
 <p>PT. PLN (PERSERO) JASA ENJINIRING</p>	
Project Name: PENANJARAN SUSUT RUMAH PT. PLN (PERSERO) DISTRIKUSI JATIM - CABANG MALANG	
Title: SINGLE LINE DIAGRAM PENYALURAN MALANG 2.2/98	
Design By: Checked By: Drawn By:	Drawing No.: 1102-3100-1-31-05-MALANG-018-07-0



PT. PLN (PERSERO)

JASA ENJINIRING

Project Name	PENURUNAN SUSUT KWH
Client	PT. PULIPERSEPODISTRIBUSI LAIN - CARANG MAUNG
Location	SUKALEHRE DIKANDIA PERYADAN GUNUNG
Scale	1:1000
Author	Andriani H
Check	Andriani H
Date	10/10/2013
Revision	1



**PT. PLN (PERSERO)
JASA ENJINIRING**

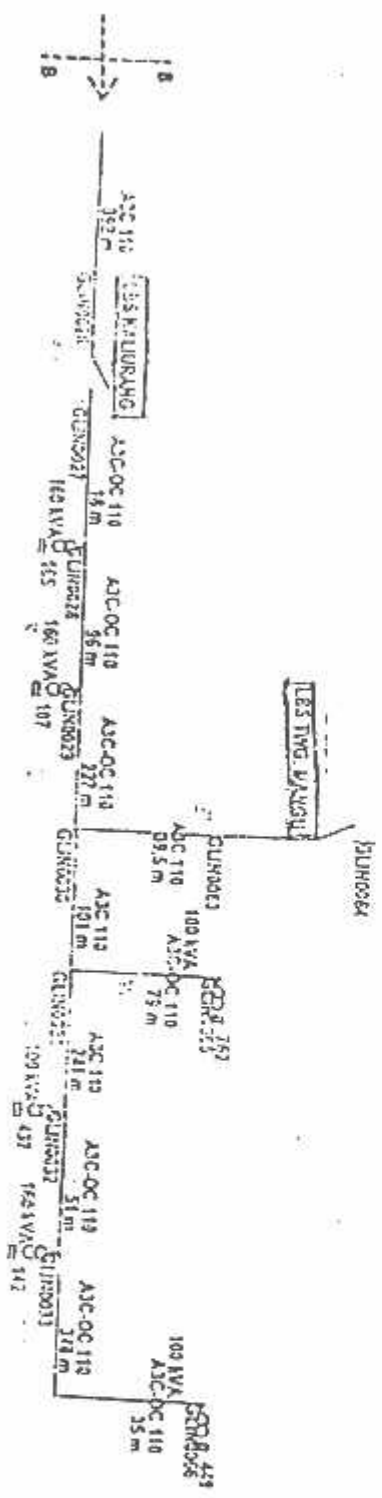
Project Name
PT. PLN (PERSERO) DISTRIKSI JATIM - CABANG MALANG

Task
SURVEI LINTAS DIARUMAH PERUMAHAN KUMUHANKS

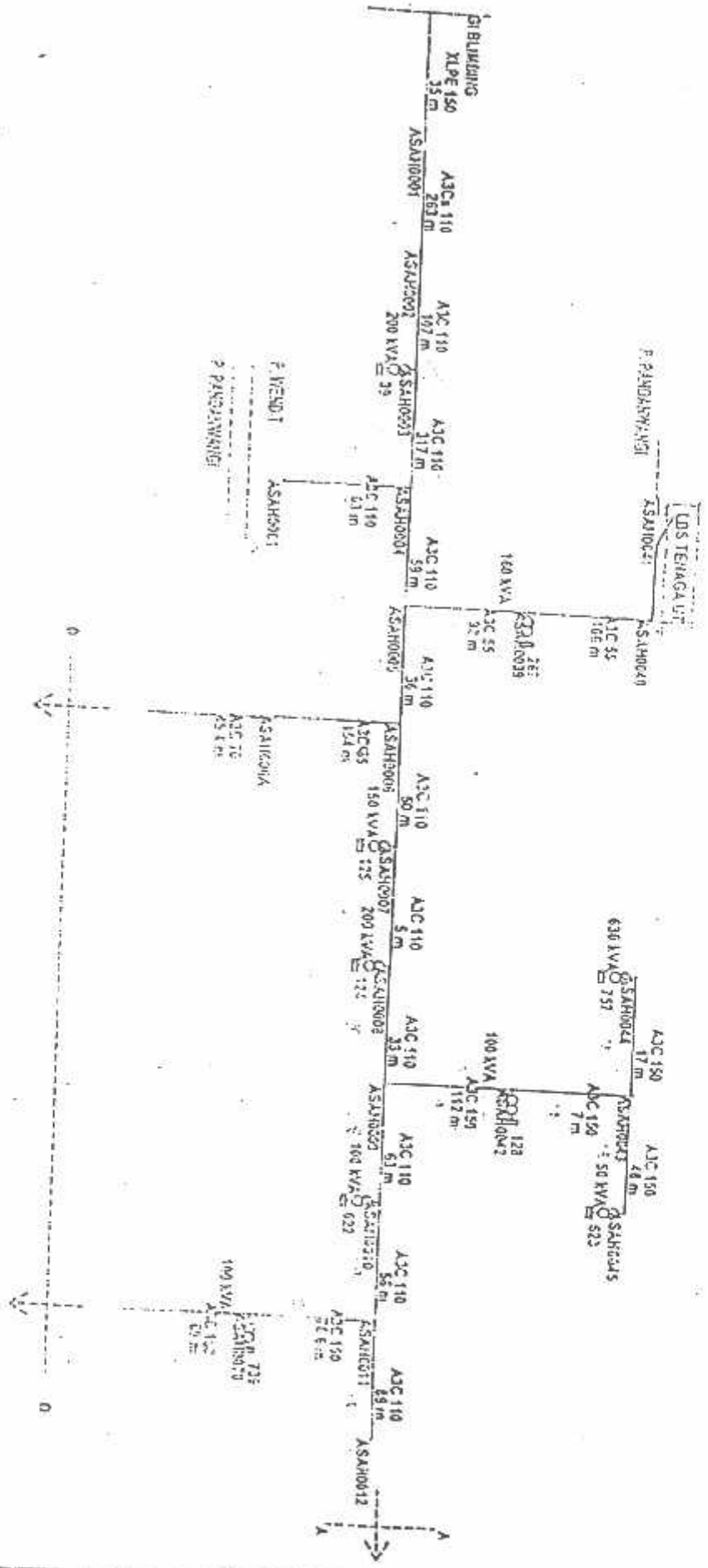
Drawn By
[Signature]

Checked By
[Signature]

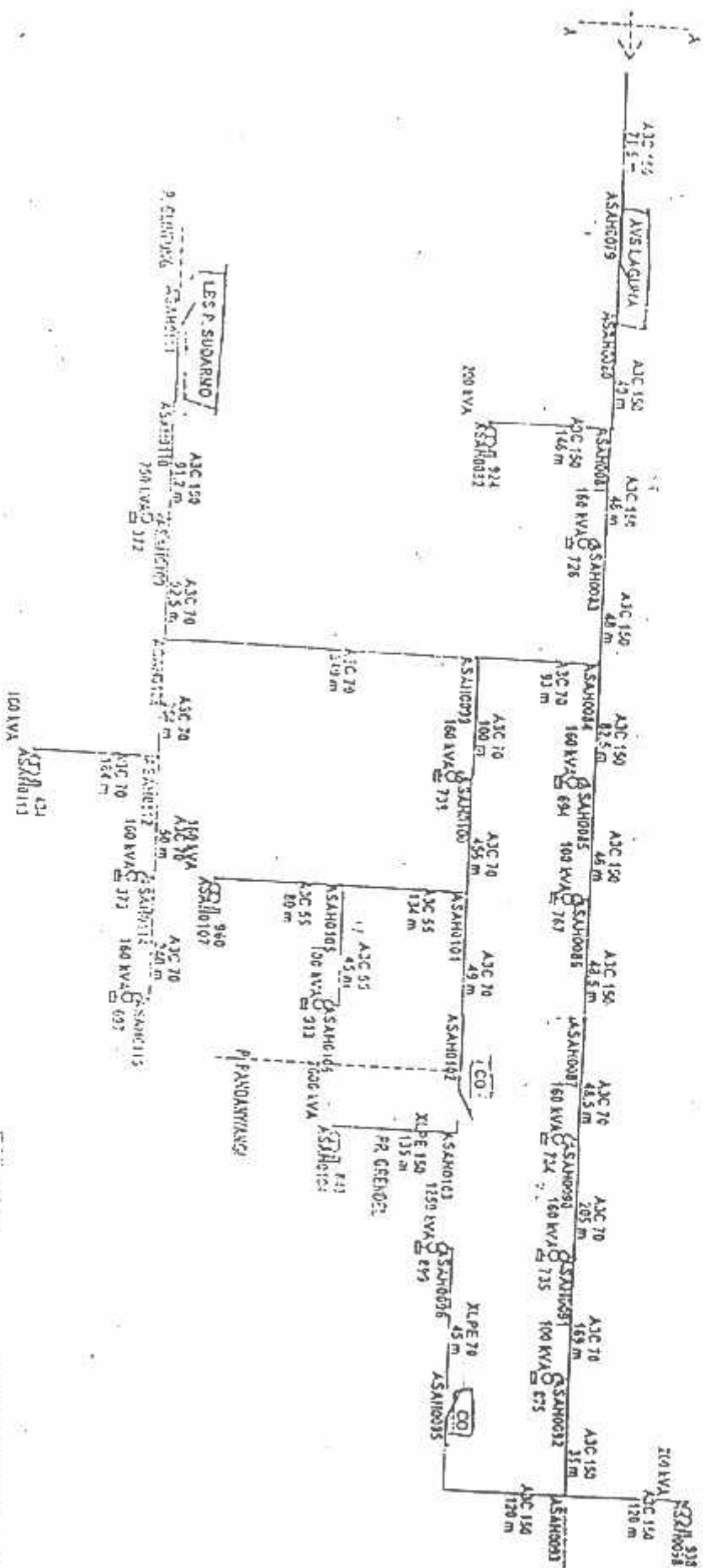
Date
0 / 0 / 19



PT. PLN (PERSERO) JASA ENJINIRING	
Project Name: PENUNJANG SUSUT WWH	
PT. PLN (PERSERO) DISTRIBUSI JATIM - CANTIC MANG	
Title: SINGLE LINE DIAGRAM PENYULANG CANTIC	
Design By: _____	Reviewed By: _____
Drawn By: _____	Checked By: _____
Date: _____	Scale: _____



	PT PLN (PERSERO)
	JASA ENGINEERING
Project Name: PERUBAHAN RANGKAIAN DI STASIUN TRANSFORMATOR 100 MVA DAN PENYALURAN TENAGA	No: SINGLE LINE DIAGRAM RANGKAIAN ASAH DAN PENYALURAN TENAGA
Design By: Checked By: Approved By:	Date: 15/05/2015
Prepared By: 15/05/2015	No. Dokumen: 15/05/2015/01



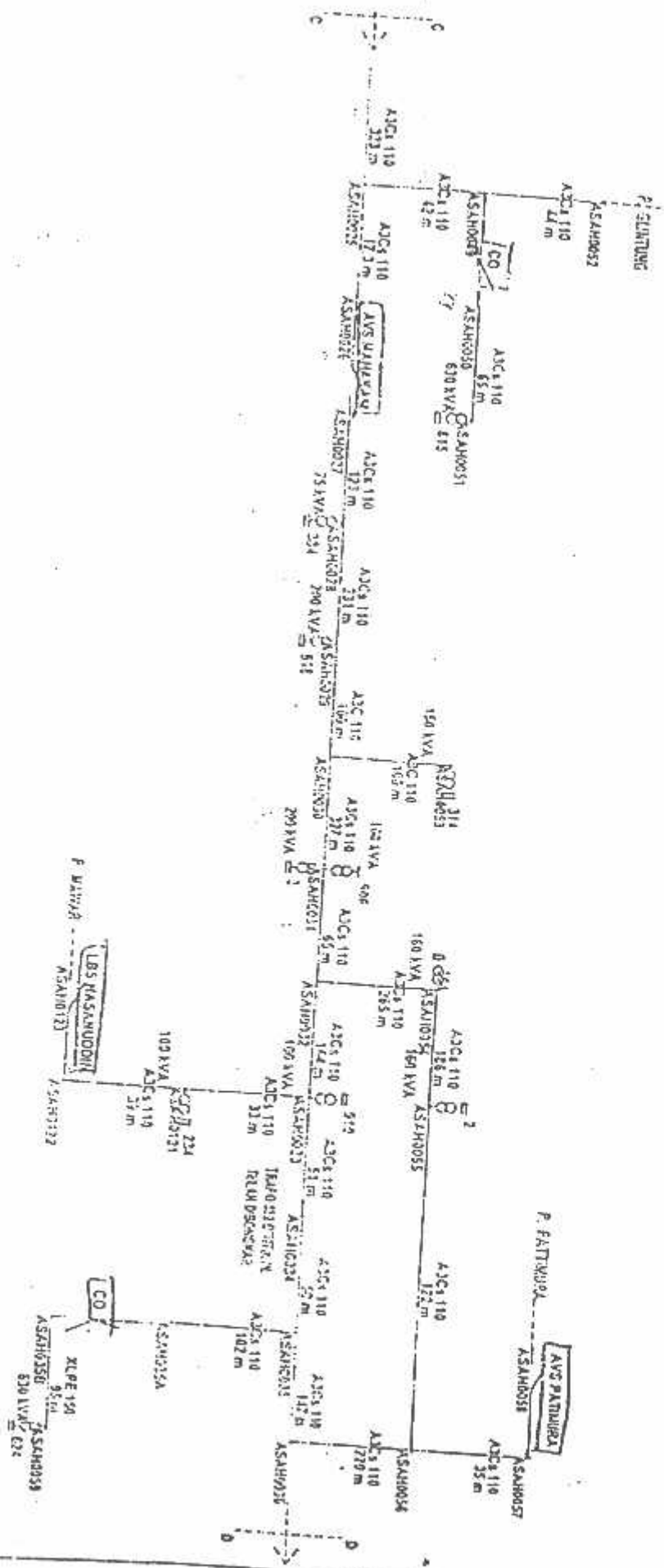
**PT. PLN (PERSERO)
JASA ENJINIRING**

Tipe: **PERALIHAN SUSUT TAWA**
PT. PLN (PERSERO) DISTRIK JATIM - CABANG MALANG

Desain Oleh: **EVILANUS ASRIWIH**
 Drawing No.:

Disetujui Oleh: **4102 3119 1 5107 ALVERO 013010**
 Tanggal: / /

Date: / /

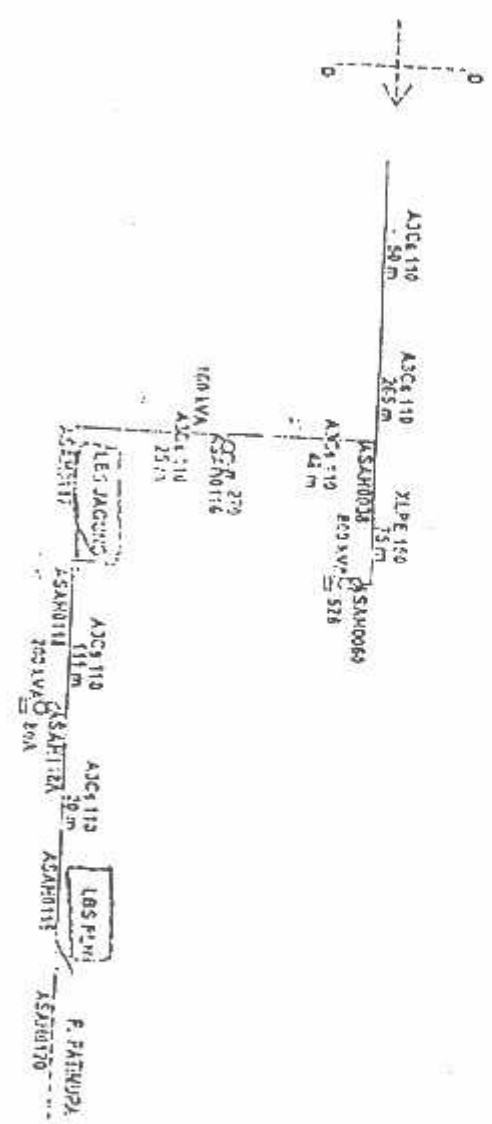


**PT. PLN (PERSERO)
JASA ENJINIRING**

PT. PERUM LITSI SURUT BWA
PT. PERUM TELKOM (TIGAS) DAN PT. CENDIC MALAYSIA

SINGLE LINE: DAKRANIZ PERUM LITSI SURUT BWA

Design By: _____
 Checked By: _____
 Approved By: _____
 Date: _____
 No. Dokumen: _____
 No. Revisi: _____
 No. Gambar: _____

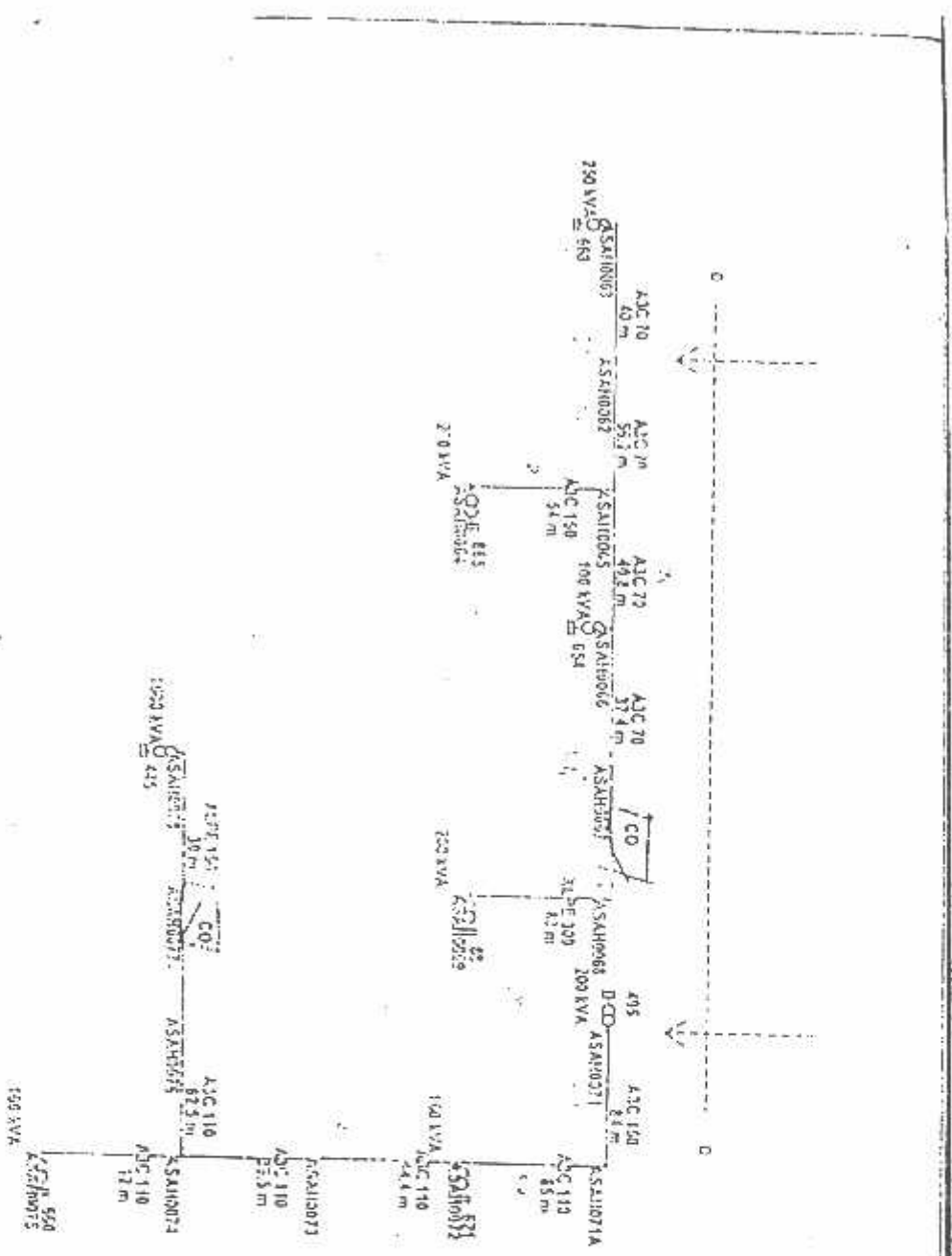


**PT. PLN (PERSERO)
JASA ENJINIRING**

**PERURUKAN SUSUT RUMAH
PT. PUI (PENCEROP) DISTRIBUTUSI JATIM - CIRANG MUYANG**

SINGLE LINE DIAGRAM PERIKULANG KAWALI

Desain By :
 Approved By :
 41073106131105ASAPM14013040

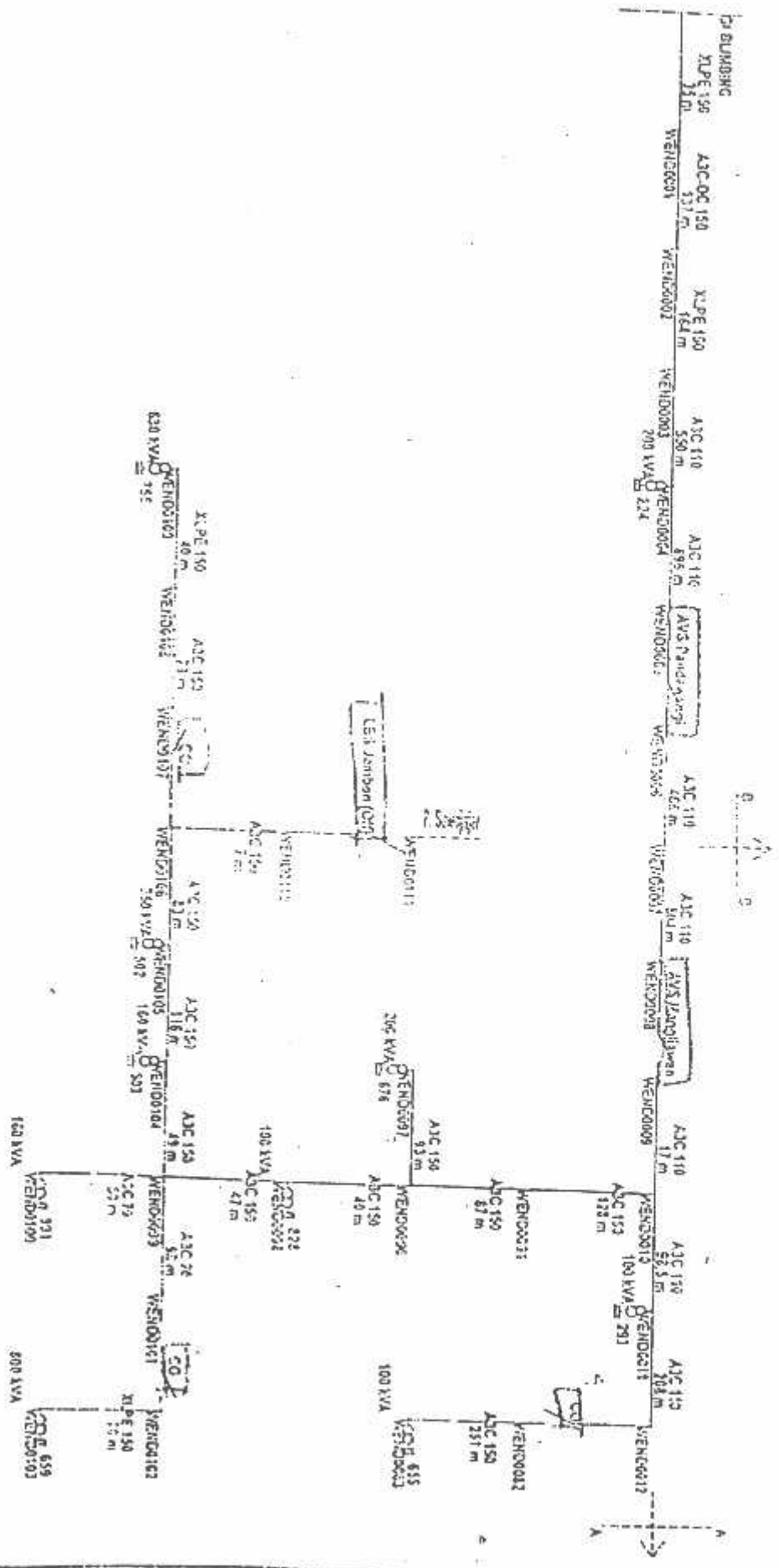



PT. PLN (PERSERO)
JASA ENJINIRING

Project Name: **PENURUNAN SUSUT PMH**
PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG

Title: **SINGLE LINE DIAGRAM PERYUANGAN ASAYAH**
 Drawn by: **Dharmanto**

Approved By: **182.3107.1.51.05 ASYARUD DUL**
 Checked by: **0**
 Date: **0**

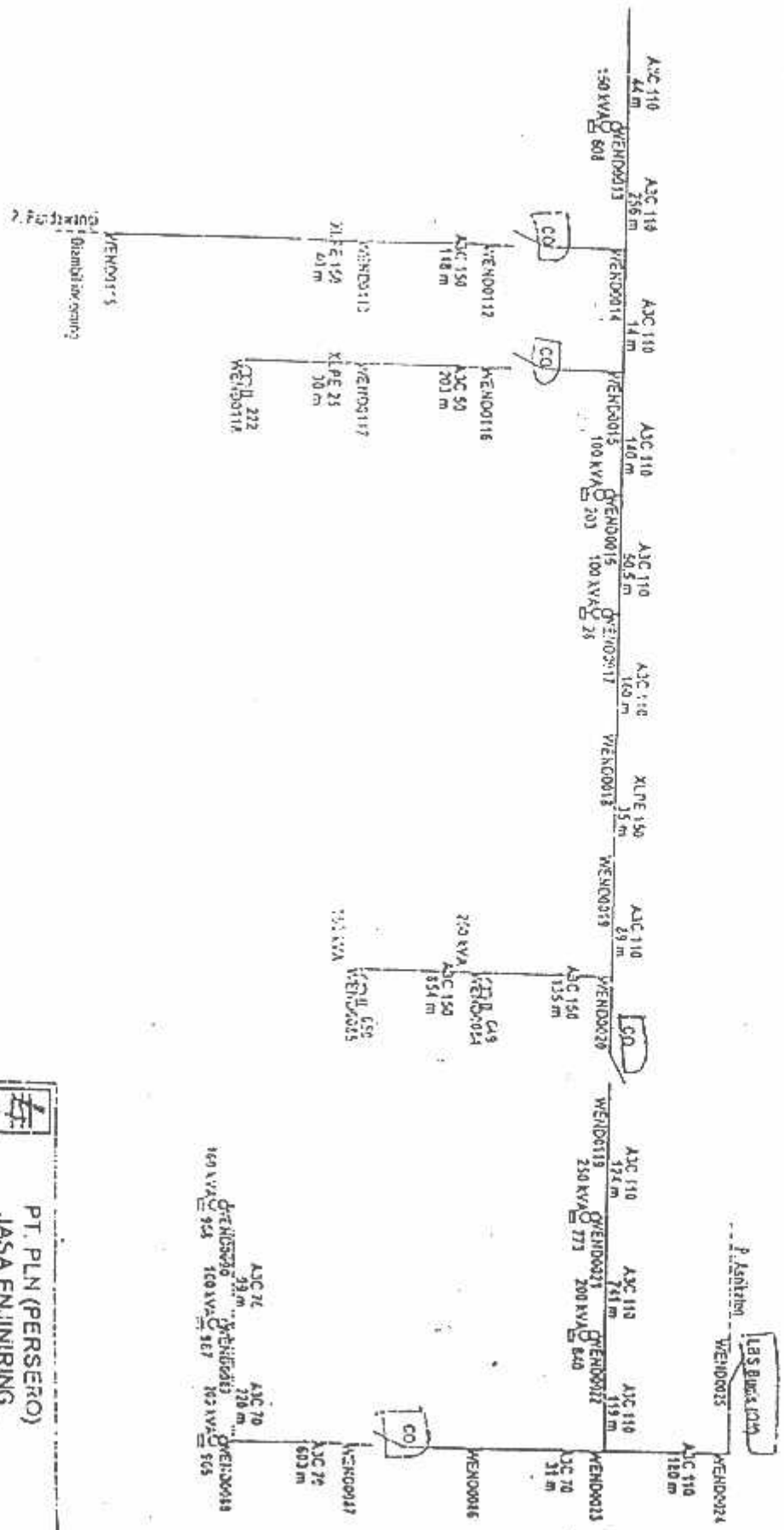




PT. PLN (PERSERO)
JASA ENJINIRING

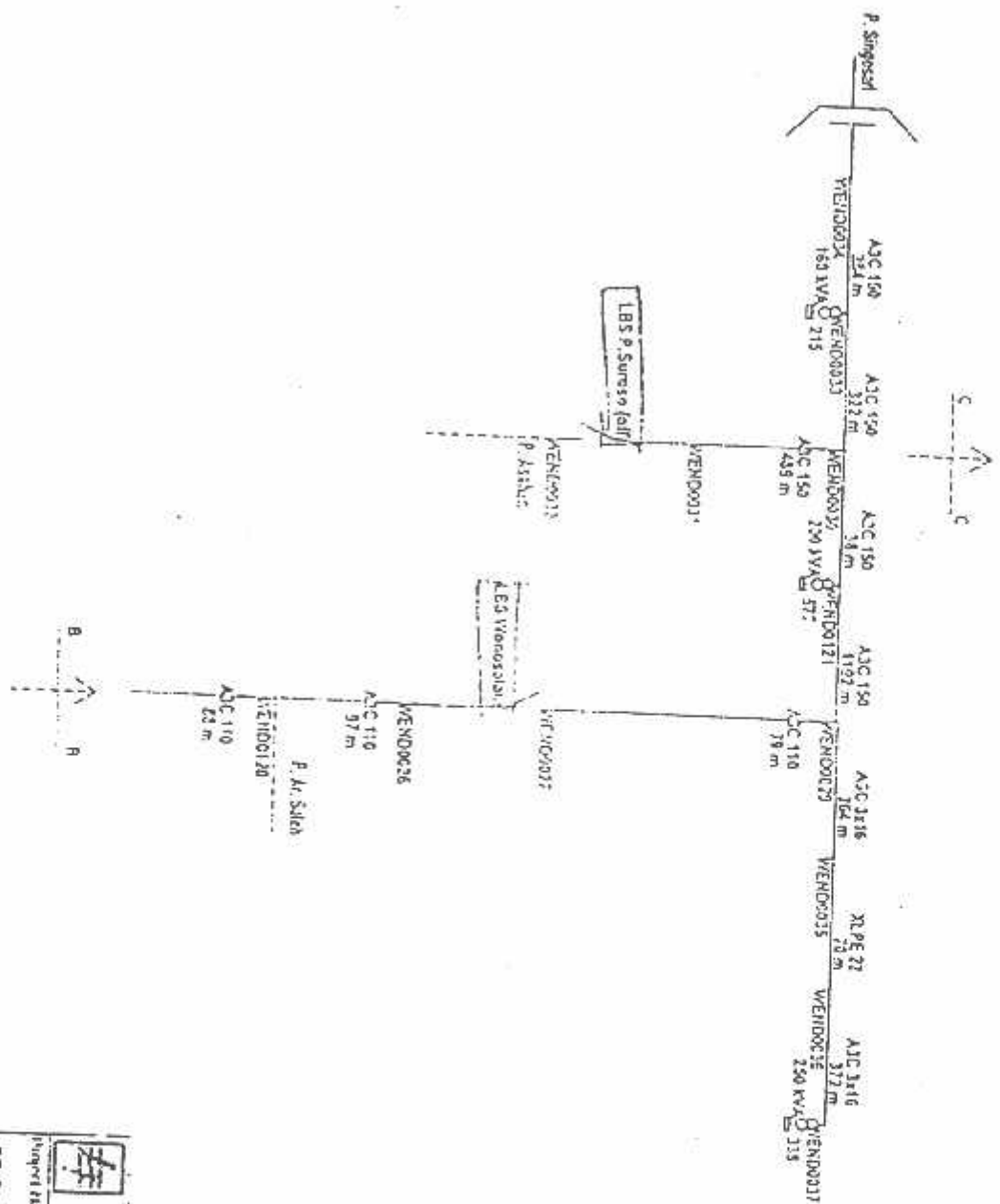
Project Name: PEMERINTAH KABUPATEN
PT. PLN (PERSERO) DISTRIKUSI JATIM - CADANG MALANG
Location: KABUPATEN MALANG
Address: JALAN ...
Approved By: ...
Checked By: ...

Rev	0				
-----	---	--	--	--	--



PT. PLN (PERSERO)
JASA ENJINIRING
PT. PLN (PERSERO) DISTRIBUTI JATIM - CABANG MALANG

Project Name	PT. PLN (PERSERO) DISTRIBUTI JATIM - CABANG MALANG
Client	PT. PLN (PERSERO) DISTRIBUTI JATIM - CABANG MALANG
Approved By	
Checked By	
Date	Nov 0 2000



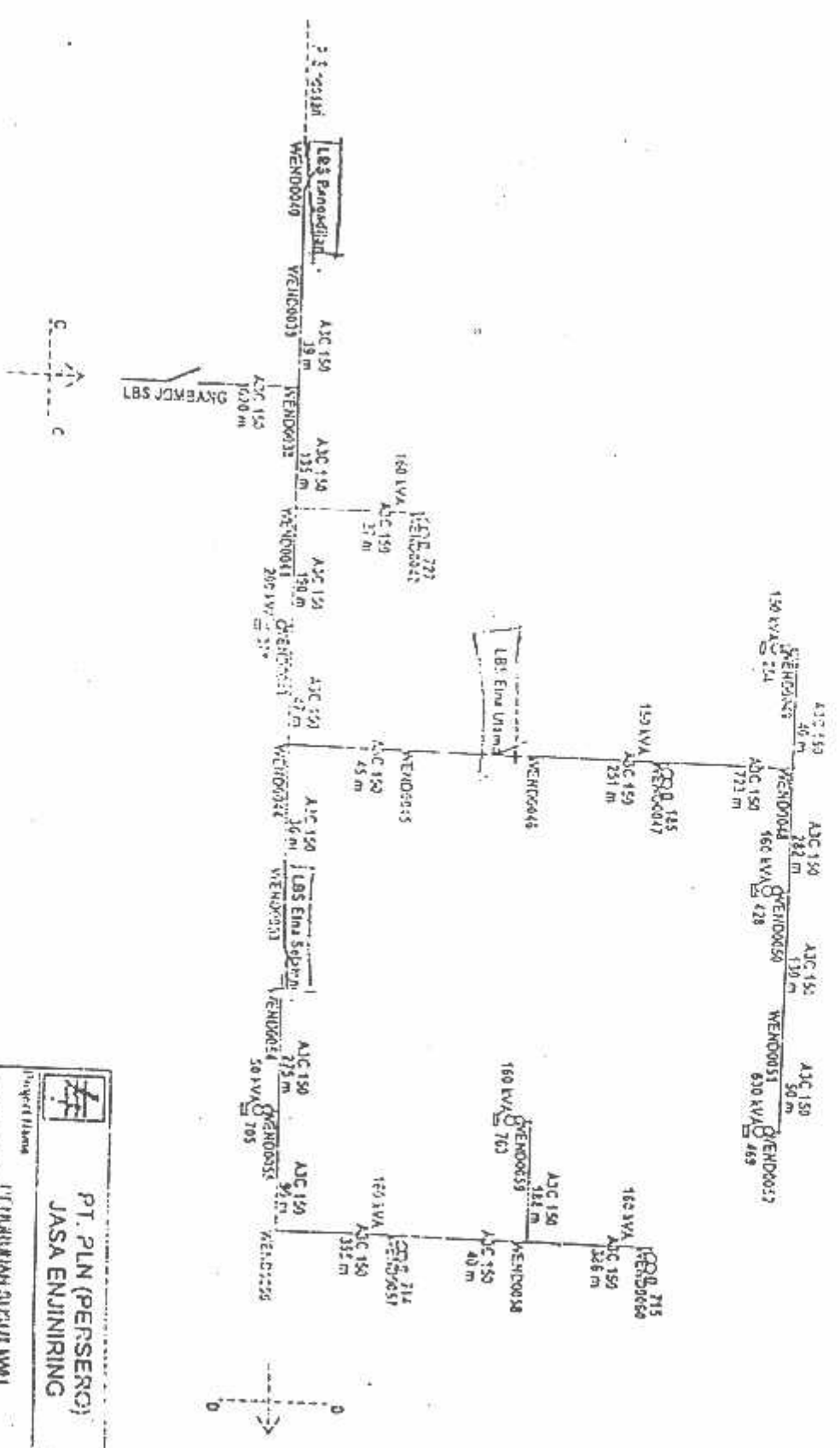
PT. PLN (PERSERO)
JASA ENJINIRING

Project Name
PETUNJAH SUSUT AWI
PT. PULI (PERSERO) DISTRIKUSI JATIM - CABANG MALANG

Approved By
 [Signature]

Checked By
 [Signature]

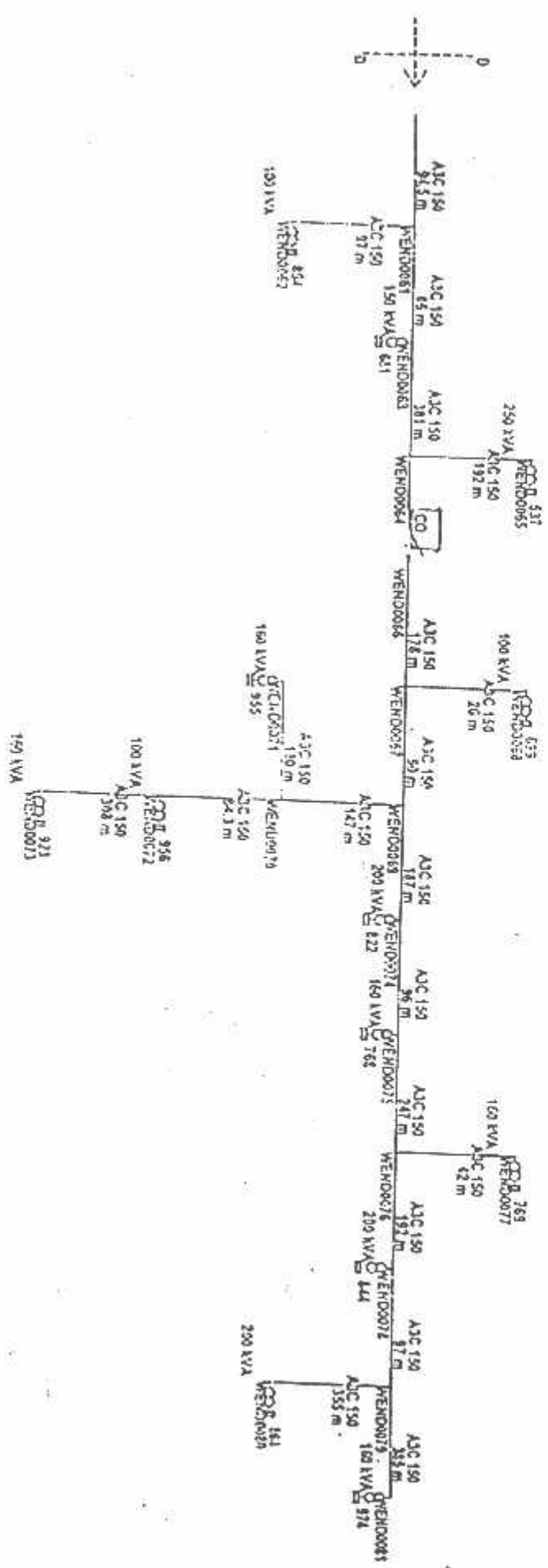
Date
 14/01/2010



**PT. PLN (PERSERO)
JASA ENJINIRING**

PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG

Pekerjaan PT. PLN (PERSERO) DISTRIBUSI JATIM - CABANG MALANG	
Title SURvei ELEMEN PASOKAN PERSALURAN AIRAS VERTIKAL	
Disapproved By (Name) (Date)	Approved By (Name) (Date)
No. 01	No. 01



**PT. PLN (PERSERO)
JASA ENJINIRING**

Project Name: **PEMBUATAN SUBSTANSI
PT. PLN (PERSERO) DISTRIBUTUSI JATIM - CABANG MALANG**

Desain By	Disain By
Approved By	4127.3107.L.51.02.WEND.012.02.0
Checked By	0
Date	

Enhanced genetic algorithm-based fuzzy multi-objective approach to distribution network reconfiguration

Y.-C. Huang

Abstract: An enhanced genetic algorithm (EGA)-based fuzzy multi-objective approach to solve a network reconfiguration problem in a radial distribution system is presented. Maximising the fuzzy satisfaction allows the operator to simultaneously consider the multiple objectives of the network reconfiguration to minimise power loss, violation of voltage and current constraints, as well as switching number, while subject to a radial network structure in which all loads must be energised. The optimisation technique of the EGA is then adopted to solve the fuzzy multi-objective problem efficiently. Test results verify the feasibility of applying the proposed method to manipulate the combinatorial optimisation network reconfiguration in distribution systems.

1 Introduction

Network reconfiguration modifies the network structure of distribution feeders by changing the open/close status of the sectionalising (normally closed) and tie (normally open) switches. This not only reduces the power loss, but also relieves the overloading of the network components. Hence, network reconfiguration is a vital task of distribution automation.

Reconfiguration has received considerable interest in recent years [1–10]. Related methods can be classified into mathematical and heuristic methods. Aoki *et al.* [1] developed a mathematical programming approach to solve optimal load allocation problems in distribution systems. Civanlar *et al.* [2] presented a computationally attractive solution procedure to reduce power loss through reconfiguration. Liu *et al.* [3] proposed two algorithms to minimise losses in distribution networks. Shirmohammadi *et al.* [4] applied optimal load flow analysis to network reconfiguration for loss minimisation. While formulating the loss reduction and load balancing as an integer programming problem, Baran *et al.* [5] proposed efficient load flow equations to calculate the power flow. In addition, related investigations [4, 6, 7] have attempted to reduce the search space by proposing heuristic methods to obtain solutions through intuitive rules based on the operator's experiences.

The reconfiguration is a complicated combinatorial, nondifferentiable and constrained optimisation problem owing to the enormous number of candidates switching combinations in distribution systems. Heuristics and experts' experience-based approaches can only obtain suboptimal solutions. To obtain global optimal or, at least, near global optimal solution, Chiang *et al.* [8, 9] proposed a two-stage solution methodology using the simulated

annealing (SA) algorithm for the reconfiguration. Although SA has the potential to search the global optimal solution, tuning the control parameters of the annealing schedule is extremely difficult and requires a significant amount of computing effort. Simple genetic algorithm (SGA) [10] and hybrid genetic algorithm [11] approaches were presented to solve the reconfiguration, but they focus only on the loss minimum objective. The evolutionary programming [12] was proposed to solve the constrained multi-objective problem.

This study presents a fuzzy multi-objective problem formulation, not only to satisfy realistic objectives and soft constraints (not hard limits) of the reconfiguration, but also to supply rigorous mathematical and heuristic approaches within the problem-solving process. The fuzzy set theory provides an excellent framework for integrating the mathematical and heuristic approach into a more realistic formulation of the reconfiguration, while keeping an efficient computation. Owing to the non-commensurable characteristics of the objectives, a conventional approach that optimises a single objective function is inappropriate for this problem. The fuzzy decision [13] is therefore adopted to simultaneously consider the multiple objectives and to obtain a fuzzy satisfaction maximising decision. The EGA is considered to be an efficient method for solving the large-scale combinatorial optimisation problem, due to the ability to search global or near global optimal solutions and its appropriateness for parallel computing.

2 Problem formulation

This paper considers the following objectives in the reconfiguration:

- (1) Minimisation of the system's power loss,
- (2) Minimisation of the voltage constraint violation,
- (3) Minimisation of the current constraints violation,
- (4) Minimisation of the switching number,

subject to the radial network structure in which all loads must be energised.

© IEE, 2002

IEE Proceedings online no. 20020512

DOI: 10.1049/ip-gsd:20020512

Paper first received 7th November 2000 and in revised form 11th April 2002

The author is with the Department of Electrical Engineering, Cheng Shiu Institute of Technology, 840, Cheng-Ching Road, Notsu-Song 833, Kaohsiung, Taiwan, Republic of China

2.1 Fuzzy multi-objective formulation

The four objectives described in the preceding text are first fuzzified and, then, dealt with by integrating them into a fuzzy satisfaction objective function M through appropriate weighting factors as shown below:

$$\text{Max } M = w_1 \mu_p + w_2 \sum_{i=1}^{N_{bs}} \mu_{V_i} + w_3 \sum_{i=1}^{N_{br}} \mu_{I_i} + w_4 \mu_S \quad (1)$$

where μ_p represents the membership value (MV) to minimise the total power loss; $\sum_{i=1}^{N_{bs}} \mu_{V_i}$ is the sum of MVs to minimise the voltage violation; $\sum_{i=1}^{N_{br}} \mu_{I_i}$ is the sum of MVs to minimise the current violation; μ_S is the MV of minimisation of the switching number; N_{bs} is the number of busbars; N_{br} is the number of branches, and w_i , $i = 1, 2, 3, 4$, are weighting factors. The membership functions (MFs) used to describe the four objectives are stated in the following.

2.2 MF for the power loss

Reconfiguration attempts to obtain the minimum power loss, while no violation of the constraint exists. A configuration with a larger power loss is given a lower MV. Because the exponential function meets this condition, it is selected for the MF of the power loss, as shown in Fig. 1, and is expressed as follows:

$$\mu_p = \exp(-(P_{loss} - P_{Lmin})/P_{Lmin}) \quad (2)$$

where μ_p denotes the MV of power loss P_{loss} , and the value of power loss P_{Lmin} is the minimum power loss which can be obtained by minimising the single objective of the power loss.

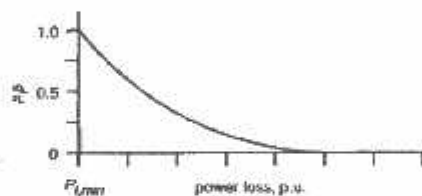


Fig. 1 Membership function of the power loss

2.3 MF for the busbar voltage

Selecting the trapezoid MF prevents undervoltage and overvoltage in each busbar, as described in the following:

$$\mu_{V_i} = \begin{cases} 0, & V < V_1 \\ \frac{V - V_1}{V_2 - V_1}, & V_1 \leq V < V_2 \\ 1.0, & V_2 \leq V < V_3 \\ \frac{V_4 - V}{V_4 - V_3}, & V_3 \leq V < V_4 \\ 0, & V \geq V_4 \end{cases} \quad (3)$$

Fig. 2 presents the MF for the voltage, where μ_{V_i} represents the MV of voltage V_i in busbar i . V_1 , V_2 , V_3 and V_4 are the specified voltage values.

2.4 MF for the branch current

The MF of the branch current is described as follows.

$$\mu_{I_i} = \begin{cases} 1.0, & I < I_1 \\ \frac{I_2 - I}{I_2 - I_1}, & I_1 \leq I < I_2 \\ 0, & I \geq I_2 \end{cases} \quad (4)$$

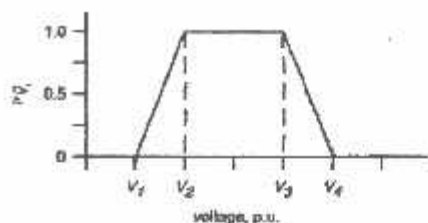


Fig. 2 Membership function of the busbar voltage

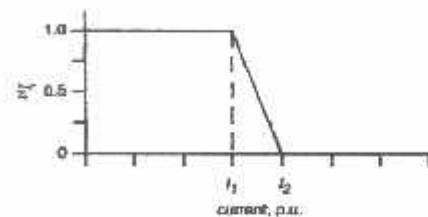


Fig. 3 Membership function of the branch current

Fig. 3 depicts the MF for the branch current, where μ_{I_i} represents the MV of current I_i flow in branch i . I_1 and I_2 are specified current values. The average current loading index (CLI_{ave}) and maximum current loading index (CLI_{max}) evaluate the current loading of the network configuration, as described in the following:

$$CLI_{ave} = \frac{1}{N} \sum_{i=1}^{N_{br}} \frac{I_{f,i}}{I_{c,i}} \quad (5)$$

$$CLI_{max} = \text{Max} \left\{ \frac{I_{f,i}}{I_{c,i}}, i = 1, 2, \dots, N_{br} \right\} \quad (6)$$

where $I_{f,i}$ denotes the current in the branch i , $I_{c,i}$ is the line capacity of the branch i , and N_{br} is the number of branches. Obviously, the minimum current loading index is equal to zero, indicating no current flow through this branch (i.e. tie line). The lower the CLI_{ave} implies a higher degree of load balance. The branch current $I_{f,i}$ can be calculated by the simplified power flow (11), as will be described later.

2.5 MF for the switching number

To reduce the operation, as well as extend the life of switches, the switching number is an essential consideration in the reconfiguration and should be kept as minimal as possible. A configuration with a larger switch number is assigned a lower MV. Because the exponential function satisfies this condition, it is used for the MF of the switching number, as shown in Fig. 4, and is written as follows:

$$\mu_S = \begin{cases} 1.0, & N_S \leq N_{min} \\ \exp(-(N_S - N_{min})/N_{min}), & N_{min} < N_S \end{cases} \quad (7)$$

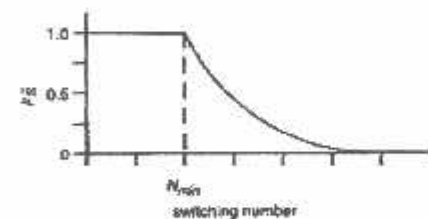


Fig. 4 Membership function of the switching number

where $\mu_{\bar{s}}$ denotes the MV of switching number N_s , and N_{min} is the preselected switching number limit.

2.6 Simplified power flow equations

Consider the radial network in Fig. 5. The line impedances between branch i and branch $i+1$ are represented by $z_i = r_i + jx_i$, and load models are expressed as constant power sinks, $S_L = P_L + jQ_L$. By assuming that the losses on the lines between branches are significantly smaller than branch power terms, the following set of simplified power flow equations is used [5]:

$$P_{i+1} = P_i - P_{L,i+1} = \sum_{k=i+2}^{N_{br}} P_{L,k} \quad (8)$$

$$Q_{i+1} = Q_i - Q_{L,i+1} = \sum_{k=i+2}^{N_{br}} Q_{L,k} \quad (9)$$

$$V_{i+1}^2 = V_i^2 - 2(r_i P_i + x_i Q_i) \quad (10)$$

where the real power P_{i+1} , reactive power Q_{i+1} , and busbar voltage V_{i+1} are the values at branch $i+1$, after determining the values of P_i , Q_i and V_i for branches $0, 1, \dots, i$.

The P_i , Q_i and V_i can be used to calculate the branch current $I_{f,i}$:

$$I_{f,i} = \sqrt{(P_i^2 + Q_i^2)/V_i^2} \quad (11)$$

The total power loss on a distribution system can be calculated by summing up each branch power loss through the following equation:

$$P_{loss} = \sum_{i=1}^{N_{br}} r_i (P_i^2 + Q_i^2)/V_i^2 \quad (12)$$

The multi-objective formulation in (1) is maximised using the EGA based approach. The following section briefly introduces the EGA.

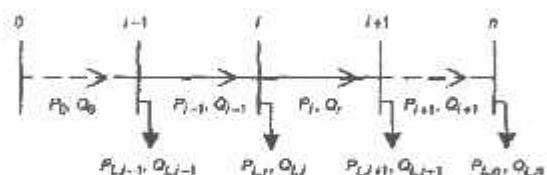


Fig. 5 One line diagram of a radial distribution system

3 The proposed EGA

The SGA [10, 14, 15] is a stochastic search mechanism based on the Darwinian principle of natural evolution. The evolutionary process causes initial solutions to move towards the optimal point over many generations. The roulette-wheel selection scheme used in SGA has the potential drawback that the best member of the population may fail to produce offspring in the next generation, possibly causing a stochastic error. The elitism and multiple-point crossover strategies [16] is embedded in the EGA to enhance the performance of the SGA. The EGA operations are described as follows.

3.1 Coding and decoding functions

A sequence of binary bits is called a string or an individual in the EGA, which corresponds to a set of genes [14, 15], i.e. a chromosome, in natural genetics. A coding function translates the decision variables into the binary strings, and a decoding function translates the binary strings back into

the decision variables. The EGA conducts its operations according to these string representations, then a fitness function evaluates the fitness of the strings.

3.2 Initialisation

An initial population of strings is randomly generated from the initial conditions of the studied system. This scheme is intended to provide an extensive basis from which the EGA begins its search. Parameter settings are also given in the initialisation stage.

3.3 Fitness function

The individuals evolve according to their fitness to the environment. Fitness is defined as a non-negative figure of merit associated with a set of decision variables. For a maximisation problem, the larger the objective function value the individual implies a larger fitness.

3.4 Selection and elitism

The criterion to select the individuals is based on their fitness values [14, 15]. Individuals with a larger fitness value, i.e. better solutions to the problem, receive correspondingly larger numbers of copies in the mating pool. For example, assume there are m individuals within the population, each having a fitness value f_i for $i=1, 2, \dots, m$. Then, in the mating pool, the number of individuals (n_i) with fitness f_i is proportional to the value calculated by

$$n_i = f_i / \left(\sum_{j=1}^m f_j \right) \quad (13)$$

Elitism is an effective means of saving early solutions by ensuring the survival of the fittest strings in each generation [16]. The elitism puts the best string of the old generation into the new generation to improve convergence performance of the SGA.

3.5 Multiple-point crossover

Individuals in the mating pool are randomly taken in pairs, without replacement, after the pool has been constructed. The crossover operation used in the SGA [14, 15] is a single-point crossover, because one crossover site is selected. This simple crossover can be generalised to multiple-point crossover in which the number of crossover points N_c is specified. These two strings are then divided into segments by randomly setting N_c crossover sites. The segments of strings from the parents are exchanged with each other to generate their offspring. Multiple-point crossover can solve certain problems of feature combinations encoded on chromosomes that a single-point crossover cannot solve [16].

3.6 Mutation

The mutation diversifies the search and prevents the premature convergence that leads to nearly the same individuals within a population after several generations [14–16]. For each bit of the offspring, the mutation changes a 1 into a 0, or vice versa, according to a fixed mutation probability. The mutation probability must be sufficiently small to ensure not only that the crossover is the primary means of creating new offspring, but also that the EGA is not reduced to a random search. However, too small a mutation probability cannot avoid premature convergence.

The crossover and mutation are repeated until the mating pool is empty. Once the new generation is obtained, the evolving process repeats the genetic operations of selection, crossover and mutation, to generate the new individuals. The process stops when the average fitness is within a

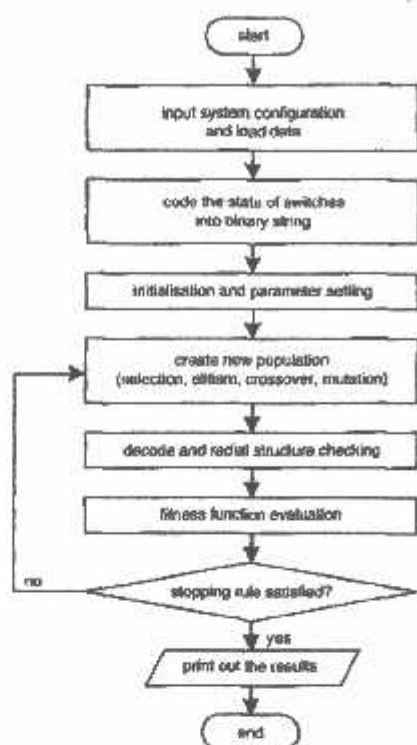


Fig. 6 Flowchart of the reconfiguration solved by the EGA

specified percentage of the best string fitness, or when the given count of total generations is reached.

4 Implementation of the proposed EGA

Fig. 6 displays the flowchart of the reconfiguration solved by the EGA. The EGA for the reconfiguration is implemented straightforwardly, except for handling the radial structure constraint. Detailed implementation is stated as follows.

4.1 Coding, decoding and radial structure checking

The binary decision variables denote the on/off status of switches to be determined in the reconfiguration. Where '0' denotes 'open' of the switch and '1' represents 'close'. The fitness value of each individual is evaluated by decoding the 1/0 status of switches to obtain the network configuration. The radial structure checking examines whether the

configuration is radial. If the network is not radial, a switch of each loop is randomly opened to satisfy the radial structure constraint.

4.2 Fitness function, selection and elitism

The fitness function f of the reconfiguration is defined as follows:

$$f = M = w_1 \mu_P + w_2 \sum_{i=1}^{N_b} \mu_{V_i} + w_3 \sum_{i=1}^{N_{br}} \mu_{I_i} + w_4 \mu_S \quad (14)$$

The fitness value of individuals is estimated from the fuzzy values of the objectives considered (14). The larger the fitness of an individual implies a larger number of copies that are selected in the mating pool. Moreover, the elitism puts the best individual of the current generation into the new generation to further enhance the convergence performance of the SGA.

4.3 Multiple-point crossover and mutation

The crossover recombines an individual with another individual split at the same crossover sites based on the preselected crossover probability. The mutation randomly alternates a bit in the individual according to the mutation probability. The EGA optimisation process repeats until the specified maximum number of generations is reached.

5 Test results

The proposed EGA-based method was developed on a PC Pentium III-600 using Turbo C programming language. The proposed method was tested on a typical distribution system to evaluate its effectiveness.

5.1 Case study

The tested case is a 70-busbar and 74-branch radial distribution system [17], as shown in Fig. 7. The tie lines of the test system are 70, 71, 72, 73 and 74, which are opened in normal configuration. Table 1 summarises their impedance data. The system voltage is 12.66 kV, and kVA base was assumed at 1000 kVA. The total system loads are 3802.19 kW and 2694.6 kVAR. The lower busbar voltage limit is 0.9 p.u. The additional branch current constraints were set at 200 A, except for the upstream branches, 1–9, whose limits were 400 A, and 47–50 as well as 53–65 branches were 300 A.

The MFs were determined first. Through the single minimisation of power loss, the minimum power loss P_{Loss} is 0.094023 p.u. (94.023 kW). The MF of the voltage used 0.9 p.u. for V_1 , 0.95 for V_2 , 1.0 for V_3 and 1.05 for V_4 (Fig. 2). For the branch current, I_1 was set at its 90% line capacity and I_2 at its specified capacity limit (Fig. 3). The reasonable switching number was set at 4 for N_{switch} .

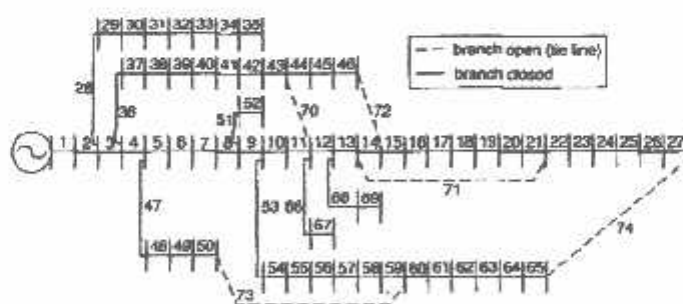


Fig. 7 Diagram of the test system

Table 1: Tie line data

Line number	Resistance, Ω	Reactance, Ω
70, 71	0.5	0.5
72, 74	1.0	1.0
73	2.0	2.0

Table 2: Optimal feeder reconfiguration

	Branch open	Switching number
Original configuration	70, 71, 72, 73, 74	—
After reconfiguration	13, 59, 70, 71, 74	4

Table 3: System state before/after network reconfiguration

	Power loss, kW	Voltage, p.u.	Current loading index
Original configuration	204.799	$V_{max}=1.0$ $V_{min}=0.8131$	$CLI_{max}=0.2201$ $CLI_{ave}=0.3203$
After reconfiguration	113.406	$V_{max}=1.0$ $V_{min}=0.9268$	$CLI_{max}=0.1980$ $CLI_{ave}=0.9203$

The proper weighting factors used are $w_1 = w_2 = w_3 = w_4 = 0.25$, in which these four objectives are assumed to be equally important. The weighting factors can be varied according to the preferences of different operators. The test results of the EGA are compared with the SGA [10] and the SA [8, 9]. The optimal configurations obtained by these three methods are all the same, as shown in Table 2. The case studied requires four switching numbers to achieve the optimal configuration.

Table 3 lists the power loss as well as the maximum and minimum voltages of the system before and after reconfiguration. The power loss before reconfiguration is 204.799 kW (5.39%) and that for the optimal configuration is 113.406 kW (2.98%). This finding indicates that 91.393 kW (2.41%) further reduction in power loss (or 44.71% improvement) is obtained. Also, the minimum voltage is upgraded. The CLI_{max} after reconfiguration is the same as that before reconfiguration, in which both occur at branch 1, i.e. is nearest the power source (Fig. 7). However, the CLI_{ave} after reconfiguration is smaller than that before reconfiguration, which arises from the reconfiguration transforming the original network structure into the optimal structure. Therefore, the proposed EGA can achieve the optimal load-balanced structure.

Fig. 8 presents the voltage profile of the system before and after reconfiguration. After reconfiguration, the voltage profile is obviously improved. Fig. 9 displays the current profile of the system before and after reconfiguration. Although the reconfiguration alters the current distribution, the more the load balancing is achieved (i.e. smaller CLI_{ave} obtained). Hence, optimal reconfiguration can improve the voltage profile and also provide a more load-balanced situation for distribution network operation by the proposed EGA.

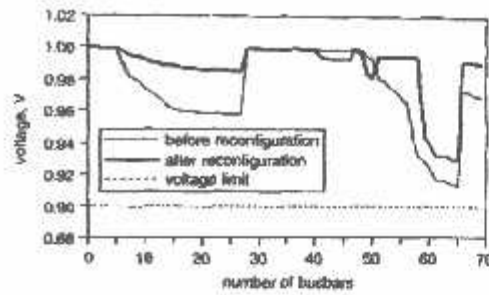


Fig. 8: System voltage profile

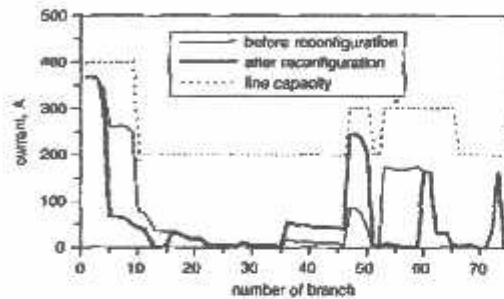


Fig. 9: System branch current

Table 4: Result comparisons of the EGA, SGA and SA

Methods	Average computing time, s	Number of global optimal reached in 100 runs
EGA	13	97
SGA [10]	11	91
SA [8, 9]	36	95

5.2 Performance comparisons of EGA, SGA and SA

Parameter settings of the EGA, as compared with the annealing schedule of the SA, can be more easily tuned to yield promising results [18] after some experiments. The optimal parameter settings of the EGA are as follows: population size of 60, maximum generation of 20, number of crossover sites (N_c) equal to 3, crossover probability equal to 0.6 and mutation probability equal to 0.001.

The parameters of SGA are set the same as EGA listed in the preceding text for comparison, except for single-point crossover operation used. The numbers of trials evaluated for the EGA and the SGA were 1200 (20×60) and that for the SA (after some experiments) was 12000 (120×100), i.e. the maximum iteration number in the SA was set at 120 and the trial number per iteration was at 100. Table 4 displays the result comparisons of the EGA, SGA and SA. The average computing time of the EGA is close to that of SGA and far less than that of SA. The number of the global optimum reached in 100 runs with different random initial solutions of the EGA is superior to that of the SGA and the SA. Although the SA has the potential to obtain the global optimal solution, the features of its searching sequentially from a single point to another and possibly cycling around a local optimal solution diminish its performance in

practice. The proposed EGA is a highly effective approach for the reconfiguration, owing to the high performance enhanced by the elitism and multiple-point crossover used in the EGA and inherent implicit parallelism of the SGA.

6 Conclusions

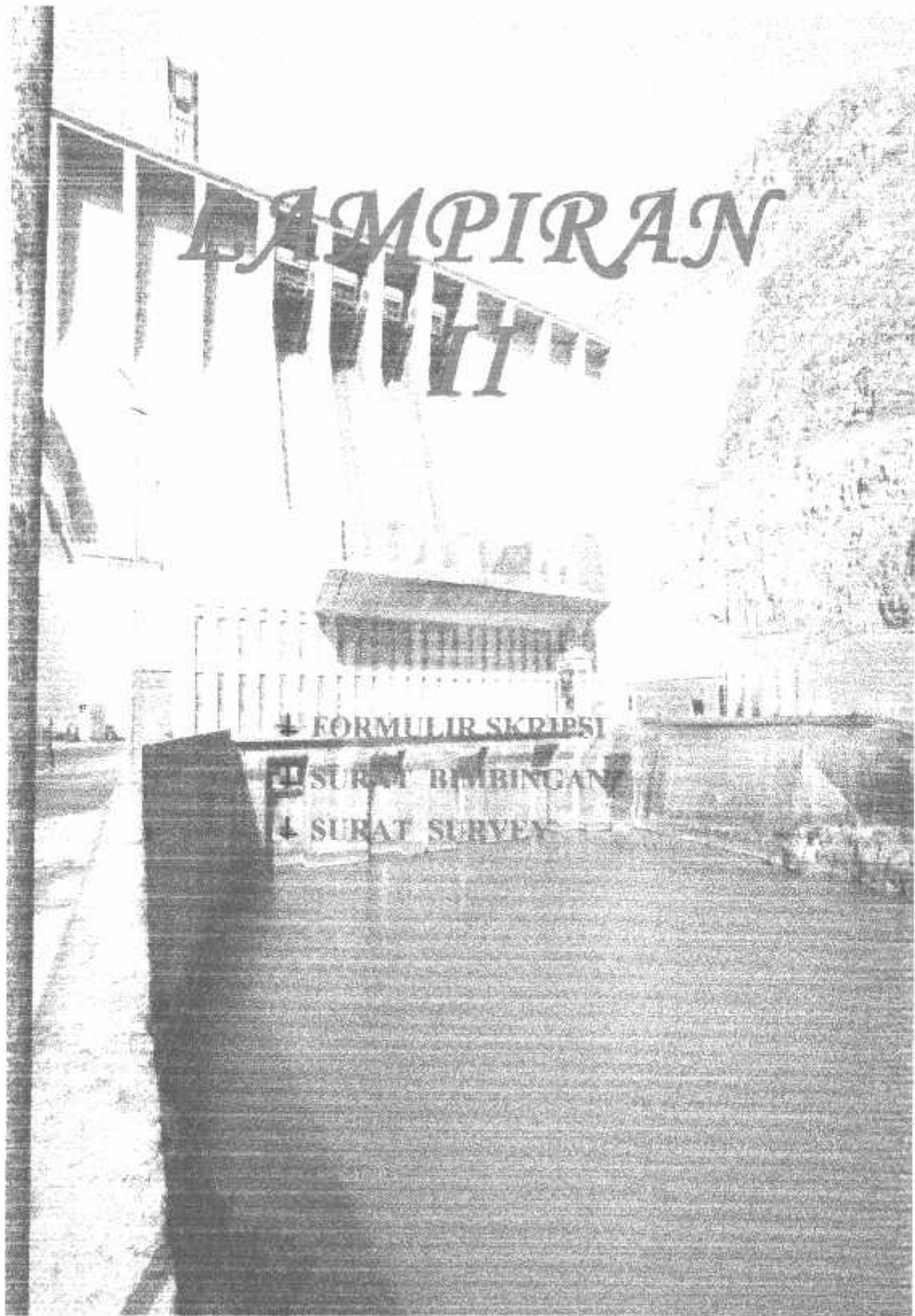
This study has presented an EGA-based fuzzy multi-objective approach to solve the network reconfiguration problem in a radial distribution system. The objectives considered attempt to maximise the fuzzy satisfaction of the minimisation of power loss, violation of voltage and current constraints, and the switching number is subject to radial network structure in which all loads must be energised. Owing to the high performance and implicit parallelism of the proposed method, the EGA can solve the fuzzy multi-objective problem efficiently. Test results confirm that the EGA-based approach can efficiently search the optimal or near-optimal network configuration. Results in this study further demonstrate the feasibility of applying the evolutionary optimisation algorithm to the network reconfiguration in actual distribution systems.

7 Acknowledgments

The author is greatly indebted to the National Science Council, Taiwan, Republic of China, for financial support under grant number NSC-89-2213-E-230-009.

8 References

- 1 AOKI, K., ICHIMORI, T., and KANEZASHI, M.: 'Normal state optimal load allocation in distribution systems', *IEEE Trans. Power Deliv.*, 1987, 2, (1), pp. 147-155
- 2 CIVANLAR, S., GRAINGER, J.J., YIN, H., and LEE, S.H.: 'Distribution feeder reconfiguration for loss reduction', *IEEE Trans. Power Deliv.*, 1988, 3, (3), pp. 1217-1223
- 3 LIU, C.C., LEE, S.J., and VU, K.: 'Loss minimization of distribution feeders: optimality and algorithms', *IEEE Trans. Power Deliv.*, 1989, 4, (2), pp. 1281-1289
- 4 SHIRMOHAMMADI, D., and HONG, H.W.: 'Reconfiguration of electric distribution networks for resistive line losses reduction', *IEEE Trans. Power Deliv.*, 1989, 4, (2), pp. 1492-1498
- 5 BARAN, M.E., and WU, F.F.: 'Network reconfiguration in distribution systems for loss reduction and load balancing', *IEEE Trans. Power Deliv.*, 1983, 4, (2), pp. 1401-1407
- 6 WU, J.S., TOMSOVIC, K.L., and CHEN, C.S.: 'A heuristic search approach to feeder switching operations for overload, faults, unbalanced flow and maintenance', *IEEE Trans. Power Deliv.*, 1991, 6, (4), pp. 1579-1585
- 7 TAYLOR, T., and LUBKENMAN, D.: 'Implementation of heuristic search strategies for distribution feeder reconfiguration', *IEEE Trans. Power Deliv.*, 1990, 5, (1), pp. 239-246
- 8 CHIANG, H.D., and JEAN-JAMEAU, R.M.: 'Optimal network reconfigurations in distribution systems, part 1: a new formulation and a solution methodology', *IEEE Trans. Power Deliv.*, 1990, 5, (4), pp. 1902-1909
- 9 CHIANG, H.D., and JEAN-JAMEAU, R.M.: 'Optimal network reconfigurations in distribution systems, part 2: solution algorithms and numerical results', *IEEE Trans. Power Deliv.*, 1990, 5, (3), pp. 1568-1574
- 10 NARA, K., SHIOSE, A., KITAGAWA, M., and ISHIHARA, T.: 'Implementation of genetic algorithm for distribution systems loss minimum re-configuration', *IEEE Trans. Power Syst.*, 1992, 7, (3), pp. 1044-1051
- 11 LIN, W.M., CHENG, F.S., and TSAY, M.T.: 'Distribution feeder reconfiguration with refined genetic algorithm', *IEE Proc., Gener. Transm. Distrib.*, 2000, 147, (6), pp. 340-354
- 12 HSIAO, Y.T., and CHEN, C.Y.: 'Multiobjective optimal feeder reconfiguration', *IEE Proc., Gener. Transm. Distrib.*, 2001, 148, (4), pp. 333-336
- 13 BELLMAN, R.E., and ZADEH, L.A.: 'Decision making in a fuzzy environment', *Manag. Sci.*, 1970, 17, pp. B-141-B-164
- 14 GOLDBERG, D.E.: 'Genetic algorithm in search, optimization and machine learning' (Addison-Wesley, New York, 1989)
- 15 GEN, M., and CHENG, R.: 'Genetic algorithms and engineering design' (John Wiley & Sons, New York, 1997)
- 16 LIN, C.T., and LEE, C.S.: 'Neural fuzzy systems' (Prentice Hall, New Jersey, 1999)
- 17 BARAN, M.E., and WU, F.F.: 'Optimal capacitor placement on radial distribution systems', *IEEE Trans. Power Deliv.*, 1989, 4, (1), pp. 725-734
- 18 DAVIS, L.: 'Genetic algorithms and simulated annealing' (Morgan Kaufmann, London, 1987)





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

**BERITA ACARA UJIAN SKRIPSI
FAKULTAS TEKNOLOGI INDUSTRI**

Nama : ARIK DWI TRISTANTO
NIM : 01.12.065
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik
Judul Skripsi : REKONFIGURASI DENGAN PENDEKATAN
MULTI – OBJEKTIF BERBASIS FUZZY
MENGUNAKAN METODE *ENHANCED GENETIC
ALGORITHM* PADA JARINGAN DISTRIBUSI 20KV
DI G.I. BLIMBING

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

Hari : Selasa
Tanggal : 04 September 2007
Dengan Nilai : 77,45 (B+) *fy*

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.P. 1018800188

Penguji Kedua

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



PERSETUJUAN PERBAIKAN SKRIPSI

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

Hari : Selasa
Tanggal : 04 September 2007

Telah dilakukan perbaikan skripsi oleh :

1. Nama : Arik Dwi Tristanto
2. NIM : 01.12.065
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : Rekonfigurasi Dengan Pendekatan Multi - Objektif Berbasis *Fuzzy* Menggunakan Metode *Enhanced Genetic Algorithm* Pada Jaringan Distribusi 20kV Di G.I. Blimbing.

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Perbaiki kesimpulan	
2.	Kemampuan feeder bagaimana	

Dosen Penguji Pertama

Ir. M. Abdul Hamid, MT.
NIP.P. 1018800188

Dosen Pembimbing

Ir. H. Taufik Hidayat, MT.
NIP.P. 1018700151

Irrine Budi S., ST., MT.
NIP. 132314400



PERSETUJUAN PERBAIKAN SKRIPSI

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

Hari : Selasa
Tanggal : 04 September 2007

Telah dilakukan perbaikan skripsi oleh :

1. Nama : Arik Dwi Trisanto
2. NIM : 01.12.065
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : Rekonfigurasi Dengan Pendekatan Multi - Objektif Berbasis *Fuzzy* Menggunakan Metode *Enhanced Genetic Algorithm* Pada Jaringan Distribusi 20 kV Di G.I. Blimbing.

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Daftar pustaka di betulkan untuk urutan tahun	
2.	Abstrak di betulkan	
3.	Rumusan masalah harus kata tanya	
4.	Tujuan jangan terlalu banyak (sesuaikan dengan buku yang sudah di ubah)	

Dosen Penguji Kedua

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

Dosen Pembimbing

Ir. H. Taufik Hidayat, MT.
NIP.P. 1018700151

Irrine Budi S., ST., MT.
NIP. 132314400



PERMOHONAN PERSETUJUAN SKRIPSI

Yang betanda tangan dibawah ini :

Nama : ARIK DWI TRISTANTO
 NIM : 01.12.085
 Semester : XI
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-1
 Konsentrasi : Teknik Elektronika / Teknik Energi Listrik
 Alamat : Jl. Simpang Sukun NO 32

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 (.....)
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

(Puri Handayani)

Malang, 9 OCTOBER.....2006
Pemohon

(ARIK DWI TRISTANTO)

Disetujui
Ketua Jurusan Teknik Elektro

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

Mengetahui
Dosen Wali

(Ir. Choirul Saich, MT)

Catatan :

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

1. $IPK \frac{321.5}{127} = 2.3$
2.
3. - praktikum pemrograman \rightarrow qlikme
logika \rightarrow ?



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

Konsentrasi : Teknik Energi Listrik / ~~Teknik Elektronika~~

1	Nama Mahasiswa : <u>AFIF DWI PRASANTO</u>	Nim : <u>C 12-065</u>
2	Waktu pengajuan	Tanggal : <u>9</u> Bulan : <u>OCTOBER</u> Tahun : <u>2006</u>
3	Spesifikasi judul (berilah tanda silang)	
	<input checked="" type="checkbox"/> a. Sistem Tenaga Elektrik	<input type="checkbox"/> e. Elektronika & Komponen
	<input type="checkbox"/> b. Energi & Konversi Energi	<input type="checkbox"/> f. Elektronika Digital & Komputer
	<input type="checkbox"/> c. Tegangan Tinggi & Pengukuran	<input type="checkbox"/> g. Elektronika Komunikasi
	<input type="checkbox"/> d. Sistem Kendali Industri	<input type="checkbox"/> h. lainnya
4	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen *) : <u>Ir. Taufiq Hidayat M.T.</u>	Mengetahui, Ketua Jurusan Ir. F. Yudi Impraptono, MT Nip. Y. 1039500274
5	Judul yang diajukan mahasiswa :	<u>EKONOMISASI DENGAN PENYALATAN MULTIELEKTRO BERBASIS FUZZY MENGGUNAKAN METODE ENHANCED GENETIC ALGORITHM DUA SARBINGAN DISTRIBUSI 20KV</u>
6	Perubahan Judul yang disetujui Dosen sesuai materi bidang ilmu
7	Catatan :	Disetujui, <u>14</u> <u>OCTOBER</u> , 200 <u>6</u> . Dosen <u>Ir. Taufiq Hidayat M.T.</u>

Perhatian :

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



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

Malang, 16 Oktober 2006

Lampiran : Satu Lembar
Perihal : **Kesediaan Sebagai
Dosen Pembimbing**

Kepada : Yth. Bapak Ir. H. Taufik Hidayat, MT
Dosen Jurusan Elektro/ T. Energi Listrik
Institut Teknologi Nasional Malang
di
Malang

Yang bertandatangan di bawah ini :

Nama : Arik Dwi Trisanto
Nim : 01.12.065
Semester : XI
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik

Dengan ini mengajukan permohonan, kiranya Bapak bersedia menjadi Dosen Pembimbing Utama, untuk penyusunan Skripsi dengan judul :

**REKONFIGURASI DENGAN PENDEKATAN MULTI-
OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE
ENHANCED GENETIC ALGORITHM PADA JARINGAN
DISTRIBUSI 20 kV**

Seperti proposal terlampir.

Adapun tugas tersebut sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik pada jurusan Teknik Elektro.

Demikian permohonan kami, atas kesediaan Bapak kami ucapkan terimakasih.

Malang, 16 Oktober 2006

Ketua
Jurusan Teknik Elektro S-1


Ir. F. Yudi Limpraptono, MT
NIP. 1039500274

Pemohon,


Arik Dwi Trisanto
Nim : 01.12.065

Form S-3a



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

PERNYATAAN KESEDIAAN SEBAGAI DOSEN PEMBIMBING SKRIPSI

Sesuai dengan Permohonan Mahasiswa :

Nama : Arik Dwi Tristanto
Nim : 01.12.065
Semester : XI
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik

Dengan ini menyatakan bersedia / ~~tidak bersedia~~ *) menjadi Dosen Pembimbing Utama, untuk penyusunan Skripsi Mahasiswa tersebut dengan judul :

**REKONFIGURASI DENGAN PENDEKATAN MULTI-
OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE
ENHANCED GENETIC ALGORITHM PADA JARINGAN
DISTRIBUSI 20 kV**

Demikian pernyataan ini kami buat untuk dapat dipergunakan sebagaimana mestinya.

Malang, 16 Oktober 2006
Yang Membuat Pernyataan,

Ir. H. Taufik Hidayat, MT
NIP. 1018700015

Catatan :

1. Setelah disetujui agar formulir ini diserahkan mahasiswa/l yang bersangkutan kepada jurusan untuk diproses lebih lanjut.
2. *) Coret yang tidak perlu

Form. S - 3b



Malang, 16 Oktober 2006

Lampiran : Satu Lembar
Perihal : **Kesediaan Sebagai
Dosen Pembimbing**

Kepada : Yth. Ibu Irrine Budi Sulistiawati, ST. MT.
Dosen Jurusan Elektro/ T.Energi Listrik
Institut Teknologi Nasional Malang
di
Malang

Yang bertandatangan di bawah ini :

Nama : Arik Dwi Tristanto
Nim : 01.12.065
Semester : XI
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik

Dengan ini mengajukan permohonan, kiranya Ibu bersedia menjadi Dosen Pembimbing Pendamping, untuk penyusunan Skripsi dengan judul :

**REKONFIGURASI DENGAN PENDEKATAN MULTI-
OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE
ENHANCED GENETIC ALGORITHM PADA JARINGAN
DISTRIBUSI 20 kV**

Seperti proposal terlampir.
Adapun tugas tersebut sebagai salah satu syarat untuk mendapatkan gelar Sarjana Teknik pada jurusan Teknik Elektro.
Demikian permohonan kami, atas kesediaan Ibu kami ucapkan terimakasih.

Malang, 16 Oktober 2006

Ketua
Jurusan Teknik Elektro S-1

Ir. F. Yudi Limpraptono, MT
NIP.1039500274

Pemohon,

Arik Dwi Tristanto
Nim : 01.12.065

Form S-3a



PERNYATAAN KESEDIAAN SEBAGAI DOSEN PEMBIMBING SKRIPSI

Sesuai dengan Permohonan Mahasiswa :

Nama : Arik Dwi Tristanto
Nim : 01.12.065
Semester : XI
Jurusan : Teknik Elektro S-1
Konsentrasi : Teknik Energi Listrik

Dengan ini menyatakan bersedia / tidak bersedia *) menjadi Dosen Pembimbing Pendamping, untuk penyusunan Skripsi Mahasiswa tersebut dengan judul :

REKONFIGURASI DENGAN PENDEKATAN MULTI-OBJEKTIF BERBASIS FUZZY MENGGUNAKAN METODE ENHANCED GENETIC ALGORITHM PADA JARINGAN DISTRIBUSI 20 kV

Demikian pernyataan ini kami buat untuk dapat dipergunakan sebagaimana mestinya.

Malang, 16 Oktober 2006
Yang Membuat Pernyataan,

Irrine Budi Sulistiawati, ST. MT.
NIP. 132314400

Catatan :

1. Setelah disetujui agar formulir ini diserahkan mahasiswa/l yang bersangkutan kepada jurusan untuk diproses lebih lanjut.
2. *) Coret yang tidak perlu



PT. BNI (PERSERO) MALANG
BANK NIAGA MALANG

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

Nomor : ITN-015/I.TA/2/07
Lampiran : -
Perihal : **BIMBINGAN SKRIPSI**

Malang, 06 Jan. 2007

Kepada : Yth. Sdr. **Ir. H. TAUFIK HIDAYAT, MT**
Dosen Pembimbing
Jurusan T. Elektro S-1
di
Malang

Dengan hormat
Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : ARIK DWI TRISTANTO
Nim : 0112065
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Konsentrasi : T. Energi Listrik

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

18 Des. 2006 s/d 18 Juni 2007

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik,
Jurusan Teknik Elektro S-1
Demikian atas perhatian serta kerjasamanya yang baik kami sampaikan terima kasih.



Ketua Jurusan
Teknik Elektro S-1

Fey
Ir. F. Yudi Liliapriono, MT
NIP. Y. 7039500274

Tembusan Kepada Yth:
1. Mahasiswa Yang Bersangkutan
2. Arsip

Form S-4a



FORMULIR BIMBINGAN SKRIPSI

Nama : ARIK DWI TRISTANTO
Nim : 01.12.065
Masa Bimbingan : 18 Desember 2006 s/d 18 Juni 2007
Judul Skripsi : Rekonfigurasi Dengan Pendekatan Multi-Objektif Berbasis Fuzzy Menggunakan Metode Enhanced Genetic Algorithm Pada Jaringan Distribusi 20kV Di G.I. Blimbing

No	Tanggal	Uraian	Paraf Pembimbing
1.	20/12/06	Selesaikan bab W (Analisisnya)	AH
2.	15/01/07	Revisi bab IV, flowchart tempatnya di rubah	AH
3.	25/01/07	Lampirkan ke bab V kesimpulan	AH
4.	5/02/07	Revisi bab V kesimpulan diambil dari	AH
5.		hasil analisis dan berupa angka?	
6.	15/02/07	Lampirkan ke bab III. Teori model	AH
7.	28/02/07	Jelaskan parameter? model	AH
8.	28/03/07	Lampirkan ke bab II. Teori distribusi	AH
9.	10/5/07	Lampirkan bab I	AH
10.	20/06/07	Ace menginput uraian	AH

Malang,
Dosen Pembimbing,


(Ir. H. Taufik Hidayat, MT.)
Nip. 1018700015

Form.S-4b



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

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

T. BNI (PERSERO) MALANG
 BANK NIAGA MALANG

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

Nomor : ITN-016/LTA/2/07
 Lampiran : -
 Perihal : **BIMBINGAN SKRIPSI**

Malang, 06 Jan. 2007

Kepada : Yth. Sdr. **IRRINE BUDI S, ST, MT**
 Dosen Pembimbing
 Jurusan T. Elektro S-1
 di
 Malang

Dengan hormat
 Sesuai dengan permohonan dan persetujuan dalam proposal skripsi untuk mahasiswa :

Nama : ARIK DWI TRISTANTO
 Nim : 0112065
 Fakultas : Teknologi Industri
 Jurusan : Teknik Elektro S-1
 Konsentrasi : T. Energi Listrik

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

18 Des. 2006 s/d 18 Juni 2007

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik,
 Jurusan Teknik Elektro S-1
 Demikian atas perhatian serta kerjasamanya yang baik kami sampaikan terima kasih.



Ketua Jurusan
 Teknik Elektro S-1

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

- Tembusan Kepada Yth:
1. Mahasiswa Yang Bersangkutan
 2. Arsip

Form S-4a



FORMULIR BIMBINGAN SKRIPSI

Nama : ARIK DWI TRISTANTO
Nim : 01.12.065
Masa Bimbingan : 18 Desember 2006 s/d 18 Juni 2007
Judul Skripsi : Rekonfigurasi Dengan Pendekatan Multi-Objektif Berbasis Fuzzy Menggunakan Metode Enhanced Genetic Algorithm Pada Jaringan Distribusi 20kV Di G.I. Blimbing

No	Tanggal	Uraian	Paraf Pembimbing
1.	25 Juni 2007	Tujuan diperjelas utali perhitungan dari Nagi Jaga yg ingin di ketelitian.	
2.	30 Juni 2007	Ac. bab 3 bab 1. 2. 3. 4. 5. 6. 7. 8. 9. 10. 11. 12. 13. 14. 15. 16. 17. 18. 19. 20. 21. 22. 23. 24. 25. 26. 27. 28. 29. 30. 31. 32. 33. 34. 35. 36. 37. 38. 39. 40. 41. 42. 43. 44. 45. 46. 47. 48. 49. 50. 51. 52. 53. 54. 55. 56. 57. 58. 59. 60. 61. 62. 63. 64. 65. 66. 67. 68. 69. 70. 71. 72. 73. 74. 75. 76. 77. 78. 79. 80. 81. 82. 83. 84. 85. 86. 87. 88. 89. 90. 91. 92. 93. 94. 95. 96. 97. 98. 99. 100.	
3.	23 Juli 2007	bab 11 dan 12 BAB 11 dan 12 tentang perhitungan & cara	
4.	20 Juli 2007	revisi bab 11	
5.	30 Juli 2007	revisi bab 11 bab 12 revisi rumus & lampiran.	
6.	14 Agustus	bab 13 flowchart diperbaiki bab 14. Proses perhitungan R & X dan power. Bgm cara membuat rekonfigurasi	
7.	27 Agustus	dit.	
8.			
9.			
10.			

Malang, 27 - 8 - 07
Dosen Pembimbing,

(Irrine Budi Sulistiawati, ST. MT.)
Nip. 132314400

Form.S-4b



PT. PLN (PERSERO)

DISTRIBUSI JAWA TIMUR

AREA PELAYANAN & JARINGAN MALANG



Certificate No: **GBC**
00475

p. : 0341 - 326034 (Hunting)
tak Pos : 211

Facsimile : 0341 - 362046

Nomor : 0009 /330/ APJ-MLG/ 2007
Surat Sdr.No : ITN-559/III.TA-2/2/2006
Lampiran :
Perihal : Survey

Malang, 3 Januari 2006

Kepada
ITN Malang
Fakultas Teknologi Industri
Jl. Raya Karanglo, Km2
Di
MALANG

Dengan Hormat,
Menindaklanjuti surat saudara No ITN-559/III.TA-2/2/2006 Tanggal 15 Desember 2006 perihal tersebut di atas, maka dengan ini kami beritahukan bahwa pada prinsipnya kami tidak keberatan / mengizinkan mahasiswa saudara atas nama :

Arik Dwi Tristanto

Nim : 01.12.065

Untuk melaksanakan survey di PT. PLN (PERSERO) Area Pelayanan dan Jaringan Malang mulai tanggal 4 Januari 2007 sampai dengan 31 Januari 2007 dengan catatan PLN hanya memberikan data yang **TIDAK BERSIFAT RAHASIA**,

Sebelumnya diminta agar mahasiswa tersebut mengisi surat pernyataan, dilengkapi dengan pas photo ukuran 3 X 4 Cm di SDM & Administrasi PT. PLN (Persero) Area Pelayanan dan Jaringan Malang.

Demikian agar menjadikan maklum.

ASMAN SDM & ADMINISTRASI

ABDURACHMAN SHODIQ, SH. ST. M. Kn

Tembusan :
1. Manajer UJ & UPJ Terkait
2. Supervisor Terkait



BIODATA PENULIS



Nama : Arik Dwi Trisanto
Tempat, Tgl Lahir : Malang, 12 Oktober 1982
Jenis Kelamin : Laki-Laki
Agama : Islam
Contact person : (0341) 353324
085 646607004

Riwayat Pendidikan:

- 1. MI Sunan Gunung Jati Malang, Th 1989-1995.**
 - 2. SMPN 17 Malang, Th 1995 – 1998.**
 - 3. SMK Nasional Malang, Th 1998 – 2001, Program Studi Listrik.**
 - 4. Institut Teknologi Nasional Malang, Th 2001 – 2007, Jurusan Teknik Elektro S-1, Program Studi Teknik Energi Listrik.**
-