

SKRIPSI

OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 KV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GI. PIER, PASURUAN



**Disusun oleh:
MOHAMMAD KHOIRUL ANAM
01.12.089**

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

SEPTEMBER 2007

LEMBAR PERSETUJUAN

**OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR
PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN
METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO)
DI GI. PIER, PASURUAN**

SKRIPSI

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

Disusun oleh:

MOHAMMAD KHOIRUL ANAM

01.12.089

Diperiksa dan disetujui,

Dosen Pembimbing I

Dosen Pembimbing II


Ir. H. TAUFIK HIDAYAT, MT

NIP.P. 1018700151

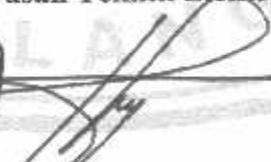

Ir. EKO NURCAHYO

NIP.P. 1028700172

Mengetahui,

Ketua Jurusan Teknik Elektro




Ir. YEDI LIMPRAPTONO, MT

NIP.Y. 1039500274

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

A B S T R A K S I

OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GI. PIER, PASURUAN.

Mohammad Khoirul Anam, Nim 01.12.089, Teknik Elektro/T.Energi Listrik S-1, Dosen Pembimbing I: Ir. H. Taufik Hidayat, MT, Dosen Pembimbing II: Ir. Eko Nurcahyo.

Dengan meningkatnya usaha disektor industri dan meningkatnya taraf hidup masyarakat maka kebutuhan akan energi listrik semakin meningkat pula, sehingga diperlukan penyediaan energi listrik beserta jaringan dan penyaluran yang sangat baik. Bertambahnya industri-industri menyebabkan peran penggunaan alat-alat listrik akan semakin luas, misalnya motor-motor listrik, trafo, AC, lampu-lampu TL dan lain-lain. Beban industri sangat banyak membutuhkan daya reaktif induktif.

Seiring dengan meningkatnya beban-beban induktif, maka daya reaktif yang ada di jaringan akan semakin besar yang selanjutnya akan memperbesar rugi-rugi daya, disamping itu dapat memperburuk kondisi tegangan. Untuk mereduksi rugi daya, solusinya dengan pemasangan kapasitor. Dalam penelitian ini merupakan salah satu solusi dalam penempatan kapasitor yang lebih optimal dari kondisi sebelumnya yaitu dengan Metode *Binary Particle Swarm Optimization* (BPSO) yang bertujuan menentukan penempatan dan kapasitas kapasitor, memperbaiki profil tegangan, mengurangi rugi-rugi daya dan nilai penghematan biaya *losses* setiap tahunnya. Perhitungan aliran daya menggunakan Metode *Newton Raphson* diterapkan untuk mengetahui perubahan hasil dari pemasangan kapasitor.

Pada penyulang Syiar 20 kV dipasang 2 buah kapasitor yaitu pada bus 12 dengan kapasitas 600 kVAR, bus 38 dengan kapasitas 900 kVAR. Besarnya penurunan rugi daya aktif adalah 22.531 kW dari sebelum pemasangan kapasitor 76.759 kW menjadi 54.228 kW, sedangkan untuk daya reaktif juga terjadi penurunan sebesar 34.442 kVAR dari sebelum pemasangan kapasitor 117.325 kVAR menjadi 82.883 kVAR. Tegangan tiap node meningkat sedangkan aliran daya dan arus menurun pada setiap saluran. Total biaya *losses* tahunan sebelum pemasangan kapasitor sebesar US \$ 12,895.49 atau sebesar Rp. 119.928.057,-/tahun, sedangkan total biaya *losses* tahunan setelah pemasangan kapasitor sebesar US \$ 9,407.03 atau sebesar Rp. 87.485.379,-/tahun (asumsi 1 US \$ adalah Rp. 9.300,-) sehingga keuntungan yang di dapat adalah sebesar US \$ 3,488.46 atau sebesar Rp. 32.442.678,-/tahun.

Kata Kunci : Sistem Distribusi Radial, Penempatan Kapasitor, Pengurangan Rugi-rugi, Kapasitor Fixed, Metode *Binary Particle Swarm Optimization* (BPSO).

KATA PENGANTAR

Dengan mengucapkan puji syukur kehadiran Allah SWT atas rahmat dan karunia-Nya, sehingga penyusun dapat menyelesaikan skripsi yang berjudul, “OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GI. PIER, PASURUAN”.

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

Atas segala bimbingan, pengarahan dan bantuan yang diberikan, sehingga skripsi ini dapat terselesaikan. Dalam kesempatan ini, penyusun menyampaikan terima kasih kepada:

1. **Ir. Mochtar Asroni, MSME**, selaku Dekan FTI ITN Malang.
2. **Ir. F Yudi Limpraptono, MT**, selaku Ketua Jurusan Teknik Elektro (S1) ITN Malang.
3. **Ir. Yusuf Ismail Nakhoda, MT**, selaku Sekretaris Jurusan Teknik Elektro (S1) ITN Malang.
4. **Ir. H. Taufik Hidayat, MT**, selaku bapak dan dosen pembimbing I.
5. **Ir. Eko Nurcahyo**, selaku bapak dan dosen pembimbing II.
6. **Ir. M. Abdul Hamid, MT**, dan **Bambang Prio H. ST, MT**, selaku dosen penguji ujian skripsi.

7. Bapak **Mahfut Budiman**, Bapak **Eddy Susianto** dan Ibu **O'on** dan seluruh staf PT. PLN (paesero) Distribusi Jawa Timur APJ. Pasuruan dan UPJ. Bangil terimakasih atas kerja samanya.
8. Keluarga besar penulis, terutama **Bapak Ibuk, Kakak dan Adik** atas ketulusan do'a serta dukungan yang tak henti-hentinya diberikan.
9. Rekan-rekan Mahasiswa dan Alumni Teknik Elektro Energi Listrik yang telah membantu atas terselesaikanya skripsi ini.

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

Akhir kata, penulis mengharapkan skripsi ini dapat berguna dan bermanfaat bagi rekan-rekan mahasiswa khususnya pada jurusan Teknik Elektro S-1 konsentrasi Teknik Energi Listrik.

Malang, September 2007

Penulis

DAFTAR ISI

	Halaman:
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI.....	iii
KATA PENGANTAR	iv
DAFTAR ISI.....	vi
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xiii
DAFTAR GRAFIK.....	xiv
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	2
1.3. Tujuan	2
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian	3
1.6. Sistematika Penulisan	4
BAB II SISTEM DISTRIBUSI TENAGA LISTRIK DAN APLIKASI KAPASITOR	
2.1. Sistem Tenaga Listrik	6
2.2. Sistem Distribusi Tenaga Listrik.....	7

2.2.1. Sistem Distribusi Primer (Jaringan Tegangan Menengah)	8
2.2.2. Sistem Distribusi Sekunder (Jaringan Tegangan Rendah).....	8
2.3. Struktur Jaringan Distribusi Tenaga Listrik	9
2.3.1. Sistem Jaringan Distribusi <i>Radial</i>	9
2.3.2. Sistem Jaringan Distribusi Tertutup (<i>Loop</i>)	10
2.3.3. Sistem Jaringan Distribusi <i>Mesh</i>	11
2.4. Macam-Macam Struktur Jaringan Distribusi <i>Radial</i>	11
2.5. Daya Dalam Sistem Tenaga Listrik	12
2.5.1. Daya Nyata (<i>Real Power</i>)	13
2.5.2. Daya Reaktif (<i>Reactive Power</i>).....	13
2.5.3 Daya Semu (<i>Apparent Power</i>)	13
2.6. Daya Reaktif dan Faktor Daya.....	14
2.7. Kapasitor Daya.....	16
2.7.1. Kapasitor Seri dan Kapasitor <i>Shunt</i>	17
2.7.2. Faktor-Faktor Pemilihan Kapasitor Seri dan Kapasitor <i>Shunt</i> ..	18
2.8. Pengaruh Pemasangan Kapasitor <i>Shunt</i>	19
2.8.1. Pengurangan Rugi-Rugi Saluran dengan Kapasitor <i>Shunt</i>	20
2.8.2. Perbaikan Tegangan.....	21
2.8.3. Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem	22
2.8.4. Perhitungan Pengaruh Perbaikan Faktor Daya	24
2.8.5. Penentuan Rating Kapasitor untuk Perbaikan Faktor Daya Beban	25
2.9. Sistem <i>Per-Unit</i>	26

BAB III OPTIMASI PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL DENGAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO)

3.1. Analisa Aliran Daya Jaringan <i>Radial</i>	28
3.1.1 Tujuan	28
3.1.2. Metode <i>Newton Raphson</i>	28
3.1.3. Algoritma Aliran Daya <i>Newton Raphson</i>	31
3.1.4. <i>Flowchart</i> Algoritma Aliran Daya <i>Newton Rahpson</i>	32
3.2. Penyelesaian Metode <i>Binary Partile Swarm Optimization</i> (BPSO)	33
3.2.1. Parameter-parameter PSO	33
3.2.2. <i>Binary Partile Swarm Optimization</i> (BPSO)	35
3.3. Fungsi dan Tujuan (<i>Objective Functions</i>).....	37
3.3.1. <i>Total Harmonic Distortion</i> (THD).....	37
3.3.2. Meminimalkan Pengeluaran Konstruksi Kapasitor	38
3.3.3. Meminimalkan Kehilangan Daya Nyata.....	38
3.3.4. Algoritma Program <i>Binary Partile Swarm Optimization</i> (BPSO)	39
3.3.5. Algoritma <i>Objection Function</i>	40
3.3.6. <i>Flowchart Binary Partile Swarm Optimization</i> (BPSO)	41
3.3.7. <i>Flowchart Objection Function</i>	42

3.4.	Data Saluran	43
3.5.	Data Pembebanan	45

BAB IV OPTIMASI PENENTUAN LETAK, KAPASITAS KAPASITOR, JUMLAH PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL PENYULANG SYIAR

4.1.	Aplikasi Program Komputer	47
4.2.	Uji Validasi Program Komputer	47
4.2.1.	Validasi Data IEEE 9 Bus	47
4.2.2.	Prosentasi Tingkat Kesalahan	50
4.3.	Sistem Distribusi Tenaga Listrik 20kV GI. PIER Penyulang Syiar	51
4.4.	Analisa Perhitungan	53
4.5.	Hasil Perhitungan Program <i>Binary Partile Swarm Optimization</i> (BPSO)	55
4.6.	Hasil Analisa Program dengan menggunakan <i>Binary Partile Swarm Optimization</i> (BPSO)	64

BAB V KESIMPULAN

5.1.	Kesimpulan	68
5.2.	Saran	69

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

Gambar:	Halaman:
2.1 Skema Penyaluran Energi Listrik.....	6
2.2 Jaringan Distribusi Tegangan Menengah (JTM), Jaringan Tegangan Rendah (JTR) dan Sambungan Rumah ke Pelanggan.....	7
2.3 Jaringan Tegangan Menengah Sistem Distribusi <i>Radial</i>	9
2.4 Jaringan Tegangan Menengah Struktur <i>Loop</i> yang dipasok dari satu sumber	10
2.5 Jaringan Tegangan Menengah Struktur <i>Mesh</i>	11
2.6 Segi Tiga Daya	14
2.7 Segi Tiga Arus.....	15
2.8 Saluran Primer Dengan Beban Tepusat	16
2.9 Kapasitor <i>Shunt</i> yang Dipasang pada Saluran	17
2.10 Vektor arus dan tegangan sebelum dan setelah pemasangan kapasitor	20
2.11 Diagram vector pada rangkaian dengan pf (<i>lagging</i>) (a) dan (c) tanpa kapasitor <i>shunt</i> , (b) dan (d) dengan kapasitor <i>shunt</i>	21
2.12 Perbaikan Faktor Daya.....	22
2.13 Diagram Fasor dan Sudut Daya Beban Distribusi	24
3.1 <i>Flowchart</i> Algoritma Aliran Daya <i>Newton Raphson</i>	32
3.2 <i>Flowchart</i> Algoritma Solusi <i>Optimal Power Flow</i> dengan Metode <i>Binary Particle Swarm Optimization (BPSO)</i>	41

3.3.	<i>Flowchart Objective Functions</i>	42
4.1.	Tampilan <i>Load Flow</i> Awal	49
4.2.	Tampilan <i>Load Flow</i> Akhir.....	49
4.3.	Tampilan Hasil Program	50
4.4.	<i>Single Line Diagram</i> Penyulang Syiar Sebelum Pencmpatan	52
4.5.	<i>Flowchart</i> Pemecahan masalah.....	54
4.6.	Tampilan Utama Program	55
4.7.	Tampilan Inputan Data (General)	56
4.8.	Tampilan Inputan Data (Data Pembebanan).....	56
4.9.	Tampilan Inputan Data (Data Saluran).....	57
4.10.	Tampilan Hasil Aliran Daya Untuk Mengetahui Profil Tegangan Dan Sudut Fasa Tiap-tiap Bus Sebelum Optimasi	57
4.11.	Tampilan Hasil Aliran Daya Untuk Arus yang Mengalir pada Saluran dan Daya tiap Saluran Sebelum Optimasi.....	58
4.12.	Tampilan Total Pembangkitan, Pembebanan, dan Rugi-rugi Sebelum Optimasi.....	58
4.13.	Tampilan Parameter Yang Digunakan	59
4.14.	Tampilan Hasil Aliran Daya Untuk Mengetahui Profil Tegangan Dan Sudut Fasa Tiap-tiap Bus Setelah Optimasi	60
4.15.	Tampilan Hasil Aliran Daya Untuk Arus yang Mengalir pada Saluran dan Daya tiap Saluran Setelah Optimasi.....	60
4.16.	Tampilan Total Pembangkitan, Pembebanan, dan Rugi-rugi Sebelum Optimasi.....	61

4.17. Tampilan Perbandingan Profil Tegangan Sebelu dan Setelah Optimasi... 61

4.18. Hasil Penempatan Metode *Binary Partile Swarm Optimization* (BPSO). 62

4.17. *Single Line Diagram* Penyulang Syiar Setelah Penempatan..... 63

DAFTAR TABEL

Tabel :	Halaman:
2-1 Kapasitor Seri dan Kapasitor <i>Shunt</i>	21
3-1 Spesifikasi Saluran	43
3-2 Data Saluran Penyulang Syiar.....	44
3-3 Data Pembebanan Sistem 20kV Penyulang Syiar.....	42
4-1. Data Saluran IEEE 9 <i>Bus</i>	48
4-2. Data Beban IEEE 9 <i>Bus</i>	48
4-3. Hasil Analisa Program dengan Metode <i>Binary Partile Swarm Optimization</i> (BPSO)	64

DAFTAR GRAFIK

Grafik:	Halaman:
4-1 Tegangan Tiap-tiap Bus Terhadap Assumsi Tegangan Awal, Tegangan Sebelum Dan Setelah Optimasi.....	65
4-2 Rugi-rugi Saluran Sebelum Dan Sesudah Optimasi	65
4-3 Harga Rugi Energi Sebelum Dan Sesudah Optimasi	66

BAB I

PENDAHULUAN

1.1. Latar Belakang

Dengan meningkatkan usaha di sektor industri dan meningkatkan taraf hidup masyarakat maka kebutuhan akan energi semakin meningkat pula, sehingga diperlukan penyediaan energi listrik beserta jaringan dan penyaluran yang sangat baik.

Alokasi daya reaktif yang tepat dalam sistem bisa menghasilkan kondisi yang optimal, yaitu kondisi rugi-rugi daya yang minimum dengan biaya investasi yang murah dan mampu memperbaiki profil tegangan pada sistem. Permasalahan besarnya rugi-rugi daya di saluran menjadi perhatian utama khususnya pada saluran distribusi 20 kV.

Alternatif yang sering dipakai untuk memperbaiki kondisi jaringan akibat rugi-rugi daya tersebut adalah dengan memasang sumber daya reaktif tambahan di sisi beban salah satunya dengan memasang kapasitor.

Banyak metode yang dipakai dalam menganalisa masalah penentuan penempatan dan kapasitas kapasitor pada jaringan distribusi, pada umumnya banyak teknik untuk membahas masalah alokasi kapasitor dengan menggunakan metode-metode konvensional tetapi pembahasan-pembahasan yang dilakukan masih kurang dan masih perlu dilakukan analisa yang lebih mendalam lagi.

Hal ini mendorong penulis untuk mengangkat permasalahan dengan menggunakan metode *Binary Particle Swarm Optimization (BPSO)* agar dapat menjadi salah satu alternatif yang efektif untuk penempatan dan kapasitas kapasitor dalam jaringan distribusi 20 kV.

1.2. Rumusan Masalah

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

1. Dimana letak dan kapasitas kapasitor serta investasi dalam peralatan optimasi daya reaktif pada saluran distribusi GI. PIER, Penyulang SYIAR Pasuruan dengan metode *Binary Particle Swarm Optimization (BPSO)*?
2. Berapa profil tegangan setelah dilakukan optimasi?
3. Berapa rugi-rugi daya pada saluran setelah penempatan kapasitor dengan menggunakan metode *Binary Particle Swarm Optimization (BPSO)*?

1.3. Tujuan

Adapun tujuan penulisan skripsi ini adalah:

1. Menentukan letak dan kapasitas kapasitor yang optimal pada sistem distribusi radial penyulang Syiar.
 2. Memperbaiki profil tegangan setelah pemasangan kapasitor.
 3. Mengurangi rugi-rugi daya aktif dan rugi-rugi daya reaktif setelah pemasangan kapasitor.
-

4. Menentukan nilai penghematan biaya *losses* setiap tahun setelah penempatan kapasitor.

1.4. Batasan Masalah

Agar permasalahan mengarah sesuai dengan tujuan, maka pembahasan makalah skripsi ini di batasi pada hal-hal sebagai berikut:

1. Jaringan yang akan dianalisa adalah jaringan distribusi tipe radial 20 kV di Penyulang Syiar.
2. Hanya kapasitor *Shunt* yang digunakan dalam perbaikan tegangan dan pengurangan rugi-rugi daya.
3. Analisa penentuan letak dan kapasitas kapasitor menggunakan metode *Binary Particle Swarm Optimization* (BPSO).
4. Jaringan dalam kondisi normal.
5. Studi aliran daya yang digunakan *Newton Raphson*.
6. Diasumsikan nilai faktor daya sebesar 0.85.

1.5. Metodologi Penelitian

Metodologi yang digunakan dalam penyusunan skripsi ini adalah:

1. Kajian literatur

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

2. Pengumpulan Data

Bentuk data yang digunakan adalah:

- ▣ Data kuantitatif, yaitu data yang dapat dihitung atau data yang berbentuk angka-angka.
 - ▣ Data kualitatif, yaitu data yang berbentuk diagram. Dalam hal ini *single line* diagram penyulang.
3. Simulasi, pembahasan masalah dan analisa perhitungan tegangan, sudut fasa tegangan, rugi-rugi daya dan aliran daya tiap saluran menggunakan metode *Newton Raphson*, untuk memecahkan permasalahan *Objective Function*, menentukan penempatan dan kapasitas kapasitor yang optimal menggunakan metode *Binary Particle Swarm Optimization (BPSO)* yang disimulasikan dengan bahasa pemrogram *Borland Delphi* versi 7.0.
 4. Validasi, merupakan pengujian atau penegasan bahawa program komputer yang digunakan adalah benar.
 5. Kesimpulan, merupakan hasil akhir dari analisa.

1.6. Sistematika Penulisan

Sistematika dari pembahasan didalam skripsi ini adalah sebagai berikut:

BAB I : PENDAHULUAN

Berisikan Latar belakang, Rumusan Masalah, Tujuan, Batasan Masalah, Metodologi Penelitian, Sistematika Penulisan.

BAB II : SISTEM DISTRIBUSI TENAGA LISTRIK DAN APLIKASI KAPASITOR

Disini akan dibahas masalah sistem jaringan distribusi, pengertian jatuh tegangan, pengaruh rendahnya jatuh daya, penjelasan teori tentang kapasitor serta faktor daya, dan penjelasan sistem *per-unit*.

BAB III : OPTIMASI PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL DENGAN METODE *BINARY PARTICLE SWARM OPTIMIZATION (BPSO)*

Pada bab ini akan dibahas metode aliran daya *Newton Raphson*, teori mengenai metode *Binary Particle Swarm Optimization (BPSO)* untuk menentukan penempatan, kapasitas kapasitor, data beban dan data saluran yang dipakai.

BAB IV : OPTIMASI PENENTUAN LETAK, JUMLAH DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL PENYULANG SYIAR, PASURUAN

Pada bab ini akan dibahas mengenai validasi program, penentuan letak dan kapasitas kapasitor menggunakan metode *Binary Particle Swarm Optimization (BPSO)*, perbandingan profil tegangan, rugi daya dan nilai biaya sebelum dan setelah penempatan kapasitor.

BAB V : KESIMPULAN

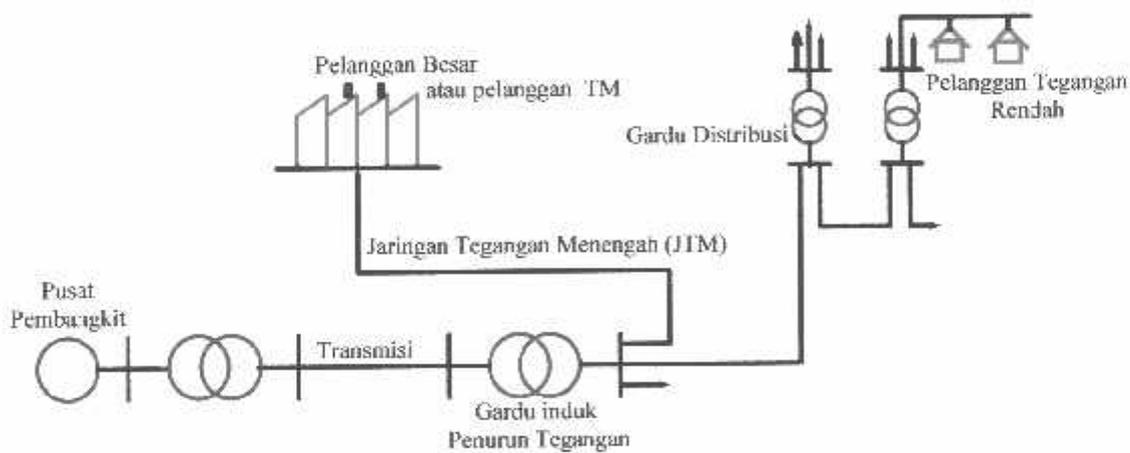
Merupakan bab terakhir yang memuat intisari dari hasil pembahasan, yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan penulisan selanjutnya.

BAB II

SISTEM DISTRIBUSI TENAGA LISTRIK DAN APLIKASI KAPASITOR

2.1. Sistem Tenaga Listrik^[1]

Sistem tenaga listrik merupakan suatu sistem terpadu yang terbentuk oleh hubungan-hubungan peralatan dan komponen-komponen listrik. Sistem tenaga listrik ini mempunyai peranan utama untuk menyalurkan energi listrik yang dibangkitkan oleh generator ke konsumen yang membutuhkan energi listrik.



Gambar 2.1. Skema Penyaluran Energi Listrik^[1]

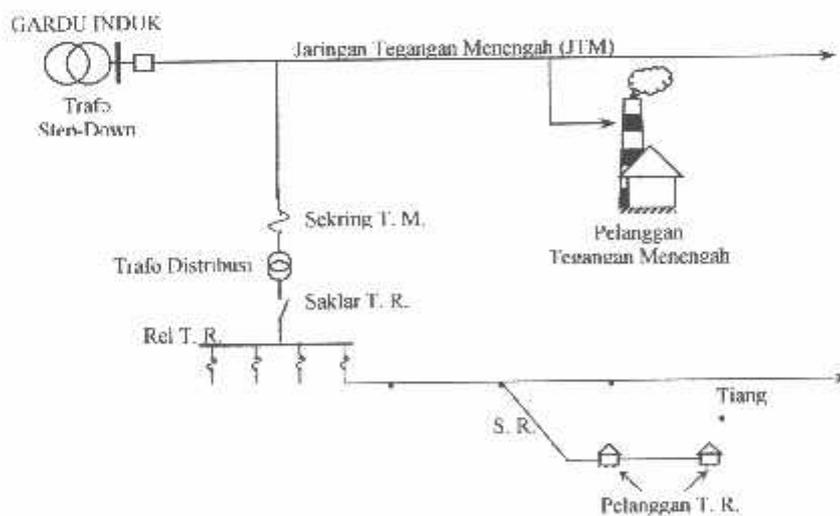
Secara garis besar suatu sistem tenaga listrik dapat dikelompokkan menjadi 3 sub sistem, yaitu:

1. Sistem Pembangkitan: Berperan sebagai sumber daya tenaga listrik dan disebut juga prodكتور energi.

2. Sistem Transmisi atau Penyaluran: Berfungsi sebagai penyalur daya listrik secara besar-besaran dari pembangkit ke bagian sistem distribusi atau konsumen.
3. Sistem Distribusi dan Beban: Berperan sebagai distributor energi ke konsumen yang memerlukan energi tersebut.

2.2. Sistem Distribusi Tenaga Listrik^[2]

Jaringan distribusi berada pada akhir dari sistem tenaga listrik, perannya mendistribusikan tenaga listrik dari Gardu Induk (GI) ke konsumen melalui gardu distribusi. Jaringan setelah keluar dari GI bisa disebut jaringan distribusi. Setelah tenaga listrik disalurkan melalui jaringan distribusi primer maka kemudian tenaga listrik diturunkan tegangannya dalam gardu-gardu distribusi menjadi tegangan rendah, kemudian disalurkan melalui jaringan tegangan rendah untuk selanjutnya disalurkan kerumah-rumah pelanggan (konsumen) PLN melalui sambungan rumah.



Gambar 2.2. Jaringan Distribusi Tegangan Menengah (JTM), Jaringan Tegangan Rendah (JTR) dan Sambungan Rumah ke Pelanggan^[2]

Dalam pendistribusian tenaga listrik ke konsumen, tegangan yang digunakan bervariasi tergantung dari jenis konsumen yang membutuhkan. Untuk konsumen industri biasanya digunakan tegangan menengah 20 kV sedangkan untuk konsumen perumahan digunakan tegangan rendah 220/380 Volt, yang merupakan tegangan siap pakai untuk peralatan-peralatan rumah tangga. Dengan demikian maka sistem distribusi tenaga listrik dapat diklasifikasikan menjadi dua bagian sistem yaitu:

1. Sistem Distribusi Primer (Jaringan Tegangan Menengah)
2. Sistem Distribusi Sekunder (Jaringan Tegangan Rendah)

Pengklasifikasian sistem distribusi tenaga listrik menjadi dua ini berdasarkan tingkat tegangan distribusinya.

2.2.1. Sistem Distribusi Primer (Jaringan Tegangan Menengah)

Tingkat Tegangan yang digunakan pada sistem distribusi primer adalah meliputi tegangan 20 kV, Oleh karena itu sistem distribusi ini sering disebut dengan sistem distribusi tegangan menengah.

2.2.2. Sistem Distribusi Sekunder (Jaringan Tegangan Rendah)

Tingkat Tegangan yang digunakan pada sistem distribusi sekunder adalah tegangan rendah yaitu 127/220 volt atau 220/380 volt, oleh karena itu sistem distribusi ini sering disebut dengan sistem distribusi tegangan rendah.

Sistem jaringan yang digunakan untuk menyalurkan dan mendistribusikan tenaga listrik tersebut dapat menggunakan sistem satu fasa dengan dua kawat maupun sistem tiga fasa dengan empat kawat.

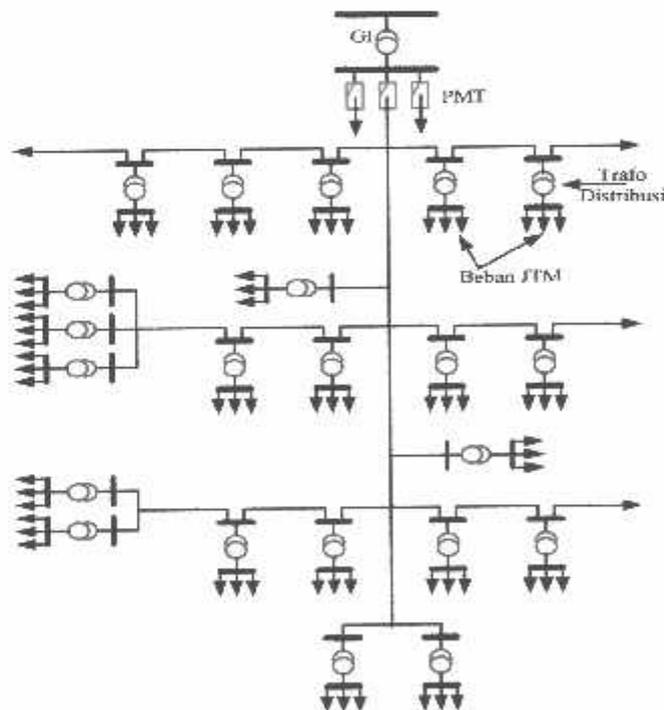
2.3. Struktur Jaringan Distribusi Tenaga Listrik^[1]

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

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

2.3.1. Sistem Jaringan Distribusi *Radial*

Bentuk jaringan ini merupakan bentuk dasar yang paling sederhana dan paling banyak dipergunakan. Sistem ini dikatakan *radial* karena dari kenyataannya bahwa jaringan ini ditarik secara *radial* dari GI. ke pusat-pusat beban atau konsumen yang dilayani. Sistem ini terdiri dari saluran utama (*tunk line*) dan saluran cabang (*lateral*) seperti pada gambar 2.3.



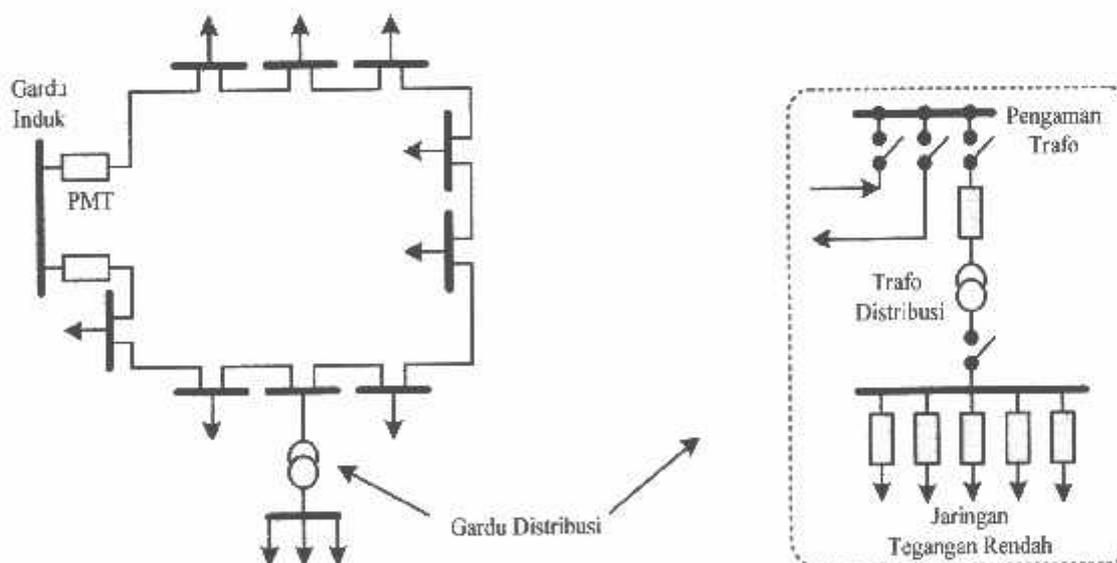
Gambar 2.3. Jaringan Tegangan Menengah Sistem Distribusi *Radial*^[1]

Pelayanan tenaga listrik untuk suatu daerah beban tertentu dilaksanakan dengan memasang transformator pada sembarang titik pada jaringan yang 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 dicabangkan pada saluran tersebut.

Kelemahan yang dimiliki oleh sistem *radial* adalah jatuh tegangan yang cukup besar dan bila terjadi gangguan pada salah satu *feeder* maka semua pelanggan yang terhubung pada *feeder* tersebut akan terganggu.

2.3.2. Sistem Jaringan Distribusi Tertutup (*Loop*).

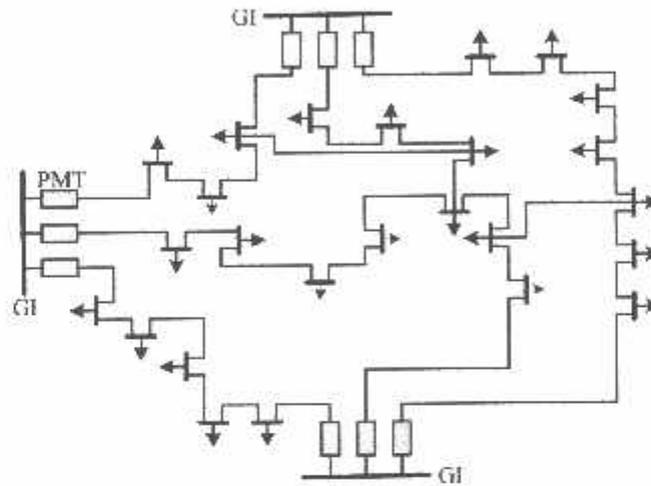
Sistem ini disebut jaringan distribusi *loop* karena saluran primer yang menyalurkan daya sepanjang daerah beban yang dilayani, membentuk suatu rangkaian *loop*.



Gambar 2.4. Jaringan Tegangan Menengah Struktur *Loop* yang dipasok dari satu sumber^[1]

2.3.3. Sistem Jaringan Distribusi *Mesh*.

Jaringan Distribusi *Mesh* merupakan jaringan yang strukturnya kompleks, dimana kelangsungan penyaluran dan pelayanannya diutamakan. Struktur jaringan ini umumnya digunakan pada jaringan tegangan rendah yang kepadatan bebannya cukup tinggi.



Gambar 2.5. Jaringan Tegangan Menengah Struktur *Mesh*⁽¹⁾

2.4. Macam-macam Struktur Jaringan Distribusi *Radial*

Tipe jaringan ini merupakan bentuk dasar, susunan maupun kebutuhan alat-alat penunjangnya paling sedikit dan paling sederhana. Tipe ini paling banyak digunakan untuk melayani konsumen terutama beban-beban rumah tangga yang tidak menuntut tingkat kontinuitas pelayanan yang tinggi. Sumber daya pada tipe ini hanya dari satu titik. Salurannya dicabang-cabang menuju ke titik-titik beban, dan antara titik sumber dengan titik beban hanya ada satu pilihan. 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 dan kelemahan sebagai berikut:

Keuntungan jaringan *radial*:

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

Kelemahan jaringan *radial*:

1. Kualitas pelayanan (penyaluran daya) dibanding dengan tipe yang lain lebih jelek, sebab jatuh tegangan dan rugi-rugi daya relatif besar. Kerugian ini terjadi pada saluran.
2. Kontinuitas penyaluran daya tidak terjamin. Dengan hanya mengandalkan satu saluran, maka bila terjadi gangguan pada saluran tersebut tidak dapat diharapkan suplai melalui saluran ini.

2.5. Daya Dalam Sistem Tenaga Listrik^[3]

Dalam sistem tenaga listrik, pembangkit-pembangkit tenaga listrik harus mampu menyediakan tenaga listrik kepada pelanggan sesuai dengan permintaan beban listrik yang ada dan hal yang harus diperhatikan adalah sistem yang tetap konstan, dalam hal ini tegangan dan frekuensi harus tetap konstan karena berhubungan dengan daya.

Daya listrik yang dibangkitkan dikenal dengan istilah:

2.5.1. Daya Nyata (*Real Power*)

Daya Nyata dinyatakan dalam persamaan:

$$P = [V] [I] \cos \varphi \dots\dots\dots(2.1)$$

Daya nyata untuk beban 3 fasa seimbang

$$P = \sqrt{3} |V_{\text{jala-jala}}| |I_{\text{jala-jala}}| \cos \varphi \dots\dots\dots(2.2)$$

2.5.2. Daya Reaktif (*Reactive power*)

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

Daya reaktif dinyatakan dalam persamaan:

$$Q = |V| |I| \sin \varphi \dots\dots\dots(2.3)$$

Daya reaktif untuk beban 3 fasa seimbang:

$$Q = \sqrt{3} |V_{\text{jala-jala}}| |I_{\text{jala-jala}}| \sin \varphi \dots\dots\dots(2.4)$$

2.5.3. Daya Semu (*Apparent Power*)

Daya semu dinyatakan dalam persamaan:

$$S = |V| |I| \dots\dots\dots(2.5)$$

Daya semu untuk beban 3 fasa seimbang:

$$S = \sqrt{3} |V_{\text{jala-jala}}| |I_{\text{jala-jala}}| \dots\dots\dots(2.6)$$

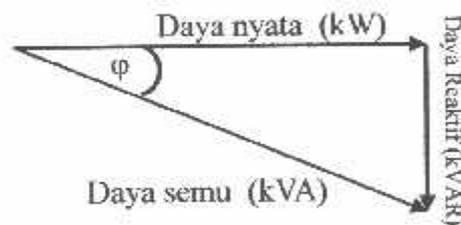
2.6. Daya Reaktif dan Faktor Daya^[1]

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 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 kompoen daya nyata (kW) dan komponen daya reaktif (kVAR).

Hubungan ini dapat digambarkan sebagai berikut:



Gambar 2.6. Segitiga Daya^[1]

Dengan Faktor daya = $\text{Cos } \varphi = \frac{P}{S}$

$$\text{Cos } \varphi = \frac{P}{\sqrt{P^2 + Q_1^2}} \dots\dots\dots(2.8)$$

P = Daya Nyata (kW)

S = Daya semu (kVA)

$Q =$ Daya reaktif (kVAR)

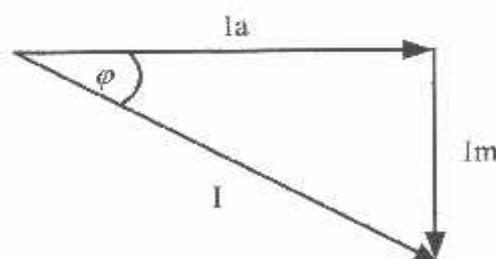
$\varphi =$ Sudut *Phase*

Dari gambar 2.6. daya semu terdiri dari komponen daya nyata dan daya reaktif.

Suatu beban akan membutuhkan suplai daya aktif jika beban tersebut bersifat induktif dan suatu beban membutuhkan suplai daya reaktif jika beban tersebut bersifat kapasitif. Jadi faktor daya dapat dilihat dari hubungan antara arus nyata, arus magnetisasi dan arus total.

- Arus nyata (I_a) adalah arus yang dibeban dan diubah ke dalam energi panas.
- Arus magnetisasi (I_m) adalah arus yang mengalir dibeban untuk menimbulkan medan magnet.
- Arus total (I) adalah arus yang mengalir di jaringan dan merupakan penjumlahan vektor dari arus nyata dengan arus magnetisasi.

Dalam bentuk vektor hubungan tersebut digambarkan sebagai berikut :



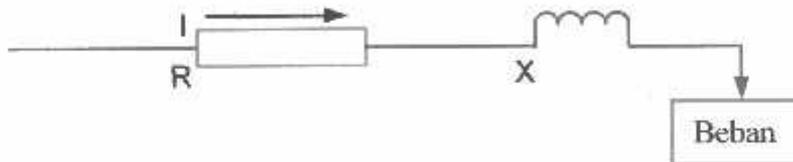
Gambar 2.7. Segitiga Arus ^[1]

Beberapa sebab sistem distribusi mempunyai faktor daya yang rendah, yaitu:

- Banyaknya pemakaian motor asinkron terutama pada industri.

- Makin meningkatnya pemakaian lampu TL untuk penerangan.
- Pemakaian pemanas air.

Menurunnya faktor daya berarti mengecilnya perbandingan antara daya nyata dengan daya semu atau berarti semakin membesarnya kebutuhan beban dan daya aktif.



Gambar 2.8. Saluran Primer Dengan Beban Tepusat^[1]

Karena pada saluran terdapat resistansi R dan reaktansi X , maka rugi daya (P_L) dirumuskan sebagai berikut:

$$\begin{aligned}
 P_L &= I^2 \times R \\
 &= (I \cos \theta)^2 R + (I \sin \theta)^2 R \dots\dots\dots (2.9)
 \end{aligned}$$

dimana: I_R adalah arus aktif

I_X adalah arus reaktif

2.7. Kapasitor Daya^[1]

Secara sederhana kapasitor terdiri dari dua buah plat logam yang dipisahkan oleh suatu bahan elektrik dan kapasitor ini mempunyai sifat menyimpan muatan listrik. Pada beberapa tahun lalu kebanyakan kapasitor terbuat dari dua buah plat aluminium murni yang dipisahkan oleh tiga atau lebih lapisan kertas yang dilapisi oleh bahan kimia. Kapasitor daya telah mengalami perkembangan yang

begitu cepat selama 30 tahun terakhir. Karena bahan elektrik yang digunakan lebih efisien serta teknologi pembuatan kapasitor lebih baik.

2.7.1. Kapasitor Seri dan Kapasitor *Shunt*

Kapasitor daya terdiri atas 2 bagian yaitu kapasitor seri dan kapasitor *shunt*:

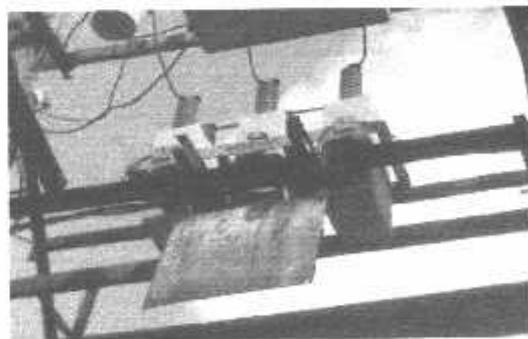
a. Kapasitor Seri

Kapasitor seri adalah kapasitor yang dihubungkan seri dengan impedansi saluran yang bersangkutan, pemakaiannya amat dibatasi pada saluran distribusi, karena peralatan pengamannya cukup rumit. Jadi secara umum dikatakan biaya untuk pemasangan kapasitor seri lebih mahal dari pada biaya pemasangan kapasitor paralel atau biasa disebut kapasitor *shunt*.

b. Kapasitor *Shunt*

Kapasitor *shunt* adalah kapasitor yang dihubungkan paralel dengan saluran dan secara intensif digunakan pada saluran distribusi. Kapasitor *shunt* mencatu daya reaktif atau arus yang menentang komponen arus beban induktif.

Dengan dipasangnya kapasitor *shunt* pada jaringan distribusi akan dapat memperbaiki profil tegangan, memperbaiki faktor daya dan menaikkan kapasitas sistem serta dapat mengurangi rugi saluran.



Gambar 2.9. Kapasitor *shunt* yang dipasang pada saluran

2.7.2. Faktor-faktor Pemilihan Kapasitor Seri dan Kapasitor *Shunt*^[4]

Faktor-faktor yang mempengaruhi pemilihan kapasitor *shunt* dan seri ditabelkan sebagai berikut:

Tabel 2-1
Kapasitor Seri dan Kapasitor *Shunt*

No	Tujuan	Kapasitor	
		Seri	Shunt
1	Memperbaiki faktor daya	Kedua	Pertama
2	Memperbaiki tingkat tegangan pada sistem saluran udara dengan faktor daya normal dan rendah	Pertama	Kedua
3	Memperbaiki tingkat tegangan pada sistem saluran udara dengan faktor daya yang tinggi	Tidak Dipakai	Pertama
4	Memperbaiki tingkat tegangan pada sistem saluran udara bawah tanah dengan faktor daya yang tinggi	Tidak Dipakai	Tidak Dipakai
5	Memperbaiki tingkat tegangan pada sistem saluran bawah tanah dengan faktor daya normal dan rendah	Pertama	Tidak Dipakai
6	Mengurangi rugi-rugi daya dan rugi-rugi energi pada saluran	Kedua	Pertama
7	Mengurangi fluktuasi tegangan	Pertama	Tidak Dipakai

2.8. Pengaruh Pemasangan Kapasitor *Shunt*^[1]

Kapasitor *shunt* adalah kapasitor yang dihubungkan paralel dengan saluran dan secara intensif digunakan pada sistem distribusi. Kapasitor *shunt* mencatu daya reaktif atau yang menentang komponen arus beban induktif. Dengan dipasang kapasitor *shunt* pada jaringan distribusi akan dapat memperbaiki profil tegangan, memperbaiki faktor daya dan menaikkan kapasitas sistem serta dapat mengurangi rugi saluran.

Adapun dua cara dalam pemakaian kapasitor *shunt*:

- ▣ Kapasitor Tetap (*Fixed Capacitor*)
- ▣ Kapasitor Saklar (*Switched Capacitor*)

a. Kapasitor Tetap (*Fixed Capacitor*)

Adalah kapasitor untuk optimasi daya reaktif yang kapasitasnya tetap dan selalu terpasang di jaringan. Penggunaan kapasitor jenis ini harus memperhatikan kenaikan tegangan yang terjadi pada saat beban ringan agar tidak melebihi tegangan yang ditetapkan. Pada sekripsi ini, saya menggunakan kapasitor tetap sebagai obyek penelitian.

b. Kapasitor Saklar (*Switched Capacitor*)

Adalah kapasitor untuk kompensasi daya reaktif yang dapat dihubungkan dan di lepaskan dari jaringan dan dapat diatur besar kapasitasnya sesuai dengan kondisi beban.

Proses membuka-menutup dari saklar kapasitor *shunt* dapat dilakukan dengan cara manual maupun dengan cara otomatis. Pengendalian secara manual (Pada lokasi atau lokasi jarak jauh) dapat dilakukan pada GI.

Untuk pengendalian secara otomatis, termasuk didalamnya peralatan pengendali tegangan, arus dan suhu. Tipe yang paling populer adalah pengendalian saklar waktu (*time-switch control*), pengendali tegangan dan pengendali tegangan arus.

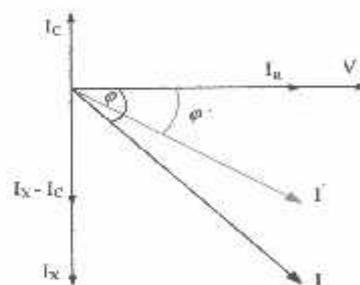
2.8.1. Pengurangan Rugi-Rugi Saluran dengan Kapasitor *Shunt*

Rugi-rugi saluran sebelum ada pengaruh pemasangan kapasitor adalah seperti pada persamaan (2.9) diatas.

Sedangkan setelah pemasangan kapasitor, maka sebagian daya reaktif yang dibutuhkan oleh beban akan disuplai oleh kapasitor tersebut, sehingga arus yang mengalir di jaringan akan lebih kecil. Persamaan rugi daya dan rugi energi setelah pemasangan kapasitor adalah:

$$P_L' = (I \cos \theta)^2 R + (I \sin \theta - I_c)^2 R \dots\dots\dots(2.10)$$

Dalam bentuk diagram vektor adalah:



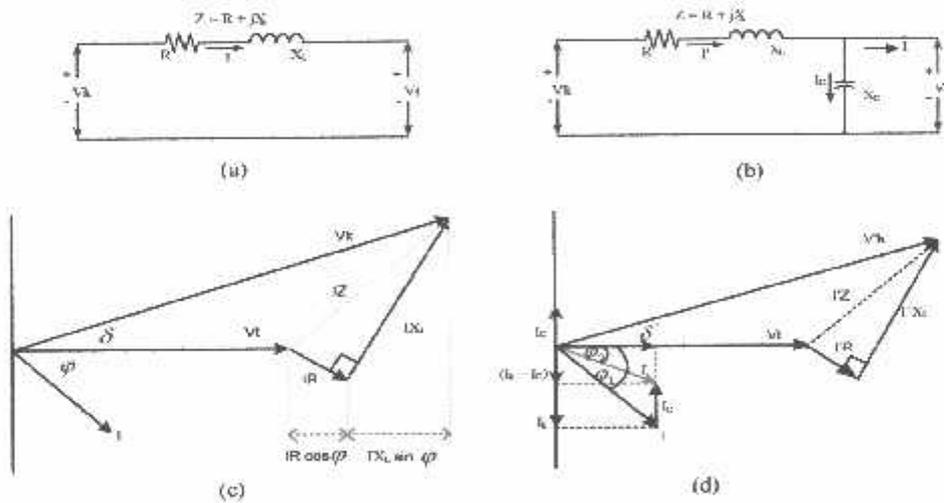
Gambar 2.10. Vektor arus dan tegangan sebelum dan setelah pemasangan kapasitor ^[1]

Sehingga pengurangan rugi daya dengan adanya pemasangan kapasitor *shunt* adalah:

$$\begin{aligned} \Delta_{pls} &= I^2R - I_1^2R \\ &= (I \cos \theta)^2R + (I \sin \theta)^2R - (I \cos \theta)^2R + (I \sin \theta - I_c)^2R \\ &= 2 (I \sin \theta)I_cR - I_c^2R \dots\dots\dots (2.11) \end{aligned}$$

2.8.2. Perbaikan Tegangan^[1]

Pemakaian kapasitor *shunt* dalam sistem tenaga listrik selain untuk perbaikan faktor daya juga bertujuan menaikkan tagangan. Dan secara vektoris dapat digambarkan sebagai berikut:



Gambar 2.11. Diagram vector pada rangkaian dengan *pf* (*lagging*) (a) dan (c) tanpa kapasitor *shunt*, (b) dan (d) dengan kapasitor *shunt*

Jatuh tegangan pada jaringan dengan *pf* (*lagging*) dapat dihitung sebagai berikut:

Sebelum pemasangan kapasitor:

$$\Delta V = IR \cos \varphi + IX_L \sin \varphi \text{ volt} \dots\dots\dots (2.12)$$

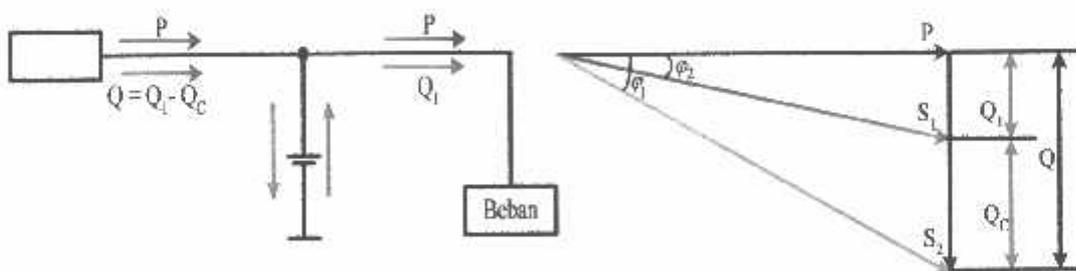
Bila kapasitor dipasang pada ujung penerima dari saluran, seperti yang terlihat pada gambar 2-11b, secara pendekatan jatuh tegangannya menjadi:

$$\delta V = IR \cos \varphi + (IX_L - I_C) \sin \varphi \quad \text{volt} \dots\dots\dots(2.13)$$

2.8.3. Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem

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

Dengan menambah kapasitor, daya reaktif (Q) akan berkurang, gambar 2.12. menunjukkan perbaikan faktor daya pada sistem, kapasitor mensuplai daya reaktif ke beban.



Gambar 2.12. Perbaikan Faktor Daya^[1]

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

$$\cos \varphi = \frac{P}{S_1} \dots\dots\dots(2.14)$$

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

$$\cos \varphi = \frac{P}{S_1}$$

$$\cos \varphi = \frac{P}{\sqrt{P^2 + Q_1^2}}$$

$$\cos \varphi = \frac{P}{\sqrt{P^2 + (Q_1 - Q_c)^2}} \dots\dots\dots(2.15)$$

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

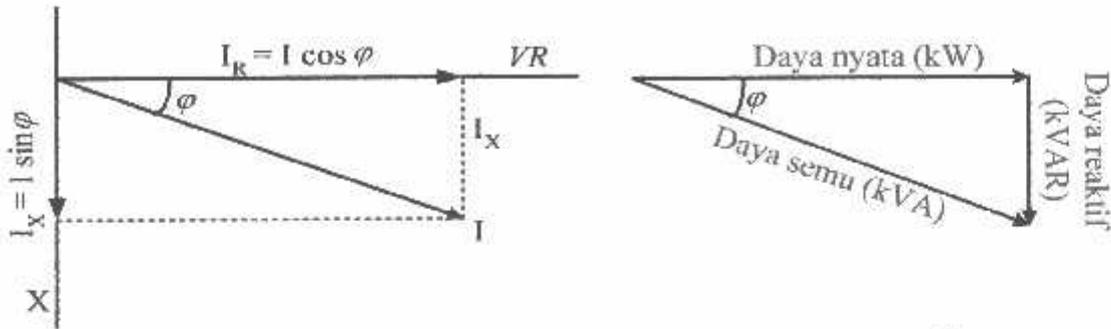
$$\% \text{ Rugi Daya} = 100 \left(\frac{\text{Faktor daya mula - mula } (\cos \varphi_1)}{\text{Faktor daya baru } (\cos \varphi_2)} \right) \dots\dots\dots(2.16)$$

% Pengurangan Rugi Daya

$$= 100 \left(1 - \left(\frac{\text{Faktor daya mula - mula } (\cos \varphi_1)}{\text{Faktor daya baru } (\cos \varphi_2)} \right)^2 \right) \dots\dots\dots(2.17)$$

2.8.4. Perhitungan Pengaruh Perbaikan Faktor Daya

Diagram *phase* dari dua komponen arus nyata itu arus aktif dan reaktif dapat dilihat pada gambar 2-13 berikut:



Gambar 2.13. Diagram Fasor dan Sudut Daya Beban Distribusi^[1]

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

$$\begin{aligned}
 I &= \text{Arus Semu} = \sqrt{(\text{ arus aktif})^2 + (\text{ arus reaktif})^2} \\
 &= \sqrt{(I \cos \varphi)^2 + (I \sin \varphi)^2} \dots\dots\dots(2.18)
 \end{aligned}$$

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

$$\begin{aligned}
 \text{Daya Semu (kVA)} &= \sqrt{(\text{Daya aktif})^2 + (\text{Daya reaktif})^2} \\
 (VI) &= \sqrt{(VI \cos \varphi)^2 + (VI \sin \varphi)^2} \dots\dots\dots(2.19)
 \end{aligned}$$

$$\text{Faktor Daya} = \frac{\text{Daya Aktif}}{\text{Daya Semu}} = \frac{kW}{kVA}$$

Daya Aktif = Daya Semu x Faktor Daya

kW = kVA x Faktor Daya

kW = kVA Cos φ(2.20)

2.8.5. Penentuan Rating Kapasitor untuk Perbaikan Faktor Daya Beban

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

$$\cos \varphi = \frac{\text{Daya Aktif}}{\text{Daya Semu}} = \frac{(kW)}{(kVA)} \dots\dots\dots (2.21)$$

$$\sin \varphi = \frac{\text{Daya Reaktif}}{\text{Daya Semu}} = \frac{(kVAR)}{(kVA)} \dots\dots\dots (2.22)$$

$$\tan \varphi = \frac{(kVAR)}{(kW)} \dots\dots\dots (2.23)$$

Karena Komponen daya aktif biasanya konstan, dan daya semu serta komponen daya reaktif berubah sesuai dengan faktor daya, maka persamaan yang dinyatakan dalam komponen daya aktif yang paling tepat digunakan, persamaan ini dapat ditulis sebagai berikut:

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

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

Dengan φ_1 = Sudut dari faktor daya mula-mula

φ_2 = Sudut dari faktor daya yang telah diperbaiki

Rating kapasitor yang dibutuhkan perbaikan factor daya:

$$\begin{aligned} \text{Daya reaktif (kVAR)} &= \text{Daya aktif} \times (\tan \varphi_1 - \varphi_2) \\ &= (kW) \times (\tan \varphi_1 - \varphi_2) \dots \dots \dots (2.24) \end{aligned}$$

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

$$\begin{aligned} \text{Daya reaktif (kVAR)} &= \text{Daya aktif} \times \Delta \tan \\ (\text{kVAR}) &= (kW) \times \Delta \tan \dots \dots \dots (2.25) \end{aligned}$$

2.9. Sistem Per-Unit^[3]

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

$$\text{Besaran per - unit} = \frac{\text{besaran sesungguhnya}}{\text{besaran dasar dengan dimensi yang sama}} \dots \dots \dots (2.26)$$

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

- Untuk data 1 fasa

Arus dasar:

$$I_d = \frac{\text{kVA dasar 1 fasa}}{\text{kV dasar L - N}} \dots \dots \dots (2.27)$$

Impedansi dasar:

$$Z_d = \frac{(\text{kV dasar L - N})^2 \times 1000}{\text{kVA dasar 1 fasa}} \dots\dots\dots (2.28)$$

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

Dalam persamaan diatas nilai-nilai besaran diberikan untuk rangkaian satu fasa.

Jadi tegangannya adalah tegangan antara fasa ke tanah dan daya setiap fasa.

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

$$Z = \frac{\text{Impedansi sebenarnya } Z (\Omega)}{\text{Impedansi dasar } Z_d} \dots\dots\dots (2.29)$$

BAB III

OPTIMASI PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL DENGAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO)

3.1. Analisa Aliran Daya Jaringan *Radial*^[2]

Sebelum melakukan optimasi dengan menggunakan metode *Binary Particle Swarm Optimization* (BPSO) dilakukan suatu proses analisa aliran daya untuk mengetahui kondisi suatu sistem distribusi *radial*.

3.1.1. Tujuan

Tujuan dari mempelajari analisa aliran daya pada skripsi ini adalah:

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

3.1.2. Metode *Newton Raphson*^[3]

Secara matematis persamaan aliran daya *Newton Raphson* dapat diselesaikan dengan menggunakan koordinat *rektanguler*, 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 simpul V_q pada suatu jaringan dengan n simpul dapat dituliskan:

$$I_p = \sum_{q=1}^n Y_{pq} V_q \dots\dots\dots(3.1)$$

Injeksi daya pada simpul p adalah:

$$S_p = P_p - jQ_p = V_p^* \cdot I_p \dots\dots\dots(3.2)$$

$$= V_p^* \sum_{q=1}^n Y_{pq} V_q \dots\dots\dots(3.3)$$

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 (3.3) dapat ditulis :

$$P_p - jQ_p = \sum_{q=1}^n |V_p| |V_q| |Y_{pq}| e^{-j(\delta_p - \delta_q + \theta_{pq})} \dots\dots\dots(3.4)$$

Dengan memisahkan bagian riil dan bagian imajiner maka diperoleh:

$$P_p = \sum_{q=1}^n |V_p| |V_q| |Y_{pq}| \cos(\delta_p - \delta_q + \theta_{pq}) \dots\dots\dots(3.5)$$

$$Q_p = \sum_{q=1}^n |V_p| |V_q| |Y_{pq}| \sin(\delta_p - \delta_q + \theta_{pq}) \dots\dots\dots(3.6)$$

Kedua persamaan diatas akan menghasilkan suatu kumpulan persamaan serempak (*simultan*) yang tidak linier untuk setiap simpul sistem tenaga listrik. Untuk mengetahui magnitude tegangan (V) dan sudut fasa (δ) disetiap simpul dapat

diselesaikan dengan menggunakan persamaan (3.5) dan (3.6) yang dilinierkan dengan metode *Newton Raphson* yang dapat dilihat dari persamaan dibawah ini:

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |V| \end{bmatrix} \dots\dots\dots(3.7)$$

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 dri 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* dan *diagonal* dari sub matriks *Jaquobian* yang dibentuk dengan mendefinisikan persamaan (3.5) dan (3.6),

dimana:

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

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

Persamaan (3.7) 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:

$$|V|^{k+1} = |V|^k + \Delta|V|^k \dots\dots\dots (3.8)$$

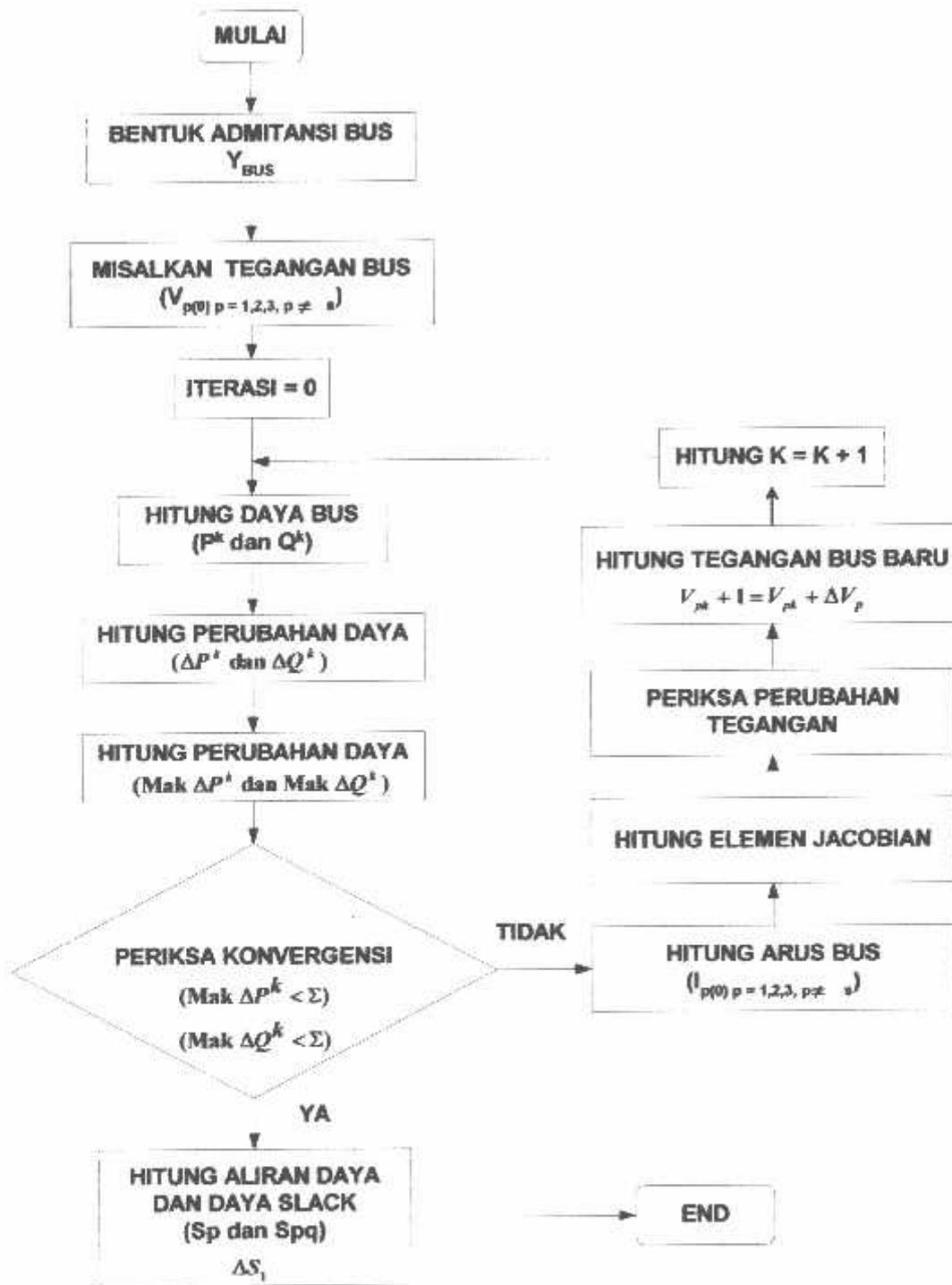
$$\delta^{k+1} = \delta^k + \Delta\delta^k \dots\dots\dots (3.9)$$

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.

3.1.3. Algoritma Aliran Daya *Newton Raphson*

1. Tentukan nilai $P_{p(\text{ditetapkan})}$ dan $Q_{p(\text{ditetapkan})}$ yang mengalir ke dalam sistem pada setiap rel untuk nilai yang ditentukan atau perkiraan dari besar dan sudut tegangan untuk itcrasi pertama atau tegangan yang ditentukan paling akhir untuk iterasi berikutnya.
 2. Hitung ΔP pada setiap rel.
 3. Hitung nilai-nilai matrik *Jacobian* dengan menggunakan nilai-nilai perkiraan atau yang ditentukan dari besar dan sudut tegangan dalam persamaan untuk turunan parsial yang ditentukan dengan differensiasi persamaan (3.5) dan (3.6).
 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 menambahkan $\Delta\delta_p$ dan $\Delta|V_q|$ pada nilai sebelumnya.
 6. Kembali ke langkah 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 telah dipilih.
-

3.1.4. Flowchart Algoritma Newton Raphson



Gambar 3.1. Flowchart Algoritma Newton Raphson

3.2. Penyelesaian Metode *Binary Particle Swarm Optimization* (BPSO)¹⁶⁾

Pada dasarnya *Binary Particle Swarm Optimization* (BPSO) adalah pengembangan dari metode PSO. Sebelum memasuki pembahasan tentang *Binary Particle Swarm Optimization* (BPSO) ada baiknya kita *mereview* kembali tentang PSO terlebih dahulu. PSO adalah suatu metode yang diperkenalkan oleh *Kennedy* dan *Elberhart* pada tahun 1995, yang metode ini diilhami dari perilaku sosial dari koloni burung atau ikan dan metode-metode dimana mereka menemukan tempat-tempat bertengger, sumber-sumber makanan atau habitat yang sesuai lainnya. Dalam PSO terdapat teori bahwa proses adaptasi terdiri dari 3 (tiga) prinsip utama yaitu mengevaluasi, membandingkan dan meniru. Jadi partikel dalam PSO akan mengvaluasi sekitarnya dan membandingkan dirinya dengan yang lain yang ada dalam koloni tersebut dan kemudian meniru dari sekitarnya yang paling kuat.

3.2.1. Parameter-parameter PSO

Elemen-elemen dasar dari teknik-teknik *Particle Swarm Optimization* (PSO) dinyatakan secara ringkas dan didefinisikan sebagai berikut:

1. Partikel (X)

Ini merupakan calon solusi yang disajikan oleh vektor dimensi d , dimana d adalah jumlah parameter yang dioptimalkan. Pada partikel X_i ke- i dapat dijelaskan seperti $X_i = [x_{i1}, x_{i2}, \dots, x_{id}]$.

2. Populasi, $pop(t)$

Sekumpulan partikel-partikel n dan dijelaskan sebagai $pop(t) = [X_1(t), X_2(t), \dots, X_n(t)]$.

3. *Swarm*

Populasi pergerakan partikel yang tidak diatur secara nyata yang cenderung berkelompok dan berdekatan bersama-sama walaupun masing-masing partikel nampaknya bergerak dalam arah acak.

4. Partikel Terbaik (*PB*)

Posisi terbaik sebelumnya untuk masing-masing partikel (posisi yang memberikan nilai kecocokan terbaik) dan dijelaskan sebagai $PB_i = [pb_{i1}, pb_{i2}, \dots, pb_{id}]$.

5. Kecepatan Partikel (*Particle Velocity*)

Kecepatan perubahan posisi atau pergerakan partikel-partikel yang disajikan dengan vektor dimensi d . Partikel V_i ke- i dapat dijelaskan seperti $V_i = [v_{i1}, v_{i2}, \dots, v_{id}]$.

6. Bobot *Inersia* (w)

Parameter kontrol yang digunakan untuk mengontrol dampak kecepatan sebelumnya pada kecepatan terkini. Semua variabel kontrol, posisi-posisi ketukan transformer dan kumpulan-kumpulan kapasitor *shunt* yang dapat diganti adalah variabel-variabel integer dan bukan merupakan variabel-variabel berkelanjutan. Oleh karena itu, nilai bobot inersia dipertimbangkan menjadi 0.8 dalam penelitian ini.

7. *Global* Terbaik (*GB*)

Posisi terbaik semua posisi terbaik individual yang dicapai sejauh ini dan dijelaskan sebagai $GB = [gb_1, gb_2, \dots, gb_d]$.

8. Kriteria Penghentian (*Stopping Criteria*)

Ini adalah kondisi dimana proses pencarian akan dihentikan. Dalam penelitian ini, pencarian akan berhenti apabila salah satu kriteria berikutnya dipenuhi, yaitu:

- a. Jumlah iterasi karena perubahan solusi terbaik terakhir adalah lebih besar dibandingkan dengan jumlah yang ditetapkan sebelumnya
- b. Jumlah iterasi mencapai jumlah maksimum yang diperkenankan

Dalam sebuah algoritma PSO, populasi memiliki partikel-partikel i yang menunjukkan calon solusi-solusi. Masing-masing partikel adalah vektor bernilai riil dimensi d , dimana d adalah jumlah parameter yang dioptimalkan. Oleh karena itu, masing-masing parameter yang dioptimalkan menyajikan sebuah dimensi bidang permasalahan.

3.2.2. *Binary Particle Swarm Optimization (BPSO)*

Dalam *Binary* sendiri menjelaskan bahwa kemungkinan dari suatu individu untuk mengambil keputusan dalam bilangan dengan 0 atau 1 dan ya atau tidak, salah atau benar, atau keputusan biner lainnya dan itu merupakan suatu fungsi dari pribadi dan factor sosial.

Parameter Vid , menggambarkan bahwa individu mempunyai kecenderungan membuat satu pilihan atau pilihan yang lain, dalam menentukan suatu kemungkinan. Jika Vid adalah lebih tinggi, individu akan lebih memungkinkan untuk memilih 1, dan menurunkan nilai-nilai kemungkinan untuk

memeilih nilai 0. Kemungkinan pilihan seperti itu dibatasi oleh nilai [0;1]. Fungsi Sigmoidal adalah suatu pilihan logis untuk melakukan ini. Sigmoidal berfungsi untuk membatasi nilai V_{id} dalam bidang [0;1].

Adapun persamaan-persamaannya adalah sebagai berikut:

Pada iterasi k , kecepatan untuk dimensi d dari partikel i diperbaharui dengan:

$$v_{id}^{k+1} = wv_{id}^k + c_1r_1(pb_{id}^k - x_{id}^k) + c_2r_2(gb_d^k - x_{id}^k) \dots\dots\dots (3.10)$$

Posisi partikel i diperbaharui dengan:

$$x_{id}^{k+1} = x_{id}^k + v_{id}^{k+1} \dots\dots\dots (3.11)$$

Untuk ruang pencarian diskrit biner, *Kennedy* dan *Eberhart* telah menyesuaikan PSO untuk mencari ruangan-ruangan biner, dengan menerapkan sebuah transformasi sigmoid dengan komponen kecepatan dengan persamaan (3.12) agar dapat menghentikan kecepatan-kecepatan dalam sebuah batasan nilai [0;1] dan mendesak nilai-nilai komponen dari lokasi-lokasi beberapa partikel yang akan menjadi 0 atau 1. Persamaan untuk memperbaharui posisi-posisi persamaan (3.11) kemudian diganti dengan persamaan (3.13).

$$v_{id}^k = \frac{1}{1 + e^{-v_{id}^k}} \dots\dots\dots (3.12)$$

$$x_{id}^k = \begin{cases} 1, & \text{if } \text{rand} < \text{sigmoid}(v_{id}^k) \\ 0, & \text{otherwise} \end{cases} \dots\dots\dots (3.13)$$

Di mana:

- w : bobot inersia
- i : ukuran populasi
- c_1, c_2 : konstanta percepatan

- r_1, r_2 : nilai acak dalam batasan $[0;1]$
 x_{id} : posisi saat ini dalam dimensi ke- i

3.3. Fungsi dan Tujuan (*Objective Functions*)

Tujuan dari penempatan kapasitor dalam sistem distribusi adalah untuk mengurangi kehilangan energi dan tingkat tegangan dalam batas yang diijinkan, serta meminimalkan biaya total pada rencana penempatan yang dipilih.

3.3.1. *Total Harmonic Distortion (THD)*^[5]

Salah satu tolok ukur kualitas daya listrik adalah level harmonisa (THD). THD menunjukkan nilai kumulatif frekuensi harmonik yang menjalar di sistem. Karena harmonik merupakan komponen arus atau tegangan yang merugikan, maka makin tinggi THD di suatu sistem makin jelek mutu kelistrikan sistem tersebut. Standart THD tegangan adalah 2%. Untuk mengatasi penjalaran harmonisa dapat dilakukan pemasangan filter di bus-bus yang memiliki harmonisa tinggi.

Pendefinisian rasio nilai *rms* komponen harmonisa kenilai *rms* komponen dasar biasanya dinyatakan dalam bentuk prosen. Indeks ini digunakan untuk mengukur deviasi bentuk gelombang periodik yang mengandung harmonisa dari gelombang sinus sempurna. Pada saat terjadi gelombang sinus sempurna, nilai THD adalah nol.

3.3.2. Meminimalkan Biaya Pengeluaran¹⁶¹

Seperti yang ditetapkan dalam persamaan berikut ini:

$$F = K^p P_{loss} + \sum_{j=1}^J K_j^c Q_j^c \dots\dots\dots(3.14)$$

Di mana:

- F : total fungsi biaya tahunan
- P_{loss} : total kerugian daya
- K_p : kerugian daya *per-unit*
- K_j^c : biaya tahunan kapasitor/kvar
- Q_j^c : ukuran kapasitor *shunt* yang ditempatkan pada bus j
- J, m : bus-bus kapasitor *shunt* dan jumlah bus

3.3.3. Meminimalkan Kehilangan Daya Nyata¹⁶¹

Tujuan ini diusahakan untuk meminimalkan biaya total dari kehilangan daya nyata yang muncul dari cabang-cabang saluran, seperti yang ditetapkan dalam batasan persamaan berikut ini:

$$P_{loss} = \sum_{h=1}^H \left(\sum_{i=0}^{m-1} P_{loss(i,i+1)}^h \right) \dots\dots\dots(3.15)$$

$$Q_{max}^c = l.Q_0^c \dots\dots\dots(3.16)$$

Di mana:

P_{loss} : total kerugian daya

L : integer

Q_0^c : ukuran kapasitor *shunt* terkecil

J, m : bus-bus kapasitor *shunt* dan jumlah bus

3.3.4. Algoritma Program Solusi *Binary Particle Swarm Optimization* (BPSO)

1. Masukkan data parameter PSO
2. Inisialisasi partikel
3. Perbaharui kecepatan dengan menggunakan persamaan

$$v_{id}^{k+1} = wv_{id}^k + c_1r_1(pb_{id}^k - x_{id}^k) + c_2r_2(gb_d^k - x_{id}^k)$$

4. Perbaharui posisi dengan menggunakan fungsi sigmoid dari Persamaan

$$v_{id}^k = \frac{1}{1 + e^{-v_{id}^k}}$$

$$x_{id}^k = \begin{cases} 1, & \text{if } rand < sigmoid(v_{id}^k) \\ 0, & \text{otherwise} \end{cases}$$

5. Apakah jumlah iterasi kurang dari jumlah iterasi maksimum
 - a. “Ya” ganti nilai partikel sebelumnya dengan nilai partikel baru
 - b. “Tidak” ganti nilai partikel sebelumnya dengan nilai partikel baru
 6. Apakah jumlah iterasi kurang dari jumlah iterasi maksimum
 - a. “Ya” ganti nilai partikel sebelumnya dengan nilai partikel baru
 - b. “Tidak” ganti nilai partikel sebelumnya dengan nilai partikel baru
-

7. Apakah iterasi sudah maksimal
 - a. “Ya” cetak hasil dan proses telah selesai
 - b. “Tidak” tambah iterasi dengan satu dan kembali kelangkah no. 3
8. Stop

3.3.5. Algoritma *Objective Functions*

1. Memasukkan data saluran, data pembebanan dan kapasitas kapasitor
2. Menghitung *Decoding* ke data beban *load flow*
3. Menghitung *load flow* dengan menggunakan metode *Newton Raphson*
4. Masukkan THD
5. Menghitung *Objective Functions*:

$$F = K^p P_{loss} + \sum_{j=1}^J K_j^c Q_j^c$$

Dengan batasan-batasan:

$$P_{loss} = \sum_{h=1}^H \left(\sum_{i=0}^{m-1} P_{loss(i,j+1)}^h \right)$$

$$Q_{max}^c = LQ_0^c$$

6. Kembali ke PSO
 7. Stop
-

7. Apakah iterasi sudah maksimal
 - a. “Ya” cetak hasil dan proses telah selesai
 - b. “Tidak” tambah iterasi dengan satu dan kembali ke langkah no. 3
8. Stop

3.3.5. Algoritma *Objective Functions*

1. Memasukkan data saluran dan data pembebanan
2. Menghitung *Decoding* ke data beban *load flow*
3. Menghitung *load flow* dengan menggunakan metode *Newton Raphson*
4. Masukkan THD
5. Menghitung *Objective Functions*:

$$F = K^p P_{loss} + \sum_{j=1}^J K_j^c Q_j^c$$

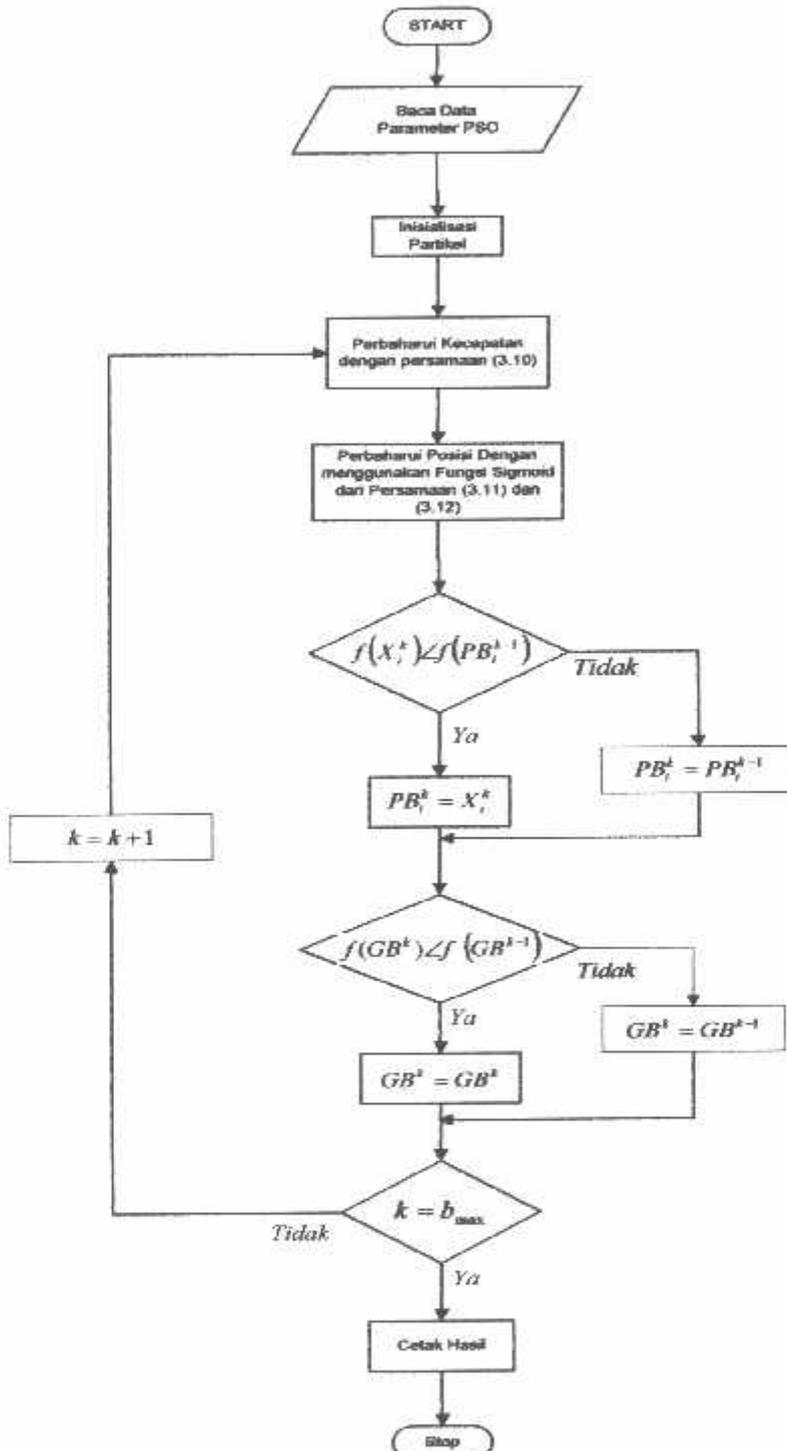
Dengan batasan-batasan:

$$P_{loss} = \sum_{h=1}^H \left(\sum_{i=0}^{m-1} P_{load}^h(i,j+1) \right)$$

$$Q_{max}^c = LQ_0^c$$

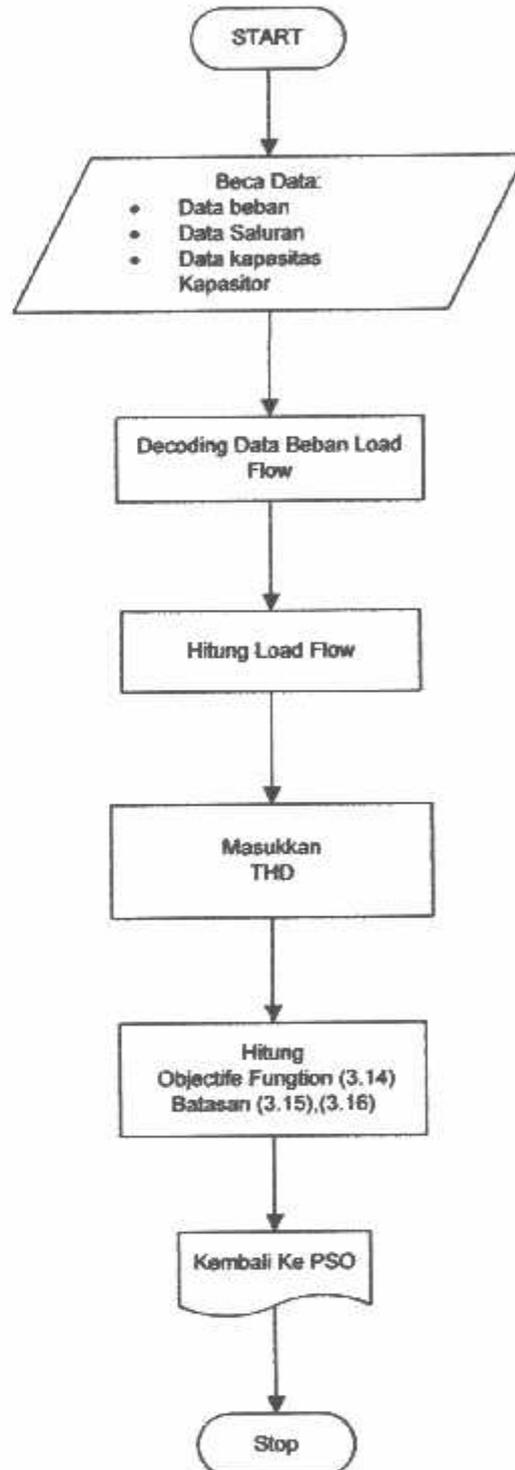
6. Kembali ke PSO
 7. Stop
-

3.3.6. *Flowchart* Algoritma Solusi *Optimal Power Flow* dengan Metode *Binary Particle Swarm Optimization (BPSO)*



Gambar 3.2. *Flowchart* Algoritma Solusi *Binary Particle Swarm Optimization (BPSO)*

3.3.7. Flowchart Objective Functions



Gambar 3.3. Flowchart Objective Functions

3.4. Data Saluran

Jaringan distribusi Penyulang Syiar menggunakan kabel saluran udara dengan spesifikasi yang seperti pada tabel 3-1.

Tabel 3-1
Spesifikasi Saluran^[1]

Jenis Konduktor	Penampang nominal (mm ²)	Impedansi saluran (Ω/km)
AAAC	150	0.2162 + j 0.3305
AAAC	120	0.2688 + j 0.3376

Dari data spesifikasi saluran pada penyulang Syiar diketahui bahwa jenis konduktor yang digunakan adalah AAAC (*all-aluminium-alloyconductors*) dengan penampang nominal 150 mm² untuk saluran utama dan 120 mm² untuk saluran cabang, sedangkan impedansi saluran 0.2162 + j0.3305 Ω/km dan 0.2688 + j0.3376 Ω/km.

Untuk perhitungan Impedansi saluran diberikan contoh pada bus dari 1 ke 2 dan dari 8 ke 9 sebagai berikut:

Jarak antara saluran dari 1 ke 2 dengan panjang = 2120 m = 2.12 km

$$R = 2.12 \text{ km} \times 0.2162 \text{ } \Omega/\text{km} = 0.45834 \text{ } \Omega$$

$$X = 2.12 \text{ km} \times 0.3305 \text{ } \Omega/\text{km} = 0.70066 \text{ } \Omega$$

Jarak antara saluran dari 8 ke 9 dengan panjang = 780 m = 0,78 km

$$R = 0.78 \text{ km} \times 0.2688 \text{ } \Omega/\text{km} = 0.20966 \text{ } \Omega$$

$$X = 0.78 \text{ km} \times 0.3376 \text{ } \Omega/\text{km} = 0.26333 \text{ } \Omega$$

Dengan cara yang sama seperti pada contoh, untuk saluran yang lainnya maka diperoleh hasil seperti pada tabel 3-2.

Tabel 3-2
Data Saluran Penyulang Syiar

Saluran		Panjang (m)	Impedensi Saluran	
Dari	Ke		R(Ω)	X(Ω)
1	2	2120	0.45834	0.70066
2	3	1150	0.24863	0.38008
3	4	750	0.16215	0.24788
4	5	680	0.14702	0.22474
5	6	430	0.09297	0.14212
5	7	570	0.12323	0.18839
3	8	320	0.06918	0.10576
8	9	780	0.20966	0.26333
8	10	1890	0.40862	0.62465
10	11	2350	0.50807	0.77668
11	12	1260	0.27241	0.41643
12	13	1740	0.37619	0.57507
13	14	930	0.20107	0.30737
14	15	2490	0.66931	0.84062
14	16	1960	0.42375	0.64778
16	17	1480	0.31998	0.48914
17	18	2490	0.53834	0.82295
18	19	430	0.09297	0.14212
19	20	1600	0.43008	0.54016
19	21	510	0.11026	0.16856
21	22	920	0.19890	0.30406
21	23	1320	0.28538	0.43626
23	24	2460	0.66125	0.83050
23	25	1270	0.34138	0.42875
18	26	1120	0.24214	0.37016
26	27	1670	0.36105	0.55194
27	28	2590	0.69619	0.87438
27	29	2850	0.61617	0.94193
29	30	100	0.02162	0.03305
30	31	120	0.02594	0.03966
31	32	250	0.05405	0.08263
32	33	1450	0.31349	0.47923
33	34	2610	0.56428	0.86261
32	35	720	0.19354	0.24307
30	36	233	0.05037	0.07701
36	37	820	0.22042	0.27683
36	38	130	0.02811	0.04297
38	39	590	0.12756	0.19500
39	40	1090	0.23566	0.36025
40	41	1100	0.29568	0.37136
41	42	1070	0.28762	0.36123
40	43	950	0.20539	0.31398
43	44	790	0.17080	0.26110

44	45	750	0.16215	0.24788
44	46	1460	0.39245	0.49290
43	47	2210	0.47780	0.73041
47	48	2250	0.48645	0.74363
48	49	2500	0.54050	0.82625

3.5. Data Pembebanan

Pembebanan diperoleh dengan mengambil data dari masing-masing trafo distribusi, dimana besarnya beban pada masing-masing fasa diasumsikan seimbang. Jika besarnya pembebanan adalah nol, maka pada bus tidak terdapat trafo distribusi tetapi hanya merupakan simpul. Pada tahap ini rugi-rugi yang terjadi pada trafo distribusi diabaikan, dengan mengasumsikan factor daya 0,85.

Dibawah ini diberikan contoh perhitungan pembebanan pada bus 2:

$$\begin{aligned}
 \text{Kapasitas Trafo} &= 150 \text{ kVA} \\
 \text{Beban Gardu} &= 57,58\% \\
 \text{Factor Daya} &= \cos \theta = 0,85 \\
 &\quad \sin \theta = 0,5268 \\
 \text{Pembebanan} &= 150 \times 57,58\% = 86,37 \text{ kVA} \\
 \text{Beban Aktif} &= 86,37 \times \cos \theta = 86,37 \times 0,85 = 73,4145 \text{ kW} \\
 \text{Beban Reaktif} &= 86,37 \times \sin \theta = 86,37 \times 0,5268 = 45,4982 \text{ kVAR}
 \end{aligned}$$

Untuk selanjutnya, dapat dilihat seperti pada tabel 3-3 dibawah ini:

Tabel 3-3

Data Pembebanan Sistem 20kV Penyulang Syiar

No Node	Kode Trafo	Lokasi	Pembebanan			Type Bus
			S (kVA)	P (kW)	Q (kVAR)	
1	-	-	0	0	0	Slack
2	T.71	Mojokopek	86.37	73.4145	45.4982	Load
3	-	-	0	0	0	Load
4	T.69	Mojoparon	85.68	72.8280	45.1347	Load
5	-	-	0	0	0	Load
6	T.90	PT. Metabisulfid	555	471.7500	292.3644	Load

7	T.203	Ds. Blawi	121.01	102.8585	63.7460	Load
8	-	-	0	0	0	Load
9	T.26	Ds. Pekoren	106	90.1000	55.8390	Load
10	T.130	Ds. Pekoren	117.31	99.7135	61.7969	Load
11	T.202	Ds. Rembang	100.92	85.7820	53.1629	Load
12	T.70	Ds. Ketapaan	72.28	61.4380	38.0759	Load
13	T.199	Ds. Kdg Banteng	95.61	81.2685	50.3657	Load
14	-	-	0	0	0	Load
15	T.200	Ds. Kdg Banteng	29.79	25.3215	15.6929	Load
16	T.251	Ds. Cengkelek	25.63	21.7855	13.5014	Load
17	T.252	Ds. Candiroboh	31.87	27.0895	16.7886	Load
18	-	-	0	0	0	Load
19	-	-	0	0	0	Load
20	T.144	Pejangkungan	66.05	56.1425	34.7940	Load
21	-	-	0	0	0	Load
22	T.125	Sbr. Glagah	43.18	36.7030	22.7465	Load
23	-	-	0	0.0000	0.0000	Load
24	T.182	Roh Wangkal	26.56	22.5760	13.9913	Load
25	T.246	Ds. Oyoran	30.71	26.1035	16.1775	Load
26	T.41	Ds. Rembang	75.05	63.7925	39.5350	Load
27	-	-	0	0.0000	0.0000	Load
28	T.186	Oro-Oro Bulu	80.13	68.1105	42.2111	Load
29	T.134	Kalisat	69.05	58.6925	36.3743	Load
30	-	-	0	0.0000	0.0000	Load
31	T.135	Ds. Kalisat	39.26	33.3710	20.6815	Load
32	-	-	0	0.0000	0.0000	Load
33	T.301	Pohkpan/Sisipan	57.27	48.6795	30.1688	Load
34	T.227	PT. CharunPohkpan	345	293.2500	181.7400	Load
35	T.25	Ds. Kdg Lingit	52.65	44.7525	27.7351	Load
36	-	-	0	0.0000	0.0000	Load
37	T.245	Pejaten	71.59	60.8515	37.7124	Load
38	T.154	Tampung Barat	74.36	63.2060	39.1716	Load
39	T.94	Ds. Tampung	149.41	126.9985	78.7066	Load
40	-	-	0	0.0000	0.0000	Load
41	T.95	Ds. Kanigoro	65.59	55.7515	34.5517	Load
42	T.145	DS. Krenggih	77.13	65.5605	40.6307	Load
43	-	-	0	0.0000	0.0000	Load
44	-	-	0	0.0000	0.0000	Load
45	T.239	Dinas Pertanian	450	382.5000	237.0522	Load
46	T.185	Ds. Slambrit	54.5	46.3250	28.7097	Load
47	T.117	Ds. Syiar	48.73	41.4205	25.6701	Load
48	T.86	Genenganwaru	96.76	82.2460	50.9715	Load
49	T.87	Ds. Wangkal	31.64	26.8940	16.6674	Load

BAB IV

OPTIMASI PENENTUAN LETAK, JUMLAH DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV TIPE RADIAL PENYULANG SYIAR, PASURUAN

4.1. Aplikasi Program Komputer

Program komputer dalam skripsi ini dijalankan dengan menggunakan bahasa pemrograman *Borland Delphi 7.0* dengan sistem operasi *Windows XP Profesional* dan diaplikasikan pada komputer *Intel Pentium 4 CPU 1.80 GHz* dengan memori 128 Mb.

4.2. Uji Validasi Program Komputer

Sebelum melakukan penentuan letak, jumlah dan kapasitas kapasitor dengan menggunakan data lapangan, maka perlu dilakukan uji validasi program komputer yang telah dibuat. Jika program yang dibuat telah mampu menentukan letak, jumlah dan kapasitas kapasitor seperti data yang terdapat pada jurnal, maka program komputer tersebut dinyatakan sudah benar dan teruji kebenarannya.

4.2.1. Validasi Data IEEE 9 Bus^[6]

Merupakan data yang diambil dari jurnal IEEE, yang terdiri dari data saluran dan pembebanan, seperti yang ditunjukkan pada tabel 4.1 dan tabel 4.2, yaitu:

1. Tabel Data Saluran.

Tabel 4.1.
Data Saluran IEEE-9 Bus

Saluran		Impedansi Saluran	
Dari	Ke	R (Ω)	X (Ω)
0	1	0.1233	0.4127
1	2	0.014	1.6051
2	3	0.7463	1.605
3	4	0.6984	1.6084
4	5	1.9831	0.8276
5	6	0.9053	0.7886
6	7	1.0552	0.864
7	8	1.7953	0.616
8	9	1.3434	0.7264

2. Tabel Data Beban.

Tabel 4.2.
Data Beban IEEE-9 Bus

No Bus	P (kW)	Q (kvar)
1	1640	360
2	980	240
3	1390	446
4	1598	740
5	1410	600
6	780	110
7	1100	60
8	980	130
9	1540	150

Berikut adalah tampilan uji validasi yang terdiri dari *Load flow* dan hasil program dengan data IEEE 9 Bus:

1. Tampilan *Load Flow*

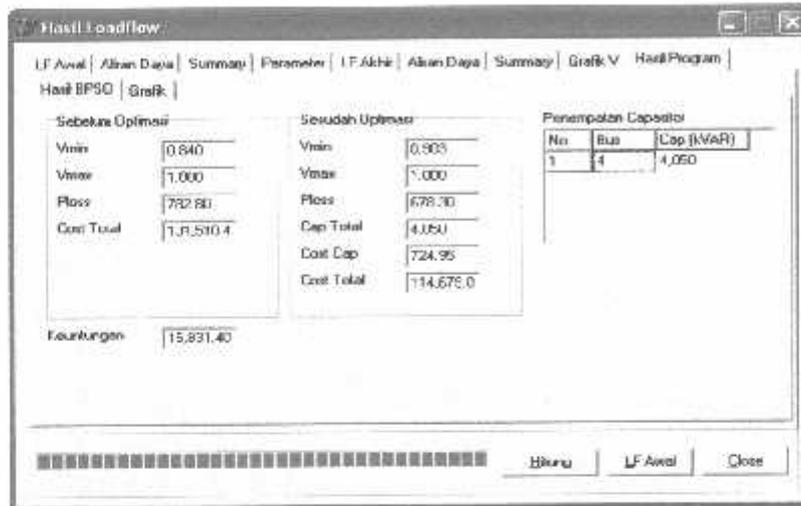
Bus	absV (pu)	sudV (deg)	Pg (kW)	Qg (kVAR)	PL (kW)	QL (kVAR)	Sups (pu)	Type Bus
1	1.00000	0.00000	12200.000	4465.024	0.000	0.000	0.000	1
2	0.98163	-0.84777	0.000	0.000	1640.000	360.000	0.000	3
3	0.97653	3.13580	0.000	0.000	960.000	240.000	0.000	3
4	0.94265	-5.13993	0.000	0.000	1390.000	446.000	0.000	3
5	0.92165	-6.97328	0.000	0.000	1538.000	740.000	0.000	3
6	0.89547	-7.43581	0.000	0.000	1410.000	600.000	0.000	3
7	0.87200	-8.00831	0.000	0.000	780.000	110.000	0.000	3
8	0.86040	-8.54111	0.000	0.000	1130.000	60.000	0.000	3
9	0.84645	-8.74676	0.000	0.000	960.000	130.000	0.000	3
10	0.83997	-8.95155	0.000	0.000	1540.000	150.000	0.000	3

Gambar 4.1. Tampilan Data *Load Flow* awal

Bus	absV (pu)	sudV (deg)	Pg (kW)	Qg (kVAR)	PL (kW)	QL (kVAR)	Sups (pu)	Type Bus
1	1.00000	0.00000	12096.298	180.865	0.000	0.000	0.000	1
2	0.98615	-0.71447	0.000	0.000	1640.000	360.000	0.000	3
3	0.95796	-3.12386	0.000	0.000	960.000	240.000	0.000	3
4	0.98515	5.43741	0.000	0.000	1390.000	446.000	0.000	3
5	0.97392	-7.50453	0.000	4050.000	1538.000	740.000	0.000	3
6	0.94548	-7.81478	0.000	0.000	1410.000	600.000	0.000	3
7	0.93391	-8.41472	0.000	0.000	780.000	110.000	0.000	3
8	0.92215	-8.37080	0.000	0.000	1130.000	60.000	0.000	3
9	0.90918	-9.35761	0.000	0.000	960.000	130.000	0.000	3
10	0.90315	-9.21760	0.000	0.000	1540.000	150.000	0.000	3

Gambar 4.2. Tampilan Data *Load Flow* akhir

2. Tampilan Hasil Program



Gambar 4.3. Tampilan Hasil

4.2.2. Prosentase Tingkat Kesalahan (*Error*)

Dari hasil uji validasi didapatkan *Ploss* setelah optimasi sebesar 678.30 kW, sedangkan *Ploss* pada jurnal sebesar 698.78 kW. Sehingga perhitungan tingkat kesalahan sebagai berikut:

$$\% \text{ error} = \frac{\text{Hasil Jurnal} - \text{Hasil Pengujian}}{\text{Hasil Jurnal}} \times 100\%$$

$$\% \text{ error} = \frac{698.78 - 678.30}{698.78} \times 100\%$$

$$\% \text{ error} = 0.02930\%$$

Dari proses validasi di atas maka didapatkan *error* sebesar 0.02930%

4.3. Sistem Distribusi Tenaga listrik 20 kV GI PIER Penyulang Syiar

Dalam perhitungan aliran daya Data yang diambil dari GI. PIER Pasuruan yang melayani 6 (enam) buah penyulang.

Namun pada sekripsi ini hanya menganalisa satu penyulang saja, yaitu pada penyulang Syiar. Sistem Distribusi Radial GI. PIER yang memakai tegangan distribusi 20 kV. Untuk menyelesaikan perhitungan Aliran Daya terlebih dahulu ditetapkan *single line diagram* sesuai dengan gambar 4.4. yang akan dianalisis, dimana dasar yang digunakan:

- Tegangan Dasar : 20 kV
- Daya Dasar : 100 kVA

Selanjutnya *bus-bus* yang diklasifikasikan, yaitu busbar GI. PIER diasumsikan sebagai *slack bus*, sedangkan *bus-bus* yang lain sepanjang saluran Radial sebagai *load bus*. Dalam hal ini tidak ada *bus* generator karena sepanjang saluran tidak terdapat pembangkitan.

4.4. Analisa Perhitungan

Perhitungan penempatan kapasitor diawali dengan melakukan studi aliran daya dengan menggunakan metode *Newton Raphson*. Studi aliran daya dilakukan untuk mengetahui harga tegangan dan sudut fasa tiap-tiap bus, arus yang mengalir pada saluran, dan aliran daya tiap saluran. Setelah studi aliran daya dilakukan, barulah dilakukan perhitungan rugi-rugi daya pada saluran.

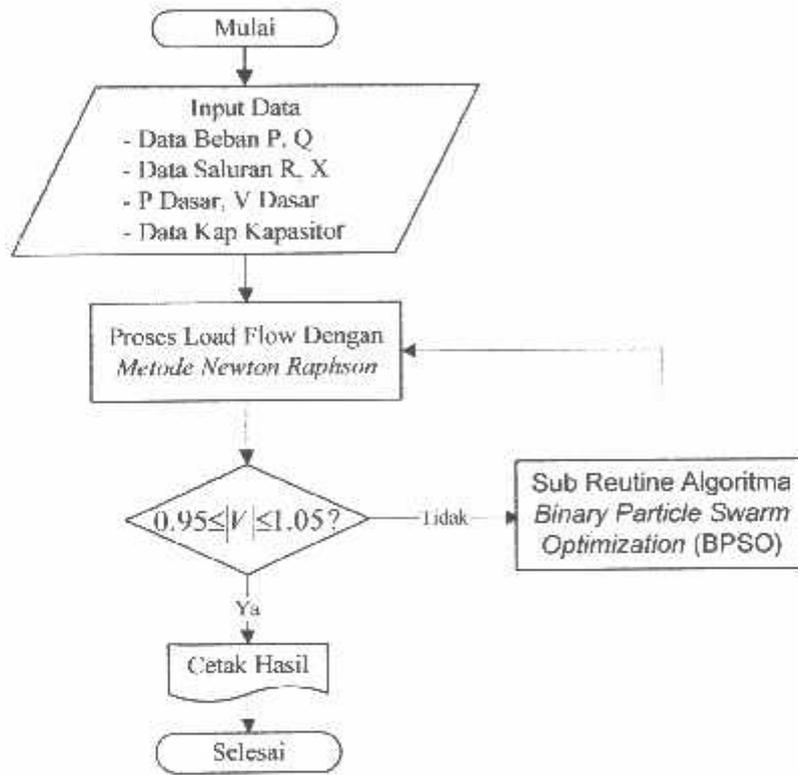
Untuk menentukan lokasi dan kapasitas kapasitor terlebih dahulu menentukan rugi daya saluran terbesar yang digunakan sebagai input untuk menjalankan metode *Binary Particle Swarm Optimization* (BPSO) agar kapasitor dapat diletakan pada lokasi yang memberikan profil tegangan yang paling optimum. Sedangkan untuk pencarian grafik yang optimal secara random ditentukan oleh program komputer demikian pula untuk perhitungan besar nilai penghematan.

Untuk memudahkan perhitungan dan analisa pada sistem tenaga, biasanya dipakai harga-harga dalam *per-unit*. Harga persatuan adalah harga sebenarnya dibagi dengan harga dasar. Harga yang dipilih pada analisa ini adalah 20 kV dan 100 MVA sebagai harga tegangan dasar dan daya dasar. Mengingat bahwa pada jaringan tidak dilakukan pengukuran faktor daya maka pada perhitungan ini mengassumsikan nilai faktor daya sebesar 0.85.

Perhitungan diawali dengan menampilkan *single line diagram* dari penyulang yang mewakili keadaan sistem yang sesungguhnya. Dari gambar 4-4 diperlihatkan *single line diagram* dari penyulang Syiar. Pada penyulang ini jumlah bus dan jumlah saluran masing-masing adalah:

- *Slack bus* = 1
- *Load bus* = 49
- Jumlah saluran = 48

⊕ *Flowchart* Pemecahan Masalah

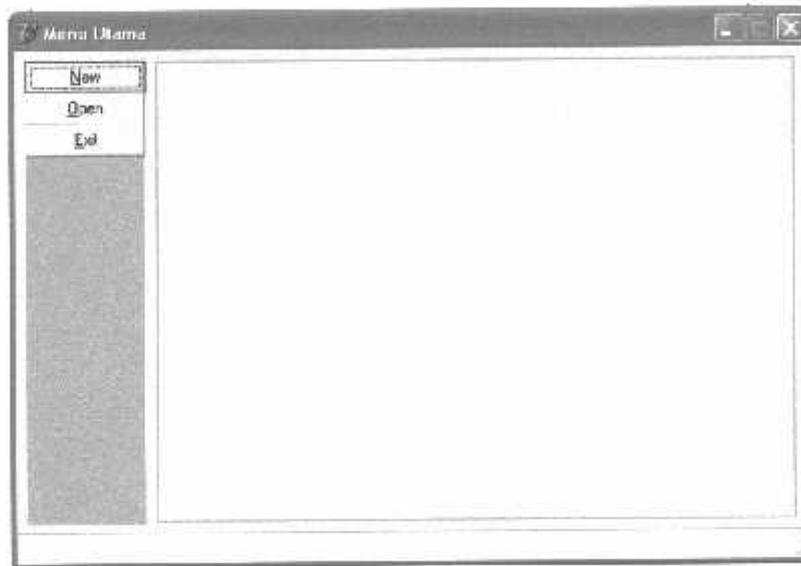


Gambar 4.5. *Flowchart* Pemecahan Masalah

4.5. Hasil Perhitungan Program BPSO

Hasil perhitungan Program BPSO sebagai berikut:

1. Tampilan Utama Program



Gambar 4-6. Tampilan Utama Program

Pada tampilan ini terdapat beberapa *fitur*, yaitu:

- ⊕ Tekan tombol *New* untuk memasukkan data baru
- ⊕ Tekan *Open* untuk data yang tersimpan
- ⊕ Tekan *Exit* untuk keluar dari program

2. Pada tampilan ini terdapat beberapa *fitur*, yaitu:

- ⊕ Imputan data yang akan dimasukkan dalam proses **LF awal**, yang tampilan-tampilanya dapat dilihat pada gambar-gambar di bawah ini, yang terdiri dari:

The screenshot shows a software window titled "Tampilan Data" with a tabbed interface. The "General" tab is active, displaying several input fields:

- Jumlah Bus: 49
- Jumlah Saluran: 48
- Tegangan Dasar: 20 (with a dropdown menu showing "kV")
- Daya Dasar: 100 (with a dropdown menu showing "kVA")
- Parameter Saluran: ohm (with a dropdown menu)

 At the bottom right of the window, there are "Next" and "Close" buttons.

Gambar 4-7. Tampilan Inputan Data (*General*)

The screenshot shows the "Data Bus" tab of the "Tampilan Data" window. It displays a table with the following columns: Bus, absV (pu), surV (deg), Pg (kW), Qg (KVAR), PL (kW), QL (KVAR), Cap (ohm), and Type Bus. The table contains 15 rows of data, with the first row highlighted.

Bus	absV (pu)	surV (deg)	Pg (kW)	Qg (KVAR)	PL (kW)	QL (KVAR)	Cap (ohm)	Type Bus
1	0	0	0	0	0	0	0	1
2	1	0	0	0	73.4145	45.4962	0	3
3	1	0	0	0	0	0	0	3
4	1	0	0	0	72.828	45.1347	0	3
5	1	0	0	0	0	0	0	3
6	1	0	0	0	471.75	282.3543	0	3
7	1	0	0	0	102.8688	63.7459	0	3
8	1	0	0	0	0	0	0	3
9	1	0	0	0	30.1	19.8389	0	3
10	1	0	0	0	88.7135	51.7960	0	3
11	1	0	0	0	65.782	53.1674	0	3
12	1	0	0	0	61.438	38.0759	0	3
13	1	0	0	0	81.2685	50.3656	0	3
14	1	0	0	0	0	0	0	3
15	1	0	0	0	0	0	0	3

Untuk bus selanjutnya dapat dilihat pada lampiran tabel 1

Gambar 4-8. Tampilan Inputan Data (Data Pembebanan)

No	Dari	Ke	R (ohm)	X (ohm)	Lc (ohm)	Tr	Tu	Su (deg)	Kap (kVA)
1	1	2	0.45034	0.70006	0	0	0	0	10000
2	2	3	0.24853	0.30000	0	0	0	0	10000
3	3	4	0.16215	0.24788	0	0	0	0	10000
4	4	5	0.14702	0.22474	0	0	0	0	10000
5	5	6	0.03297	0.14212	0	0	0	0	10000
6	6	7	0.12373	0.10838	0	0	0	0	10000
7	3	6	0.06710	0.10576	0	0	0	0	10000
8	8	9	0.20866	0.26333	0	0	0	0	10000
9	8	10	0.40862	0.62465	0	0	0	0	10000
10	10	11	0.50007	0.77668	0	0	0	0	10000
11	11	12	0.27241	0.41643	0	0	0	0	10000
12	12	13	0.37619	0.57507	0	0	0	0	10000
13	13	14	0.29102	0.30737	0	0	0	0	10000
14	14	15	0.02931	0.02402	0	0	0	0	10000

Untuk saluran selanjutnya dapat dilihat pada lampiran tabel 2
 Gambar 4-9. Tampilan Inputan Data (Data Saluran)

3. Pada tampilan ini terdapat beberapa *fitur*, yaitu:

- ⊕ Hasil dari **LF Awal** yang masing-masing tampilan dapat dilihat pada gambar-gambar di bawah ini, dengan cara:
 - Tekan Tombol **Next** kemudian tekan tombol **LF Awal** untuk melihat hasil perhitungan aliran daya *Newton Raphson* sebelum optimasi.

Bus	abv (pu)	vudV (kV)	Pg (kW)	Qg (kVAr)	PL (kW)	GL (kVAr)	Sus (lb)	Type Bus	
1	1.00000	0.00000	2994.035	1925.289	0.000	0.000	0.000	1	
2	0.99320	-0.17529	0.000	0.000	73.415	45.458	0.000	3	
3	0.98867	-0.24354	0.000	0.000	0.000	0.000	0.000	3	
4	0.99909	0.28251	0.000	0.000	72.828	45.175	0.000	3	
5	0.98867	-0.29415	0.000	0.000	0.000	0.000	0.000	3	
6	0.98846	-0.30000	0.000	0.000	471.750	292.364	0.000	3	
7	0.96961	-0.29504	0.000	0.000	102.869	63.746	0.000	3	
8	0.98880	-0.28916	0.000	0.000	0.000	0.000	0.000	3	
9	0.98874	-0.29062	0.000	0.000	90.700	55.839	0.000	3	
10	0.90441	-0.43428	0.000	0.000	0.000	95.713	61.797	0.000	3
11	0.97917	-0.54189	0.000	0.000	85.782	53.163	0.000	3	
12	0.97648	-0.61324	0.000	0.000	61.438	38.046	0.000	3	
13	0.97299	-0.70910	0.000	0.000	61.269	50.366	0.000	3	
14	0.97135	-0.75837	0.000	0.000	0.000	0.000	0.000	3	
15	0.97087	-0.76001	0.000	0.000	25.322	15.633	0.000	3	

Untuk bus selanjutnya dapat dilihat pada lampiran tabel 3
 Gambar 4-10. Tampilan Hasil Aliran Daya Untuk Mengetahui Profil Tegangan Dan Sudut Fasa Tiap-tiap Bus Sebelum Optimasi

No	Dari	Ke	P (kW)	Q (kVAR)	Arus re (A)	Arus im (A)	Dasi	Ke	P (kW)	Q (kVAR)
1	1	2	2994.035	1825.289	149.702	96.204	2	1	-2979.518	-1
2	2	3	2906.102	1857.590	146.013	93.963	3	2	-2898.606	-1
3	3	4	647.527	401.395	32.641	20.464	4	3	-647.666	-4
4	4	5	574.858	356.482	28.971	18.164	5	4	-574.696	-3
5	5	6	471.323	292.476	23.785	14.914	6	5	-471.770	-2
6	5	7	102.303	63.753	5.195	3.251	7	5	102.650	-4
7	3	8	2250.679	1444.142	113.372	73.498	8	3	-2243.416	-1
8	8	9	99.106	55.846	4.542	2.847	9	8	-99.100	-4
9	8	10	2159.310	1384.365	108.830	70.852	10	8	-2152.431	-1
10	10	11	2052.717	1314.052	103.780	67.477	11	10	-2044.931	-1
11	11	12	1969.149	1248.988	99.433	64.721	12	11	-1965.315	-1
12	12	13	1893.977	1205.049	95.309	62.738	13	12	-1889.907	-1
13	13	14	1807.638	1147.086	92.164	60.088	14	13	-1805.704	-1
14	14	15	75.723	45.695	1.293	0.825	15	14	-75.327	-1

Untuk saluran Selanjutnya dapat dilihat pada lampiran tabel 4

Gambar 4-11. Tampilan Hasil Aliran Daya Untuk Arus yang Mengalir pada Saluran dan Daya tiap Saluran Sebelum Optimasi

Summary Loadflow		
Jumlah Pembangkitan	2994.035 + j 1825.289	kVA
Jumlah Pembebanan	2917.275 + j 1807.964	kVA
Jumlah Rugi-Rugi	76.759 + j 117.325	kVA
Iterasi	3	
Waktu Hlang	0 0 31	

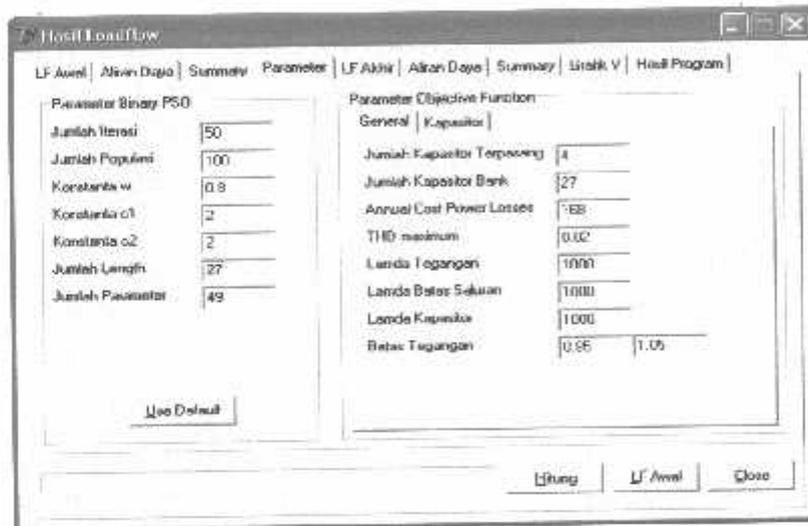
Gambar 4-12. Tampilan Total Pembangkitan, Pembebanan, dan Rugi-rugi Sebelum Optimasi

4. Pada tampilan ini terdapat beberapa *fitur*, yaitu:

- ⊕ Parameter *Binary Particle Swarm Optimization* (BPSO)
- ⊕ Parameter *Objective Function*,

Yang masing-masing dapat dilihat dengan cara:

- Tekan tombol **Parameter** dan *Use Default*



Gambar 4-13. Tampilan Parameter Yang Digunakan

5. Pada tampilan ini terdapat beberapa *fitur*, yaitu:

- ⊕ Hasil perhitungan Aliran Daya dengan *Metode Newton Raphson* setelah optimasi, yang masing-masing tampilan dapat dilihat pada gambar-gambar di bawah ini, dengan cara:

- Tekan tombol **Hitung**

Bus	absV (pu)	sudV (deg)	Pg (kW)	Qg (KVAR)	PL (kW)	QL (KVAR)	Supr (pu)	Type Bus
1	1.00000	0.00000	2971.505	390.947	0.000	0.000	0.000	1
2	0.99592	-0.27368	0.000	0.000	73.415	45.498	0.000	3
3	0.99581	-0.42067	0.000	0.000	0.000	0.000	0.000	3
4	0.99329	-0.43452	0.000	0.000	72.828	45.135	0.000	3
5	0.99288	0.44567	0.000	0.000	0.000	0.000	0.000	3
6	0.99266	-0.45146	0.000	0.000	471.750	292.364	0.000	3
7	0.99282	-0.44734	0.000	0.000	102.859	63.746	0.000	3
8	0.99344	-0.45576	0.000	0.000	0.000	0.000	0.000	3
9	0.99336	-0.45751	0.000	0.000	90.100	55.839	0.000	3
10	0.99140	-0.65867	0.000	0.000	90.710	61.797	0.000	3
11	0.98826	-0.90530	0.000	0.000	85.792	53.163	0.000	3
12	0.99020	1.03495	0.000	0.000	61.438	38.076	0.000	3
13	0.98687	-1.21113	0.000	600.000	81.269	50.386	0.000	3
14	0.98677	-1.28562	0.000	0.000	0.000	0.000	0.000	3
15	0.98569	-1.28771	0.000	0.000	25.322	15.693	0.000	3

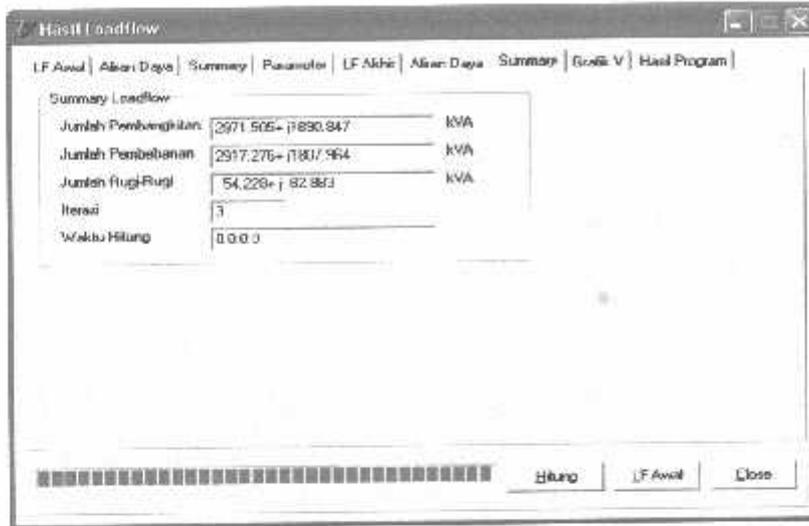
Untuk bus selanjutnya dapat dilihat pada lampiran tabel 5

Gambar 4-14. Tampilan Hasil Aliran Daya Untuk Mengetahui Profil Tegangan Dan Sudut Fasa Tiap-tiap Bus Setelah Optimasi

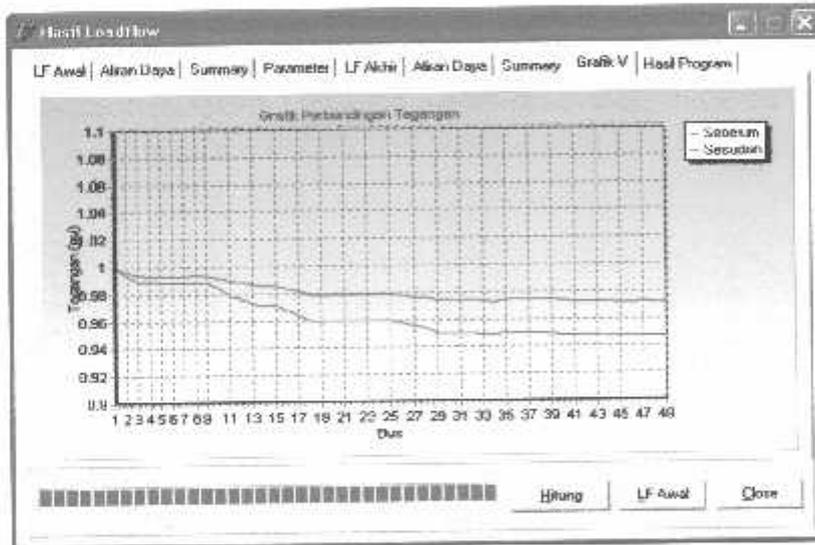
No	Dari	Ke	P (kW)	Q (KVAR)	Arus ke (A)	Arus m (A)	Dari	Ke	P (kW)	Q (KVAR)
1	2	3	2071.505	390.947	148.575	19.547	2	1	2961.212	-3
2	2	3	2007.797	329.615	144.900	17.241	3	2	2882.503	-3
3	3	4	647.923	401.989	32.449	20.463	4	3	647.684	-4
4	4	5	574.955	396.489	26.800	10.164	5	4	574.686	-3
5	5	6	471.820	292.470	23.645	14.913	6	5	471.750	-2
6	5	7	102.860	63.753	5.116	3.251	7	5	102.859	-4
7	3	0	2234.680	-80.467	112.452	-3.223	8	3	2233.705	1
8	8	9	90.100	55.846	4.513	2.047	9	8	-90.100	-1
9	8	10	2143.698	-137.652	107.939	-6.070	10	8	-2138.823	1
10	10	11	2039.710	206.749	102.947	-9.244	11	10	-2033.682	2
11	11	12	1947.900	-268.210	88.654	-11.899	12	11	-1945.209	2
12	12	13	1883.771	-310.399	95.581	-13.961	13	12	-1880.261	3
13	13	14	1798.992	233.870	90.876	13.773	14	13	-1797.294	-2
14	14	15	25.322	15.693	1.266	0.025	15	14	-25.321	-

Untuk saluran Selanjutnya dapat dilihat pada lampiran tabel 6

Gambar 4-15. Tampilan Hasil Aliran Daya Untuk Arus yang Mengalir pada Saluran dan Daya tiap Saluran Setelah Optimasi



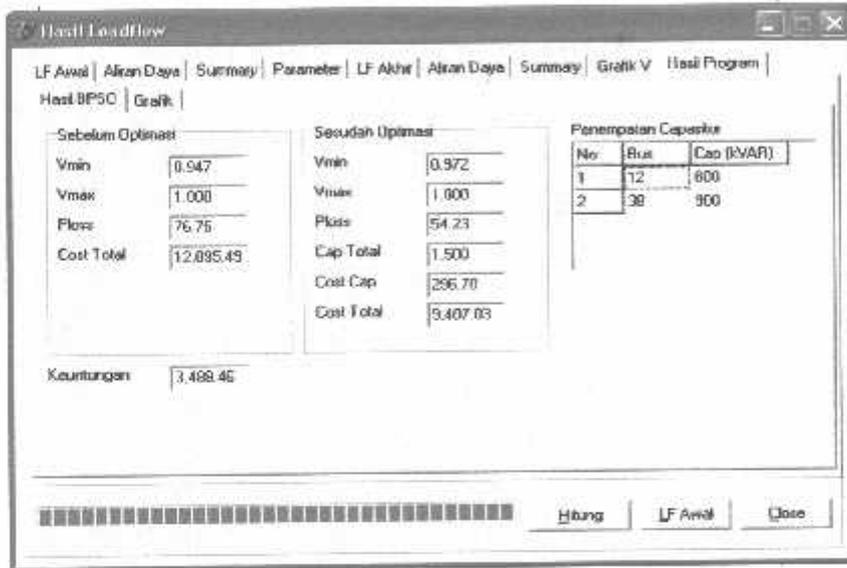
Gambar 4-16. Tampilan Total Pembangkitan, Pembebanan, dan Rugi-rugi Setelah Optimasi



Gambar 4-17. Tampilan Perbandingan Profil Tegangan Sebelum dan Setelah Optimasi

6. Hasil Program merupakan *Fitur* terakhir, terdiri dari:

- ⊕ Hasil Program sebelum optimasi, sesudah optimasi, penempatan kapasitor, keuntungan yang diperoleh dan grafik. Dengan cara:
 - Menekan menu **Hasil Program**



Gambar 4-18. Hasil Penempatan Metode *Binary Particle Swarm Optimization* (BPSO)

Dari hasil perhitungan program di atas dapat dilihat bahwa penyulang Syiar perlu dipasang kapasitor untuk menaikkan profil tegangan yang dianggap kritis yang beroperasi diluar batas yang diijinkan oleh PLN (0.95% - 1.05%) yang terjadi pada bus 33-34 dan 37-49. Mengurangi rugi-rugi saluran dan untuk mengurangi biaya operasional seminim mungkin. Seperti terlihat pada gambar 4-19. tampilan hasil perhitungan dengan metode *Binary Particle Swarm Optimization* (BPSO).

Berikut adalah *Single Line Diagram* lokasi penempatan kapasitor baru pada penyulang Syiar setelah optimasi:

4.6. Hasil Analisa Program dengan Menggunakan Metode *Binary Particle Swarm Optimization* (BPSO)

Dari proses perhitungan program, maka lokasi penempatan kapasitor terdapat pada bus 12 dengan kapasitas 600 kVAR tipe *Fixed* dan bus 38 dengan kapasitas 900 kVAR tipe *Fixed*. Dimana besar keuntungan dari biaya *losses* tahunan yang dikeluarkan adalah sebesar 3,488.46 US\$ atau Sebesar Rp 32.442.678,- (dengan asumsi 1 US\$ adalah Rp. 9.300,-).

Dibawah ini, merupakan contoh perhitungan manual total biaya *losses* tahunan:

- Total biaya *losses* tahunan sebelum optimasi:

$$\begin{aligned} \text{Cost total sebelum} &= \text{Ploss sebelum} \times \text{harga Ploss tahunan} \\ &= 76,76 \times 168 \\ &= 12.896,49 \text{ US\$} \end{aligned}$$

- Total biaya *losses* tahunan setelah optimasi:

$$\begin{aligned} \text{Cost total setelah} &= (\text{Ploss setelah} \times \text{harga Ploss tahunan}) + \text{harga kapasitor} \\ &= (54,23 \times 168) + 296,70 \\ &= 9.407,03 \text{ US\$} \end{aligned}$$

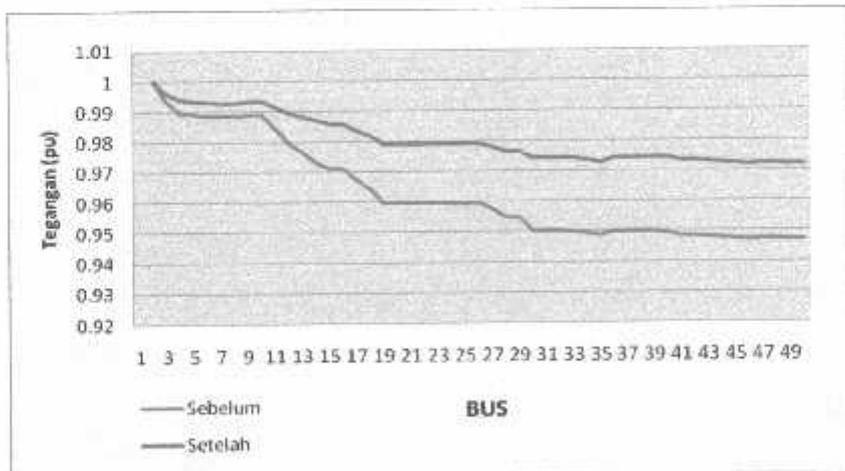
- Keuntungan total biaya *losses* tahunan:

$$\begin{aligned} \text{Keuntungan tahunan} &= \text{Cost total sebelum} - \text{Cost total setelah} \\ &= 12.896,49 - 9.407,03 \\ &= 3.488,46 \text{ US\$} \end{aligned}$$

Untuk lebih jelasnya dapat dilihat pada table 4-3. dibawah ini:

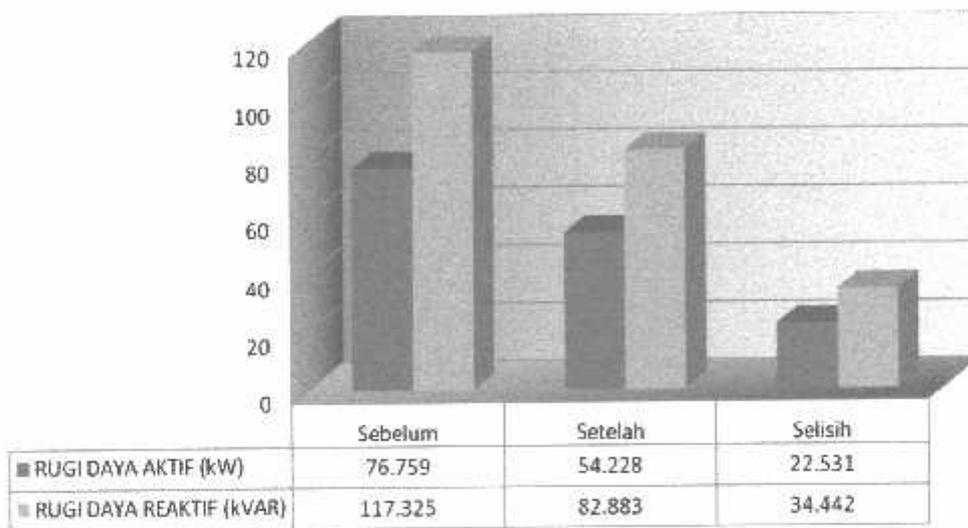
Tabel 4-3
Hasil Analisa Program dengan Metode *Binary Particle Swarm Optimization* (BPSO)

No		Sebelum	Setelah	
			Bus	Kapasitor (kVAR)
1	Lokasi dan kapasitas kapasitor yang akan dipasang			
			12	600
			38	900
2	Tegangan Terendah Pada Bus 49 (pu)	0.94716	0.97204	
3	Rugi Daya:	76.759	54.228	
	Aktif (kW)		22.531	
	Selisih (kW)			
	Pengurangan (%)		29.35	
	Reaktif (kVAR)		117.325	82.883
Selisih (kVAR)	34.442			
Pengurangan (%)			29.35	
4	Total Biaya (\$/Tahun)	12.895.49	9.407.03	
	Selisih (\$/Tahun)		3.488.46	
	Penghematan (%)		27.05	



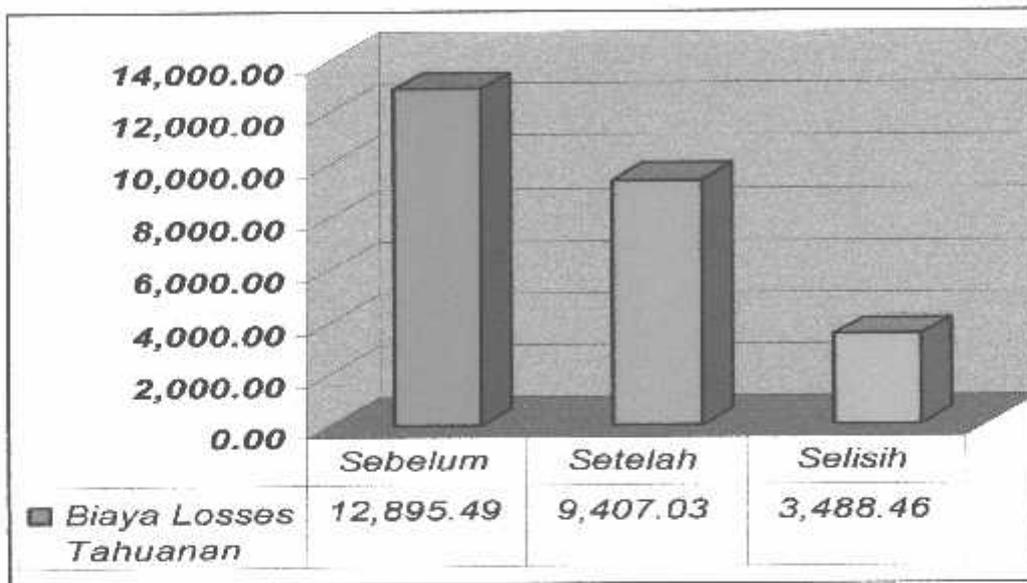
Grafik 4-1. Tegangan Tiap-Tiap Bus Terhadap Asumsi Tegangan Awal, Tegangan Sebelum dan Setelah Optimasi.

Dari grafik 4-1 terlihat bahwa tegangan terendah terjadi pada bus 49 sebesar 0,94716 pu atau 94,716% dan setelah optimasi naik menjadi 0,97204 pu atau 97,204%. Batas tegangan yang diijinkan oleh PLN adalah sebesar 0,95 pu sampai 1,05 pu. Sedangkan rugi-rugi daya sebelum dan sesudah optimasi dapat ditunjukkan seperti pada grafik 4-2



Grafik 4-2. Rugi-Rugi Saluran Sebelum dan Setelah Optimasi

Dari grafik 4-2 terlihat bahwa penurunan rugi daya aktif adalah 22.531 kW dari 76.759 kW menjadi 54.228 kW sehingga terjadi penurunan sebesar 29,35%, Sedangkan untuk daya reaktif juga terjadi penurunan sebesar 34.442 kVAR dari 117.325 kVAR menjadi 82.883 kVAR sehingga terjadi penurunan sebesar 29,35%.



Grafik 4-3. Biaya *Losses* Tahunan Sebelum dan Setelah Optimasi

Dari grafik 4-3 diatas dapat lihat bahwa biaya *losses* tahunan yang dikeluarkan sebelum optimasi sebesar 12,895.49 \$/Tahun atau sebesar Rp. 119.928.057,-/Tahun dan besar biaya *losses* tahunan yang dikeluarkan setelah optimasi adalah sebesar 9,407.03 US\$ atau sebesar Rp. 87.485.379,-/Tahun, sehingga diperoleh nilai penghematan sebesar 3,488.46 \$/Tahun atau sebesar Rp. 32.442.678,-/Tahun dengan nilai prosentase 27.05% (dengan asumsi 1 US\$ adalah Rp. 9.300,-).

BAB V KESIMPULAN

5.1. Kesimpulan

Berdasarkan hasil analisa dapat diperoleh beberapa kesimpulan sebagai berikut:

1. Berdasarkan hasil perhitungan, letak kapasitor yang optimal terletak pada *bus* 12 dan *bus* 38 dengan kapasitas masing-masing sebesar 600 kVAR dan 900 kVAR.
2. Dari hasil perhitungan aliran daya sebelum optimasi, tegangan terendah saluran yang terjadi sebesar 0.94716 pu atau 94,716% sebesar 18.9432 kV pada *bus* 49 dan setelah optimasi diperbaiki menjadi 0.97204 pu atau 97,204% sebesar 19.4408 kV.
3. Besarnya penurunan rugi-rugi daya setelah optimasi adalah rugi daya aktif sebesar 29.35% dari 76.759 kW menjadi 54.228 kW, dan daya reaktif sebesar 29.35% dari 117.325 kVAR menjadi 82.883 kVAR.
4. Nilai penghematan yang diperoleh sebesar Rp. 32.442.678/Tahun atau sebesar 26.93%, dimana biaya *losses* tahunan sebelum pemasangan sebesar 12,895.49 US \$/Tahun atau sebesar Rp. 119.928.057,-/Tahun, dan total biaya *losses* tahunan setelah pemasangan sebesar 9,407.03 US \$/Tahun atau sebesar Rp. 87.485.379,-/Tahun (dengan asumsi 1 US \$ adalah Rp. 9300,-).

5.2. Saran

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

Salah satu cara yang bermanfaat dalam mengoptimalkan suplai daya dari pembangkit sampai beban telah disampaikan penulis melalui skripsi ini.

DAFTAR PUSTAKA

- [1]. Hasan Basri, "**Sistem Distribusi Tenaga Listrik**".
 - [2]. Djiteng Marsudi, "**Operasi Sistem Tenaga Listrik**", Balai Penerbit Dan Humas ISTN, 1990.
 - [3]. William D Stevenson, Jr. "**Analisa Sistem Tenaga Listrik**". Edisi ke empat, Erlangga, 1993, Jakarta.
 - [4]. **AS. Pabla** *Punjab State Electricity Board Chandigarh*, Ir. Abdul Hadi "**Sistem Distribusi Daya Listrik**" Penerbit Erlangga 1986, Jakarta.
 - [5]. PT. PLN (persero) Distribusi Jawa Timur, "**Laporan Akhir Studi Pengembangan Sistem Distribusi Tenaga Listrik Kota Malang Dan Skitarnya**," ITS. Maret 2006, Surabaya.
 - [6]. Tamer Mohamed Khalil, Hosem K.H. Youssef, M.M. Abdel Aziz, "**A Binary Particle Swarm Optimization for Optimal Placement and Sizing of Capacitor Bank in Radial Distribution Feeders with Distorted Substation Voltages**", AIML 06 International Conference, Sharm El Sheikh, 13-15 June 2006, Egypt.
-



LAMPIRAN

- ✓ DATA MENTAH DAN DATA HASIL PERHITUNGAN
 - ✓ TAMPILAN BAHASA PEMROGRAMAN
-

Tabel 1
Data Hasil Perhitungan Pembebanan Sistem 20kV Penyulang Syiar

No Node	Kode Trafo	Lokasi	Pembebanan			Type Bus
			S (kVA)	P (kW)	Q (kVAR)	
1	-	-	0	0	0	Slack
2	T.71	Mojokopek	86.37	73.4145	45.4982	Load
3	-	-	0	0	0	Load
4	T.69	Mojoparon	85.68	72.8280	45.1347	Load
5	-	-	0	0	0	Load
6	T.90	PT. Metabisulfid	555	471.7500	292.3644	Load
7	T.203	Ds. Blawi	121.01	102.8585	63.7460	Load
8	-	-	0	0	0	Load
9	T.26	Ds. Pekoren	106	90.1000	55.8390	Load
10	T.130	Ds. Pekoren	117.31	99.7135	61.7969	Load
11	T.202	Ds. Rembang	100.92	85.7820	53.1629	Load
12	T.70	Ds. Ketapaan	72.28	61.4380	38.0759	Load
13	T.199	Ds. Kdg Banteng	95.61	81.2685	50.3657	Load
14	-	-	0	0	0	Load
15	T.200	Ds. Kdg Banteng	29.79	25.3215	15.6929	Load
16	T.251	Ds. Cengkelek	25.63	21.7855	13.5014	Load
17	T.252	Ds. Candiroboh	31.87	27.0895	16.7886	Load
18	-	-	0	0	0	Load
19	-	-	0	0	0	Load
20	T.144	Pejangkungan	66.05	56.1425	34.7940	Load
21	-	-	0	0	0	Load
22	T.125	Sbr. Glagah	43.18	36.7030	22.7465	Load
23	-	-	0	0.0000	0.0000	Load
24	T.182	Roh Wangkal	26.56	22.5760	13.9913	Load
25	T.246	Ds. Oyoran	30.71	26.1035	16.1775	Load
26	T.41	Ds. Rembang	75.05	63.7925	39.5350	Load
27	-	-	0	0.0000	0.0000	Load
28	T.186	Oro-Oro Bulu	80.13	68.1105	42.2111	Load
29	T.134	Kalisat	69.05	58.6925	36.3743	Load
30	-	-	0	0.0000	0.0000	Load
31	T.135	Ds. Kalisar	39.26	33.3710	20.6815	Load
32	-	-	0	0.0000	0.0000	Load
33	T.301	Pohkpan/Sisipan	57.27	48.6795	30.1688	Load
34	T.227	PT. Charun Pohkpan	345	293.2500	181.7400	Load
35	T.25	Ds. Kdg Lingit	52.65	44.7525	27.7351	Load
36	-	-	0	0.0000	0.0000	Load
37	T.245	Pejaten	71.59	60.8515	37.7124	Load
38	T.154	Tampung Barat	74.36	63.2060	39.1716	Load
39	T.94	Ds. Tampung	149.41	126.9985	78.7066	Load
40	-	-	0	0.0000	0.0000	Load
41	T.95	Ds. Kanigoro	65.59	55.7515	34.5517	Load
42	T.145	DS. Krenggih	77.13	65.5605	40.6307	Load
43	-	-	0	0.0000	0.0000	Load
44	-	-	0	0.0000	0.0000	Load
45	T.239	Dinas Pertanian	450	382.5000	237.0522	Load
46	T.185	Ds. Slambrit	54.5	46.3250	28.7097	Load
47	T.117	Ds. Syiar	48.73	41.4205	25.6701	Load
48	T.86	Genenganwaru	96.76	82.2460	50.9715	Load
49	T.87	Ds. Wangkal	31.64	26.8940	16.6674	Load

Tabel 2
Data Hasil Perhitungan Saluran Penyulang Syiar

Saluran		Panjang (m)	Impedensi Saluran	
Dari	Ke		R(Ω)	X(Ω)
1	2	2120	0.45834	0.70066
2	3	1150	0.24863	0.38008
3	4	750	0.16215	0.24788
4	5	680	0.14702	0.22474
5	6	430	0.09297	0.14212
5	7	570	0.12323	0.18839
3	8	320	0.06918	0.10576
8	9	780	0.20966	0.26333
8	10	1890	0.40862	0.62465
10	11	2350	0.50807	0.77668
11	12	1260	0.27241	0.41643
12	13	1740	0.37619	0.57507
13	14	930	0.20107	0.30737
14	15	2490	0.66931	0.84062
14	16	1960	0.42375	0.64778
16	17	1480	0.31998	0.48914
17	18	2490	0.53834	0.82295
18	19	430	0.09297	0.14212
19	20	1600	0.43008	0.54016
19	21	510	0.11026	0.16856
21	22	920	0.19890	0.30406
21	23	1320	0.28538	0.43626
23	24	2460	0.66125	0.83050
23	25	1270	0.34138	0.42875
18	26	1120	0.24214	0.37016
26	27	1670	0.36105	0.55194
27	28	2590	0.69619	0.87438
27	29	2850	0.61617	0.94193
29	30	100	0.02162	0.03305
30	31	120	0.02594	0.03966
31	32	250	0.05405	0.08263
32	33	1450	0.31349	0.47923
33	34	2610	0.56428	0.86261
32	35	720	0.19354	0.24307
30	36	233	0.05037	0.07701
36	37	820	0.22042	0.27683
36	38	130	0.02811	0.04297
38	39	590	0.12756	0.19500
39	40	1090	0.23566	0.36025
40	41	1100	0.29568	0.37136
41	42	1070	0.28762	0.36123
40	43	950	0.20539	0.31398
43	44	790	0.17080	0.26110
44	45	750	0.16215	0.24788
44	46	1460	0.39245	0.49290
43	47	2210	0.47780	0.73041
47	48	2250	0.48645	0.74363
48	49	2500	0.54050	0.82625

Tabel 3
Profil Tegangan dan Sudut Fasa Tiap Bus Penyulang Syiar
Sebelum Optimasi

Bus	Tegangan abs (pu)	Sudut V (deg)
1	1.00000	0.00000
2	0.99320	-0.17528
3	0.98961	-0.26894
4	0.98909	-0.28291
5	0.98867	-0.29415
6	0.98846	-0.30000
7	0.98861	-0.29584
8	0.98883	-0.28916
9	0.98874	-0.29092
10	0.98441	-0.40428
11	0.97917	-0.54199
12	0.97648	-0.61324
13	0.97289	-0.70910
14	0.97105	-0.75837
15	0.97097	-0.76001
16	0.96722	-0.86134
17	0.96438	-0.93865
18	0.95966	-1.06769
19	0.95960	-1.06955
20	0.95948	-1.07194
21	0.95955	-1.07088
22	0.95951	-1.07192
23	0.95948	-1.07285
24	0.95941	-1.07433
25	0.95944	-1.07373
26	0.95772	-1.12129
27	0.95494	-1.19832
28	0.95472	-1.20306
29	0.95041	-1.32476
30	0.95025	-1.32904
31	0.95020	-1.33061
32	0.95009	-1.33362
33	0.94954	-1.34909
34	0.94869	-1.37301
35	0.95005	-1.33450
36	0.95001	-1.33595
37	0.94995	-1.33731
38	0.94988	-1.33957
39	0.94934	-1.35481
40	0.94849	-1.37869
41	0.94832	-1.38232
42	0.94823	-1.38423
43	0.94788	-1.39592
44	0.94750	-1.40654
45	0.94718	-1.40635
46	0.94741	-1.40838
47	0.94751	-1.40635
48	0.94723	-1.41405
49	0.94716	-1.41616

Tabel 4
Aliran Daya Tiap Saluran Penyulang Syiar
Sebelum Optimasi

Saluran		P (kW)	Q (kVAR)	Saluran		P (kW)	Q (kVAR)
Dari	Ke			Dari	Ke		
1	2	2994.035	1925.289	2	1	-2979.516	-1903.094
2	3	2906.102	2390.822	3	2	-2898.606	-1846.137
3	4	647.927	401.995	4	3	-647.686	-401.627
4	5	574.858	356.492	5	4	-574.686	-356.229
5	6	471.823	292.476	6	5	-471.750	-292.364
5	7	102.863	63.753	7	5	-102.858	-63.746
3	8	2250.679	1444.142	8	3	-2249.416	-1442.211
8	9	90.106	55.846	9	8	-90.100	-55.839
8	10	2159.310	1386.365	10	8	-2152.431	-1375.849
10	11	2052.717	1314.052	11	10	-2044.931	-1302.149
11	12	1959.149	1248.986	12	11	-1955.315	-1243.125
12	13	1893.877	1205.049	13	12	-1888.907	-1197.451
13	14	1807.638	1147.086	14	13	-1805.204	-1143.365
14	15	25.323	15.695	15	14	-25.321	-15.693
14	16	1779.881	1127.670	16	14	-1774.893	-1120.045
16	17	1753.107	1106.544	17	16	-1749.432	-1100.926
17	18	1722.343	1084.137	18	17	-1716.349	-1074.975
18	19	141.546	87.739	19	18	-141.539	-87.728
19	20	56.148	34.800	20	19	-56.142	-34.794
19	21	85.391	52.928	21	19	-85.388	-52.923
21	22	36.704	22.748	22	21	-36.703	-22.746
21	23	48.684	30.175	23	21	-48.682	-30.171
23	24	22.577	13.993	24	23	-22.576	-13.991
23	25	26.104	16.179	25	23	-26.103	-16.178
18	26	1574.803	987.236	26	18	-1572.533	-983.765
26	27	1508.740	944.230	27	26	-1505.623	-939.464
27	28	68.123	42.226	28	27	-68.110	-42.211
27	29	1437.500	897.237	29	27	-1432.649	-889.822
29	30	1373.957	853.448	30	29	-1373.800	-853.209
30	31	420.341	260.902	31	30	-420.413	-260.875
31	32	387.042	240.194	32	31	-387.011	240.146
32	33	342.257	212.409	33	32	-342.116	-212.194
33	34	293.437	182.025	34	33	-293.250	-181.740
32	35	44.754	27.737	35	32	-44.752	-27.735
30	36	953.370	592.307	36	30	-953.194	-592.038
36	37	60.855	37.716	37	36	-60.851	-37.712
36	38	892.340	554.322	38	36	-892.254	-554.191
38	39	829.048	515.019	39	38	-828.711	-514.504
39	40	701.712	435.798	40	39	-701.266	-435.116
40	41	121.334	75.209	41	40	-121.317	-75.188
41	42	65.565	40.637	42	41	-65.560	-40.631
40	43	579.933	359.907	43	40	-579.667	-359.500
43	44	429.041	266.091	44	43	-428.920	-265.906
44	45	382.591	237.192	45	44	-382.500	-237.052
44	46	46.328	28.714	46	44	-46.325	-28.710
43	47	150.626	93.409	47	43	-150.584	-93.345
47	48	109.164	67.675	48	47	-109.142	-67.641
48	49	26.896	16.671	49	48	-26.894	-16.667

Tabel 5
Profil Tegangan dan Sudut Fasa Tiap Bus Penyulang Syiar
Setelah Optimasi

Bus	Tegangan Abs (pu)	Sudut V (deg)
1	1.00000	0.00000
2	0.99592	-0.27368
3	0.99381	-0.42067
4	0.99329	-0.43452
5	0.99288	-0.44567
6	0.99266	-0.45146
7	0.99282	-0.44734
8	0.99344	-0.45576
9	0.99336	-0.45751
10	0.99146	-0.65857
11	0.98926	-0.90530
12	0.98820	-1.03486
13	0.98687	-1.21113
14	0.98577	-1.28562
15	0.98569	-1.28721
16	0.98352	-1.44171
17	0.98185	-1.55911
18	0.97914	-1.75552
19	0.97907	-1.75731
20	0.97896	-1.75961
21	0.97902	-1.75859
22	0.97899	-1.75958
23	0.97895	-1.76048
24	0.97889	-1.76189
25	0.97891	-1.76132
26	0.97809	-1.83955
27	0.97664	-1.96203
28	0.97643	-1.96656
29	0.97440	-2.16609
30	0.97433	-2.17309
31	0.97427	-2.17459
32	0.97417	-2.17745
33	0.97363	-2.19217
34	0.97280	-2.21491
35	0.97413	-2.17828
36	0.97426	-2.18651
37	0.97420	-2.18780
38	0.97424	-2.19376
39	0.97416	-2.22557
40	0.97333	-2.24825
41	0.97317	-2.25170
42	0.97308	-2.25351
43	0.97274	-2.26461
44	0.97237	-2.27469
45	0.97206	-2.28324
46	0.97229	-2.27645
47	0.97238	-2.27451
48	0.97211	-2.28183
49	0.97204	-2.28383


```

unit uBPSO;

interface

uses uUtils, uObjFunc, uHasil, uRandom;

type
  TIndividu=record
    chrom:bArr2;
    fitness:double;
  end;

  TSwarm=record
    now:TIndividu;
    prev:TIndividu;
    velo:dArr2;
  end;

  TSwarms=array of TSwarm;

  TBPSO=class
  private
    FIterasi, FLength, FParam, Fc1, Fc2, FPopSize:integer;
    Fw:double;
    FSwarm:TSwarms;
    FGlobal:TIndividu;
    FRandom:TRandomu;
    FMax:dArr1;
    function getIndividu(const rIndi:TIndividu):TIndividu;
    procedure InitSwarm;
    function FindGlobalBest:TIndividu;
    function Sigmoid(const rValue:double):double;
    procedure UpdateSwarm;
    procedure doHitung;
    function getBestChrom:bArr2;
    function getMax:dArr1;
  public
    constructor Create(const rIterasi, rLength, rParam, rc1, rc2,
      rPopSize:integer;
      const rw:double);
    destructor Destroy;override;
    property BestChrom:bArr2 read getBestChrom;
    property Max:dArr1 read getMax;
  end;

implementation

//constructor
constructor TBPSO.Create(const rIterasi, rLength, rParam, rc1, rc2,
  rPopSize:integer;
  const rw:double);

begin
  inherited Create;
  FIterasi:=rIterasi;
  FLength:=rLength;
  FParam:=rParam;
  Fc1:=rc1;
  Fc2:=rc2;
  FPopSize:=rPopSize;
  Fw:=rw;
  FRandom:=TRandomu.Create;

```

```

end;

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

//data processing
function TBPSO.getIndividu(const rIndi:TIndividu):TIndividu;
var i,j:integer;
begin
  SetLength(result.chrom, FParam, FLength);
  for i:=0 to FParam-1 do
  begin
    for j:=0 to FLength-1 do
    begin
      result.chrom[i,j]:=rIndi.chrom[i,j];
    end;
  end;
  result.fitness:=rindi.fitness;
end;

procedure TBPSO.InitSwarm;
var i,j,k:integer;
begin
  SetLength(FSwarm, FPopSize);
  SetLength(FMax, FIterasi);
  for i:=0 to FPopSize-1 do
  begin
    SetLength(FSwarm[i].now.chrom, FParam, FLength);
    SetLength(FSwarm[i].prev.chrom, FParam, FLength);
    SetLength(FSwarm[i].velo, FParam, FLength);
    for j:=0 to FParam-1 do
    begin
      for k:=0 to FLength-1 do
      begin
        if k=FLength-1 then
        begin
          FSwarm[i].now.chrom[j,k]:=FRandom.NextBoolean(0.02);
        end
        else
        begin
          FSwarm[i].now.chrom[j,k]:=FRandom.NextBoolean(3/FParam);
        end;
      end;
    end;
    FSwarm[i].now.fitness:=gObjFunc.doHitung(FSwarm[i].now.chrom);
    FSwarm[i].prev:=getIndividu(FSwarm[i].now);
  end;
end;

function TBPSO.findGlobalBest:TIndividu;
var i,no:integer;
    min:double;
begin
  min:=FSwarm[0].now.fitness;

```

```

no:=0;
for i:=1 to FPopSize-1 do
begin
  if min>FSwarm[i].now.fitness then
  begin
    min:=FSwarm[i].now.fitness;
    no:=i;
  end;
end;
result:=getIndividu(FSwarm[no].now);
end;

Function TBPSO.Sigmoid(const rValue:double):double;
begin
  result:=1/(1+exp(-rValue));
end;

procedure TBPSO.UpdateSwarm;
var i,j,k,no,p,g,x:integer;
    r1,r2,min:double;
begin
  for i:=0 to FPopSize-1 do
  begin
    for j:=0 to FParam-1 do
    begin
      for k:=0 to FLength-1 do
      begin
        r1:=random;
        r2:=random;
        if FSwarm[i].now.chrom[j,k]=true then
        begin
          x:=1;
        end
        else
        begin
          x:=0;
        end;
        if FSwarm[i].prev.chrom[j,k]=true then
        begin
          p:=1;
        end
        else
        begin
          p:=0;
        end;
        if FGlobal.chrom[j,k]=true then
        begin
          q:=1;
        end
        else
        begin
          g:=0;
        end;
        FSwarm[i].velo[j,k]:=Fw*FSwarm[i].velo[j,k]+
          Fc1*r1*(p-x)+Fc2*r2*(g-x);
      end;
    end;
  end;
  for i:=0 to FPopSize-1 do
  begin
    for j:=0 to FParam-1 do

```

```

begin
  for k:=0 to FLength-2 do
  begin
    r1:=random;
    r2:=Sigmoid(FSwarm[i].velo[j,k]);
    if r1<r2 then
    begin
      FSwarm[i].now.chrom[j,k]:=true;
    end
    else
    begin
      FSwarm[i].now.chrom[j,k]:=false;
    end;
  end;
end;
FSwarm[i].now.fitness:=gObjFunc.doHitung(FSwarm[i].now.chrom);
if FSwarm[i].now.fitness<FSwarm[i].prev.fitness then
begin
  FSwarm[i].prev:=getIndividu(FSwarm[i].now);
end;
end;
min:=FSwarm[0].now.fitness;
no:=0;
for i:=1 to FPopSize-1 do
begin
  if min>FSwarm[i].now.fitness then
  begin
    min:=FSwarm[i].now.fitness;
    no:=i;
  end;
end;
if FGlobal.fitness>min then
begin
  FGlobal:=getIndividu(FSwarm[no].now);
end;
end;

procedure TBPSO.doHitung;
var i:integer;
begin
  InitSwarm;
  FGlobal:=FindGlobalBest;
  for i:=1 to FIterasi do
  begin
    UpdateSwarm;
    FMax[i-1]:=FGlobal.fitness;
    frmHasil.pbGen.StepBy(1);
  end;
end;

function TBPSO.getBestChrom:bArr2;
var i,j:integer;
begin
  doHitung;
  SetLength(result,FParam,FLength);
  for i:=0 to FParam-1 do
  begin
    for j:=0 to FLength-1 do
    begin
      result[i,j]:=FGlobal.chrom[i,j];
    end;
  end;
end;

```

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

```

unit uObjFunc;

interface

uses uUtils, uLoadflow, uNewtonRaphson, uComplex, uVarGenetic, uRandom;

type
  TCapDat=record
    no:integer;
    size,cost:double;
  end;

  TCapDatArr1=array of TCapDat;

  TObjFunc=class
  private
    FNcap, FJmlCap:integer;
    FKp, FTHDmax, FLmdV, FLmdS, FLmdC:double;
    rBatasV:TBatas;
    FCapDat:TCapDatArr1;
    procedure DecodeChrom(const rChrom:bArr2;
      var rLbus:TBusArr1;
      var rCostCap:double;
      var rJmlCap:integer);
    function getFinV(const rLbus:TBusArr1):double;
    function getFinS(const rBranch:TBranchArr1):double;
  public
    constructor Create(const rJmlCap:integer;
      const rKp,rTHDmax,rLmdV,rLmdS,rLmdC:double;
      const rBatasV:TBatas;
      const rCapDat:TCapDatArr1);
    function doHitung(const rChrom:bArr2):double;
    procedure doHitungAwal(var rminV,rmaxV,rmaxTHD,rPloss,
      rCostTotal:double);
    procedure doHitungAkhir(const rChrom:bArr2;
      var rLbus:TBusArr1;
      var rPos:iArr1;
      var rCap:dArr1;
      var rminV,rmaxV,rmaxTHD,rPloss,rCostCap,rCostTotal:double);
  end;

var gObjFunc:TObjFunc;

implementation

//constructor
constructor TObjFunc.Create(const rJmlCap:integer;
  const rKp,rTHDmax,rLmdV,rLmdS,rLmdC:double;
  const rBatasV:TBatas;
  const rCapDat:TCapDatArr1);
var i:integer;
begin
  inherited Create;
  FNcap:=high(rCapDat)+1;
  SetLength(FCapDat,FNcap);
  for i:=0 to FNcap-1 do
  begin
    FCapDat[i].no:=rCapDat[i].no;
    FCapDat[i].size:=rCapDat[i].size;
    FCapDat[i].cost:=rCapDat[i].cost;
  end;
end;

```

```

FJmlCap:=rJmlCap;
FKp:=rKp;
FTHDmax:=rTHDmax;
FLmdV:=rLmdV;
FLmdS:=rLmdS;
FLmdC:=rLmdC;
FBatasV.min:=rBatasV.min;
FBatasV.max:=rBatasV.max;
end;

//data processing
procedure TObjFunc.DecodeChrom(const rChrom:bArr2;
    var rLbus:TBusArr1;
    var rCostCap:double;
    var rJmlCap:integer);
var i,j,Nbus,rows,cols,no:integer;
    cap,dc:double;
    chrom:bArr1;
begin
    Nbus:=high(gBus)+1;
    SetLength(rLbus,Nbus);
    for i:=0 to Nbus-1 do
    begin
        rLbus[i].absV:=gBus[i].absV;
        rLbus[i].sudV:=-gBus[i].sudV;
        rLbus[i].Pgen:=gBus[i].Pgen;
        rLbus[i].Qgen:=gBus[i].Qgen;
        rLbus[i].PL:=gBus[i].PL;
        rLbus[i].QL:=gBus[i].QL;
        rLbus[i].Cap:=gBus[i].Cap;
        rLbus[i].typeBus:=gBus[i].typeBus;
    end;
    rows:=high(rChrom)-1;
    cols:=high(rChrom[0])+1;
    rCostCap:=0;
    rJmlCap:=0;
    for i:=0 to rows-1 do
    begin
        if rChrom[i,cols-1]=true then
        begin
            //
            SetLength(chrom,cols-1);
            for j:=0 to cols-2 do
            begin
                chrom[j]:=rChrom[i,j];
            end;
            dc:=DecodeBinToFloat1Base0(chrom);
            no:=round(1+dc*(27-1))-1;
            rLbus[i].Qgen:=FCapDat[no].size;
            rCostCap:=rCostCap+FCapDat[no].cost*FCapDat[no].size;
            inc(rJmlCap);
            //
            (cap:=0;
            for j:=0 to cols-2 do
            begin
                if rChrom[i,j]=true then
                begin
                    cap:=cap+FCapDat[j].size;
                    rCostCap:=rCostCap+FCapDat[j].cost*FCapDat[j].size;
                end;
            end;
            end;
        end;
    end;
end;

```

```

    if cap<>0 then
    begin
        rLbus[i].Qgen:=cap;
        inc(rJmlCap);
    end;
end;
end;
end;

function TObjFunc.getPinV(const rLbus:TBusArr1):double;
var i,Nbus:integer;
begin
    Nbus:=high(rLbus)+1;
    result:=0;
    for i:=0 to Nbus-1 do
    begin
        if rLbus[i].absV>FBatasV.max then
        begin
            result:=result+(rLbus[i].absV-FBatasV.max);
        end;
        if rLbus[i].absV<FBatasV.min then
        begin
            result:=result+(FBatasV.min-rLbus[i].absV);
        end;
    end;
end;

function TObjFunc.getPinS(const rBranch:TBranchArr1):double;
var i,Nsal:integer;
    absS:double;
begin
    result:=0;
    Nsal:=high(rBranch)+1;
    for i:=0 to Nsal-1 do
    begin
        if rBranch[i].Sij.real>0 then
        begin
            absS:=sqrt(sqr(rBranch[i].Sij.real)+sqr(rBranch[i].Sij.imag));
        end
        else
        begin
            absS:=sqrt(sqr(rBranch[i].Sji.real)+sqr(rBranch[i].Sji.imag));
        end;
        if absS>rBranch[i].KapSal then
        begin
            result:=result+(absS-rBranch[i].KapSal);
        end;
    end;
end;

function TObjFunc.doHitung(const rChrom:bArr2):double;
var Lbus:TBusArr1;
    CostCap,Ploss,pinV,pinS,pinC:double;
    JmlCap:integer;
begin
    DecodeChrom(rChrom,Lbus,CostCap,JmlCap);
    NewtonRaphson(Lbus,gBranch,gParamLF);
    if gParamLF.Iterasi<15 then
    begin
        Ploss:=gParamLF.SumLoss.real;
        pinV:=getPinV(Lbus);
    end;
end;

```

```

pinS:=-getPinS(gBranch);
pinC:=0;
if JmlCap>FJmlCap then
begin
pinC:=JmlCap-FJmlCap;
end;
result:=-FKp*Ploss+CostCap+FLmdV*pinV+FLmdS*pinS+FLmdC*pinC;
end
else
begin
result:=-1000000000;
end;
end;

procedure TObjFunc.doHitungAwal(var rminV,rmaxV,rmaxTHD,rPloss,
rCostTotal:double);
var i,Nbus:integer;
begin
NewtonRaphson(gBus,gBranch,gParamLF);
rPloss:=gParamLF.SumLoss.real;
rCostTotal:=FKp*rPloss;
rmaxTHD:=0;
Nbus:=high(gbus)+1;
rminV:=gbus[0].absV;
rmaxV:=-gbus[0].absV;
for i:=0 to Nbus-1 do
begin
if rminV>gbus[i].absV then
begin
rminV:=gbus[i].absV;
end;
if rmaxV<gbus[i].absV then
begin
rmaxV:=-gbus[i].absV;
end;
end;
end;

procedure TObjFunc.doHitungAkhir(const rChrom:bArr2;
var rLbus:TBusArr1;
var rPos:iArr1;
var rCap:dArr1;
var rminV,rmaxV,rmaxTHD,rPloss,rCostCap,rCostTotal:double);
var i,sa,JmlCap,Nbus:integer;
begin
DecodeChrom(rChrom,rLbus,rCostCap,JmlCap);
SetLength(rPos,JmlCap);
SetLength(rCap,JmlCap);
sa:=0;
for i:=0 to high(rLbus) do
begin
if rLbus[i].typeBus=3 then
begin
if rLbus[i].Qgen<>0 then
begin
rPos[sa]:=i;
rCap[sa]:=rLbus[i].Qgen;
inc(sa);
end;
end;
end;
end;

```

```
NewtonRaphson(rLbus, gBranch, gParamLF);
rPloss:=gParamLF.SumLoss.real;
rCostTotal:=FKp*rPloss+rCostCap;
rmaxTHD:=0;
Nbus:=high(rLbus)+1;
rminV:=rLbus[0].absV;
rmaxV:=rLbus[0].absV;
for i:=0 to Nbus-1 do
begin
  if rminV>rLbus[i].absV then
  begin
    rminV:=rLbus[i].absV;
  end;
  if rmaxV<rLbus[i].absV then
  begin
    rmaxV:=rLbus[i].absV;
  end;
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 btnExitClick(Sender: TObject);
    procedure btnNewClick(Sender: TObject);
    procedure btnOpenClick(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmMenu: TfrmMenu;

implementation

uses uInputLEChild, uComplex, uUtils, uLoadflow;

{$R *.dfm}

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

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, OL, CapSal, Pmin, Pmax, Harga, Length:double;

```

```

R, X, Lc, Tr, Tu, Su, VKonst, PKonst, Pbase, Vbase:double;
begin
try
if OpenFileDialog.Execute then
begin
NamaFile:=OpenDialog1.FileName;
AssignFile(output, NamaFile);
Reset(output);
Readln(output, Nbus);
Readln(output, Nsal);
Readln(output, Vbase);
Readln(output, VKonst);
Readln(output, Pbase);
Readln(output, PKonst);
Readln(output, param);
gParamLF.Vbase:=Vbase;
gParamLF.VKonst:=-VKonst;
gParamLF.Pbase:=Pbase;
gParamLF.PKonst:=PKonst;
if Param=1 then
begin
gParamLF.ParamBranch:=pbPu;
end
else if Param=2 then
begin
gParamLF.ParamBranch:=pbOhm;
end;
gParamLF.MaxIterasi:=15;
gParamLF.Toleransi:=0.0001;
try
frmInput:=TfrmInputLFChild.Create(Application);
frmInput.edtNbus.Text:=IntToStr(Nbus);
frmInput.edtNsal.Text:=IntToStr(Nsal);
frmInput.edtVbase.Text:=FloatToStr(Vbase);
if VKonst=1 then
begin
frmInput.cmbVKonst.Text:='V';
end
else if VKonst=1000 then
begin
frmInput.cmbVKonst.Text:='kV';
end
else if VKonst=1000000 then
begin
frmInput.cmbVKonst.Text:='MV';
end;
frmInput.edtPbase.Text:=FloatToStr(Pbase);
if PKonst=1 then
begin
frmInput.cmbPKonst.Text:='VA';
frmInput.fgBus.Cells[3,0]:='Pg (W)';
frmInput.fgBus.Cells[4,0]:='Qg (VAR)';
frmInput.fgBus.Cells[5,0]:='PL (W)';
frmInput.fgBus.Cells[6,0]:='QL (VAR)';
frmInput.fgBranch.Cells[9,0]:='Kap (VA)';
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
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]:='P (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(ke);
    frmInput.fgBranch.Cells[3,i+1]:=FloatToStr(R);
    frmInput.fgBranch.Cells[4,i+1]:=FloatToStr(X);
    frmInput.fgBranch.Cells[5,i+1]:=FloatToStr(Lc);
    frmInput.fgBranch.Cells[6,i+1]:=FloatToStr(Tr);
    frmInput.fgBranch.Cells[7,i+1]:=FloatToStr(Tu);
    frmInput.fgBranch.Cells[8,i+1]:=FloatToStr(Su);
    frmInput.fgBranch.Cells[9,i+1]:=FloatToStr(CapSal);
end;
Readln(output,Ngen);
if Ngen<>0 then
begin
    frmInput.fgGen.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;
end;

```

```
    CloseFile(output);
    frmInput.Caption:='Tampilan Data';
    frmInput.btnNext.Caption:='&Next';
    frmInput.ShowModal;
  finally
    frmInput.Free;
  end;
end;
except
  MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);
end;
end;
end;
```

```
unit uBasil;

interface

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

type
  TfrmBasil = class(TForm)
    TabSheet2: TTabSheet;
    fgBus: TStringGrid;
    TabSheet3: TTabSheet;
    fgBranch: TStringGrid;
    TabSheet4: TTabSheet;
    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;
    TabSheet9: TTabSheet;
    Panel1: TPanel;
    btnClose: TButton;
    btnLFAwal: TButton;
    TabSheet1: TTabSheet;
    TabSheet5: TTabSheet;
    TabSheet6: TTabSheet;
    TabSheet7: TTabSheet;
    Chart1: TChart;
    Series1: TLineSeries;
    Series2: TLineSeries;
    fgBus2: TStringGrid;
    fgBranch2: TStringGrid;
    GroupBox1: TGroupBox;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    lblGen2: TLabel;
    lblLoad2: TLabel;
    lblLoss2: TLabel;
    edtSumGen2: TEdit;
    edtSumLoad2: TEdit;
    edtSumLoss2: TEdit;
    edtIterasi2: TEdit;
    edtTime2: TEdit;
    pbGen: TProgressBar;
    GroupBox2: TGroupBox;
    Label6: TLabel;
  end;
end;
```

```
Label7: TLabel;
Label14: TLabel;
edtIterasiMax: TEdit;
edtPopSize: TEdit;
edtc1: TEdit;
edtLength: TEdit;
GroupBox3: TGroupBox;
TabSheet8: TTabSheet;
PageControl2: TPageControl;
TabSheet10: TTabSheet;
Label28: TLabel;
fgHasilSPEA: TStringGrid;
TabSheet11: TTabSheet;
Chart2: TChart;
Series5: TLineSeries;
btnHitung: TButton;
btnUseDefault: TButton;
GroupBox4: TGroupBox;
Label13: TLabel;
edtw: TEdit;
Label18: TLabel;
Label30: TLabel;
edtc2: TEdit;
Label31: TLabel;
edtParams: TEdit;
PageControl1: TPageControl;
TabSheet12: TTabSheet;
TabSheet13: TTabSheet;
Label15: TLabel;
edtCountCap: TEdit;
Label19: TLabel;
edtJmlCapBank: TEdit;
Label16: TLabel;
edtKp: TEdit;
Label17: TLabel;
edtTHDmax: TEdit;
Label20: TLabel;
edtLmdV: TEdit;
Label21: TLabel;
edtLmdS: TEdit;
Label22: TLabel;
edtVmin: TEdit;
edtVmax: TEdit;
fgKap: TStringGrid;
Label23: TLabel;
edtLmdC: TEdit;
Label24: TLabel;
edtVmin1: TEdit;
Label25: TLabel;
edtVmax1: TEdit;
Label26: TLabel;
edtPloss1: TEdit;
Label27: TLabel;
edtCosTotal1: TEdit;
GroupBox: TGroupBox;
Label37: TLabel;
Label38: TLabel;
edtSelisih: TEdit;
Label29: TLabel;
edtVmin2: TEdit;
Label32: TLabel;
```

```

edtVmax2: TEdit;
Label33: TLabel;
edtPloss2: TEdit;
Label34: TLabel;
edtCapTotal: TEdit;
Label35: TLabel;
edtCostCap: TEdit;
Label36: TLabel;
edtCostTotal2: TEdit;
procedure btnCloseClick(Sender: TObject);
procedure btnLFAwalClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure btnUseDefaultClick(Sender: TObject);
procedure edtJmlCapBankChange(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmHasil: TfrmHasil;

implementation

uses uUtils, uLoadflow, uNewtonRaphson, uComplex, uTopology,
    uVarGenetic, uObjFunc, uBPSO;

{$R *.dfm}

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

procedure TfrmHasil.btnLFAwalClick(Sender: TObject);
var i:integer;
    mulai,selesai,selang:TDateTime;
    jam,menit,detik,mdetik:word;
begin
  mulai:=time;
  NewtonRaphson(gBus,gBranch,gParamLF);
  selesai:=time;
  selang:=selesai-mulai;
  Series1.Clear;
  Series2.Clear;
  for i:=0 to high(gBus) do
  begin
    fgBus.Cells[0,i+1]:=IntToStr(i+1);
    fgBus.Cells[1,i+1]:=RealToStr(gBus[i].absV,5);
    fgBus.Cells[2,i+1]:=RealToStr(gBus[i].sudV*
      ANGLE_OF_DEGREES/ANGLE_OF_RADIAN,5);
    fgBus.Cells[3,i+1]:=RealToStr(gBus[i].Pgen,3);
    fgBus.Cells[4,i+1]:=RealToStr(gBus[i].Qgen,3);
    fgBus.Cells[5,i+1]:=RealToStr(gBus[i].PL,3);
    fgBus.Cells[6,i+1]:=RealToStr(gBus[i].QL,3);
    fgBus.Cells[7,i+1]:=RealToStr(gBus[i].Cap,3);
    fgBus.Cells[8,i+1]:=IntToStr(gBus[i].typeBus);
    Series1.Add(gBus[i].absV,IntToStr(i+1));
  end;
end;

```

```

for i:=0 to high(gBranch) do
begin
  fgBranch.Cells[0,i+1]:=intToStr(i+1);
  fgBranch.Cells[1,i+1]:=IntToStr(gBranch[i].dari);
  fgBranch.Cells[2,i+1]:=-IntToStr(gBranch[i].ke);
  fgBranch.Cells[3,i+1]:=RealToStr(gBranch[i].Sij.real,3);
  fgBranch.Cells[4,i+1]:=RealToStr(gBranch[i].Sij.imag,3);
  fgBranch.Cells[5,i+1]:=RealToStr(gBranch[i].Aij.real,3);
  fgBranch.Cells[6,i+1]:=RealToStr(gBranch[i].Aij.imag,3);
  fgBranch.Cells[7,i+1]:=IntToStr(gBranch[i].ke);
  fgBranch.Cells[8,i+1]:=IntToStr(gBranch[i].dari);
  fgBranch.Cells[9,i+1]:=RealToStr(gBranch[i].Sji.real,3);
  fgBranch.Cells[10,i+1]:=RealToStr(gBranch[i].Sji.imag,3);
  fgBranch.Cells[11,i+1]:=RealToStr(gBranch[i].Aji.real,3);
  fgBranch.Cells[12,i+1]:=-RealToStr(gBranch[i].Aji.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);
btnUseDefault.Enabled:=true;
end;

procedure TfrmHasil.FormCreate(Sender: TObject);
begin
  fgHasilSPEA.Cells[0,0]:='No';
  fgHasilSPEA.Cells[1,0]:='Bus';
  fgHasilSPEA.Cells[2,0]:='Cap (kVAR)';
  fgKap.Cells[0,0]:='No';
  fgKap.Cells[1,0]:='Size (kvar)';
  fgKap.Cells[2,0]:='Cost ($/kvar)';
end;

procedure TfrmHasil.btnHitungClick(Sender: TObject);
var BatasCap:TBatas;
    Ncap,i,j,param,length,iterasiMax,popsize,CountCap:integer;
    Vmin1,Vmax1,Cos.Total1,THD1,Ploss1,CapTotal:double;

Kp,THDmax,LmdV,LmdS,LmdC,w,Vmin2,Vmax2,CostCap,CostTotal2,THD2,Ploss2:double;
BatasV:TBatas;
CapDat:TCapDatArr1;
Max,Cap:dArr1;
BestChrom:bArr2;
Nbus,Nsal,ia,ja,cl,c2:integer;
V,Sg,SL:CxArr1;
Lc,Tr,HasilCap:dArr2;
TypBus,Pos:iArr1;
Z,Tp,Alir,Arus:CxArr2;
mulai,selesai,selang:TDateTime;
jam,menit,detik,mdetik:word;
bpsc:TBPSC;
LBis:TBusArr1;
begin
  CountCap:=StrToInt(edtCountCap.Text);
  Ncap:=StrToInt(edtJmlCapBank.Text);
  Kp:=StrToFloat(edtKp.Text);
  THDmax:=StrToFloat(edtTHDmax.Text);

```

```

LmdV:=StrToFloat(edtLmdV.Text);
LmdS:=StrToFloat(edtLmdS.Text);
LmdC:=StrToFloat(edtLmdC.Text);
BatasV.min:=StrToFloat(edtVmin.Text);
BatasV.max:=StrToFloat(edtVmax.Text);
SetLength(CapDat,Ncap);
for i:=1 to Ncap do
begin
  CapDat[i-1].no:=i;
  CapDat[i-1].size:=StrToFloat(fgKap.Cells[1,i]);
  CapDat[i-1].cost:=StrToFloat(fgKap.Cells[2,i]);
end;
gObjFunc:=TObjFunc.Create(CountCap,Kp,THDmax,LmdV,LmdS,LmdC,
  BatasV,CapDat);
gObjFunc.doHitungAwal(Vmin1,Vmax1,THD1,Ploss1,CostTotal1);
edtVmin1.Text:=FormatFloat('#,##0.000',Vmin1);
edtVmax1.Text:=FormatFloat('#,##0.000',Vmax1);
edtPloss1.Text:=FormatFloat('#,##0.00',Ploss1);
edtCostTotal1.Text:=FormatFloat('#,##0.00',CostTotal1);
PopSize:=StrToInt(edtPopSize.Text);
IterasiMax:=StrToInt(edtiterasiMax.Text);
pbGen.Max:=IterasiMax;
Length:=StrToInt(edtLength.Text);
Param:=StrToInt(edtParams.Text);
w:=StrToFloat(edtw.Text);
c1:=StrToInt(edtc1.Text);
c2:=StrToInt(edtc2.Text);
bpso:=TBPSO.Create(IterasiMax,Length,Param,c1,c2,PopSize,w);
BestChrom:=bpso.BestChrom;
Max:=bpso.Max;
gObjFunc.doHitungAkhir(BestChrom,Lbus,Pos,Cap,Vmin2,Vmax2,THD2,Ploss2,
  CostCap,CostTotal2);
edtVmin2.Text:=FormatFloat('#,##0.000',Vmin2);
edtVmax2.Text:=FormatFloat('#,##0.000',Vmax2);
edtPloss2.Text:=FormatFloat('#,##0.00',Ploss2);
edtCostTotal2.Text:=FormatFloat('#,##0.00',CostTotal2);
edtCostCap.Text:=FormatFloat('#,##0.00',CostCap);
fgHasilSPEA.RowCount:=high(Pos)+2;
CapTotal:=0;
for i:=0 to high(Pos) do
begin
  fgHasilSPEA.Cells[0,i+1]:=IntToStr(i+1);
  fgHasilSPEA.Cells[1,i+1]:=IntToStr(Pos[i]);
  fgHasilSPEA.Cells[2,i+1]:=FormatFloat('#,##0',Cap[i]);
  CapTotal:=CapTotal+Cap[i];
end;
edtCapTotal.Text:=FormatFloat('#,##0',CapTotal);
edtSelisih.Text:=FormatFloat('#,##0.00',(CostTotal1-CostTotal2));
Series5.Clear;
for i:=0 to high(Max) do
begin
  Series5.Add(Max[i],IntToStr(i+1));
end;
//
DecodeCommDataToLFData(LBus,Nbus,Nsal,V,Sg,SL,Cap,TypBus,
  gBranch,Z,Tp,Lc,Tr);
mulai:=time;
NewtonRaphson(gParamLF,V,Sg,SL,Cap,TypBus,Z,Tp,Alir,Arus,
  Lc,Tr);
selesai:=time;
selang:=selesai-mulai;

```

```

edtLmdV.Text:='1000';
edtLmdS.Text:='1000';
edtLmdC.Text:='1000';
edtVmin.Text:='0.95';
edtVmax.Text:='1.05';
fgKap.RowCount:=28;
for i:=1 to 27 do
begin
  fgKap.Cells[0,i]:=IntToStr(i);
end;
fgKap.Cells[1,1]:='150';fgKap.Cells[2,1]:='0.5';
fgKap.Cells[1,2]:='300';fgKap.Cells[2,2]:='0.35';
fgKap.Cells[1,3]:='450';fgKap.Cells[2,3]:='0.253';
fgKap.Cells[1,4]:='600';fgKap.Cells[2,4]:='0.22';
fgKap.Cells[1,5]:='750';fgKap.Cells[2,5]:='0.276';
fgKap.Cells[1,6]:='900';fgKap.Cells[2,6]:='0.183';
fgKap.Cells[1,7]:='1050';fgKap.Cells[2,7]:='0.228';
fgKap.Cells[1,8]:='1200';fgKap.Cells[2,8]:='0.17';
fgKap.Cells[1,9]:='1350';fgKap.Cells[2,9]:='0.207';
fgKap.Cells[1,10]:='1500';fgKap.Cells[2,10]:='0.201';
fgKap.Cells[1,11]:='1650';fgKap.Cells[2,11]:='0.193';
fgKap.Cells[1,12]:='1800';fgKap.Cells[2,12]:='0.187';
fgKap.Cells[1,13]:='1950';fgKap.Cells[2,13]:='0.211';
fgKap.Cells[1,14]:='2100';fgKap.Cells[2,14]:='0.076';
fgKap.Cells[1,15]:='2250';fgKap.Cells[2,15]:='0.197';
fgKap.Cells[1,16]:='2400';fgKap.Cells[2,16]:='0.17';
fgKap.Cells[1,17]:='2550';fgKap.Cells[2,17]:='0.189';
fgKap.Cells[1,18]:='2700';fgKap.Cells[2,18]:='0.187';
fgKap.Cells[1,19]:='2850';fgKap.Cells[2,19]:='0.183';
fgKap.Cells[1,20]:='3000';fgKap.Cells[2,20]:='0.18';
fgKap.Cells[1,21]:='3150';fgKap.Cells[2,21]:='0.195';
fgKap.Cells[1,22]:='3300';fgKap.Cells[2,22]:='0.174';
fgKap.Cells[1,23]:='3450';fgKap.Cells[2,23]:='0.188';
fgKap.Cells[1,24]:='3600';fgKap.Cells[2,24]:='0.17';
fgKap.Cells[1,25]:='3750';fgKap.Cells[2,25]:='0.183';
fgKap.Cells[1,26]:='3900';fgKap.Cells[2,26]:='0.182';
fgKap.Cells[1,27]:='4050';fgKap.Cells[2,27]:='0.179';
btnHitung.Enabled:=true;
end;

procedure TfrmHasil.edtJmlCapBankChange(Sender: TObject);
var i:integer;
begin
  if edtJmlCapBank.Text='' then
  begin
    fgKap.RowCount:=2;
  end
  else
  begin
    fgKap.RowCount:=StrToInt(edtJmlCapBank.Text)+1;
    for i:=1 to StrToInt(edtJmlCapBank.Text) do
    begin
      fgKap.Cells[0,i]:=IntToStr(i);
    end;
  end;
end;

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.
```

LAMPIRAN

✓ SURAT-SURAT PENGESAHAN



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

BERITA ACARA UJIAN SKRIPSI FAKULTAS TEKNOLOGI INDUSTRI

1. **Nama** : MOHAMMAD KHOIRUL ANAM
2. **NIM** : 01.12.089
3. **Jurusan** : Teknik Elektro S-1
4. **Konsentrasi** : Teknik Energi Listrik
5. **Judul Skripsi** : OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GI. PIER, PASURUAN

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

Hari : Selasa
Tanggal : 04 September 2007
Dengan Nilai : 74,9 (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 H. 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 : MOHAMMAD KHOIRUL ANAM
2. NIM : 01.12.089
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GL. PIER, PASURUAN

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Bagaimana menentukan biayanya?	

Anggota Penguji

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

Dosen Pembimbing I

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

Dosen Pembimbing II

Ir. Eko Nurcahyo
NIP. P. 1028700172



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 : MOHAMMAD KHOIRUL ANAM
2. NIM : 01.12.089
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI 20 kV MENGGUNAKAN METODE *BINARY PARTICLE SWARM OPTIMIZATION* (BPSO) DI GL. PIER, PASURUAN

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Abstrak disempurnakan	
2.	Tujuan penelitian diringkas	
3.	Keterangan gambar 1 spasi	
4.	Lengkapi halaman 64 dan 66	
5.	Catatan kaki dicantumkan, halaman 38 dan seterusnya	
6.	Daftar pustaka diperbaiki	

Anggota Penguji

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

Dosen Pembimbing I

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

Dosen Pembimbing II

Ir. Eko Nurcahyo
NIP. P. 1028700172

PERMOHONAN PERSETUJUAN SKRIPSI

betanda tangan dibawah ini :

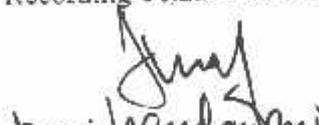
Nama : M KHOIRUL ANAM
No : 0112039
Kelas : XI
Jurusan : Teknologi Industri
Konsentrasi : Teknik Elektro S-1
Mata Kuliah : Teknik Elektronika / Teknik Energi Listrik
Tempat : DS CABEAN KIDUL, BULU, REMBANG, JATEM

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

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

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

Yang diteliti kebenaran data tersebut diatas
Recording Teknik Elektro

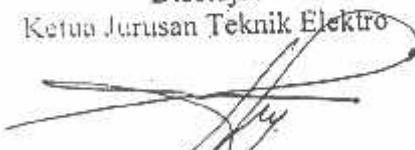

(..... Ir. F. Yudi Limpraptono))

Malang, DESEMBER 2006

Pemohon


(..... M. KHOIRUL ANAM))

Disetujui
Ketua Jurusan Teknik Elektro


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

Mengetahui
Dosen Wali

(.....)

catatan :

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

IPK 3.26/3.32 = 2.46

MC DS. Kony II → E

- praktikum PLC belum



LEMBAR PENGAJUAN JUDUL SKRIPSI JURUSAN TEKNIK ELEKTRO S-1

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

1.	Nama Mahasiswa: <u>M KHOIRUL AHAM</u>	Nim: <u>0112089</u>
2.	Waktu Pengajuan	Tanggal:
		Bulan: <u>DEBUARI</u>
Tahun: <u>2007</u>		
Spesifikasi Judul (berilah tanda silang)**)		
3.	a. <input checked="" type="checkbox"/> Sistem Tenaga Elektrik	e. <input type="checkbox"/> Elektronika & Komponen
	b. <input type="checkbox"/> Energi & Konversi Energi	f. <input type="checkbox"/> Elektronika Digital & Komputer
	c. <input type="checkbox"/> Tegangan Tinggi & Pengukuran	g. <input type="checkbox"/> Elektronika Komunikasi
	d. <input type="checkbox"/> Sistem Kendali Industri	h. <input type="checkbox"/> lainnya
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen*)	Ketua Jurusan
	<u>Ir. Taufik Hidayat, MT</u>	<u>Ir. F. Yudi Limpraptono, MT</u> NIP. P. 1039500274
5.	Judul yang diajukan mahasiswa:	<u>*Optimasi perancangan Rapsi for load tertinggi Distribusi Listrik 30 kv menggunakan Metode Drany Dan ke swan Optimalisasi</u>
6.	Perubahan judul yang disetujui Dosen sesuai materi bidang ilmu
Catatan:		
.....		
7.	Persetujuan Judul skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu	Disetujui Dosen 200

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

Form S-2

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

sentrasi : Teknik Energi Listrik/Teknik Elektronika

Nama Mahasiswa : MUHAMMAD FHOIBUL ANAM		Nim : 0112000	
Keterangan	Tanggal	Waktu	Tempat
Pelaksanaan	25 Mei 2007	10.00 s.d. 12.00	Ruang : 50108 201
Spesifikasi judul			
a. Sistem Tenaga Elektrik b. Energi & Konversi Energi c. Tegangan Tinggi & Pengukuran d. Sistem Kendali Industri		e. Elektronika & Komponen f. Elektronika Digital & Komputer g. Elektronika Komunikasi h. lainnya	
Judul Proposal yang diseminarkan Mahasiswa	OPTIMASI KEMAMPUAN KAPASITAS PADA TERAPEUTIS PADA PERALATAN OPTIMISASI (THER 2011) YANG MENGGUNAKAN PRINSIP PRIMER PARTICLE SIZES OPTIMIZATION TO 20 MHz. BERTUJUAN SUDAH PASIFIKASI		
Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian			
Catatan :			
Persetujuan Judul Skripsi			
Disetujui, Dosen Keahlian I	Disetujui, Dosen Keahlian II		
Mengetahui, Ketua Jurusan.	Disetujui, Calon Dosen Pembimbing ybs.		
Ir. F. Yudi Limpraptono, MT Nip. Y. 1039500274	Dr. H. YAUFIK Hidayat MT		

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



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

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

01 MALANG
MALANG

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

Malang, 08 Juni 2007

nomor : ITN-315/LTA/2/2007
inspirasi : -
 perihal : **BIMBINGAN SKRIPSI**
 kepada : Yth. Sdr. **Ir. H. TAUFIK HIDAYAT, MT**

Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

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

Nama : MOHAMMAD KHOIRUL. A
Nim : 0112089
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Kosentrasi : Teknik Energi Listrik S-1

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

25 Mei 2007 s/d 25 November 2007

Sebagai satu syarat untuk menempuh Ujian Sarjana Teknik, Jurusan
Teknik Elektro-S1
Demikian atas perhatian serta kerendahan hati baik kami sampaikan
terima kasih.



Ketua Jurusan
Teknik Elektro S-1

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

Tembusan Kepada Yth:

1. Mahasiswa Yang Bersangkutan
2. Arsip

Form. S4a



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

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

01 MALANG
MALANG

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

Malang, 08 Juni 2007

mor : ITN-316/I.TA/2/2007
npiran : -
ihal : **BIMBINGAN SKRIPSI**
pada : Yth. Sdr. Ir. **EKO NURCAHYO**

Dosen Pembimbing
Jurusan Teknik Elektro S-1
di
Malang

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

Nama : MOHAMMAD KHOIRUL. A
Nim : 0112089
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro S-1
Kosentrasi : Teknik Energi Listrik S-1

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

25 Mei 2007 s/d 25 November 2007

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

Demikian atas perhatian serta kerendahan hati kami sampaikan
terima kasih.



Ketua Jurusan
Teknik Elektro S-1

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

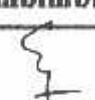
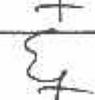
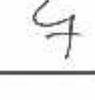
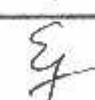
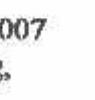
Tembusan Kepada Yth:

1. Mahasiswa Yang Bersangkutan
2. Arsip

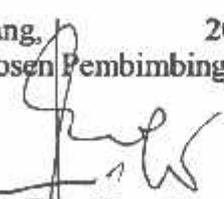
Form. S4a

FORMULIR BIMBINGAN SKRIPSI

Nama : MOHAMMAD KHOIRUL ANAM
 Nim : 01.12.089
 Masa Bimbingan : 25 Mei 2007 s/d 25 Nopember 2007
 Judul Skripsi : **OPTIMASI PENEMPATAN DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI PRIMER 20kV MENGGUNAKAN METODE BINARY PARTICLE SWARM OPTIMIZATION DI GL PIER PASURUAN**

Tanggal	Uraian	Parap Pembimbing
24 Mei '07	Selaraskan BAB I, II & III	
9 Juni '07	Revisi Bab III	
20 Juni '07	Flow chart Bab IV dan susunan Bab VI	
29 Juni '07	Revisi Bab IV	
4 Juli '07	Lampirkan ke Bab V	
15 Juli '07	Revisi latar belakang, Training program.	
20 Agustus '07	Tambahkan validasinya	
25 Agustus '07	ACC Manjulan seminar Uswi.	
26 Agustus '07	Revisi Bab I, tentukan masalah.	
29 Agustus '07	ACC Manjulan ujian komprehensif.	

Malang, 2007
 Dosen Pembimbing,


Ir. Eko Nurcahyo
 NIP. P. 1028700172

Form.S-4b

FORMULIR PENDAFTARAN UJIAN SKRIPSI

MAHASISWA :

	:	MOHAMMAD KHOTIBUL ANAM
	:	012084
is	:	Teknologi Industri
i	:	Teknik Elektro S-1
trasi	:	1. Teknik Energi Listrik *) 2. Teknik Elektronika *) 3. Teknik Komputer dan Informatika *)
di Malang	:	DEPUL KARTIKA ASPI < PUSKOPAD > 4-11 TUNJUNG SEKAP
Telp. Rumah / HP	:	085649241100
enulisan Skripsi	:	25 MEI 2007 s/d 25 NOPEMBER 2007
Pembimbing	:	1. Ir. H. TAUFIK Hidayat, MT. 2. Ir. CRO MURCAHYO.
skripsi	:	OPTIMASI PENEMPATAN DARI KAPASITAS KAPASITOR PADA TINGKATAN DISTRIBUSI 20 KV MENGGUNAKAN METODE BINARY PARTICLE SWARM OPTIMIZATION DI S-1 PER RUPUK

PERSYARATAN YANG HARUS DIPENUHI : **)

Persyaratan	Paraf ***)
Telah mengumpulkan 140 SKS dengan IPK ≥ 2	
Tidak ada Nilai E	
Telah menyelesaikan / mengumpulkan Laporan Praktek Kerja	
Telah menempuh semua praktikum yang disyaratkan Jurusan	
Menyerahkan Kartu Peserta Seminar Hasil	
Mengumpulkan fotokopi Skripsi yang telah ditandatangani Dosen Pembimbing angka 3 (tiga) eksemplar	
Menyerahkan Surat Pias Bimbingan Skripsi dari Dosen Pembimbing	
Telah melunasi persyaratan administrasi (kuitansi warna kuning Rp. 100.000,-)	

Mengetahui,
 Jurusan Teknik Elektro S-1

Yudi Limpraptono, MT
 NIP. Y. 103 95 00274

Malang, AGUSTUS 2007
 Mahasiswa Yang Bersangkutan

(MOHAMMAD KHOTIBUL ANAM)

n :

**PLN (PERSERO)
TRIBUSI JAWA TIMUR
AREA PELAYANAN & JARINGAN PASURUAN**

dirman No. 69 Pasuruan 67115

: (0343) 426515 s/d 426517

Telex :

Faximile : (0343) 426518

Alamat Kawat : PLN PSR

No.

0749/041/APJ-PSR/2007

12 JUL 2007

: *Persetujuan Survey.*

KEPADA Yth. :

**DEKAN FAKULTAS TEKNOLOGI INDUSTRI
I T N MALANG
Jl. Bendungan Sigura-gura No. 2
MALANG - 65145**

Menjawab surat Saudara Nomor : ITN-1330/III.TA-2/2/07 dan No.: ITN-1331/III.TA-2/2/07 tanggal 14 Juni 2007 perihal Survey, dengan ini disampaikan bahwa pada dasarnya kami dapat mengijinkan pelaksanaan Survey kepada mahasiswa Saudara sebagai berikut :

1. Nama : HADI SURYA
NIM : 01.12.095

2. Nama : M. KHOIRUL ANAM
NIM : 01.12.089

Adapun pelaksanaannya mulai tanggal 16 Juli 2007 sampai dengan selesai dan ditempatkan di Unit Pelayanan Dan Jaringan Bangil PT PLN (Persero) Area Pelayanan dan Jaringan Pasuruan.

Untuk penjelasan lebih lanjut, diminta yang bersangkutan menghubungi langsung **ASISTEN MANAJER SDM DAN ADMINISTRASI PT PLN (Persero) Area Pelayanan Dan Jaringan Pasuruan** Jl. Panglima Sudirman No. 69 Pasuruan dengan membawa salinan surat ini.

Perlu kami sampaikan bahwa apabila diperlukan data, kami hanya akan memberikan yang bersifat **TIDAK RAHASIA**.

Demikian, atas perhatiannya disampaikan terima kasih.

PH MANAJER
ASISTEN MANAJER PEMASARAN,



Ir. WAHJUDI

Tembusan Yth. :

- MANAJER UPJ BANGIL