

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



**ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN  
DISTRIBUSI RADIAL DENGAN METODE PARALLEL  
TABU SEARCH DI G.I SENGKALING  
PANYULANG PUJON**

**SKRIPSI**

*Disusun Oleh :*  
**FARIED SUJANTO**  
**NIM 98.12.010**

**SEPTEMBER 2005**



## LEMBAR PERSETUJUAN

**ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN  
DISTRIBUSI RADIAL DENGAN METODE PARALLEL  
TABU SEARCH DI G.I SENGKALING PENYULANG PUJON**

**SKRIPSI**

*Disusun Guna Melengkapi dan Memenuhi Syarat-Syarat  
Guna Mencapai Gelar Sarjana Teknik Elektro S-1*

**Disusun Oleh :**

**FARIED SUJANTO  
NIM. 98.12.010**

**Mengetahui,  
Ketua Jurusan Teknik Elektro S-1**

**Diperiksa dan Disetujui  
Dosen Pembimbing**

**(Ir. F. Yudi Limpraptono, MT)  
NIP. Y. 10 9500274**

**(Ir. H. Taufik Hidayat, MT)  
NIP. Y. 1018700151**

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

# LEMBAR PERSEMBAHAN

Special For



YA ALLAH SWT, atas semua rahmat dan  
hidayah-MU Serta kemudahan2 yang Engkau  
berikan kepada KU

Serta JunjunganQ Nabi MUHAMMAD SAW yang  
telah memberi sauri teladan kepada  
seluruh umat manusia

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

Ayah dan ibu yang tak prnh henti-hentinya mndoakan  
aku n Dukungan Finansialnya (maafin Q ya. DosaQ  
kpdMU Terlalu bnyk dan terima kasih banyak).

Embah uti maksih atas dukungan do'anya dan minta  
maaf ya mbah.....ya saya bnyk salah

KKq @si\_mba lin, mba yuyung, masamad TPhan's piz  
doo. Supan an bntakSayan kALAN chngg @ bisa  
menyerasakan kullah, semoga Allah SWT selalu  
memberikan Rahmet, hidayah pada @r smua

## SPECIAL THANK'S TO :

### Dosen Pembimbing

Bapak Ir. H. Taufik Hidayat, MT terima kasih buat bapak yang telah membimbing dan membantu saya saya selama skripsi dan meluangkan waktunya di sela-sela kesibukan kerja bapak mulai dari seminar proposal, seminar hasil hingga kompre dan mohon maaf atas kesalahan yg kami perbuat selama ini

Mas Jayeng → thank's buat bantuan mulai dri sminar smpe kompre mas... agadepin are-are ojo kereng-kereng ya

### Mas Uars Wiseno ST

progRam WaLoPu...  
maksh mas sss!! SeMoG  
KesehataN dan Rejeki yang  
Amiin

Mas...  
THUM'kaSiH  
Sakali

### Teman-teman sepeniuaan

Agus kantung → masah banyak bantuan garapin skripsi tung \*\* @ Iwan kholik → wah...makin garapn skripsiku n km

teman paling baik sukses aja buatkan n slalu ku ingt. @ oni

K → ...

deni → mks jek... bantuannya palidasi..

km plng hebat.. llyo opo komperere dol sing murah.. ta

he..he @jepri → akhirnya... lulus bareng jepri... yo

keluar yeko ITN \*\* @ dadang → ojo pacaran trus apa rabi

he..he... JPN YANG LAINNYA YG BELUM DISEBUT... mohon

maaf n sukses semua ya).

### Teman-teman '98

6/11 → ingar pantang meyerah jkn trus... ss \*\* @ mban dady →

cepat di garap skripsinya jgn smpe jd mbanze itn

merenge' he..he \*\* @ ...

arif bima → lapa oe di

os2n bentelér kab Jepang ingarap skripsinya sempet tuwe di

campus? Yang blm di sebutkan cba trus n sukses..sss!!

## ABSTRAKSI

### ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL DENGAN METODE PARALEL TABU SEACH DI G.I SENGKALING PENYULANG PUJON

( Faried Sujianto, Nim 98.12.010, Teknik Elektro/T.Energi Listrik)  
(Dosen Pembimbing : Ir. H. Taufik Hidayat, MT)

Kata Kunci : Sistem Distribusi Radial, Penempatan Kapasitor, Pengurangan Rugi-rugi, Node Sensitif, Fixed Capacitor, metode Paralel Tabu Seach.

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 komponen rugi-rugi daya, disamping itu dapat memperburuk kondisi tegangan. Untuk mereduksi rugi daya solusinya dengan pemasangan kapasitor. Dalam skripsi ini digunakan Strategi Pencarian Heuristik untuk penempatan kapasitor dan besar kapasitas kapasitor yang optimal pada jaringan. Perhitungan aliran daya Newton Raphson diterapkan untuk mengetahui perubahan hasil dari pemasangan kapasitor.

Pada penyulang Pujon 20 kV dipasang 3 buah kapasitor yaitu pada node 26 dengan kapasitas 450 kVAR, node 55 dengan kapasitas 450 kVAR dan node 98 dengan kapasitas 1200 kVAR. Besarnya penurunan rugi daya aktif adalah 24,492 kW dari sebelum pemasangan kapasitor 107,982 kW menjadi 83,490 kW, sedangkan untuk daya reaktif juga terjadi penurunan sebesar 41,223 kVAR dari sebelum pemasangan kapasitor 164,463 kVAR menjadi 123,240 kVAR Tegangan tiap node meningkat sedangkan aliran daya dan arus menurun pada setiap saluran.

Biaya tahunan sebelum pemasangan kapasitor sebesar 18.140 US \$ atau sebesar Rp. 163.260.000,-/tahun, sedangkan total biaya tahunan setelah pemasangan kapasitor sebesar 14.458 US \$ atau sebesar Rp. 130.122.000,-/tahun (asumsi 1 US \$ adalah Rp. 9.000,-) sehingga keuntungan yang di dapat adalah sebesar Rp. 33.138.000,-/tahun.

## KATA PENGANTAR

Dengan mengucapkan puji syukur kehadiran Tuhan Yang Maha Esa, atas rahmat dan karunia-Nya memungkinkan penulis dapat menyelesaikan skripsi yang berjudul, "*ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL DENGAN METODE PARALLEL TABU SEARCH DI G.I SENGKALING PENYULANG PUJON*".

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 tersusunnya skripsi ini, maka penulis menyampaikan terima kasih kepada :

1. **Dr. Ir. Abraham Lomi, MSEE**, selaku Rektor ITN Malang.
2. **Ir. Mochtar Asroni, MSME**, selaku Dekan FTI ITN Malang.
3. **Ir. F Yudi Limpraptono, MT**, selaku Ketua Jurusan Teknik Elektro (S1) ITN Malang.
4. **Ir. H. Taufik Hidayat, MT**, selaku dosen pembimbing.
5. **Ugro, ST**, sebagai *programmer*.
6. Kedua orang tuaku yang telah membesarkan, mendidik dan selalu mendoakanku sampai terwujud cita-citaku serta kakak-kakakku *thank's* ya atas doa dan bantuannya.

7. Rekan-rekan elektro energi listrik yang saling membantu dan saling mengisi dalam pengerjaan skripsi ini.

Akhirnya penulis mengharapkan skripsi dapat berguna dan bermanfaat bagi rekan-rekan mahasiswa khususnya pada jurusan Teknik Elektro konsentrasi Teknik Energi Listrik.

Malang, September 2005

Penulis

## DAFTAR ISI

	Halaman :
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PERSETUJUAN</b> .....	ii
<b>ABSTRAKSI</b> .....	iii
<b>KATA PENGANTAR</b> .....	iv
<b>DAFTAR ISI</b> .....	vi
<b>DAFTAR GAMBAR</b> .....	x
<b>DAFTAR GRAFIK</b> .....	xii
<b>DAFTAR TABEL</b> .....	xiii
<b>BAB I PENDAHULUAN</b> .....	1
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan .....	3
1.4. Batasan Masalah .....	3
1.5. Metodologi Penelitian .....	4
1.6. Sistematika Penulisan .....	5
1.7. Kontribusi Penelitian .....	6
<b>BAB II SISTEM DISTRIBUSI DAN APLIKASI KAPASITOR</b> .....	7
2.1. Sistem Distribusi Tenaga Listrik .....	7
2.1.1 Sistem Distribusi Primer (Jaringan Tegangan Menengah) .....	8
2.1.2 Sistem Distribusi Primer (Jaringan Tegangan Rendah) .....	8



2.2.	Struktur Jaringan Distribusi Tenaga Listrik .....	9
2.3.	Sistem Jaringan Distribusi Radial .....	9
2.3.1.	Sistem Radial Pohon .....	10
2.3.2.	Sistem Radial dengan Tie dan Switch Pemisah .....	11
2.3.3.	Sistem Radial dengan Pembagian Phase Area .....	11
2.3.4.	Sistem Radial Dengan Beban Terpusat .....	12
2.4.	Daya Reaktif dan Faktor Daya .....	13
2.5.	Kapasitor Daya .....	15
2.5.1.	Kapasitor Seri dan Kapasitor Shunt .....	15
2.5.2.	Faktor-Faktor pemilihan Kapasitor Seri dan Kapasitor shunt..	15
2.6.	Pengaruh Pemasangan Kapasitor Shunt .....	17
2.6.1.	Pengurangan Rugi-Rugi dengan Kapasitor Shunt .....	18
2.6.2.	Perbaikan Tegangan .....	19
2.6.3.	Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem .....	20
2.6.4.	Perhitungan Pengaruh Perbaikan Faktor Daya .....	22
2.7.	Sistem Per-Unit .....	23

<b>BAB III</b>	<b>ANALISA PENENTUAN LETAK DAN KAPASITAS</b>	
	<b>KAPASITOR PADA JARINGAN DISTRIBUSI PRIMER</b>	
	<b>20 KV TIPE RADIAL DENGAN METODE</b>	
	<b><i>PARALLEL TABU SEACH</i></b> .....	25
3.1.	Analisa Aliran Daya Jaringan Radial.....	25
3.1.1.	Tujuan .....	25
3.1.2.	Metode Newton Raphson.....	25
3.1.3.	Diagram Alir Aliran Daya Newton Raphson .....	29
3.2.	Metode Tabu Seach (TS) .....	30

3.2.1.	Perpindahan dan Seleksi .....	30
3.2.2.	Daftar Tabu .....	30
3.2.3.	Kriteria Aspirasi .....	31
3.3	Metode Parallel Tabu Search.....	32
3.4	Total Biaya Sebelum dan sesudah Pemasangan Kapasitor .....	33
3.4.1.	Biaya Daya ( $K_p$ ).....	34
3.5	Algoritma Penyelesaian Masalah .....	34

#### **BAB IV STUDI PENENTUAN LETAK DAN KAPASITAS**

##### **KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL**

<b>PENYULANG PUJON .....</b>	<b>36</b>	
4.1.	Analisis Penempatan Kapasitor Dengan Metode Parallel Tabu Search .....	36
4.2.	Flowchat Penyelesaian Masalah .....	37
4.3.	Data Perhitungan .....	39
4.3.1.	Data Saluran .....	39
4.3.2.	Data Pembebanan .....	41
4.3.3.	Data apasitas Kapasitor dan Harga.....	43
4.4	Analisa Perhitungan .....	44
4.4.1	Perhitungan Penghematan Biaya ( <i>Saving Cost</i> ).....	45
4.5	Hasil Program.....	47
4.5.1.	Tampilan Program.....	47
4.5.2.	Hasil penempatan kapasitor menggunakan Metode <i>Parallel Tabu Search</i> .....	53

**BAB V KESIMPULAN** ..... 58

5.1. Kesimpulan ..... 58

**DAFTAR PUSTAKA**

**LAMPIRAN**

## DAFTAR GAMBAR

Gambar :	Halaman :
2-1. Jaringan Distribusi Tegangan Menengah (JTM) Jaringan Tegangan Rendah dan Sambungan Rumah Ke Pclanggan .....	7
2-2. Contoh Sistem Distribusi Radial .....	9
2-3. Contoh Jaringan Distribusi Radial Dengan <i>Tie</i> dan <i>Switch</i> Pemisah .....	11
2-4. Contoh Jaringan Distribusi Radial Dengan Phase Area .....	12
2-5. Contoh Jaringan Distribusi Radial Dengan Beban Terpusat .....	13
2-6. Segitiga Daya .....	14
2-7. Segitiga Arus .....	14
2-8. Saluran Primer Dengan Beban Terpusat .....	18
2-9. Vektor Diagram Sebelum dan Sesudah Pemasangan Kapasitor Shunt Pada Jaringan .....	20
2-10. Perbaikan Faktor Daya .....	21
2-11. Diagram Fasor dan Sudut Daya Beban Distribusi .....	22
3-1. Diagram Alir Aliran Daya Newton Raphson.....	29
4-2. Tampilan Utama Program.....	47
4-3. Tampilan Utama Program (Data General).....	47
4-4. Tampilan Utama Program (Data Saluran).....	48
4-5. Tampilan Utama Program (Data Pembebanan).....	48
4-6. Tampilan Utama Program (Data Kapasitor).....	48

4-7.	Nilai Tegangan Tiap Node Sebelum Penempatan Menggunakan Metode Aliran Daya Newton Raphson .....	49
4-8	Hasil Aliran Daya Sebelum Penempatan Menggunakan Metode Aliran Daya Newton Raphson .....	49
4-9	Hasil Metode Parallel Tabu Search Dalam menentukan lokasi, Kapasitas kapasitor serta total biaya .....	50
4-10	Hasil Analisa Aliran Daya Newton Raphson setelah penempatan Kapasitor Nilai Tegangan Tiap Node .....	50
4-11	Hasil Analisa Aliran Daya Newton Raphson setelah penempatan Kapasitor Metode Parallel Tabu Search Nilai Daya dan Rugi-rugi Pada saluran .....	51
4-12	Single line diagram penyulang pujan sebelum penempatan kapasitor .....	58
4-13	Single line diagram penyulang pujan setelah penempatan Kapasitor metode parallel tabu search .....	59

## DAFTAR GRAFIK

Grafik :	Halaman :
4-1. Profil Tegangan Sebelum dan Sesudah Penempatan Kapasitor .....	56
4-2. Rugi Rugi Daya Sebelum dan Sesudah Pencmpatan Kapasitor .....	56
4-3. Nilai Biaya Sebelum dan Setelah Pemasangan Kapasitor Metode Parallel Tabu Search .....	57

## DAFTAR TABEL

Tabel :	Halaman :
2-1. Kapasitor Seri dan Kapasitor Shunt.....	16
4-1. Spesifikasi Saluran.....	39
4-2. Data Saluran Penyulang Pujon .....	40
4-3. Data Pembebanan Sistem 20 kV Penyulang Pujon .....	41
4-4. Ukuran dan Harga Kapasitor Yang Tersedia .....	43
4-5. Data Kapasitas Kapasitor dan Harga.....	49
4-6. Nilai Tegangan Sebelum dan Sesudah Pemasangan Kapasitor .....	52
4-7. Hasil Penempatan Kapasitor Metode Parallel Tabu Search.....	53

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Dengan meningkatnya usaha di sektor industri dan meningkatnya taraf hidup masyarakat maka kebutuhan akan energi listrik semakin meningkat pula, sehingga diperlukan penyediaan energi listrik 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 penerangan dan lain lain. Beban industri sangat banyak membutuhkan daya reaktif induktif. Dengan meningkatnya beban-beban induktif, maka memperbesar komponen rugi-rugi daya, disamping itu dapat memperburuk kondisi tegangan.

Alternatif yang sering dipakai untuk memperbaiki kondisi jaringan akibat adanya rugi-rugi tersebut adalah memasang daya reaktif tambahan disisi beban salah satunya adalah pemasangan kapasitor.

Banyak metode yang dipakai dalam menganalisa masalah penentuan penempatan dan besar kapasitor pada jaringan distribusi, teknik untuk masalah alokasi kapasitor bisa dikelompokkan menjadi empat kategori<sup>[4]</sup> : Analitis, Pemrograman Numerik, Heuristik dan berdasarkan metode Intelegensi Artifisial. Metode Intelegensi Artifisial saat ini merupakan metode yang paling menarik dimana metode ini didalamnya terdapat *Algoritma Genetik*, *Simulated Annealing*, *Expert System*, *Artificial Neural Network Fuzzy Set* dan *Tabu Search*. Salah satu metode yang akan dibahas dalam skripsi ini adalah metode *Parallel Tabu Search*.



Dalam skripsi ini metode *Parallel Tabu Search* diajukan untuk memperbaiki performa dari metode *Tabu search*.

## 1.2. Rumusan Masalah

Salah satu cara pemasangan sumber daya reaktif tambahan disisi beban adalah dengan pemasangan kapasitor pada jaringan distribusi primer, sehingga dapat mengurangi rugi-rugi disaluran. Yang menjadi permasalahan adalah bagaimana cara menentukan letak dan kapasitas kapasitor yang optimal pada jaringan distribusi radial agar rugi-rugi yang dihasilkan menjadi seminim mungkin dan juga menghemat biaya instalasi kapasitor tersebut yang semaksimal mungkin.

Oleh karena itu pada skripsi ini akan menganalisa hal tersebut dengan suatu alternatif metode *Tabu search* merupakan suatu algoritma yang dapat menentukan lokasi, jumlah, kapasitas dan biaya pemasangan kapasitor yang dipasang pada jaringan distribusi.

Dari permasalahan yang timbul diatas, maka skripsi ini berjudul:

**“ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN  
DISTRIBUSI RADIAL DENGAN METODE *PARALLEL TABU SEARCH*  
DI GARDU INDUK SENGKALING PENYULANG PUJON”**

---

### 1.3. Tujuan.

Adapun tujuan penulisan skripsi ini adalah

1. Menentukan letak dan kapasitas kapasitor yang optimal pada sistem distribusi radial penyulang Pujon.
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 setelah penempatan kapasitor.

### 1.4. Batasan Masalah

Untuk menyederhanakan masalah yang akan dibahas, maka diberikan asumsi-asumsi serta batasan-batasan sebagai berikut:

1. Jaringan yang akan dianalisa adalah jaringan distribusi radial 20 kV di penyulang Pujon
  2. Hanya Kapasitor Shunt yang digunakan dalam perbaikan tegangan dan pengurangan rugi daya.
  3. Analisa penentuan letak kapasitor dan pemilihan kapasitas kapasitor menggunakan metode *Parallel Tabu Search*.
  4. Tidak membahas secara detail metode aliran daya yang digunakan yaitu metode Newton Raphson.
  5. Di asumsikan faktor daya sebesar 0,86
-

### 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 dan pembahasan masalah

Analisa perhitungan tegangan, sudut fasa tegangan, rugi-rugi daya dan aliran daya tiap saluran menggunakan metode Newton Raphson, untuk menentukan penempatan kapasitor yang optimal menggunakan Metode *Parallel Tabu Search* yang disimulasikan dengan program komputer.

---

## 1.6. Sistematika Penulisan

Sistematika dari pembahasan di dalam skripsi ini adalah sebagai berikut :

### **BAB I : PENDAHULUAN**

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

### **BAB II : SISTEM DISTRIBUSI DAN APLIKASI KAPASITOR**

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

### **BAB III : ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL DENGAN MENGGUNAKAN METODE *PARALLEL TABU SEARCH***

Pada bab ini akan dibahas metode aliran daya Newton Raphson, teori-teori mengenai metode *Parallel Tabu Search* untuk menentukan penempatan dan kapasitas kapasitor, penentuan *saving cost* serta algoritma metode *Parallel Tabu Search*.

### **BAB IV : STUDI PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL DENGAN MENGGUNAKAN METODE *PARALLEL TABU SEARCH***

Pada bab ini akan dibahas mengenai penentuan letak dan kapasitas kapasitor menggunakan metode *Parallel Tabu Search*, 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.

### **1.7 Kontribusi Penelitian**

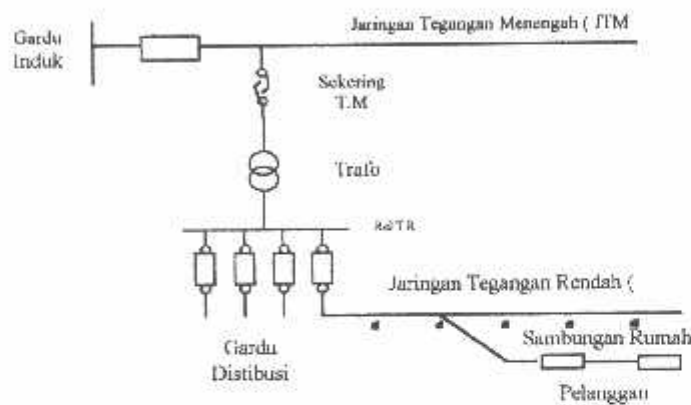
Dengan adanya penulisan skripsi ini diharapkan akan dapat memberikan alternatif penentuan jumlah, ukuran dan lokasi kapasitor yang optimal dengan lebih mudah sehingga menghasilkan perhitungan yang lebih valid (mendekati sempurna) dan mendapatkan nilai penghematan yang maksimal, sehingga kemungkinan dapat diaplikasikan di lapangan dan dapat membantu PT PLN (Persero) untuk mengoptimalkan penggunaan kapasitor dalam sistem distribusi radial.

---

## BAB II

### SISTEM DISTRIBUSI DAN APLIKASI KAPASITOR

#### 2.1. Sistem Distribusi Tenaga Listrik



**Gambar 2-1**  
**Jaringan Distribusi Tegangan Menengah (JTM),**  
**Jaringan Tegangan Rendah dan Sambungan**  
**Rumah Ke Pelanggan <sup>[1]</sup>**

Jaringan setelah keluar dari gardu induk ( G.I ) disebut jaringan distribusi. Setelah tenaga listrik disalurkan melalui jaringan distribusi primer kemudian tenaga listrik di turunkan tegangannya dalam gardu-gardu distribusi menjadi tegangan rendah, kemudian disalurkan melalui Jaringan Tegangan Rendah untuk selanjutnya disalurkan kerumah-rumah atau pelanggan (konsumen) PLN melalui sambungan rumah.

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.

#### **2.1.1. 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.

#### **2.1.2. 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.

---

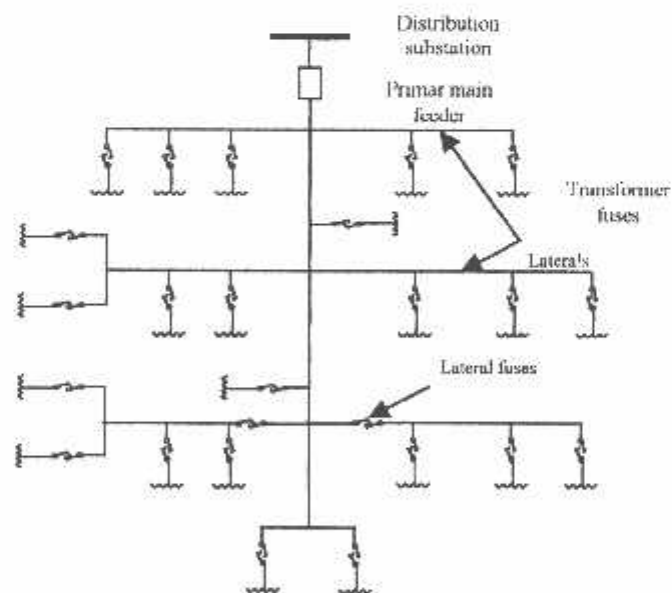
## 2.2. Struktur Jaringan Distribusi Tenaga Listrik

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

1. Sistem Jaringan Distribusi Radial
2. Sistem Jaringan Distribusi Tertutup (Loop)
3. Sistem Jaringan Distribusi Mesh.

## 2.3. 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 gardu induk ke pusat-pusat beban atau konsumen yang dilayani. Sistem ini terdiri dari aluran utama (tunk line) dan saluran cabang ( lateral ) seperti pada gambar 2-2.



**Gambar 2-2**  
**Contoh Sistem Distribusi Radial <sup>[1]</sup>**



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.

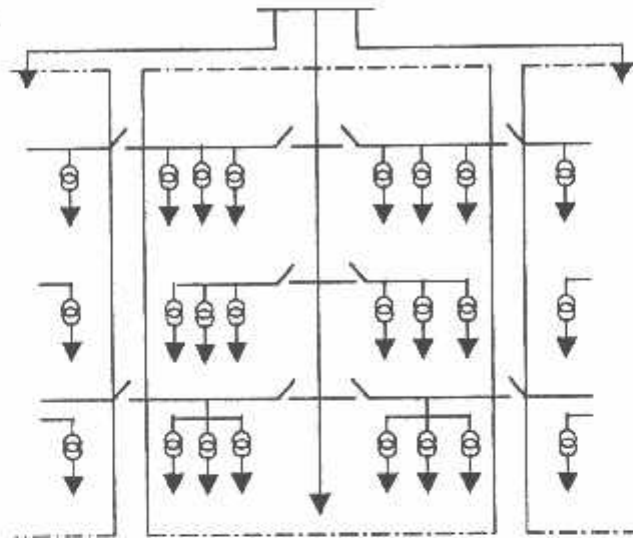
Kelemahan yang dimiliki oleh sistem radial adalah jatuh tegangan yang cukup besar dan bila terjadi gangguan pada sistem akan dapat mengakibatkan jatuhnya sebagian atau bahkan seluruh beban sistem.

### **2.3.1. Sistem Radial Pohon**

Sistem radial jaringan pohon ini merupakan bentuk yang paling dasar dari sistem jaringan radial. Saluran utama (main feeder) ditarik dari suatu gardu induk sesuai dengan kebutuhan kemudian dicabangkan melalui saluran cabang (lateral feeder). Ukuran dari masing-masing saluran tergantung dari kerapatan arus yang ditanggung. Dari gambar 2-2, main feeder merupakan saluran yang dialiri arus terbesar, selanjutnya arus mengecil pada tiap cabang tergantung dari besarnya beban.

---

### 2.3.2. Sistem Radial dengan Tie dan Switch Pemisah.



**Gambar 2-3**  
**Contoh Jaringan Distribusi Radial**  
**Dengan Tie dan Swicth Pemisah <sup>[2]</sup>**

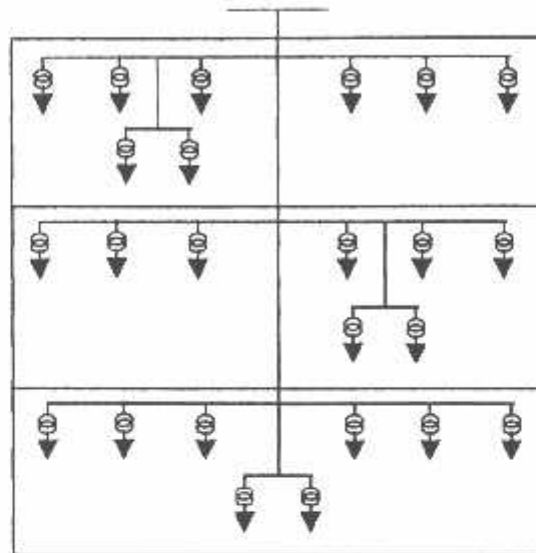
Sistem ini merupakan pengembangan dari sistem radial pohon, untuk meningkatkan keandalan sistem saat terjadinya gangguan maka feeder yang terganggu akan dialokalisasi sedangkan

Area yang semula dilayani feeder tersebut pelayanannya dialihkan pada feeder yang sehat atau yang tidak terganggu. Sistem radial dengan Tie dan Switch pemisah dapat dilihat pada gambar 2-3.

### 2.3.3. Sistem Radial dengan Pembagian Phase Area

Pada bentuk ini masing-masing fasa dari jaringan bertugas melayani daerah beban yang berlainan. Bentuk ini akan dapat menimbulkan kondisi sistem tiga fasa yang tidak seimbang (simetris), bila digunakan pada daerah beban yang baru

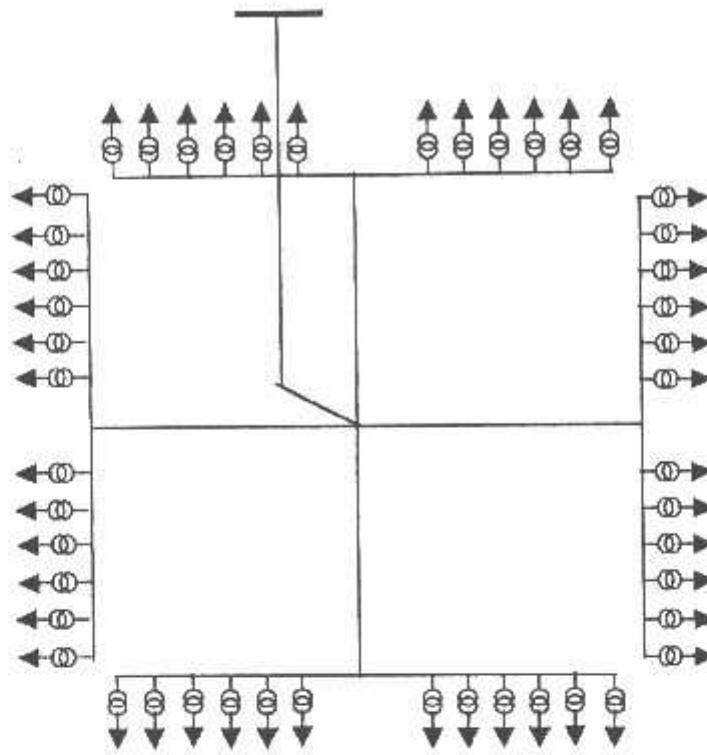
dan belum mantap pembagian bebannya. Contoh dari sistem jaringan ini dapat dilihat pada gambar 2-4.



**Gambar 2-4**  
Contoh Jaringan Distribusi Radial Dengan Phase Area <sup>[2]</sup>

#### 2.3.4. Sistem Radial dengan Beban Tepusat

Bentuk dari sistem ini mensuplai daya dengan menggunakan main feeder yang disebut express feeder langsung ke pusat beban dan dari titik pusat beban ini disebut dengan menggunakan back feeder radial seperti terlihat pada gambar 2-5.



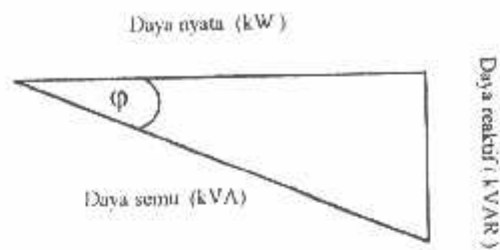
Gambar 2-5  
Contoh Jaringan Distribusi Radial Dengan Beban Terpusat <sup>(2)</sup>

#### 2.4. Daya Reaktif dan Faktor Daya<sup>(1)</sup>

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.

$$\text{Faktor daya} = \frac{\text{DayaNyata}(kW)}{\text{Dayasemu}(kVA)} \dots\dots\dots (2.1)$$

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<sup>[3]</sup>**

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

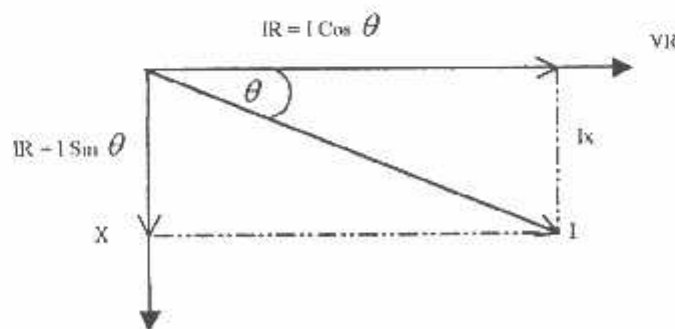
P = Daya Nyata (kW)

S = Daya semu (kVA)

Q = Daya reaktif (kVar)

$\varphi$  = Sudut Phase

Faktor daya dapat pula dilihat hubungannya dengan arus nyata dan arus total. Arus nyata ( $I_R$ ) adalah arus yang mengalir ada beban yang diubah menjadi tenaga. Sedangkan arus total ( $I$ ) adalah arus yang mengalir pada jaringan dan merupakan jumlah vektor antara arus magnetisasi dan arus nyata. Arus magnetisasi ( $I_X$ ) merupakan arus yang mengalir pada beban dan menimbulkan medan magnet. Arus yang terbaca dalam ampere meter adalah arus total ( $I$ ) hubungan ketiga arus tersebut seperti pada gambar 2-7 sebagai berikut :



**Gambar 2-7**  
**Segitiga Arus<sup>[3]</sup>**

## 2.5. Kapasitor Daya <sup>[3]</sup>

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 alumunium murni yang dipisahkan oleh tiga atau lebih lapisan kertas yang di lapisi oleh bahan kimia. Kapasitor daya telah mengalami perkembangan yang begitu cepat selama 30 tahun terakhir. Karena bahan elektik yang digunakan lebih efisien serta teknologi pembuatan kapasitor lebih baik.

### 2.5.1. Kapasitor seri dan Kapasitir shunt

Kapasitor daya terdiri atas 2 bagian yaitu kapasitor seri dan kapasitor shunt, dalam sistem tenaga membangkitkan daya reaktif untuk memperbaiki faktor daya dan tegangan, schingga meningkatkan kapasitas sistem dan mengurangi rugi-rugi jaringan

Biaya pemasangan kapasitor seri lebih tinggi dari pada biaya pemasangan kapasitor shunt, hal ini disebabkan karena peralatan pelindung untuk kapasitor seri lebih banyak, biasanya kapasitor seri di desain untuk daya yang lebih besar dari pada kapasitor shunt dengan tujuan untuk mengatasi kanaikan beban.

### 2.5.2. Faktor-faktor pemilihan kapasitor Seri dan kapasitor shunt <sup>[3]</sup> :

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

## 2.6. Pengaruh Pemasangan Kapasitor Shunt<sup>[3]</sup>

Kapasitor shunt adalah kapasitor yang dihubungkan paralel dengan saluran dan secara intensip 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
- Kapasitor Saklar

### a. Kapasitor Tetap

Adalah kapasitor untuk kompensasi 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

### b. Kapasitor Saklar

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 G.I. Untuk pengendalian secara otomatis, termasuk didalamnya peralatan pengendali

---

tegangan, arus dan suhu. Type yang paling populer adalah pengendalian saklar waktu (time-switch control), pengendali tegangan dan pengendali tegangan arus.

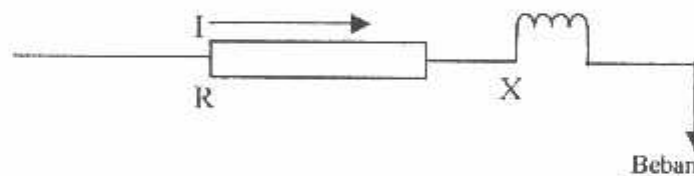
### 2.6.1. Pengurangan Rugi-Rugi dengan Kapasitor Shunt<sup>[3]</sup>

Rugi-rugi saluran per fase dari saluran 3 fase seimbang dengan beban terpusat seperti pada gambar 2-5 adalah  $I^2(R-jX)$  atau dapat dibedakan menjadi :

- rugi daya aktif  $(I^2R) = (I_R^2 + I_X^2)R$
- rugi daya Reaktif  $(I^2X) = (I_R^2 + I_X^2)X$

dimana :  $I_R$  adalah komponen arus aktif

$I_X$  adalah komponen arus reaktif



**Gambar 2-8**  
Saluran Primer Dengan Beban Tepusat<sup>[3]</sup>

Rugi-rugi daya ( $I^2R$ ) dapat dibagi menjadi dua komponen yaitu komponen arus aktif dan komponen arus reaktif.

Rugi-rugi karena komponen arus aktif tidak akan mempengaruhi penempatan kapasitor shunt pada saluran, hal ini dapat dijelaskan sebagai berikut :

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

$$I^2R = (I \cos \theta)^2R + (I \sin \theta)^2R \dots\dots\dots (2.2)$$



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

$$I_1^2 R = (I \cos \theta)^2 R + (I \sin \theta - I_c)^2 R \dots \dots \dots (2.3)$$

Sehingga pengurangan rugi daya sebagai akibat pemasangan kapasitor didapat :

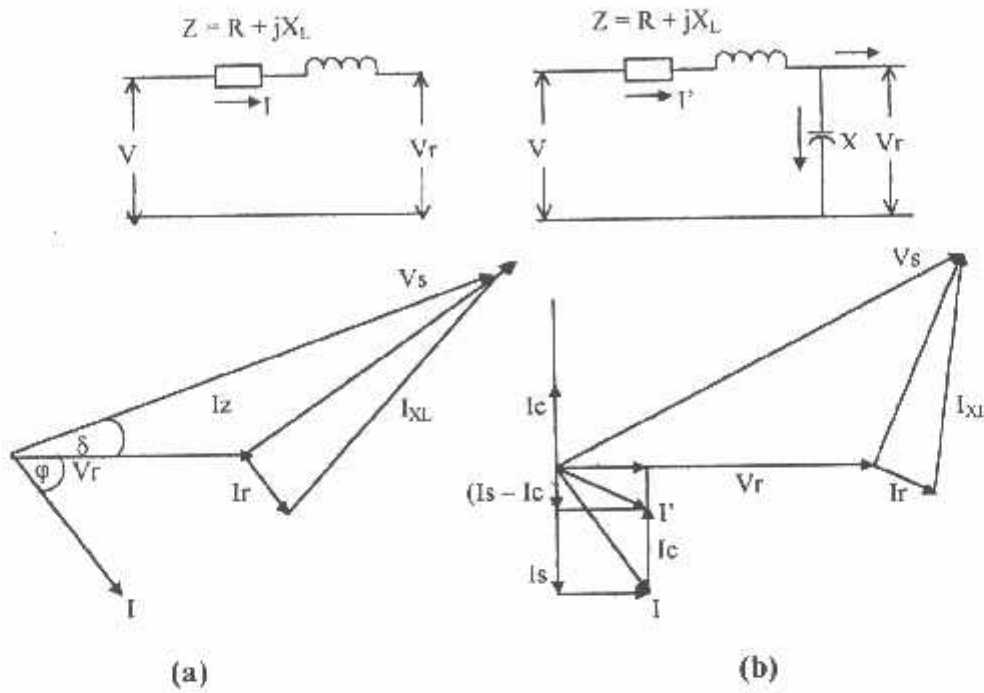
$$\begin{aligned} \Delta_{pls} &= I^2 R - I_1^2 R \\ &= (I \cos \theta)^2 R + (I \sin \theta)^2 R - (I \cos \theta)^2 R + (I \sin \theta - I_c)^2 R \\ &= 2 (I \sin \theta) I_c R - I_c^2 R \dots \dots \dots (2.4) \end{aligned}$$

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

$$3R[2(I \sin \theta)I_c - I_c^2] \text{ Watt} \dots \dots \dots (2.5)$$

### 2.6.2. Perbaikan Tegangan <sup>[3]</sup>

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



**Gambar 2-9**  
**Vektor Diagram Sebelum (a) dan Sesudah (b) Pemasangan**  
**Kapasitor Shunt Pada Jaringan<sup>(3)</sup>**

Kerugian jaringan disebabkan arus beban I sebelum kapasitor dipasang :

$$E = RI_r + XI_L \dots\dots\dots (2.6)$$

Kerugian tegangan setelah kapasitor dipasang :

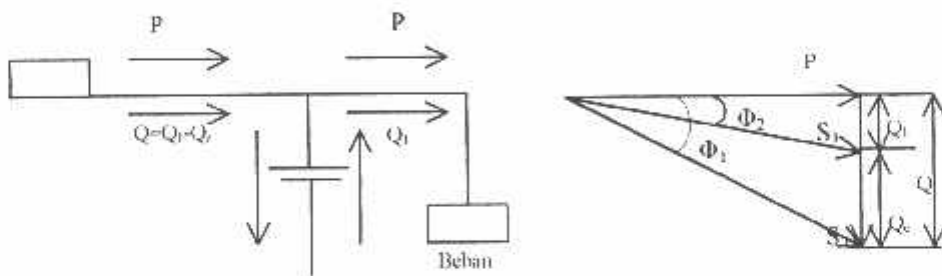
$$E = RI_r + XI_L - Xic \dots\dots\dots (2.7)$$

**2.6.3. Perbaikan Faktor Daya dan Kenaikan Kapasitas Sistem<sup>(3)</sup>**

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 2.10 menunjukkan perbaikan faktor daya pada sistem, kapasitor mensuplai daya reaktif ke beban.



**Gambar 2-10**  
Perbaikan Faktor Daya <sup>[3]</sup>

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

$$\cos \theta = \frac{P}{S_1} \dots \dots \dots (2.8)$$

Bila suatu kapasitor  $Q_c$  kVAR dipaang pada beban, faktor daya dapat diperbaiki dari  $\cos \theta_1$  menjadi  $\cos \theta_2$  dimana:

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

$$\cos \theta = \frac{P}{\sqrt{(P^2 + Q^2)}}$$

$$\cos \theta = \frac{P}{\sqrt{P^2 + (Q_1 - Q_c)^2}} \dots\dots\dots (2.9)$$

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

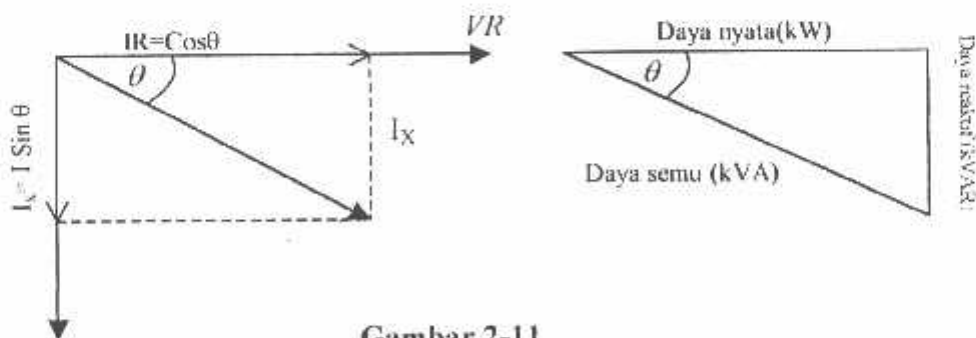
$$\% \text{ Rugi Daya} = 100 \left( \frac{\text{Faktor daya mula} - \text{mula}(\cos \theta_1)}{\text{Faktor daya baru}(\cos \theta_2)} \right) \dots\dots\dots (2.10)$$

%Pengurangan Rugi Daya

$$= 100 \left( 1 - \left( \frac{\text{Faktor daya mula} - \text{mula}(\cos \theta_1)}{\text{Faktor daya baru}(\cos \theta_2)} \right)^2 \right) \dots\dots\dots (2.11)$$

#### 2.6.4. Perhitungan Pengaruh Perbaikan Faktor Daya <sup>[3]</sup>

Diagram phase dari dua komponen arus nyata itu arus aktif dan reaktif dapat dilihat pada gambar 2-11 berikut;



**Gambar 2-11**  
Diagram Fasor dan Sudut Daya Beban Distribusi <sup>[3]</sup>

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

- Untuk data satu fasa

Arus dasar

$$I_d = \frac{kVA.dasar.1.fasa}{kV.dasar.L - N} \dots\dots\dots (2.13)$$

Impedansi dasar

$$Z_d = \frac{(kV.dasar.L - N)^2 * 1000}{kVA.dasar.1.fasa} \dots\dots\dots (2.14)$$

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

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

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

$$Z = \frac{\text{Impedansi sebenarnya } Z(\Omega)}{\text{Impedansi Dasar } Z_d} pu \dots\dots\dots (2.15)$$

## BAB III

# ANALISA PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI PRIMER 20 kV TIPE RADIAL DENGAN METODE *PARALLEL TABU SEARCH*

### 3.1. Analisa Aliran Daya Jaringan Radial <sup>31</sup>

Sebelum melakukan optimasi dengan menggunakan metode *Parallel Tabu Search* dilakukan suatu proses analisa aliran daya untuk mengetahui kondisi suatu sistem distribusi radial.

#### 3.1.1. Tujuan

Tujuan 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

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 :

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

Injeksi daya pada simpul  $p$  adalah :

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

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

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 ( $\delta$ ) 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 :

$\Delta P$  = selisih injeksi bersih daya nyata dengan penjumlahan aliran daya nyata tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan iterasi ke-k

$\Delta Q$  = selisih injeksi bersih daya reaktif dengan penjumlahan aliran daya reaktif tiap saluran yang menghubungkan simpul dengan V yang didapat dari perhitungan iterasi ke-k

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

$\Delta |V|$  = vektor koreksi magnitude tegangan

H, L, M, N merupakan elemen-elemen off diagonal dan diagonal dari sub matriks

Jacobian yang dibentuk dengan mendefinisikan persamaan (3.5) dan (3.6),

dimana:

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

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

Persamaan (3.7) diselesaikan untuk menghitung vektor koreksi magnitude tegangan  $\Delta(|V|)$  dan sudut fasa tegangan ( $\Delta \delta$ ) yang baru. Sehingga diperoleh harga magnitude tegangan dan sudut fasa yang baru, yaitu :

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



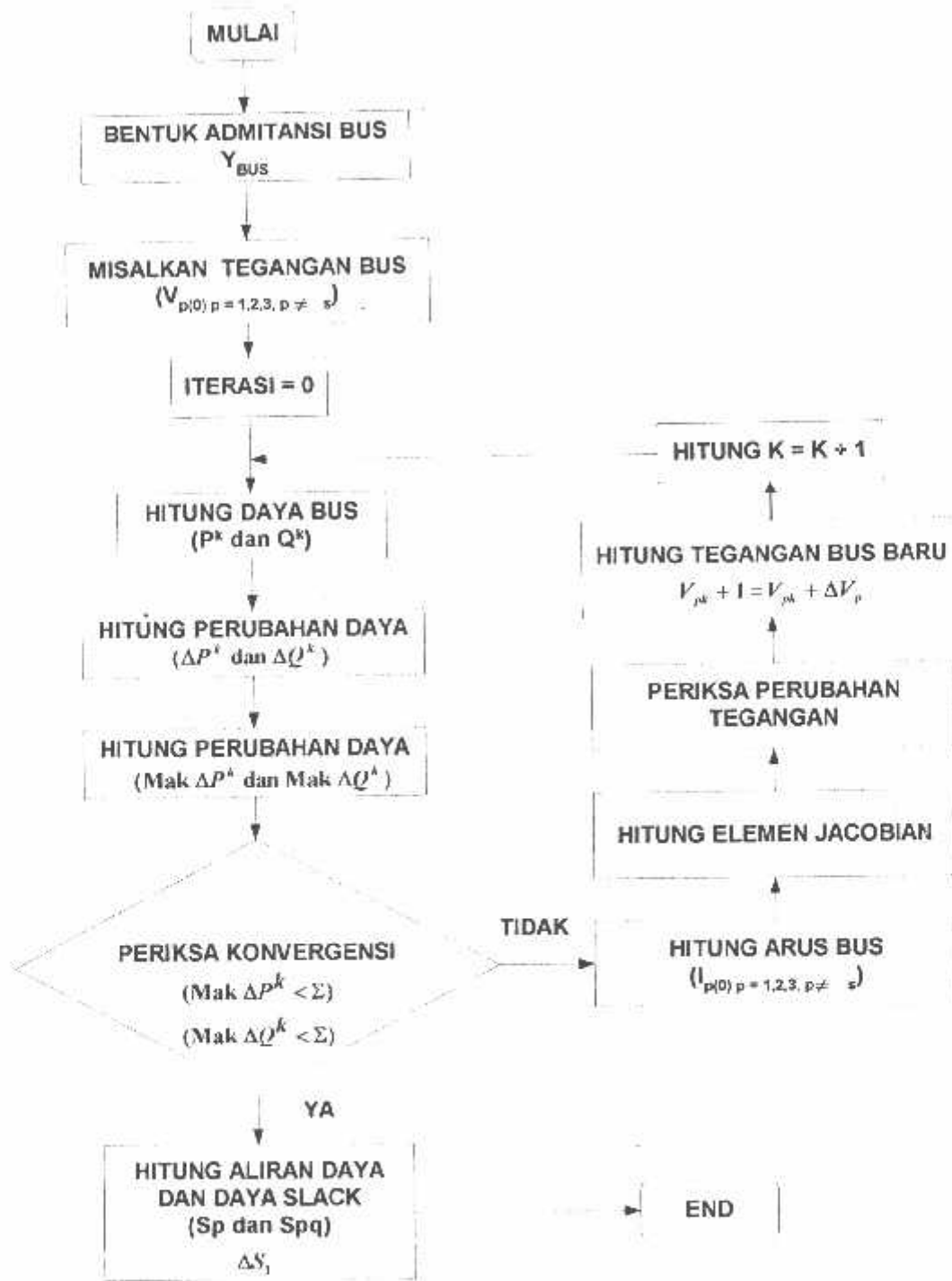
$$\delta^{k+1} = \delta^k + \Delta\delta^k \dots\dots\dots (3.9)$$

Proses perhitungan akan berulang sampai selisih daya nyata dan daya reaktif antara yang dijadwalkan dengan yang dihitung, yaitu  $\Delta P$  dan  $\Delta Q$  untuk semua simpul mendekati nilai toleransi atau proses perhitungan iterasi mencapai konvergen.

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 iterasi pertama atau tegangan yang ditentukan paling akhir untuk iterasi berikutnya.
2. Hitung  $\Delta P$  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  $\delta_q$  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  $\Delta P$  dan  $\Delta Q$  atau semua nilai  $\Delta\delta$  dan  $\Delta|V|$  lebih kecil dari suatu indeks ketetapan yang telah dipilih.

### 3.1.3. Diagram Alir Aliran Daya Newton Raphson



Gambar 3-1.

Diagram Alir Aliran Daya Newton Raphson<sup>[4]</sup>

### 3.2. Metode Tabu Search (TS) <sup>[5]</sup>

Pada prinsipnya metode *Tabu Search* ini merupakan suatu metode yang menampilkan suatu solusi yang bersifat coba-coba untuk mendapatkan hasil yang terbaik, dengan menggunakan program komputer.

#### 3.2.1. Perpindahan dan Seleksi

Langkah pertama dari *Tabu Search* (TS) adalah mendefinisikan sekelompok perpindahan yang mungkin diaplikasikan untuk solusi bersifat coba-coba, juga menghasilkan perpindahan yang baru. Diantara semua solusi yang dihasilkan, TS mencari satu perpindahan yang dapat memperbaiki sebagian besar fungsi obyektif. Dalam situasi tertentu, jika tidak ada perpindahan yang menjadi lebih baik yang berarti ada beberapa obyektif lokal, maka TS memilih satu diantara mereka yang memberikan fungsi obyektif.

Dalam problem penempatan kapasitor, proses pencarian dari solusi coba-coba ke yang lainnya disebut perpindahan. solusi coba-coba menunjukkan calon lokasi dari kapasitor, kapasitas terpasang, kehilangan daya, besarnya tegangan, dan nilai fungsi obyektif yang berkaitan dengan perpindahan ini.

#### 3.2.2. Daftar Tabu.

Guna menghindari kembali ke solusi yang baru saja dilakukan, maka perpindahan-perpindahan solusi yang telah dicoba harus diingat. Hal ini dilakukan dengan menyimpan perpindahan pada struktur data, seperti panjang struktur

---

pertama yang disebut daftar tabu. Elemen-elemen daftar tabu disebut perpindahan tabu. Segera setelah solusi coba-coba dibangkitkan, maka solusi coba-coba ini dicek apakah ada dalam daftar tabu atau tidak. Jika ada, maka ruang pencarian yang berhubungan akan dikurangi, begitu seterusnya.

Dimensi daftar tabu ini disebut dengan ukuran daftar tabu. Pilihan daftar tabu sangat penting. Jika ukurannya terlalu besar maka solusi dengan kualitas yang lebih tinggi tidak dapat dicapai. Sedangkan jika terlalu kecil maka perputaran kembali bisa terjadi dalam proses pencarian dan proses ini sering kembali kembali kesolusi yang baru saja dilakukan/ dikunjungi. Dalam skripsi kali ini menggunakan daftar perpaduan yang dimanis yang berperan sebagai daftar tabu, yang ukurannya adalah 20 untuk mencatat solusi yang baru saja dilakukan/dikunjungi. Hal ini berarti bahwa perpindahan akan disimpan dalam tabu untuk durasi waktu selama 20 perpindahan. Untuk perpindahan waktu yang keselanjutannya dicek apakah ada dalam daftar tabu apa tidak, jika ada maka periksa kembali apakah lebih baik dari yang ada dalam daftar tabu apa tidak, dan jika lebih baik diambil dan dipilih dari ketujuh yang ada dalam daftar tabu yang terjelek di buang dan apabila tidak proses dilanjutkan.

### **3.2.3. Kriteria Aspirasi**

Karena daftar tabu mengarahkan pada perpindahan tertentu yang menarik dan berharga yang mungkin mengarah kesolusi yang lebih baik daripada yang terbaik yang pernah ditemukan selama ini, oleh karena itu kriteria aspirasi

---

digunakan untuk memungkinkan perpindahan tabu yang diberikan jika perpindahan tersebut dinilai berharga atau menarik. Dengan kata lain, kriteria aspirasi adalah untuk mendapatkan perpindahan tabu yang sangat baik untuk diseleksi menjadi yang terbaik sebagai hasil optimal.

### 3.2. Metode *Parallel Tabu Search*

*Parallel Tabu Search* (PTS) mempertimbangkan kedua penyusunan ulang nilai sekitar dan keseragaman panjang tabu. Tabu search adalah salah satu metode yang efisien untuk metode penyelesaian masalah optimasi. Bagaimanapun Tabu search cenderung untuk meningkatkan upaya komputasi dalam komputasi. Nilai sekitar sebagai ukuran dari permasalahan yang lebih besar. Penyelesaian terbaik dalam keaslian nilai sekitar, dipilih melalui penyelesaian terbaik dalam subnilai sekitar. Dengan konsep penguraian kembali nilai sekitar, perkalian panjang tabu. Perkalian panjang tabu maksudnya penyetelan panjang tabu berbeda untuk membuat bermacam-macam petunjuk pencarian. Tabu search hanya mempunyai satu parameter yang disebut panjang tabu. Untuk menaikkan efisiensi Tabu search, panjang tabu perkalian diperkenalkan untuk menghasilkan penyelesaian yang lebih baik. Perbedaan panjang tabu memberikan penyelesaian pada masing-masing iterasi.

*Algoritma Parallel tabu search :*

Step 1 : Menyetel kondisi awal; menghitung iterasi  $k=0$ , solusi awal  $X_0$

---

- Step 2 : Memperbarui  $k = k+1$  untuk panjang tabu  $T_l(L=1, \dots, q)$  dimana  $q$  adalah jumlah total panjang tabu.
- Step 3 : Membuat subnilai sekitar  $N_i^{sub} (i=1, \dots, p)$  sekitar  $X_{k-1}$
- Step 4 : Mengevaluasi penyelesaian terbaik  $X_k^l (L=1, \dots, q)$  dalam masing-masing subnilai sekitar  $N_i^{sub}$
- Step 5 : Menuju step 7 jika sudah tercapai penyatuan criteria. Sebaliknya, menuju Step 6
- Step 6 : Memperbarui daftar tabu dan menuju Step 2
- Step 7 : Menghitung penyelesaian terbaik dalam  $X_k^l$
- Step 8 : Stop

Step 3 dan 4 tempat bidang maksudnya komputasi parallel untuk masing-masing subnilai

### 3.4. Total Biaya Sebelum dan Setelah Pemasangan Kapasitor<sup>[5]</sup>

Total biaya sebelum pemasangan kapasitor merupakan hubungan antara rugi daya yang ada di saluran sebelum dipasang kapasitor dan biaya daya per tahun. Hubungan tersebut seperti persamaan dibawah ini:

$$F = K_p \cdot P_{\text{loss},T} \dots \dots \dots (3.12)$$

Sehingga total biaya setelah pemasangan kapasitor adalah hubungan antara rugi-rugi daya pada jaringan setelah dipasangnya kapasitor dengan biaya instalasi kapasitor tersebut. Hubungan tersebut akan dijelaskan pada persamaan di bawah ini:

$$F = K^P + P_{\text{loss}} + \sum_{j=1}^k K_j^C Q_j^C \quad \dots\dots\dots(3.13)$$

Dimana:

- F = Total biaya tahunan setelah pemasangan kapasitor(\$/tahun)
- $K^P$  = Biaya daya (\$/kW/tahun)
- $P_{\text{loss}}$  = Total rugi daya saluran
- $K^C$  = Biaya instalasi kapasitor (\$/kVar/tahun)
- $Q^C$  = Kapasitas terpasang

### 3.4.1. Biaya Daya ( $K_p$ )<sup>[5]</sup>

Biaya daya ( $K_p$ ) adalah biaya investasi per-kW ditambah biaya operasi dan pemeliharaan. Dalam skripsi ini berdasarkan referensi maka biaya daya ( $K_p$ ) diset yaitu sebesar U.S \$ 168/kW/Tahun.

## 3.5. ALGORITMA PEMECAHAN MASALAH<sup>[6]</sup>

Penyelesaian perhitungan tegangan dan sudut fasa tegangan serta rugi daya tiap saluran menggunakan aliran daya metode *Newton Raphson*, sedangkan penentuan letak dan kapasitas kapasitor menggunakan metode *Parallel Tabu Search*. Urutan langkah-langkah yang digunakan dalam program komputer dapat dilihat pada algoritma program sebagai berikut :

1. Buat program *loadflow* untuk menghitung tegangan bus sebagai dan rugi-rugi setempat (*sectional – losses*).
2. Cari fungsi dari tegangan, rugi daya aktif dan *decision* untuk *parallel tabu search*.

3. Identifikasi kandidat node pada bus dengan fungsi membership  $\mu_s$  bus  $K$  (*bus yang telah dipilih*).
  4. Instal kapasitor pada bus  $K$  dengan merubah ukuran dalam *integer steps*. Pilih  $Q_c$  yang memiliki *cost* lebih rendah tanpa melanggar batas ( $Q_c \text{ max}$ ).
  5. Tambah  $Q_c$  pada bus  $K$  dan melakukan aliran daya kembali. Keluar jika tidak terdapat pelanggaran tegangan, menuju ke langkah 2. Nilai batasan tegangan adalah .
  6. Mengasumsikan bahwa  $M$  buses telah dipilih untuk menempatkan kapasitor baru, mengatur kapasitor pertama ( $i=1$ ) dalam *integer step* saat memasang yang lain. Pilih  $Q_c$  untuk bagian pertama yang memiliki *cost* terendah.
  7. Ulangi langkah 6 jika fungsi *cost* masih tinggi, fungsi *cost* adalah biaya operasi dalam setahun.
-



## BAB IV

### STUDI PENENTUAN LETAK DAN KAPASITAS KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL PENYULANG PUJON

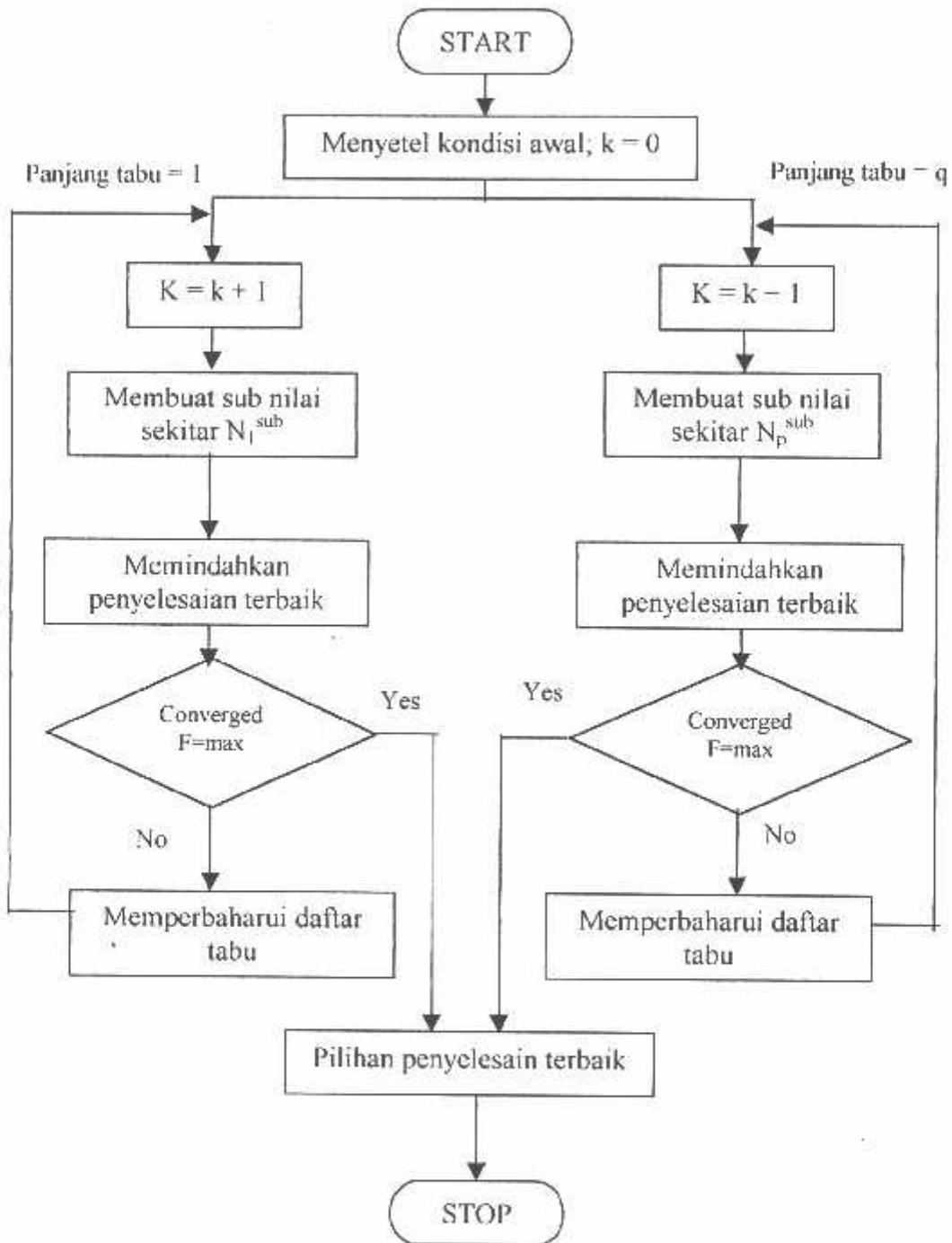
#### 4.1. Analisa Penempatan Kapasitor Dengan Metode *Parallel Tabu Search*.

Dalam memilih suatu metode solusi untuk aplikasi praktis sering sulit, pilihan itu memerlukan analisis yang cermat atas kelebihan-kelebihan dan kekurangan-kekurangan dari metode yang ada. Untuk solusi dari permasalahan diatas maka digunakan suatu alternatif dengan menggunakan metode *Parallel Parallel Tabu Search* yang menyediakan cara yang lebih baik dalam menganalisa masalah penempatan kapasitor.

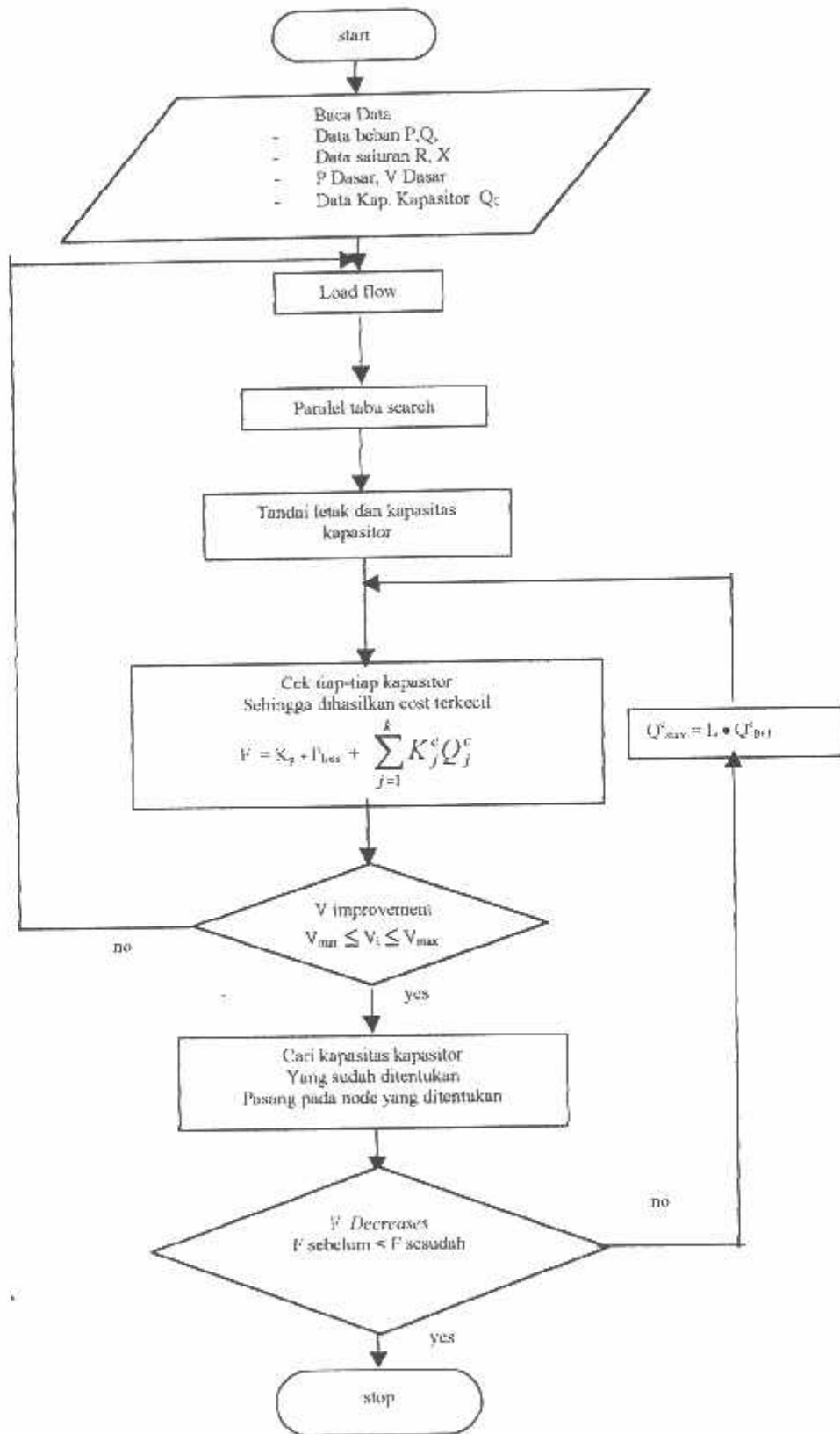
Dalam perhitungan analisa penempatan kapasitor disini menggunakan program komputer. Pada prinsipnya penggunaan program komputer dapat menganalisa sebuah jaringan distribusi radial dengan jumlah cabang atau jumlah bus yang tidak terbatas, tergantung dari memori yang tersedia pada komputer yang digunakan.

## 4.2. Flowchart Penyelesaian Masalah

### A. Flowchart *Parallel Tabu Search*



## B. Flowchart Penempatan Kapasitor



### 4.3. Data Perhitungan

Dalam analisa ini digunakan data saluran distribusi radial Penyulang Pujon. Data diperoleh dari PT. PLN (PERSERO) Distribusi Area Malang UP-J - Batu Malang,.

Untuk menyelesaikan perhitungan aliran daya terlebih dahulu ditetapkan single line diagram yang akan dianalisa. *Single line* penyulang Pujon dapat dilihat pada lampiran. Agar memudahkan perhitungan maka digunakan sistem per-unit (pu), dimana dasar yang digunakan :

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

#### 4.3.1. Data Saluran

Jaringan distribusi Penyulang Pujon menggunakan kabel saluran udara dengan spesifikasi pada tabel.1

**Tabel 4.1**  
**Spesifikasi Saluran**

Jenis Konduktor	Penampang nominal (mm <sup>2</sup> )	Impedansi saluran ( $\Omega$ /km)
AAAC	150	0.2162 + j 0.3305
AAAC	120	0.2688 + j 0.3376
AAAC	70	0.4608 + j 0.3572
AAAC	50	0.6452 + j 0.3678
AAAC	35	0.9217 + j 0.3790

Sumber PT. PLN (Persero) Distribusi JATIM  
Area Malang UP - J Batu

Data saluran penyulang Pujon dapat dilihat pada tabel 4.2

Tabel 4.2  
DATA SALURAN PENYULANG PUJON

No	Diri	Ke	Panjang	Impedansi Saluran	
Saluran	Node	Node	(m)	R	X
1	1	2	500.3	0.1265	0.3918
2	2	3	1432.8	0.3067	0.4735
3	3	4	683.8	0.1865	0.2655
4	4	5	65.4	0.0165	0.0252
5	4	6	683.3	0.1477	0.2258
6	5	7	2062.8	0.4503	0.6884
7	7	8	154.4	0.0334	0.051
8	8	9	814.5	0.0699	0.157
9	9	10	896.5	0.1933	0.2963
10	10	11	483.5	0.1002	0.1532
11	11	12	462	0.0677	0.1469
12	12	13	1424.5	0.306	0.4708
13	12	14	1233	0.2686	0.4075
14	7	15	1244.6	0.2881	0.4114
15	15	16	50.5	0.0196	0.0296
16	16	17	20.4	0.0044	0.0067
17	15	18	330	0.0731	0.1117
18	18	19	182.2	0.040	0.0615
19	18	20	402.8	0.1083	0.138
20	18	21	304.3	0.0656	0.1006
21	21	22	293.7	0.079	0.0962
22	21	23	1257.4	0.2719	0.4166
23	23	24	400.5	0.0679	0.1364
24	23	25	154.2	0.0366	0.0567
25	23	26	627.4	0.5783	0.2378
26	26	27	6.4	0.0050	0.0024
27	26	28	204.2	0.1062	0.0774
28	28	29	1145	1.0582	0.4343
29	29	30	668.2	0.1877	0.2686
30	30	31	741.2	0.0308	0.0487
31	31	32	668.6	0.1635	0.2605
32	31	33	272.5	0.1758	0.1002
33	33	34	1127.05	0.7272	0.4145
34	34	35	102.2	0.0650	0.0378
35	34	36	496.35	0.1073	0.164
36	36	37	262.5	0.0611	0.0934
37	37	38	294.5	0.0637	0.0873
38	31	39	643.3	0.039	0.3118
39	39	40	302.31	0.1961	0.1112
40	39	41	648.11	0.2182	0.1812
41	41	42	105.9	0.0229	0.035
42	39	43	40.5	0.0068	0.0134
43	43	44	1653.8	0.3575	0.5465
44	44	45	785.2	0.1666	0.2585
45	45	46	335	0.0739	0.111
46	46	47	783.2	0.1715	0.2622
47	47	48	571	0.1295	0.1867
48	48	49	365.5	0.079	0.1206
49	49	50	729.9	0.1578	0.2412
50	49	51	54.1	0.0117	0.0179

No	Diri	Ke	Panjang	Impedansi Saluran	
Saluran	Node	Node	(m)	R	X
51	51	52	121.6	0.0280	0.0400
52	51	53	840.35	0.2033	0.3106
53	53	54	1215.3	0.2904	0.6127
54	54	55	1192.7	0.5466	0.428
55	55	56	1123.5	0.2429	0.3713
56	56	57	1953.9	0.2926	0.4673
57	57	58	1067.1	0.2367	0.3627
58	56	59	398.5	0.0657	0.131
59	59	60	296.6	0.0642	0.0681
60	59	61	1093.4	0.2364	0.3614
61	61	62	1182.8	0.2514	0.3642
62	44	63	497.1	0.1075	0.1843
63	63	64	570.4	0.1230	0.1905
64	64	65	907	0.1961	0.2696
65	65	66	1789.6	0.3628	0.5648
66	63	67	135.2	0.0262	0.0447
67	67	68	621.6	0.1776	0.2715
68	67	69	110.15	0.0238	0.0384
69	69	70	891.3	0.327	0.2948
70	70	71	603.2	0.3368	0.1994
71	71	72	27.3	0.0059	0.0005
72	72	73	233.1	0.0504	0.077
73	72	74	258.7	0.0559	0.0658
74	74	75	198.1	0.0428	0.0653
75	75	76	1536.1	0.3325	0.5083
76	76	77	381.43	0.0885	0.1281
77	76	78	472	0.1021	0.158
78	78	79	469.6	0.0672	0.1488
79	79	80	721.4	0.158	0.2384
80	80	81	640.5	0.2033	0.3108
81	81	82	1428.3	0.3068	0.4721
82	75	83	322.1	0.0896	0.1066
83	83	84	186.4	0.0403	0.0616
84	83	85	96.5	0.0213	0.0305
85	85	86	243.9	0.0527	0.0806
86	85	87	305.3	0.0444	0.0679
87	87	88	590.3	0.1278	0.1951
88	88	89	656.6	0.1852	0.2831
89	89	90	458.3	0.0991	0.1615
90	90	91	1182.2	0.2513	0.3641
91	89	92	207.4	0.0446	0.0685
92	92	93	1061.5	0.225	0.3508
93	93	94	1321.52	0.6527	0.4661
94	93	95	365.1	0.2485	0.1416
95	95	96	476.9	0.3086	0.1759
96	95	97	332.8	0.2146	0.1223
97	97	98	1161.5	0.7494	0.4272
98	98	99	100.4	0.0217	0.0332
99	99	100	785.6	0.507	0.286
100	100	101	285.3	0.1841	0.1046

Sumber PT. PLN (Persero) Distribusi JATIM  
Area Malang UP - J Batu

### 4.3.2. Data Pembebanan

Data pembebanan diperoleh dengan mengambil data dari masing-masing trafo distribusi, data pembebanan dapat dilihat pada tabel 3

Tabel 4.3

#### DATA PEMBEBANAN SISTEM 20 kV PENYULANG PUJON

No	Kode	Lokasi	Pembebanan			Type
			S (kVA)	P (kW)	Q (kVAR)	
1			0,0000	0,0000	0,0000	Slack
2	T55	Jl Raya Ngandut	52	44,7200	28,5304	Load
3	T43	Jl Raya Mojorejo	70	60,2000	35,7140	Load
4			0	0,0000	0,0000	Load
5	T63	Dk Mantung	43	36,9800	21,9388	Load
6	T38	Jl Raya Beji	32	27,5200	16,3264	Load
7			0	0,0000	0,0000	Load
8	T72	emanuel temas	63	54,1800	32,1426	Load
9	T115	Jl Wukir Temas	0	0,0000	0,0000	Load
10	T97	Ds Temas Klerek	142	122,1200	72,4484	Load
11	T95	Ds Torong Rejo	93	79,9800	47,4486	Load
12			0	0,0000	0,0000	Load
13	T96	Ds Torong Rejo tutup	80	68,8000	40,8160	Load
14	T158	Ds wukir	0	0,0000	0,0000	Load
15			0	0,0000	0,0000	Load
16	T31	Jl Sudiro	38	32,6800	19,3876	Load
17	T126	Jl Sudiro	138	118,6800	70,4078	Load
18			0	0,0000	0,0000	Load
19	T105	R.s Paru-paru	73	62,7800	37,2446	Load
20	T189	Jl Kasiman	78	67,0800	39,7856	Load
21			0	0,0000	0,0000	Load
22	T188	Jl Lest	99	77,4000	45,9180	Load
23			0	0,0000	0,0000	Load
24	T9	Jl Hasanudin	167	143,6200	88,2034	Load
25	T32	Jl Indragiri	84	72,2400	42,8568	Load
26			0	0,0000	0,0000	Load
27	T46	Ds. Sumberejo	113	97,1800	57,8526	Load
28	119	Ds. Sumberejo	0	0,0000	0,0000	Load
29	78	Ds Satrehan	122	104,9200	62,2444	Load
30	25	Jl Trunojoyo	26	22,3800	13,2652	Load
31			0	0,0000	0,0000	Load
32	30	Ds Senggoriti	39	33,5400	19,8978	Load
33	T129	Jl Flamboyan	110	94,6000	58,1220	Load
34			0	0,0000	0,0000	Load
35	99	Jl Flamboyan	67	57,6200	34,1834	Load
36	200	Ds Tanbuh	10	8,6000	5,1020	Load
37	201	Ds Tanbuh	8	6,8800	4,0816	Load
38	202	Ds Tanbuh	2	1,7200	1,0204	Load
39	0		0	0,0000	0,0000	Load
40	168	Bukti Pinus	12	10,3200	6,1224	Load
41	174	Jl Senggoriti	94	80,6400	47,8588	Load
42	175	Jl Senggoriti	16	13,7600	8,1632	Load
43	16	Jl Senggoriti	71	61,0600	36,2242	Load
44			0	0,0000	0,0000	Load
45	57	Dk Sebaluh	65	55,9000	33,1630	Load
46	153	Dk Sebaluh	83	71,3800	42,3468	Load

47	59	Dk Maron	69	59,3400	35,2038	Load
48	87	Dk Maron	65	55,8000	33,1630	Load
49			0	0,0000	0,0000	Load
50	112	Ds Gunung Sari	55	47,3000	28,0610	Load
51			0	0,0000	0,0000	Load
52	88	Ds Pujon Kidul	34	29,2400	17,3468	Load
53	89	Ds Pujon Kidul	103	88,5800	52,5506	Load
54	101	Dk Biyan	33	28,3800	16,8368	Load
55	102	Dk Bakir	82	70,5200	41,8364	Load
56			0	0,0000	0,0000	Load
57	103	Dk Cukai	38	32,6800	19,3876	Load
58	167	Dk Dadapan	12	10,3200	6,1224	Load
59			0	0,0000	0,0000	Load
60	104	Ds Cukai	45	38,7000	22,8580	Load
61	168	Dk Trates	23	19,7800	11,7346	Load
62	167	Dk Dadapan	12	10,3200	6,1224	Load
63			0	0,0000	0,0000	Load
64	58	Dk Pandemas	36	30,9600	18,3672	Load
65	80	Ds Jurang Rejo	120	103,2000	61,2240	Load
66	217	Ds Brau	14	12,0400	7,1428	Load
67			0	0,0000	0,0000	Load
68	212	Dk Don Bayi	16	13,7600	8,1632	Load
69	36	Ds Pandesari	108	92,8800	55,1016	Load
70	145	Dk Watu Gong	47	40,4200	23,9794	Load
71	19	Jl Raya Pujon	83	71,3800	42,3466	Load
72			0	0,0000	0,0000	Load
73	198	Pasar Baru	55	47,3000	28,0610	Load
74	123	Jl Raya Pujon	105	90,3000	53,5710	Load
75			0	0,0000	0,0000	Load
76			0	0,0000	0,0000	Load
77	83	Dk Kalangan	70	60,2000	35,7140	Load
78	79	Ds Wiyun Rejo	123	105,7800	62,7546	Load
79	80	Ds Madrejo	47	40,4200	23,9794	Load
80	223	Dk Sebo	16	13,7600	8,1632	Load
81	94	Dk Delik	54	46,4400	27,5508	Load
82	218	Dk Delik	77	66,2200	39,2854	Load
83			0	0,0000	0,0000	Load
84	195	KOP SAE	29	24,9400	14,7858	Load
85			0	0,0000	0,0000	Load
86	82	KOP SAE	114	98,0400	58,1628	Load
87	33	Ds Ngroto	132	113,5200	67,3464	Load
88	61	Ds Ngroto	72	61,8200	36,7344	Load
89			0	0,0000	0,0000	Load
90	64	Dk Jebaksari	36	30,9600	18,3672	Load
91	147	Dk Torong	29	24,9400	14,7858	Load
92	63	Dk Mantung	43	36,9800	21,9386	Load
93			0	0,0000	0,0000	Load
94	196	Dk Bunder	16	13,7600	8,1632	Load
95			0	0,0000	0,0000	Load
96	197	Dk Ngebrong	16	13,7600	8,1632	Load
97	65	Ds Ngabab	123	105,7800	62,7546	Load
98	66	Ds Ngabab	63	54,1800	32,1426	Load
99	219	Ds Manting	46	39,5600	23,4682	Load
100	78	Ds Manting	0	0,0000	0,0000	Load
101	77	Dk Gerih	100	96,0000	51,0200	Load

Sumber PT.PLN (Persero) Distribusi JATIM  
Area Malang UP - J Batu

### 4.3.3. Data Kapasitas Kapasitor dan Harga (Cost/ kVAR)<sup>[5]</sup>.

Pada analisa menentukan nilai biaya, nilai kapasitas kapasitor dan harga sudah ditentukan dalam U.S.S /kVAR. Tabel 4.4 menunjukkan data yang tersedia oleh *suplier* untuk *feeder* distribusi. 27 kapasitas kapasitor sebagai pilihan ditunjukkan pada tabel 4.5 berdasarkan Cost/kVAR, nilai berasal dari tabel 4.4 dengan asumsi *life expectancy* selama 10 th.

**Tabel 4.4**  
Ukuran dan Harga Kapasitor Yang Tersedia<sup>[5]</sup>

Size kVAR	150	300	450	600	900	1200
Cost \$	750	975	1140	1320	1650	2040

**Tabel 4.5**  
Data Kapasitas Kapasitor dan Harga<sup>[5]</sup>

No	Kapasitas (Kvar)	Harga (\$/kVAR)
1	150	0.500
2	300	0.350
3	450	0.253
4	600	0.220
5	750	0.226
6	900	0.183
7	1050	0.228
8	1200	0.170
9	1350	0.207
10	1500	0.201
11	1650	0.193
12	1800	0.187
13	1950	0.211
14	2100	0.176
15	2250	0.197
16	2400	0.170
17	2550	0.189
18	2700	0.187
19	2850	0.183
20	3000	0.180
21	3150	0.163
22	3300	0.174
23	3450	0.188
24	3600	0.170
25	3750	0.183
26	3900	0.182
27	4050	0.179



#### 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 di tiap-tiap bus. Setelah studi aliran daya dilakukan, barulah dilakukan perhitungan aliran daya dan rugi-rugi daya pada saluran.

Untuk memudahkan perhitungan dan analisa pada sistem tenaga, biasanya dipakai harga-harga dalam per-satuan. Harga per-satuan adalah harga yang sebenarnya dibagi dengan harga dasar, dimana harga dasar ini dapat dipilih sembarang. Harga yang dipilih pada studi ini adalah 20 kV dan 100 kVA sebagai harga tegangan dasar dan daya dasar. Mengingat bahwa pada jaringan tidak dilakukan pengukuran faktor daya, maka pada perhitungan ini di asumsikan harga faktor daya sebesar 0,86.

Perhitungan diawali dengan menampilkan single line diagram dari penyulang yang mewakili keadaan sistem yang sesungguhnya. Dari gambar 4-12 diperlihatkan single line diagram dari penyulang Pujon. Pada penyulang ini jumlah bus dan jumlah saluran masing-masing adalah :

- Slack bus = 1
- Load bus = 101
- Jumlah saluran = 100

Dengan mengacu pada gambar single line diagram penyulang Pujon, maka dapat diperoleh hasil impedansi saluran seperti pada table 4 dibawah ini. Setelah

---

itu baru kita lakukan perhitungan aliran daya dengan menggunakan metode *Newton Raphson*.

Setelah dilakukan analisa aliran daya dengan menggunakan metode *Newton Raphson* maka diperoleh besarnya tegangan tiap-tiap bus dan rugi-rugi saluran

#### 4.4.1. Perhitungan Penghematan Biaya (*Saving Cost*)

Nilai penghematan biaya (*saving cost*) merupakan nilai pengurangan biaya sebelum penempatan dan biaya setelah penempatan, yaitu merupakan penghematan biaya yang diperoleh PT. PLN selaku distributor. Nilai total biaya adalah :

$$F = K_p \cdot P_{\text{loss}} + \sum_{j=1}^k K_j^c Q_j^c$$

Dimana  $K_p$  adalah *cost per power loss* (\$/kW/Tahun).

$K^c$  adalah harga kapasitor (\$/kVAR).

$Q^c$  adalah kapasitas kapasitor (kVAR).

$P_{\text{loss}}$  adalah nilai rugi-rugi daya aktif total pada sistem.

Nilai  $K_p$  sudah ditentukan yaitu \$ 168 \$/kW/Tahun<sup>91</sup>, dengan asumsi nilai kurs adalah U.S.\$ 1 – Rp 9.000,00.

➤ nilai biaya sebelum penempatan adalah

$$F = K_p \cdot P_{\text{loss}} + \sum_{j=1}^k K_j^c Q_j^c$$

$$F = 168 \cdot 107.982 + \sum_{j=1}^k K_0^c Q_0^c$$

$$F = 168 \cdot 107.982 + 0 \cdot 0$$

$$= \$ 18.140.97 / \text{th}$$

$$= \text{Rp. } 163.268.820 / \text{th}$$

➤ nilai biaya setelah penempatan metode *Parallel Tabu Search* adalah

$$F = K_p \cdot P_{\text{loss}} + \sum_{j=1}^k K_j^c Q_j^c$$

$$F = 168 \cdot 85.373 + (450 \cdot 0.25300 + 450 \cdot 0.25300 + 1200 \cdot 0.17000)$$

$$= \$ 14.773,834 / \text{th}$$

$$= \text{Rp } 132.957.665 / \text{th}$$

*saving cost* metode *Parallel Tabu Search*

= biaya sebelum penempatan - biaya setelah penempatan

$$= \$ 18.140.97 / \text{th} - \$ 14.773,834 / \text{th}$$

$$= \text{Rp. } 163.268.820 / \text{th} - \text{Rp } 132.957.665 / \text{th}$$

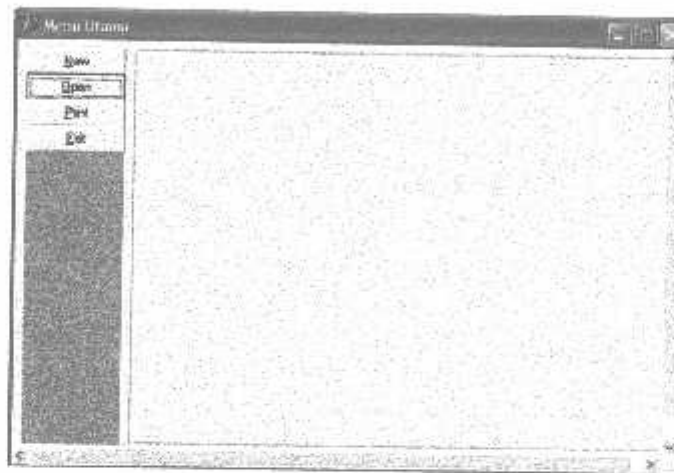
$$= \text{Rp } 30.311.155 / \text{th}$$

## 4.5. Hasil Program

### 4.5.1. Tampilan Program

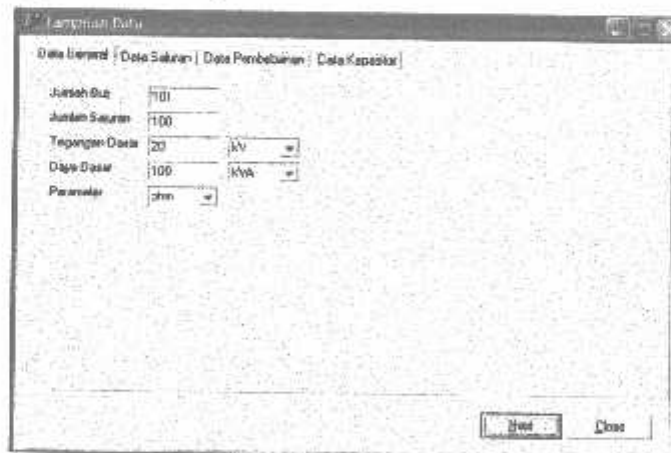
Dalam Skripsi ini menggunakan bantuan bahasa pemrograman Borland Delphi versi 7.0 dan diaplikasikan pada komputer AMD 2200 MHz, dengan memory 256 Mb. Mengenai jalannya program sebagai berikut:

#### 1. Tampilan Utama Program



**Gambar 4.2**  
**Tampilan Utama Program**

2. Tekan tombol New untuk memasukkan data baru atau tekan open untuk data yang sudah tersimpan



**Gambar 4.3**  
**Tampilan Utama Program (Data General)**

No	Dari	Ke	P (pu)	D (pu)	Lc (pu)	Tr
1	2	3	0.155	0.188	0	0
2	3	4	0.368	0.473	0	0
3	4	5	0.158	0.285	0	0
4	5	6	0.065	0.082	0	0
5	6	7	0.147	0.225	0	0
6	7	8	0.453	0.689	0	0
7	8	9	0.034	0.051	0	0
8	9	10	0.026	0.137	0	0
9	10	11	0.130	0.263	0	0
10	11	12	0.182	0.152	0	0
11	12	13	0.057	0.143	0	0
12	13	14	0.308	0.478	0	0
13	14	15	0.288	0.405	0	0
14	15	16	0.320	0.414	0	0
15	16	17	0.096	0.028	0	0

**Gambar 4.4**  
Tampilan Utama Program (Data Saluran)

No	Bus (pu)	SUDV (MVA)	Pmax (MW)	Qmax (MVAR)	Pmin (pu)	Qmin (MVAR)	Cap	Typ-Bus
1	1	0	0	0	0	0	0	1
2	1	0	0	0	44.72	26.53	0	3
3	1	0	0	0	60.2	-20.714	0	3
4	1	0	0	0	0	0	0	3
5	1	0	0	0	36.90	21.923	0	3
6	1	0	0	0	27.23	18.326	0	3
7	1	0	0	0	0	0	0	3
8	5	0	0	0	54.78	-32.143	0	3
9	1	0	0	0	0	0	0	3
10	1	0	0	0	22.12	72.488	0	3
11	1	0	0	0	73.98	47.449	0	3
12	1	0	0	0	0	0	0	3
13	1	0	0	0	69.0	-40.818	0	3
14	1	0	0	0	0	0	0	3
15	1	0	0	0	0	0	0	3

**Gambar 4.5**  
Tampilan Utama Program (Data Pembebanan)

No	Cap (MVAR)	Charge (MVAH/yr)
1	500	0.50000
2	300	0.30000
3	450	0.45000
4	600	0.60000
5	750	0.75000
6	900	0.90000
7	1050	1.05000
8	1200	1.20000
9	1350	1.35000
10	1500	1.50000
11	1650	1.65000
12	1800	1.80000
13	1950	1.95000
14	2100	2.10000
15	2250	2.25000
16	2400	2.40000
17	2550	2.55000
18	2700	2.70000
19	2850	2.85000
20	3000	3.00000
21	3150	3.15000
22	3300	3.30000
23	3450	3.45000
24	3600	3.60000
25	3750	3.75000
26	3900	3.90000
27	4050	4.05000

**Gambar 4.6**  
Tampilan Utama Program (Data Kapasitor)

3. Tekan tombol Next kemudian tekan tombol Hitung NR (Newton Raphson). Untuk melihat hasil perhitungan aliran daya sebelum dilakukan penempatan kapasitor.

No	Aliran Daya	Loss Kapasitor
1	1.00000	0.000
2	0.99760	0.000
3	0.99176	0.000
4	0.98829	0.000
5	0.98829	0.000
6	0.98939	0.000
7	0.97737	0.000
8	0.97732	0.000
9	0.97720	0.000
10	0.96984	0.000
11	0.97697	0.000
12	0.97694	0.000
13	0.97074	0.000
14	0.97684	0.000
15	0.97299	0.000
16	0.97289	0.000
17	0.97289	0.000
18	0.97174	0.000
19	0.97173	0.000
20	0.97171	0.000
21	0.97074	0.000
22	0.97072	0.000
23	0.96571	0.000
24	0.96565	0.000

**Gambar 4.7**  
**Nilai Tegangan Tiap Node Sebelum Penempatan**  
**Menggunakan Metode Aliran Daya Newton Raphson**

No	Das	Ka	P (kW)	Q (kVAR)	Loss (A)
1	2		3928.102	2430.760	0.000
2	2	1	-3921.407	-2420.520	0.000
3	2	3	3976.697	2393.999	0.000
4	3	2	-3960.536	-2393.306	0.000
5	3	4	3990.336	2333.592	0.000
6	4	3	-3790.893	-2319.159	0.000
7	4	5	36.980	21.939	0.000
8	4	5	-3753.913	-2297.220	0.000
9	5	4	-36.990	-21.939	0.000
10	6	4	-3746.531	-2285.026	0.000
11	6	7	3719.071	2268.700	0.000
12	7	6	-3697.070	-2236.067	0.000
13	7	8	325.181	199.010	0.000
14	7	15	3371.889	2043.067	0.000
15	8	7	-325.168	-192.991	0.000
16	8	9	270.989	160.848	0.000
17	9	8	-270.965	-160.812	0.000
18	9	10	270.965	160.812	0.000

**Summary Result**

Total Pembangkitan: 3928.102 + 2430.760 kVA

Total Pembebanan: 3921.170 + 2266.297 kVA

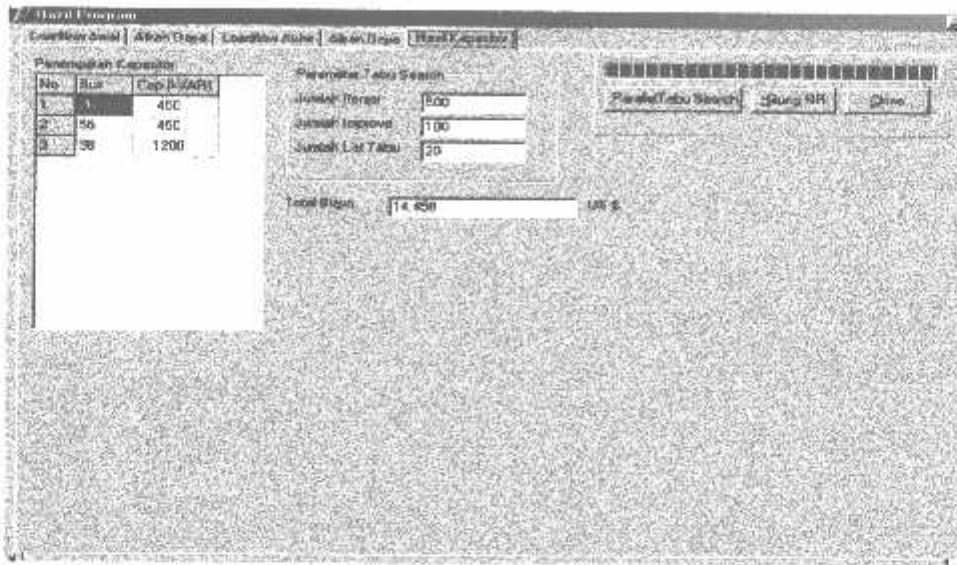
Total Rugi-Rugi Solusian: 107.982 + 1164.463 kVA

Roux: 3

Waktu Komputasi (jurnal: desk:nodek): 0.01210

**Gambar 4.8**  
**Hasil Aliran Daya Sebelum Penempatan**  
**Menggunakan Metode Aliran Daya Newton Raphson**

4. Pilih Tombol *Parallel Tabu Search* untuk penempatan kapasitor



**Gambar 4.9**  
Hasil Metode *Parallel Tabu Search* Dalam Menentukan Lokasi, Kapasitas Kapasitor Serta Total Biaya

5. Pilih tombol loadflow akhir dan aliran daya untuk melihat nilai tegangan dan aliran daya setelah penempatan kapasitor

The screenshot shows the 'Loadflow Akhir' window with a table of results for 34 buses. The table columns are: Bus, Abs V (pu), Sud V (deg), Pgen (MW), Qgen (MVA), Pload (MW), and Qload (MVA).

Bus	Abs V (pu)	Sud V (deg)	Pgen (MW)	Qgen (MVA)	Pload (MW)	Qload (MVA)
1	1.0000	0.0000	2903.610	139.637	0.000	0.000
2	0.99971	-0.10487	0.000	0.000	44.720	26.530
3	0.99961	-0.36305	0.000	0.000	60.200	35.714
4	0.99380	-0.51797	0.000	0.000	0.000	0.000
5	0.99380	-0.51806	0.000	0.000	36.980	21.939
6	0.99240	-0.64014	0.000	0.000	27.520	16.326
7	0.98823	-1.01303	0.000	0.000	0.000	0.000
8	0.98819	-1.01525	0.000	0.000	54.180	32.143
9	0.98813	-1.02065	0.000	0.000	0.000	0.000
10	0.98799	-1.03203	0.000	0.000	122.120	72.448
11	0.98797	-1.03628	0.000	150.000	79.880	47.448
12	0.98794	-1.03721	0.000	0.000	0.000	0.000
13	0.98794	-1.04071	0.000	0.000	68.800	40.818
14	0.98794	-1.03721	0.000	0.000	0.000	0.000
15	0.98601	-1.21936	0.000	0.000	0.000	0.000
16	0.98600	-1.21976	0.000	0.000	32.680	19.388
17	0.98600	-1.21983	0.000	0.000	119.680	70.408
18	0.98547	-1.27401	0.000	0.000	0.000	0.000
19	0.98545	-1.27431	0.000	0.000	62.780	37.245
20	0.98544	-1.27472	0.000	0.000	67.080	38.796
21	0.98502	-1.32210	0.000	0.000	0.000	0.000
22	0.98489	-1.32289	0.000	0.000	77.400	45.918
23	0.98329	-1.51830	0.000	0.000	0.000	0.000
24	0.98322	-1.52005	0.000	0.000	143.420	85.289
25	0.98225	-1.65944	0.000	0.000	72.580	42.887

**Gambar 4.10**  
Hasil Analisa Aliran Daya Newton Raphson Setelah Penempatan Kapasitor Nilai Tegangan Tiap Node

Hasil Program						
Loadflow Awal	Aliran Daya	Loadflow Akhir	Aliran Daya	Hasil Kapasitor		
No	Dari	Ke	P (kW)	Q (kVAR)	Angkanya (A)	Angkanya (A)
1		2	3903.610	138.537	0.000	
2	2	1	-3998.822	-132.221	0.000	
3	2	3	3954.102	106.691	0.000	
4	3	2	-3842.563	-88.049	0.000	
5	3	4	3782.363	52.335	0.000	
6	4	3	-3775.622	-42.031	0.000	
7	4	5	36.980	21.939	0.000	
8	4	6	3738.642	20.092	0.000	
9	5	4	-36.980	-21.939	0.000	
10	6	4	3733.416	-12.103	0.000	
11	6	7	3205.896	-4.223	0.000	
12	7	6	-3630.197	28.222	0.000	
13	7	8	325.156	42.972	0.000	
14	7	15	3365.042	-71.194	0.000	
15	8	7	-325.147	-42.958	0.000	
16	8	9	270.967	10.815	0.000	
17	9	8	-270.950	-10.769	0.000	
18	9	10	270.950	10.789	0.000	

Summary Result		
Total Pembangkitan	3903.610 + j2389.537	kVA
Total Pembebanan	3820.120 + j2266.297	kVA
Total Rugi-Rugi Saluran	83.490 + j123.240	kVA
Jenis	3	
Waktu Komputasi (jam:menit:detik)	0:24:35:350	

**Gambar 4.11**  
**Hasil Analisa Aliran Daya Newton Rahson**  
**Setelah Penempatan Kapasitor metode Parallel Tabu Search**  
**Nilai Daya Dan Rugi-Rugi Pada Saluran**



Tabel 4.6  
 Nilai Tegangan sebelum dan sesudah pemasangan kapasitor

Nilai Tegangan Sebelum Penempatan				Nilai Tegangan Setelah Penempatan			
Bus	Abs V (pu)	Bus	Abs V (pu)	Bus	Abs V (pu)	Bus	Abs V (pu)
1	1.00000	51	0.95603	1	1.00000	51	0.97897
2	0.99750	52	0.95602	2	0.99871	52	0.97896
3	0.99176	53	0.95572	3	0.99561	53	0.97903
4	0.98829	54	0.95509	4	0.99380	54	0.97911
5	0.98829	55	0.95470	5	0.99380	55	0.97923
6	0.98558	56	0.95457	6	0.99240	56	0.97909
7	0.97737	57	0.95450	7	0.98823	57	0.97903
8	0.97732	58	0.95443	8	0.98819	58	0.97902
9	0.97720	59	0.95454	9	0.98813	59	0.97907
10	0.97694	60	0.95453	10	0.98799	60	0.97905
11	0.97687	61	0.95450	11	0.98797	61	0.97903
12	0.97684	62	0.95449	12	0.98794	62	0.97902
13	0.97674	63	0.95678	13	0.98784	63	0.97910
14	0.97684	64	0.95669	14	0.98794	64	0.97902
15	0.97290	65	0.95658	15	0.98601	65	0.97891
16	0.97289	66	0.95655	16	0.98600	66	0.97888
17	0.97289	67	0.95658	17	0.98600	67	0.97904
18	0.97174	68	0.95656	18	0.98547	68	0.97903
19	0.97173	69	0.95641	19	0.98546	69	0.97899
20	0.97171	70	0.95517	20	0.98544	70	0.97868
21	0.97074	71	0.95436	21	0.98502	71	0.97849
22	0.97072	72	0.95432	22	0.98499	72	0.97848
23	0.96671	73	0.95431	23	0.98328	73	0.97847
24	0.96665	74	0.95401	24	0.98322	74	0.97844
25	0.96662	75	0.95378	25	0.98325	75	0.97842
26	0.96624	76	0.95323	26	0.98315	76	0.97788
27	0.96624	77	0.95321	27	0.98315	77	0.97785
28	0.96618	78	0.95309	28	0.98309	78	0.97774
29	0.96582	79	0.95301	29	0.98274	79	0.97766
30	0.96432	80	0.95291	30	0.98213	80	0.97757
31	0.96394	81	0.95280	31	0.98195	81	0.97746
32	0.96391	82	0.95269	32	0.98192	82	0.97736
33	0.96383	83	0.95354	33	0.98185	83	0.97850
34	0.96364	84	0.95353	34	0.98166	84	0.97850
35	0.96363	85	0.95347	35	0.98165	85	0.97853
36	0.96363	86	0.95344	36	0.98165	86	0.97851
37	0.96363	87	0.95334	37	0.98165	87	0.97861
38	0.96363	88	0.95304	38	0.98165	88	0.97892
39	0.96156	89	0.95266	39	0.98093	89	0.97941
40	0.96156	90	0.95263	40	0.98093	90	0.97939
41	0.96151	91	0.95260	41	0.98088	91	0.97936
42	0.96151	92	0.95258	42	0.98088	92	0.97955
43	0.96147	93	0.95222	43	0.98090	93	0.98027
44	0.95761	94	0.95218	44	0.97941	94	0.98023
45	0.95708	95	0.95196	45	0.97920	95	0.98044
46	0.95688	96	0.95195	46	0.97913	96	0.98043
47	0.95646	97	0.95175	47	0.97902	97	0.98061
48	0.95619	98	0.95127	48	0.97897	98	0.98145
49	0.95605	99	0.95126	49	0.97897	99	0.98144
50	0.95601	100	0.95111	50	0.97893	100	0.98129
		101	0.95105			101	0.98123

#### 4.5.2. Hasil Penempatan Kapasitor Menggunakan Metode *Parallel Tabu Search*.

Setelah ditentukan penempatan kapasitor pada jaringan 20 kV Sistem Distribusi Radial Penyulang Pujon Gardu Induk Sengkaling menggunakan metode *Parallel Tabu Search*, maka didapatkan beberapa hasil pada tabel 4.7:

**Tabel 4.7**  
**Hasil Penempatan Kapasitor**  
**Metode *Parallel Tabu Search***

	Sebelum Penempatan	Setelah Penempatan Metode <i>Parallel Tabu Search</i>
1. Penempatan dan Kapasitas kapasitor	-	Node : 26 kapasitas 450 55 kapasitas 450 98 kapasitas 1200
2. Tegangan terendah	Node 101 0.95105 pu atau 19.021 kV	Node : 101 0.98123 pu atau 19.6246 kV
3. Rugi Daya Aktif	107.982 kW	83.490 kW pengurangan : 24.492 kW
Rugi Daya Reaktif	164.463Kvar	123.240 Kvar pengurangan : 41.223 kVAR
4. Total biaya	\$ 18.140/ year	\$ 14.458/ year
Nilai Penghematan	-	\$ 3.682 / year

Hasil penempatan kapasitor metode Parallel Tabu Search dapat diuraikan sebagai berikut

1. Berdasarkan hasil perhitungan dengan metode Parallel Tabu Search

Penempatan teletak pada node 26, 55 dan 98, pada node 26 kapasitas yang terpasang sebesar 450 kVAR, pada node 55 kapasitas yang terpasang sebesar 450 kVAR dan node 98 kapasitas yang terpasang sebesar 1200 kVAR.

Hasil penempatan kapasitor dapat dilihat pada *single line diagram* gambar 4.13

2. Dari hasil perhitungan aliran daya pada penyulang Pujon sebelum kompensasi, tegangan terendah terjadi pada node 101, yang besarnya 0.95105 pu atau 19.021 kV. Setelah kompensasi profil tegangan pada semua node penyulang Pujon bertambah baik.

- Nilai tegangan pada node ke-101 terjadi kenaikan sebesar 0.98123 pu atau 19.6246 kV

Profil tegangan tiap node sebelum dan sesudah penempatan kapasitor dapat dilihat pada grafik 4.1

3. Dari analisa aliran daya sebelum penempatan kapasitor didapat hasil rugi daya aktif di saluran 107.982 kW dan rugi daya reaktif 164.463kVAR. Setelah penempatan kapasitor besar rugi daya pada sistem terjadi penurunan, yaitu :

- Rugi daya aktif sebesar 83.490 kW,  
terjadi pengurangan rugi-rugi sebesar: 24.492 kW

- Rugi daya reaktif sebesar 123.240 kVAR. terjadi pengurangan rugi daya reaktif sebesar: 41.223 kVAR

Nilai rugi-rugi daya dan sesudah penempatan dapat dilihat pada grafik.4.2.

4. Nilai biaya sebelum penempatan sebesar \$ 18.140/ year. setelah penempatan kapasitor, nilai penghematan yang dicapai oleh PT. PLN UP-J Batu selaku distributor listrik selama setahun sebesar \$ 3.682 / year dimana total biaya setelah penempatan sebesar \$ 14.458/year

Nilai penghematan sebelum dan setelah penempatan kapasitor dapat dilihat pada grafik 4.3.

$$Z_{\text{dasar}} = \frac{(\text{tegdasarKV}_m)^2}{(\text{dsrmVA})^2} = \frac{20^2}{0,1^2} = 4000\Omega$$

$$Z_{\text{pm}} = \frac{Z_{\Omega}}{4000}$$

aliran daya dari bus 1 ke bus 2 sebelum penempatan

$$P_{pq} + jQ_{pq} = V_p i_{pq}^*$$

$$= V_p [(V_p - V_q) Y_{pq}]^*$$

$$P_{12} + jQ_{12} = V_1 i_{12}^*$$

$$= V_1 [(V_1 - V_2) Y_{12}]^*$$

$$= 1 < 0^\circ [(1 < 0^\circ - 0,99760 < -0,06438)]$$

$$\left( \frac{1}{3,1375 \times 10^{-5} + j4,795 \times 10^{-5}} \right) \times 100$$

$$= 3928,102 + j2430,760$$

aliran daya dari bus 2 ke bus 1 sebelum penempatan

$$P_{qp} + jQ_{qp} = V_q i_{qp}^*$$

$$= V_q [(V_q - V_p) Y_{qp}]^*$$

$$P_{21} + jQ_{21} = V_2 i_{21}^*$$

$$= V_2 [(V_2 - V_1) Y_{21}]^*$$

$$= 0,99760 < -0,06438^\circ [(0,99760 < -0,06438^\circ - 1 < 0^\circ)]$$

$$\left( \frac{1}{3,1375 \times 10^{-5} + j4,795 \times 10^{-5}} \right) \times 100$$

$$= -3921,407 + j - 2420,528$$

Rugi daya pada saluran dari bus 1 ke 2

$$= (P_{12} + jQ_{12}) + (P_{21} + jQ_{21})$$

$$= (3928,102 + j2430,760) + (-3921,407 + j - 2420,528)$$

$$= 6,695 + j10,232$$

dengan perhitungan yang sama dengan menggunakan komputer

aliran daya dari bus 1 ke bus 2 setelah penempatan

$$\begin{aligned}
 P_{pq} + jQ_{pq} &= V_p i_{pq}^* \\
 &= V_p [(V_p - V_q) Y_{pq}]^* \\
 P_{12} + jQ_{12} &= V_1 i_{12}^* \\
 &= V_1 [(V_1 - V_2) Y_{12}]^* \\
 &= 1 < 0^\circ [(1 < 0^\circ - 0,99871 < -0,10487^\circ) \\
 &\quad \left( \frac{1}{3,1375 \times 10^{-5} + j4,795 \times 10^{-5}} \right)] \times 100 \\
 &= 3903,610 + j139,537
 \end{aligned}$$

aliran daya dari bus 2 ke bus 1 sesudah penempatan

$$\begin{aligned}
 P_{qp} + jQ_{qp} &= V_q i_{qp}^* \\
 &= V_q [(V_q - V_p) Y_{qp}]^* \\
 P_{21} + jQ_{21} &= V_2 i_{21}^* \\
 &= V_2 [(V_2 - V_1) Y_{21}]^* \\
 &= 0,99871 < -0,10487^\circ [(0,99871 < -0,10487^\circ - 1 < 0^\circ) \\
 &\quad \left( \frac{1}{3,1375 \times 10^{-5} + j4,795 \times 10^{-5}} \right)] \times 100 \\
 &= -3898,822 + j-132,221
 \end{aligned}$$

Rugi daya pada saluran dari bus 1 ke 2

$$\begin{aligned}
 &= (P_{12} + jQ_{12}) + (P_{21} + jQ_{21}) \\
 &= (3902,610 + j139,537) + (-3898,822 + j-132,221) \\
 &= 4,788 + j7,316
 \end{aligned}$$

dengan perhitungan yang sama dengan menggunakan komputer

Dari Bus	Ke bus	Rugi daya Sebelum optimasi		Dari Bus	Ke bus	Rugi daya Sesudah optimasi	
		$P_{\text{loss}}$ (kW)	$Q_{\text{loss}}$ (kVar)			$P_{\text{loss}}$ (kW)	$Q_{\text{loss}}$ (kVar)
1	2	6.695	10.252	1	2	4.788	7.316
2	3	16.151	24.697	2	3	11.539	17.647
3	4	9.673	14.413	3	4	6.741	10.304
4	5	0.000	0.000	4	5	0.000	0.000
4	6	22.001	33.633	4	6	5.226	7.989

### Cos $\phi$ sebelum penempatan

$$2430,760 / 3928,102 = 0,6188$$

$$\text{Tan } \phi = 0,6188$$

$$\phi = 31,749$$

$$\text{Cos } \phi = 0,86$$

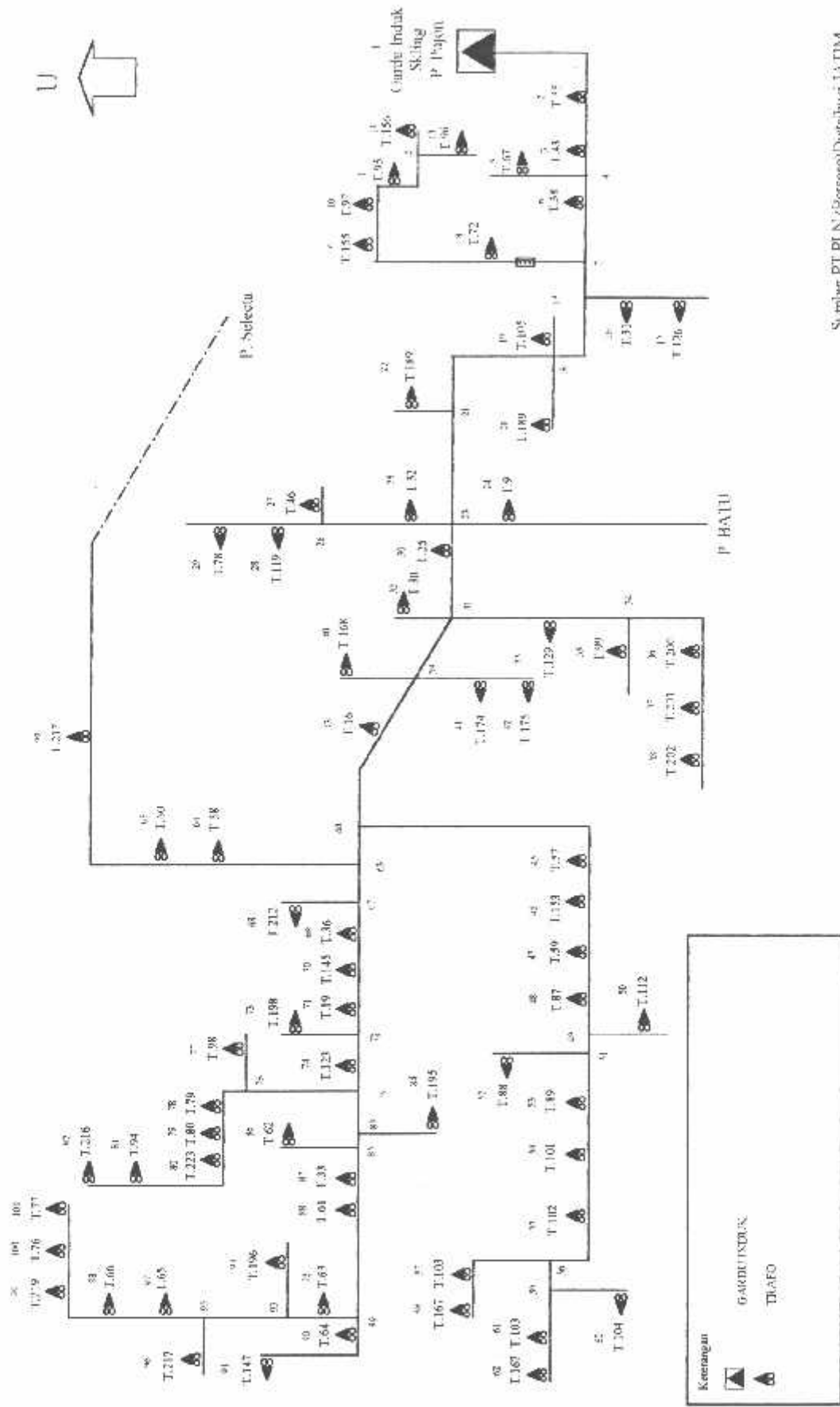
### Cos $\phi$ setelah penempatan

$$139,573 / 3903,610 = 0,035$$

$$\text{Tan } \phi = 0,035$$

$$\phi = 2,004$$

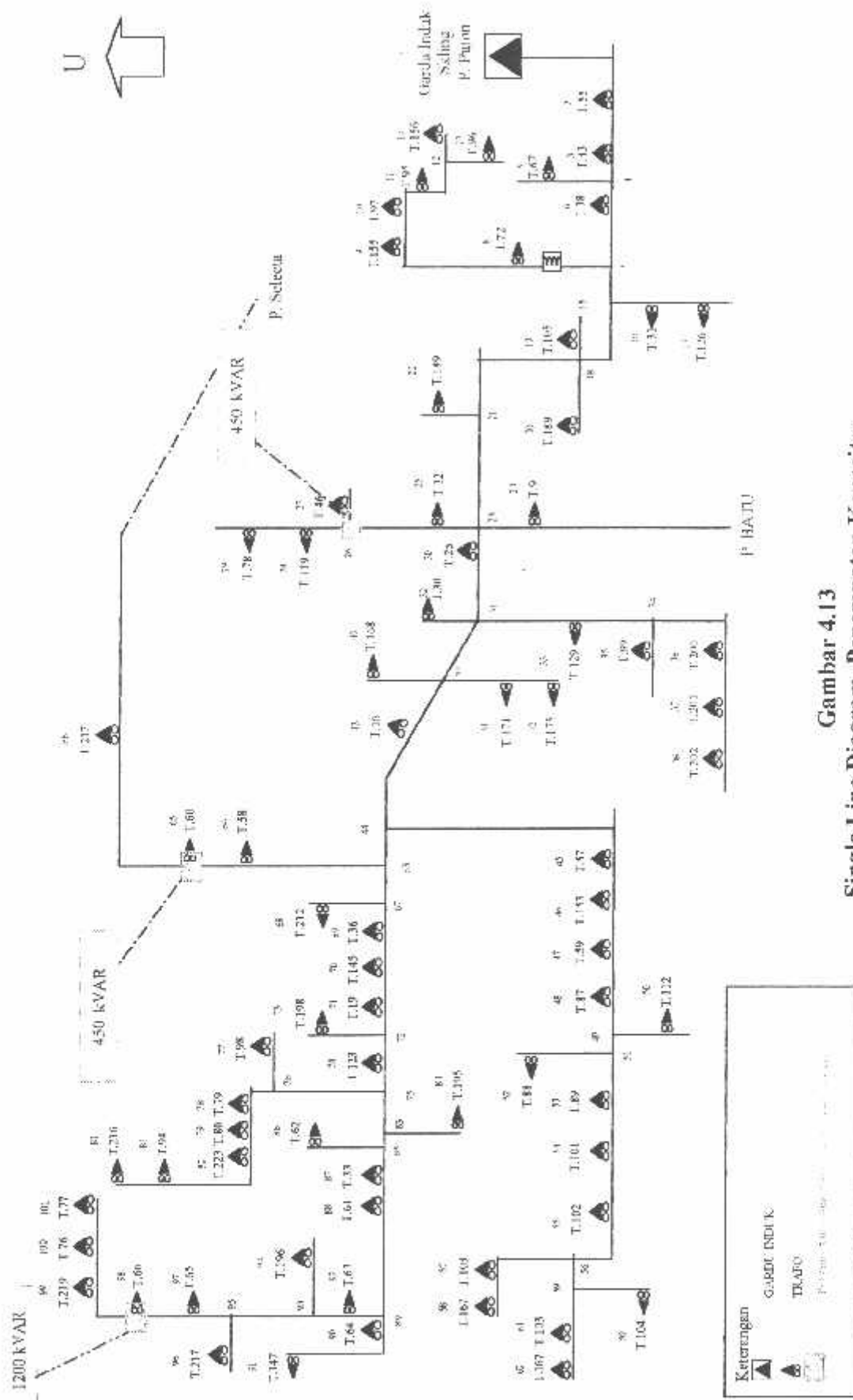
$$\text{Cos } \phi = 0,99$$



Gambar 4.12  
Single Line Diagram Penyulang Pujon Sebelum  
Penempatan Kapasitor

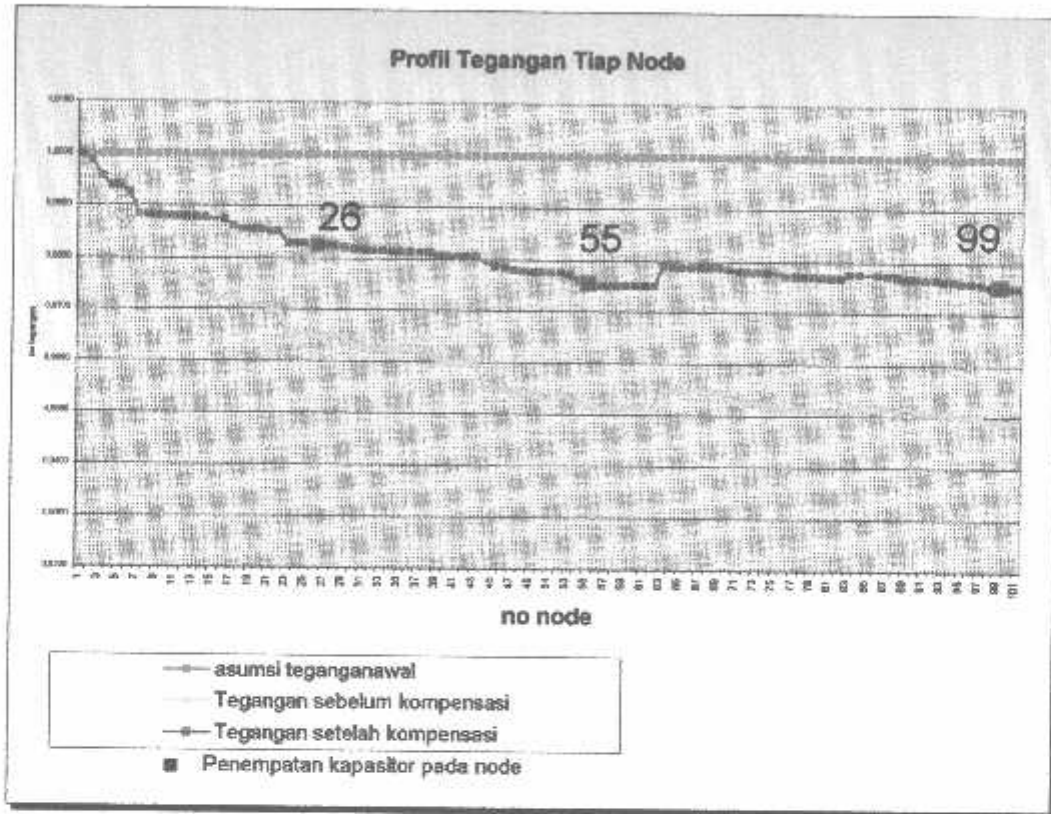
Sumber: PT. PLN (Persero) Distribusi J.A DIM  
Area Malang, TP - J Batu



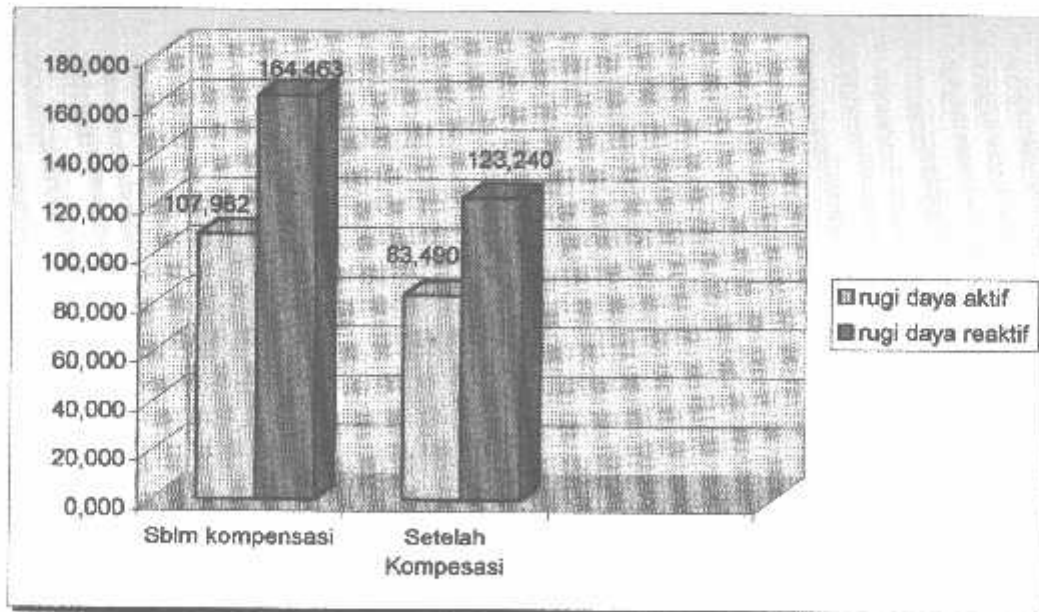


Gambar 4.13  
Single Line Diagram Penempatan Kapasitor  
Metode Paralel Tabu Search

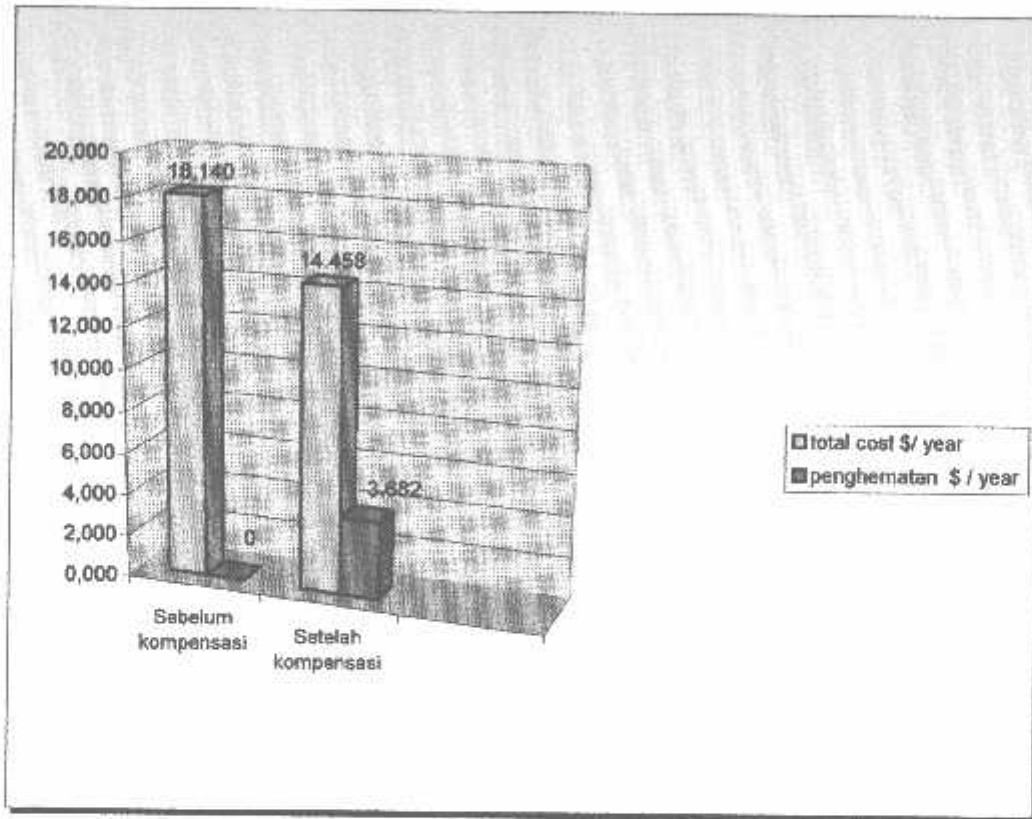
**Grafik 4.1.**  
**Profil tegangan sebelum dan sesudah penempatan kapasitor**



**Grafik 4.2.**  
**Rugi-rugi Daya sebelum dan sesudah penempatan kapasitor**



**Grafik 4.3.**  
**Nilai Biaya Sebelum Dan Setelah Pemasangan Kapasitor**  
*Metode Parallel Tabu Search*



## BAB V KESIMPULAN

### 5.1 Kesimpulan

Setelah dilakukan analisa mengenai penentuan letak dan kapasitas kapasitor pada jaringan distribusi radial dengan menggunakan metode *Parallel Tabu Search*, maka dapat diambil kesimpulan :

1. Berdasarkan hasil perhitungan dengan metode *Parallel Tabu Search*, letak kapasitor yang optimal terletak pada node 26, 55 dan 98. serta kapasitas kapasitor masing-masing adalah 450,450 dan 1200 kVAR
2. Dari hasil perhitungan aliran daya pada penyulang Pujon sebelum dikompensasi, tegangan paling rendah terjadi pada node ke-101, yang besarnya 0,95105 atau 19.021 kV. Setelah kompensasi profil tegangan pada semua node penyulang Pujon bertambah baik, untuk node ke-101 menjadi 0,98123 atau 19.6246 kV mengalami kenaikan sebesar 3,075 %.
3. Besar rugi daya mengalami penurunan yaitu untuk daya aktif sebesar 24.492 kW dari 107.982 kW menjadi 83.490 kW. Sedangkan untuk daya reaktif sebesar 41.223 kVAR dari 164.463Kvar menjadi 123.240 kVAR.
4. Dari hasil perhitungan program maka dapat diketahui besar total biaya dari instalasi kapasitor sebelum penempatan sebesar \$18.140/year. Sedangkan total biaya setelah penempatan sebesar \$ 14.458/ year tersebut, jadi dalam setahun PLN selaku pihak distribusi tenaga listrik mengalami keuntungan sebesar \$ 3.682 / year. Atau Rp 33.138.000/ th dengan asumsi nilai dolar adalah \$1 = Rp 9000,00.

## DAFTAR PUSTAKA

- [1]. Djiteng Maesudi, 1990, "Operasi sistem tenaga Listrik", Balai Penerbit Dan Humas ISTN
- [2]. Turan Gonen, **"Electric Power Distribution System Engineering"**, University of Missouri at Colombia
- [3]. Hasan Basri, **"Sistem Distribusi Tenaga Listrik"**.
- [4]. William D Stevenson, Jr. **"Analisa Sistem Tenaga Listrik"**. Edisi keempat, Erlangga, Jakarta, 1993.
- [5]. Y.baghzouz, **Shunt Capacitor Sizing For Radial Distribusi Feeders With Distorted Substation Voltages**, vol.5, no 2, April 1990.
- [6]. Hiroyuki Mori, Yoshihiro Ogita, **"Capasitor Placement Using Parallel Tabu Search in Distribution System"**, Meiji University. 1999
- [7]. Sulasno, **"Analisa Sistem Tenaga Listrik"**, Cetakan Kesatu, Satya Wacana, Semarang, 1993.
- [8]. Hiroyuki Mori, Takanori Hayashi, **"New Parallel Tabu Search For Voltage And Reactive Power Control in Power Systems"**, Meiji University. 1998
- [9]. S.F.Mekhamer, S.A Soliman, M.A. Moustafa, M.E. El-Hawary, **Application of fuzzy logic for reactive-power compensation of radial distribution feeders**, Vol.18 No.1,February 2003

# LAMPPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
JURUSAN TEKNIK ELEKTRO

**BERITA ACARA UJIAN SKRIPSI  
FAKULTAS TEKNOLOGI INDUSTRI**

1. Nama : Faried Sujianto
2. NIM : 9812010
3. Jurusan : Teknik Elektro
4. Konsentrasi : Energi Listrik (S-1)
5. Judul Skripsi : ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN  
DISTRIBUSI RADIAL DENGAN METODE PARALLEL TABU  
SEARCH DI G.I SENGKALING PENYULANG PUJON

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

6. Hari : Selasa
7. Tanggal : 4 Oktober 2005
8. Dengan Nilai : 75,75 (B+) *Sm*



Ir. Mochtar Astroni, MSME  
NIP. Y. 1018100036

Panitia Ujian Skripsi



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

Anggota Penguji

Dosen penguji I

Ir. M. Abd. Hamid, MT  
NIP. Y. 1010088188

Dosen penguji II

Ir. Djojo Phiatmono, MT  
NIP. Y. 1018500107



### PESETUJUAN PERBAIKAN SKRIPSI

Dari hasil ujian skripsi Jurusan Teknik Elektro Jenjang Strata Satu (S-1) yang diselenggarakan pada:

Hari : Selasa

Tanggal : 4 Oktober 2005

Telah dilakukan perbaikan skripsi oleh:

1. Nama : Faried Sujianto
2. NIM : 9812010
3. Jurusan : Teknik Elektro
4. Konsentrasi : Energi Listrik (S-1)
5. Judul Skripsi : ANALISIS PENEMPATAN KAPASITOR PADA JARINGAN DISTRIBUSI RADIAL DENGAN METODE PARALLEL TABU SEARCH DI G.I SENGKALING PENYULANG PUJON

Perbaikan meliputi

No	Materi Perbaikan	Keterangan
1	Chek hal 54 poin 1	
2	Chek hal 43 US \$/kVAR/thn	
3	Perlu di buat contoh perhitungan untuk rugi-rugi daya dll	
4	Hitung Cos $\phi$ baru penempatan kapasitor	

Disetujui dan Diperiksa

Dosen penguji I

Ir. M. Abd. Hamid, MT  
NIP. Y.1010088188

Dosen penguji II

  
Ir. Djojo Priatmono, MT  
NIP. Y.1018500107

Mengetahui / Menyetujui  
Dosen Pembimbing

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





FORMULIR BIMBINGAN SKRIPSI

Nama : Faried Sujianto  
Nim : 98.12.010  
Masa Bimbingan : 03 Januari 2005 s/d 03 Juni 2005. *1/*  
Judul Skripsi : Analisis Penempatan Kapasitor Pada Jaringan Distribusi Radial Dengan Metode Parallel Tabu Search Di G.I Sengkaling Penyulang Pujon.

No	Tanggal	Uraian	Paraf Pembimbing
1.	10-03-05	Tujuan dan kesimpulan disimpulkan Saran : Sebaiknya yang sesuai	<i>HS</i>
2.	14-03-05	Tambahan teori rupa tegangan perubahan masalah di nilai dari angka yang berbeda	<i>HS</i>
3.	17-03-05	Software yg digunakan apa? koding program & validasinya?	<i>HS</i>
4.	17-03-05	Terbukan nilai G.I. konvergensi pada flowchart Parallel Tabu Search dan flowchart penyelesaian masalah	<i>HS</i>
5.	21-03-05	menentukan rupa 2 tegangan, rupa dengan sebelum dan sesudah	<i>HS</i>
6.			
7.			
8.			
9.			
10.			

Malang, 200  
Dosen Pembimbing,

*Taufik Hidayat*  
(Ir. Taufik Hidayat, MT)

Form.S-4b

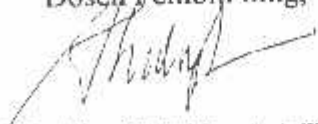


FORMULIR BIMBINGAN SKRIPSI

Nama : Faried Sujianto  
Nim : 98.12.010  
Masa Bimbingan : 03 Juni 2005 s/d 03 November 2005  
Judul Skripsi : Analisis Penempatan Kapasitor Pada Jaringan Distribusi Radial Dengan Metode Paralel Tabu Search Di G.1 Sengkaling Penyulang Pujon.

No	Tanggal	Uraian	Paraf Pembimbing
1.	1/07/05	Dangutan ke 66 v	Al
2.	05/09/05	Suaran (kul 59) diinlaykan	Al
3.	18/09/05	Sinkronkan antara busbar dengan busbar metal	Al
4.	26/09/05	Ace mengahut rayon	Al
5.			
6.			
7.			
8.			
9.			
10.			

Malang, 26/05 2005  
Dosen Pembimbing,

  
(Ir. Taufik Hidayat, MT)



PERKUMPULAN PENGELOLA PENDIDIKAN UML M DAN TEKNOLOGI NASIONAL MALANG  
**INSTITUT TEKNOLOGI NASIONAL MALANG**

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

BNI (PERSERO) MALANG  
BANK NIAGA MALANG

Kampus I : Jl. Bandungan Sigura-gura No. 2 Telp. (0341) 561431 (Hunting) Fax. (0341) 553015 Malang 65145  
Kampus II : Jl. Raya Karanglo, Km 2 Telp. (0341) 417636 Fax. (0341) 417634 Malang

Malang, 13 Januari. 2005

Nomor : ITN-021/I.SKP /2/05  
Lampiran : satu lembar  
Perihal : **BIMBINGAN SKRIPSI**

Kepada : Yth. Sdr. **Ir. H. TAUFIK HIDAYAT, MT**  
Dosen Institut Teknologi Nasional  
di-  
Malang

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

Nama : FARIED SUJANTO  
Nim : 9812010  
Fakultas : Teknologi Industri  
Jurusan : Teknik Elektro  
Konsentrasi : T. Energi Listrik (S-1)

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

3 Januari 2005 s/d 3 Juni 2005

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



Ketua  
Jurusan Teknik Elektro

*H. I Made Wartana, MT*  
H. I Made Wartana, MT  
Nip. 131 991 182

**Tindakan :**

1. Mahasiswa yang bersangkutan
2. Arsip.

Form. S-4a

### DATA SALURAN PENYULANG PUJON

No	Dari	ke	Panjang	Impedensi Saluran	
	saluran	Node		Node	R
1	1	2	580.3	0.1255	0.1918
2	2	3	1432.8	0.3097	0.4735
3	3	4	363.9	0.1868	0.2855
4	4	5	85.4	0.0185	0.0282
5	4	6	583.3	0.1477	0.2258
6	6	7	2082.8	0.4503	0.6884
7	7	8	154.4	0.0334	0.051
8	8	9	414.5	0.0896	0.137
9	9	10	898.5	0.1938	0.2963
10	10	11	463.5	0.1002	0.1532
11	11	12	452	0.0977	0.1494
12	12	13	1424.5	0.308	0.4708
13	12	14	1233	0.2866	0.4076
14	7	15	1244.8	0.2891	0.4114
15	15	16	90.5	0.0196	0.0299
16	16	17	20.4	0.0044	0.0067
17	15	18	336	0.0731	0.1117
18	18	19	182.2	0.049	0.0615
19	18	20	402.8	0.1083	0.136
20	18	21	304.3	0.0958	0.1008
21	21	22	283.7	0.078	0.0982
22	21	23	1257.4	0.2719	0.4156
23	23	24	406.5	0.0879	0.1344
24	23	25	154.2	0.0665	0.0567
25	23	26	627.4	0.5783	0.2378
26	26	27	8.4	0.0059	0.0024
27	26	28	204.2	0.1882	0.0774
28	28	29	1145	1.0562	0.4343
29	23	30	868.2	0.1877	0.2889
30	30	31	141.2	0.0306	0.0467
31	31	32	848.8	0.1835	0.2805
32	31	33	272.5	0.1758	0.1002
33	33	34	1127.05	0.7272	0.4145
34	34	35	102.2	0.0659	0.0376
35	34	36	496.35	0.1073	0.164
36	36	37	282.5	0.0811	0.0934
37	37	38	294.5	0.0837	0.0973
38	31	39	843.3	0.039	0.3118
39	39	40	302.31	0.1951	0.1112
40	39	41	848.11	0.1185	0.1812
41	41	42	105.9	0.0229	0.035
42	39	43	40.5	0.0088	0.0134
43	43	44	1653.6	0.3575	0.5485
44	44	45	785.2	0.1898	0.2585
45	45	46	335	0.0728	0.111
46	46	47	783.2	0.1715	0.2622
47	47	48	571	0.1236	0.1887
48	48	49	368.5	0.079	0.1208
49	49	50	729.8	0.1578	0.2412
50	49	51	54.1	0.0117	0.0179

No	Dari	ke	Panjang	Impedensi Saluran	
	saluran	Node		Node	R
51	51	52	121.8	0.0283	0.0403
52	51	53	940.35	0.2033	0.3108
53	53	54	1715.3	0.7904	0.6127
54	54	55	1182.7	0.5486	0.426
55	55	56	1173.5	0.2429	0.3713
56	56	57	1363.3	0.2628	0.4473
57	57	58	1067.1	0.2307	0.3527
58	56	59	396.5	0.0657	0.131
59	59	60	286.8	0.0842	0.0981
60	59	61	1063.4	0.2364	0.3614
61	61	62	1182.8	0.2514	0.3842
62	44	63	497.1	0.1075	0.1643
63	63	64	570.4	0.1233	0.1895
64	64	65	907	0.1961	0.2968
65	65	66	1789.8	0.3826	0.548
66	63	67	136.2	0.0282	0.0447
67	67	68	821.6	0.1776	0.2715
68	67	69	110.15	0.0238	0.0364
69	69	70	881.3	0.827	0.2948
70	70	71	803.2	0.1304	0.1964
71	71	72	27.3	0.0059	0.0009
72	72	73	233.1	0.0504	0.077
73	72	74	258.7	0.0559	0.0855
74	74	75	198.1	0.0428	0.0655
75	75	76	1538.1	0.2325	0.5063
76	78	77	381.43	0.0988	0.1261
77	76	78	472	0.1021	0.158
78	78	79	449.6	0.0972	0.1408
79	79	80	721.4	0.156	0.2384
80	80	81	940.5	0.2033	0.3109
81	61	82	1428.3	0.3088	0.4721
82	75	83	322.1	0.0866	0.1063
83	83	84	186.4	0.0403	0.0618
84	83	85	98.5	0.0213	0.0328
85	85	86	243.9	0.0527	0.0806
86	65	87	205.3	0.0444	0.0679
87	87	88	590.3	0.1276	0.1951
88	88	89	856.8	0.1852	0.2831
89	89	90	458.3	0.0991	0.1515
90	90	91	1162.2	0.2513	0.3841
91	89	92	207.4	0.0448	0.0689
92	92	93	1061.5	0.225	0.3508
93	93	94	1321.52	0.8527	0.4861
94	93	95	385.1	0.2485	0.1416
95	95	96	478.3	0.3086	0.1759
96	95	97	332.5	0.2146	0.1273
97	97	98	1161.5	0.7484	0.4272
98	98	99	100.4	0.0217	0.0332
99	9	100	788.8	0.507	0.268
100	100	101	285.3	0.1841	0.1049

Sumber PT.PLN (Persero) Distribusi JATIM  
Area Malang UP - J Retu

## DATA PEMBEBANAN SISTEM 20 kV PENYULANG PUJON

No	Kode	Lokasi	Pembebanan			type
			S (kVA)	P (kW)	Q (kVAR)	
1			0,0000	0,0000	0,0000	Slack
2	T55	Jl Raya Ngandut	52	44,7200	26,5304	Load
3	T43	Jl Raya Mojorejo	70	60,2000	35,7140	Load
4			0	0,0000	0,0000	Load
5	T63	Dk Mantung	43	36,9800	21,9388	Load
6	T38	Jl Raya Beji	32	27,5200	16,3264	Load
7			0	0,0000	0,0000	Load
8	T72	emanuel temas	63	54,1800	32,1426	Load
9	T115	Jl.Wukir Temas	0	0,0000	0,0000	Load
10	T97	Ds Temas Klerek	142	122,1200	72,4484	Load
11	T95	Ds Torong Rejo	93	79,9800	47,4488	Load
12			0	0,0000	0,0000	Load
13	T96	Ds Torong Rejo tutup	80	68,8000	40,8160	Load
14	T156	Ds wukir	0	0,0000	0,0000	Load
15			0	0,0000	0,0000	Load
16	T31	Jl Sudiro	38	32,6800	19,3876	Load
17	T126	Jl Sudiro	138	118,6800	70,4076	Load
18			0	0,0000	0,0000	Load
19	T105	R.s Panu-paru	73	62,7800	37,2446	Load
20	T189	Jl Kasiman	78	67,0800	39,7958	Load
21			0	0,0000	0,0000	Load
22	T188	Jl Lesti	90	77,4000	45,9180	Load
23			0	0,0000	0,0000	Load
24	T9	Jl Hasanudin	167	143,6200	85,2034	Load
25	T32	Jl Indragiri	84	72,2400	42,8568	Load
26			0	0,0000	0,0000	Load
27	T46	Ds. Sumberejo	113	97,1800	57,6526	Load
28	119	Ds. Sumberejo	0	0,0000	0,0000	Load
29	78	Ds Satrehan	122	104,9200	62,2444	Load
30	25	Jl Trunojoyo	26	22,3600	13,2652	Load
31			0	0,0000	0,0000	Load
32	30	Ds Songgoriti	39	33,5400	19,8978	Load
33	T129	Jl Flamboyan	110	94,6000	56,1220	Load
34			0	0,0000	0,0000	Load
35	99	Jl Flamboyan	67	57,8200	34,1834	Load
36	200	Ds Tanbuh	10	8,6000	5,1020	Load
37	201	Ds Tanbuh	8	6,8800	4,0816	Load
38	202	Ds Tanbuh	2	1,7200	1,0204	Load
39	0		0	0,0000	0,0000	Load
40	168	Bukit Pinus	12	10,3200	6,1224	Load
41	174	Jl Songgoriti	94	80,8400	47,9588	Load
42	175	Jl Songgoriti	16	13,7600	8,1632	Load
43	16	Jl Songgoriti	71	61,0600	36,2242	Load
44			0	0,0000	0,0000	Load
45	57	Dk Sebaluh	65	55,9000	33,1630	Load
46	153	Dk Sebaluh	83	71,3800	42,3466	Load

47	59	Dk Maron	69	59,3400	35,2038	Load
48	87	Dk Maron	65	55,9000	33,1630	Load
49			0	0,0000	0,0000	Load
50	112	Ds Gunung Sari	55	47,3000	28,0610	Load
51			0	0,0000	0,0000	Load
52	88	Ds Pujon Kidul	34	29,2400	17,3468	Load
53	89	Ds Pujon Kidul	103	88,5800	52,5506	Load
54	101	Dk Biyan	33	28,3800	16,8366	Load
55	102	Dk Bakir	82	70,5200	41,8364	Load
56			0	0,0000	0,0000	Load
57	103	Dk Cukal	38	32,6800	19,3876	Load
58	167	Dk dadapan	12	10,3200	6,1224	Load
59			0	0,0000	0,0000	Load
60	104	Ds Cukal	45	38,7000	22,9590	Load
61	166	Dk Tretes	23	19,7800	11,7346	Load
62	167	Dk Dadapan	12	10,3200	6,1224	Load
63			0	0,0000	0,0000	Load
64	58	Dk Pandemas	36	30,9600	18,3672	Load
65	60	Ds Jurang Rejo	120	103,2000	61,2240	Load
66	217	Ds Brau	14	12,0400	7,1428	Load
67			0	0,0000	0,0000	Load
68	212	Dk Don Bayi	16	13,7600	8,1632	Load
69	38	Ds Pandesari	108	82,8800	55,1018	Load
70	145	Dk Watu Gong	47	40,4200	23,9794	Load
71	19	Jl Raya Pujon	83	71,3800	42,3466	Load
72			0	0,0000	0,0000	Load
73	198	Pasar Baru	55	47,3000	28,0610	Load
74	123	Jl Raya Pujon	105	90,3000	53,5710	Load
75			0	0,0000	0,0000	Load
76			0	0,0000	0,0000	Load
77	98	Dk Kalangan	70	60,2000	35,7140	Load
78	79	Ds Wiyun Rejo	123	105,7800	62,7546	Load
79	80	Ds Madirejo	47	40,4200	23,9794	Load
80	223	Dk Sobo	16	13,7600	8,1632	Load
81	94	Dk Delik	54	46,4400	27,5508	Load
82	216	Dk Delik	77	66,2200	39,2854	Load
83			0	0,0000	0,0000	Load
84	195	KOP SAE	29	24,9400	14,7958	Load
85			0	0,0000	0,0000	Load
86	62	KOP SAE	114	98,0400	58,1628	Load
87	33	Ds Ngroto	132	113,5200	67,3464	Load
88	61	Ds Ngroto	72	61,9200	36,7344	Load
89			0	0,0000	0,0000	Load
90	64	Dk Iebaksari	36	30,9600	18,3672	Load
91	147	Dk Torong	29	24,9400	14,7958	Load
92	63	Dk Mantung	43	36,9600	21,9386	Load
93			0	0,0000	0,0000	Load
94	196	Dk Bunder	18	13,7600	8,1632	Load
95			0	0,0000	0,0000	Load
96	197	Dk Ngebrong	16	13,7600	8,1632	Load
97	65	Ds Ngabab	123	105,7800	62,7546	Load
98	66	Ds Ngabab	63	54,1800	32,1426	Load
99	219	Ds Manting	46	39,5600	23,4692	Load
100	76	Ds Manting	0	0,0000	0,0000	Load
101	77	Dk Gerih	100	86,0000	51,0200	Load

Sumber PT PLN (Persero) Distribusi JATIM  
Area Malang UP - J Batu

### Aliran Daya Tiap Saluran Sebelum Kompensasi

Saluran	P (kW)	Q (kVAR)
1-2	3928.102	2430.760
2-3	3876.687	2393.998
3-4	3800.336	2333.592
4-5	36.980	21.939
4-6	3753.913	2297.220
6-7	3719.071	2269.700
7-8	325.181	193.010
8-9	270.988	160.848
9-10	270.965	160.812
10-11	148.795	88.287
11-12	68.807	40.826
12-13	68.805	40.824
12-14	0.000	0.000
7-15	3371.889	2043.057
15-16	151.362	89.799
16-17	118.680	70.408
15-18	3209.581	1936.522
18-19	62.781	37.246
18-20	67.082	39.798
18-21	3077.005	1855.333
21-22	77.402	45.920
21-23	2997.355	1805.974
23-24	143.627	85.213
23-25	274.502	162.825
25-26	202.235	119.953
26-27	97.180	57.653
26-28	104.970	62.264
28-29	104.962	62.261
23-30	2570.392	1544.434
30-31	2543.517	1524.268
31-32	33.541	19.899
31-33	169.454	100.529
33-34	74.836	44.396
34-35	57.621	34.183
34-36	17.200	10.204
36-37	8.600	5.102
37-38	1.720	1.020
31-39	2339.801	1402.736
39-40	10.320	6.122
39-41	94.604	56.128
41-42	13.760	8.163
39-43	2230.794	1334.243
43-44	2169.574	1297.774
44-45	619.232	368.041
45-46	563.092	334.514
46-47	491.627	292.037
47-48	432.134	256.599
48-49	376.149	223.306
49-50	47.301	28.063
49-51	328.806	195.179
51-52	29.240	17.347
51-53	299.561	177.825

Saluran	P (kW)	Q (kVAR)
53-54	210.914	125.171
54-55	182.404	108.233
55-56	111.816	66.351
56-57	43.002	25.513
57-58	10.320	6.122
56-59	68.803	40.820
59-60	38.700	22.960
59-61	30.101	17.858
61-62	10.320	6.122
44-63	1544.162	920.287
63-64	146.220	86.760
64-65	115.250	68.378
65-66	12.040	7.143
63-67	1396.995	832.079
67-68	13.760	8.163
67-69	1383.024	823.594
69-70	1289.976	768.234
70-71	1248.369	742.440
71-72	1176.235	698.940
72-73	47.300	28.062
72-74	1128.904	670.832
74-75	1038.340	616.857
75-76	333.021	197.753
76-77	60.201	35.716
76-78	272.682	161.828
78-79	166.874	99.030
79-80	126.444	75.036
80-81	112.675	66.858
81-82	66.225	39.293
75-83	705.148	418.841
83-84	24.940	14.940
83-85	680.079	403.848
85-86	98.042	58.166
85-87	582.000	345.626
87-88	468.424	278.194
88-89	406.400	241.301
89-90	55.902	33.166
90-91	24.941	14.797
89-92	350.385	207.961
92-93	313.384	185.091
93-94	13.761	8.163
93-95	299.539	177.699
95-96	13.760	8.163
95-97	285.696	169.489
97-98	179.851	106.696
98-99	125.580	74.502
99-100	86.019	51.031
100-101	86.005	51.023

### Tegangan dan Sudut Fasa Tiap Node Penyulang Pujon Sebelum Kompensasi

Bus	Tegangan re (pu)	Sudut Fasa (rad)
1	1.00000	0.00000
2	0.99760	-0.064438
3	0.99176	-0.22279
4	0.98829	-0.31765
5	0.98829	-0.31774
6	0.98558	-0.39240
7	0.97737	-0.62113
8	0.97732	-0.62265
9	0.97720	-0.62605
10	0.97694	-0.63343
11	0.97687	-0.63552
12	0.97684	-0.63646
13	0.97674	-0.63944
14	0.97684	-0.63646
15	0.97290	-0.74727
16	0.97289	-0.74769
17	0.97289	-0.74777
18	0.97174	-0.78014
19	0.97173	-0.78045
20	0.97171	-0.78087
21	0.97074	-0.80861
22	0.97072	-0.80923
23	0.96671	-0.92380
24	0.96665	-0.92561
25	0.96662	-0.92370
26	0.96624	-0.92044
27	0.96624	-0.92042
28	0.96618	-0.91989
29	0.96582	-0.91679
30	0.96432	-0.99257
31	0.96394	-1.00371
32	0.96391	-1.00459
33	0.96383	-1.00360
34	0.96364	-1.00340
35	0.96363	-1.00339
36	0.96363	-1.00367
37	0.96363	-1.00375
38	0.96363	-1.00376
39	0.96156	-1.07225
40	0.96156	-1.07224
41	0.96151	-1.07387
42	0.96151	-1.07392
43	0.96147	-1.07506
44	0.95761	-1.18734
45	0.95708	-1.20269
46	0.95688	-1.20867
47	0.95646	-1.22100
48	0.95619	-1.22881
49	0.95605	-1.23317
50	0.95601	-1.23426
51	0.95603	-1.23373
52	0.95602	-1.23385

Bus	Tegangan re (Opu)	Sudut Fasa (rad)
53	0.95572	-1.24266
54	0.95509	-1.24741
55	0.95470	-1.25028
56	0.95457	-1.25427
57	0.95450	-1.25612
58	0.95449	-1.25647
59	0.95454	-1.25514
60	0.95453	-1.25550
61	0.95450	-1.25618
62	0.95449	-1.25656
63	0.95678	-1.21154
64	0.95669	-1.21418
65	0.95658	-1.21749
66	0.95655	-1.21816
67	0.95658	-1.21751
68	0.95656	-1.21787
69	0.95641	-1.22232
70	0.95517	-1.25870
71	0.95436	-1.28260
72	0.95432	-1.28362
73	0.95431	-1.28397
74	0.95401	-1.29290
75	0.95370	-1.29945
76	0.95323	-1.31576
77	0.95321	-1.31646
78	0.95309	-1.31986
79	0.95301	-1.32226
80	0.95291	-1.32517
81	0.95280	-1.32855
82	0.95269	-1.33156
83	0.95354	-1.30669
84	0.95353	-1.30684
85	0.95347	-1.30883
86	0.95344	-1.30959
87	0.95334	-1.31264
88	0.95304	-1.32145
89	0.95266	-1.33255
90	0.95263	-1.33337
91	0.95260	-1.33429
92	0.95258	-1.33487
93	0.95222	-1.34549
94	0.95218	-1.34545
95	0.95196	-1.34522
96	0.95195	-1.34520
97	0.95175	-1.34499
98	0.95127	-1.33450
99	0.95126	-1.33490
100	0.95111	-1.34474
101	0.95105	-1.34468



### Rugi-Rugi Aliran Daya Tiap Saluran Sebelum Kompensasi

Saluran	P (kW)	Q (kVAR)
1-2	6.695	10.232
2-3	16.151	24.692
3-4	9.673	14.433
4-5	0.000	0.000
4-6	7.322	11.194
6-7	22.001	33.633
7-8	0.013	0.019
8-9	0.023	0.036
9-10	0.050	0.077
10-11	0.008	0.012
11-12	0.002	0.002
12-13	0.005	0.008
12-14	0.000	0.000
7-15	10.946	16.736
15-16	0.002	0.003
16-17	0.000	0.000
15-18	1.713	4.145
18-19	0.001	0.001
18-20	0.002	0.002
18-21	2.249	3.439
21-22	0.002	0.002
21-23	8.834	13.502
23-24	0.007	0.010
23-25	0.027	0.015
25-26	0.033	0.036
26-27	0.000	0.000
26-28	0.008	0.003
28-29	0.042	0.017
23-30	4.515	6.901
30-31	0.721	1.104
31-32	0.001	0.001
31-33	0.018	0.011
33-34	0.015	0.008
34-35	0.001	0.000
34-36	0.000	0.000
36-37	0.000	0.000
37-38	0.000	0.000
31-39	4.083	16.243
39-40	0.000	0.000
39-41	0.004	0.006
41-42	0.000	0.000
39-43	0.160	0.245
43-44	6.180	9.446
44-45	0.240	0.367
45-46	0.085	0.130
46-47	0.153	0.234
47-48	0.085	0.130
48-49	0.041	0.064
49-50	0.001	0.002
49-51	0.004	0.007
51-52	0.000	0.000
51-53	0.067	0.103

Saluran	P (kW)	Q (kVAR)
53-54	0.132	0.101
54-55	0.068	0.132
55-56	0.011	0.015
56-57	0.002	0.003
57-58	0.000	0.000
56-59	0.002	0.002
59-60	0.000	0.000
59-61	0.000	0.000
61-62	0.000	0.000
44-63	0.947	1.448
63-64	0.001	0.015
64-65	0.000	0.000
65-66	0.000	0.000
63-67	0.220	0.322
67-68	0.000	0.000
67-69	0.168	0.258
69-70	1.187	1.815
70-71	0.754	1.153
71-72	0.030	0.046
72-73	0.000	0.000
72-74	0.264	0.404
74-75	0.172	0.263
75-76	0.138	0.209
76-77	0.001	0.002
76-78	0.028	0.043
78-79	0.020	0.015
79-80	0.009	0.015
80-81	0.010	0.014
81-82	0.005	0.008
75-83	0.129	0.197
83-84	0.000	0.000
83-85	0.037	0.056
85-86	0.002	0.003
85-87	0.056	0.086
87-88	0.104	0.149
88-89	0.114	0.174
89-90	0.001	0.002
90-91	0.001	0.000
89-92	0.021	0.031
92-93	0.084	0.129
93-94	0.001	0.000
93-95	0.083	0.047
95-96	0.000	0.000
95-97	0.065	0.038
97-98	0.091	0.051
98-99	0.001	0.001
99-100	0.014	0.008
100-101	0.005	0.003

**Tabel Aliran Daya Tiap Saluran Setelah Kompensasi**

Saluran	P (kW)	Q (kVAR)
1-2	3903.610	139.573
2-3	3854.102	105.691
3-4	3782.363	52.335
4-5	36.980	21.939
4-6	3738.642	20.092
6-7	3705.896	-4.223
7-8	325.156	42.972
8-9	270.967	10.815
9-10	270.950	10.789
10-11	148.793	-61.715
11-12	68.807	40.826
12-13	68.805	40.824
12-14	-0.000	0.000
7-5	3365.042	-71.194
15-16	151.362	89.799
16-17	118.680	70.408
15-18	3205.876	-172.924
18-19	62.781	37.246
18-20	67.082	39.798
18-21	3074.076	-252.928
21-22	77.402	45.920
21-23	2995.063	-301.312
23-24	143.626	85.213
23-25	274.653	-287.111
25-26	202.372	-329.991
26-27	97.180	57.653
26-28	104.968	62.264
28-29	104.961	62.261
23-30	2570.436	-109.117
30-31	2544.863	-127.292
31-32	33.541	19.899
31-33	169.453	100.528
33-34	74.835	44.396
34-35	57.621	34.183
34-36	17.200	10.204
36-37	8.600	5.102
37-38	10720	1.020
31-39	2341.356	-248.505
39-40	10.320	6.122
39-41	94.604	56.128
41-42	13.760	8.163
39-43	2233.502	-315.236
43-44	2172.325	-351.638
44-45	619.440	-81.888
45-46	563.367	-115.312
46-47	491.924	-157.755
47-48	432.465	-193.141
48-49	376.493	-226.414
49-50	47.301	28.063
49-51	329.152	-254.538
51-52	29.240	17.347
51-53	299.906	-271.893

Saluran	P (kW)	Q (kVAR)
53-54	211.240	-324.577
54-55	182.550	-341.654
55-56	111.815	66.349
56-57	43.002	25.513
57-58	10.320	6.122
56-59	68.803	40.820
59-60	38.700	22.960
59-61	30.101	17.858
61-62	10.320	6.122
44-63	1548.387	-276.626
63-64	146.219	86.759
64-65	115.249	68.377
65-66	12.040	7.143
63-67	1401.475	-364.444
67-68	13.760	8.163
67-69	1387.555	-372.852
69-70	1294.547	-428.150
70-71	1253.193	-453.558
71-72	1181.183	496.868
72-73	47.300	28.062
72-74	1133.883	-524.929
74-75	1043.196	579.092
75-76	333.011	197.739
76-77	60.201	35.716
76-78	272.679	161.823
78-79	166.872	99.027
79-80	126.443	75.034
80-81	112.674	66.857
81-82	66.225	39.292
75-83	710.185	-776.830
83-84	24.940	14.796
83-85	685.044	-791.935
85-86	98.042	58.166
85-87	586.941	-850.194
87-88	473.297	-917.729
88-89	411.022	-955.006
89-90	55.902	33.166
90-91	354.598	-988.970
89-92	354.598	-988.970
92-93	317.489	-1011.106
93-94	13.761	8.163
93-95	303.057	-1020.296
95-96	13.760	8.163
95-97	288.564	-1028.876
97-98	182.147	-1091.995
98-99	125.579	74.501
99-100	86.018	51.030
100-101	86.005	51.023

### Tegangan dan Sudut Fasa Tiap Node Penyulang Pujon Setelah Kompensasi

Bus	Tegangan re (pu)	Sudut Fasa (deg)
1	1.00000	0.00000
2	0.99871	0.10487
3	0.99561	0.36305
4	0.99380	0.51797
5	0.99380	0.51806
6	0.99240	0.64014
7	0.98823	1.01303
8	0.98819	1.01525
9	0.98813	1.02056
10	0.98799	1.03203
11	0.98797	1.03628
12	0.98794	1.09721
13	0.98784	1.04011
14	0.98794	1.03721
15	0.98601	1.21938
16	0.98600	1.21976
17	0.98600	1.21983
18	0.98547	1.27401
19	0.98546	1.27431
20	0.98544	1.27472
21	0.98502	1.32210
22	0.98499	1.32269
23	0.98328	1.51830
24	0.98322	1.52005
25	0.98325	1.52484
26	0.98315	1.56025
27	0.98315	1.56023
28	0.98309	1.55971
29	0.98274	1.55672
30	0.98213	1.63072
31	0.98195	1.64895
32	0.98192	1.64980
33	0.98185	1.64885
34	0.98166	1.64866
35	0.98165	1.64865
36	0.98165	1.64891
37	0.98165	1.64899
38	0.98165	1.64900
39	0.98093	1.76505
40	0.98093	1.76504
41	0.98088	1.76661
42	0.98088	1.76665
43	0.98090	1.76991
44	0.97941	1.96566
45	0.97920	1.99175
46	0.97913	2.00234
47	0.97902	2.02566
48	0.97897	2.04142
49	0.97897	2.05089
50	0.97893	2.05194
51	0.97897	2.05222
52	0.97896	2.05233

Bus	Tegangan re (pu)	Sudut Fasa (deg)
53	0.97903	2.07441
54	0.97911	2.13208
55	0.97923	2.17176
56	0.97909	2.17555
57	0.97903	2.17731
58	0.97902	2.17764
59	0.97907	2.17638
60	0.97905	2.17672
61	0.97903	2.17737
62	0.97902	2.17773
63	0.97910	2.00811
64	0.97902	2.01063
65	0.97891	2.01379
66	0.97888	2.01443
67	0.97904	2.01906
68	0.97903	2.01940
69	0.97899	2.02793
70	0.97868	2.09728
71	0.97849	2.14351
72	0.97848	2.14554
73	0.97847	2.14587
74	0.97844	2.16443
75	0.97842	2.17836
76	0.97788	2.19386
77	0.97785	2.19452
78	0.97774	2.19776
79	0.97766	2.20003
80	0.97757	2.20280
81	0.97746	2.20601
82	0.97736	2.20888
83	0.97850	2.19777
84	0.97850	2.19791
85	0.97853	2.20363
86	0.97851	2.20436
87	0.97861	2.21524
88	0.97892	2.24656
89	0.97641	2.29037
90	0.97939	2.29114
91	0.97936	2.29201
92	0.97955	2.30061
93	0.98027	2.35184
94	0.98023	2.35180
95	0.98044	2.39603
96	0.98043	2.39601
97	0.98061	2.43418
98	0.98145	2.56756
99	0.98144	2.56794
100	0.98129	2.56779
101	0.98123	2.56773

### Rugi-rugi Aliran Daya Tiap Saluran Setelah Kompensasi

Saluran	P (kW)	Q (kVAR)
1-2	4.788	7.316
2-3	11.539	17.642
3-4	6.741	10.304
4-5	0.000	0.000
4-6	5.226	7.989
6-7	15.699	23.999
7-8	0.009	0.014
8-9	0.017	0.026
9-10	0.037	0.056
10-11	0.006	0.100
11-12	0.002	0.002
12-13	0.005	0.008
12-14	0.000	0.000
7-15	7.804	11.931
15-16	0.002	0.003
16-17	0.000	0.000
15-18	0.038	0.040
18-19	0.001	0.001
18-20	0.002	0.002
18-21	1.612	2.464
21-22	0.002	0.002
21-23	6.348	9.703
23-24	0.002	0.010
23-25	0.041	0.023
25-26	0.224	0.092
26-27	0.000	0.000
26-28	0.007	0.003
28-29	0.041	0.017
23-30	3.213	4.910
30-31	0.513	0.786
31-32	0.001	0.001
31-33	0.0018	0.010
33-34	0.014	0.008
34-35	0.001	0.000
34-36	0.000	0.000
36-37	0.000	0.000
37-38	0.000	0.000
31-39	2.931	4.482
39-40	0.000	0.000
39-41	0.004	0.006
41-42	0.000	0.000
39-43	0.117	0.178
43-44	4.498	6.876
44-45	0.173	0.264
45-46	0.003	0.096
46-47	0.119	0.182
47-48	0.072	0.110
48-49	0.040	0.061
49-50	0.001	0.002
49-51	0.005	0.008
51-52	0.000	0.000
51-53	0.086	0.133

Saluran	P (kW)	Q (kVAR)
53-54	0.031	0.240
54-55	0.215	0.167
55-56	0.010	0.016
56-57	0.002	0.003
57-58	0.000	0.000
56-59	0.002	0.002
59-60	0.000	0.000
59-61	0.001	0.001
61-62	0.000	0.000
44-63	0.693	1.060
63-64	0.010	0.015
64-65	0.009	0.014
65-66	0.000	0.000
63-67	0.159	0.245
67-68	0.000	0.000
67-69	0.128	0.804
69-70	0.934	1.429
70-71	0.605	0.894
71-72	0.025	0.039
72-73	0.000	0.001
72-74	0.228	0.349
74-75	0.159	0.757
75-76	0.131	0.199
76-77	0.001	0.002
76-78	0.027	0.041
78-79	0.005	0.014
79-80	0.009	0.014
80-81	0.009	0.014
81-82	0.005	0.007
75-83	0.201	0.308
83-84	0.000	0.000
83-85	0.061	0.093
85-86	0.002	0.003
85-87	0.124	0.189
87-88	0.355	0.543
88-89	0.522	0.799
89-90	0.001	0.002
90-91	0.001	0.001
89-92	0.126	0.197
92-93	0.671	1.027
93-94	0.001	0.000
93-95	0.732	0.417
95-96	0.000	0.000
95-97	0.637	0.364
97-98	2.388	64.116
98-99	0.001	0.002
99-100	0.013	0.007
100-101	0.005	0.003

unit Utama;

interface

uses

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

type

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

var

frmUtama: TfrmUtama;

implementation

uses Input, Komplex, NRpolar, TypeData;

{ \$R \*.dfm }

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

```
procedure TfrmUtama.btnNewClick(Sender: TObject);
begin
  frmInput.Caption:='Input Data';
  frmInput.btnSave.Caption:='-&Save';
  frmInput.cmbKonstV.Text:='-kV';
```

```

frmInput.cmbKonstP.Text:='MVA';
frmInput.cmbKonstParam.Text:='pu';
frmInput.Show;
end,

procedure TFormUtama.btnOpenClick(Sender: TObject);
var NamaFile:string;
    output:TextFile;
    i,j,Typ,dari,ke:integer;
    Cap,absV,sudV,Pgen,Qgen,Pload,Qload:double;
    R,X,Lc,Tr:double;
begin
try
if OpenFileDialog1.Execute then
begin
NamaFile:=OpenDialog1.FileName;
AssignFile(output,NamaFile);
Reset(output);
Readln(output,Nbus);
Readln(output,Nsal);
Readln(output,Vbase);
Readln(output,KonstV);
Readln(output,Pbase);
Readln(output,KonstP);
Readln(output,param);
Zbase:=sqrt(Vbase*KonstV)/(Pbase*KonstP);
Ibase:=(Vbase*KonstV)/Zbase;
frmInput.edtNbus.Text:=IntToStr(Nbus);
frmInput.edtNsal.Text:=IntToStr(Nsal);
frmInput.edtVbase.Text:=FloatToStr(Vbase);
if KonstV=1 then
begin
frmInput.cmbKonstV.Text:='V';
end
else if KonstV=1000 then
begin
frmInput.cmbKonstV.Text:='kV';
end
else if KonstV=1000000 then
begin
frmInput.cmbKonstV.Text:='MV';
end;
frmInput.edtPbase.Text:=FloatToStr(Pbase);
if KonstP=1 then
begin
frmInput.cmbKonstP.Text:='VA';

```

---

```

end
else if KonstP=1000 then
begin
  frmInput.cmbKonstP.Text:='kVA';
end
else if KonstP=1000000 then
begin
  frmInput.cmbKonstP.Text:='MVA';
end;
if param=1 then
begin
  frmInput.cmbKonstParam.Text:='pu';
end
else if param=2 then
begin
  frmInput.cmbKonstParam.Text:='ohm';
end;
SetLength(Va,Nbus+1);
SetLength(Sga,Nbus+1);
SetLength(SLa,Nbus+1);
SetLength(Capa,Nbus+1);
SetLength(Typa,Nbus+1);
SetLength(Za,Nbus+1,Nbus+1);
SetLength(Lca,Nbus+1,Nbus+1);
SetLength(Tra,Nbus+1,Nbus+1);
for i:=1 to Nbus do
begin
  Readln(output,absV,sudV,Pgen,Qgen,Pload,Qload,Cap,Typ);
  Va[i]:=TKomplex.Create(absV,sudV);
  Sga[i]:=TKomplex.Create(Pgen/Pbase,Qgen/Pbase);
  SLa[i]:=TKomplex.Create(Pload/Pbase,Qload/Pbase);
  Capa[i]:=cap;
  Typa[i]:=Typ;
  frmInput.fgBus.Cells[0,i]:=IntToStr(i);
  frmInput.fgBus.Cells[1,i]:=FloatToStr(absV);
  frmInput.fgBus.Cells[2,i]:=FloatToStr(sudV);
  frmInput.fgBus.Cells[3,i]:=FloatToStr(Pgen);
  frmInput.fgBus.Cells[4,i]:=FloatToStr(Qgen);
  frmInput.fgBus.Cells[5,i]:=FloatToStr(Pload);
  frmInput.fgBus.Cells[6,i]:=FloatToStr(Qload);
  frmInput.fgBus.Cells[7,i]:=FloatToStr(Cap);
  frmInput.fgBus.Cells[8,i]:=IntToStr(Typ);
end;
for i:=1 to Nbus do
begin
  for j:=1 to Nbus do

```

```

begin
  Za[i,j]:=TKomplex.Create;
end;
end;
for i:=1 to Nsal do
begin
  Readln(output,dari,ke,R,X,Lc,Tr);
  if param=1 then
  begin
    Za[dari,ke].Assign(R,X);
  end
  else if param=2 then
  begin
    Za[dari,ke].Assign(R/Zbase,X/Zbase);
  end;
  Lca[dari,ke]:=Lc;
  Tra[dari,ke]:=Tr;
  frmInput.fgSaluran.Cells[0,i]:=IntToStr(i);
  frmInput.fgSaluran.Cells[1,i]:=IntToStr(dari);
  frmInput.fgSaluran.Cells[2,i]:=IntToStr(ke);
  frmInput.fgSaluran.Cells[3,i]:=FloatToStr(R);
  frmInput.fgSaluran.Cells[4,i]:=FloatToStr(X);
  frmInput.fgSaluran.Cells[5,i]:=FloatToStr(Lc);
  frmInput.fgSaluran.Cells[6,i]:=FloatToStr(Tr);
end;
Rcadln(output,Ncap);
Readln(output,Kp);
frmInput.edtNcap.Text:=IntToStr(Ncap);
frmInput.edtKp.Text:=RealToStr(Kp,2);
SetLength(Capasitor,Ncap+1);
SetLength(BiayaCap,Ncap+1);
for i:=1 to Ncap do
begin
  Readln(output,R,X);
  Capasitor[i]:=R;
  BiayaCap[i]:=X;
  frmInput.fgCap.Cells[1,i]:=RealToStr(R,0);
  frmInput.fgCap.Cells[2,i]:=RealToStr(X,5);
end;
CloseFile(output);
frmInput.Caption:='Tampilan Data';
frmInput.btnSave.Caption:='&Next';
frmInput.Show;
end;
except
  MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);

```



end;  
end;

end.

```
unit About;
```

```
interface
```

```
uses
```

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

```
type
```

```
  TfrmAbout = class(TForm)
```

```
  private
```

```
    { Private declarations }
```

```
  public
```

```
    { Public declarations }
```

```
  end;
```

```
var
```

```
  frmAbout: TfrmAbout;
```

```
implementation
```

```
  {$R *.dfm}
```

```
end.
```

---

unit Input;

interface

uses

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

type

```
TfrmInput = class(TForm)
  PageControl1: TPageControl;
  TabSheet1: TTabSheet;
  TabSheet2: TTabSheet;
  TabSheet3: TTabSheet;
  Panel1: TPanel;
  Label1: TLabel;
  Label2: TLabel;
  edtNbus: TEdit;
  edtNsal: TEdit;
  fgBus: TStringGrid;
  fgSaluran: TStringGrid;
  btnClose: TButton;
  btnSave: TButton;
  label3: TLabel;
  Label4: TLabel;
  edtVbase: TEdit;
  Label5: TLabel;
  cmbKonstV: TComboBox;
  cmbKonstP: TComboBox;
  cmbKonstParam: TComboBox;
  edtPbase: TEdit;
  SaveDialog1: TSaveDialog;
  TabSheet4: TTabSheet;
  Label6: TLabel;
  edtNCap: TEdit;
  Label7: TLabel;
  edtKp: TEdit;
  fgCap: TStringGrid;
  procedure btnCloseClick(Sender: TObject);
  procedure edtNbusChange(Sender: TObject);
  procedure edtNsalChange(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure cmbKonstPChange(Sender: TObject);
  procedure cmbKonstParamChange(Sender: TObject);
  procedure btnSaveClick(Sender: TObject);
  procedure edtNCapChange(Sender: TObject);
```

---

```

private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmInput: TfrmInput;

implementation

uses Hasil, TypeData;

{$R *.dfm}

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

procedure TfrmInput.edtNbusChange(Sender: TObject);
var i:integer;
begin
  if edtNbus.Text="" then
  begin
    fgBus.RowCount:=2;
  end
  else
  begin
    fgBus.RowCount:=StrToInt(edtNbus.Text)+1;
    for i:=1 to StrToInt(edtNbus.Text) do
    begin
      fgBus.Cells[0,i]:=IntToStr(i);
      fgBus.Cells[1,i]:=IntToStr(1);
      fgBus.Cells[2,i]:=IntToStr(0);
    end;
  end;
end;

procedure TfrmInput.edtNsalChange(Sender: TObject);
var i:integer;
begin
  if edtNsal.Text="" then
  begin
    fgSaluran.RowCount:=2;
  end
end;

```

---

```

else
begin
  fgSaluran.RowCount:=StrToInt(edtNsal.Text)+1;
  for i:=1 to StrToInt(edtNsal.Text) do
  begin
    fgSaluran.Cells[0,i]:=IntToStr(i);
  end;
end;
end;
end;

```

```

procedure TfrmInput.FormCreate(Sender: TObject);
begin
  fgBus.Cells[0,0]:='No';
  fgBus.Cells[1,0]:='Abs V (pu)';
  fgBus.Cells[2,0]:='Sud V (deg)';
  fgBus.Cells[3,0]:='Pgen (MW)';
  fgBus.Cells[4,0]:='Qgen (MVAR)';
  fgBus.Cells[5,0]:='Pload (MW)';
  fgBus.Cells[6,0]:='Qload (MVAR)';
  fgBus.Cells[7,0]:='Cap';
  fgBus.Cells[8,0]:='Typ Bus';
  fgSaluran.Cells[0,0]:='No';
  fgSaluran.Cells[1,0]:='Dari';
  fgSaluran.Cells[2,0]:='Ke';
  fgSaluran.Cells[3,0]:='R (pu)';
  fgSaluran.Cells[4,0]:='X (pu)';
  fgSaluran.Cells[5,0]:='Le (pu)';
  fgSaluran.Cells[6,0]:='Tr';
  fgCap.Cells[0,0]:='No';
  fgCap.Cells[1,0]:='Cap (kVAR)';
  fgCap.Cells[2,0]:='Tlarga (kVAR/$)';
end;

```

```

procedure TfrmInput.cmbKonstPChange(Sender: TObject);
begin
  if cmbKonstP.Text='VA' then
  begin
    fgBus.Cells[3,0]:='Pgen (W)';
    fgBus.Cells[4,0]:='Qgen (VAR)';
    fgBus.Cells[5,0]:='Pload (W)';
    fgBus.Cells[6,0]:='Qload (VAR)';
  end
  else if cmbKonstP.Text='kVA' then
  begin
    fgBus.Cells[3,0]:='Pgen (kW)';
    fgBus.Cells[4,0]:='Qgen (kVAR)';
  end;
end;

```

```

    fgBus.Cells[5,0]:='Pload (kW)';
    fgBus.Cells[6,0]:='Qload (kVAR)';
end
else if cmbKonstP.Text='MVA' then
begin
    fgBus.Cells[3,0]:='Pgen (MW)';
    fgBus.Cells[4,0]:='Qgen (MVAR)';
    fgBus.Cells[5,0]:='Pload (MW)';
    fgBus.Cells[6,0]:='Qload (MVAR)';
end;
end;

procedure TForm1.cmbKonstParamChange(Sender: TObject);
begin
    if cmbKonstParam.Text='pu' then
    begin
        fgSaluran.Cells[3,0]:='R (pu)';
        fgSaluran.Cells[4,0]:='X (pu)';
        fgSaluran.Cells[5,0]:='Lc (pu)';
    end
    else if cmbKonstParam.Text='ohm' then
    begin
        fgSaluran.Cells[3,0]:='R (ohm)';
        fgSaluran.Cells[4,0]:='X (ohm)';
        fgSaluran.Cells[5,0]:='Lc (ohm)';
    end;
end;

procedure TForm1.btnSaveClick(Sender: TObject);
var NamaFile:string;
    input:TextFile;
    i,typ,dari,ke:integer;
    R,X,Lc,Tr:double;
    absV,sudV,Pgen,Qgen,Pload,Qload,Cap:double;
begin
    KonstP:=1000000;
    KonstV:=1000;
    param:=1;
    if cmbKonstP.Text='VA' then
    begin
        KonstP:=1;
    end
    else if cmbKonstP.Text='kVA' then
    begin
        KonstP:=1000;
    end
end

```

---

```

else if cmbKonstP.Text='MVA' then
begin
  KonstP:=1000000;
end;
if cmbKonstV.Text='V' then
begin
  KonstV:=1;
end
else if cmbKonstV.Text='kV' then
begin
  KonstV:=1000;
end
else if cmbKonstV.Text='MV' then
begin
  KonstV:=1000000;
end;
if cmbKonstParam.Text='pu' then
begin
  param:=1;
end
else if cmbKonstParam.Text='ohm' then
begin
  param:=2;
end;
if btnSave.Caption='&Save' then
begin
  try
    if SaveDialog1.Exccute then
    begin
      Nbus:=StrToInt(edtNbus.Text);
      Nsal:=StrToInt(edtNsal.Text);
      Ncap:=StrToInt(edtNcap.Text);
      Vbase:=StrToFloat(edtVbase.Text);
      Pbase:=StrToFloat(edtPbase.Text);
      Kp:=StrToFloat(edtKp.Text);
      NamaFile:=SaveDialog1.FileName;
      AssignFile(input,NamaFile+'.txt');
      Rewrite(input);
      Writeln(input,Nbus);
      Writeln(input,Nsal);
      Writeln(input,Vbase:6:2);
      Writeln(input,KonstV:7:0);
      Writeln(input,Pbase:6:2);
      Writeln(input,KonstP:7:0);
      Writeln(input,param);
      for i:=1 to Nbus do

```

```

begin
  absV:=StrToFloat(fgBus.Cells[1,i]);
  sudV:=StrToFloat(fgBus.Cells[2,i]);
  Pgen:=StrToFloat(fgBus.Cells[3,i]);
  Qgen:=StrToFloat(fgBus.Cells[4,i]);
  Pload:=StrToFloat(fgBus.Cells[5,i]);
  Qload:=StrToFloat(fgBus.Cells[6,i]);
  Cap:=StrToFloat(fgBus.Cells[7,i]);
  Typ:=StrToInt(fgBus.Cells[8,i]);
  Writeln(input,absV:7:5,',',sudV:7:5,',',Pgen:9:3,',',Qgen:9:3,
  ',Pload:9:3,',',Qload:9:3,',',Cap:7:5,',',Typ);
end;
for i:=1 to Nsal do
begin
  dari:=StrToInt(fgSaluran.Cells[1,i]);
  ke:=StrToInt(fgSaluran.Cells[2,i]);
  R:=StrToFloat(fgSaluran.Cells[3,i]);
  X:=StrToFloat(fgSaluran.Cells[4,i]);
  Lc:=StrToFloat(fgSaluran.Cells[5,i]);
  Tr:=StrToFloat(fgSaluran.Cells[6,i]);
  Writeln(input,dari,',',ke,',',R:7:5,',',X:7:5,',',
  Lc:7:5,',',Tr:7:5);
end;
Writeln(input,Ncap);
Writeln(input,Kp:10:2);
for i:=1 to Ncap do
begin
  R:=StrToFloat(fgCap.Cells[1,i]);
  X:=StrToFloat(fgCap.Cells[2,i]);
  Writeln(input,R:7:0,',',X:7:5);
end;
CloseFile(input);
MessageDlg('File berhasil disimpan!',mtInformation,[mbOK],0);
end;
except
  MessageDlg('Tolong dicek angka-angkanya kembali!',mtWarning,[mbOK],0);
end;
end
else if btnSave.Caption='&Next' then
begin
  frmHasil.Show;
  frmHasil.fgLoadflow.RowCount:=Nbus-1;
  frmHasil.fgLoadflow2.RowCount:=Nbus+1;
  frmHasil.fgAliranDaya.RowCount:=Nsal*2+1;
  frmHasil.fgAliranDaya2.RowCount:=Nsal*2+1;
end;

```



```
end;

procedure TForm1.edtNCapChange(Sender: TObject);
var i:integer;
begin
  try
    if edtNcap.Text="" then
      begin
        fgCap.RowCount:=2;
      end
    else if StrToInt(edtNcap.Text)=0 then
      begin
        fgCap.RowCount:= 2;
      end
    else if StrToInt(edtNcap.Text)>=1 then
      begin
        fgCap.RowCount:=StrToInt(edtNcap.Text)+ 1;
        for i:=1 to StrToInt(edtNcap.Text) do
          begin
            fgCap.Cells[0,i]:=IntToStr(i);
          end;
        end;
      end
    except
      messageDlg('Masukkan input angka!',mtWarning,[mbOK],0);
    end;
  end;
end;

end;
```

```
unit Komplex;
```

```
interface
```

```
uses SysUtils;
```

```
type
```

```
  Arr1=array of double;
```

```
  Arr2=array of array of double;
```

```
  iArr1=array of integer;
```

```
  iArr2=array of array of integer;
```

```
TKomplex=class
```

```
private
```

```
  re,im:double;
```

```
  function GetRe:double;
```

```
  function GetIm:double;
```

```
  procedure SetRe(const aRe:double);
```

```
  procedure SetIm(const aIm:double);
```

```
public
```

```
  constructor Create;overload;
```

```
  constructor Create(const aRe:double);overload;
```

```
  constructor Create(const aRe,aIm:double);overload;
```

```
  constructor Create(const aKomplex:TKomplex);overload;
```

```
  procedure Assign(const aRe,aIm:double);overload;
```

```
  procedure Assign(const aKomplex:TKomplex);overload;
```

```
  function doLambah(const aRe,aIm:double):TKomplex;overload;
```

```
  function doTambah(const aKomplex:TKomplex):TKomplex;overload;
```

```
  function doKurang(const aRe,aIm:double):TKomplex;overload;
```

```
  function doKurang(const aKomplex:TKomplex):TKomplex;overload;
```

```
  function doKali(const aRe:double):TKomplex;overload;
```

```
  function doKali(const aRe,aIm:double) TKomplex;overload;
```

```
  function doKali(const aKomplex:TKomplex):TKomplex;overload;
```

```
  function doBagi(const aRe,aIm:double):TKomplex;overload;
```

```
  function doBagi(const aKomplex:TKomplex):TKomplex;overload;
```

```
  function doConj:TKomplex;
```

```
  function doNegative:TKomplex;
```

```
  function GetAbs:double;
```

```
  function GetAngleRad:double;
```

```
  function GetAngleDeg:double;
```

```
  function doPangkat(const pangkat:double):TKomplex;
```

```
  function toStringJ(len:byte):string;
```

```
  function toStringI(len:byte):string;
```

```
  destructor Destroy;override;
```

```
  property xRe:double read GetRe write SetRe;
```

```
  property xIm:double read GetIm write SetIm;
```

```

end;

CArr1=array of TKomplex;
CArr2=array of array of TKomplex;

function RealToStr(Num:double;Pecahan:byte):String;
function StrToReal(Huruf:string):double;
function Pangkat(Val,pangkat:double):double;

implementation

function RealToStr(Num:double;Pecahan:byte):String;
var Hasil:String;
    le:byte;
begin
    le:=sizeof(Num);
    Str(Num:le:Pecahan,Hasil);
    Result:=Hasil;
end;

function Pangkat(Val,pangkat:double):double;
begin
    Result:=exp(Pangkat*ln(Val));
end;

function StrToReal(Huruf:string):double;
var Temp:double;
    Code:integer;
begin
    val(Huruf,Temp,Code);
    Result:=Temp;
end;

//constructor
constructor TKomplex.Create;
begin
    inherited Create;
    re:=0;
    im:=0;
end;

constructor TKomplex.Create(const aRe:double);
begin
    inherited Create;
    re:=aRe;
    im:=0;

```

---

```
end;

constructor TKomplex.Create(const aRe,aIm:double);
begin
  inherited Create;
  re:=aRe;
  im:=aIm;
end;

constructor TKomplex.Create(const aKomplex:TKomplex);
begin
  inherited Create;
  re:=aKomplex.xRe;
  im:=aKomplex.xIm;
end;

//data accessing
function TKomplex.GetRe:double;
begin
  result:=re;
end;

function TKomplex.GetIm:double;
begin
  result:=im;
end;

procedure TKomplex.SetRe(const aRe:double);
begin
  re:=aRe;
end;

procedure TKomplex.SetIm(const aIm:double);
begin
  im:=aIm;
end;

procedure TKomplex.Assign(const aRe,aIm:double);
begin
  re:=aRe;
  im:=aIm;
end;

procedure TKomplex.Assign(const aKomplex:TKomplex);
begin
  re:=aKomplex.xRe;
```

---

```
    im:=aKomplex.xIm;  
end;
```

```
//data processing  
function TKomplex.doTambah(const aRe,aim:double):TKomplex;  
begin  
    result:=TKomplex.Create;  
    result.xRe:=re+aRe;  
    result.xIm:=re+aim;  
end;
```

```
function TKomplex.doTambah(const aKomplex:TKomplex):TKomplex;  
begin  
    result:=TKomplex.Create;  
    result.xRe:=re+aKomplex.xRe;  
    result.xIm:=im+aKomplex.xIm;  
end;
```

```
function TKomplex.doKurang(const aRe,aim:double):TKomplex;  
begin  
    result:=TKomplex.Create;  
    result.xRe:=re-aRe;  
    result.xIm:=re-aim;  
end;
```

```
function TKomplex.doKurang(const aKomplex:TKomplex):TKomplex;  
begin  
    result:=TKomplex.Create;  
    result.xRe:=re-aKomplex.xRe;  
    result.xIm:=im-aKomplex.xIm;  
end;
```

```
function TKomplex.doKali(const aRe:double):TKomplex;  
var tmpRe,tmpIm:double;  
begin  
    tmpRe:=re;  
    tmpIm:=im;  
    result:=TKomplex.Create;  
    result.xRe:=tmpRe*aRe;  
    result.xIm:=tmpIm*aRe;  
end;
```

```
function TKomplex.doKali(const aRe,aim:double):TKomplex;  
var tmpRe,tmpIm:double;  
begin  
    tmpRe:=re;
```

---

```

    tmpIm:=im;
    result:=TKomplex.Create;
    result.xRe:=-tmpRe*aRe-tmpIm*aIm;
    result.xIm:=tmpRe*aIm+tmpIm*aRe;
end;

```

```

function TKomplex.doKali(const aKomplex:TKomplex):TKomplex;
var tmpRe,tmpIm:double;
begin
    tmpRe:=re;
    tmpIm:=im;
    result:=TKomplex.Create;
    result.xRe:=tmpRe*aKomplex.xRe-tmpIm*aKomplex.xIm;
    result.xIm:=tmpRe*aKomplex.xIm+tmpIm*aKomplex.xRe;
end;

```

```

function TKomplex.doBagi(const aRe,aIm:double):TKomplex;
var tmpRe,tmpIm:double;
begin
    try
        tmpRe:=re;
        tmpIm:=im;
        result:=TKomplex.Create;
        result.xRe:=(tmpRe*aRe-tmpIm*aIm)/(sqrt(aRe)+sqrt(aIm));
        result.xIm:=(tmpIm*aRe+tmpRe*aIm)/(sqrt(aRe)+sqrt(aIm));
    except
        raise Exception.Create('Bilangan pembagi nol!');
    end;
end;

```

```

function TKomplex.doBagi(const aKomplex:TKomplex):TKomplex;
var tmpRe,tmpIm:double;
begin
    try
        tmpRe:=re;
        tmpIm:=im;
        result:=TKomplex.Create;
        result.xRe:=(tmpRe*aKomplex.xRe+tmpIm*aKomplex.xIm)/
            (sqrt(aKomplex.xRe)+sqrt(aKomplex.xIm));
        result.xIm:=(tmpIm*aKomplex.xRe-tmpRe*aKomplex.xIm)/
            (sqrt(aKomplex.xRe)+sqrt(aKomplex.xIm));
    except
        raise Exception.Create('Bilangan pembagi nol!');
    end;
end;

```

```
function TKomplex.doConj:TKomplex;  
var tmpRe,tmpIm:double;  
begin  
  tmpRe:=re;  
  tmpIm:=-im;  
  result:=TKomplex.Create(tmpRe,tmpIm);  
end;
```

```
function TKomplex.doNegative:TKomplex;  
var tmpRe,tmpIm:double;  
begin  
  tmpRe:=-re;  
  tmpIm:=-im;  
  result:=TKomplex.Create(tmpRe,tmpIm);  
end;
```

```
function TKomplex.GetAbs:double;  
begin  
  result:=sqrt(sqr(re)+sqr(im));  
end;
```

```
function TKomplex.GetAngleRad:double;  
begin  
  try  
    result:=arctan(im/re);  
  except  
    raise Exception.Create('Bilangan tidak bisa dicari sudutnya!');  
  end;  
end;
```

```
function TKomplex.GetAngleDeg:double;  
var pi:double;  
begin  
  try  
    pi:=4*arctan(1);  
    result:=arctan(im/re)*180/pi;  
  except  
    raise Exception.Create('Bilangan tidak bisa dicari sudutnya!');  
  end;  
end;
```

```
function TKomplex.doPangkat(const pangkat:double):TKomplex;  
var theta,sum:double;  
begin  
  try  
    result:=TKomplex.Create;
```

```

theta:=arctan(im/re);
sum:=exp((pangkat/2)*ln(sqr(re)+sqr(im)));
result.xRe:=sum*cos(pangkat*theta);
result.xIm:=sum*sin(pangkat*theta);
except
raise Exception.Create("Bilangan tidak bisa dipangkatkan!");
end;
end;

```

```

function TKomplex.toStringJ(len:byte):string;
begin
result:=RealToStr(re,len);
if im<0 then
begin
result:=result+' - j'+RealToStr(abs(im),len);
end
else if im>0 then
begin
result:=result+' + j'+RealToStr(abs(im),len);
end;
end;
end;

```

```

function TKomplex.toStringI(len:byte):string;
begin
result:=RealToStr(re,len);
if im<0 then
begin
result:=result+' - '+RealToStr(abs(im),len)+'i';
end
else if im>0 then
begin
result:=result+' + '+RealToStr(abs(im),len)+'i';
end;
end;
end;

```

```

//destructor
destructor TKomplex.Destroy;
begin
inherited Destroy;
end;

end.

```

---



```

unit NRpolar;

interface

uses Komplex,Loadflow,SysUtils,Matrik;

type
  TNRpolar=class(TLoadflow)
  private
    iterasi:integer;
    procedure MismatchDaya(const ald:integer;
      var Max:double;
      var dS:Arr1);
    procedure Jaqobian(const ald:integer;var Jaq:Arr2);
    procedure UpdateTegangan(const ald:integer;
      const dS:Arr1;const Jaq:Arr2);
    function GetIterasi:integer;
  public
    procedure doHitung;
    property LFIterasi:integer read GetIterasi;
  end;

```

implementation

```

procedure TNRpolar.MismatchDaya(const ald:integer;
  var max:double;
  var dS:Arr1);
var i,j,sp,sq:integer;
  sumP,sumQ,Gij,Bij,Ui,Uj,di,dj:double;
begin
  SetLength(dS,ald+1);
  sp:=0;
  sq:=LFNbus-1;
  for i:=1 to LFNbus do
  begin
    Ui:=GetV(i).xRe;
    di:=GetV(i).xIm;
    if GetTyp(i)<>1 then
    begin
      sp:=sp+1;
      sumP:=0;
      for j:=1 to LFNbus do
      begin
        Gij:=GetY(i,j).xRe;
        Bij:=GetY(i,j).xIm;
        Uj:=GetV(j).xRe;

```

```

    dj:=GetV(j).xIm;
    sumP:=sumP+Ui*Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
end;
dS[sp]:=GetSg(i).xRe-GetSL(i).xRe-sumP;
end;
if GetTyp(i)=3 then
begin
    sq:=sq+1;
    sumQ:=0;
    for j:=1 to LFNbus do
    begin
        Gij:=GetY(i,j).xRe;
        Bij:=GetY(i,j).xIm;
        Uj:=GetV(j).xRe;
        dj:=GetV(j).xIm;
        sumQ:=sumQ+Ui*Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
    end;
    dS[sq]:=GetSg(i).xIm-GetSL(i).xIm-sumQ;
end;
end;
max:=dS[1];
for i:=2 to aId do
begin
    if max<abs(dS[i]) then max:=abs(dS[i]);
end;
end;

```

```

procedure TNRpolar.Jacobian(const aId:integer;var Jaq:Arr2);
var i,j,k,row,col:integer;
    Gij,Bij,Ui,di,Uj,dj,Uk,dk,Gjk,Bjk,sum:double;
    Pj,Qj:double;
begin
    SetLength(Jaq,aId+1,aId+1);
    //Pembentukan Jacobian H
    row:=0;
    for i:=1 to LFNbus do
    begin
        if GetTyp(i)<>1 then
        begin
            row:=row+1;
            col:=0;
            Ui:=GetV(i).xRe;
            di:=GetV(i).xIm;
            for j:=1 to LFNbus do
            begin
                if GetTyp(j)<>1 then

```

```

begin
  col:=col+1;
  Uj:=GetV(j).xRe;
  dj:=GetV(j).xIm;
  Gij:=GetY(i,j).xRe;
  Bij:=GetY(i,j).xIm;
  if j=i then
    begin
      sum:=0;
      for k:=1 to LFNbus do
        begin
          Uk:=GetV(k).xRe;
          dk:=GetV(k).xIm;
          Gjk:=GetY(j,k).xRe;
          Bjk:=GetY(j,k).xIm;
          sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);
        end;
      Qj:=sum*Uj;
      Jaq[row,col]:=-Qj-Bij*sqr(Ui);
    end
  else
    begin
      Jaq[row,col]:=Ui*Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
    end;
  end;
end;
end;
end;
//Pembentukan Jaqobian L
row:=LFNbus-1;
for i:=1 to LFNbus do
begin
  if GetTyp(i)=3 then
    begin
      row:=row+1;
      col:=LFNbus-1;
      Ui:=GetV(i).xRe;
      di:=GetV(i).xIm;
      for j:=1 to LFNbus do
        begin
          if GetTyp(j)=3 then
            begin
              col:=col+1;
              Uj:=GetV(j).xRe;
              dj:=GetV(j).xIm;
              Gij:=GetY(i,j).xRe;

```

---

```

Bij:=GetY(i,j).xIm;
if j=i then
begin
sum:=0;
for k:=1 to LFNbus do
begin
Uk:=GetV(k).xRe;
dk:=GetV(k).xIm;
Gjk:=GetY(j,k).xRe;
Bjk:=GetY(j,k).xIm;
sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);
end;
Qj:=sum*Ui;
Jaq[row,col]:=Qj-Bij*Ui;
end
else
begin
Jaq[row,col]:=Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
end;
end;
end;
end;
//Pembentukan Matrik N
row:=0;
for i:=1 to LFNbus do
begin
if GetTyp(i) <> 1 then
begin
row:=row+1;
col:=LFNbus-1;
Ui:=GetV(i).xRe;
di:=GetV(i).xIm;
for j:=1 to LFNbus do
begin
if GetTyp(j)=3 then
begin
col:=col+1;
Uj:=GetV(j).xRe;
dj:=GetV(j).xIm;
Gij:=GetY(i,j).xRe;
Bij:=GetY(i,j).xIm;
if j=i then
begin
sum:=0;
for k:=1 to LFNbus do

```

```

begin
  Uk:=GetV(k).xRe;
  dk:=GetV(k).xIm;
  Gjk:=GetY(j,k).xRe;
  Bjk:=GetY(j,k).xIm;
  sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
end;
Pj:=sum*Uj;
Jaq[row,col]:=Pj+Gij*Ui;
end
else
begin
  Jaq[row,col]:=-Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
end;
end;
end;
end;
end;
//Pembentukan Jaqobian M
row:=LFNbus-1;
for i:=1 to LFNbus do
begin
  if GetTyp(i)=3 then
  begin
    row:=row+1;
    col:=0;
    Ui:=GetV(i).xRe;
    di:=GetV(i).xIm;
    for j:=1 to LFNbus do
    begin
      if GetTyp(j) <> 1 then
      begin
        col:=col+1;
        Uj:=GetV(j).xRe;
        dj:=GetV(j).xIm;
        Gij:=GetY(i,j).xRe;
        Bij:=GetY(i,j).xIm;
        if j=i then
        begin
          sum:=0;
          for k:=1 to LFNbus do
          begin
            Uk:=GetV(k).xRe;
            dk:=GetV(k).xIm;
            Gjk:=GetY(j,k).xRe;
            Bjk:=GetY(j,k).xIm;

```

```

        sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
    end;
    Pj:=sum*Ui;
    Jaq[row,col]:=-Pj-Gij*sqr(Ui);
end
else
begin
    Jaq[row,col]:=-Ui*Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
end;
end;
end;
end;
end;
end;
end;
end;

```

```

procedure TNRpolar.UpdateTegangan(const ald:integer;
    const dS:Arr1;const Jaq:Arr2);
var i,Ka,La:integer;
    a,b:double;
    mt1,mt2,mt3:TMatrik;
    tmp:TKomplex;
begin
    mt1:=TMatrik.Create(Jaq);
    mt2:=TMatrik.Create(ald,1);
    for i:=1 to ald do
    begin
        mt2.SetElement(i,1,dS[i]);
    end;
    mt3:=mt1.doEllGauss(mt2);
    Ka:=0;
    La:=LFNbus-1;
    tmp:=TKomplex.Create;
    for i:=1 to LFNbus do
    begin
        a:=GetV(i).xIm;
        b:=GetV(i).xRe;
        if GetTyp(i)<>1 then
        begin
            Ka:=Ka+1;
            a:=a+mt3.GetElement(Ka,1)//YF[Ka];
        end;
        if GetTyp(i)=3 then
        begin
            La:=La+1;
            b:=b+mt3.GetElement(La,1)//YE[La];
        end;
    end;
end;

```

---

```

    tmp.Assign(b,a);
    SetV(i,tmp);
end;
tmp.Free;
mt1.Free;
mt2.Free;
mt3.Free;
end;

proccedure TNRpolar.doHitung;
var i,Ngen,id:integer;
    max:double;
    dS:Arr1;
    Jaq:Arr2;
    Va:CArr1;
begin
    iterasi:=0;
    Ngen:=GetNgen;
    id:=LFNbus-1-LFNbus-Ngen-1;
    Admitansi;
    for i:=1 to LFIterasiMax do
    begin
        MismatchDaya(id,max,dS);
        if max<LFToleransi then
        begin
            break;
        end;
        Jaqobian(id,Jaq);
        UpdateTegangan(id,dS,Jaq);
        iterasi:=iterasi+1;
    end;
    Va:=LFV;
    Va:=Polar2rec(Va);
    LFV:=Va;
    AlirDaya;
    Dayareaktif;
    DayaSlack;
    ArusSaluran;
    HitungLosses;
    for i:=1 to LFNbus do
    begin
        Va[i].Free;
    end;
end;

function TNRpolar.GetIterasi:integer;

```

```
begin
  result:-iterasi,
end;

end.
```

---



```

unit TabuList;

interface

uses Komplex;

type
  TIndividu=record
    bus:integer;
    cap,fitness:double;
  end;

  TIndividuArr=array of TIndividu;

  TTabuList=class
  private
    FNlist:integer;
    FLists:TIndividuArr;
    FImprove:boolean;
    procedure SetNList(const dNList:integer);
    function GantiIndividu(const dIndi:TIndividu):TIndividu;
    function GetIndividuMax:TIndividu;
  public
    constructor Create;overload;
    constructor Create(const dNList:integer);overload;
    function isOnList(const dBus:integer;
      const dCap:double):boolean;
    procedure FillTabuList(const dBus:integer;const dCap,dFitness:double);
    procedure Clear;
    destructor Destroy;override;
    property NList:integer read FNList write SetNList;
    property Improve:boolean read FImprove;
    property IndividuMax:TIndividu read GetIndividuMax;
  end;

implementation

{ TTabuList }

constructor TTabuList.Create;
begin
  inherited Create;
end;

procedure TTabuList.Clear;
var i:integer;

```

---

```

begin
  for i:=1 to FNList do
    begin
      FLists[i].bus:=0;
      FLists[i].cap:=0;
      FLists[i].fitness:=0;
    end;
  end;

constructor TTabuList.Create(const dNList: integer);
begin
  inherited Create;
  FNList:=dNList;
  SetLength(FLists,FNList+1);
end;

destructor TTabuList.Destroy;
begin
  inherited Destroy;
end;

procedure TTabuList.FillTabuList(const dBus: integer; const dCap,
  dFitness: double);
var i,pos:integer;
    mahal:double;
begin
  if FLists[FNList].fitness=0 then
    begin
      for i:=1 to FNList do
        begin
          if FLists[i].fitness=0 then
            begin
              FLists[i].bus:=dBus;
              FLists[i].cap:=dCap;
              FLists[i].fitness:=dFitness;
              FImprove:=true;
              break;
            end;
          end;
        end;
    end;
  else
    begin
      FImprove:=true;
      for i:=1 to FNList do
        begin
          if dFitness>FLists[i].fitness then

```

---

```

begin
  FImprove:=false;
  break;
end;
end;
pos:=1;
mahal:=FLists[1].fitness;
for i:=1 to FNList do
begin
  if mahal<FLists[i].fitness then
  begin
    pos:=i;
    mahal:=FLists[i].fitness;
  end;
end;
if dFitness<FLists[pos].fitness then
begin
  FLists[pos].bus:=dbus;
  FLists[pos].cap:=dcap;
  FLists[pos].fitness:=dfitness;
end;
end;
end;

```

```

function TTabuList.GantiIndividu(const dIndi: TIndividu): TIndividu;
begin
  result.bus:=dIndi.bus;
  result.cap:=dIndi.cap;
  result.fitness:=dIndi.fitness;
end;

```

```

function TTabuList.GetIndividuMax: TIndividu;
var i:integer;
begin
  result:=GantiIndividu(FLists[1]);
  for i:=1 to FNList do
  begin
    if result.fitness<>0 then
    begin
      if result.fitness>FLists[i].fitness then
      begin
        result:=GantiIndividu(FLists[i]);
      end;
    end;
  end;
end;

```

---

```
function TTabuList.isOnList(const dBus: integer;  
  const dCap: double): boolean;  
var i: integer;  
begin  
  result:=false;  
  for i:=1 to FNList do  
    begin  
      if (dBus=FLists[i].bus) and (dCap=FLists[i].cap) then  
        begin  
          result:=true;  
          break;  
        end;  
      end;  
    end;  
end;
```

```
procedure TTabuList.SetNList(const dNList: integer);  
begin  
  FNList:=dNList;  
  SetLength(FLists,FNList+1);  
end;
```

```
end.
```

---

```

unit TabuSearchPro;

interface

uses Komplex, TabuList, SysUtils, I.FNRPolar, Hasil;

type
  TTabuSearch = class
  private
    FNbus, FNTabu, FNcap, FImprove, FIterasi: integer;
    FImproveMax, FIterasiMax: integer;
    FKp, FPbase, FCostMin, FCostCek: double;
    FCapasitor, FBiayaCap, FCapasitas: Arr1;
    FBeban, FR, FX, FLc, FTr: Arr2;
    FTabuList: TTabuList;
    procedure SetNbus(const dNbus: integer);
    procedure SetNTabu(const dNTabu: integer);
    procedure SetNcap(const dNcap: integer);
    procedure SetPbase(const dPbase: double);
    procedure SetImproveMax(const dImproveMax: integer);
    procedure SetIterasiMax(const dIterasiMax: integer);
    procedure SetKp(const dKp: double);
    procedure SetCapasitor(const dCap: Arr1);
    procedure SetBiayaCap(const dBiayaCap: Arr1);
    procedure SetLc(const dLc: Arr2);
    procedure SetTr(const dTr: Arr2);
    function GetCapasitas: Arr1;
    function doCalcFitness(const dbus: integer, const dcap: double): double;
    function pnpTahap1: integer;
    function pnpTahap2(const dbus: integer): double;
  public
    constructor Create, overload;
    constructor Create(const dNTabu: integer, const dBeban, dR, dX, dLc,
      dTr: Arr2, const dCapasitor, dBiayaCap: Arr1;
      const dKp: double); overload;
    procedure doHitung;
    destructor Destroy; override;
    property Nbus: integer read FNbus write SetNbus;
    property NTabu: integer read FNTabu write SetNTabu;
    property Ncap: integer read FNcap write SetNcap;
    property ImproveMax: integer read FImproveMax write SetImproveMax;
    property IterasiMax: integer read FIterasiMax write SetIterasiMax;
    property Pbase: double read FPbase write SetPbase;
    property Lc: Arr2 write SetLc;
    property Tr: Arr2 write SetTr;
    property Kp: double read FKp write SetKp;
  end;

```

---

```

property Capasitor:Arr1 write SetCapasitor;
property BiayaCap:Arr1 write SetBiayaCap;
property Capasitas:Arr1 read GetCapasitas;
end;

```

implementation

```

//constructor
constructor TTabuSearch.Create;
begin
  inherited Create;
  FImprove:=0;
  FIterasiMax:=0;
  FPbase:=100;
  FCostMin:=0;
  FCostcek:=0;
end;

constructor TTabuSearch.Create(const dNTabu:integer,const dBeban,dR,dX,
  dLc,dTr:Arr2;const dCapasitor,dBiayaCap:Arr1;
  const dKp:double);
var i,j:integer;
begin
  FNbus:=high(dBeban);
  FNTabu:=dNTabu;
  FTabuList:=TTabuList.Create(FNTabu);
  FNcap:=high(dCapasitor);
  SetLength(FBeban,FNbus+1,9);
  SetLength(FR,FNbus+1,FNbus+1);
  SetLength(FX,FNbus+1,FNbus+1);
  SetLength(FLc,FNbus+1,FNbus+1);
  SetLength(FTr,FNbus+1,FNbus+1);
  SetLength(FCapasitor,FNcap+1);
  SetLength(FBiayaCap,FNcap+1);
  FKp:=dKp;
  for i:=1 to FNbus do
  begin
    FBeban[i,1]:=dBeban[i,1];
    FBeban[i,2]:=dBeban[i,2];
    FBeban[i,3]:=dBeban[i,3];
    FBeban[i,4]:=dBeban[i,4];
    FBeban[i,5]:=dBeban[i,5];
    FBeban[i,6]:=dBeban[i,6];
    FBeban[i,7]:=dBeban[i,7];
    FBeban[i,8]:=dBeban[i,8];
    for j:=1 to FNbus do

```

---

```

begin
  FR[i,j]:=dR[i,j];
  FX[i,j]:=dX[i,j];
  FLc[i,j]:=dLc[i,j];
  FTr[i,j]:=dTr[i,j];
end;
end;
for i:=1 to FNcap do
begin
  FCapasitor[i]:=dCapasitor[i];
  FBiayaCap[i]:=dBiayaCap[i];
end;
FImprove:=0;
FIterasiMax:=0;
FPbase:=100;
FCostMin:=0;
FCostcek:=0;
end;

//data accessing
procedure TTabuSearch.SetNbus(const dNbus:integer);
begin
  FNbus:=dNbus;
end;

procedure TTabuSearch.SetNtabu(const dNtabu:integer);
begin
  FNtabu:=dNtabu;
  FTabuList:=TTabuList.Create(FNtabu);
end;

procedure TTabuSearch.SetLc(const dLc:Arr2);
var i,j,Ncek,Ncek2:integer;
begin
  if FNbus>0 then
  begin
    Ncek:=high(dLc);
    if Ncek<>FNbus then
    begin
      raise Exception.Create('Dimensi matrik tidak sama!');
    end;
    Ncek2:=high(dLc[0]);
    if Ncek<>Ncek2 then
    begin
      raise Exception.Create('Bukan matrik bujur sangkar!');
    end;
  end;
end;

```

---

```

end
else
begin
  FNbus:=high(dLc);
  Ncek2:=high(dLc[0]);
  if FNbus<>Ncek2 then
  begin
    raise Exception.Create('Bukan matrik bujur sangkar!');
  end;
end;
SetLength(FLc,FNbus+1,FNbus+1);
for i:=1 to FNbus do
begin
  for j:=1 to FNbus do
  begin
    FLc[i,j]:=dLc[i,j];
  end;
end;
end;
end;

```

```

procedure TTabuSearch.SciTr(const dTr:Arr2);
var i,j,Ncek,Ncek2:integer;
begin
  if FNbus>0 then
  begin
    Ncek:=high(dTr);
    if Ncek<>FNbus then
    begin
      raise Exception.Create('Dimensi matrik tidak sama!');
    end;
    Ncek2:=high(dTr[0]);
    if Ncek<>Ncek2 then
    begin
      raise Exception.Create('Bukan matrik bujur sangkar!');
    end;
  end
  else
  begin
    FNbus:=high(dTr);
    Ncek2:=high(dTr[0]);
    if FNbus<>Ncek2 then
    begin
      raise Exception.Create('Bukan matrik bujur sangkar!');
    end;
  end;
  SetLength(FTr,FNbus+1,FNbus+1);

```

---



```

for i:=1 to FNbus do
begin
  for j:=1 to FNbus do
  begin
    FTr[i,j]:=dTr[i,j];
  end;
end;
end;

```

```

//data processing
function TTabuSearch.doCalcFitness(const dbus:integer;
  const dcap:double):double;
var i,j,iterasi:integer;
  costCap,sgr,sgx,slr,slx,slor,slorx:double;
  AlirR,AlirX,dBeban:Arr2;

```

```

begin
  Setf.length(dBeban,FNbus+1,9);
  for i:=1 to FNbus do
  begin
    for j:=1 to 8 do
    begin
      dBeban[i,j]:=FBeban[i,j];
    end;
  end;
  result:=0;
  costCap:=0;
  if dBeban[dbus,4]=0 then
  begin
    dBeban[dbus,4]:=-dCap;

```

```

NewtonRaphson(FNbus,FR,FX,FLc,FTr,FPbase,iterasi,dBeban,AlirR,AlirX,sgr,
  sgx,slr,slx,slor,slorx);
  for i:=1 to FNbus do
  begin
    if (dBeban[i,4]<>0) and (i<>1) then
    begin
      for j:=1 to FNcap do
      begin
        if dBeban[i,4]=FCapasitor[j] then
        begin
          costCap:=costCap+FBiayaCap[j]*FCapasitor[j];
          break;
        end;
      end;
    end;
  end;
end;

```

---

```

    if iterasi<15 then
    begin
        result:=-slor*Kp+CostCap;
    end;
end;
end;

function TTabuSearch.pnpTahap1:integer;
var i:integer;
    cek,dcap:double;
    same:boolean;
    indiMax:TIndividu;
begin
    dcap:=FCapasitor[1];
    for i:=FNbus downto 2 do
    begin
        same:=FTabuList.isOnList(i,dcap);
        if same=false then
        begin
            cek:=doCalcFitness(i,dcap);
            if cek<>0 then
            begin
                FTabuList.FillTabuList(i,dcap,cek);
                FIterasi:=FIterasi+1;
                frmHasil.pblIterasi.StepBy(1);
                if FTabuList.Improve=true then
                begin
                    FImprove:=0;
                end
                else
                begin
                    FImprove:=FImprove+1;
                end;
            end;
        end;
    end;
    if FImprove>=FImproveMax then
    begin
        break;
    end;
    if FIterasi=FIterasiMax then
    begin
        break;
    end;
end;
indiMax:=FTabuList.IndividuMax;
result:=indiMax.bus;

```

```

FCostcek:=IndiMax.fitness;
if FCostMin=0 then
begin
  FCostMin:=FCostCek;
end;
if FCostcek>FCostMin then
begin
  FIterasi:=FIterasiMax;
end
else if FCostcek<FCostMin then
begin
  FCostMin:=FCostcek;
end;
end;

```

```

function TTabuSearch.pnpTahap2(const dbus:integer):double;
var i:integer;
    dcap,cek:double;
    dIndi:TIndividu;
    same:boolean;
begin
  for i:=1 to FNcap do
  begin
    dcap:=FCapasitor[i];
    same:=FTabuList.isOnList(dbus,dcap);
    if same=false then
    begin
      cek:=doCalcFitness(dbus,dcap);
      if cek<>0 then
      begin
        FTabuList.FillTabuList(dbus,dcap,cek);
        FIterasi:=FIterasi+1;
        frmHasil.pbIterasi.StepBy(1);
        if FTabuList.Improve=true then
        begin
          FImprove:=0;
        end
        else
        begin
          FImprove:=FImprove+1;
        end;
      end;
    end;
  end;
  if FImprove>=FImproveMax then
  begin
    break;
  end;

```

---

```

end;
if FIterasi>=FIterasiMax then
begin
break;
end;
end;
dIndi:=FTabuList.IndividuMax;
result:=dIndi.cap;
FCostcek:=dIndi.fitness;
if FCostMin=0 then
begin
FCostMin:=FCostCek;
end;
if FCostcek>FCostMin then
begin
FIterasi:=FIterasiMax;
end
else if FCostcek<FCostMin then
begin
FCostMin:=FCostcek;
end;
end;
end;

```

```

procedure TTabuSearch.doHitung;
var bus,i:integer;
cap:double;
begin
repeat
bus:=pnpTahap1;
cap:=pnpTahap2(bus);
if FIterasi<FIterasiMax then
begin
FBeban[bus,4]:=cap;
FTabuList.Clear;
end;
until (FImprove>=FImproveMax) or (FIterasi>=FIterasiMax);
SetLength(FCapasitas, FNbus+1);
for i:=1 to FNbus do
begin
FCapasitas[i]:=FBeban[i,4];
end;
end;
//data output
function TTabuSearch.GetCapasitas:Arr1;
var i:integer;
begin

```

```

SetLength(result,high(FCapasitas)+1);
for i:=1 to high(FCapasitas) do
begin
  result[i]:=FCapasitas[i];
end;
end;

//destructor
destructor TTabuSearch.Destroy;
begin
  try
    FTabuList.Free;
  finally
    inherited Destroy;
  end;
end;

procedure TTabuSearch.SetBiayaCap(const dBiayaCap: Arr1);
var i:integer;
begin
  SetLength(FBiayaCap,FNcap+1);
  for i:=1 to FNcap do
  begin
    FBiayaCap[i]:=dBiayaCap[i];
  end;
end;

procedure TTabuSearch.SetCapasitor(const dCap: Arr1);
var i:integer;
begin
  SetLength(FCapasitor,FNcap+1);
  for i:=1 to FNcap do
  begin
    FCapasitor[i]:=dCap[i];
  end;
end;

procedure TTabuSearch.SetKp(const dKp: double);
begin
  FKp:=dKp;
end;

procedure TTabuSearch.SetNcap(const dNcap: integer);
begin
  FNcap:=dNcap;
end;

```

---

```
procedure TTabuSearch.SetImproveMax(const dImproveMax: integer);  
begin  
  FImproveMax:=dImproveMax;  
end;
```

```
procedure TTabuSearch.SetIterasiMax(const dIterasiMax: integer);  
begin  
  FIterasiMax:=dIterasiMax;  
end;
```

```
procedure TTabuSearch.SetPbase(const dPbase: double);  
begin  
  FPbase:=dPbase;  
end;
```

```
end.
```

---

```

unit Sensitivitas;

interface

uses Komplek,Matrik;

type
  TSentivitas=class
  private
    Nbus:integer;
    PLoss:double;
    V:CArr1;
    Cap,dPLdP,dPLdQ:Arr1;
    Typ:iArr1;
    Z,Y:CArr2;
    Lc,Tr:Arr2;
    function GetNbus:integer;
    function GetV:CArr1;
    function GetCap:Arr1;
    function GetTyp:iArr1;
    function GetZ:CArr2;
    function GetLc:Arr2;
    function GetTr:Arr2;
    procedure SetNbus(const dNbus:integer);
    procedure SetV(const dV:CArr1);
    procedure SetCap(const dCap:Arr1);
    procedure SetTyp(const dTyp:iArr1);
    procedure SetZ(const dZ:CArr2);
    procedure SetLc(const dLc:Arr2);
    procedure SetTr(const dTr:Arr2);
    procedure UbahTegPolar;
    procedure Admitansi;
    procedure Jaqobian(const id:integer;
      var Jaq:Arr2);
    procedure HitungPloss;
    procedure HitungdP(const id:integer;
      var dP:Arr1);
    function GetPloss:double;
    procedure EllGauss(const Id:integer,const Jaq:Arr2;
      const dS:Arr1;
      var YE:Arr1);
    procedure findSensitiv(const id:integer,const YE:Arr1);
    function GetdPLdP:Arr1;
    function GetdPLdQ:Arr1;
  public
    constructor Create;overload;

```

```

constructor Create(const dNbus:integer);overload;
procedure doHitung;
destructor Destroy;override;
property seNbus:integer read GetNbus write SetNbus;
property seV:CArr1 read GetV write SetV;
property seCap:Arr1 read GetCap write SetCap;
property seTyp:iArr1 read GetTyp write SetTyp;
property seZ:CArr2 read GetZ write SetZ;
property seLc:Arr2 read GetLc write SetLc;
property seTr:Arr2 read GetTr write SetTr;
property sePloss:double read GetPloss;
property sedPLdP:Arr1 read GetdPLdP;
property sedPI.dQ:Arr1 read GetdPI.dQ;
end;

```

implementation

```

//constructor
constructor TSentivitas.Create;
begin
  inherited Create;
  Nbus:=0;
end;

constructor TSentivitas.Create(const dNbus:integer);
begin
  inherited Create;
  Nbus:=dNbus;
end;

//data accessing
function TSentivitas.GetNbus:integer;
begin
  result:=Nbus;
end;

function TSentivitas.GetV:CArr1;
var i:integer;
begin
  SetLength(result,Nbus+1);
  for i:=1 to Nbus do
  begin
    result[i]:=TKomplex.Create(V[i]);
  end;
end;
end;

```

---



```
function TSentivitas.GetCap:Arr1;  
var i:integer;  
begin  
  SetLength(result,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      result[i]:=Cap[i];  
    end;  
end;
```

```
function TSentivitas.GetTyp:iArr1;  
var i:integer;  
begin  
  SetLength(result,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      result[i]:=Typ[i];  
    end;  
end;
```

```
function TSentivitas.GetZ:CArr2;  
var i,j:integer;  
begin  
  SetLength(result,Nbus+1,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      for j:=1 to Nbus do  
        begin  
          result[i,j]:=TKomplex.Create;  
          result[i,j].Assign(Z[i,j]);  
        end;  
      end;  
    end;  
end;
```

```
function TSentivitas.GetLc:Arr2;  
var i,j:integer;  
begin  
  SetLength(result,Nbus+1,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      for j:=1 to Nbus do  
        begin  
          result[i,j]:=Lc[i,j];  
        end;  
      end;  
    end;  
end;
```

---

```

function TSentivitas.GetTr:Arr2;
var i,j:integer;
begin
  SetLength(result,Nbus+1,Nbus+1);
  for i:=1 to Nbus do
    begin
      for j:=1 to Nbus do
        begin
          result[i,j]:=Tr[i,j];
        end;
      end;
    end;
end;

procedure TSentivitas.SetNbus(const dNbus:integer);
begin
  Nbus:=dNbus;
end;

procedure TSentivitas.SetV(const dV:CArr1);
var i:integer;
begin
  SetLength(V,Nbus+1);
  for i:=1 to Nbus do
    begin
      V[i]:=TKomplex.Create(dV[i]);
    end;
end;

procedure TSentivitas.SetCap(const dCap:Arr1);
var i:integer;
begin
  SetLength(Cap,Nbus+1);
  for i:=1 to Nbus do
    begin
      Cap[i]:=dCap[i];
    end;
end;

procedure TSentivitas.SetTyp(const dTyp:iArr1);
var i:integer;
begin
  SetLength(Typ,Nbus+1);
  for i:=1 to Nbus do
    begin
      Typ[i]:=dTyp[i];
    end;
end;

```

---

```
end;  
end;
```

```
procedure TSentivitas.SetZ(const dZ:CArr2);  
var i,j:integer;  
begin  
  SetLength(Z,Nbus+1,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      for j:=1 to Nbus do  
        begin  
          Z[i,j]:=TKomplex.Create(dZ[i,j]);  
        end;  
      end;  
    end;  
end;
```

```
procedure TSentivitas.SetLc(const dLc:Arr2);  
var i,j:integer;  
begin  
  SetLength(Lc,Nbus+1,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      for j:=1 to Nbus do  
        begin  
          Lc[i,j]:=dLc[i,j];  
        end;  
      end;  
    end;  
end;
```

```
procedure TSentivitas.SetTr(const dTr:Arr2);  
var i,j:integer;  
begin  
  SetLength(Tr,Nbus+1,Nbus+1);  
  for i:=1 to Nbus do  
    begin  
      for j:=1 to Nbus do  
        begin  
          Tr[i,j]:=dTr[i,j];  
        end;  
      end;  
    end;  
end;
```

```
//data processing  
procedure TSentivitas.UbahTegPolar;  
var i:integer;  
    aRe,alm:double;
```

```

begin
  for i:=1 to Nbus do
  begin
    aRe:=V[i].GetAbs;
    aIm:=V[i].GetAngleRad;
    V[i].Assign(aRe,aIm);
  end;
end;

procedure TSentivitas.Admitansi;
var Ca,Za:CArr2;
    Lca:Arr2;
    temp:TKomplex;
    SumA,SumB:double;
    i,j,k:integer;
begin
  SetLength(Ca,Nbus+1,Nbus+1);
  SetLength(Y,Nbus+1,Nbus+1);
  SetLength(Za,Nbus+1,Nbus+1);
  SetLength(Lca,Nbus+1,Nbus+1);
  temp:=TKomplex.Create;
  temp.Assign(1,0);
  for i:=1 to Nbus do
  begin
    for j:=1 to Nbus do
    begin
      Za[i,j]:=TKomplex.Create(Z[i,j]);
      Lca[i,j]:=Lc[i,j];
    end;
  end;
  for i:=1 to Nbus do
  begin
    for j:=1 to Nbus do
    begin
      Ca[i,j]:=TKomplex.Create;
      Y[i,j]:=TKomplex.Create;
      if Za[i,j].xIm<0 then
      begin
        Za[j,i].Assign(Za[i,j]);
        Lca[j,i]:=Lca[i,j];
      end;
    end;
  end;
  for i:=1 to Nbus do
  begin
    for j:=1 to Nbus do

```

```

begin
  Ca[i,j].Assign(0,0);
  if Za[i,j].xIm <> 0 then
    begin
      Ca[i,j]:=temp.doBagi(Za[i,j]);
    end;
  end;
end;
for i:=1 to Nbus do
begin
  for j:=1 to Nbus do
  begin
    if i=j then
    begin
      SumA:=0;
      SumB:=0;
      for k:=1 to Nbus do
      begin
        if i <> k then
        begin
          SumA:=SumA+Ca[i,k].xRe;
          SumB:=SumB+Ca[i,k].xIm+Lca[i,k];
        end;
      end;
      Y[i,j].Assign(sumA,sumB);
    end
    else
    begin
      Y[i,j]:=Ca[i,j].doNegative;
    end;
  end;
end;
for i:=1 to Nbus do
begin
  for j:=1 to Nbus do
  begin
    if Tr[i,j] <> 0 then
    begin
      Tr[i,j]:=1/Tr[i,j];
      sumA:=Y[i,i].xRe-Ca[i,j].xRe;
      sumB:=Y[i,i].xIm-Ca[i,j].xIm;
      Y[i,i].Assign(sumA,sumB);
      sumA:=Y[i,i].xRe+Ca[i,j].xRe*sqr(Tr[i,j]);
      sumB:=Y[i,i].xIm+Ca[i,j].xIm*sqr(Tr[i,j]);
      Y[i,i].Assign(sumA,sumB);
      sumA:='Tr[i,j]*Y[i,j].xRe;

```

```

    sumB:=Tr[i,j]*Y[i,j].xIm;
    Y[i,j].Assign(sumA,sumB);
    Y[j,i].Assign(Y[i,j]);
  end;
end;
end;
for i:=1 to Nbus do
begin
  if Cap[i]<>0 then
  begin
    sumB:=Y[i,i].xIm+Cap[i];
    Y[i,i].Assign(Y[i,i].xRe,sumB);
  end;
end;
for i:=1 to Nbus do
begin
  for j:=1 to Nbus do
  begin
    Ca[i,j].Free;
    Za[i,i].Free;
  end;
end;
temp.Free;
end;

procedure TSentivitas.Jaqobian(const id:integer;
  var Jaq:Arr2);
var i,j,k,row,col:integer;
  Gij,Bij,Ui,di,Uj,dj,Uk,dk,Gjk,Bjk,sum:double;
  Pj,Qj:double;
begin
  SetLength(Jaq,id+1,id+1);
  //Pembentukan Jaqobian H
  row:=0;
  for i:=1 to Nbus do
  begin
    row:=row+1;
    col:=0;
    Ui:=V[i].xRe;
    di:=V[i].xIm;
    for j:=1 to Nbus do
    begin
      col:=col+1;
      Uj:=V[j].xRe;
      dj:=V[j].xIm;
      Gij:=Y[i,j].xRe;

```

```

Bij:=Y[i,j].xIm;
if j=i then
begin
sum:=0;
for k:=1 to Nbus do
begin
Uk:=V[k].xRe;
dk:=V[k].xIm;
Gjk:=Y[j,k].xRe;
Bjk:=Y[j,k].xIm;
sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);
end;
Qj:=-sum*Uj;
Jaq[row,col]:=-Qj-Bij*sqr(Ui);
end
else
begin
Jaq[row,col]:=-Ui*Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
end;
end;
end;
end;
//Pembentukan Jaqobian L
row:=Nbus;
for i:=1 to Nbus do
begin
row:=row+1;
col:=Nbus;
Ui:=V[i].xRe;
di:=V[i].xIm;
for j:=1 to Nbus do
begin
col:=col+1;
Uj:=V[j].xRe;
dj:=V[j].xIm;
Gij:=Y[i,j].xRe;
Bij:=Y[i,j].xIm;
if j=i then
begin
sum:=0;
for k:=1 to Nbus do
begin
Uk:=V[k].xRe;
dk:=V[k].xIm;
Gjk:=Y[j,k].xRe;
Bjk:=Y[j,k].xIm;
sum:=sum+((Gjk*sin(dj-dk)-Bjk*cos(dj-dk))*Uk);

```

```

    end;
    Qj:=sum*Ui;
    Jaq[row,col]:=Qj-Bij*Ui;
end
else
begin
    Jaq[row,col]:=Uj*(Gij*sin(di-dj)-Bij*cos(di-dj));
end;
end;
end;
//Pembentukan Matrik N
row:=0;
for i:=1 to Nbus do
begin
    row:=row+1;
    col:=Nbus;
    Ui:=V[i].xRe;
    di:=V[i].xIm;
    for j:=1 to Nbus do
    begin
        col:=col+1;
        Uj:=V[j].xRe;
        dj:=V[j].xIm;
        Gij:=Y[i,j].xRe;
        Bij:=Y[i,j].xIm;
        if j=i then
        begin
            sum:=0;
            for k:=1 to Nbus do
            begin
                Uk:=V[k].xRe;
                dk:=V[k].xIm;
                Gjk:=Y[j,k].xRe;
                Bjk:=Y[j,k].xIm;
                sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
            end;
            Pj:=-sum*Uj;
            Jaq[row,col]:=Pj+Gij*Ui;
        end
        else
        begin
            Jaq[row,col]:=Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
        end;
    end;
end;
end;
//Pembentukan Jaqobian M

```



```

row:=Nbus;
for i:-1 to Nbus do
begin
row:=row+1;
col:=0;
Ui:=V[i].xRe;
di:=V[i].xIm;
for j:-1 to Nbus do
begin
col:=col+1;
Uj:=V[j].xRe;
dj:=V[j].xIm;
Gij:=Y[i,j].xRe;
Bij:=Y[i,j].xIm;
if j=i then
begin
sum:=0;
for k:=1 to Nbus do
begin
Uk:=V[k].xRe;
dk:=V[k].xIm;
Gjk:=Y[j,k].xRe;
Bjk:=Y[j,k].xIm;
sum:=sum+((Gjk*cos(dj-dk)+Bjk*sin(dj-dk))*Uk);
end;
Pj:=sum*Uj;
Jaq[row,col]:=Pj-Gij*sqr(Ui);
end
else
begin
Jaq[row,col]:=-Ui*Uj*(Gij*cos(di-dj)+Bij*sin(di-dj));
end;
end;
end;
end;
end;

```

```

procedure TSentivitas.HitungPloss;
var i,j:integer;
    Ui,Uj,di,dj,gij:double;
begin
Ploss:=0;
for i:-1 to Nbus do
begin
Ui:=V[i].xRe;
di:=V[i].xIm;
for j:=1 to Nbus do

```

```

begin
  if (Y[i,j].xIm <> 0) and (V[i].xRe > V[j].xRe) then
  begin
    Uj := V[j].xRe;
    dj := V[j].xIm;
    gij := -Y[i,j].xRe;
    Ploss := Ploss + (gij*(sqr(Ui)+sqr(Uj)-2*Ui*Uj*cos(di-dj)));
  end;
end;
end;
end;

```

```

function TSentivitas.GetPloss:double;
begin
  result := Ploss;
end;

```

```

procedure TSentivitas.HitungdP(const id:integer;
  var dP:Arr1);
var i,j,k,sV,s0:integer;
  Uj,dj,Uk,dk,gjk:double;
  dPdV,dPd0:Arr1;
begin
  SetLength(dPdV,Nbus+1);
  SetLength(dPd0,Nbus+1);
  SetLength(dP,id+1);
  for i:=1 to Nbus do
  begin
    dPd0[i]:=0;
    dPdV[i]:=0;
    for j:=1 to Nbus do
    begin
      Uj:=V[j].xRe;
      dj:=V[j].xIm;
      for k:=1 to Nbus do
      begin
        if k <> j then
        begin
          if (Y[j,k].xIm <> 0) and (V[j].xRe > V[k].xRe) then
          begin
            if (j=i) or (k=i) then
            begin
              Uk:=V[k].xRe;
              dk:=V[k].xIm;
              gjk := -Y[j,k].xRe;
              if j=i then

```



```

end;
mYE:=mJaq.doEllGauss(mdS);
SetLength(YE,ld+1);
for i:=1 to ld do
begin
  YE[i]:=mYE.GetElement(i,1);
end;
mJaq.Free;
mdS.Free;
mYE.Free;
end;

```

```

procedure TSentivitas.findSensitiv(const id:integer,const YE:Arr1);
var i,sP,sQ:integer;
begin
  SetLength(dPLdP,Nbus+1);
  SetLength(dPLdQ,Nbus+1);
  sP:=0;
  sQ:=Nbus;
  for i:=1 to Nbus do
  begin
    sP:=sP+1;
    dPLdP[i]:=YE[sP];
    sQ:=sQ+1;
    dPLdQ[i]:=YE[sQ];
  end;
end;

```

```

procedure TSentivitas.doHitung;
var id:integer;
    dP,YE:Arr1;
    Jaq:Arr2;
begin
  id:=2*Nbus;
  UbahTegPolar;
  Admitansi;
  Jaqobian(id,Jaq);
  HitungPloss;
  HitungdP(id,dP);
  FillGauss(id,Jaq,dP,YE);
  FindSensitiv(id,YE);
end;

```

```

function TSentivitas.GetdPLdP:Arr1;
var i:integer;
begin

```

```

SetLength(result,Nbus+1);
for i:=1 to Nbus do
begin
  result[i]:=dPLdP[i];
end;
end;

function TSentivitas.GetdPLdQ:Arr1;
var i:integer;
begin
  SetLength(result,Nbus+1);
  for i:=1 to Nbus do
  begin
    result[i]:=dPLdQ[i];
  end;
end;

//destructor
destructor TSentivitas.Destroy;
var i,j:integer;
begin
  try
    for i:=1 to Nbus do
    begin
      V[i].Free;
      for j:=1 to Nbus do
      begin
        Z[i,j].Free;
        Y[i,j].Free;
      end;
    end;
  finally
    inherited Destroy;
  end;
end;
end.

```