

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



**SKRIPSI**



***OPTIMAL POWER FLOW*  
MENGGUNAKAN METODE *TABU SEARCH ALGORITHM*  
PADA SUB-SISTEM 150 kV PAITON-BALI**

*Disusun oleh;*

**Nama : DESTA DWI MULYANTO**

**Nim : 00.12.025**

**September 2005**

---

## LEMBAR PERSETUJUAN

### **OPTIMAL POWER FLOW MENGUNAKAN METODE TABU SEARCH ALGORITHM PADA SUB-SISTEM 150 kV PAITON-BALI**

#### SKRIPSI

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

Disusun Oleh :

**DESTA DWI MULYANTO**  
**NIM. 00.12.025**



Mengetahui,  
Ketua Jurusan Teknik Elektro

**Ir. F. Yudi Limpraptono, MT**  
**NIP.Y.193 950 0274**

Menyetujui,  
Dosen Pembimbing

**Ir. Yusuf Ismail Nakhoda, MT**  
**NIP.P.101 880 0189**

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

## PERSEMBAHANKU

Assalamualaikum Wr Wb

Puji syukur atas rahmat dan hidayah dari Allah SWT yang mana hanya karena ridhonya lah skripsi ini dapat selesai dengan sempurna.

Terima kasih untuk papaku selaku penyandang dana dalam pembuatan skripsi dan kuliahku selama ini, untuk mamaku terima kasih atas dukungan dan juga bimbingan yang diberikan serta seluruh keluargaku yang selalu memberikan semangat kepadaku. Buat bapak Yusuf Ismail Nakhoda terima kasih atas bimbingan dan ilmu yang diberikan selama ini. Untuk Viviku "terima kasih bintangnya dan juga doa yang selalu memberikan semangat padaku, aku senang kmu mau jadi bagian dari hidupku". Untuk Mia temen baru ku" jaga vivi ya?, my friend Erny "makasih telah mau berteman denganku selama ini." Dan akhirnya untuk temen-teman seperjuanganku R. Widya TV Bhekotak, Wiky Godek, Agung Mujaer "Makasih semangat persahabatannya dan ayo kita harus sukses", for Gus Asyik & uqi terima kasih atas masukan-masukannya mengenai agama", Erma & Aryo "oyo ngamu'an harus belajar sabar jangan lupa perbanyak anak ya he..3x " untuk mbak Anisku yang cuantek dewe "terima kasih atas kesabarannya dan nasehatnya", terima kasih juga untuk para instruktur Laboratorium lingkungan yang kenal aku, untuk personil OrangeZ Band, Anton, Agung Mujaer, Rohandi, Ari Rt, "terima kasih telah membuat lagu-laguku lebih hidup, ayo kapan buat lagu & ngeband lagi?"

Untuk para anggota PK-Speed "harus tetep kompak ojo dewe-dewe".  
Untuk temen-ku yang udah lulus Reza, yuni, irfan, pandu, Lia dkk  
"terima kasih atas kenang-kenangannya di bangku kuliah" untuk temen-  
temen gank elektro energi listrik S-1 2000, ewin, sonimen, beni, joko  
solo, yuono joko, tomi, ruli, andi cepu, andi mojokerto, deni, mojako,  
samsul, samsi, budi, sastro, tini, tante ani, neni, aan, jeck, abas,  
lukman hakim, lukman arema, joe, khoirul, ali, dan semuanya maaf jika  
ada yang gak di sebutin pokoknya untuk seluruh anak-anak jurusan  
elektro energi listrik S-1 angkatan 2000 kelas 1,2 & 3 terima kasih  
atas persahabatannya selama ini.

Akhirnya terima kasih terhadap seluruh pihak sponsor atas  
dukungan dan partisipasinya. Sekian dan dengan ucapan Alhamdulillah  
hirobbil alamin

Wassalamualaikum Wr Wb



## ABSTRAKSI

### OPTIMAL POWER FLOW MENGGUNAKAN METODE TABU SEARCH PADA SUB-SISTEM 150 kV PAITON-BALI

( Desta Dwi Mulyanto, Nim 00.12.025, Teknik Elektro S-1/ Teknik Energi Listrik)  
( Dosen Pembimbing: Ir. Yusuf Ismail Nakhoda, MT.)

Kata Kunci: *Optimal Power Flow, Tabu Search Algorithm.*

Kebutuhan daya listrik yang selalu berubah dari waktu ke waktu, menimbulkan suatu persoalan yaitu bagaimana mengoperasikan suatu sistem tenaga listrik yang selalu dapat memenuhi permintaan daya pada setiap saat, dengan kualitas yang baik dan harga yang murah. Oleh karena itu pada suatu operasi pada beban tertentu, perhitungan ekonomis harus tetap merupakan suatu prioritas atau nilai yang harus diperhitungkan di samping hal-hal lain, sehingga nantinya diperlukan suatu rencana operasi yang optimum dengan tetap memenuhi beberapa persyaratan pengoperasian sistem tenaga listrik yaitu antara lain: daya yang dibangkitkan cukup untuk memasok beban dan rugi-rugi daya pada saluran transmisi serta tegangan bus sesuai dengan ratingnya.

Skripsi ini membahas masalah *Optimal Power Flow* dengan menggunakan metode *Tabu Search Algorithm*. Hasil dari analisa tersebut nantinya dapat digunakan sebagai salah satu acuan dalam operasi pembangkitan dan penyaluran daya yang ekonomis dan optimal, terutama mengenai biaya pembangkitan. *Input* dari program ini adalah biaya bahan bakar dan hasil perhitungan aliran daya, sedangkan hasil akhir dari program ini yaitu hasil perhitungan aliran daya, tegangan dan sudut fasa tiap-tiap bus serta biaya pembangkitan yang paling optimal.

Analisa dilakukan dengan bantuan program komputer dengan menggunakan bahasa pemrograman Delphi versi 7.0 dan telah berhasil dicoba pada sub-sistem 150 kV Paiton-Bali yang terdiri dari 25 bus dan 34 saluran transmisi, dimana telah berhasil dilakukan penghematan biaya pembangkitan sebesar Rp 799.208,90 atau terjadi optimasi sebesar 18,5%.

## KATA PENGANTAR

Dengan memanjatkan puji syukur kehadiran Allah SWT, yang telah memberikan rahmat dan hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan skripsi ini yang merupakan salah satu persyaratan dalam menyelesaikan studi program strata satu(S-1) jurusan Teknik Elektro/ Program Studi teknik Energi Listrik, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang.

Sebelum dan selama penulisan skripsi, penulis telah banyak mendapatkan bantuan dan bimbingan dari berbagai pihak. Untuk itu pada kesempatan ini penulis menyampaikan terima kasih yang sebesar-besarnya kepada:

1. Bapak Ir. F Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro
2. Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku dosen pembimbing dalam penulisan skripsi ini.
4. Bapak Tri Agus Cahyono, selaku Manager Bidang Enjiniring, yang telah memberikan kemudahan untuk mendapatkan data di PT. PLN(Persero)P3B Region Jawa Timur dan Bali.
5. Mas Ugro Wiseno, yang telah membantu penulis dalam pengerjaan program dalam penulisan skripsi ini
6. Semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah membantu dalam penulisan skripsi ini.

Semoga segala bantuan dan bimbingan yang diberikan mendapat balasan dari Allah SWT. Dan harapan penulis semoga skripsi ini memberikan manfaat bagi pembaca semua dan semoga Allah SWT selalu membimbingkan memberikan rahmat dan hidayah-nya kepada kita semua, Amin

Malang, Oktober 2005

Penulis

## DAFTAR ISI

|                                                                                          |      |
|------------------------------------------------------------------------------------------|------|
| HALAMAN JUDUL.....                                                                       | i    |
| LEMBAR PERSETUJUAN.....                                                                  | ii   |
| ABSTRAKSI.....                                                                           | iii  |
| KATA PENGANTAR.....                                                                      | iv   |
| DAFTAR ISI.....                                                                          | iv   |
| DAFTAR GAMBAR.....                                                                       | ix   |
| DAFTAR TABEL.....                                                                        | xi   |
| DAFTAR GRAFIK.....                                                                       | xiii |
| BAB I PENDAHULUAN.....                                                                   | 1    |
| 1.1. Latar Belakang.....                                                                 | 1    |
| 1.2. Permasalahan.....                                                                   | 1    |
| 1.3. Tujuan.....                                                                         | 2    |
| 1.4. Batasan Masalah.....                                                                | 2    |
| 1.5. Metodologi Penelitian.....                                                          | 3    |
| 1.6. Kontribusi Penelitian.....                                                          | 3    |
| BAB II KARAKTERISTIK UNIT PEMBANGKIT, OPERASI EKONOMIS DAN<br>SALURAN TRANSMISI.....     | 4    |
| 2.1. Sistem Operasi Pada Sistem tenaga Listrik.....                                      | 4    |
| 2.2. Karakteristik Unit Pembangkit.....                                                  | 7    |
| 2.2.1. Karakteristik <i>Input-output</i> .....                                           | 7    |
| 2.2.2. Karakteristik <i>Heat-Rate</i> .....                                              | 9    |
| 2.2.3. Karakteristik <i>Incremental Heat Rate</i> dan <i>Incremental Fuel Coat</i> ..... | 10   |
| 2.3. Economic Dispatch.....                                                              | 11   |
| 2.3.1. Economic Dispatch Dengan Mengabaikan Rugi-rugi Transmisi.....                     | 14   |

|                                                                                                          |    |
|----------------------------------------------------------------------------------------------------------|----|
| 2.3.2. Economic Dispatch Dengan Memperhitungkan Rugi-rugi Transmisi.....                                 | 14 |
| 2.4. Fungsi Biaya bahan bakar.....                                                                       | 15 |
| 2.5. Sistem Jaringan Transmisi.....                                                                      | 16 |
| 2.5.1. Saluran Transmisi Pendek.....                                                                     | 17 |
| 2.5.2. Saluran Transmisi Menengah.....                                                                   | 18 |
| 2.5.3. Saluran Transmisi Panjang.....                                                                    | 19 |
| 2.6. Sistem Per-Unit.....                                                                                | 20 |
| BAB III <i>OPTIMAL POWER FLOW</i> Menggunakan Metode <i>Tabu Search Algorithm</i> ..                     | 22 |
| 3.1. Konsep Aliran Daya.....                                                                             | 22 |
| 3.1.1 Metode Newton-Rapshon .....                                                                        | 23 |
| 3.2. <i>Optimal Power Flow</i> .....                                                                     | 25 |
| 3.3. Metode <i>Tabu Search Algorithm</i> .....                                                           | 26 |
| 3.3.1. Konsep Dasar.....                                                                                 | 26 |
| 3.3.2. Prosedur <i>Tabu search</i> .....                                                                 | 27 |
| 3.3.3. Kriteria Penghentian .....                                                                        | 30 |
| 3.3.4. Solusi <i>Tabu Search Algorithm</i> pada <i>Optimal Power Flow</i> .....                          | 31 |
| 3.3.5. Proses Formulasi <i>Tabu Search</i> .....                                                         | 33 |
| BAB IV HASIL DAN ANALISIS HASIL.....                                                                     | 34 |
| 4.1. Program Komputer <i>Optimal Power Flow</i> menggunakan Metode <i>Tabu Search</i><br>Algorithm ..... | 34 |
| 4.1.1. Algoritma Program .....                                                                           | 34 |
| 4.1.2. Flowchart Program.....                                                                            | 37 |
| 4.2. Validasi Program menggunakan data Sistem IEEE 30-Bus .....                                          | 39 |
| 4.2.1 Data Sistem IEEE 30-Bus.....                                                                       | 39 |

|                                                                                                                                                           |    |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------|----|
| 4.2.1. Hasil Perhitungan Validasi Sistem IEEE 30-Bus Menggunakan Metode <i>Tabu Search Algorithm</i> .....                                                | 44 |
| 4.3. Data SuB-Sistem 150 kV Paiton-Bali .....                                                                                                             | 44 |
| 4.3.1. Data Pembangkitan dan Pembebanan Pada Sub-Sistem 150 kV Paiton-Bali ....                                                                           | 48 |
| 4.3.2. Data Saluran Transmisi Pada Sub-Sistem 150 kV Paiton-Bali.....                                                                                     | 49 |
| 4.3.3. Hasil Perhitungan Biaya Operasi Pada Pembangkit Termal Sub-Sistem 150 kV Paiton-Bali Menggunakan Metode <i>Tabu search Algorithm</i> .....         | 50 |
| 4.3.4. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Dengan Program Optimal Power Flow Menggunakan Metode <i>Tabu Search Algorithm</i> . .. | 59 |
| 4.3.4.1. Tingkat Optimum Biaya Pembangkitan.....                                                                                                          | 60 |
| 4.3.4.2. Perbandingan Tingkat Tegangan Pada Tiap Bus .....                                                                                                | 61 |
| 4.3.4.3. Perbandingan Tingkat Rugi Daya Pada Saluran .....                                                                                                | 62 |
| BAB V KESIMPULAN DAN SARAN.....                                                                                                                           | 63 |
| 5.1. Kesimpulan .....                                                                                                                                     | 63 |
| 5.2. Saran.....                                                                                                                                           | 64 |

## DAFTAR PUSTAKA

## LAMPIRAN

## DAFTAR GAMBAR

|              |                                                                                                             |    |
|--------------|-------------------------------------------------------------------------------------------------------------|----|
| Gambar 2.1.  | Unit Boiler-Turbin-Generator .....                                                                          | 8  |
| Gambar 2.2.  | Kurva Karakteristik <i>Input-Output</i> Pembangkit Termal .....                                             | 9  |
| Gambar 2.3.  | Kurva karakteristik <i>Heat-Rate</i> Unit Pembangkit .....                                                  | 9  |
| Gambar 2.4.  | Kurva Karakteristik Incremental Head-Rate/Fuel Coat .....                                                   | 11 |
| Gambar 2.5.  | N-Unit Pembangkit Termal Melayani Beban $P_R$ .....                                                         | 12 |
| Gambar 2.6.  | N-Unit Pembangkit Termal Melayani Beban $P_R$ Melalui<br>Saluran Transmisi .....                            | 15 |
| Gambar 2.7.  | Rangkaian Setara Saluran Transmisi .....                                                                    | 16 |
| Gambar 2.8.  | Rangkaian Setara Saluran Transmisi Pendek .....                                                             | 17 |
| Gambar 2.9.  | Rangkaian Setara Saluran Transmisi Menengah .....                                                           | 18 |
| Gambar 2.10. | Rangkaian Setara Saluran Transmisi Panjang .....                                                            | 19 |
| Gambar 4.1   | Flowchart Program <i>Optimal Power Flow</i> Menggunakan Metode <i>Tabu</i><br><i>Search Algorithm</i> ..... | 37 |
| Gambar 4.2.  | Flowchart Perhitungan Aliran Daya Metode <i>Newton Raphson</i> .....                                        | 38 |
| Gambar 4.3.  | Tampilan Program Utama .....                                                                                | 41 |
| Gambar 4.4.  | Tampilan Input Data IEEE 30-Bus .....                                                                       | 42 |
| Gambar 4.5.  | Tampilan Parameter <i>Tabu Search Algorithm</i> .....                                                       | 42 |
| Gambar 4.6.  | Tampilan Hasil Validasi IEEE-30 Bus .....                                                                   | 43 |
| Gambar 4.7.  | Single Line Diagram Sub-Sistem 150 kV Paiton-Bali .....                                                     | 46 |
| Gambar 4.8.  | Tampilan Input Data General Sub-Sistem Paiton-Bali .....                                                    | 50 |
| Gambar 4.9   | Tampilan Hasil Perhitungan Tegangan, Sudut Tegangan dan Daya<br>Setiap Bus Sebelum Optimasi .....           | 51 |

|              |                                                                                                  |    |
|--------------|--------------------------------------------------------------------------------------------------|----|
| Gambar 4.10. | Tampilan Perhitungan Aliran Daya Sebelum Optimasi.....                                           | 51 |
| Gambar 4.11. | Tampilan Hasil Perhitungan tegangan, Sudut Tegangan dan Daya<br>Setiap Bus Setelah Optimasi..... | 52 |
| Gambar 4.12. | Tampilan hasil Perhitungan Aliran Daya Setelah Optimasi .....                                    | 52 |
| Gambar 4.13. | Tampilan Hasil Perhitungan Total Biaya Sebelum dan Setelah<br>Optimasi.....                      | 53 |

## DAFTAR TABEL

|             |                                                                                                            |    |
|-------------|------------------------------------------------------------------------------------------------------------|----|
| Tabel 4.1.  | Data Generator Dan Konstanta Biaya IEEE 30-Bus.....                                                        | 39 |
| Tabel 4.2   | Data Bus IEEE 30 Bus .....                                                                                 | 39 |
| Tabel 4.3.  | Data Saluran IEEE 30-Bus.....                                                                              | 40 |
| Tabel 4.4.  | Perbandingan Nilai Tegangan dan Keluaran Daya aktif Antara Hasil Program Dengan Hasil Referensi .....      | 44 |
| Tabel 4.5.  | Perameter Unit termal .....                                                                                | 45 |
| Tabel 4.6.  | Persamaan Biaya Pembangkitan .....                                                                         | 45 |
| Tabel 4.7.  | Penomoran Bus .....                                                                                        | 47 |
| Tabel 4.8.  | Data Pembangkitan dan Pembebanan Sub-Sistem 150 kV Paiton-Bali Tanggal 30 Maret 2005 Pukul 18.00 WIB ..... | 48 |
| Tabel 4.9.  | Data Saluran Transmisi Sub-Sistem 150 kV Paiton-Bali .....                                                 | 49 |
| Tabel 4.10. | Hasil Perhitungan Tegangan, Sudut Tegangan Dan Daya Pada Setiap Bus Sebelum Optimasi .....                 | 54 |
| Tabel 4.11. | Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi .....                                         | 55 |
| Tabel 4.12. | Total Pembangkitan, Pembebanan, dan Rugi-rugi Sebelum Optimasi ..                                          | 56 |
| Tabel 4.13. | Hasil Perhitungan Daya Yang dibangkitkan dan Biaya Operasi Sebelum Optimasi .....                          | 56 |
| Tabel 4.14. | Hasil Perhitungan Tegangan, Sudut Tegangan dan Daya Pada Setiap Bus Setelah Optimasi .....                 | 57 |
| Tabel 4.15. | Hasil Perhitungan Aliran daya Antar Saluran Setelah Optimasi .....                                         | 58 |
| Tabel 4.16. | Total Pembangkitan, Pembebanan, dan Rugi-rugi Setelah Optimasi ...                                         | 58 |
| Tabel 4.17. | Hasil Perhitungan Daya Yang dibangkitkan dan Biaya Operasi Setelah Optimasi .....                          | 59 |

|             |                                                       |    |
|-------------|-------------------------------------------------------|----|
| Tabel 4.18. | Perbandingan Tingkat Optimum biaya Pembangkitan ..... | 60 |
| Tabel 4.19. | Perbandingan Tingkat Tegangan Tiap Bus .....          | 61 |

## DAFTAR GRAFIK

|             |                                                                |    |
|-------------|----------------------------------------------------------------|----|
| Grafik 4.1. | Hasil Perhitungan Daya yang Dibangkitkan Sebelum Optimasi..... | 56 |
| Grafik 4.2. | Hasil Perhitungan Daya yang Dibangkitkan Setelah Optimasi..... | 59 |
| Grafik 4.3. | Tingkat Optimum Biaya Pembangkitan .....                       | 60 |
| Grafik 4.4. | Perhitungan Rugi-rugi Daya.....                                | 62 |

# **BAB I**

## **PENDAHULUAN**

### **1.1. Latar Belakang**

Secara garis besar suatu sistem tenaga listrik bisa dibagi menjadi 3 bagian yaitu : sisi pembangkit tenaga listrik, jaringan transmisi dan beban. Untuk suatu operasi pada beban tertentu, perhitungan ekonomis harus tetap merupakan suatu prioritas atau nilai yang harus diperhitungkan disamping hal-hal lain sehingga nantinya diperlukan suatu rencana operasi yang optimum dengan tetap memenuhi beberapa persyaratan pengoperasian sistem tenaga listrik yaitu antara lain : daya yang dibangkitkan cukup untuk memasok beban dan rugi-rugi daya pada saluran transmisi, tegangan bus sesuai dengan ratingnya serta tidak adanya pembebanan lebih pada unit-unit pembangkit yang beroperasi.

Koordinasi antara unit-unit pembangkit yang ada pada sistem tenaga listrik sangat diperlukan untuk mencapai biaya operasi total yang se-optimum mungkin, dengan tetap memperhatikan batasan-batasan dalam sistem.

### **1.2. Permasalahan**

Dalam perencanaan suatu operasi yang optimum diperlukan pembagian pembebanan pada unit-unit pembangkit yang beroperasi sehingga daya yang dibangkitkan cukup memasok beban dan rugi – rugi transmisi yang terjadi.

Berdasarkan hal di atas disusun sebuah pokok permasalahan yakni seberapa besar kemampuan perhitungan dari metode *Enhanced Genetic Algorithm*

dalam pemecahan masalah *Optimal Power Flow* (OPF), sehingga hasil dari analisa tersebut dapat digunakan sebagai salah satu acuan dalam operasi pembangkitan dan penyaluran daya yang ekonomis dan optimal serta untuk mengetahui kondisi sistem selama keadaan operasi. Dalam hal ini kriteria yang akan dibahas adalah pengaturan keluaran daya aktif pembangkit pada suatu sistem tenaga listrik, dengan memperhatikan batasan-batasan yang ada dalam sistem itu sendiri, sehingga didapatkan biaya operasi pembangkitan yang minimal dan batasan-batasan dalam sistem terpenuhi.

### **1.3. Tujuan**

Tujuan pembahasan penelitian ini adalah untuk menganalisis pembagian beban listrik pada unit-unit pembangkit termal menggunakan metode *Tabu Search Algorithm*, untuk memperoleh biaya pembangkitan yang optimal dengan tetap memperhatikan batasan-batasan yang ada dalam sistem tenaga listrik itu sendiri,

### **1.4. Batasan Masalah**

Agar didapatkan sasaran pembahasan yang sesuai dengan judul penelitian ini, maka perlu diadakan pembatasan masalah yang akan dibahas yaitu :

1. Analisa perhitungan pada pembangkit termal saja.
2. Tidak membahas biaya *start up* dan biaya *shut down*.
3. Tidak membahas masalah *spinning reserve*.

### 1.5. Metode Pemecahan Masalah

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

#### 1. Studi Literatur

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

#### 2. Pengumpulan Data

Bentuk data yang digunakan :

- a. Data Kuantitatif, yaitu data yang dapat dihitung atau data yang berbentuk angka – angka.
- b. Data Kualitatif, yaitu data yang berbentuk diagram.

#### 3. Simulasi dan pembahasan masalah

Analisa perhitungan tegangan, sudut phasa tegangan, aliran daya pada tiap saluran serta biaya pembangkitan dengan menggunakan metode *Enhanced Genetic Algorithm* yang disimulasikan dengan komputer.

### 1.6. Kontribusi Penelitian

Dari hasil penelitian ini diharapkan dapat menjadi alternatif pemecahan masalah bagi PT. PLN (persero) dalam usaha meminimalkan biaya operasi khususnya biaya pembangkitan yang merupakan biaya terbesar pada sistem tenaga listrik namun tetap memperhatikan kualitas pelayanan daya listrik.

## **BAB II**

### **KARAKTERISTIK UNIT PEMBANGKIT, OPERASI EKONOMIS DAN PENGATURAN SALURAN TRANSMISI**

#### **2.1. Sistem Operasi Pada Sistem Tenaga Listrik**

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

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

ada, baik itu unit pembangkit termal maupun hidro. Tentu saja kebutuhan beban dalam satu harinya tidak merata akan tetapi dari jam ke jam berbeda sesuai kebutuhan konsumen. Berdasarkan pada lengkung beban yang telah ada maka dapat ditentukan berapa unit pembangkit yang bekerja dan siap bekerja pada hari itu.

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

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

Oleh karena biaya operasi untuk memproduksi daya listrik suatu pembangkit hidro (PLTA) sangat kecil jika dibandingkan dengan pembangkit

termal (PLTU, PLTGU, PLTD) maka dalam pembahasan selanjutnya untuk mendapatkan biaya operasi yang ekonomis akan ditekankan pada unit pembangkit termal saja karena disini akan membutuhkan biaya operasi yang cukup tinggi sehingga usaha penghematan biaya bahan bakar akan sangat berarti. Dengan kata lain dengan mengkoordinasikan operasi pembangkit-pembangkit yang tersedia dengan tepat dan sesuai dengan beban maka akan dapat dicapai suatu keadaan operasi yang ekonomis.

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

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

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

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

## **2.2. Karakteristik Unit Pembangkit**

### **2.2.1. Karakteristik Input-Output<sup>[1]</sup>**

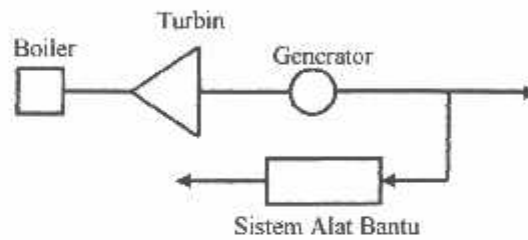
Hal yang paling mendasar dalam operasi ekonomis adalah dengan membuat karakteristik input-output dari unit pembangkit termal. Karena ini diperoleh dari desain perencanaan atau melalui tes pembangkit. Adapun definisi dari karakteristik input-output dari pembangkit itu sendiri adalah formula yang menyatakan hubungan antara input pembangkit sebagai fungsi dari output pembangkit. Sedangkan ciri dari unit boiler turbin generator dapat digambarkan dalam gambar 2.1, dimana unit ini memuat sebuah boiler yang menghasilkan uap untuk menjalankan turbin yang dikopel dengan rotor dari generator.

Pada pembangkit termal, input diberikan dalam satuan panas *Btu/jam* atau *Kalori/jam* dari bahan bakar yang diberikan pada boiler untuk menghasilkan output pembangkit. Sedangkan notasi yang digunakan adalah  $H$  ( $Mbtu/h$ ) atau dalam satuan yang lain ( $Mkal/h$ ). Adapun dalam penelitian ini, perhitungan yang dilakukan adalah  $Mkal/jam$ . Selain itu input dari pembangkit dapat pula

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

$$F = H \times \frac{\text{Rupiah}}{\text{Mbtu}} \dots\dots\dots (2.1)$$

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



Gambar 2.1<sup>[1]</sup>  
Unit Boiler-Turbin-Generator

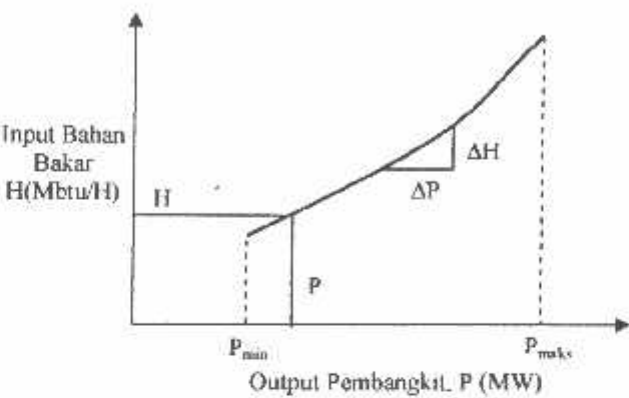
Seperti digambarkan pada gambar 2.1, maka output dari pembangkit tidak hanya untuk dihubungkan dengan sistem saja akan tetapi juga untuk sistem alat bantu dalam pembangkit. Di sini output pembangkit didefinisikan sebagai daya yang dikeluarkan oleh generator untuk beban, diluar untuk keperluan pembangkit itu sendiri. Jadi untuk karakteristik input-output, daya output adalah berupa daya netral dari pembangkit, notasi yang digunakan adalah  $P (MW)$ .

Persamaan karakteristik input-output pembangkit dapat dilihat pada persamaan 2.2. dan 2.3 di bawah ini, sedangkan kurva dari karakteristik input-output pembangkit dapat dilihat pada gambar 2.2.

$$H = f(P) \dots\dots\dots (2.2)$$

atau

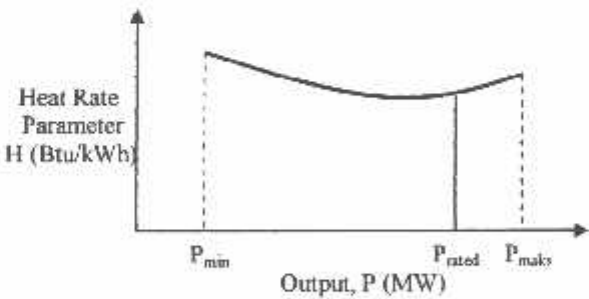
$$F = f(P) \dots\dots\dots (2.3)$$



Gambar 2.2<sup>[1]</sup>  
 Kurva Karakteristik Input-Output Pembangkit Termal

2.2.2. Karakteristik *Heat-Rate*<sup>[1]</sup>

Karakteristik *heat-rate* merupakan karakteristik yang menunjukkan efisiensi dari sebuah mesin. Karakteristik *heat-rate* sebuah unit pembangkit menunjukkan input kalor yang diberikan untuk menghasilkan energi sebesar 1 kilowatt jam pada megawatt output dari suatu unit. Kurva dari karakteristik *heat-rate* ini dapat dilihat pada gambar 2.3 di bawah ini :



Gambar 2.3<sup>[1]</sup>  
 Kurva Karakteristik *Heat-Rate* Unit Pembangkit

**2.2.3. Karakteristik *Incremental Heat Rate* dan *Incremental Fuel Cost*<sup>[1]</sup>**

Perwujudan yang lain dari karakteristik pembangkit adalah karakteristik *incremental heat rate* dan karakteristik *incremental fuel cost*. Karakteristik ini menyatakan hubungan daya output pembangkit sebagai fungsi *incremental heat rate* atau *incremental fuel cost*. Karakteristik *incremental heat rate* ini menunjukkan besarnya perubahan input energi bila ada perubahan output pembangkit pada megawatt output pembangkit.

Kurva karakteristik *incremental heat rate* atau *incremental fuel cost* dapat dilihat pada gambar 2.4. Sedangkan persamaan dari *incremental heat rate* dan persamaan *incremental fuel cost* dapat dilihat pada persamaan 2.4 sampai persamaan 2.7.

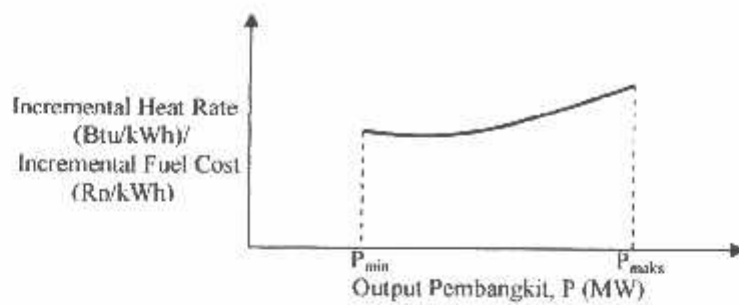
Incremental heat-rate =  $\frac{\Delta H \left( \frac{MBtu}{kWh} \right)}{\Delta P}$  ..... (2.4.)

Incremental fuel cost =  $\frac{\Delta F \left( \frac{Rupiah}{kWh} \right)}{\Delta P}$  ..... (2.5.)

Dengan mengambil harga  $\Delta P$  mendekati nol, maka dapat dinyatakan dengan persamaan berikut ini :

Incremental heat rate =  $\frac{dH \left( \frac{MBtu}{kWh} \right)}{dP}$  ..... (2.6.)

Incremental fuel cost =  $\frac{dF \left( \frac{Rupiah}{kWh} \right)}{dP}$  ..... (2.7.)



Gambar 2.4<sup>[1]</sup>  
Kurva Karakteristik *Incremental Heat Rate/Fuel Cost*

### 2.3. *Economic Dispatch*

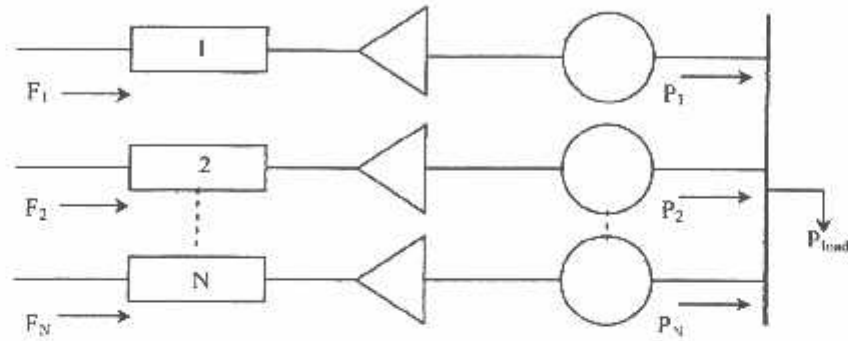
Dalam pembahasan tentang operasi pada sistem tenaga listrik yang ekonomis, maka kita akan selalu membicarakan tentang *economic dispatch*. Pengertian dari *economic dispatch* itu sendiri adalah pembagian pembebanan pada pembangkit-pembangkit yang ada dalam suatu tenaga listrik, secara optimum ekonomis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

#### 2.3.1. *Economic Dispatch Dengan Mengabaikan Rugi-rugi Transmisi*<sup>[1]</sup>

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

Sistem dengan mengabaikan rugi-rugi transmisi dapat dilihat pada gambar 2.5. Sistem ini terdiri dari  $N$  buah unit pembangkit termal yang dihubungkan pada

single busbar yang melayani beban  $P_R$ . Input dari masing-masing unit ditunjukkan oleh  $F_i$ , yang mewakili biaya dari satu unit pembangkit dan output dari masing-masing unit,  $P_i$ , adalah daya yang dihasilkan oleh satu unit pembangkit.



Gambar 2.5<sup>[1]</sup>  
N Unit Pembangkit Termal Melayani Beban  $P_R$

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah dari biaya dari masing-masing unit pembangkit. Dan pembatas yang paling penting adalah bahwa jumlah output dari masing-masing unit pembangkit sama dengan beban di konsumen. Yang menjadi permasalahan adalah meminimalkan total biaya  $F_T$  dengan memperhatikan pembatas  $\phi$  bahwa daya yang dihasilkan oleh pembangkit sama dengan beban yang diterima. Secara matematika pernyataan yang tersebut di atas dapat dinyatakan dengan persamaan berikut :

$$F_T = F_1 + F_2 + F_3 + \dots + F_N \dots \dots \dots (2.8)$$

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

$$\phi = 0 = P_R - \sum_{i=1}^N P_i \dots \dots \dots (2.9)$$

Persamaan di atas adalah pembatas yang merupakan masalah dari optimasi dan ini dapat dipecahkan dengan menggunakan kalkulus tingkat lanjut yang melibatkan fungsi Lagrange. Dimana fungsi didapat dengan cara menambahkan pembatas  $\phi$  yang telah dikalikan dengan faktor pengali Lagrange  $\lambda$  pada fungsi tujuan  $F_T$ . Fungsi Lagrange dapat ditunjukkan dengan persamaan di bawah ini :

$$L = F_T + \lambda \phi \dots\dots\dots (2.10)$$

Persamaan Lagrange di atas merupakan fungsi dari output pembangkit  $P_i$  dan faktor pengali Lagrange  $\lambda$ . Keadaan optimal dari fungsi tujuan  $F_T$  dapat diperoleh dengan operasi gradien dari persamaan Lagrange sama dengan nol.

$$\frac{\partial L}{\partial P_i} = \frac{dF_i(P_i)}{dP_i} - \lambda = 0 \dots\dots\dots (2.11)$$

$$\text{atau } 0 = \frac{dF_i}{dP_i} - \lambda \dots\dots\dots (2.12)$$

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

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

$$\frac{dF_i}{dP_i} = \lambda \text{ ada N buah persamaan} \dots\dots\dots (2.13)$$

$$P_i \min \leq P_i \leq P_i \text{ maks ada } 2 \text{ N pertidaksamaan} \dots\dots\dots (2.14)$$

$$\sum_{i=1}^N P_i = P_R \dots\dots\dots (2.15)$$

Dari batasan pertidaksamaan pembatas di atas dapat diperluas menjadi :

$$\frac{dF_i}{dP_i} = \lambda \text{ untuk } P_i \min \leq P_i \leq P_i \text{ maks} \dots\dots\dots (2.16)$$

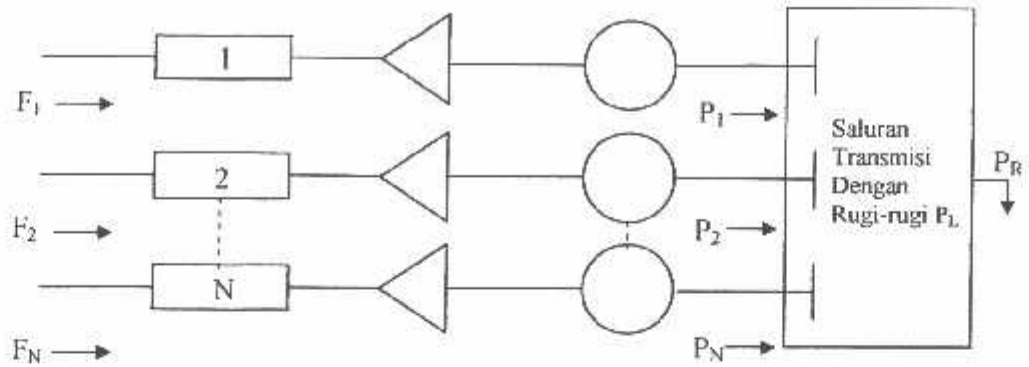
$$\frac{dF_i}{dP_i} \leq \lambda \text{ untuk } P_i = P_i \text{ maks} \dots\dots\dots (2.17)$$

$$\frac{dF_i}{dP_i} \geq \lambda \text{ untuk } P_i = P_i \min \dots\dots\dots (2.18)$$

### 2.3.2. *Economic Dispatch Dengan Memperhitungkan Rugi-rugi Transmisi*<sup>[1]</sup>

Sistem dengan memperhitungkan rugi-rugi transmisi dapat dilihat pada gambar 2.6. Sistem ini terdiri dari N buah unit pembangkit termal dihubungkan melalui saluran transmisi yang melayani beban  $P_R$ . Input dari masing-masing unit ditunjukkan oleh  $F_i$  yang mewakili biaya dari satu unit pembangkit dan output dari masing-masing unit  $P_i$  adalah daya yang dihasilkan oleh satu unit pembangkit.

Total biaya rata-rata yang harus ditanggung oleh sistem adalah jumlah dari biaya dari masing-masing unit pembangkit. Dan batasan yang paling penting adalah bahwa jumlah output dari masing-masing unit pembangkit sama dengan beban di konsumen ditambah rugi-rugi transmisi.



Gambar 2.6<sup>[1]</sup>  
N Unit Pembangkit Termal Melayani Beban  $P_R$  Melalui Saluran Transmisi

$$P_R + P_L - \sum_{i=1}^N P_i = \phi = 0 \quad \dots\dots\dots (2.19)$$

$$L = L_T + \lambda \phi \quad \dots\dots\dots (2.20)$$

$$\phi = \sum_{i=1}^N P_i - P_R - P_L = 0 \quad \dots\dots\dots (2.21)$$

Persamaan Lagrangenya adalah

$$L = \sum_{i=1}^N F_i - \lambda (\sum_{i=1}^N P_i - P_R - P_L) \quad \dots\dots\dots (2.22)$$

$$\frac{\partial L}{\partial P_i} = \frac{dF_i}{dP_i} - \lambda (1 - \frac{\partial P_L}{\partial P_i}) \quad \dots\dots\dots (2.23)$$

#### 2.4. Fungsi Biaya Bahan Bakar<sup>[1]</sup>

Biaya bahan bakar merupakan unsur biaya yang penting dalam operasi sistem pembangkit thermal. Fungsi biaya bahan bakar  $F_i(P_i)$  untuk setiap

pembangkit terhadap daya keluaran diekspresikan dalam bentuk fungsi kuadrat, yang dapat dinyatakan sebagai berikut:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i \dots\dots\dots (2.24)$$

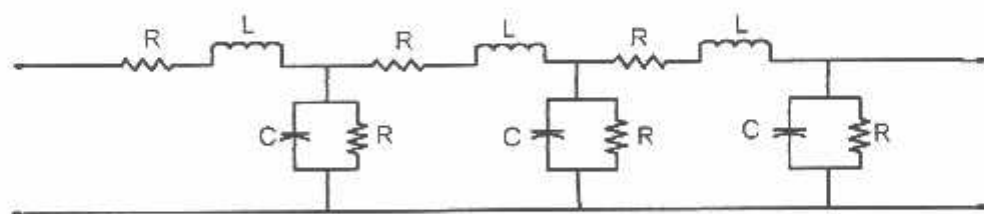
Dimana:

$a, b, c$  = konstanta persamaan dari unit ke - i

$P_i$  = daya keluaran dari unit ke - i

## 2.5. Sistem Jaringan Transmisi

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



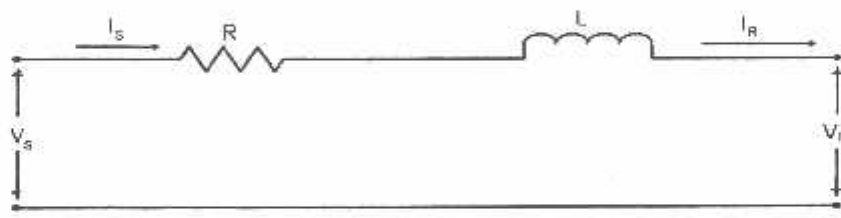
Gambar 2.7<sup>[2]</sup>  
Rangkaian Setara Saluran Transmisi

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

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

2.5.1. Saluran Transmisi Pendek

Rangkaian ekivalen untuk saluran transmisi pendek diperlihatkan pada gambar 2.8, dimana  $I_s$  dan  $I_R$  merupakan arus pada ujung pengirim dan ujung penerima, sedangkan  $V_S$  dan  $V_R$  adalah tegangan saluran terhadap netral pada ujung pengirim dan ujung penerima.



Gambar 2.8<sup>[2]</sup>  
Rangkaian Setara Saluran Transmisi Pendek

Karena tidak ada cabang paralel (shunt), arus pada ujung–ujung pengirim dan penerima akan sama besar :

$$I_s = I_R \dots\dots\dots (2.25)$$

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

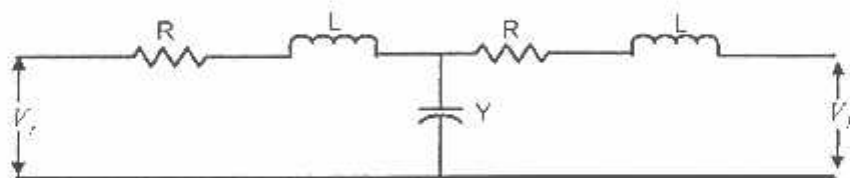
$$V_S = V_R + I_R Z \dots\dots\dots (2.26)$$

### 2.5.2. Saluran Transmisi Menengah

Bertambahnya saluran menyebabkan kapasitansi shunt bertambah besar dan tidak dapat diabaikan. Saluran transmisi jarak menengah pada umumnya digambarkan dengan rangkaian  $\pi$  atau rangkaian T. Dari dua versi ini rangkaian  $\pi$  lebih umum dipakai dari pada rangkaian T.



(a) Rangkaian Ekuivalen  $\pi$



(b) Rangkaian Ekuivalen T

Gambar 2.9<sup>[2]</sup>  
Rangkaian Setara Saluran Transmisi Menengah

Untuk rangkaian  $\pi$  berlaku :

$$V_s = \left[ \frac{ZY}{2} + 1 \right] V_R + ZI_R \dots\dots\dots (2.27)$$

$$I_s = \left[ \frac{ZY}{2} + 1 \right] YV_R + \left[ \frac{ZY}{4} + 1 \right] I_R \dots\dots\dots (2.28)$$

Untuk rangkaian T berlaku :

$$V_s = \left[ \frac{ZY}{2} + 1 \right] V_R + \left[ \frac{ZY}{4} + 1 \right] ZI_R \dots\dots\dots (2.29)$$

$$I_s = YV_R + \left[ \frac{ZY}{2} + 1 \right] I_R \dots\dots\dots (2.30)$$

Dimana :

$V_s, I_s$  = Tegangan, Arus sisi kirim

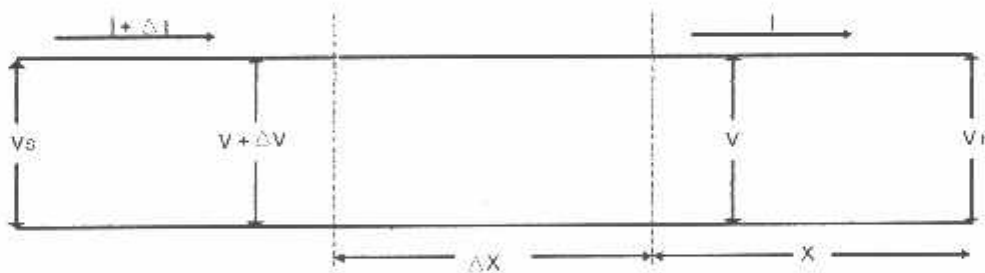
$V_R, I_R$  = Tegangan, Arus sisi terima

$Z$  = Impedansi seri total saluran transmisi

$Y$  = Admitansi shunt total saluran

### 2.5.3. Saluran Transmisi Panjang

Pada saluran panjang parameter-parameter saluran tidak terpusat menjadi satu, melainkan tersebar merata di seluruh panjang saluran.



Gambar 2.10<sup>[21]</sup>  
Diagram Skema Saluran Transmisi Panjang

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

$$V = \frac{V_R + I_R Z_C}{2} e^{+\gamma x} + \frac{V_G - I_G Z_C}{2} e^{-\gamma x} \dots\dots\dots (2.31)$$

$$I = \frac{\frac{V_R}{Z_C} + I_R}{2} e^{+\gamma x} + \frac{\frac{V_R}{Z_C} - I_R}{2} e^{-\gamma x} \dots\dots\dots (2.32)$$

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

$$V(x) = V_R \cosh \tau x + I_R Z_C \sinh \tau x \dots\dots\dots (2.33)$$

$$I(x) = I_R \cosh \tau x + \frac{V_R}{Z_C} \sinh \tau x \dots\dots\dots (2.34)$$

Dimana :

$$\begin{aligned} \tau &= \text{Konstanta rambatan pada saluran} &&= \sqrt{ZY} \\ Z_C &= \text{Impedansi karakteristik saluran} &&= \sqrt{\frac{Z}{Y}} \end{aligned}$$

**2.6. Sistem Per-Unit<sup>[3]</sup>**

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

$$\text{Besaran Per-Unit} = \frac{\text{Besaran sebenarnya}}{\text{Besaran dasar}} \dots\dots\dots (2.35)$$

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

- Untuk data 1 phasa

Arus dasar

$$I_d = \frac{\text{kVA dasar 1 phasa}}{\text{kV dasar L - N}} \dots\dots\dots (2.36)$$

Impedansi dasar

$$Z_d = \frac{(kV \text{ dasar L - N})^2 \times 10^{-3}}{kVA \text{ dasar 1 fase}} \dots\dots\dots (2.37)$$

$$Z_d = \frac{(kV \text{ dasar L - N})^2}{MVA \text{ dasar 1 fase}} \dots\dots\dots (2.38)$$

■ Untuk data 3 phasa

Arus dasar

$$I_d = \frac{kVA \text{ dasar 3 phasa}}{\sqrt{3} \text{ kV dasar L - L}} \dots\dots\dots (2.39)$$

Impedansi dasar

$$Z_d = \frac{(kV \text{ dasar L - L})^2 \times 10^{-3}}{kVA \text{ dasar 3 phasa}} \dots\dots\dots (2.40)$$

$$Z_d = \frac{(kV \text{ dasar L - L})^2}{MVA \text{ dasar 3 phasa}} \dots\dots\dots (2.41)$$

Dalam persamaan di atas nilai-nilai besaran diberikan untuk rangkaian satu phasa. Jadi tegangannya adalah tegangan 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 :

$$Z = \frac{\text{Impedansi sebenarnya } Z (\Omega)}{\text{Impedansi dasar } Z_d (\Omega)} \dots\dots\dots (2.42)$$

**BAB III**  
***OPTIMAL POWER FLOW MENGGUNAKAN***  
***METODE TABU SEARCH ALGORITHM***

**3.1. Konsep Aliran Daya<sup>[3]</sup>**

Penyelesaian aliran daya pada setiap bus menggunakan metode Newton-Rapshon perlu diketahui minimal dua parameter, dari empat parameter yang diperlukan (daya aktif  $P$ , daya reaktif  $Q$ , besar tegangan  $|v|$ , dan sudut fasa tegangan  $\delta$ ). Berdasarkan pada parameter yang diketahui, bus pada sistem dapat diklasifikasikan dalam tiga kategori bus, yaitu :

1. Bus Beban

Parameter-parameter yang diketahui adalah  $P$  dan  $Q$ , parameter-parameter yang tidak diketahui adalah  $|v|$  dan  $\delta$ .

2. Bus Generator

Parameter-parameter yang diketahui adalah  $P$  dan  $|v|$ , parameter-parameter yang tidak diketahui adalah  $Q$  dan  $\delta$ , kendala untuk bus ini adalah daya  $Q$ , bila dalam perhitungan integrasinya tidak dipenuhi, maka bus ini diganti menjadi bus beban.

3. Bus *Swing/Slack*

Parameter-parameter yang diketahui adalah  $|v|$  dan  $\delta$ , biasanya dianggap sebagai bus referensi. Bus ini selalu mempunyai generator. Dalam perhitungan

aliran daya, P dan Q bus ini tidak perlu diperhitungkan. Gunanya untuk memenuhi kekurangan daya seluruhnya setelah perhitungan selesai dilakukan.

Daya total yang mengalir pada setiap bus dituliskan sebagai berikut :

$$S_k = P_k + jQ_k = V_k^* I_k$$

Atau

$$P_k - jQ_k = V_k^* \sum_{n=1}^N Y_{kn} V_n \dots\dots\dots (3.1)$$

Dari persamaan  $V_k = v_k + j\delta_k$  dan  $Y_{k11} = G_{k11} - jB_{k11}$ , maka persamaan 3.1, menjadi :

$$P_k - jQ_k = (v_k + j\delta_k) \sum_{n=1}^N (G_{kn} - jB_{kn})(V_n + j\delta_n) \dots\dots\dots (3.2)$$

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

$$P_k = \sum_{n=1}^N \{v_k(v_n G_{kn} + \delta_k B_{kn}) + \delta_k(\delta_n G_{kn} - v_k B_{kn})\} \dots\dots\dots (3.3)$$

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

### 3.1.1. Metode Newton Rapshon<sup>[4]</sup>

Proses yang dilakukan adalah membandingkan antara daya yang ditempatkan berdasarkan data ( $P_{k, \text{ sched}}$  dan  $Q_{k, \text{ sched}}$ ) dengan daya hasil perhitungan ( $P_{k, \text{ calc}}$  dan  $Q_{k, \text{ calc}}$ ) menggunakan persamaan 3.3. dan 3.4. di atas. Selisih daya yang diterapkan dan perhitungan ( $\Delta P_k$  dan  $\Delta Q_k$ ) dihitung dengan persamaan :

$$\Delta P_k = P_{k, \text{ sched}} - P_{k, \text{ calc}} \dots\dots\dots (3.5)$$

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

Selisih daya dihitung dengan persamaan 3.5. dan persamaan 3.6. digunakan untuk menghitung nilai perubahan parameter tegangan bus, yaitu  $\Delta|V_k|$  dan  $\Delta\delta_k$ , yaitu dengan menggunakan elemen Jacobian, sehingga koreksi terhadap nilai parameter tegangan yang telah ditetapkan nilai awal sebelumnya. Elemen Jacobian sendiri merupakan turunan parsial P dan Q terhadap masing-masing variabel pada persamaan 3.3. dan 3.4., yang dalam bentuk matriks dituliskan sebagai :

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

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |v| \end{bmatrix} \dots \dots \dots (3.7)$$

Dimana elemen-elemen jacobian dapat dihitung dengan menurunkan persamaan-persamaan 3.3. dan 3.4.

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

menghitung kembali daya  $P_{k, calc}$  dan  $Q_{k, calc}$  menurut persamaan 3.3. dan 3.4. Proses ini terus berulang, yang disebut iterasi hingga mencapai kondisi dimana nilai perubahan daya  $\Delta P$  dan  $\Delta Q$  konvergen mencapai suatu nilai minimum yang telah ditentukan (berkisar 0,001 hingga 0,0001 pu)

### 3.2. Optimal Power Flow<sup>[5]</sup>

OPF dipakai untuk mengoptimasi fungsi obyektif dalam pengoperasian sistem tenaga (biaya bahan bakar pada pembangkit termal) pada saat terpenuhinya pengaturan sistem operasi pembatas, termasuk ketentuan pembatas oleh jaringan.

Masalah OPF dapat diformulasikan sebagai masalah optimasi matematika seperti berikut:

$$\text{Min} f(x,u) \dots\dots\dots(3.8)$$

$$\text{S.t. } g(x,u) = 0 \dots\dots\dots(3.9)$$

$$h(x,u) \leq 0 \dots\dots\dots(3.10)$$

$$u \in U \dots\dots\dots(3.11)$$

dimana

$$x = [ \theta^T \quad U_L^T ]^T \dots\dots\dots(3.12)$$

$$u = [ P_G^T \quad U_G^T ]^T \dots\dots\dots(3.13)$$

Batas-batas persamaan (3.9) adalah persamaan aliran daya non linear. Batas-batas pertidaksamaan (3.10) adalah batas-batas fungsi pengoperasian, seperti :

- Batas-batas besarnya tegangan bus beban ;
- Kemampuan reaktif generator ;
- Batas-batas keluaran daya aktif *slack* bus.

Batas-batas (3.11) mendefinisikan bagian yang mungkin terjadi pada masalah kontrol variabel, seperti :

- Batas-batas unit keluaran daya aktif;
- Batas-batas besarnya tegangan bus generator.

### 3.3. Metode *Tabu Search*

#### 3.3.1. Konsep Dasar

*Tabu Search* adalah suatu metode optimasi yang didasarkan pada prinsip yang berasal dari proses biologi, yang menghindari jebakan optimum lokal dengan menggunakan struktur memori yang melarang kembali ke solusi sebelumnya (*backtracking*). *Tabu Search* merupakan suatu prosedur optimasi kuat yang telah sukses dijalankan pada sejumlah masalah suatu optimasi kombinatorial.

*Tabu Search* menggunakan sistem memori fleksibel ( berbeda dengan sistem "tanpa memori" seperti *Genetic Algorithm (GA)*). Perhatian khusus diberikan kepada komponen memori jangka pendek dari *Tabu Search*, yang menghasilkan solusi yang melebihi solusi terbaik yang bisa dihasilkan oleh metode lain untuk masalah-masalah yang beragam.

*Tabu Search* juga mempunyai banyak ciri-ciri yang kuat meliputi:

- Memperoleh solusi berkualitas tinggi yang tidak sangat tergantung pada pilihan solusi awal.

- Tidak diperlukannya model matematis rumit dari masalah. Dalam hal ini masalah yang sulit di rumuskan bisa diimplementasikan dengan mudah.
- Kemampuan untuk memulai perhitungan dengan solusi tertentu dan mencoba memperbaikinya. Ciri tersebut dapat digunakan untuk meningkatkan suatu solusi keluaran yang diperoleh dari metode sub-optimal.

Dalam solusi *Tabu Search* pada masalah *OPF* optimasi kombinatorial dipecahkan dengan menggunakan algoritma *Tabu Search* sedangkan masalah *Economic Dispatch* (EDP) dipecahkan melalui metode *Newton Rapshon*.

### 3.3.2. Prosedur *Tabu Search*

*Tabu Search* adalah sebuah prosedur peningkatan iteratif yang dimulai dari beberapa solusi awal dan mencoba untuk menentukan solusi yang lebih baik dengan cara algoritma penurunan terbesar. Tetapi *tabu search* ditandai kemampuan untuk menghindari optima lokal (yang biasanya menyebabkan algoritma penurunan sedrhana berhenti) dengan menggunakan memori jangka pendek dari solusi-solusi terbaru. Selain itu *tabu list* memungkinkan *backtracking* kepada solusi sebelumnya, yang pada akhirnya akan menghasilkan solusi yang lebih baik melalui arah yang berbeda dan membimbing langkah pergerakan maju menuju suatu titik optimum.

Dua elamen utama dari algoritma *tabu serch* (TSA) adalah kriteria daftar tabu(TL) dan level aspirasi(AL).

### 1. Kriteria Daftar Tabu(TL)

Guna menghindari kembali ke optimum lokal yang baru saja dilakukan, perpindahan selanjutnya yang sudah ada seperti yang ada dalam daftar tabu yang telah ditetapkan untuk mencapai solusi yang optimal harus tidak diperbolehkan. Hal ini dilakukan dengan menyimpan perpindahan pada struktur data yang disebut *daftar tabu*. Elemen-elemen daftar tabu disebut perpindahan tabu. Berkaitan dengan perpindahan tabu kita dapat menyimpan bias pencarian menuju suatu titik dengan nilai fungsi yang lebih rendah dan keluar dari solusi optima lokal. Segera setelah solusi coba-coba (*Trial*) dibangkitkan, maka solusi coba-coba dicek apakah ada dalam daftar tabu atau tidak, jika ada, maka ruang pencarian yang berhubungan akan dikurangi, begitu seterusnya. Pendekatan ini dimaksudkan untuk meningkatkan optimalitas lokal dengan strategi pelarangan gerakan tertentu (atau secara lebih luas, menghentikan). Tujuan dari klasifikasi *tabu* adalah untuk mencegah adanya siklus atau pengulangan pencarian solusi dengan cara yang sama (pernah digunakan).

Elemen TL ditentukan oleh sebuah fungsi yang menggunakan informasi historis dari proses pencarian, yaitu peningkatan sampai iterasi  $Z$  dimasa lalu, dimana  $Z$  (ukuran daftar tabu) dapat ditetapkan atau bervariasi tergantung pada aplikasi pada tahap pencarian.

TL diperbolehkan mencatat gerakan dalam urutan pembuatannya. Setiap waktu elemen baru ditambahkan pada bagian bawah daftar, elemen terlama dalam daftar akan dihilangkan dibagian atasnya

Adapun filosofi dari pencarian dengan menggunakan daftar tabu(*tabu list*) bertujuan agar langkah pencarian solusi terbaik tidak terjebak pada optimum lokal namun langkah pencarian dibimbing untuk selalu menuju ke titik optimum global dan mencegah terjadinya *backtracking* dimana suatu kondisi dimana kembali pada solusi yang pernah dilalui atau suatu keadaan yang menjauhi titik optimum global. Hal ini dikarenakan pada sistem *tabu search* menggunakan memori jangka pendek yang mencatat atau merekam solusi-solusi terbaik yang pernah didapatkan untuk dimasukkan dalam daftar tabu. Selain itu struktur memori tersebut yang melarang atau menghentikan gerakan tertentu yang akan kembali ke solusi yang telah digunakan.

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 ke solusi yang baru saja dilakukan. Secara empiris, ukuran daftar tabu harus tumbuh sesuai dengan ukuran masalah. Dalam skripsi kali ini menggunakan daftar perpaduan yang dinamis yang berperan sebagai daftar tabu, yang ukurannya adalah 10 untuk mencatat solusi yang baru saja dilakukan. Hal ini berarti bahwa perpindahan akan disimpan dalam tabu untuk durasi waktu selama 10 perpindahan. Untuk perpindahan yang selanjutnya di cek apakah lebih baik dari yang ada dalam daftar tabu, dan jika lebih baik diambil dan dipilih dan dipilih dari daftar tabu yang terjelek dibuang dan apabila tidak proses dilanjutkan.

## 2. *Kriteria Aspirasi(Al)*

Kriteria aspirasi merupakan setiap masukan didalam daftar tabu yang berupa suatu nilai tertentu sebagai fungsi evaluasi, oleh karena itu kriteria aspirasi digunakan untuk memungkinkan perpindahan tabu yang diberikan jika perpindahan tersebut dinilai memberikan solusi lebih baik. Dengan kata lain, kriteria aspirasi adalah untuk memungkinkan perpindahan tabu yang sangat baik untuk diseleksi jika tingkat aspirasinya diperoleh dengan baik. Status tabu dari sebuah gerakan bukanlah absolut, tapi dapat dilanggar jika kondisi tertentu bisa dipenuhi, yang dinyatakan dalam level aspirasi(AL). jika kriteria aspirasi yang tepat terpenuhi, gerakan bisa diterima meski berada dalam klasifikasi tabu. Intinya, AL dimaksudkan untuk meniadakan status tabu jika suatu pergerakan”cukup baik” dengan pengertian menghasilkan solusi dengan fungsi objektif yang lebih baik dari pada solusi sebelumnya dengan gerakan yang sama. Akibatnya, AL yang dihubungkan dengan setiap gerakan dalam TL adalah sama dengan nilai fungsi objektif yang diperoleh ketika menjalankan gerakan. Efektifitas dari penggunaan kriteria aspirasi adalah menambahkan sedikit fleksibilitas dalam tabu search dengan mengarahkan pencarian kepada gerakan yang atraktif.

### 3.3.3. **Kriteria Penghentian**

Ada beberapa kondisi penghentian dari pencarian. Dalam hal ini pencarian dihentikan jika dua kondisi berikut terpenuhi:

1. Jumlah iterasi yang dijalankan sejak solusi terbaik yang berubah terakhir adalah lebih besar daripada jumlah iterasi maksimum.
2. Jumlah Iterasi maksimum bisa dicapai.

### 3.3.4. Solusi *Tabu Search Algorithm* pada *Optimal Power Flow*<sup>[6]</sup>

Metode *Tabu Search* yang merupakan metode untuk memecahkan masalah optimisasi yang dikembangkan terutama untuk memecahkan combinatorial permasalahan optimisasi dengan efisiensi tinggi. Masalah OPF di dalam masalah optimisasi secara umum dapat ditulis sebagai berikut:

Minimize  $f(x)$

yang digunakan untuk  $g_i(x) = 0$

$$h_j(x) \leq 0$$

$$l \leq x \leq u$$

Dimana:  $f(x)$  merupakan total biaya pembangkitan,  $g_i(x)=0$  adalah batas persamaan sedangkan  $h_j(x) \leq 0$

Adalah batasan ketidaksamaan

Fungsi Objektif, persamaan dan batasan ketidaksamaan, diuraikan sebagai berikut.

Fungsi Objektif: total biaya pembangkitan<sup>[1]</sup>

$$F_T = \sum_{i=1}^{N_G} f(P_{Gi})$$

Batasan Persamaan<sup>[1]</sup>

$$P_{sch,k} - \sum_{i=1}^{N_A} |Y_{ik} V_i V_k| \cos(\theta_{ik} + \delta_k - \delta_i) = 0$$

dan

$$Q_{sch,i} - \sum_{k=1}^{N_B} |Y_{ik}| V_i V_k \sin(\theta_{ik} + \delta_k - \delta_i) = 0$$

Batasan Ketidaksamaan: Batas kendali<sup>[1]</sup>

$$P_{Gi}^{min} \leq P_{Gi} \leq P_{Gi}^{max}$$

$$Q_{Gi}^{min} \leq Q_{Gi} \leq Q_{Gi}^{max}$$

$$V_j^{min} \leq V_j \leq V_j^{max}$$

$$T_k^{min} \leq T_k \leq T_k^{max}$$

$$V_{ref}^{min} \leq V_{ref} \leq V_{ref}^{max}$$

Dimana:

|                              |                                              |
|------------------------------|----------------------------------------------|
| $F_T$                        | Total biaya pembangkitan                     |
| $f(P_{Gi})$                  | Biaya bahan bakar generator I                |
| $N_{Gi}$                     | Jumlah generator                             |
| $P_{sch,k}$                  | Daya aktif yang dijadwalkan pada bus k       |
| $Q_{sch,k}$                  | Jadwalkan tenaga reaktif pada bus k          |
| $Y_{ik}$                     | Magnitude admitansi antara buses i dan k     |
| $\theta_{ik}$                | Sudut admitansi antara buses i dan k         |
| $V_i$                        | Tegangan magnitude pada bus I                |
| $\delta_i$                   | Sudut tegangan pada bus I                    |
| $N_B$                        | Jumlah buses                                 |
| $P_{Gi}^{min}, P_{Gi}^{max}$ | Batas keluaran daya aktif dari generator I   |
| $Q_{Gi}^{min}, Q_{Gi}^{max}$ | Batas keluaran daya reaktif dari generator I |
| $V_j^{min}, V_j^{max}$       | Batas besar tegangan pada bus j              |

#### 3.3.4.1. Proses formulasi Tabu search

- 1) Hasilkan secara acak suatu solusi awal( $x_0$ ). Set inisial  $x_0$  sebagai sebagai solusi optimal  $x'' (x'' = x_0)$
- 2) Ciptakan secara acak *Neighborhood*  $N(x)$  di sekitar  $x_0$ . Tetapkan  $x'$  adalah solusi yang terbaik di dalam *Neighborhood*
- 3) Jika  $f(x') < f(x'')$  pergi untuk melangkah 4, selain itu menetapkan  $x'$  adalah solusi yang terbaik *replacing*  $x'' (x'' = x')$
- 4) Jika  $x'$  bukanlah di (dalam) tabu daftar, kemudian membaharui daftar tabu dengan memasukkan atau menyisipkan  $x'$  di (dalam) daftar Tabu.
- 5) Memeriksa ukuran penghentian. Jika aturan tempat berhenti dicukupi, kemudian stop dan menyimpan solusi yang terbaik jika tidak kembali ke melangkah 2.

## BAB IV

### HASIL DAN ANALISIS HASIL

#### 4.1. Program Komputer *Optimal Power Flow* Menggunakan Metode Tabu Search.

Dalam penyelesaian masalah OPF diperlukan bantuan program computer untuk proses perhitungan yang membutuhkan ketelitian dan keakuratan.

Program computer ini menggunakan bahasa pemrograman Delphi 7. Bahasa pemrograman Delphi merupakan bahasa pemrograman yang relative mudah untuk dipelajari dan digunakan.

##### 4.1.1. Algoritma Program

Untuk algoritma aliran daya dengan metode Newton Rapshon adalah sebagai berikut:

1. Bentuk matriks admitansi bus:  $[Y_{bus}]$
2. Penetapan harga awal tegangan dan sudut fasa untuk semua bus kecuali bus slack  $[V_i(0), \delta(0)]$
3. Menentukan nomor iterasi awal,  $k=0$
4. Menghitung injeksi daya aktif dan reaktif pada setiap bus dengan persamaan kecuali bus slack
5. Menghitung selisih daya yang dijadwalkan dengan injeksi daya bus dari perhitungan.

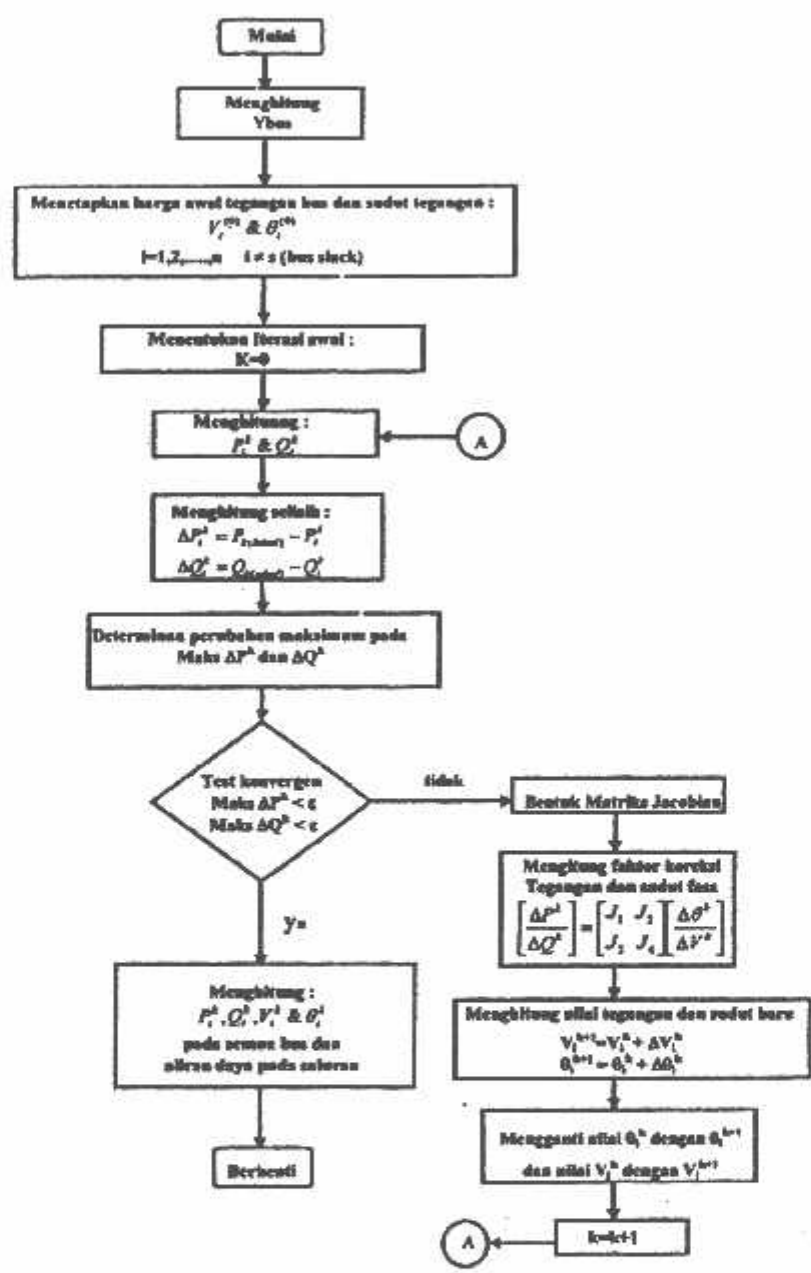
6. Menentukan perubahan maksimal pada daya aktif dan daya reaktif
7. Membandingkan apakah selisih daya sudah sama atau lebih kecil dari  $\epsilon$
8. Jika “ya” hitung daya aktif dan reaktif, tegangan dan sudut fasa tegangan pada setiap bus, serta aliran daya pada saluran dan perhitungan selesai, jika “tidak” lanjutkan ke langkah berikutnya
9. Membentuk elemen matriks Jacobian
10. Menghitung faktor koreksi tegangan dan sudut fasa setiap bus kecuali bus slack dan bus generator
11. Menghitung nilai tegangan dan sudut fasa yang baru
12. Mengganti nilai sudut fasa yang lama dengan sudut fasa yang baru, tegangan yang lama dengan tegangan yang baru
13. Perhitungan dilanjutkan ke langkah 4 dengan nilai iterasi yang baru sampai hasil yang didapatkan konvergen.

Sedangkan urutan langkah-langkah dalam program komputer yang menggunakan metode TSA, dapat dilihat pada algoritma program berikut:

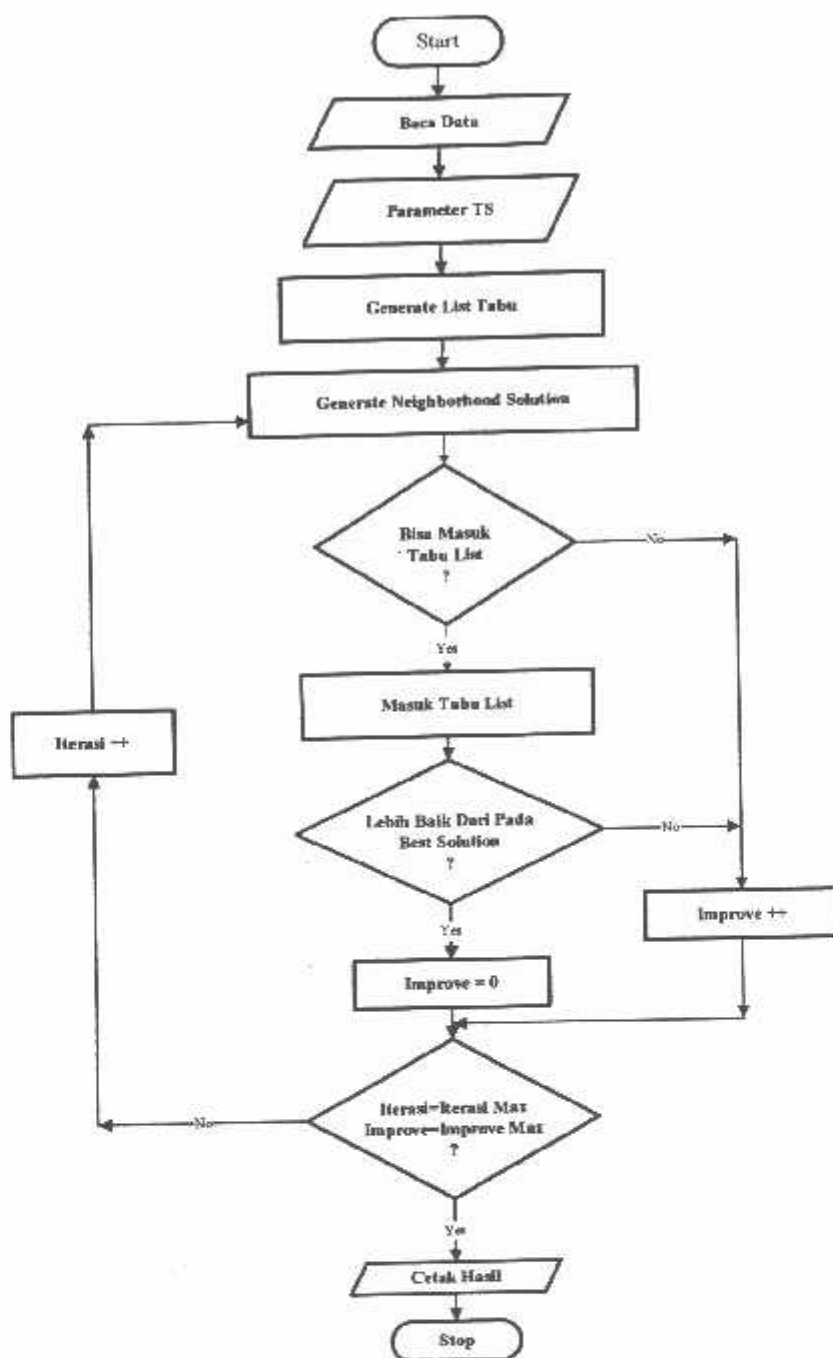
1. Memasukkan inputan data beban yang meliputi tegangan ( $V$ ), sudut fasa tegangan, daya aktif ( $P$ ), daya reaktif ( $Q$ ), dan data impedansi saluran, data pembangkitan  $P_{maks}$ ,  $P_{min}$ ,  $Q_{maks}$ ,  $Q_{min}$ , konstanta biaya.
2. Memasukkan parameter Tabu Search
3. Pembentukan Daftar Tabu (*Tabu List*)
4. Pembentukan *neighborhood solution* yang di dapat dari parameter-parameter hasil perhitungan aliran daya dengan menggunakan metode Newton Rapshon.

5. Memeriksa apakah *neighborhood solution* yang didapat bisa masuk *Tabu list*?, jika tidak lanjutkan ke langkah 9
6. Memasukkan solusi pada *Tabu List*
7. Melakukan pengecekan apakah solusi yang didapat apakah lebih baik dari *Best Solution*? Jika tidak lanjutkan ke langkah 9.
8. Memasukkan perbaikan sama dengan nol.
9. Melanjutkan ke perbaikan selanjutnya
10. Melakukan pengecekan apakah iterasi telah maksimal dan apakah *Improve* telah maksimal?, jika tidak lanjutkan ke langkah 12.
11. Cetak hasil solusi.
12. Melakukan iterasi selanjutnya.
13. *Stop*

### 4.1.2. Flowchart Program



Gambar 4.1  
Flowchart Perhitungan Aliran Daya Metode Newton Rapshon



Gambar 4.2  
Flowchart Metode Tabu Search Algorithm

4.2. Validasi Program Menggunakan Data Sistem IEEE 30-Bus

4.2.1. Data Sistem IEEE 30-Bus<sup>[8]</sup>

Tabel 4.1<sup>[8]</sup>  
Data Generator Dan Konstanta Biaya IEEE 30-Bus

| No | No Bus | $P_G^{Min}$ | $P_G^{Max}$ | $Q_G^{Min}$ | $Q_G^{Max}$ | Konstanta Biaya |        |        |
|----|--------|-------------|-------------|-------------|-------------|-----------------|--------|--------|
|    |        | MW          | MW          | MVAR        | MVAR        | A               | B      | C      |
| 1  | 1      | 50          | 200         | -20         | 150,0       | 0,00375         | 2,0000 | 0,0000 |
| 2  | 2      | 20          | 80          | -20         | 60,0        | 0,01750         | 1,7500 | 0,0000 |
| 3  | 5      | 15          | 50          | -15         | 62,5        | 0,01750         | 107500 | 0,0000 |
| 4  | 8      | 10          | 35          | -15         | 50,0        | 0,00834         | 3,2500 | 0,0000 |
| 5  | 11     | 10          | 30          | -10         | 40,0        | 0,02500         | 3,0000 | 0,0000 |
| 6  | 13     | 12          | 40          | -15         | 45,0        | 0,02500         | 3,0000 | 0,0000 |

Tabel 4.2<sup>[8]</sup>  
Data Bus IEEE 30 Bus

| No Bus | Tegangan (pu) | $P_{Gen}$ (MW) | $Q_{Gen}$ (MVAR) | $P_{Load}$ (MW) | $Q_{Load}$ (MVAR) | Tipe Bus |
|--------|---------------|----------------|------------------|-----------------|-------------------|----------|
| 1      | 1,0500        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 1        |
| 2      | 1,0338        | 57,5600        | 0,0000           | 21,7000         | 12,7000           | 2        |
| 3      | 1,0000        | 0,0000         | 0,0000           | 2,4000          | 1,2000            | 3        |
| 4      | 1,0000        | 0,0000         | 0,0000           | 7,6000          | 1,6000            | 3        |
| 5      | 1,0058        | 24,5600        | 0,0000           | 94,2000         | 19,0000           | 2        |
| 6      | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 7      | 1,0000        | 0,0000         | 0,0000           | 22,8000         | 10,9000           | 3        |
| 8      | 1,0230        | 35,0000        | 0,0000           | 30,0000         | 30,0000           | 2        |
| 9      | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 10     | 1,0000        | 0,0000         | 0,0000           | 5,8000          | 2,0000            | 3        |
| 11     | 1,0913        | 17,9300        | 0,0000           | 0,0000          | 0,0000            | 2        |
| 12     | 1,0000        | 0,0000         | 0,0000           | 11,2000         | 7,5000            | 3        |
| 13     | 1,0883        | 16,9100        | 0,0000           | 0,0000          | 0,0000            | 2        |
| 14     | 1,0000        | 0,0000         | 0,0000           | 6,2000          | 1,6000            | 3        |
| 15     | 1,0000        | 0,0000         | 0,0000           | 8,2000          | 2,5000            | 3        |
| 16     | 1,0000        | 0,0000         | 0,0000           | 3,5000          | 1,8000            | 3        |
| 17     | 1,0000        | 0,0000         | 0,0000           | 9,0000          | 5,8000            | 3        |
| 18     | 1,0000        | 0,0000         | 0,0000           | 3,2000          | 0,9000            | 3        |
| 19     | 1,0000        | 0,0000         | 0,0000           | 9,5000          | 3,4000            | 3        |
| 20     | 1,0000        | 0,0000         | 0,0000           | 2,2000          | 0,7000            | 3        |
| 21     | 1,0000        | 0,0000         | 0,0000           | 17,5000         | 11,2000           | 3        |
| 22     | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 23     | 1,0000        | 0,0000         | 0,0000           | 3,2000          | 1,6000            | 3        |
| 24     | 1,0000        | 0,0000         | 0,0000           | 8,7000          | 6,7000            | 3        |
| 25     | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 26     | 1,0000        | 0,0000         | 0,0000           | 3,5000          | 2,3000            | 3        |
| 27     | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 28     | 1,0000        | 0,0000         | 0,0000           | 0,0000          | 0,0000            | 3        |
| 29     | 1,0000        | 0,0000         | 0,0000           | 2,4000          | 0,9000            | 3        |
| 30     | 1,0000        | 0,0000         | 0,0000           | 10,6000         | 1,9000            | 3        |

Tabel 4.3<sup>[8]</sup>  
Data Saluran IEEE 30-Bus

| No | Dari Bus | Ke Bus | R(pu)  | X(pu)  | B(pu)  |
|----|----------|--------|--------|--------|--------|
| 1  | 1        | 2      | 0,0192 | 0,0575 | 0,0264 |
| 2  | 1        | 3      | 0,0452 | 0,1852 | 0,0204 |
| 3  | 2        | 4      | 0,0570 | 0,1737 | 0,0184 |
| 4  | 3        | 4      | 0,0132 | 0,0379 | 0,0042 |
| 5  | 2        | 5      | 0,0472 | 0,1983 | 0,0209 |
| 6  | 2        | 6      | 0,0581 | 0,1763 | 0,0187 |
| 7  | 4        | 6      | 0,0119 | 0,0414 | 0,0045 |
| 8  | 5        | 7      | 0,0460 | 0,1160 | 0,0102 |
| 9  | 6        | 7      | 0,0267 | 0,0820 | 0,0085 |
| 10 | 6        | 8      | 0,0120 | 0,0420 | 0,0045 |
| 11 | 6        | 9      | 0,0000 | 0,2082 | 0,0000 |
| 12 | 6        | 10     | 0,0000 | 0,5560 | 0,0000 |
| 13 | 9        | 11     | 0,0000 | 0,280  | 0,0000 |
| 14 | 9        | 10     | 0,0000 | 0,1100 | 0,0000 |
| 15 | 4        | 12     | 0,0000 | 0,2560 | 0,0000 |
| 16 | 12       | 13     | 0,0000 | 0,1400 | 0,0000 |
| 17 | 12       | 14     | 0,1231 | 0,2559 | 0,0000 |
| 18 | 12       | 15     | 0,0602 | 0,1304 | 0,0000 |
| 19 | 12       | 16     | 0,0945 | 0,1987 | 0,0000 |
| 20 | 14       | 15     | 0,2210 | 0,1997 | 0,0000 |
| 21 | 16       | 17     | 0,0824 | 0,1932 | 0,0000 |
| 22 | 15       | 18     | 0,1070 | 0,2185 | 0,0000 |
| 23 | 18       | 19     | 0,0639 | 0,1292 | 0,0000 |
| 24 | 19       | 20     | 0,0340 | 0,0680 | 0,0000 |
| 25 | 10       | 20     | 0,0936 | 0,2090 | 0,0000 |
| 26 | 10       | 17     | 0,0324 | 0,0845 | 0,0000 |
| 27 | 10       | 21     | 0,0348 | 0,0749 | 0,0000 |
| 28 | 10       | 22     | 0,0727 | 0,1499 | 0,0000 |
| 29 | 21       | 22     | 0,0116 | 0,0236 | 0,0000 |
| 30 | 15       | 23     | 0,1000 | 0,2020 | 0,0000 |
| 31 | 22       | 24     | 0,1150 | 0,1790 | 0,0000 |
| 32 | 23       | 24     | 0,1320 | 0,2700 | 0,0000 |
| 33 | 24       | 25     | 0,1885 | 0,3292 | 0,0000 |
| 34 | 25       | 26     | 0,2544 | 0,3800 | 0,0000 |
| 35 | 24       | 27     | 0,1093 | 0,2087 | 0,0000 |
| 36 | 28       | 27     | 0,0000 | 0,3960 | 0,0000 |
| 37 | 27       | 29     | 0,2198 | 0,4153 | 0,0000 |
| 38 | 27       | 30     | 0,3202 | 0,6027 | 0,0000 |
| 39 | 29       | 30     | 0,2399 | 0,4533 | 0,0000 |
| 40 | 8        | 28     | 0,0636 | 0,2000 | 0,0214 |
| 41 | 6        | 28     | 0,0169 | 0,0599 | 0,0065 |

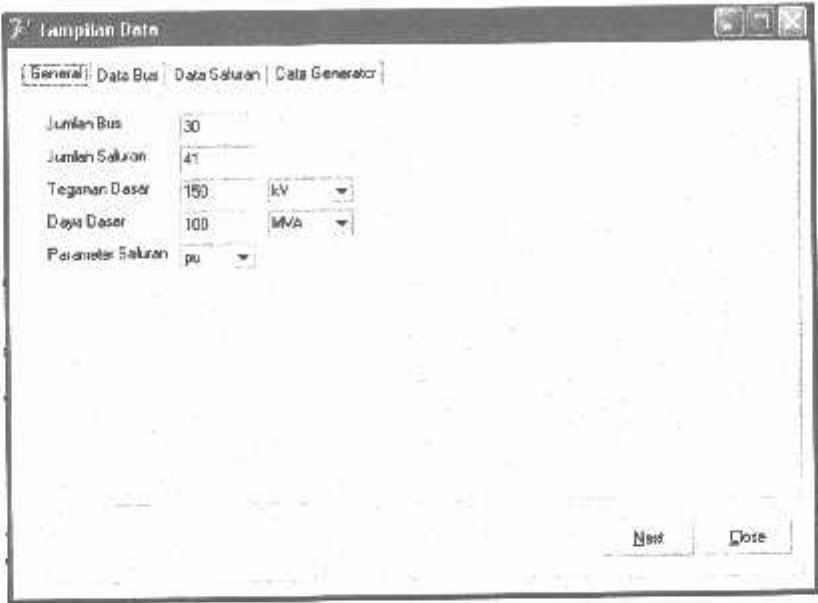
#### 4.2.2. Hasil Perhitungan Validasi Sistem IEEE 30-Bus Menggunakan Metode Tabu Search Algorithm.

Tampilan utama dari program komputer ini dapat dilihat pada gambar 4.3 dibawah ini:

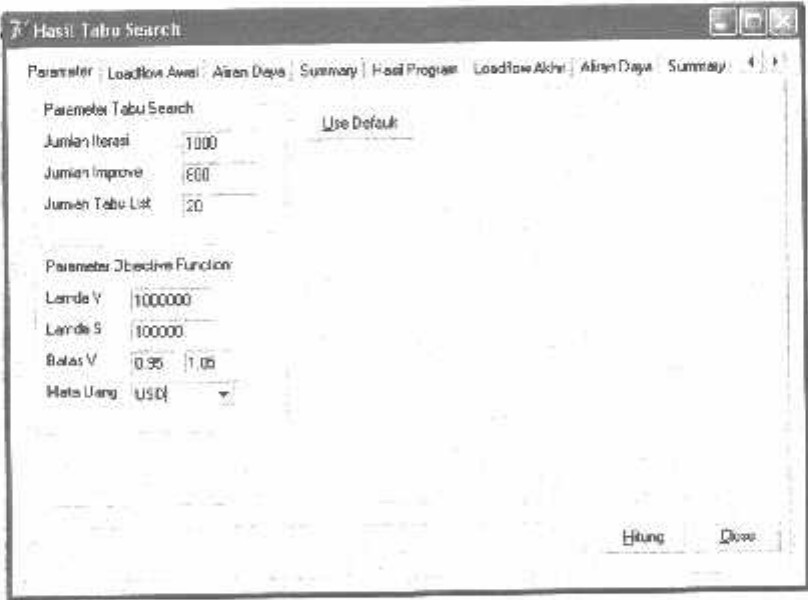


Gambar 4.3  
Tampilan Program Utama

Kemudian setelah itu tekan tombol open untuk membuka file yang sudah tersimpan.



Gambar 4.4  
Tampilan Input Data IEEE-30



Gambar 4.5  
Tampilan Parameter *Tabu Search Algorithm*

Setelah menetapkan parameter tersebut, maka komputasi dapat dilakukan tampilan dari perhitungan validasi dapat dilihat pada gambar di bawah ini:

| Bus | absV [pu] | sudV [deg] | Pg [MW] | Qg [MVAR] | PL [MW] | SL [MVAR] | Sups [pu] | Type Bus |
|-----|-----------|------------|---------|-----------|---------|-----------|-----------|----------|
| 1   | 1.0500    | 0.0000     | 176.200 | 4.275     | 0.000   | 0.000     | 0.000     | 1        |
| 2   | 1.0800    | 0.02687    | 48.750  | 12.000    | 22.700  | 12.700    | 0.000     | 2        |
| 3   | 1.02614   | -0.06233   | 0.000   | 0.000     | 2.400   | 1.200     | 0.000     | 3        |
| 4   | 1.02253   | -0.07431   | 0.000   | 0.000     | 7.600   | 1.600     | 0.000     | 3        |
| 5   | 1.01200   | -0.05304   | 21.440  | 1.649     | 12.200  | 19.000    | 0.000     | 2        |
| 6   | 1.01813   | -0.06348   | 0.000   | 0.000     | 0.000   | 0.000     | 0.000     | 3        |
| 7   | 1.00677   | -0.07919   | 0.000   | 0.000     | 22.800  | 10.900    | 0.000     | 3        |
| 8   | 1.02000   | 0.08859    | 21.950  | 35.998    | 30.000  | 30.000    | 0.000     | 2        |
| 9   | 1.02771   | -0.12214   | 0.000   | 0.000     | 0.000   | 0.000     | 0.000     | 3        |
| 10  | 1.00520   | -0.15035   | 0.000   | 0.000     | 5.000   | 2.000     | 0.000     | 3        |
| 11  | 1.08200   | 0.08990    | 12.420  | 28.385    | 0.000   | 0.000     | 0.000     | 2        |
| 12  | 1.02741   | -0.15015   | 0.000   | 0.000     | 11.200  | 7.500     | 0.000     | 3        |
| 13  | 1.06700   | 0.13480    | 12.028  | 32.265    | 0.000   | 0.000     | 0.000     | 2        |
| 14  | 1.01070   | -0.16502   | 0.000   | 0.000     | 6.700   | 1.600     | 0.000     | 3        |
| 15  | 1.00495   | -0.16653   | 0.000   | 0.000     | 8.200   | 2.500     | 0.000     | 3        |

(a) Hasil Loadflow

**Hasil Setting Program**

| No | Bus | absP (pu) | Pg (MW) | Qg (MVAR) |
|----|-----|-----------|---------|-----------|
| 1  | 1   | 1.05000   | 176.200 | 0.00      |
| 2  | 2   | 1.03600   | 48.75   | 0.00      |
| 3  | 5   | 1.01200   | 21.44   | 0.00      |
| 4  | 8   | 1.02000   | 21.95   | 0.00      |
| 5  | 11  | 1.06200   | 12.42   | 0.00      |
| 6  | 13  | 1.06700   | 12.02   | 0.00      |

**Biaya Pembangkitan**

| No | Bus | Biaya PLN (Rp) | Biaya TS (Rp) |
|----|-----|----------------|---------------|
| 1  | 1   | 117.91         | 210.36        |
| 2  | 2   | 156.71         | 125.90        |
| 5  | 5   | 401.56         | 306.74        |
| 8  | 8   | 123.97         | 75.36         |
| 11 | 11  | 61.83          | 41.12         |
| 13 | 13  | 57.86          | 38.67         |

**Rekapitulasi Biaya**

|                      |        |            |                      |        |            |
|----------------------|--------|------------|----------------------|--------|------------|
| Sebelum Optimalisasi | 321.85 | US \$/hour | Setelah Optimalisasi | 113.72 | US \$/hour |
| Setelah Optimalisasi | 802.13 | US \$/hour |                      |        |            |

(b) Hasil Perhitungan Biaya Pembangkitan

Gambar 4.6  
Tampilan Hasil Validasi IEEE 30-Bus

Untuk perbandingan nilai tegangan dan keluaran daya aktif antara hasil program dengan hasil referensi<sup>[5]</sup> dapat dilihat pada tabel 4.4 di bawah ini:

Tabel 4.4  
Perbandingan Nilai Tegangan dan Keluaran Daya aktif  
Antara Hasil Program Dengan Hasil Referensi

| No<br>Unit | No<br>Bus | Hasil Referensi        |                        |                 | Hasil program          |                        |                 |
|------------|-----------|------------------------|------------------------|-----------------|------------------------|------------------------|-----------------|
|            |           | V <sub>G</sub><br>(pu) | P <sub>G</sub><br>(MW) | Biaya<br>(\$/h) | V <sub>G</sub><br>(pu) | P <sub>G</sub><br>(MW) | Biaya<br>(\$/h) |
| 1          | 1         | 1,0500                 | 175,94                 | 209,82          | 1,05000                | 176,200                | 210,35          |
| 2          | 2         | 1,0300                 | 48,75                  | 128,92          | 1,03800                | 48,75                  | 126,90          |
| 3          | 5         | 1,0120                 | 21,44                  | 308,53          | 1,01200                | 21,44                  | 308,74          |
| 4          | 8         | 1,0200                 | 21,95                  | 75,35           | 1,02000                | 21,95                  | 75,36           |
| 5          | 11        | 1,0820                 | 12,42                  | 41,13           | 1,08200                | 12,42                  | 41,12           |
| 6          | 13        | 1,0670                 | 12,02                  | 39,67           | 1,06700                | 12,02                  | 39,67           |
| TOTAL      |           | 6,2610                 | 292,79                 | 802,06          | 6,26900                | 292,78                 | 802,13          |

Sedangkan perbandingan biaya total antara hasil program dengan hasil referensi, adalah biaya total hasil program sebesar 802,13 \$/h, sedangkan biaya total hasil referensi sebesar 802,06 \$/h.

Sehingga kesalahan perhitungan untuk tegangan sebesar 0,1%, keluaran daya aktif sebesar 0,003 %, dan untuk biaya pembangkitan sebesar 0,008%.

### 4.3. Data Sistem Jawa Timur dan Bali

#### 4.3.1. Data pembangkitan Termis Pada Sistem Jawa Timur dan Bali<sup>[4]</sup>

Pembangkitan termal yang berada pada pengawasan PT. PLN(Persero) P3B Area IV yang meliputi Wilayah Jawa Timur dan Bali Berjumlah 10 Buah Pembangkit yang terdiri dari 2 PLTGU, 4 PLTU, 3 PLTG dan 1 PLTD. Pada skripsi ini akan membahas pembangkitan termal yang berada pada sub-sistem Paiton dan Bali untuk mempermudah perhitungan, pembangkit termal yang

dibahas dalam skripsi ini adalh PLTU Paiton 1-2, PLTG Gilimanuk, PLTG Pesanggaran, dan PLTD Pesanggaran.

Untk data dari bentuk karakteristik semua unit pembangkit yang dibahas dalam skripsi ini beserta kapasitasnya juga biaya bahan bakar yang digunakan dalam perhitungan adalah berdasarkan data dari PT,PLN(Persero) Indonesia Power dan PT,PLN(Persero) PJB.

Tabel 4.5<sup>[4]</sup>  
 Parameter Unit termal

| Nama Pembangkit  | A          | B        | C          | $P_g^{min}$<br>MW | $P_g^{max}$<br>MW |
|------------------|------------|----------|------------|-------------------|-------------------|
| PLTGU Paiton 1-2 | 6,180000   | 1306,150 | 388144,168 | 150               | 740,0             |
| PLTG Gilimanuk   | 1,406900   | 1599,000 | 87435,000  | 50                | 133,8             |
| PLTD Pesanggaran | 14,293200  | 1670,000 | 88960,000  | 21                | 75,0              |
| PLTG Pesanggaran | 113,900000 | 5297,000 | 139560,000 | 15                | 125,5             |

Untuk persamaan biaya pembangkitan dari masing-masing unit pembangkit yang berdasarkan data-data di ats adalah sebagai berikut:

Tabel 4.6<sup>[4]</sup>  
 Persamaan Biaya Pembangkitan

| No | Nama Pembangkit  | Persamaan Biaya Pembangkitan                 |
|----|------------------|----------------------------------------------|
| 1  | PLTU Paiton 1-2  | $F= 6,180000P^2 + 1306,150P + 388144,168$    |
| 2  | PLTG Gilimanuk   | $F = 1,406900P^2 + 1599,000P + 87435,000$    |
| 3  | PLTD Pesanggaran | $F = 14,293200P^2 + 1670,000P + 88960,000$   |
| 4  | PLTG Pesanggaran | $F = 113,900000P^2 + 5297,000P + 139560,000$ |

Tabel 4.8<sup>[10]</sup>  
 Data Pembangkitan dan Pembebanan Sub-Sistem 150 kV Paiton-Bali  
 Tanggal 30 Maret 2005 Pukul 18.00 WIB

| No Bus | Tegangan (pu) | P <sub>Gm</sub> (MW) | Q <sub>Gm</sub> (MVAR) | P <sub>Load</sub> (MW) | Q <sub>Load</sub> (MVAR) | Tipe Bus |
|--------|---------------|----------------------|------------------------|------------------------|--------------------------|----------|
| 1      | 1,03333       | 0,0000               | 0,000                  | 11,700                 | 5,700                    | 1        |
| 2      | 1,00000       | 0,0000               | 0,0000                 | 16,500                 | 6,500                    | 3        |
| 3      | 1,00000       | 0,0000               | 0,0000                 | 17,000                 | 6,700                    | 3        |
| 4      | 1,00000       | 0,0000               | 0,0000                 | 39,300                 | 19,800                   | 3        |
| 5      | 1,00000       | 0,0000               | 0,0000                 | 39,800                 | 17,500                   | 3        |
| 6      | 1,00000       | 0,0000               | 0,0000                 | 23,400                 | 9,300                    | 3        |
| 7      | 1,00000       | 0,0000               | 47,000                 | 58,900                 | 30,300                   | 3        |
| 8      | 1,00000       | 0,0000               | 0,0000                 | 39,800                 | 21,300                   | 2        |
| 9      | 1,00000       | 0,0000               | 0,0000                 | 24,400                 | 5,800                    | 3        |
| 10     | 1,00000       | 0,0000               | 0,0000                 | 17,800                 | 7,400                    | 3        |
| 11     | 1,00000       | 0,0000               | 0,0000                 | 37,800                 | 14,800                   | 3        |
| 12     | 0,99260       | 99,900               | 0,0000                 | 5,7000                 | 2,000                    | 2        |
| 13     | 1,00000       | 0,0000               | 0,0000                 | 11,700                 | 4,600                    | 3        |
| 14     | 1,00000       | 0,0000               | 0,0000                 | 6,000                  | 2,800                    | 3        |
| 15     | 1,00000       | 0,0000               | 50,0000                | 69,300                 | 22,800                   | 3        |
| 16     | 1,00000       | 0,0000               | 0,0000                 | 4,0000                 | 0,500                    | 3        |
| 17     | 1,00000       | 0,0000               | 0,0000                 | 24,300                 | 8,800                    | 3        |
| 18     | 1,00000       | 0,0000               | 0,0000                 | 32,800                 | 13,300                   | 3        |
| 19     | 1,00000       | 0,0000               | 0,0000                 | 74,400                 | 28,900                   | 3        |
| 20     | 0,93600       | 31,400               | 0,0000                 | 0,0000                 | 0,0000                   | 2        |
| 21     | 0,93600       | 84,600               | 0,0000                 | 0,0000                 | 0,0000                   | 2        |
| 22     | 1,00000       | 0,0000               | 25,000                 | 51,400                 | 19,300                   | 3        |
| 23     | 1,00000       | 0,0000               | 25,000                 | 60,300                 | 17,700                   | 3        |
| 24     | 1,00000       | 0,0000               | 0,0000                 | 33,700                 | 9,600                    | 3        |
| 25     | 1,00000       | 0,0000               | 0,0000                 | 14,400                 | 7,000                    | 3        |

*Keiterangan:*     1: bus slack  
                          2: bus generator  
                          3: bus beban

untuk data saluran, jika sirkuit berjumlah 2, maka R, X dan B dari 2 sirkuit tersebut diparalel. Berikut adalah perhitungannya:

\* saluran 1 (hubungan bus 1-2):  $R_{paralel} = \frac{R_1 \cdot R_2}{R_1 + R_2} = \frac{0,0105 \cdot 0,0105}{0,0105 + 0,0105} = 0,0053$

untuk X dan B serta R berikutnya yang jumlah sirkuitnya 2 dilakukan hal yang sama, hasil selengkapnya dapat dilihat pada tabel 4.9.

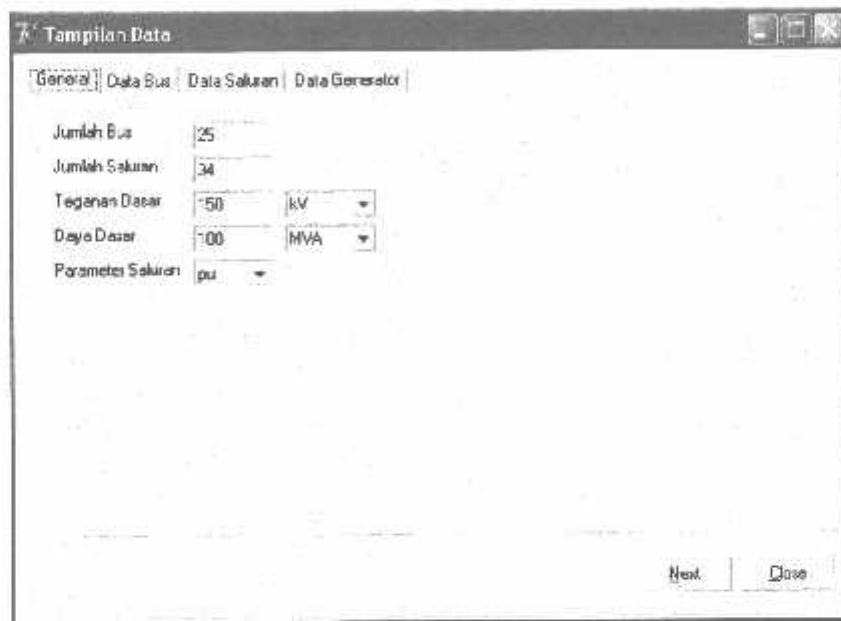
Tabel 4.9<sup>[10]</sup>  
Data Saluran Transmisi  
Sub-Sistem 150 kV Paiton-Bali

| No | Hubungan | Jumlah Sirkuit | R(pu)  | X(pu)  | B(pu)  |
|----|----------|----------------|--------|--------|--------|
| 1  | 1-2      | 2              | 0,0053 | 0,0179 | 0,0065 |
| 2  | 1-9      | 2              | 0,0048 | 0,0346 | 0,0251 |
| 3  | 2-3      | 1              | 0,0100 | 0,0340 | 0,0123 |
| 4  | 2-4      | 2              | 0,0079 | 0,0269 | 0,0097 |
| 5  | 4-5      | 2              | 0,0135 | 0,0460 | 0,0166 |
| 6  | 5-6      | 1              | 0,0167 | 0,0571 | 0,0206 |
| 7  | 5-7      | 1              | 0,0315 | 0,1077 | 0,0389 |
| 8  | 6-7      | 1              | 0,0148 | 0,0506 | 0,0183 |
| 9  | 7-8      | 1              | 0,0337 | 0,1151 | 0,0415 |
| 10 | 7-10     | 2              | 0,0104 | 0,0353 | 0,0128 |
| 11 | 7-11     | 1              | 0,0430 | 0,1469 | 0,0530 |
| 12 | 8-11     | 1              | 0,0177 | 0,0604 | 0,0218 |
| 13 | 9-10     | 2              | 0,0091 | 0,0310 | 0,0112 |
| 14 | 9-11     | 2              | 0,0966 | 0,0457 | 0,0340 |
| 15 | 11-12    | 2              | 0,0016 | 0,0035 | 0,0000 |
| 16 | 12-13    | 2              | 0,0116 | 0,0336 | 0,0124 |
| 17 | 12-17    | 1              | 0,0399 | 0,1314 | 0,0502 |
| 18 | 13-14    | 1              | 0,0270 | 0,0783 | 0,0288 |
| 19 | 13-15    | 1              | 0,0412 | 0,1194 | 0,0439 |
| 20 | 14-15    | 1              | 0,0142 | 0,0411 | 0,0151 |
| 21 | 15-16    | 1              | 0,0362 | 0,0693 | 0,0240 |
| 22 | 15-17    | 1              | 0,0556 | 0,1054 | 0,0369 |
| 23 | 15-18    | 1              | 0,0105 | 0,0304 | 0,0112 |
| 24 | 15-19    | 1              | 0,0105 | 0,0304 | 0,0112 |
| 25 | 15-22    | 1              | 0,0293 | 0,0561 | 0,0195 |
| 26 | 15-24    | 2              | 0,0059 | 0,0170 | 0,0063 |
| 27 | 16-17    | 1              | 0,0194 | 0,0371 | 0,0129 |
| 28 | 18-19    | 1              | 0,0045 | 0,0130 | 0,0048 |
| 29 | 19-20    | 1              | 0,0020 | 0,0044 | 0,0000 |
| 30 | 19-21    | 1              | 0,0020 | 0,0056 | 0,0000 |
| 31 | 19-22    | 1              | 0,0127 | 0,0243 | 0,0084 |
| 32 | 19-23    | 2              | 0,0037 | 0,0070 | 0,0025 |
| 33 | 23-24    | 2              | 0,0078 | 0,0149 | 0,0052 |
| 34 | 24-25    | 2              | 0,0103 | 0,0298 | 0,0110 |

#### 4.3.3. Hasil Perhitungan Biaya Operasi Pada Pembangkit Termal Sub-Sistem

##### Paiton-Bali Menggunakan Tabu Search Algorithm

Hasil perhitungan dari program ini adalah hasil perhitungan aliran daya dan operasi pembangkit yang optimal, setelah data yang ada dimasukkan ke dalam program, maka hasil perhitungan dapat dilihat dalam tabel-tabel di bawah ini:



The screenshot shows a software window titled "Tampilan Data" with four tabs: "General", "Data Bus", "Data Saluran", and "Data Generator". The "General" tab is active, displaying the following input fields:

|                   |     |     |
|-------------------|-----|-----|
| Jumlah Bus        | 25  |     |
| Jumlah Saluran    | 34  |     |
| Tegangan Dasar    | 50  | kV  |
| Daya Dasar        | 100 | MVA |
| Parameter Saluran | pu  |     |

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

Gambar 4.8  
Tampilan Input Data General Sub-Sistem Paiton-Bali

| Hasil Tabu Search |               |             |         |               |                |             |           |          |  |
|-------------------|---------------|-------------|---------|---------------|----------------|-------------|-----------|----------|--|
| Parameter         | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary   |          |  |
| Bus               | absV (pu)     | sudV (deg)  | Pg (MW) | Qg (MVAR)     | PL (MW)        | QL (MVAR)   | Saps (pu) | Type Bus |  |
| 1                 | 1.03233       | 0.0000      | 586.132 | -49.600       | 11.700         | 5.700       | 0.000     | 1        |  |
| 2                 | 1.01862       | -0.03639    | 0.000   | 0.000         | 16.500         | 6.500       | 0.000     | 3        |  |
| 3                 | 1.01511       | -0.04145    | 0.000   | 0.000         | 17.000         | 6.700       | 0.000     | 3        |  |
| 4                 | 1.00384       | -0.06456    | 0.000   | 0.000         | 29.300         | 19.900      | 0.000     | 3        |  |
| 5                 | 0.99608       | -0.15306    | 0.000   | 0.000         | 39.900         | 17.500      | 0.000     | 3        |  |
| 6                 | 0.99782       | -0.16809    | 0.000   | 0.000         | 23.400         | 9.300       | 0.000     | 3        |  |
| 7                 | 1.00632       | -0.20780    | 0.000   | 47.000        | 59.900         | 30.300      | 0.000     | 3        |  |
| 8                 | 0.98201       | -0.29962    | 0.000   | 0.000         | 39.900         | 21.300      | 0.000     | 3        |  |
| 9                 | 1.04926       | -0.11646    | 0.000   | 0.000         | 24.400         | 5.800       | 0.000     | 3        |  |
| 10                | 1.02652       | -0.18079    | 0.000   | 0.000         | 17.800         | 7.400       | 0.000     | 3        |  |
| 11                | 0.98634       | -0.32661    | 0.000   | 0.000         | 37.800         | 14.800      | 0.000     | 3        |  |
| 12                | 0.98266       | -0.33691    | 99.900  | 209.954       | 5.700          | 2.000       | 0.000     | 2        |  |
| 13                | 0.96869       | -0.34511    | 0.000   | 0.000         | 11.700         | 4.600       | 0.000     | 3        |  |

Gambar 4.9  
Tampilan Hasil Perhitungan Tegangan, Sudut Tegangan dan Daya Setiap Bus Sebelum Optimasi

| Hasil Tabu Search |               |             |         |               |                |             |           |          |  |
|-------------------|---------------|-------------|---------|---------------|----------------|-------------|-----------|----------|--|
| Parameter         | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary   |          |  |
| Bus               | absV (pu)     | sudV (deg)  | Pg (MW) | Qg (MVAR)     | PL (MW)        | QL (MVAR)   | Saps (pu) | Type Bus |  |
| 1                 | 1.05000       | 0.0000      | 589.633 | 2.480         | 11.700         | 5.700       | 0.000     | 1        |  |
| 2                 | 1.03438       | 0.03090     | 0.000   | 0.000         | 16.500         | 6.500       | 0.000     | 3        |  |
| 3                 | 1.03064       | -0.03581    | 0.000   | 0.000         | 17.000         | 6.700       | 0.000     | 3        |  |
| 4                 | 1.01750       | -0.07053    | 0.000   | 0.000         | 29.300         | 19.900      | 0.000     | 3        |  |
| 5                 | 1.00451       | -0.12601    | 0.000   | 0.000         | 39.900         | 17.500      | 0.000     | 3        |  |
| 6                 | 1.00219       | -0.15310    | 0.000   | 0.000         | 23.400         | 9.300       | 0.000     | 3        |  |
| 7                 | 1.00672       | 0.16610     | 0.000   | 14.571        | 59.900         | 30.300      | 0.000     | 3        |  |
| 8                 | 0.98530       | -0.23307    | 0.000   | 0.000         | 39.900         | 21.300      | 0.000     | 3        |  |
| 9                 | 1.05291       | 0.09366     | 0.000   | 0.000         | 24.400         | 5.800       | 0.000     | 3        |  |
| 10                | 1.02904       | 0.12327     | 0.000   | 0.000         | 17.800         | 7.400       | 0.000     | 3        |  |
| 11                | 0.99176       | -0.24886    | 0.000   | 0.000         | 37.800         | 14.800      | 0.000     | 3        |  |
| 12                | 0.99462       | -0.25404    | 122.127 | 105.905       | 5.700          | 2.000       | 0.000     | 2        |  |
| 13                | 0.97483       | -0.26436    | 0.000   | 0.000         | 11.700         | 4.600       | 0.000     | 3        |  |

Gambar 4.10  
Tampilan Perhitungan Aliran Daya Sebelum Optimasi

Hasil Tabu Search

Parameter | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary

| Bus | absV (pu) | sudV (deg) | Pg (MW) | Qg (MVAR) | PL (MW) | QL (MVAR) | Supr (pu) | Type Bus |
|-----|-----------|------------|---------|-----------|---------|-----------|-----------|----------|
| 1   | 1.05000   | 0.00000    | 503.533 | 2.481     | 11.700  | 5.700     | 0.000     | 1        |
| 2   | 1.03438   | -0.03090   | 0.000   | 0.000     | 16.500  | 0.500     | 0.000     | 3        |
| 3   | 1.03094   | -0.03581   | 0.000   | 0.000     | 17.000  | 6.700     | 0.000     | 3        |
| 4   | 1.01750   | -0.07093   | 0.000   | 0.000     | 39.300  | 19.800    | 0.000     | 3        |
| 5   | 1.00451   | -0.12501   | 0.000   | 0.000     | 39.800  | 17.500    | 0.000     | 3        |
| 6   | 1.00219   | -0.15310   | 0.000   | 0.000     | 23.400  | 9.300     | 0.000     | 3        |
| 7   | 1.00672   | -0.16610   | 0.000   | 14.571    | 58.900  | 30.300    | 0.000     | 3        |
| 8   | 0.98530   | -0.23307   | 0.000   | 0.000     | 39.800  | 21.300    | 0.000     | 3        |
| 9   | 1.05291   | -0.09366   | 0.000   | 0.000     | 24.400  | 5.600     | 0.000     | 3        |
| 10  | 1.02904   | -0.12527   | 0.000   | 0.000     | 17.800  | 7.400     | 0.000     | 3        |
| 11  | 0.99176   | -0.24606   | 0.000   | 0.000     | 37.600  | 14.900    | 0.000     | 3        |
| 12  | 0.99462   | -0.25404   | 122.127 | 105.905   | 5.700   | 2.000     | 0.000     | 2        |
| 13  | 0.97453   | -0.26436   | 0.000   | 0.000     | 11.700  | 4.600     | 0.000     | 3        |

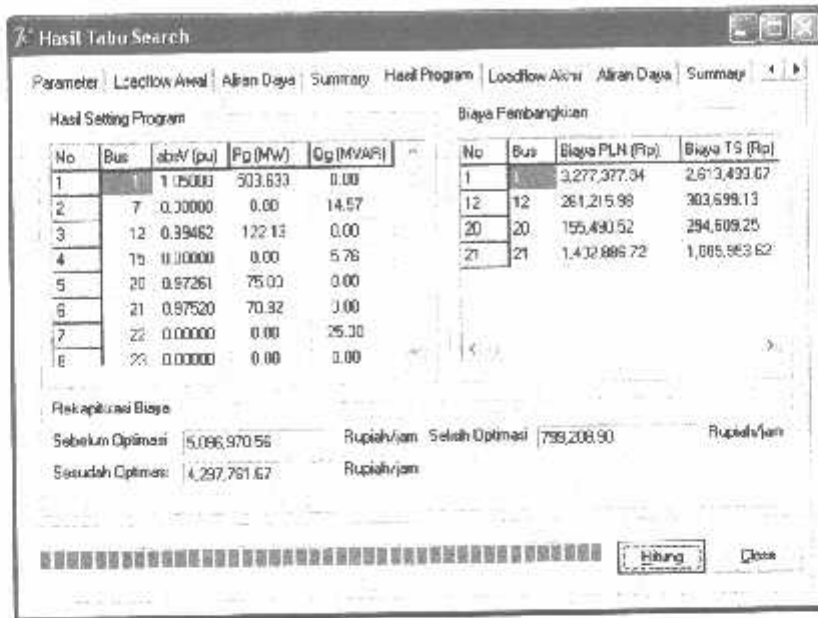
Gambar 4.11  
Tampilan Hasil Perhitungan tegangan, Sudut Tegangan dan Daya Setiap Bus Setelah Optimasi

Hasil Tabu Search

Parameter | Loadflow Awal | Aliran Daya | Summary | Hasil Program | Loadflow Akhir | Aliran Daya | Summary

| No | Der | Ke | P (MW)   | Q (MVAR) | Arus re (A) | Arus im (A) | Der | Ke | P (MW)   |
|----|-----|----|----------|----------|-------------|-------------|-----|----|----------|
| 1  |     | 2  | 198.052  | 35.137   | 1257.475    | 223.053     | 2   | 1  | -198.105 |
| 2  | 1   | 9  | 293.891  | -38.356  | 1665.910    | -243.528    | 9   | 1  | -290.066 |
| 3  | 2   | 3  | 17.030   | 4.179    | 108.875     | 30.300      | 3   | 2  | -17.000  |
| 4  | 2   | 4  | 162.575  | 19.294   | 1043.464    | 156.657     | 4   | 2  | -160.593 |
| 5  | 4   | 5  | 121.293  | -5.214   | 795.137     | 22.248      | 5   | 4  | -113.373 |
| 6  | 5   | 6  | 45.290   | -10.589  | 306.826     | -31.971     | 6   | 5  | -44.909  |
| 7  | 5   | 7  | 34.313   | -15.273  | 238.662     | -71.940     | 7   | 5  | -33.905  |
| 8  | 6   | 7  | 21.509   | -16.942  | 158.591     | -83.561     | 7   | 6  | -21.407  |
| 9  | 7   | 8  | 58.638   | -0.717   | 384.079     | 59.575      | 8   | 7  | -57.539  |
| 10 | 7   | 10 | -115.137 | -28.740  | 727.026     | -314.809    | 10  | 7  | 117.599  |
| 11 | 7   | 11 | 53.651   | -8.669   | 361.259     | 2.353       | 11  | 7  | -52.626  |
| 12 | 8   | 11 | 17.739   | 17.707   | 144.452     | -68.850     | 11  | 8  | -17.637  |
| 13 | 9   | 10 | 137.894  | 41.795   | 833.481     | 344.649     | 10  | 9  | 135.389  |

Gambar 4.12  
Tampilan hasil Perhitungan Aliran Daya Setelah Optimasi



Gambar 4.13  
Tampilan Hasil Perhitungan Total Biaya Sebelum dan Setelah Optimasi

Tabel 4.10  
 Hasil Perhitungan Tegangan, Sudut Tegangan Dan Daya Pada Setiap Bus  
 Sebelum Optimasi

| No | Tegangan<br>(pu) | Sudut<br>tegangan | Pembangkitan |         | Pembebanan |        |
|----|------------------|-------------------|--------------|---------|------------|--------|
|    |                  |                   | MW           | MVAR    | MW         | MVAR   |
| 1  | 1,03333          | 0,00000           | 586,192      | -49,660 | 11,700     | 5,700  |
| 2  | 1,01862          | -0,03638          | 0,000        | 0,000   | 16,500     | 6,500  |
| 3  | 1,01511          | -0,04145          | 0,000        | 0,000   | 17,000     | 6,700  |
| 4  | 1,00384          | -0,08456          | 0,000        | 0,000   | 39,300     | 19,800 |
| 5  | 0,99608          | -0,15306          | 0,000        | 0,000   | 39,800     | 17,500 |
| 6  | 0,99782          | -0,18808          | 0,000        | 0,000   | 23,400     | 9,300  |
| 7  | 1,00632          | -0,20780          | 0,000        | 47,000  | 58,900     | 30,300 |
| 8  | 0,98201          | -0,29962          | 0,000        | 0,000   | 39,800     | 21,300 |
| 9  | 1,04926          | -0,11646          | 0,000        | 0,000   | 24,400     | 5,800  |
| 10 | 1,02652          | -0,16078          | 0,000        | 0,000   | 17,800     | 7,400  |
| 11 | 0,98834          | -0,32661          | 0,000        | 0,000   | 37,800     | 14,800 |
| 12 | 0,99266          | -0,33691          | 99,900       | 209,554 | 5,700      | 2,000  |
| 13 | 0,96869          | -0,34511          | 0,000        | 0,000   | 11,700     | 4,600  |
| 14 | 0,94728          | -0,42828          | 0,000        | 0,000   | 6,000      | 2,800  |
| 15 | 0,93900          | -0,47032          | 0,000        | 50,000  | 69,300     | 22,800 |
| 16 | 0,94912          | -0,45352          | 0,000        | 0,000   | 4,000      | 0,500  |
| 17 | 0,95438          | -0,44242          | 0,000        | 0,000   | 24,800     | 8,800  |
| 18 | 0,93341          | -0,48073          | 0,000        | 0,000   | 32,800     | 13,300 |
| 19 | 0,93427          | -0,48093          | 0,000        | 0,000   | 76,400     | 28,900 |
| 20 | 0,93600          | -0,47987          | 31,400       | 22,446  | 0,000      | 0,000  |
| 21 | 0,93600          | -0,47549          | 0,000        | -1,141  | 0,000      | 0,000  |
| 22 | 0,93223          | -0,48853          | 0,000        | 25,000  | 51,400     | 19,300 |
| 23 | 0,93273          | -0,48475          | 0,000        | 25,000  | 60,300     | 17,700 |
| 24 | 0,93325          | -0,48186          | 0,000        | 0,000   | 33,700     | 9,600  |
| 25 | 0,92971          | -0,48609          | 0,000        | 0,000   | 14,400     | 7,000  |

Tabel 4.11  
 Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi

| Bus  |    | Daya     |          | Bus  |    | Daya     |         |
|------|----|----------|----------|------|----|----------|---------|
| Dari | Ke | MW       | MVAR     | Dari | Ke | MW       | MVAR    |
| 1    | 2  | 220,823  | 22,742   | 2    | 1  | -218,375 | -15,843 |
| 1    | 9  | 353,669  | -78,102  | 9    | 1  | -347,790 | 115,033 |
| 2    | 3  | 17,031   | 4,261    | 3    | 2  | -17,000  | -6,700  |
| 2    | 4  | 184,844  | 5,082    | 4    | 2  | -182,240 | 1,802   |
| 4    | 5  | 142,940  | -21,602  | 5    | 4  | -140,150 | 27,790  |
| 5    | 6  | 55,614   | -20,273  | 6    | 5  | -55,038  | 18,150  |
| 5    | 7  | 44,735   | -25,017  | 7    | 5  | -43,958  | 19,876  |
| 6    | 7  | 31,638   | -27,450  | 7    | 6  | -31,391  | 24,617  |
| 7    | 8  | 79,211   | -25,017  | 8    | 7  | -77,122  | 1,450   |
| 7    | 10 | -141,288 | -27,450  | 10   | 7  | 143,355  | 18,381  |
| 7    | 11 | 78,526   | -2,520   | 11   | 7  | -75,893  | 9,713   |
| 8    | 11 | 37,322   | -14,011  | 11   | 8  | -36,988  | 19,658  |
| 9    | 10 | 163,450  | 31,187   | 10   | 9  | -161,155 | -25,782 |
| 9    | 11 | 159,940  | -152,020 | 11   | 9  | -118,204 | 164,700 |
| 11   | 12 | 193,285  | -208,871 | 12   | 11 | -191,959 | 211,773 |
| 12   | 13 | 207,486  | -8,313   | 13   | 12 | -202,412 | 7,414   |
| 12   | 17 | 78,673   | 4,094    | 17   | 12 | -76,134  | -5,250  |
| 13   | 14 | 96,425   | -5,410   | 14   | 13 | -93,748  | 7,888   |
| 13   | 15 | 94,287   | -6,604   | 15   | 13 | -90,381  | 9,934   |
| 14   | 15 | 87,748   | -10,688  | 15   | 14 | -86,515  | 11,568  |
| 15   | 16 | -22,530  | -3,869   | 16   | 15 | 22,740   | -0,008  |
| 15   | 17 | -24,074  | -3,921   | 17   | 15 | 24,440   | 2,000   |
| 15   | 18 | 32,192   | 5,327    | 18   | 15 | -32,064  | -6,919  |
| 15   | 19 | 31,904   | 2,758    | 19   | 15 | -31,781  | -4,367  |
| 15   | 22 | 27,076   | -4,269   | 22   | 15 | -26,830  | 1,325   |
| 15   | 24 | 63,029   | 9,673    | 24   | 15 | -62,756  | -9,991  |
| 16   | 17 | -26,740  | -0,492   | 17   | 16 | 26,830   | -1,550  |

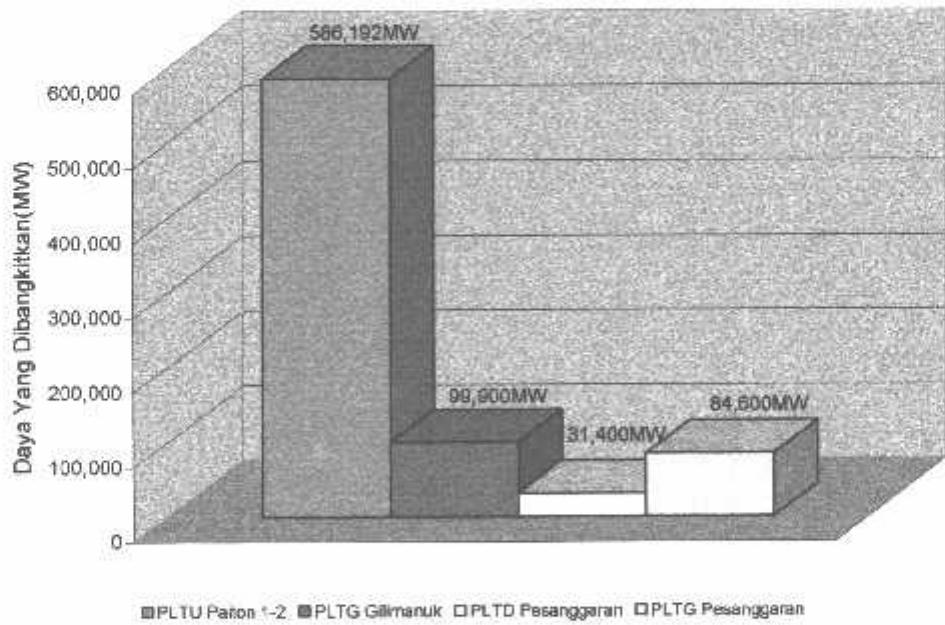
Tabel 4.12  
Total Pembangkitan, Pembebanan, dan Rugi-rugi Sebelum Optimasi

| Pembangkitan |             | Pembebanan |             | Rugi-rugi |            |
|--------------|-------------|------------|-------------|-----------|------------|
| 802,092MW    | 328,199MVAR | 716,900MW  | 282,400MVAR | 85,192MW  | 45,799MVAR |

Tabel 4.13  
Hasil Perhitungan Daya Yang dibangkitkan dan Biaya Operasi Sebelum Optimasi

| No    | Nama Pembangkit  | Daya Yang Dibangkitkan (MW) | Biaya Operasi (Rp/Jam) |
|-------|------------------|-----------------------------|------------------------|
| 1     | PLTU Paiton1-2   | 586,192                     | 3.277.377,34           |
| 2     | PLTG Gilimanuk   | 99,900                      | 261.215,98             |
| 3     | PLTD Pesanggaran | 31,400                      | 155.490,52             |
| 4     | PLTG Pesanggaran | 84,600                      | 1.402.886,72           |
| Total |                  | 802,092                     | 5.096.970,56           |

Grafik 4.1  
Hasil Perhitungan Daya yang Dibangkitkan Sebelum Optimasi



Tabel 4.14  
Hasil Perhitungan Tegangan, Sudut Tegangan dan Daya Pada Setiap Bus  
Setelah Optimasi

| No | Tegangan<br>(pu) | Sudut<br>tegangan | Pembangkitan |         | Pembebanan |        |
|----|------------------|-------------------|--------------|---------|------------|--------|
|    |                  |                   | MW           | MVAR    | MW         | MVAR   |
| 1  | 1,05000          | 0,00000           | 503,633      | 2,481   | 11,700     | 5,700  |
| 2  | 1,03438          | -0,03090          | 0,000        | 0,000   | 16,500     | 6,500  |
| 3  | 1,03094          | -0,03581          | 0,000        | 0,000   | 17,000     | 6,700  |
| 4  | 1,01750          | -0,07093          | 0,000        | 0,000   | 39,300     | 19,800 |
| 5  | 1,00451          | -0,12601          | 0,000        | 0,000   | 39,800     | 17,500 |
| 6  | 1,00219          | -0,15310          | 0,000        | 0,000   | 23,400     | 9,300  |
| 7  | 1,00672          | -0,16610          | 0,000        | 14,571  | 58,900     | 30,300 |
| 8  | 0,98530          | -0,23307          | 0,000        | 0,000   | 39,800     | 21,300 |
| 9  | 1,05291          | -0,09366          | 0,000        | 0,000   | 24,400     | 5,800  |
| 10 | 1,02904          | -0,12927          | 0,000        | 0,000   | 17,800     | 7,400  |
| 11 | 0,99176          | -0,24686          | 0,000        | 0,000   | 37,800     | 14,800 |
| 12 | 0,99462          | -0,25404          | 122,127      | 105,905 | 5,700      | 2,000  |
| 13 | 0,97453          | -0,26436          | 0,000        | 0,000   | 11,700     | 4,600  |
| 14 | 0,96623          | -0,33898          | 0,000        | 0,000   | 6,000      | 2,800  |
| 15 | 0,96425          | -0,37570          | 0,000        | 5,757   | 69,300     | 22,800 |
| 16 | 0,96992          | -0,36085          | 0,000        | 0,000   | 4,000      | 0,500  |
| 17 | 0,97274          | -0,35079          | 0,000        | 0,000   | 24,800     | 8,800  |
| 18 | 0,96558          | -0,38555          | 0,000        | 0,000   | 32,800     | 13,300 |
| 19 | 0,96928          | -0,38575          | 0,000        | 0,000   | 76,400     | 28,900 |
| 20 | 0,97261          | -0,38309          | 75,000       | 39,540  | 0,000      | 0,000  |
| 21 | 0,97520          | -0,38321          | 70,820       | 77,807  | 0,000      | 0,000  |
| 22 | 0,96444          | -0,39285          | 0,000        | 25,000  | 51,400     | 19,300 |
| 23 | 0,96452          | -0,38842          | 0,000        | 0,000   | 60,300     | 17,700 |
| 24 | 0,96195          | -0,38590          | 0,000        | 0,000   | 33,700     | 9,600  |
| 25 | 0,95854          | -0,38988          | 0,000        | 0,000   | 14,400     | 7,000  |

Tabel 4.15  
Hasil Perhitungan Aliran daya Antar Saluran Setelah Optimasi

| Bus  |    | Daya     |          | Bus  |    | Daya     |         |
|------|----|----------|----------|------|----|----------|---------|
| Dari | Ke | MW       | MVAR     | Dari | Ke | MW       | MVAR    |
| 1    | 2  | 198,052  | 35,137   | 2    | 1  | -196,105 | -29,972 |
| 1    | 9  | 293,881  | -38,356  | 9    | 1  | -290,066 | 60,308  |
| 2    | 3  | 17,030   | 4,178    | 3    | 2  | -17,000  | -6,700  |
| 2    | 4  | 162,575  | 19,294   | 4    | 2  | -160,593 | -14,587 |
| 4    | 5  | 121,293  | -5,214   | 5    | 4  | -119,373 | 8,363   |
| 5    | 6  | 45,260   | -10,589  | 6    | 5  | -44,909  | 7,642   |
| 5    | 7  | 34,313   | -15,273  | 7    | 5  | -33,905  | 8,800   |
| 6    | 7  | 21,509   | -16,942  | 7    | 6  | -21,407  | 13,597  |
| 7    | 8  | 58,688   | -0,717   | 8    | 7  | -57,539  | -3,592  |
| 7    | 10 | -116,137 | -28,740  | 10   | 7  | 117,599  | 31,047  |
| 7    | 11 | 53,861   | -8,669   | 11   | 7  | -52,626  | 2,305   |
| 8    | 11 | 17,739   | -17,707  | 11   | 8  | -17,637  | 13,794  |
| 9    | 10 | 137,094  | 41,795   | 10   | 9  | -135,399 | -38,449 |
| 9    | 11 | 128,572  | -107,902 | 11   | 9  | -104,719 | 112,073 |
| 11   | 12 | 137,181  | -142,973 | 12   | 11 | -136,543 | 144,370 |
| 12   | 13 | 182,290  | -34,044  | 13   | 12 | -178,268 | 32,819  |
| 12   | 17 | 70,680   | -6,421   | 17   | 12 | -68,664  | 3,343   |
| 13   | 14 | 84,339   | -18,139  | 14   | 13 | -82,250  | 18,775  |
| 13   | 15 | 82,228   | -19,280  | 15   | 13 | -79,196  | 19,816  |
| 14   | 15 | 76,250   | -21,575  | 15   | 14 | -75,304  | 21,500  |
| 15   | 16 | -18,921  | -0,091   | 16   | 15 | 19,063   | -4,128  |
| 15   | 17 | -20,431  | -0,137   | 17   | 15 | 20,687   | -6,300  |
| 15   | 18 | 25,703   | -13,976  | 18   | 15 | -25,609  | 12,161  |
| 15   | 19 | 22,744   | -24,692  | 19   | 15 | -22,622  | 22,950  |
| 15   | 22 | 22,306   | -13,539  | 22   | 15 | -22,106  | 10,295  |
| 15   | 24 | 53,799   | -5,924   | 24   | 15 | -53,614  | 5,290   |
| 16   | 17 | -23,063  | 3,628    | 17   | 16 | 23,177   | -5,843  |

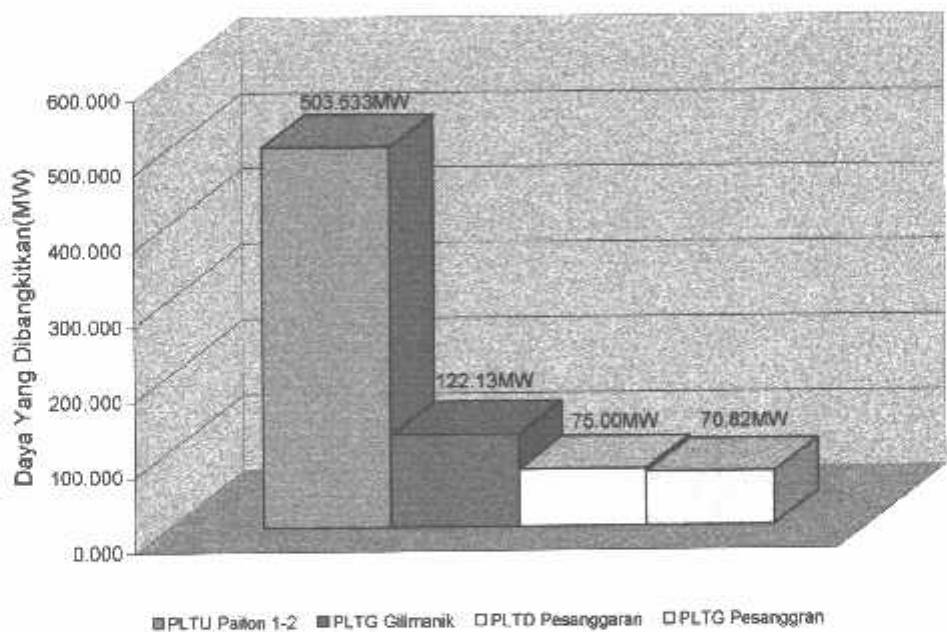
Tabel 4.16  
Total Pembangkitan, Pembebanan, dan Rugi-rugi Setelah Optimasi

| Pembangkitan |             | Pembebanan |             | Rugi-rugi |            |
|--------------|-------------|------------|-------------|-----------|------------|
| 771,580 MW   | 271,061MVAR | 716,900 MW | 245,380MVAR | 54,680 MW | 25,681MVAR |

Tabel 4.17  
 Hasil Perhitungan Daya Yang dibangkitkan dan Biaya Operasi Setelah Optimasi

| No    | Nama Pembangkit  | Daya Yang Dibangkitkan (MW) | Biaya Operasi (Rp/Jam) |
|-------|------------------|-----------------------------|------------------------|
| 1     | PLTU Paiton1-2   | 503,633                     | 2.613.499,67           |
| 2     | PLTG Gilimanuk   | 122,13                      | 303.699,13             |
| 3     | PLTD Pesanggaran | 75,00                       | 294.609,25             |
| 4     | PLTG Pesanggaran | 70,82                       | 1.085.953,62           |
| Total |                  | 771,583                     | 4.297.761,67           |

Grafik 4.2  
 Hasil Perhitungan Daya Yang Dibangkitkan Setelah Optimasi



#### 4.3.4. Perbandingan Hasil Perhitungan Sebelum dan Sesudah Optimasi Dengan Program *Optimal Power Flow* Menggunakan Metode *Tabu Search Algorithm*.

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

## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Setelah dilakukan perhitungan menggunakan metode *Tabu Search Algorithm*, maka dapat diambil kesimpulan sebagai berikut:

1. Dari hasil perhitungan *loadflow* awal(sebelum optimasi)menggunakan metode Newton Rapshon, maka total daya yang dibangkitkan sebesar 802,092 MW, total pembebanan 716,900 MW dan rugi-ruginya sebesar 85,192 MW.
2. Setelah dilakukan optimasi menggunakan metode TS maka *loadflow* akhirnya (setelah optimasi), untuk total daya pembangkitan sebesar 771,580 MW, total pembebanan sebesar 716,900 MW dan rugi-ruginya sebesar 54,680 MW.
3. Total biaya pembangkitan sebelum optimasi sebesar Rp. 5.096.970,56 perjam, sedangkan setelah optimasi biaya pembangkitan sebesar Rp. 4.297.761,67 perjam sehingga selisih antara biaya produksi sebelum dan setelah optimasi adalah sebesar Rp. 799.208,90 perjam atau terjadi penghematan sebesar 18,5%

### 3.2. Saran

Diharapkan penelitian selanjutnya diharapkan dapat memperluas pembahasan, minimal mengganti area penelitian, serta membahas kapasitas saluran, biaya *start up* dan *shut down*, Serta masalah *Spinning Reserve* sehingga hasil yang diperoleh dari penelitian lebih lengkap dan akurat, karena menyertakan seluruh komponen yang berhubungan dengan masalah optimalisasi dalam penyaluran daya listrik.

## DAFTAR PUSTAKA

- [1]. Wood, Allen J, Power "*Generation Operation And Control*", 2<sup>nd</sup> ed, John Wiley & Sons, Inc, New York, 1996.
- [2]. A.D, Jose, P. Pablo, K. Hussein, "*Optimal Power Flow Based on Tabu Search For Mesh Distribution Network*", Univ. de Zaragoza Spain, 14<sup>th</sup> PSCC, Sevilla, 24-28 June 2002
- [3]. Marsudi, Djiteng, "*Operasi Sistem Tenaga Listrik*", Balai Penerbit & Humas ISTN, Jakarta, 1990.
- [4]. Syafena, Jatri, Skripsi, "*Econimic Dispatch Dengan Optimasi Daya Aktif dan Reaktif Pada pembangkit Thermal di Area IV*", 2002
- [5]. Bakirtzis, Anastasios G, Pandel N. Biskas, Christoforos E. Zoumas, Vasilios Petridis, "*Optimal Power Flow by Enhanced Genetic Algorithm*", IEEE Transactions On Power System, vol.17. No.2. May 2002.
- [6]. Kulworawanichpong T, Sujitjorn S, "*Optimal Power Flow Using Tabu Search*", IEEE Power Engineering Review, June 2002.
- [7]. F. Glover, "Tabu Search(part I)", ORSA Journal Computing, No. 1, pp. 190-206.
- [8]. Nagrath, I.J. And Kothari D.P., "*Modern Power System Analysis*", New Delhi: Tata McGraw-Hill Publishing Company Limited.
- [9]. Juliato, Patria, Skripsi, "*Optimal Power Flow Menggunakan Metode Enhanced Genetic Algorithm Pada Sub-Sistem 150 kV Paiton-Bali*", 2004.
- [10]. Data Load Flow Region Jawa Bali, Sub-Sistem 150 kV Paiton dan Bali, Tanggal 30 Maret 2005 Pukul 18.00 WIB, PT. PLN(Persero) P3B Region Jawa Timur dan Bali, Waru.

# LAMPIRAN

---



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

---

**BERITA ACARA UJIAN SKRIPSI**  
**FAKULTAS TEKNOLOGI INDUSTRI**

**Nama Mahasiswa** : DESTA DWI MULYANTO  
**N.I.M.** : 00.12.025  
**Jurusan** : Teknik Elektro S-1  
**Konsentrasi** : Teknik Energi Listrik  
**Judul Skrip** : **OPTIMAL POWER FLOW MENGGUNAKAN  
METODE TABU SEARCH ALGORITHM  
PADA SALURAN TRANSMISI 150 kV  
SUB-SISTEM PAITON-BALI**

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

**Hari** : Jum'at  
**Tanggal** : 07 Oktober 2005  
**Dengan Nilai** : 84,1 (A) *Bm*

**Panitia Ujian**

  
( Ir. Mochtar Asroni, MSME )  
**Ketua**

  
( Ir. F. Yudi Limpraptono, MT. )  
**Sekretaris**

**Anggota Penguji**

  
( Ir. Choirul Saleh, MT. )  
**Penguji Pertama**

  
( Ir. M. Abdul Hamid, MT )  
**Penguji Kedua**

---



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

---

## LEMBAR BIMBINGAN SKRIPSI

Nama Mahasiswa : DESTA DWI MULYANTO  
N.I.M. : 00.12.025  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Teknik Energi Listrik  
Judul Skrip : *OPTIMAL POWER FLOW MENGGUNAKAN  
METODE TABU SEARCH ALGORITHM  
PADA SALURAN TRANSMISI 150 kV  
SUB-SISTEM PAITON-BALI*

Tanggal Mengajukan Skripsi : 18 Agustus 2005  
Tanggal Menyelesaikan Skripsi : 03 Oktober 2005  
Dosen Pembimbing : Ir.Yusuf Ismail Nakhoda, MT  
Telah di evaluasi dengan nilai : 88 (Delapan Puluh Delapan) 

Malang, Oktober 2005

Mengetahui,  
Ketua Jurusan Teknik Elektro

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

Disetujui,  
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT  
NIP.P.101 880 0189

---



### FORMULIR BIMBINGAN SKRIPSI

Nama : DESTA DWI MULYANTO  
Nim : 00.12.025  
Masa Bimbingan : 07 September 2005 s/d 07 Maret 2006  
Judul Skripsi : **OPTIMAL POWER FLOW MENGGUNAKAN  
METODE TABU SEARCH PADA SUB-SISTEM  
150 kV PAITON-BALI**

| No. | Tanggal    | Uraian                                                      | Parap Pembimbing |
|-----|------------|-------------------------------------------------------------|------------------|
| 1.  | 18-08-2005 | Konsultasi Bab I, II                                        |                  |
| 2.  | 21-08-2005 | Konsultasi perbaikan bab I & II sekaligus Pengajuan bab III |                  |
| 3.  | 28-08-2005 | Konsultasi perbaikan bab III & melakukan test uji validasi  |                  |
| 4.  | 30-08-2005 | Cek kembali error hasil uji validasi dengan journal         |                  |
| 5.  | 03-09-2005 | Konsultasi hasil error uji validasi dan pengajuan bab IV    |                  |
| 6.  | 04-09-2005 | Konsultasi perbaikan bab IV dan pengajuan makalah seminar   |                  |
| 7.  | 09-09-2005 | Periksa kembali Selisih Biaya program Apakah Sudah Optimal  |                  |
| 8.  | 12-09-2005 | Perbaiki Dan Periksa Lagi Masalah Selisih Total Biaya       |                  |
| 9.  | 20-09-2005 | Tanda tangan makalah seminar hasil                          |                  |
| 10. | 25-09-2005 | Konsultasi seluruh bab dan tanda tangan skripsi             |                  |

Malang, Oktober 2005  
Dosen Pembimbing,

Ir. Yusuf Ismail Nakhoda, MT

Form.S-4b



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

Jl. Suningrat No. 45 Taman Sidoarjo 61257

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

Kotak Pos : 4119 SBS  
Bank : Bank Mandiri

Nomor : 730/RITB/2005.  
Surat Sdr. No : ITN-999/IIITA/2/2005.  
Lampiran : 1 (satu) lampiran.  
Perihal : Ijin Survey/ Pengambilan Data.

Kepada

Yth. Dekan Fakultas Teknik,  
Institut Teknologi Nasional Malang  
Di  
M A L A N G

Menunjuk surat Saudara nomor : ITN-999/IIITA/2/2005 tanggal 14 Maret 2005 perihal : Survey  
Permintaan Data, dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada  
Mahasiswa Saudara, bernama :

• DESTA DWI M Nim : 00.12.025.

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

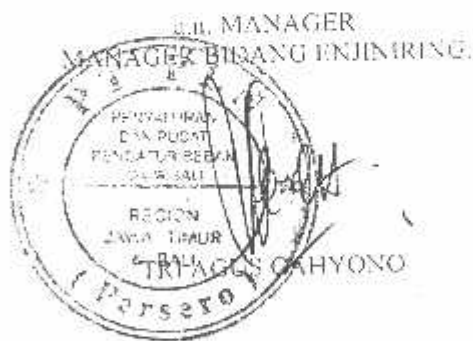
1. Mahasiswa tersebut diatas supaya mengisi dan menanda tangani Surat Pernyataan 1 (satu) lembar  
bermeterai Rp. 5.000,-
2. Mahasiswa yang bersangkutan agar mematuhi peraturan/keputusan yang berlaku di PT. PLN  
(PERSERO) sehingga faktor-faktor kerawasaan harus benar-benar diutamakan,
3. Semua biaya perjalanan, penginapan, makan dan lain sebagainya tidak menjadi tanggungan PT.  
PLN (Persero) P3B Region Jawa Timur dan Bali.
4. Buku Laporan Kerja Praktek Mahasiswa tersebut agar dikirimkan kepada PT. PLN (Persero) P3B  
Region Jawa Timur dan Bali 1 (satu) buah.
5. Untuk informasi lebih lanjut dapat menghubungi PT. PLN (Persero) P3B Region Jawa Timur dan  
Bali Cq. Bidang Engineering.

Demikian surat maklum dan terima kasih atas perhatian saudara.

Tembusan Yth. :

1. ALSDN PLN P3B.
2. Sdr. Desta Dwi M.

**BUKTI**





## SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : DESTA DWI MULYANTO  
Pria/ Wanita : PRIA  
Tempat / Tanggal lahir : MALANG / 24 - DESEMBER - 1981  
SA  
Alamat / no telepon : JL RAYA BOGOR 280 TUMPANG  
MALANG / (0341) 787511  
Pekerjaan : MAHASISWA

Dengan ini saya menerangkan bahwa :

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

Mengetahui  
Manager Bidang SDM & AD  
  
PRIZKAENAL ARIFIN

Surabaya,  
Yang membuat pernyataan

  
METERAI TEMPEL 6/04  
Rp 5000  
Lima Ribu Rupiah  
DESTA DWI M



## SURAT PERNYATAAN

Yang bertanda tangan dibawah ini, saya :

Nama : DESTA DWI MULYANTO  
Pria/ Wanita : PRIA  
Tempat / Tanggal lahir : MALANG / 24 - DESEMBER - 1981  
Alamat / no telepon : 68  
JL RAYA BOGOR 280 TUMPANG  
MALANG / (0341) 787511  
Pekerjaan : MAHASISWA

Dengan ini saya menerangkan bahwa :

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

Mengetahui  
Manager Bidang SDM & AD  
  
IR. ZAENAL ARIFIN

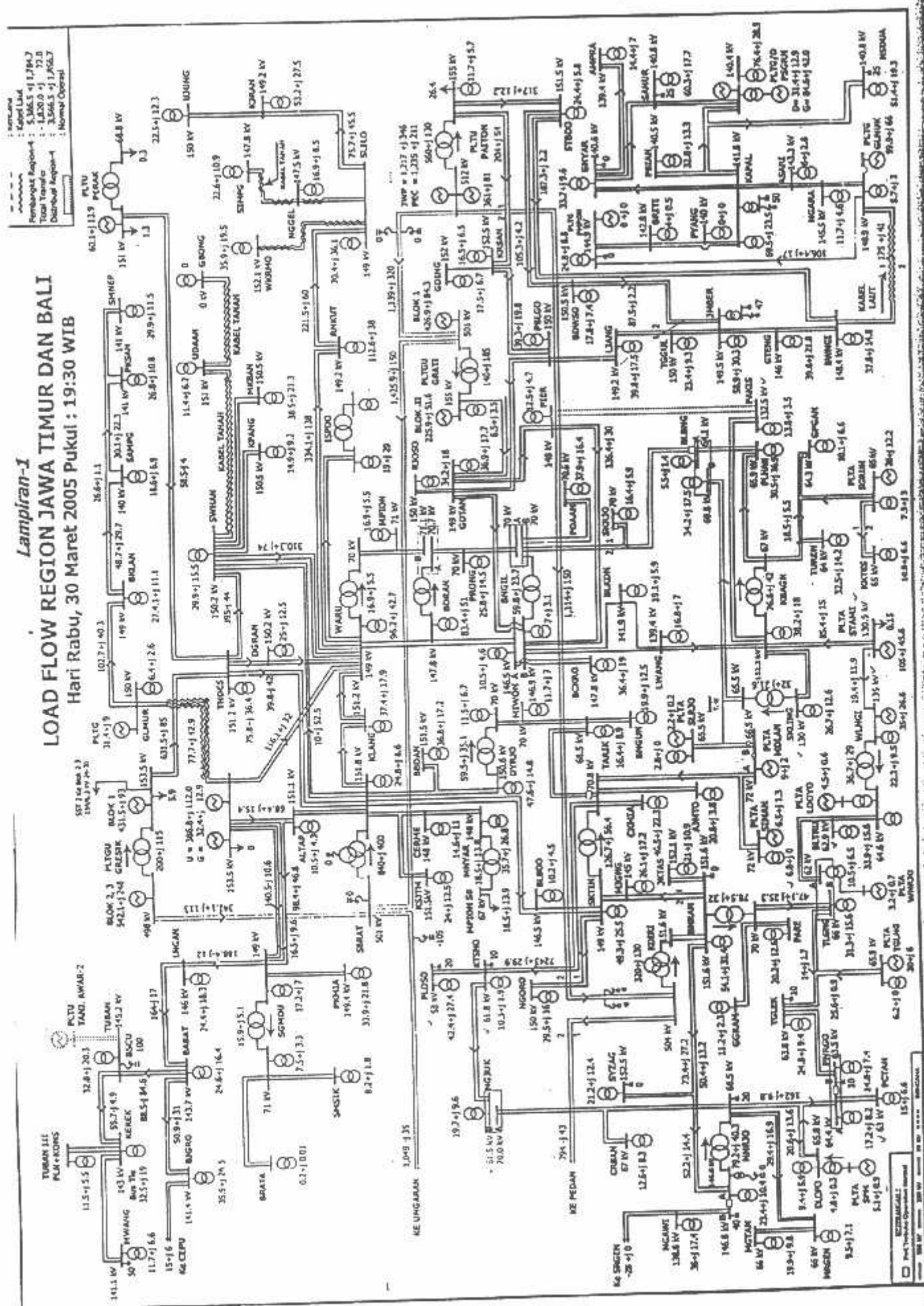
Surabaya,  
Yang membuat pernyataan

  
METERAI  
TEMPEL  
6000  
Rp 6.000  
DETA DWI M

### Lampiran-I

Hari Rabu, 30 Maret 2005 Pukul : 19:30 WIB

|                      |                       |
|----------------------|-----------------------|
| - - -                | - Retained            |
| ~~~~~                | - Retained Ltd.       |
| Fertilizant Region-4 | = 5,365.5 + j 1,784.7 |
| Total Transfer       | = - 1,820.9 + j 72.0  |
| Distribut Region-4   | = 3,566.5 + j 1,856.7 |
|                      | = Normal Output       |







## DATA KAKAKI EKSLIK TRANSMISI VPT &amp; SRB

[illegible]

## 6. UPT Probolinggo

|   | PITON5 | KRSAN5 | 2 | 150 | 20,204 | ACSR AW      | 330 mm <sup>2</sup>     | 7.12 | 1000 | 1250 | 1250 | 0.1172 | 0.4003 | 2.8530 | 0.0105 | 0.0355 | 0.0130 | 192,2529 |
|---|--------|--------|---|-----|--------|--------------|-------------------------|------|------|------|------|--------|--------|--------|--------|--------|--------|----------|
| 1 | PITON5 | STBDO5 | 1 | 150 | 55,433 | ACSR ZEBRA   | 2 x 435 mm <sup>2</sup> | 1500 | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | PITON5 | STBDO5 | 2 | 150 | 55,433 | ACSR ZEBRA   | 2 x 435 mm <sup>2</sup> | 1500 | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | GOTAN4 | 1 | 70  | 4,452  | ACSR OSTRICH | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | GOTAN4 | 2 | 70  | 4,452  | ACSR OSTRICH | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | POJAN4 | 1 | 70  | 9,700  | ACSR OSTRICH | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | POJAN4 | 2 | 70  | 9,700  | ACSR OSTRICH | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | BORAN4 | 1 | 70  | 11,700 | ACSR PIPER   | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | PRONG4 | 1 | 70  | 11,700 | ACSR PIPER   | 300 MCM                 | 440  | 2000 | 1250 | 1250 | 0.1387 | 0.2607 | 4.0230 | 0.0095 | 0.0692 | 0.0502 | 415,5650 |
| 1 | BNGIL4 | BLBNG4 | 1 | 70  | 40,000 | ACSR PIGEON  | 300 AWG                 | 300  | 200  | 800  | 800  | 0.3053 | 0.3584 | 3.202  | 0.2492 | 0.2926 | 0.0063 | 36,3720  |
| 1 | BNGIL4 | BLBNG4 | 2 | 70  | 40,000 | ACSR PIGEON  | 300 AWG                 | 300  | 200  | 800  | 800  | 0.3053 | 0.3584 | 3.202  | 0.2492 | 0.2926 | 0.0063 | 36,3720  |

|                                                                                                                                                     |        |
|-----------------------------------------------------------------------------------------------------------------------------------------------------|--------|
| Modifikasi TIL Bay Grati - Pier 1 sirih menjadi ( Grati - Gditan - Pier ) dan Probolinggo - Gondangwetan 1 sirih menjadi ( Pblgo - Grati - Gditan ) |        |
| GRATIS - GdITAN'S                                                                                                                                   | 150    |
|                                                                                                                                                     | 10.743 |
|                                                                                                                                                     | TACSO  |







```

unit uLoadflow;

interface

uses uUtils,uComplex,SysUtils;

type
  TParam=(psPu,psOhm);

  TGeneral=record
    MaxIterasi,Iterasi:integer;
    Vbase,VKonst,Pbase,PKonst,Zbase,Ibase,Toleransi:double;
    Param:TParam;
    sumGen,sumLoad,sumLoss:TComplex;
  end;

var gGeneral:TGeneral;
    gLengthBranch:dArr1;
    gBus,gBranch,gGen:dArr2;

procedure DecodeData(const rBus,rBranch:dArr2;
  var rGeneral:TGeneral;
  var rNbus,rNsal:integer;
  var rE,rF,rPg,rQg,rPL,rQL,rCap:dArr1;
  var rTyp:iArr1;
  var rR,rX,rLc,rTr,rTu,rSu:dArr2);

function FindSumGen(const rNbus:integer;const rTyp:iArr1):integer;
procedure Admittansi(const rNbus:integer;
  const rR,rX,rLc,rTr,rTu,rSu:dArr2;
  const rCap:dArr1;
  var rG,rB:dArr2);
procedure RecToPolar(const rRecRe,rRecIm:dArr1;
  var rPolRe,rPolIm:dArr1);
procedure PolarToRec(const rPolRe,rPolIm:dArr1;
  var rRecRe,rRecIm:dArr1);
procedure AliranDaya(const rNbus:integer;const rE,rF:dArr1;
  const rG,rB,rLc:dArr2;
  var rAlirP,rAlirQ:dArr2);
procedure DayaGen(const rNbus:integer;
  const rE,rF,rPL,rQL:dArr1;
  const rG,rB:dArr2;const rTyp:iArr1;
  var rQg:dArr1);
procedure DayaSlack(const rNbus:integer;
  const rAlirP,rAlirQ:dArr2;
  const rTyp:iArr1;
  const rPL,rQL:dArr1;
  var rPg,rQg:dArr1);
procedure ArusBranch(const rNbus:integer;
  const rE,rF:dArr1;
  const rLc,rG,rB:dArr2;
  var rArusRe,rArusIm:dArr2);
procedure Summary(const rNbus:integer;
  const rPg,rQg,rPL,rQL:dArr1;
  var rSumGen,rSumLoad,rSumLoss:TComplex);
function MaxArray(const rData:dArr1):double;

procedure UpdateBusGen(const rNbus:integer;
  const rE,rF,rPg,rQg:dArr1;
  const rGeneral:TGeneral;

```

```

var rBus:dArr2;
procedure UpdateBranch(const rNsal:integer;
const rAlirRe,rAlirIm,rArusRe,rArusIm:dArr2;
const rGeneral:TGeneral;
var rBranch:dArr2);

implementation

procedure DecodeData(const rBus,rBranch:dArr2;
var rGeneral:TGeneral;
var rNbus,rNsal:integer;
var rE,rF,rPg,rQg,rPL,rQL,rCap:dArr1;
var rTyp:iArr1;
var rR,rX,rLc,rTr,rTu,rSu:dArr2);
var i,j,dari,ke:integer;
begin
rGeneral.Zbase:=sqr(rGeneral.Vbase*rGeneral.VKonst)/
(rGeneral.Pbase*rGeneral.PKonst);
rGeneral.Ibase:=rGeneral.Vbase*rGeneral.VKonst/rGeneral.Zbase;
rNbus:=high(rBus)+1;
rNsal:=high(rBranch)-1;
SetLength(rE,rNbus);
SetLength(rF,rNbus);
SetLength(rPg,rNbus);
SetLength(rQg,rNbus);
SetLength(rPL,rNbus);
SetLength(rQL,rNbus);
SetLength(rCap,rNbus);
SetLength(rTyp,rNbus);
SetLength(rR,rNbus,rNbus);
SetLength(rX,rNbus,rNbus);
SetLength(rLc,rNbus,rNbus);
SetLength(rTr,rNbus,rNbus);
SetLength(rTu,rNbus,rNbus);
SetLength(rSu,rNbus,rNbus);
for i:=0 to rNbus-1 do
begin
rE[i]:=rBus[i,0];
rF[i]:=rBus[i,1];
rPg[i]:=rBus[i,2]/rGeneral.Pbase;
rQg[i]:=rBus[i,3]/rGeneral.Pbase;
rPL[i]:=rBus[i,4]/rGeneral.Pbase;
rQL[i]:=rBus[i,5]/rGeneral.Pbase;
if rGeneral.Param=psPu then
begin
rCap[i]:=rBus[i,6];
end
else
begin
rCap[i]:=rBus[i,6]/rGeneral.Zbase;
end;
rTyp[i]:=round(rBus[i,7]);
for j:=0 to rNbus-1 do
begin
rR[i,j]:=0.0;
rX[i,j]:=0.0;
rLc[i,j]:=0.0;
rTr[i,j]:=0.0;
rTu[i,j]:=0.0;
rSu[i,j]:=0.0;

```

```

end;
end;
for i:=0 to rNsal-1 do
begin
  dari:=round(rBranch[i,0])-1;
  ke:=round(rBranch[i,1])-1;
  if rGeneral.Param=psPu then
  begin
    rR[dari,ke]:=rBranch[i,2];
    rX[dari,ke]:=rBranch[i,3];
    rLc[dari,ke]:=rBranch[i,4];
    rR[ke,dari]:=rBranch[i,2];
    rX[ke,dari]:=rBranch[i,3];
    rLc[ke,dari]:=rBranch[i,4];
  end
  else if rGeneral.Param=psOhm then
  begin
    rR[dari,ke]:=rBranch[i,2]/rGeneral.Zbase;
    rX[dari,ke]:=rBranch[i,3]/rGeneral.Zbase;
    rLc[dari,ke]:=rBranch[i,4]/rGeneral.Zbase;
    rR[ke,dari]:=rBranch[i,2]/rGeneral.Zbase;
    rX[ke,dari]:=rBranch[i,3]/rGeneral.Zbase;
    rLc[ke,dari]:=rBranch[i,4]/rGeneral.Zbase;
  end;
  rTr[dari,ke]:=rBranch[i,5];
  rTu[dari,ke]:=rBranch[i,6];
  rSu[dari,ke]:=rBranch[i,7];
end;
end;

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

procedure Admittansi(const rNbus:integer;
  const rR,rX,rLc,rTr,rTu,rSu:dArr2;
  const rCap:dArr1;
  var rG,rB:dArr2);
var i,j,k:integer;
  a,b:double;
  Cr,Ci:dArr2;
  sum,Ca,Za:TComplex;
begin
  SetLength(Cr,rNbus,rNbus);
  SetLength(Ci,rNbus,rNbus);
  sum:=TComplex.Create(1.0,0.0);
  for i:=0 to rNbus-1 do
  begin
    for j:=0 to rNbus-1 do
    begin

```

---

```

if rX[i,j] <> 0 then
begin
  Za:=TComplex.Create(rR[i,j],rX[i,j]);
  Ca:=sum.Divide(Za);
  Cr[i,j]:=Ca.Real;
  Ci[i,j]:=Ca.Imag;
  Ca.Free;
  Za.Free;
end
else
begin
  Cr[i,j]:=-0.0;
  Ci[i,j]:=-0.0;
end;
end;
sum.Free;
Setf.length(rG,rNbus,rNbus);
Setf.length(rB,rNbus,rNbus);
for i:=0 to rNbus-1 do
begin
  for j:=0 to rNbus-1 do
  begin
    if j=i then
    begin
      rG[i,j]:=0.0;
      rB[i,j]:=0.0;
      for k:=0 to rNbus-1 do
      begin
        rG[i,j]:=rG[i,j]+Cr[i,k];
        rB[i,j]:=rB[i,j]+Ci[i,k]+rLc[i,k];
      end;
    end
    else
    begin
      rG[i,j]:=-Cr[i,j];
      rB[i,j]:=-Ci[i,j];
    end;
  end;
end;
for i:=0 to rNbus-1 do
begin
  for j:=0 to rNbus-1 do
  begin
    if rTr[i,j] <> 0 then
    begin
      rG[i,i]:=rG[i,i]-Cr[i,j];
      rB[i,i]:=rB[i,i]-Ci[i,j]-rLc[i,j];
      rG[i,i]:=rG[i,i]+Cr[i,j]/sqrt(rTr[i,j]);
      rB[i,i]:=rB[i,i]+Ci[i,j]/sqrt(rTr[i,j]);
      rG[j,j]:=rG[j,j]-Cr[i,j];
      rB[j,j]:=rB[j,j]-Ci[i,j]-rLc[i,j];
      rG[j,j]:=rG[j,j]+Cr[i,j];
      rB[j,j]:=rB[j,j]+Ci[i,j];
      rG[i,j]:=-1*Cr[i,j]/rTr[i,j];
      rB[i,j]:=-1*Ci[i,j]/rTr[i,j];
      rG[j,i]:=rG[i,j];
      rB[j,i]:=rB[i,j];
      { CLc:=TComplex.Create(0.0,aLc[i,j]);
      result[i,i]:=result[i,i]-Cx[i,j]-CLc;

```

```

    result[i,i]:=result[i,i]+Cx[i,j]/sqr(at);
    result[j,j]:=result[j,j]-Cx[i,j]-CLc;
    result[j,j]:=result[j,j]+Cx[i,j];
    result[i,j]:=-Cx[i,j]/at;
    result[j,i]:=result[i,j];
    CLc.Free;
end;
end;
end;
for i:=0 to rNbus-1 do
begin
    for j:=0 to rNbus-1 do
    begin
        if rTu[i,j]<>0 then
        begin
            a:=rTu[i,j]*cos(rSu[i,j]);
            b:=rTu[i,j]*sin(rSu[i,j]);
            rG[i,i]:=rG[i,i]-Cr[i,j];
            rB[i,i]:=rB[i,i]-Ci[i,j]-rLc[i,j];
            rG[i,i]:=rG[i,i]+Cr[i,j]/(sqr(a)+sqr(b));
            rB[i,i]:=rB[i,i]+Ci[i,j]/(sqr(a)+sqr(b))+rLc[i,j];
            Za:=TComplex.Create(Cr[i,j],Ci[i,j]);
            Za:=Za.Negative;
            Ca:=TComplex.Create(a,b);
            sum:=Za.Divide(Ca);
            rG[j,i]:=sum.Real;
            rB[j,i]:=sum.Imag;
            sum.Free;
            Ca:=Ca.Conj;
            sum:=Za.Divide(Ca);
            rG[i,j]:=sum.Real;
            rB[i,j]:=sum.Imag;
            sum.Free;
            Ca.Free;
            Za.Free;
            {sUpfc:=TComplex.Create(aUpfc[i,j].tap*cos(aUpfc[i,j].sudut),
                aUpfc[i,j].tap*sin(aUpfc[i,j].sudut));
            CLc:=TComplex.Create(0.0,aLc[i,j]);
            result[i,i]:=result[i,i]-Cx[i,j]-CLc;
            result[i,i]:=result[i,i]+Cx[i,j]/sqr(sUpfc.Abs)+CLc;
            result[i,j]:=-Cx[i,j]/conj(sUpfc);
            result[j,i]:=-Cx[i,j]/sUpfc;
            CLc.Free;
            sUpfc.Free;}
        end;
    end;
end;
end;
for i:=0 to rNbus-1 do
begin
    if rCap[i]<>0 then
    begin
        rB[i,i]:=rB[i,i]+rCap[i];
    end;
end;
end;

procedure RecToPolar(const rRecRe,rRecIm:dArr1;
    var rPolRe,rPolIm:dArr1);
var i:integer;
begin

```

```

if high(rRecRe) <> high(rRecIm) then
begin
  raise Exception.Create('Dimensi kedua Vector tidak sama!');
end;
SetLength(rPolRe, high(rRecRe) + 1);
SetLength(rPolIm, high(rRecRe) + 1);
for i := 0 to high(rRecRe) do
begin
  rPolRe[i] := sqrt(sqrt(rRecRe[i]) + sqrt(rRecIm[i]));
  rPolIm[i] := arctan(rRecIm[i] / rRecRe[i]);
end;
end;

procedure PolarToRec(const rPolRe, rPolIm: dArr1;
  var rRecRe, rRecIm: dArr1);
var i: integer;
begin
  if high(rRecRe) <> high(rRecIm) then
  begin
    raise Exception.Create('Dimensi kedua Vector tidak sama!');
  end;
  SetLength(rRecRe, high(rPolRe) + 1);
  SetLength(rRecIm, high(rPolRe) + 1);
  for i := 0 to high(rPolRe) do
  begin
    rRecRe[i] := rPolRe[i] * cos(rPolIm[i]);
    rRecIm[i] := rPolRe[i] * sin(rPolIm[i]);
  end;
end;

procedure AliranDaya(const rNbus: integer; const rE, rF: dArr1;
  const rG, rB, rLc: dArr2;
  var rAlirP, rAlirQ: dArr2);
var i, j: integer;
    Vi, Vj, Ya, Lca, Sa, tmp1, tmp2: TComplex;
begin
  SetLength(rAlirP, rNbus, rNbus);
  SetLength(rAlirQ, rNbus, rNbus);
  for i := 0 to rNbus - 1 do
  begin
    Vi := TComplex.Create(rE[i], rF[i]);
    for j := 0 to rNbus - 1 do
    begin
      rAlirP[i, j] := 0.0;
      rAlirQ[i, j] := 0.0;
      if j <> i then
      begin
        if rB[i, j] < 0 then
        begin
          Vj := TComplex.Create(rE[j], rF[j]);
          Ya := TComplex.Create(rG[i, j], rB[i, j]);
          Lca := TComplex.Create(0.0, rLc[i, j]);
          tmp1 := Vi.Conj.Multiply(Vi.Subtract(Vj)).Multiply(Ya.Negative);
          tmp2 := Vi.Conj.Multiply(Vi).Multiply(Lca);
          Sa := tmp1.Add(tmp2);
          rAlirP[i, j] := Sa.Real;
          rAlirQ[i, j] := -Sa.Imag;
          //result[i, j] := conj(aV[i]) * (aV[i] - aV[j]) * (-aY[i, j]) +
          //      conj(aV[i]) * aV[j] * dLc;
          //result[i, j] := conj(result[i, j]);
        end;
      end;
    end;
  end;
end;

```

```

    Sa.Free;
    tmp2.Free;
    tmp1.Free;
    l.ca.Free;
    Ya.Free;
    Vj.Free;
end;
end;
end;
Vi.Free;
end;
end;

procedure DayaGen(const rNbus:integer;
    const rE,rF,rPL,rQL:dArr1;
    const rG,rB:dArr2;const rTyp:iArr1;
    var rQg:dArr1);
var i,j:integer;
    sum:double;
begin
    for i:=0 to rNbus-1 do
    begin
        sum:=0.0;
        if rTyp[i]=2 then
        begin
            for j:=0 to rNbus-1 do
            begin
                //sum:=sum+(Fi*(Ej*Gij+Fj*-Bij)-Ei*(Fj*Gij-Ej*-Bij));
                sum:=sum+(rF[i]*(rE[j]*rG[i,j]+rF[j]*-rB[i,j])-
                    rE[i]*(rF[j]*rG[i,j]-rE[j]*-rB[i,j]));
            end;
            rQg[i]:=sum+rQL[i];
        end;
    end;
end;

procedure DayaSlack(const rNbus:integer;
    const rAlirP,rAlirQ:dArr2;
    const rTyp:iArr1;
    const rPL,rQL:dArr1;
    var rPg,rQg:dArr1);
var i,j:integer;
    sumP,sumQ:double;
begin
    for i:=0 to rNbus-1 do
    begin
        if rTyp[i]=1 then
        begin
            sumP:=0.0;
            sumQ:=0.0;
            for j:=0 to rNbus-1 do
            begin
                if rAlirQ[i,j]<>0 then
                begin
                    sumP:=sumP+rAlirP[i,j];
                    sumQ:=sumQ+rAlirQ[i,j];
                end;
            end;
            rPg[i]:=sumP+rPL[i];
            rQg[i]:=sumQ+rQL[i];
        end;
    end;
end;

```

---

```

    end;
end;
end;

procedure ArusBranch(const rNbus:integer;
    const rE,rF:dArr1;
    const rLc,rG,rB:dArr2;
    var rArusRe,rArusIm:dArr2);
var i,j:integer;
    Vi,Vj,Ya,xLc,Arus,tmp1,tmp2:TComplex;
begin
    SetLength(rArusRe,rNbus,rNbus);
    SetLength(rArusIm,rNbus,rNbus);
    for i:=0 to rNbus-1 do
        begin
            Vi:=TComplex.Create(rE[i],rF[i]);
            for j:=0 to rNbus-1 do
                begin
                    if rB[i,j]<>0 then
                        begin
                            Vj:=TComplex.Create(rE[j],rF[j]);
                            Ya:=TComplex.Create(rG[i,j],rB[i,j]);
                            xLc:=TComplex.Create(0,rLc[i,j]);
                            tmp1:=Ya.Negative.Multiply(Vi.Subtract(Vj));
                            tmp2:=Vi.Multiply(xLc);
                            Arus:=tmp1.Add(tmp2);
                            rArusRe[i,j]:=Arus.Real;
                            rArusIm[i,j]:=-Arus.Imag;
                            Arus.Free;
                            tmp2.Free;
                            tmp1.Free;
                            xLc.Free;
                            Ya.Free;
                            Vj.Free;
                            //result[i,j]:=(aV[i]-aV[j])*(-aY[i,j])+aV[i]*xLc;
                            //result[i,j]:=Conj(result[i,j]);
                        end
                    else
                        begin
                            rArusRe[i,j]:=0.0;
                            rArusIm[i,j]:=0.0;
                        end;
                    end;
                    Vi.Free;
                end;
            end;
        end;
    end;

procedure Summary(const rNbus:integer;
    const rPg,rQg,rPL,rQL:dArr1;
    var rSumGen,rSumLoad,rSumLoss:TComplex);
var i:integer;
    sumPg,sumPL,sumQg,sumQL:double;
begin
    sumPg:=0.0;
    sumQg:=0.0;
    sumPL:=0.0;
    sumQL:=0.0;
    for i:=0 to rNbus-1 do
        begin
            sumPg:=sumPg+rPg[i];

```

---

```

sumQg:=sumQg+rQg[i];
sumPl:=sumPl+rPl[i];
sumQl:=sumQl+rQl[i];
end;
rSumGen:=TComplex.Create(sumPg,sumQg);
rSumLoad:=TComplex.Create(sumPl,sumQl.);
rSumLoss:=rSumGen.Subtract(rSumLoad);
end;

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

procedure UpdateBusGen(const rNbus:integer;
const rE,rF,rPg,rQg:dArr1;
const rGeneral:TGeneral;
var rBus:dArr2);
var i:integer;
begin
for i:=0 to rNbus-1 do
begin
rBus[i,0]:=rE[i];
rBus[i,1]:=rF[i];
rBus[i,2]:=rPg[i]*rGeneral.Pbase;
rBus[i,3]:=rQg[i]*rGeneral.Pbase;
end;
end;

procedure UpdateBranch(const rNsal:integer;
const rAlirRe,rAlirIm,rArusRe,rArusIm:dArr2;
const rGeneral:TGeneral;
var rBranch:dArr2);
var i,dari,ke:integer;
begin
for i:=0 to rNsal-1 do
begin
dari:=round(rBranch[i,0])-1;
ke:=round(rBranch[i,1])-1;
rBranch[i,9]:=rAlirRe[dari,ke]*rGeneral.Pbase;
rBranch[i,10]:=rAlirIm[dari,ke]*rGeneral.Pbase;
rBranch[i,11]:=rAlirRe[ke,dari]*rGeneral.Pbase;
rBranch[i,12]:=rAlirIm[ke,dari]*rGeneral.Pbase;
rBranch[i,13]:=rArusRe[dari,ke]*rGeneral.Ibase;
rBranch[i,14]:=rArusIm[dari,ke]*rGeneral.Ibase;
rBranch[i,15]:=rArusRe[ke,dari]*rGeneral.Ibase;
rBranch[i,16]:=rArusIm[ke,dari]*rGeneral.Ibase;
end;
end;

end.

```

```

unit uInputLF;

interface

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

type
  TfrmInput = class(TForm)
    PageControl1: TPageControl;
    Panel1: TPanel;
    TabSheet1: TTabSheet;
    TabSheet2: TTabSheet;
    TabSheet3: TTabSheet;
    btnClose: TButton;
    btnNext: TButton;
    SaveDialog1: TSaveDialog;
    Label1: TLabel;
    Label2: TLabel;
    Label3: TLabel;
    Label4: TLabel;
    Label5: TLabel;
    edtNbus: TEdit;
    edtNsal: TEdit;
    edtVbase: TEdit;
    edtIbase: TEdit;
    cmbParam: TComboBox;
    cmbVKonst: TComboBox;
    cmbPkonst: TComboBox;
    fgBus: TStringGrid;
    fgBranch: TStringGrid;
    TabSheet4: TTabSheet;
    fgGen: TStringGrid;
    procedure FormCreate(Sender: TObject);
    procedure btnCloseClick(Sender: TObject);
    procedure edtNbusChange(Sender: TObject);
    procedure edtNsalChange(Sender: TObject);
    procedure cmbPkonstChange(Sender: TObject);
    procedure cmbParamChange(Sender: TObject);
    procedure btnNextClick(Sender: TObject);
    procedure TabSheet4Show(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;

var
  frmInput: TfrmInput;

implementation

uses uLoadflow, uNewtonPolar, uUtils, uParamTS;

{$R *.dfm}

procedure TfrmInput.FormCreate(Sender: TObject);
begin
  edtNbus.Text:='1';

```

---

```

edtNsal.Text:='1';
edtVhase.Text:='150';
edtPbase.Text:='100';
cmbVkonst.Text:='kV';
cmbPkonst.Text:='MVA';
cmbParam.Text:='pu';
fgBus.Cells[0,0]:='Bus';
fgBus.Cells[1,0]:='absV (pu)';
fgBus.Cells[2,0]:='sudV (deg)';
fgBus.Cells[3,0]:='Pg (MW)';
fgBus.Cells[4,0]:='Qg (MVAR)';
fgBus.Cells[5,0]:='PL (MW)';
fgBus.Cells[6,0]:='QL (MVAR)';
fgBus.Cells[7,0]:='Cap (pu)';
fgBus.Cells[8,0]:='Type Bus';
fgBranch.Cells[0,0]:='No';
fgBranch.Cells[1,0]:='Dari';
fgBranch.Cells[2,0]:='Ke';
fgBranch.Cells[3,0]:='R (pu)';
fgBranch.Cells[4,0]:='X (pu)';
fgBranch.Cells[5,0]:='Lc (pu)';
fgBranch.Cells[6,0]:='Tr';
fgBranch.Cells[7,0]:='Tu';
fgBranch.Cells[8,0]:='Su (deg)';
fgBranch.Cells[9,0]:='Kap (MVA)';
fgBranch.Cells[10,0]:='P (MW)';
fgBranch.Cells[11,0]:='Q (MVAR)';
fgBranch.Cells[12,0]:='Arus re (A)';
fgBranch.Cells[13,0]:='Arus im (A)';
fgBranch.Cells[14,0]:='Dari';
fgBranch.Cells[15,0]:='Ke';
fgBranch.Cells[16,0]:='P (MW)';
fgBranch.Cells[17,0]:='Q (MVAR)';
fgBranch.Cells[18,0]:='Arus re (A)';
fgBranch.Cells[19,0]:='Arus im (A)';
fgBranch.Cells[20,0]:='Kap (kVA)';
fgBranch.Cells[21,0]:='Length (m)';
fgGen.Cells[0,0]:='No';
fgGen.Cells[1,0]:='Bus';
fgGen.Cells[2,0]:='Qmin (MVAR)';
fgGen.Cells[3,0]:='Qmax (MVAR)';
fgGen.Cells[4,0]:='a2';
fgGen.Cells[5,0]:='a1';
fgGen.Cells[6,0]:='a0';
fgGen.Cells[7,0]:='Fix Cost';
fgGen.Cells[8,0]:='Var Cost';
fgGen.Cells[9,0]:='Pmin (MW)';
fgGen.Cells[10,0]:='Pmax (MW)';
end;

```

```

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

```

```

procedure TfrmInput.edtNbusChange(Sender: TObject);
var i:integer;
begin
  if edtNbus.Text="" then
    begin

```

```

fgBus.Cells[5,0]:='PL (W)';
fgBus.Cells[6,0]:='QL (VAR)';
fgBranch.Cells[9,0]:='Kap (VA)';
fgBranch.Cells[10,0]:='P (W)';
fgBranch.Cells[11,0]:='Q (VAR)';
fgBranch.Cells[16,0]:='P (W)';
fgBranch.Cells[17,0]:='Q (VAR)';
fgGen.Cells[9,0]:='Pmin (W)';
fgGen.Cells[10,0]:='Pmax (W)';
end
else if cmbPKonst.Text='kVA' then
begin
gGeneral.PKonst:=1000;
fgBus.Cells[3,0]:='Pg (kW)';
fgBus.Cells[4,0]:='Qg (kVAR)';
fgBus.Cells[5,0]:='PL (kW)';
fgBus.Cells[6,0]:='QL (kVAR)';
fgBranch.Cells[9,0]:='Kap (VA)';
fgBranch.Cells[10,0]:='P (kW)';
fgBranch.Cells[11,0]:='Q (kVAR)';
fgBranch.Cells[16,0]:='P (kW)';
fgBranch.Cells[17,0]:='Q (kVAR)';
fgGen.Cells[9,0]:='Pmin (kW)';
fgGen.Cells[10,0]:='Pmax (kW)';
end
else if cmbPKonst.Text='MVA' then
begin
gGeneral.PKonst:=1000000;
fgBus.Cells[3,0]:='Pg (MW)';
fgBus.Cells[4,0]:='Qg (MVAR)';
fgBus.Cells[5,0]:='PL (MW)';
fgBus.Cells[6,0]:='QL (MVAR)';
fgBranch.Cells[9,0]:='Kap (VA)';
fgBranch.Cells[10,0]:='P (MW)';
fgBranch.Cells[11,0]:='Q (MVAR)';
fgBranch.Cells[16,0]:='P (MW)';
fgBranch.Cells[17,0]:='Q (MVAR)';
fgGen.Cells[9,0]:='Pmin (MW)';
fgGen.Cells[10,0]:='Pmax (MW)';
end;
end;

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

```

---

end;

```
procedure TForm1.btnNextClick(Sender: TObject);
var NamaFile,Nama:string;
    input:TextFile;
    NCable,Nbus,Nsal,Ngen,i,typ,dari,ke,param:integer;
    R,X,L,c,Tr,Tu,Su,Cap,CapSal,phi,Pmin,Pmax,l,larga,length:double;
    VKonst,PKonst,Vbase,Pbase,absV,sudV,Pg,Qg,PL,QL:double;
    jam,menit,detik,mdetik:word;
    mulai,selesai,selang:TDateTime;
    load:string;
begin
    if btnNext.Caption='&Save' then
    begin
        Nbus:=StrToInt(edtNbus.Text);
        Nsal:=StrToInt(edtNsal.Text);
        Vbase:=StrToFloat(edtVbase.Text);
        VKonst:=1;
        if cmbVKonst.Text='V' then
        begin
            VKonst:=1;
        end
        else if cmbVKonst.Text='kV' then
        begin
            VKonst:=1000;
        end
        else if cmbVKonst.Text='MV' then
        begin
            VKonst:=1000000;
        end;
        Phase:=StrToFloat(edtPhase.Text);
        PKonst:=1;
        if cmbPKonst.Text='VA' then
        begin
            PKonst:=1;
        end
        else if cmbPKonst.Text='kVA' then
        begin
            PKonst:=1000;
        end
        else if cmbPKonst.Text='MVA' then
        begin
            PKonst:=1000000;
        end;
        Param:=1;
        if cmbParam.Text='pu' then
        begin
            Param:=1;
        end
        else if cmbParam.Text='ohm' then
        begin
            Param:=-2;
        end;
    try
        if SaveDialog1.Execute then
        begin
            NamaFile:=SaveDialog1.FileName;
            AssignFile(input,NamaFile+'.txt');
            Rewrite(input);
            Writeln(input,Nbus);
        end;
    end;
end;
```

---

```

Writeln(input,Nsal);
Writeln(input,Vbase:6:2);
Writeln(input,VKonst:7:0);
Writeln(input,Pbase:6:2);
Writeln(input,PKonst:7:0);
Writeln(input,param);
phi:=4*arctan(1);
for i:=1 to Nbus do
begin
  absV:=StrToFloat(fgBus.Cells[1,i]);
  sudV:=StrToFloat(fgBus.Cells[2,i])*180/phi;
  Pg:=StrToFloat(fgBus.Cells[3,i]);
  Qg:=StrToFloat(fgBus.Cells[4,i]);
  PL:=StrToFloat(fgBus.Cells[5,i]);
  QL:=StrToFloat(fgBus.Cells[6,i]);
  Cap:=StrToFloat(fgBus.Cells[7,i]);
  Typ:=StrToInt(fgBus.Cells[8,i]);
  Writeln(input,absV:7:5,' ',sudV:7:5,' ',Pg:9:3,' ',Qg:9:3,
    ' ',PL:9:3,' ',QL:9:3,' ',Cap:7:5,' ',Typ);
end;
for i:=1 to Nsal do
begin
  dari:=StrToInt(fgBranch.Cells[1,i]);
  ke:=StrToInt(fgBranch.Cells[2,i]);
  R:=StrToFloat(fgBranch.Cells[3,i]);
  X:=StrToFloat(fgBranch.Cells[4,i]);
  Lc:=StrToFloat(fgBranch.Cells[5,i]);
  Tr:=StrToFloat(fgBranch.Cells[6,i]);
  Tu:=StrToFloat(fgBranch.Cells[7,i]);
  Su:=StrToFloat(fgBranch.Cells[8,i]);
  CapSal:=StrToFloat(fgBranch.Cells[9,i]);
  Length:=StrToFloat(fgBranch.Cells[10,i]);
  Writeln(input,dari,' ',ke,' ',R:7:5,' ',X:7:5,' ',
    Lc:7:5,' ',Tr:7:5,' ',Tu:7:5,' ',Su:7:5,' ',
    CapSal:7:2,' ',Length:7:2);
end;
Ngen:=0;
load:=3;
for i:=1 to Nbus do
begin
  if fgBus.Cells[8,i]<>load then
  begin
    inc(Ngen);
  end;
end;
Writeln(input,Ngen);
for i:=1 to Ngen do
begin
  dari:=StrToInt(fgGen.Cells[1,i]);
  R:=StrToFloat(fgGen.Cells[2,i]);
  X:=StrToFloat(fgGen.Cells[3,i]);
  Lc:=StrToFloat(fgGen.Cells[4,i]);
  Tr:=StrToFloat(fgGen.Cells[5,i]);
  Tu:=StrToFloat(fgGen.Cells[6,i]);
  Su:=StrToFloat(fgGen.Cells[7,i]);
  CapSal:=StrToFloat(fgGen.Cells[8,i]);
  Pmin:=StrToFloat(fgGen.Cells[9,i]);
  Pmax:=StrToFloat(fgGen.Cells[10,i]);
  Writeln(input,dari,' ',R:7:2,' ',X:7:2,' ',Lc:7:5,' ',
    Tr:7:5,' ',Tu:7:5,' ',Su:7:2,' ',CapSal:7:2,' ',

```

```

        Pmin:7:5,' ',Pmax:7:2);
    end;
end;
CloseFile(input);
MessageDlg('File berhasil disimpan!',mtInformation,[mbOK],0);
except
    MessageDlg('Tolong dicek angka-angkanya kembali!',
        mtWarning,[mbOK],0);
end;
end
else if btnNext.Caption='&Next' then
begin
    if gGeneral.PKonst=1 then
    begin
        frmParamTS.fgHasil.Cells[3,0]:='Pg (W)';
        frmParamTS.fgHasil.Cells[4,0]:='Qg (VAR)';
    end
    else if gGeneral.PKonst=1000 then
    begin
        frmParamTS.fgHasil.Cells[3,0]:='Pg (kW)';
        frmParamTS.fgHasil.Cells[4,0]:='Qg (kVAR)';
    end
    else if gGeneral.PKonst=1000000 then
    begin
        frmParamTS.fgHasil.Cells[3,0]:='Pg (MW)';
        frmParamTS.fgHasil.Cells[4,0]:='Qg (MVAR)';
    end;
    frmParamTS.Show;
end;
end;

procedure TfrmInput.TabSheet4Show(Sender: TObject);
var i,Ngen:integer;
    load:string;
begin
    Ngen:=0;
    load:='3';
    for i:=1 to fgBus.RowCount-1 do
    begin
        if fgBus.Cells[8,i]<>load then
        begin
            inc(Ngen);
        end;
    end;
    fgGen.RowCount:=Ngen+1;
    Ngen:=0;
    for i:=1 to fgBus.RowCount-1 do
    begin
        if fgBus.Cells[8,i]<>load then
        begin
            inc(Ngen);
            fgGen.Cells[0,Ngen]:=IntToStr(Ngen);
            fgGen.Cells[1,Ngen]:=IntToStr(i);
            if fgGen.Cells[2,Ngen]='' then
            begin
                fgGen.Cells[2,Ngen]:='- 1000';
            end;
            if fgGen.Cells[3,Ngen]='' then
            begin
                fgGen.Cells[3,Ngen]:='1000';
            end;
        end;
    end;
end;

```

---

```
end;  
if fgGen.Cells[4,Ngen]=" then  
begin  
    fgGen.Cells[4,Ngen]:='0';  
end;  
if fgGen.Cells[5,Ngen]=" then  
begin  
    fgGen.Cells[5,Ngen]:='0';  
end;  
if fgGen.Cells[6,Ngen]=" then  
begin  
    fgGen.Cells[6,Ngen]:='0';  
end;  
if fgGen.Cells[7,Ngen]=" then  
begin  
    fgGen.Cells[7,Ngen]:='0';  
end;  
if fgGen.Cells[8,Ngen]=" then  
begin  
    fgGen.Cells[8,Ngen]:='0';  
end;  
if fgGen.Cells[9,Ngen]=" then  
begin  
    fgGen.Cells[9,Ngen]:='0';  
end;  
if fgGen.Cells[10,Ngen]=" then  
begin  
    fgGen.Cells[10,Ngen]:='1000';  
end;  
end;  
end;  
end;  
end;
```

---

```

unit uFitness;

interface

uses uUtils,uNewtonPolar,uLoadflow;

function FindLength:integer;
function FindBatasChrom(var rBatasV:TBatas):TBatasArr1;
function InitChrom:dArr1;
function doHitungFitness(const rChrom:dArr1;
    const rBatasV:TBatas;
    const rLamdaV,rLamdaS:double):double;
procedure doHitungFitnessAkhir(const rChrom:dArr1;
    var rLBus,rLBranch:dArr2;
    var rGeneral:TGeneral);
function FindTotalCostGen(const rLBus:dArr2):double;
function DecodeChromAkhir(const rChrom:dArr1):dArr2;
function FindCostGenArr(const rLBus:dArr2):dArr1;

implementation

function FindLength:integer;
var i:integer;
begin
    result:=0;
    for i:=0 to high(gBus) do
    begin
        if gBus[i,7]=1 then
        begin
            inc(result);
        end
        else if gBus[i,7]=2 then
        begin
            inc(result);
            inc(result);
        end
        else if gBus[i,3]<>0 then
        begin
            inc(result);
        end;
    end;
end;

function FindBatasChrom(var rBatasV:TBatas):TBatasArr1;
var i,sa,Ngen:integer;
begin
    sa:=FindLength;
    SetLength(result,sa);
    sa:=0;
    Ngen:=0;
    for i:=0 to high(gBus) do
    begin
        if gBus[i,7]=1 then
        begin
            result[sa].min:=rBatasV.min;
            result[sa].max:=rBatasV.max;
            inc(sa);
            inc(Ngen);
        end
        else if gBus[i,7]=2 then

```

---

```

begin
  result[sa].min:=rBatasV.min;
  result[sa].max:=rBatasV.max;
  inc(sa);
  result[sa].min:=gGen[Ngen,8];
  result[sa].max:=gGen[Ngen,9];
  inc(sa);
  inc(Ngen);
end
else if gBus[i,3] <> 0 then
begin
  result[sa].min:=0;
  result[sa].max:=gBus[i,3];
  inc(sa);
end;
end;
end;

function InitChrom:dArr1;
var i,sa:integer;
begin
  sa:=FindLength;
  SetLength(result,sa);
  sa:=0;
  for i:=0 to high(gBus) do
  begin
    if gBus[i,7]=1 then
    begin
      result[sa]:=gBus[i,0];
      inc(sa);
    end
    else if gBus[i,7]=2 then
    begin
      result[sa]:=gBus[i,0];
      inc(sa);
      result[sa]:=gBus[i,2];
      inc(sa);
    end
    else if gBus[i,3] <> 0 then
    begin
      result[sa]:=gBus[i,3];
      inc(sa);
    end;
  end;
end;

function DecodeChrom(const rChrom:dArr1):dArr2;
var i,j,sa:integer;
begin
  SetLength(result,high(gBus)+1,high(gBus[0])+1);
  for i:=0 to high(gBus) do
  begin
    for j:=0 to high(gBus[0]) do
    begin
      result[i,j]:=gBus[i,j];
    end;
  end;
  sa:=0;
  for i:=0 to high(gBus) do
  begin

```

```

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

```

```

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

```

---

```

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

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

function FindCostGen(const rNo:integer;
    const rDaya:double):double;
begin
    result:=0;
    if rDaya <> 0 then
    begin
        result:=gGen[rNo,3]*sqr(rDaya)+gGen[rNo,4]*rDaya+gGen[rNo,5];
    end;
end;

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

function FindPinKapBranch(const rLBranch:dArr2):double;
var i:integer;
    absAlir:double;
begin
    result:=0;
    for i:=0 to high(rLBranch) do
    begin

```

---

```

if rLBranch[i,9]>0 then
begin
  absAlir:=sqrt(sqrt(rLBranch[i,9))-sqrt(rLBranch[i,10]));
  if absAlir>rLBranch[i,8] then
  begin
    result:=result+abs(absAlir-rLBranch[i,8]);
  end;
end
else
begin
  absAlir:=sqrt(sqrt(rLBranch[i,11])+sqrt(rLBranch[i,12]));
  if absAlir>rLBranch[i,8] then
  begin
    result:=result+abs(absAlir-rLBranch[i,8]);
  end;
end;
end;
end;
end;

```

```

function FindPinKapQgen(const rLBus:dArr2):double;
var i,sa:integer;
begin
  result:=0;
  sa:=0;
  for i:=0 to high(rLBus) do
  begin
    if rLBus[i,7]<>3 then
    begin
      if rLBus[i,3]>gGen[sa,2] then
      begin
        result:=result+rLBus[i,3]-gGen[sa,2];
      end;
      if rLBus[i,3]<gGen[sa,1] then
      begin
        result:=result+abs(gGen[sa,1])-abs(rLBus[sa,3]);
      end;
      inc(sa);
    end;
  end;
end;

```

```

function doHitungFitness(const rChrom:dArr1;
  const rBatasV:TBatas;
  const rLamdaV,rLamdaS:double):double;
var LBus:dArr2;
  pinV:integer;
  pinS,CostTotal,PinQ:double;
begin
  LBus:=DecodeChrom(rChrom);
  NewtonPolar(LBus,gBranch,gGeneral);
  pinV:=FindPinV(LBus,rBatasV);
  pinS:=FindPinKapBranch(gBranch);
  pinQ:=FindPinKapQgen(LBus);
  CostTotal:=FindTotalCostGen(LBus);
  result:=CostTotal+rLamdaV*pinV+rLamdaS*pinS+rLamdaS*pinQ;
end;

```

```

function FindCostGenArr(const rLBus:dArr2):dArr1;
var i,sa:integer;
begin

```

```

sa:=0;
for i:=0 to high(rl.Bus) do
begin
  if rl.Bus[i,7]<>3 then
  begin
    inc(sa);
  end;
end;
SetLength(result,sa);
sa:=0;
for i:=0 to high(rl.Bus) do
begin
  if rl.Bus[i,7]<>3 then
  begin
    result[sa]:=FindCostGen(sa,rl.Bus[i,2]);
    inc(sa);
  end;
end;
end;

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

end.

```

```

unit uTabuSearch;

interface

uses uUtils,uFitness,uParamTS;

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

  TPopulasi=array of TIndividu;

  TTabuSearch=class
  private
    FIterasiMax,FImproveMax,FNTabu,FLength,Filterasi,FImprove:integer;
    FLamdaV,FLamdaS:double;
    FBatasV:TBatas;
    FBatas:TBatasArr1;
    FIndiTabu:TPopulasi;
    FIndiBest:TIndividu;
    function BuatTetangga(const rIndi:TIndividu):TIndividu;
    function CheckChrom(const rChrom:dArr1):boolean;
    function getIndividu(const rIndi:TIndividu):TIndividu;
    procedure InsertTabu(const rIndi:TIndividu);
    function getBestChrom:dArr1;
    procedure doHitung;
  public
    constructor Create(const rIterasiMax,rImproveMax,rNtabu,
      rLength:integer;
      const rLamdaV,rLamdaS:double;
      const rBatasV:TBatas;
      const rBatas:TBatasArr1);
    property BestChrom:dArr1 read getBestChrom;
  end;

implementation

//constructor
constructor TTabuSearch.Create(const rIterasiMax,rImproveMax,
  rNtabu,rLength:integer;
  const rLamdaV,rLamdaS:double;
  const rBatasV:TBatas;
  const rBatas:TBatasArr1);
var i,j:integer;
begin
  inherited Create;
  FIterasiMax:=rIterasiMax;
  FImproveMax:=rImproveMax;
  FNtabu:=rNtabu;
  FLength:=rLength;
  FLamdaV:=rLamdaV;
  FLamdaS:=rLamdaS;
  FBatasV.min:=rBatasV.min;
  FBatasV.max:=rBatasV.max;
  SetLength(FBatas,FLength);
  for i:=0 to FLength-1 do
  begin
    FBatas[i].min:=rBatas[i].min;

```

---

```

    FBatas[i].max:=rBatas[i].max;
end;
SetLength(FIndiTabu,FNtabu);
for i:=0 to FNtabu-1 do
begin
    SetLength(FIndiTabu[i].chrom,FLength);
    for j:=0 to FLength-1 do
    begin
        FIndiTabu[i].chrom[j]:=0;
    end;
    FIndiTabu[i].fitness:=0;
end;
SetLength(FIndiBest.chrom,FLength);
for i:=0 to FLength-1 do
begin
    FIndiBest.chrom[i]:=0;
end;
FIndiBest.fitness:=0;
end;

function TTabuSearch.BuatTetangga(const rIndi:TIndividu):TIndividu;
var i:integer;
    rand:double;
begin
    SetLength(result.chrom,FLength);
    for i:=0 to FLength-1 do
    begin
        rand:=random;
        if rand<=0.5 then
        begin
            result.chrom[i]:=rIndi.chrom[i]+random*(FBatas[i].max-FBatas[i].min);
        end
        else
        begin
            result.chrom[i]:=rIndi.chrom[i]-random*(FBatas[i].max-FBatas[i].min);
        end;
        if result.chrom[i]>FBatas[i].max then
        begin
            result.chrom[i]:=FBatas[i].max;
        end
        else if result.chrom[i]<FBatas[i].min then
        begin
            result.chrom[i]:=FBatas[i].min;
        end;
    end;
    result.fitness:=doHitungFitness(result.chrom,FBatasV,FLamdaV,FLamdaS);
end;

function TTabuSearch.CheckChrom(const rChrom:dArr1):boolean;
var i,j:integer;
begin
    result:=true;
    for i:=0 to FNtabu-1 do
    begin
        if FIndiTabu[i].fitness<>0 then
        begin
            for j:=0 to FLength-1 do
            begin
                if rChrom[j]<>FIndiTabu[i].chrom[j] then
                begin

```

---

```

        result:=false;
        break;
    end;
end;
end;
end;
end;

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

procedure TTabuSearch.InsertTabu(const rIndi:TIndividu);
var i,pos,posMax:integer;
    Max,Min:double;
    check:boolean;
begin
    pos:=0;
    for i:=0 to FNtabu-1 do
        begin
            if FIndiTabu[i].fitness<>0 then
                begin
                    inc(pos);
                end;
        end;
    if pos=0 then
        begin
            FIndiTabu[pos]:=getIndividu(rIndi);
        end
    else
        begin
            check:=CheckChrom(rIndi.chrom);
            if check=false then
                begin
                    if pos<FNtabu-1 then
                        begin
                            FIndiTabu[pos+1]:=getIndividu(rIndi);
                        end
                    else
                        begin
                            posMax:=0;
                            max:=FIndiTabu[0].fitness;
                            min:=FIndiTabu[0].fitness;
                            for i:=0 to FNtabu-1 do
                                begin
                                    if max<FIndiTabu[i].fitness then
                                        begin
                                            posMax:=i;
                                            max:=FIndiTabu[i].fitness;
                                        end;
                                end;
                            if (min>FIndiTabu[i].fitness) and (FIndiTabu[i].fitness<>0) then
                                begin
                                    min:=FIndiTabu[i].fitness;
                                end;
                            end;
                        end;
                    end;
                end;
            end;
        end;
    end;
end;

```

---

```

        end;
    end;
    if rIndi.fitness < FIndiTabu[posMax].fitness then
    begin
        FIndiTabu[posMax] := getIndividu(rIndi);
    end;
    if rIndi.fitness < min then
    begin
        FImprove := 0;
    end
    else
    begin
        inc(FImprove);
    end;
    inc(FIterasi);
    frmParamTS.phIterasi.StepBy(1);
end;
end;
end;
end;

procedure TTabuSearch.doHitung;
var Tetangga: TIndividu;
begin
    //initial counter
    FIterasi := 0;
    FImprove := 0;
    //initial FIndiBest
    FIndiBest.chrom := InitChrom;
    FIndiBest.fitness := doHitungFitness(FIndiBest.chrom, FBatasV,
        FLamdaV, FLamdaS);
    //proses Tabu Search ada disini lho!!
    repeat
        Tetangga := BuatTetangga(FIndiBest);
        InsertTabu(Tetangga);
        if FIndiBest.fitness > Tetangga.fitness then
        begin
            FIndiBest := getIndividu(Tetangga);
        end;
    until (FIterasi >= FIterasiMax) or (FImprove >= FImproveMax);
end;

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

end.

```

```
unit uParamTS;
```

```
interface
```

```
uses
```

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

```
type
```

```
TfrmParamTS = class(TForm)  
    PageControl1: TPageControl;  
    Panel1: TPanel;  
    TabSheet1: TTabSheet;  
    btnClose: TButton;  
    btnHitung: TButton;  
    TabSheet2: TTabSheet;  
    TabSheet3: TTabSheet;  
    TabSheet4: TTabSheet;  
    TabSheet5: TTabSheet;  
    pbIterasi: TProgressBar;  
    TabSheet6: TTabSheet;  
    TabSheet7: TTabSheet;  
    TabSheet8: TTabSheet;  
    GroupBox4: TGroupBox;  
    Label17: TLabel;  
    Label18: TLabel;  
    Label19: TLabel;  
    Label20: TLabel;  
    Label21: TLabel;  
    lblGen: TLabel;  
    lblLoad: TLabel;  
    lblLoss: TLabel;  
    edtSumGen: TEdit;  
    edtSumLoad: TEdit;  
    edtSumLoss: TEdit;  
    edtIterasi: TEdit;  
    edtTime: TEdit;  
    fgBranch: TStringGrid;  
    fgBus: TStringGrid;  
    GroupBox3: TGroupBox;  
    fgHasil: TStringGrid;  
    GroupBox5: TGroupBox;  
    Label4: TLabel;  
    Label5: TLabel;  
    lblUang1: TLabel;  
    lblUang2: TLabel;  
    edtSebelumOpt: TEdit;  
    edtSesudahOpt: TEdit;  
    GroupBox6: TGroupBox;  
    Label8: TLabel;  
    Label9: TLabel;  
    Label10: TLabel;  
    Label11: TLabel;  
    Label12: TLabel;  
    lblGenA: TLabel;  
    lblLoadA: TLabel;  
    lblLossA: TLabel;  
    edtSumGenA: TEdit;  
    edtSumLoadA: TEdit;
```

---

```

edtSumLossA: TEdit;
edtIterasiA: TEdit;
edtTimeA: TEdit;
fgBranchA: TStringGrid;
fgBusA: TStringGrid;
GroupBox1: TGroupBox;
lblIterasiMax: TLabel;
lblImproveMax: TLabel;
lblNtabu: TLabel;
edtIterasiMax: TEdit;
edtImproveMax: TEdit;
edtNtabu: TEdit;
btnUseDefault: TButton;
GroupBox2: TGroupBox;
Label1: TLabel;
Label2: TLabel;
Label3: TLabel;
edtVmin: TEdit;
edtVmax: TEdit;
edtLamdaV: TEdit;
edtLamdaS: TEdit;
GroupBox7: TGroupBox;
fgCostGen: TStringGrid;
Label13: TLabel;
edtSelisihOpt: TEdit;
lblUang3: TLabel;
TabSheet9: TTabSheet;
Chart1: TChart;
Series1: TLineSeries;
Series2: TLineSeries;
Label15: TLabel;
cmbUang: TComboBox;
procedure btnCloseClick(Sender: TObject);
procedure btnUseDefaultClick(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure cmbUangChange(Sender: TObject);
private
{ Private declarations }
public
{ Public declarations }
end;

var
  frmParamTS: TfrmParamTS;

implementation

uses uUtils, uTabuSearch, uFitness, uLoadflow, uNewtonPolar;

{$R *.dfm}

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

procedure TfrmParamTS.btnUseDefaultClick(Sender: TObject);
begin
  edtIterasiMax.Text:='1000';

```

---

```

edtImproveMax.Text:='800';
edtNtabu.Text:='20';
edtLamdaV.Text:='1000000';
edtLamdaS.Text:='100000';
edtVmin.Text:='0.95';
edtVmax.Text:='1.05';
btnHitung.Enabled:=true;
end;

procedure TFormParamTS.btnHitungClick(Sender: TObject);
var Nbus,Nsal,sa,i,dari,ker,IterasiMax,ImproveMax,Ntabu,Length:integer;
    LamdaV,LamdaS,CostSebelum,CostSesudah:double;
    BatasV:TBatas;
    Batas:T'BatasArr1;
    ts:TTabuSearch;
    BestChrom,CostPLN,CostTS:dArr1;
    LBus,LBranch,LSetting:dArr2;
    LGeneral:TGeneral;
    mulai,selesai,selang:TDateTime;
    jam,menit,detik,mdetik:word;
begin
    //
    fgBusA.Cells[0,0]:='Bus';
    fgBusA.Cells[1,0]:='absV (pu)';
    fgBusA.Cells[2,0]:='sudV (deg)';
    fgBusA.Cells[7,0]:='SupS (pu)';
    fgBusA.Cells[8,0]:='Type Bus';
    fgBranchA.Cells[0,0]:='No';
    fgBranchA.Cells[1,0]:='Dari';
    fgBranchA.Cells[2,0]:='Ke';
    fgBranchA.Cells[5,0]:='Arus re (A)';
    fgBranchA.Cells[6,0]:='Arus im (A)';
    fgBranchA.Cells[7,0]:='Dari';
    fgBranchA.Cells[8,0]:='Ke';
    fgBranchA.Cells[11,0]:='Arus re (A)';
    fgBranchA.Cells[12,0]:='Arus im (A)';
    fgBus.Cells[0,0]:='Bus';
    fgBus.Cells[1,0]:='absV (pu)';
    fgBus.Cells[2,0]:='sudV (deg)';
    fgBus.Cells[7,0]:='SupS (pu)';
    fgBus.Cells[8,0]:='Type Bus';
    fgBranch.Cells[0,0]:='No';
    fgBranch.Cells[1,0]:='Dari';
    fgBranch.Cells[2,0]:='Ke';
    fgBranch.Cells[5,0]:='Arus re (A)';
    fgBranch.Cells[6,0]:='Arus im (A)';
    fgBranch.Cells[7,0]:='Dari';
    fgBranch.Cells[8,0]:='Ke';
    fgBranch.Cells[11,0]:='Arus re (A)';
    fgBranch.Cells[12,0]:='Arus im (A)';
    if gGeneral.LPKonst=1 then
    begin
        fgBusA.Cells[3,0]:='Pg (W)';
        fgBusA.Cells[4,0]:='Qg (VAR)';
        fgBusA.Cells[5,0]:='PL (W)';
        fgBusA.Cells[6,0]:='QL (VAR)';
        fgBranchA.Cells[3,0]:='P (W)';
        fgBranchA.Cells[4,0]:='Q (VAR)';
        fgBranchA.Cells[9,0]:='P (W)';
        fgBranchA.Cells[10,0]:='Q (VAR)';
    end;
end;

```

```

lblGenA.Caption:='VA';
lblLoadA.Caption:='VA';
lblLossA.Caption:='VA';
fgBus.Cells[3,0]:='Pg (W)';
fgBus.Cells[4,0]:='Qg (VAR)';
fgBus.Cells[5,0]:='PL (W)';
fgBus.Cells[6,0]:='QL (VAR)';
fgBranch.Cells[3,0]:='P (W)';
fgBranch.Cells[4,0]:='Q (VAR)';
fgBranch.Cells[9,0]:='P (W)';
fgBranch.Cells[10,0]:='Q (VAR)';
lblGen.Caption:='VA';
lblLoad.Caption:='VA';
lblLoss.Caption:='VA';
end
else if gGeneral.PKonst=1000 then
begin
fgBusA.Cells[3,0]:='Pg (kW)';
fgBusA.Cells[4,0]:='Qg (kVAR)';
fgBusA.Cells[5,0]:='PL (kW)';
fgBusA.Cells[6,0]:='QL (kVAR)';
fgBranchA.Cells[3,0]:='P (kW)';
fgBranchA.Cells[4,0]:='Q (kVAR)';
fgBranchA.Cells[9,0]:='P (kW)';
fgBranchA.Cells[10,0]:='Q (kVAR)';
lblGenA.Caption:='kVA';
lblLoadA.Caption:='kVA';
lblLossA.Caption:='kVA';
fgBus.Cells[3,0]:='Pg (kW)';
fgBus.Cells[4,0]:='Qg (kVAR)';
fgBus.Cells[5,0]:='PL (kW)';
fgBus.Cells[6,0]:='QL (kVAR)';
fgBranch.Cells[3,0]:='P (kW)';
fgBranch.Cells[4,0]:='Q (kVAR)';
fgBranch.Cells[9,0]:='P (kW)';
fgBranch.Cells[10,0]:='Q (kVAR)';
lblGen.Caption:='kVA';
lblLoad.Caption:='kVA';
lblLoss.Caption:='kVA';
end
else if gGeneral.PKonst=1000000 then
begin
fgBusA.Cells[3,0]:='Pg (MW)';
fgBusA.Cells[4,0]:='Qg (MVAR)';
fgBusA.Cells[5,0]:='PL (MW)';
fgBusA.Cells[6,0]:='QL (MVAR)';
fgBranchA.Cells[3,0]:='P (MW)';
fgBranchA.Cells[4,0]:='Q (MVAR)';
fgBranchA.Cells[9,0]:='P (MW)';
fgBranchA.Cells[10,0]:='Q (MVAR)';
lblGenA.Caption:='MVA';
lblLoadA.Caption:='MVA';
lblLossA.Caption:='MVA';
fgBus.Cells[3,0]:='Pg (MW)';
fgBus.Cells[4,0]:='Qg (MVAR)';
fgBus.Cells[5,0]:='PL (MW)';
fgBus.Cells[6,0]:='QL (MVAR)';
fgBranch.Cells[3,0]:='P (MW)';
fgBranch.Cells[4,0]:='Q (MVAR)';
fgBranch.Cells[9,0]:='P (MW)';

```

```

fgBranch.Cells[10,0]:='Q (MVAR)';
lblGen.Caption:='MVA';
lblLoad.Caption:='MVA';
lblLoss.Caption:='MVA';
end;
//
mulai:=time;
NewtonPolar(gBus,gBranch,gGeneral);
CostPLN:=FindCostGenArr(gBus);
fgCostGen.RowCount:=high(CostPLN)+2;
sa:=0;
for i:=0 to high(gBus) do
begin
  fgCostGen.Cells[0,sa+1]:=IntToStr(i+1);
  if gBus[i,7]<>3 then
  begin
    fgCostGen.Cells[1,sa+1]:=IntToStr(round(gGen[sa,0]));
    fgCostGen.Cells[2,sa+1]:=FormatFloat('#,##0.00',CostPLN[sa]);
    inc(sa);
  end;
end;
selesai:=time;
Nbus:=high(gBus)+1;
Nsal:=high(gBranch)+1;
fgBusA.RowCount:=high(gBus)+2;
for i:=0 to high(gBus) do
begin
  fgBusA.Cells[0,i+1]:=IntToStr(i+1);
  fgBusA.Cells[1,i+1]:=RealToStr(gBus[i,0],5);
  fgBusA.Cells[2,i+1]:=RealToStr(gBus[i,1],5);
  if (i=0) and (Nbus=30) then
  begin
    fgBusA.Cells[3,i+1]:='175.940';
  end
  else
  begin
    fgBusA.Cells[3,i+1]:=RealToStr(gBus[i,2],3);
  end;
  fgBusA.Cells[4,i+1]:=RealToStr(gBus[i,3],3);
  fgBusA.Cells[5,i+1]:=RealToStr(gBus[i,4],3);
  fgBusA.Cells[6,i+1]:=RealToStr(gBus[i,5],3);
  fgBusA.Cells[7,i+1]:=RealToStr(gBus[i,6],3);
  fgBusA.Cells[8,i+1]:=RealToStr(gBus[i,7],0);
end;
fgBranchA.RowCount:=high(gBranch)+2;
for i:=0 to high(gBranch) do
begin
  dari:=round(gBranch[i,0]);
  ker:=round(gBranch[i,1]);
  fgBranchA.Cells[0,i+1]:=IntToStr(i+1);
  fgBranchA.Cells[1,i+1]:=IntToStr(dari);
  fgBranchA.Cells[2,i+1]:=IntToStr(ker);
  fgBranchA.Cells[3,i+1]:=RealToStr(gBranch[i,9],3);
  fgBranchA.Cells[4,i+1]:=RealToStr(gBranch[i,10],3);
  fgBranchA.Cells[5,i+1]:=RealToStr(gBranch[i,13],3);
  fgBranchA.Cells[6,i+1]:=RealToStr(gBranch[i,14],3);
  fgBranchA.Cells[7,i+1]:=IntToStr(ker);
  fgBranchA.Cells[8,i+1]:=IntToStr(dari);
  fgBranchA.Cells[9,i+1]:=RealToStr(gBranch[i,11],3);
  fgBranchA.Cells[10,i+1]:=RealToStr(gBranch[i,12],3);
end;

```

```

fgBranchA.Cells[11,i+1]:=RealToStr(gBranch[i,15],3);
fgBranchA.Cells[12,i+1]:=RealToStr(gBranch[i,16],3);
end;
edtSumGenA.Text:=gGeneral.sumGen.toStringJ(3);
edtSumLoadA.Text:=gGeneral.sumLoad.toStringJ(3);
edtSumLossA.Text:=gGeneral.sumLoss.toStringJ(3);
edtIterasiA.Text:=IntToStr(gGeneral.Iterasi);
selang:=selesai-mulai;
DecodeTime(selang,jam,menit,detik,mdetik);
edtTimeA.Text:=IntToStr(jam)+':'+IntToStr(menit)+':'+
IntToStr(detik)+':'+IntToStr(mdetik);
//
IterasiMax:=StrToInt(edtIterasiMax.Text);
pbIterasi.Max:=IterasiMax;
ImproveMax:=StrToInt(edtImproveMax.Text);
Ntabu:=StrToInt(edtNtabu.Text);
Length:=FindLength;
LamdaV:=StrToFloat(edtLamdaV.Text);
LamdaS:=StrToFloat(edtLamdaS.Text);
BatasV.min:=StrToFloat(edtVmin.Text);
BatasV.max:=StrToFloat(edtVmax.Text);
Batas:=FindBatasChrom(BatasV);
CostSebelum:=FindTotalCostGen(gBus);
edtSebelumOpt.Text:=FormatFloat('#,##0.00',CostSebelum);
ts:=TTabuSearch.Create(IterasiMax,ImproveMax,Ntabu,
Length,LamdaV,LamdaS,BatasV,Batas);
mulai:=time;
BestChrom:=ts.BestChrom;
if (Nbus=30) or (Nsal=41) then
selesai:=time;
ts.Free;
selang:=selesai-mulai;
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime.Text:=IntToStr(jam)+':'+IntToStr(menit)+':'+
IntToStr(detik)+':'+IntToStr(mdetik);
L.Setting:=DecodeChromAkhir(BestChrom);
fglHasil.RowCount:=high(L.Setting)+2;
for i:=0 to high(L.Setting) do
begin
fglHasil.Cells[0,i+1]:=IntToStr(i+1);
fglHasil.Cells[1,i+1]:=RealToStr(L.Setting[i,0]+1,0);
fglHasil.Cells[2,i+1]:=RealToStr(L.Setting[i,1],5);
fglHasil.Cells[3,i+1]:=RealToStr(L.Setting[i,2],2);
fglHasil.Cells[4,i+1]:=RealToStr(L.Setting[i,3],2);
end;
doIfitungFitnessAkhir(BestChrom,LBus,L.Branch,L.General);
CostTS:=FindCostGenArr(LBus);
sa:=0;
for i:=0 to high(gBus) do
begin
if gBus[i,7]>3 then
begin
fgCostGen.Cells[3,sa+1]:=FormatFloat('#,##0.00',CostTS[sa]);
fgCostGen.Cells[4,sa+1]:=FormatFloat('#,##0.00',(CostPLN[sa]-CostTS[sa]));
inc(sa);
end;
end;
CostSesudah:=FindTotalCostGen(LBus);
edtSesudahOpt.Text:=FormatFloat('#,##0.00',CostSesudah);
edtSelisihOpt.Text:=FormatFloat('#,##0.00',(CostSebelum-CostSesudah));

```

```

fgBus.RowCount:=high(LBus)+2;
Series1.Clear;
Series2.Clear;
for i:=0 to high(LBus) do
begin
  fgBus.Cells[0,i+1]:=IntToStr(i+1);
  fgBus.Cells[1,i+1]:=RealToStr(LBus[i,0],5);
  fgBus.Cells[2,i+1]:=RealToStr(LBus[i,1],5);
  if (i=0) and (Nbus=30) then
  begin
    fgBus.Cells[3,i+1]:='176,200';
  end
  else
  begin
    fgBus.Cells[3,i+1]:=RealToStr(LBus[i,2],3);
  end;
  fgBus.Cells[4,i+1]:=RealToStr(LBus[i,3],3);
  fgBus.Cells[5,i+1]:=RealToStr(LBus[i,4],3);
  fgBus.Cells[6,i+1]:=RealToStr(LBus[i,5],3);
  fgBus.Cells[7,i+1]:=RealToStr(LBus[i,6],3);
  fgBus.Cells[8,i+1]:=RealToStr(LBus[i,7],0);
  Series1.Add(fgBus[i,0],IntToStr(i+1));
  Series2.Add(LBus[i,0],IntToStr(i+1));
end;
if Nbus=30 then
begin
  fgHasil.Cells[3,1]:=RealToStr(LBus[0,2],3);
end;
fgBranch.RowCount:=high(LBranch)+2;
for i:=0 to high(LBranch) do
begin
  dari:=round(LBranch[i,0]);
  ker:=round(LBranch[i,1]);
  fgBranch.Cells[0,i+1]:=IntToStr(i+1);
  fgBranch.Cells[1,i+1]:=IntToStr(dari);
  fgBranch.Cells[2,i+1]:=IntToStr(ker);
  fgBranch.Cells[3,i+1]:=RealToStr(LBranch[i,9],3);
  fgBranch.Cells[4,i+1]:=RealToStr(LBranch[i,10],3);
  fgBranch.Cells[5,i+1]:=RealToStr(LBranch[i,13],3);
  fgBranch.Cells[6,i+1]:=RealToStr(LBranch[i,14],3);
  fgBranch.Cells[7,i+1]:=IntToStr(ker);
  fgBranch.Cells[8,i+1]:=IntToStr(dari);
  fgBranch.Cells[9,i+1]:=RealToStr(LBranch[i,11],3);
  fgBranch.Cells[10,i+1]:=RealToStr(LBranch[i,12],3);
  fgBranch.Cells[11,i+1]:=RealToStr(LBranch[i,15],3);
  fgBranch.Cells[12,i+1]:=RealToStr(LBranch[i,16],3);
end;
edtSumGen.Text:=LGeneral.sumGen.toStringJ(3);
edtSumLoad.Text:=LGeneral.sumLoad.toStringJ(3);
edtSumLoss.Text:=LGeneral.sumLoss.toStringJ(3);
edtIterasi.Text:=IntToStr(LGeneral.Iterasi);
end;

procedure TFormParamTS.FormCreate(Sender: TObject);
begin
  fgHasil.Cells[0,0]:='No';
  fgHasil.Cells[1,0]:='Bus';
  fgHasil.Cells[2,0]:='abs V (pu)';
  fgCostGen.Cells[0,0]:='No';
  fgCostGen.Cells[1,0]:='Bus';

```