

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



**ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL  
DENGAN METODE PARALLEL SIMULATED  
ANNEALING PADA PT PJB**

**SKRIPSI**

**Disusun Oleh:  
ROSMANSYAH  
NIM:00.12.065**

**APRIL 2005**



GENERAL INVESTIGATION DIVISION  
FEDERAL BUREAU OF INVESTIGATION  
U. S. DEPARTMENT OF JUSTICE  
WASHINGTON, D. C. 20535

MEMORANDUM FOR THE DIRECTOR, FBI  
FROM: SAC, NEW YORK (100-100000)  
SUBJECT: [REDACTED]

NY 100-100000

100-100000  
NY 100-100000  
100-100000

100-100000

**LEMBAR PERSETUJUAN**

**ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL  
DENGAN METODE PARALLEL SIMULATED ANNEALING  
PADA PT PJB**

**SKRIPSI**

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

Disusun oleh :  
**ROSMANSYAH**

00.12.065



Mengetahui,  
Ketua Jurusan Teknik Elektro

Menyetujui,  
Dosen Pembimbing

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

(Ir.H.Almizan Abdullah, MSEE)  
NIP : 103 900 0208

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

**2005**

## KATA MUTIARA

Carilah oleh kalian derajat yang tinggi disisi Allah yaitu hendaknya kamu bersikap penyantun terhadap orang yang tidak mengetahui tentang diri kamu, dan hendaknya kamu memberi kepada orang yang tidak pernah memberi kepadamu.

*Al Hadist riwayat 'Addi*

Menyebut-nyebut nikmat Allah adalah bersyukur, meninggalkan berarti ingkar. Barang siapa yang tidak mensyukuri nikmat yang sedikit berarti dia tidak mensyukuri nikmat yang banyak; dan barang siapa yang tidak bersyukur terhadap orang lain, berarti tidak bersyukur kepada Allah. Bersatu adalah rahmat, bercerai berai adalah azab.

*Al Hadist riwayat Baihaqi*

Rendah diri tidak menambah kepada seorang hamba kecuali pengangkatan, karena itu berendahdirilah kalian, niscaya Allah akan mengangkat derajat kalian. Memaafkan tidaklah menambah kepada seorang hamba kecuali kemuliaan, karena itu perbanyaklah maaf kalian niscaya Allah akan memuliakan kalian. Tiadalah sedekah itu melainkan makin menambah banyak harta, karena itu bersedekahlah kalian niscaya Allah merahmati kalian.

*Al Hadist riwayat 'Ibnu Abud Dun-ya*

Orang yang gagal ada dua macam yaitu mereka yang bertindak tanpa berfikir dan mereka yang berfikir tanpa bertindak,

*John Charles Salak*

Manakala seseorang tidak tahu mana pelabuhan yang ditujunya, tidak ada angin yang menjadi angin baik,

*Lucius Anneaeus Seneca*

Semakin keras sikap anda terhadap diri sendiri, semakin mudah hidup memperlakukan anda.

*Zig Ziglar*

## LEMBAR PERSEMBAHAN

Assalamualaikum warahmatullahi wabarakatuh.

Alhamdulillah, empat tahun bukanlah waktu yang singkat, banyak sejarah hidup yang telah tercatat. Karena itu ijinkanlah saya menulis sebagian nama-nama yang tercatat dalam sejarah tersebut :

### A. Social

*Bapak (Alm H. Muhammad Rusmi) & Ibu (Hj. Kursiah)*

TERIMA KASIH yang seganal-ganalnya hagan ABAH lawan MAMA. pian yang mengganalakan ulun, membari makani ulun, maongkosi ulun, maurusi ulun, menyayangi ulun dan segalanya tu pang. Abah dan Mama Mansyah sudah lulus. Ulun kawa jua jadi urang yang pian handaki, amanat pian Bah ai sudah ulun laksanakan.

Mudahan ulun termasuk orang yang berbakti kepada Pian lawan Mama dan Pian dapat balasan yang lebih baik dari ALLAH SWT. Walaupun Pian Bahaai kada kawa melihat ulun secara langsung, tapi ulun yakin Pian lawan Mama bangga dan bahagia banar atas lulusnya ulun ini. Abah dan Mama mudah-mudahan dengan tuntung ulun ini, kawa membantu pian dalam segala hal. Terima kasih karena berakat doa pian ulun kawa kaya ini. Ulun minta Maaplah yang seganal-ganalnya atas segala dosa-dosa ulun lawan pian.

*Ya...Allah, ampunilah segala dosa kedua orangtua ku, terimalah semua amal ibadah mereka dan sayangilah mereka sebagaimana mereka menyayangi aku sewaktu aku masih kecil'*

*Isteriku (Intan Mayasari, SE)*

Ayang terima kasih atas cinta dan kasih sayang telah kau curahkan kepadaku serta telah kau percayakan aku untuk mendampingi mu seumur hidup. Ayang ku yang cantik dan manis, makaciiiih baaaanyak atas segala dukungan dan motifasinya. Ayang kaulah yang selama ini telah membuat diriku sangatlah berbahagia, kamu yang sering ngasih aku masukan-masukan yang positif pokonya kaulah bidadariku yang telah ALLAH anugerahkan kepadaku. Makacih ya!!!! Ayang yang telah sering nganterin aku ke mas ugro, bantuin ngerjain skripsiku siang dan malam ketempatnya Hari, ngelonin aku waktu tidur dan banyak lagi dech.....malu kalau dibuka semuanya takut dibaca orang.....he...he,....

Maafin atas segala kesalahan ku. Semoga kita jadi keluarga sakinah mawaddah warohmah segala usaha kita sukses demi anak dan cucu kita nanti amien ya robbal a'lamin.

Tolong jaga anak kita yang masih 3 bulan dalam kandungan mu. Bilangin ama mansah junior jangan rewel sebab ak yang ngidam, ak pagi dan malam muntah dan sakit perut. Semoga kita mendapat anak yang taat kepada ALLAH SWT, berbakti kpd kita berdua, baik kepada sesama, cakep atau cantik n berguna bg semuanya.

*Kakek & Nenek*

Makasih atas segala ilmu kebatinannya ulun salut banar lawan pian bedua. Doakan tarus ulun lah.

*Mertuaku ( Samsul Arifin & alm.Nurfidiyah )*

Ayah, terima kasih atas segala doanya dan atas segala motifasi yang telah ayah berikan. Maafin saya ya Yah, kalau ada kata-kata saya yang tidak berkenan di hati Ayah karena saya orangnya senang becanda, dan kadang-kadang kecoplosan. Tapi itu semua Ayah kok yang ngajarin he...he....

Yah.....kau adalah mertua yang paling baik dech, jarang ada mertua yang mudah diajak tuker pikiran dalam segala hal. Pokoknya Ayah adalah mertua is the best. Nanti mansyah yakin klu usaha Ayah dan Mansyah akan jadi besar.....Amiennnnnn...

Umi....hanya doa yang saya bisa panjatkan, semoga Allah mengampuni dosa dan menerima segala amal ibadah yang telah Umi lakukan.

*Kakakku & Ponakanku*

Buat kakakku (Rusidah dan Riduan Noor) makasih selalu mendokan, memberi semangat ulun, membantu mengurus keuangan ulun. Maafkanlah atas segala kesalahan ulun.

Rizky & Rikzan jangan nakallah, Jangan terlalu boros, kasihan mama lawan Nini mencarikan duitnya. Doakan paman supaya jadi orang yang sukses. Kena nyaman paman menyarikan gawian ikam.

*Om & TanteKu*

Makasihlah Paman hasan & Acil Lily atas doa dan dorongannya, buat Acil Tini sabar aja lah!!!!Mansyah minta maaf banar karena kada kawa datang waktu Paman Aran meninggal, Ulun hanya kawa mendoakan mudahan amal ibadah sidin diterima Allah SWT.Amien...

Makasih juga buat Julak laki lawan Julak bini & Paman Ancah, Paman Udin (kai),alm Paman Aran,Paman Aman,Acil Tini,Acil Yani, Acil Isam maksih atas segala doa dan dukungannya.

*Kakak Iparku*

Mas (Inul,Helmi,Zul) n mbak Iva makasih ya atas dukungan dan doanya semoga usaha kita akan semakin maju dan berkembang.

*Saudara Sepupuku*

Buat Nita(godong),Daus,Yuli,HiJrin,Alfi,Fahmi,Erwin,Ramadhan,Nouval rajin-rajin belajarlah jangan bemainan dan bepacaran tarus dangarkan pandiran urang tuha jangan sampai wani.

*Keluarga H. Kurdian*

Abah,Mama,Kakak Haji Yandi,Kakak Suma,Daffa,Naffa,dan Kakak Yadi, aku terima kasih banyaklah atas doa dan dukungannya.Haji makasihlah imak percaya lawan auk dan auk lawas bediam wadah imak pokoknya imak baik banar tu pang.

Mudahan imak lakas dapat emas atau berlian dan usahanya sukses selalu....berelaan lah

*Ponakanku Dimalang*

Nigel,Jhorgi,Letty,Wildan,Kemal,Gilang,dan Qinan jangan nakal ya. Taat pada orang tua dan yang rajin belajarnya, semoga kalian sukses selalu.

*Sahabat-sahabatku*

Ali (kantut) ikam luluslah. Parak am kita jadi baiparan....he...he... Andre (Aan) maju terus ama lila sampai pelaminan. Junait ui...capati meumpati kami ba 5. Aulia (Nini) imak jangan telalu baharap lawan urang, kalau pina tagugur....begayaan lia ai.... Tiny (Zack) kamu bisa tin...capati lulus & Surya Ndul rajini imak ke mas ugro biar lakas tuntung TA imak. Spesial buat Hari..."makasih banyak lah atas semua kerjasamanya, jangan jadi perajuan, kada baik".

Semoga kita selalu menjadi sahabat selamanya...maafkan atas kesalahanku selama ini, atas Segala perkataanku yang kurang berkenan dihati bubuhan ikam semunyaan, tapi aku yakin ikam tahu ja,bahwa aku tu sayang lawan bubuhan ikam semunyaan. Harapanku walaupun kita sudah lulusan tapi kita jangan sampai putus bahubungan.....terima kasih dingsanakai, semoga kita slalu dalam lindungan Allah SWT.

*Mas Ugro & Wife + Rafi*

Thanks ya.....telah membuatkan aku program DELPHI nya buat analisa perhitungan skripsiku dan atas semua bantuannya.

*AFC (Almizan Fans Club)*

• Temen-temen AFC ( Harry, Hanafi, Ina, Tamrin, Riza, Aang, Lingga, Pandu, Irfan, Dedy Gendut, Dedy 99, Robert, Herman, Bobby, Wahyu, dan ada 2 lagi yang namanya saya lupa.) Sesuai semboyan,.....Hidup Babe

**B. Academics**

*Bpk Dr. Ir. Abraham Lomi, MSEE*

Walaupun cuma dua semester menjadi dosen wali...secara tidak langsung Bpk punya peran dalam kelulusanku.

*Bpk Ir. I Made Wartanana, MT*

Terimakasih telah menyetujui proposal skripsi ku, walaupun sebelumnya sering menolak judul yang kuajukan. Tapi, dengan begitu aku jadi dapat bimbingan yang cocok buatku..he..he..!!!!

*Bpk Ir. H. Almizan Abdullah, MSEE*

• Terima kasih saya ucapkan yang sebesar-besarnya, semoga Allah membalas semua kebaikan yang telah bapak lakukan. Semoga saran-saran yang bapak berikan dapat saya laksanakan semuanya. Menurut saya bapak lah Dosen yag terbaik di ITN.

*Mas Jayeng*

Jujur aja, sampai saat ini aku belum tahu siapa nama mas yang sebenarnya, aku cuma ikut anak-anak manggil mas Jayeng. Maaf ya... aku sering ngerepotin mas selama aku kuliah di ITN, terutama saat aku lagi garap TA ku...makasihya.....!!!!

*Mas Yudy*

Mas, makasihya selalu ngasih tahu aku klo aku lagi bingung cari dosen...

*Bu Pudji*

Bu...he..he..mudah-mudahan ibu ga' jengkel karena aku selalu mondar-mandir ke ruangan ibu, jangan diambil hati ya. Makasih lo Bu.....!!!!

## **KATA PENGANTAR**

Dengan segala kerendahan hati, penulis mengucapkan puji syukur kepada Allah SWT karena hanya dengan lindungan, rahmat dan karuniaNya-lah penulis dapat menyelesaikan penyusunan skripsi ini, sebagai syarat untuk melengkapi dan memenuhi syarat mencapai gelar sarjana.

Skripsi yang berjudul **“ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL DENGAN MENGGUNAKAN PARALLEL SIMULATED ANNEALING PADA PT PJB”** ini tersusun juga atas bantuan dari berbagai pihak. Untuk itu pada kesempatan ini penulis menyampaikan terima kasih yang sebesar-besarnya kepada :

1. Bapak DR. Ir. Abraham Lomi, MSEE selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. I Made Wartana, MT selaku Pembantu Dekan 1 Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
4. Bapak Ir. F. Yudi Limpraptono, MT selaku Ketua Jurusan Teknik Elektro Energi Listrik S-1 Institut Teknologi Nasional Malang.
5. Bapak Ir. H. Almizan Abdullah, MSEE selaku Dosen Pembimbing yang telah meluangkan waktunya untuk membimbing penulis.
6. Seluruh Dosen dan Karyawan yang secara langsung maupun tidak langsung telah membantu terselesaikannya penyusunan Skripsi ini.



7. Ayah dan Ibu yang selalu memberikan dukungan lahir maupun batin kepada penulis.
8. Rekan-rekan Mahasiswa Teknik Elektro Energi Listrik yang telah membantu baik secara langsung maupun tidak langsung dalam penyusunan Skripsi ini.
9. Semua pihak yang tidak dapat penulis sebutkan satu per satu, yang telah membantu dalam penyelesaian skripsi ini.

Akhirnya, sebagai puncak dari tujuan penulisan skripsi ini adalah semoga skripsi ini dapat bermanfaat bagi kita semua.

Malang, April 2005

Penulis

## ABSTRAKSI

### ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL DENGAN METODE *PARALLEL SIMULATED ANNEALING* PADA PT. PEMBANGKIT JAWA-BALI

( Rosmansyah, 00.12.065, Teknik Elektro Energi Listrik )  
( Dosen Pembimbing : Ir. H. Almizan Abdullah, MSEE )

**Kata Kunci :** *Komitmen unit, Simulated Annealing, Komputasi Spekulatif, Parallel Simulated Annealing, Optimasi Biaya Pembangkitan.*

Adanya persoalan dalam menghadapi kebutuhan daya listrik yang tidak tetap dari waktu ke waktu, menimbulkan suatu permasalahan yaitu bagaimana mengoperasikan suatu sistem tenaga listrik yang selalu dapat memenuhi permintaan daya pada setiap saat, dengan kualitas baik dan harga yang murah. Oleh karena itu, komitmen Unit pada sistem tenaga listrik dimaksudkan untuk menentukan pola operasi unit pembangkit yang menghasilkan biaya operasi yang paling minimal. Dalam prakteknya masalah komitmen unit adalah optimasi kombinasi yang melibatkan banyak batasan seperti beban, spinning reserve, minimum up dan minimum down times dari unit pembangkit.

Skripsi ini menganalisis permasalahan *komitmen unit* atau penjadwalan operasi unit-unit pembangkit dalam melayani beban sistem selama periode waktu tertentu dengan menggunakan metode *Simulated Annealing* dengan Pendekatan *Komputasi Spekulatif*. 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 koefisien biaya bahan bakar (*Fuel Cost*), daya maksimum dan minimum, dan data pembebanan tiap jam tiap unit pembangkit.

Analisa dilakukan dengan bantuan program komputer dengan menggunakan bahasa pemrograman Delphi versi 8.0 dan telah sukses dicoba pada PT. Pembangkitan Jawa Bali yang terdiri dari 38 unit pembangkit, dimana telah berhasil dilakukan penghematan biaya pembangkitan untuk hari kamis, 4 Desember 2003 dengan metode *Parallel Simulated Annealing* menggunakan pendekatan *Komputasi Spekulatif* sebesar Rp. 15,155,177,773; total biaya PLN : Rp. 19,668,568,812 sehingga selisih biaya : Rp. 4,513,391,039

## DAFTAR ISI

	<b>Halaman</b>
<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PERSETUJUAN</b> .....	ii
<b>ABSTRAKSI</b> .....	iii
<b>KATA PENGHANTAR</b> .....	iv
<b>DAFTAR ISI</b> .....	vi
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR TABEL</b> .....	x
<b>DAFTAR GRAFIK</b> .....	xii
<b>BAB I PENDAHULUAN</b>	
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Tujuan.....	2
1.4. Batasan Masalah.....	3
1.5. Metodologi Penelitian .....	3
1.6. Sistematika Pembahasan .....	4
<b>BAB II LANDASAN TEORI</b>	
2.1. Sistem Tenaga Listrik.....	6
2.2. Karakteristik Pembangkit Listrik .....	
2.2.1. Karakteristik Masukkan-Keluaran .....	11

2.2.2.	Karakteristik Laju Tambahan	
	Biaya Bahan Bakar .....	13
2.3.	Komitmen Unit .....	13
2.3.1	Kendala Pada Komitmen Unit.....	16
2.3.2	Kendala Unit Termis .....	18
2.3.3	Biaya start-Up.....	19
2.4	Formulasi Masalah Komitmen Unit.....	20
2.5	Teori Dasar Simulated Annealing .....	21
2.5.1	Simulated Annealing (SA) .....	21
2.5.2	Konsep Parallel Simulated Annealing.....	23
2.5.3	Algoritma Simulated Annealing.....	23
2.5.4	Pendekatan Komputasi Spekulatif.....	27
2.6	Implementasi Dari Parallel SA.....	28

### **BAB III DATA JOURNAL DAN DATA SISTEM PEMBANGKIT**

#### **TERMALPT. PJB**

3.1.	Pendahuluan .....	29
3.2.	Data Journal.....	30
3.3.	Data Unit Pembangkit.....	31
3.4.	Aplikasi Parallel SA Dengan Kombinasi KS & SS Pada PT PJB.....	33
3.5.	Beban Sistem.....	35

**BAB IV ANALISA DATA DAN METODE PSA DENGAN  
KOMBINASI KS & SS PADA PT. PJB**

4.1. Penggunaan Program Komputer Menggunakan Metode PSA.....	37
4.2. Algoritma Program.....	37
4.3. Flowchart Algoritma Program.....	39
4.4. Validasi Program.....	40
4.5. Tampilan Program.....	44
4.6. Analisa Program Dan Hasil Perhitungan.....	54

**BAB V KESIMPULAN DAN SARAN**

5.1. Kesimpulan.....	69
5.2. Saran.....	69

**DAFTAR PUSTAKA**

**LAMPIRAN**

## DAFTAR GAMBAR

Gambar 2.1	Elemen Pokok Sistem Tenaga Listrik .....	7
Gambar 2.2	Unit Boiler Dan Generator .....	11
Gambar 2.3	Kurva Karakteristik Inout-Output Pembangkit Termal.....	12
Gambar 2.4	Kurva Karakteristik Laju Tambahan Biaya Bahan Bakar.....	13
Gambar 2.5	Sistem Interkoneksi .....	15
Gambar 2.7	Komputasi Spekulatif Dengan Dua Iterasi .....	27
Gambar 2.8	Komputasi Spekulatif Dengan Tiga Iterasi .....	27
Gambar 4.1	Flowchart SA Dengan Komputasi Spekulatif .....	39
Gambar 4.2	Menu Utama Hasil Uji Validasi Metode PSA Dengan KS ...	40
Gambar 4.3	Tampilan Data Untuk Validasi.....	41
Gambar 4.4	Data Generator Untuk Validasi .....	41
Gambar 4.5	Data Pembebanan Untuk validasi.....	42
Gambar 4.6	Hasil Perhitungan Untuk Daya Generator Pada Validasi.....	42
Gambar 4.7	Hasil Perhitungan Biaya Validasi.....	43
Gambar 4.8	Kurva Biaya Operasional Validasi.....	43
Gambar 4.9	Menu Utama.....	45
Gambar 4.10	Tampilan Hasil Tes Program Menggunakan Metode PSA Dengan KS .....	53

## **DAFTAR TABEL**

Tabel 3.1	Data Unit Generator Pada Journal.....	30
Tabel 3.2	Data Beban Unit Generator Pada Journal.....	30
Tabel 3.3	Data Unit Termal Pada PT PJB.....	31
Tabel 3.4	Data Biaya Dan Parameter Unit Termal Pada PT PJB.....	32
Tabel 3.5	Data Beban Unit Termal Pada PT PJB.....	36
Tabel 4.1	Perbandingan Biaya Hasil Journal Dengan hasil validasi .....	44
Tabel 4.2	Kombinasi Penjadwalan Unit Termal Pada PT PJB Kamis 4 Desember 2003 .....	55
Tabel 4.3	Kombinasi Penjadwalan Unit Termal Pada PT PJB Sabtu 6 Desember 2003.....	56
Tabel 4.4	Kombinasi Penjadwalan Unit Termal Pada PT PJB Minggu 7 Desember 2003 .....	57
Tabel 4.5	Kombinasi Penjadwalan Unit Pembangkit Termal Pada PT PJB Menggunakan Metode PSA Dengan KS Kamis, 4 Desember 2003 .....	58
Tabel 4.6	Kombinasi Penjadwalan Unit Pembangkit Termal Pada PT PJB Menggunakan Metode PSA Dengan KS Sabtu, 6 Desember 2003.....	59
Tabel 4.7	Kombinasi Penjadwalan Unit Pembangkit Termal Pada PT PJB Menggunakan Metode PSA Dengan KS Minggu, 7 Desember 2003 .....	60

<b>Tabel 4.8</b>	<b>Perbandingan Biaya Operasional Per Jam PT PJB Dengan Metode PSA Dengan KS Kamis, 4 Desember 2003 .....</b>	<b>62</b>
<b>Tabel 4.9</b>	<b>Perbandingan Biaya Operasional Per Jam PT PJB Dengan Metode PSA Dengan KS Sabtu, 6 Desember 2003.....</b>	<b>63</b>
<b>Tabel 4.10</b>	<b>Perbandingan Biaya Operasional Per Jam PT PJB Dengan Metode PSA Dengan KS Minggu, 7 Desember 2003 .....</b>	<b>64</b>
<b>Tabel 4.11</b>	<b>Perbandingan Total Biaya Operasional PT PJB Dengan Total Biaya Program.....</b>	<b>65</b>



## **DAFTAR GRAFIK**

<b>Grafik 4.1</b>	<b>Kurva Biaya Operasional PT PJB Metode PSA Dengan KS</b>	
	Kamis, 4 Desember 2003 .....	66
<b>Grafik 4.2</b>	<b>Kurva Biaya Operasional PT PJB Metode PSA Dengan KS</b>	
	Sabtu, 6 Desember 2003.....	67
<b>Grafik 4.3</b>	<b>Kurva Biaya Operasional PT PJB Metode PSA Dengan KS</b>	
	Minggu, 7 Desember 2003 .....	68

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Komitmen Unit pada sistem tenaga listrik dimaksudkan untuk menentukan pola operasi unit pembangkit yang menghasilkan biaya operasi yang paling minimal. Dalam prakteknya masalah komitmen unit adalah optimasi kombinasi yang melibatkan banyak batasan seperti beban, spinning reserve, minimum up dan minimum down times dari unit pembangkit.

Masalah komitmen unit dikarakteristikan dengan optimasi dengan banyak batasan, hal ini menimbulkan banyak riset untuk menyelesaikan masalah tersebut. Salah satu metode yang dapat menyelesaikan masalah tersebut adalah Simulated Annealing.

Simulated Annealing (SA) adalah teknik optimasi baru yang telah dibuktikan sangat efektif untuk menyelesaikan optimasi dengan banyak kombinasi. Penyederhanaan batasan membuat Algoritma SA menjadi sangat menarik untuk menyelesaikan masalah unit commitment.

Pada SA, masalah optimasi disimulasikan sebagai sebuah proses annealing. Proses alami dari optimasi didasari dari fenomena pendinginan metal secara pelan-pelan (annealing) membuktikan bahwa struktur metal dapat mencapai struktur kristal atau minimum energy. Proses yang alami sebuah transisi dari sebuah struktur dengan level energi  $E$  ke level energi  $E + \Delta E$  dapat terjadi dengan probabilitas yang nilainya diberikan oleh fungsi Boltzman  $e^{-\Delta/kT}$ .

Kemudian temperatur diturunkan dengan probabilitas yang ditentukan oleh konstanta Boltzman menjadi lebih rendah. Proses diatas dilaksanakan dengan pendinginan metal secara pelan-pelan untuk menghindari dari struktur kristal yang diumpamakan sebagai status lokal energi minimum dan dapat mencapai status global minimum energi.

## **1.2. Rumusan Masalah**

Dalam mengoperasikan pusat-pusat pembangkit listrik, diperlukan optimasi biaya operasional pembangkit dengan menekannya serendah mungkin. Untuk itu diperlukan penentuan penjadwalan unit pembangkit yang ekonomis. Berdasarkan uraian pada latar belakang timbul pertanyaan seberapa besar kemampuan perhitungan metode *parallel SA* dengan *Komputasi Spekulatif (KS)*, untuk menyelesaikan masalah komitmen unit pada PT. PJB.

Berdasarkan gambaran permasalahan tersebut maka skripsi ini diberi judul:

**“ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL DENGAN METODE PARALLEL SIMULATED ANNEALING PADA PT PJB”.**

## **1.3. Tujuan**

Tujuan dari pembahasan skripsi ini adalah menerapkan suatu metode alternatif untuk menentukan pola operasi unit pembangkit thermal, menentukan besar pembangkitkan unit pembangkit thermal yang on dan mencari biaya pembangkitan perjam dan total biaya pembangkitan.

#### 1.4. Batasan Masalah

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

1. Penjadwalan dilakukan dalam satu hari (24 jam dengan range tiap jam) pada periode studi pada tanggal 4, 6 dan 7 Desember 2003.
2. Tidak membahas masalah rugi-rugi saluran transmisi.
3. Tidak membahas *combined cycle* pada PLTGU.
4. Untuk ST (*steam turbin*) pada *combined cycle*, diambil data parameter dari pola PLTGU CC-3.3.1. yang beroperasi.
5. Tidak membahas masalah biaya cadangan berputar (*spinning reserve*), hanya memperhatikan kendala batasan cadangan berputar.
6. Tidak ada biaya untuk *shut down* unit.

#### 1.5. Metodologi Penelitian

Metodologi yang digunakan dalam pembahasan dilaksanakan dengan langkah-langkah sebagai berikut:

- Studi kepustakaan mengenai hal-hal yang berhubungan dengan pembahasan masalah.
- Studi lapangan untuk mendapatkan data parameter unit termal yang dibutuhkan dari objek penelitian yaitu pada PT. PJB yang diperlukan berpedoman pada teori yang diperoleh dari studi kepustakaan.
- Menentukan optimasi penjadwalan pembebanan pada komitmen unit menggunakan metode *parallel simulated annealing* dengan Komputasi

*Spekulatif (KS)*. Membuat evaluasi, sehingga dapat disimpulkan apakah metode yang diterapkan lebih efisien atau ekonomis.

## **1.6. Sistematika Pembahasan**

Adapun sistematika pembahasan pada skripsi ini adalah sebagai berikut :

### **Bab I : PENDAHULUAN.**

Didalamnya berisi tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metologi penelitian serta sistematika pembahasan.

### **Bab II : TINJAUAN PUSTAKA.**

Berisi antara lain tentang pengenalan sistem tenaga listrik, teori dasar komitmen unit, Dan metode *parallel simulated annealing dengan komputasi spekulatif*.

### **Bab III: DATA SISTEM UNIT PEMBANGKIT TERMAL PADA PT. PEMBANGKIT JAWA BALI.**

Bab ini berisi data daya terpasang unit pembangkit, data karakteristik unit pembangkit, data unit pembangkit siap operasi dan tidak siap operasi, data pembebanan harian, data kendala unit pembangkit dan data biaya. Juga dicantumkan data pembangkit yang di pakai untuk uji validasi program komputer.

### **Bab IV : ANALISIS DATA PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL MENGGUNAKAN METODE *PARALLEL SIMULATED ANNEALING DENGAN KOMPUTASI SPEKULATIF* PADA PT. PEMBANGKIT JAWA BALI.**

Bab ini berisi tampilan program, algoritma program, analisa program dan evaluasi hasil perhitungan dan perbandingan biaya operasional metode *parallel simulated annealing* dengan *komputasi spekulatif* Pada PT. PJB.

**Bab V : KESIMPULAN DAN SARAN.**

Bab ini berisi kesimpulan dari kajian data dan berisi saran.

**DAFTAR PUSTAKA**

## **BAB II**

### **TINJAUAN PUSTAKA**

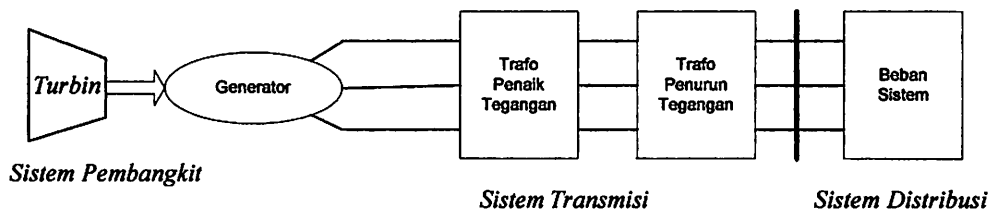
#### **2.1. Sistem Tenaga Listrik<sup>[1]</sup>**

Untuk keperluan sistem tenaga listrik bagi para pelanggan, diperlukan beberapa peralatan listrik. Berbagai peralatan listrik ini dihubungkan satu sama lain sehingga mempunyai hubungan inter relasi dan secara keseluruhan membentuk suatu sistem tenaga listrik. Yang di maksud dengan sistem tenaga listrik di sisni adalah sekumpulan pusat-pusat listrik dan gardu induk (pusat beban) yang satu sama lain dihubungkan oleh jaringan transmisi sehingga merupakan sebuah kesatuan yang terinterkoneksi.

Karena berbagai persoalan teknis, tenaga listrik hanya dapat dibangkitkan pada lokasi tertentu. Mengingat pemakaian tenaga listrik atau pelanggan tenaga listrik tersebar di berbagai tempat, maka penyaluran tenaga listrik dari tempat dibangkitkan sampai ketempat pelanggan memerlukan berbagai penanganan teknis.

Tenaga listrik dibangkitkan dari pusat-pusat listrik, seperti : PLTA, PLTU, PLTD, PLTG, dan PLTGU, kemudian disalurkan melalui saluran transmisi setelah tegangan dinaikan terlebih dahulu oleh transformator penaik tegangan yang terdapat di pusat-pusat tenaga listrik. Setelah tenaga listrik disalurkan melalui saluran transmisi, maka sampailah tenaga listrik tersebut pada gardu induk ( GI ) yang untuk kemudian tegangannya diturunkan oleh trafo penurun tegangan menjadi tegangan menengah atau tegangan rendah.

Jaringan setelah keluar dari gardu induk umumnya di sebut jaringan distribusi dan jaringan antara pusat listrik dengan gardu induk di sebut jaringan transmisi. Setelah disalurkan melalui jaringan distribusi primer maka tenaga listrik kemudian diturunkan tegangannya oleh gardu induk distribusi menjadi tegangan 380/220 volt atau 220/127 volt dan baru kemudian disalurkan ke pelanggan listrik



**Gambar 2.1. Elemen Pokok Sistem Tenaga Listrik**

Dari uraian di atas kiranya dapat di mengerti bahwa besar kecilnya tegangan listrik ditentukan sepenuhnya oleh konsumen, yaitu tergantung bagaimana para konsumen akan menggunakan peralatan listriknya, kemudian PT. PLN (PERSERO) harus bisa mengimbangi kebutuhan tenaga listrik tersebut, dalam arti selalu menyesuaikan daya listrik yang dibangkitkan dengan mermintaan tenaga listrik oleh pelanggan listrik.

Biaya operasi dari sistem tenaga listrik pada umumnya merupakan bagian biaya yang terbesar dari biaya operasi suatu sistem tenaga listrik. Secara garis besar biaya operasi dari suatu sistem tenaga listrik terdiri atas :

- a. Biaya pembelian tenaga listrik.
- b. Biaya pegawai
- c. Biaya bahan bakar dan materi operasi.
- d. Biaya lain-lain.



Dari ke empat biaya tersebut di atas, biaya bahan bakar pada umumnya adalah biaya yang terbesar. Untuk PLN biaya bahan bakar adalah kira-kira 60% dari biaya operasi secara keseluruhan.

Karena daya listrik yang dibangkitkan harus sama dengan tenaga listrik yang dibutuhkan oleh konsumen, maka manajemen operasi sistem tenaga listrik harus memperhatikan hal-hal sebagai berikut :

- a. Prakiraan beban.
- b. Syarat-syarat pemeliharaan beban.
- c. Kendala yang diinginkan.
- d. Pengaturan dan penyaluran beban.
- e. Proses produksi tenaga listrik yang ekonomis.

Kelima hal di atas masih harus sering di kaji ulang terhadap berbagai kendala seperti :

- a. Aliran beban dalam jaringan.
- b. Daya hubung singkat dan gangguan yang sering menimpa peralatan.
- c. Stabilitas sistem.
- d. Penyediaan suku cadang dan dana.

Dengan memperhatikan kendala-kendala di atas maka seringkali harus dilakukan pengaturan kembali terhadap rencana pemeliharaan dan alokasi beban. Makin besar sistem, makin banyak hal yang harus diamati dan dikoordinasi, sehingga diperlukan perencanaan, pelaksanaan, pengendalian dan evaluasi sistem yang cermat.

Dalam mengoperasikan sistem tenaga listrik ditemui berbagai persoalan. Hal ini antara lain disebabkan karena pemakaian tenaga listrik yang selalu berubah dari waktu ke waktu, biaya bahan bakar yang relatif tinggi serta kondisi alam dan lingkungan yang sering mengganggu jalannya operasi. Berbagai persoalan pokok yang dihadapi dalam mengoperasikan sistem tenaga listrik adalah :

a. Pengaturan frekwensi.

Sistem tenaga listrik harus dapat memenuhi kebutuhan akan tenaga listrik dari para konsumen dari waktu ke waktu. Untuk ini daya yang dibangkitkan dalam sistem tenaga listrik harus selalu sama dengan beban sistem, hal ini diamati melalui frekwensi sistem. Kalau daya yang dibangkitkan dalam sistem lebih kecil dari pada beban system maka frekwensi akan turun dan begitu pula sebaliknya, apabila daya yang dibangkitkan lebih besar dari pada beban maka frekwensi turun.

b. Pemeliharaan peralatan.

Peralatan yang beroperasi dalam sistem tenaga listrik perlu dipelihara secara periodik dan juga perlu segera diperbaiki apabila ada kerusakan.

c. Biaya operasi.

Biaya operasi khususnya biaya bahan bakar adalah biaya yang terbesar dari suatu perusahaan listrik sehingga perlu dipakai teknik-teknik optimasi untuk menekan biaya tersebut.

d. Perkembangan sistem.

Beban selalu berubah sepanjang waktu dan juga selalu berkembang seiring dengan perkembangan kegiatan masyarakat yang tidak dapat dirumuskan

secara eksak, sehingga perlu diamati serta terus menerus agar pengembangan system yang harus dilakukan selalu dapat mengikuti perkembangan beban, sehingga tidak akan terjadi pemadaman tenaga listrik.

e. Gangguan dalam sistem.

Gangguan dalam sistem tenaga listrik adalah sesuatu yang tidak dapat sepenuhnya dihindarkan. Penyebab gangguan yang paling besar adalah petir, hal ini sesuai dengan isokeraunik level yang tinggi di negara kita.

f. Tegangan dalam sistem.

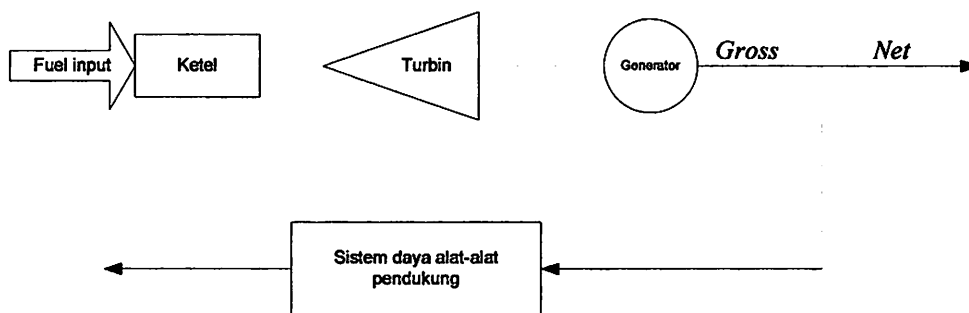
Tegangan merupakan salah satu unsur kualitas penyediaan tenaga listrik dalam sistem, oleh karena itu perlu diperhatikan dalam pengoperasian sistem.

## 2.2. Karakteristik Pembangkit Listrik <sup>[2]</sup>

Hal yang paling mendasar dalam optimasi ekonomi dari sebuah pembangkit listrik tenaga termal adalah dengan ditentukannya karakteristik masukan keluaran (*input-output characteristic*) pusat listrik tersebut. Dalam mendefinisikan karakteristik masukan keluaran, akan dibicarakan tentang *gross input* dan *net output* yang dihasilkan pusat listrik tersebut. *Gross input* pembangkit termal menyatakan jumlah keseluruhan bahan bakar yang diperlukan, sedangkan *net output* adalah daya nyata (*real power*) yang dihasilkan generator.

Tipe sebuah pembangkit listrik tenaga termal tampak pada gambar 2.2. Bagan tersebut terdiri atas sebuah ketel yang menghasilkan uap untuk menggerakkan turbin uap yang dikopel dengan sebuah generator listrik. Daya listrik yang dihasilkan tidak seluruhnya disalurkan ke sistem tetapi sebagian kecil digunakan untuk

mengoperasikan peralatan yang terdapat pada pusat listrik tersebut, seperti ketel, pompa, kompresor dan sebagainya, serta untuk mencatu peralatan kontrol, komunikasi, penerangan dan komputer.



**Gambar 2.2. Unit Boiler Dan Generator**

### 2.2.1. Karakteristik Masukan-Keluaran [2]

Masukan sebuah pembangkit termal umumnya dinyatakan sebagai banyaknya energi persatuan waktu dari bahan bakar yang diberikan ke ketel untuk meneghasilkan daya listrik yang merupakan keluaran dari pusat listrik tersebut.

Terdapat dua notasi, yaitu :

H dengan satuan [Mbtu/hour]

F dengan satuan [\$US/hour]

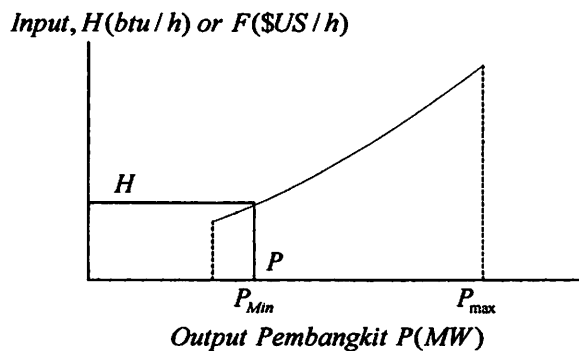
Di mana  $F=H \times \$US/Btu$ , dan  $\$US/Btu$  menyatakan harga bahan bakar persatuan energi yang dikandung oleh bahan bakar tersebut.

Sedangkan keluaran dari pembangkit listrik termal adalah daya nyata yang dihasilkan oleh generator dikurangi daya nyata yang di pakai oleh pusat listrik tersebut. Notasi yang umum digunakan adalah :

P dengan satuan [MW].

Jadi dapat disimpulkan bahwa masukan pusat listrik merupakan fungsi terhadap keluarannya, maka hubungan tersebut dapat di tulis sebagai berikut :  $H = f(P)$  [Mbtu/h] atau  $F = f(P)$  [\$US/h].

Pembahasan selanjutnya akan berpedoman atas dasar fungsi biaya bahan baker ( $F = f(P)$ ) [\$US/h], sedangkan kurva dari karakteristik masukan-keluaran dari sebuah pembangkit termal yang telah diidealkan ditunjukkan pada gambar 2.3. Masukan adalah sebuah ordinat yang berupa banyaknya energi yang diperlukan persatuan waktu [Mbtu] atau juga merupakan biaya bahan bakar yang dikosumsi persatuan waktu [\$US/h], sedangkan keluaran adalah daya listrik [MW] yang dihasilkan blok tersebut untuk melayani beban sistem.



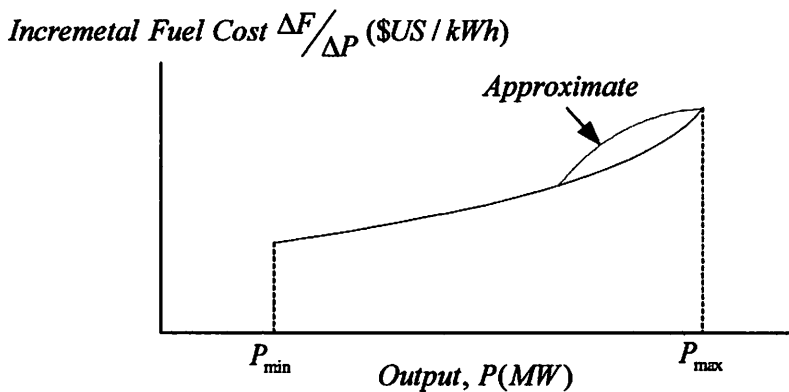
**Gambar 2.3. Kurva karakteristik inout-output pembangkit termal**

Data yang dibutuhkan untuk menggambarkan diagram fungsi karakteristik masukan-keluaran dapat diperoleh dari perhitungan pada saat perencanaan atau tes yang telah dilakukan terhadap unit pembangkit yang bersangkutan.

### 2.2.2. Karakteristik Laju Tambahan Biaya Bahan Bakar

Karakteristik laju tambahan biaya bahan bakar atau *incremental fuel cost characteristic* adalah turunan pertama dari fungsi biaya bahan bakar terhadap tingkat pembebanan  $P$  [MW] dari pusat listrik yang bersangkutan. Fungsi ini menunjukkan besarnya kenaikan atau penurunan biaya bahan bakar untuk setiap satu satuan perubahan beban.

Secara luas, fungsi biaya bahan bakar akan digunakan untuk menentukan pembebanan ekonomi dari sebuah pembangkit listrik tenaga termal. Tampak pada gambar 2.4. adalah kurva laju tambahan biaya bahan bakar yang telah diidealkan dari sebuah pembangkit termal.



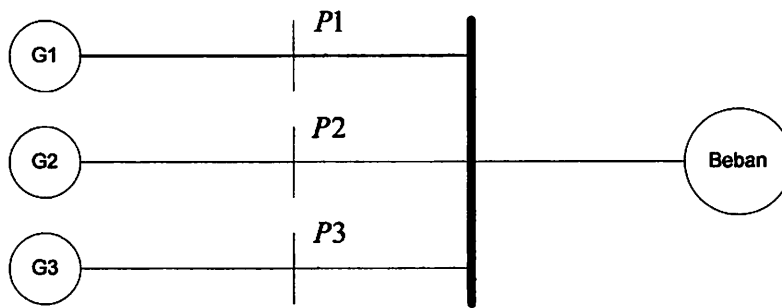
Gamabar 2.4. Kurva karakteristik laju tambahan biaya bahan bakar

### 2.3. Komitmen Unit <sup>[1][2]</sup>

Kebutuhan akan energi listrik sudah mengalami pergeseran yang cukup signifikan dari masa lalu hingga sekarang, di mana pada masa lalu listrik hanya digunakan untuk sarana penerangan saja, sedangkan pada masa sekarang ini hamper sepenuhnya menjadi denyut nadi manusia seiring dengan perkembangan

teknologi yang ada, di mana saat ini hampir semua peralatan yang di pakai manusia menggunakan listrik sebagai sumber energinya. Apalagi dengan perkembangan sektor industri di mana energi listrik menjadi bagian yang sangat vital untuk kelangsungan proses produksinya. Guna memenuhi kebutuhan akan energi listrik ini, maka diperlukan suatu pengaturan operasi system tenaga listrik yang sangat ekonomis, di mana bukan hanya faktor pemenuhan kebutuhan beban yang harus dipenuhi tetapi juga faktor kualitas, kendala dan nilai ekonomis menjadi syarat mutlak yang harus dipenuhi.

Untuk mengatasi masalah di atas, maka system tenaga listrik yang ada sekarang ini menggunakan sistem interkoneksi, di mana prinsip dari system interkoneksi ini adalah menggabungkan beberapa pusat pembangkit yang tersebar di berbagai lokasi, baik unit hidro maupun unit termis secara parallel melalui suatu jaringan transmisi bertegangan tinggi untuk menyuplai beban gabungan. Hal ini berarti bahwa seluruh pembangkit yang berada dalam satu wilayah menjadi satu kesatuan yang terpadu. Untuk system interkoneksi yang besar, yang terdiri dari banyak unit pembangkit dan banyak pusat beban (gardu induk), sarana pengendalian operasi dengan menggunakan sarana komunikasi saja tidak cukup, tetapi harus di tambah dengan peralatan telemetering dan alat alat pengolah data elektronis, seperti computer. Di samping itu, pada pengoperasian system yang terinterkoneksi PLN berkewajiban menyediakan energi listrik dengan rating tegangan yang berada dalam batas-batas tertentu.



**Gambar 2.5. Sistem Interkoneksi [2]**

Tetapi pada kenyataannya, pengoperasian secara interkoneksi menimbulkan masalah teknis yang cukup rumit dan kompleks. Hal ini dikarenakan terbentuknya dua kepentingan PLN dalam menjalankan misinya. Misi utama PLN adalah sebagai perusahaan jasa teknik kelistrikan adalah mencari laba, dan untuk itu suatu langkah ekonomis harus di ambil untuk memperoleh keuntungan yang maksimal atas modal yang ditanamkan, yaitu dengan mengoptimalkan pengoperasian berarti harus di capai biaya operasi yang seminimal mungkin, khususnya biaya bahan baker, mengingat bahwa biaya bahan bakar merupakan unsur terbesar dalam total biaya operasi. Kemudian misi PLN yang lainnya adalah mengelola dan menyediakan energi listrik bagi masyarakat dengan kualitas dan keandalan yang terbaik. Melihat kedua misi ini, maka diperlukan suatu perencanaan penyaluran dan penyediaan energi listrik yang memenuhi factor kualitas dan keandalan dengan biaya yang seekonomis mungkin.

Komitmen unit merupakan suatu metode solusi yang tepat untuk mengatasi permasalahan di atas guna mencapai penjadwal unit pembangkit yang harus beroperasi untuk periode waktu tertentu agar di capai biaya operasi yang



seekonomis mungkin. Pada masalah komitmen unit diasumsikan bahwa ada sejumlah N unit pembangkit yang tersedia dan harus dioperasikan untuk memenuhi permintaan beban.

### 2.3.1. Kendala Pada Komitmen Unit [2]

Dalam pengoperasian unit pembangkit untuk memenuhi kebutuhan beban terdapat berbagai kendala yang merupakan syarat kendala. Kendala tersebut antara lain:

a. Kendala permintaan beban.

Daya yang dihasilkan dari semua unit komitmen harus memenuhi persamaan keseimbangan sebagai berikut :

$$\sum_{i=1}^N U_{it} P_{it} = P_D \quad ; \quad 1 \leq t \leq T \quad \dots\dots\dots(1)$$

b. Cadangan Berputar (*Spining reserve*)

*Spining reserve* adalah kata yang digunakan untuk menggambarkan jumlah total daya yang mungkin dibangkitkan dari semua unit yang tersinkronisasi (dihubungkan paralel) pada system dikurangi dengan beban saat itu dan rugi-rugi penyaluran yang terjadi. *Spining reserve* harus ada untuk menghindari penurunan frekwensi system yang terlalu besar bila satu atau lebih unit harus keluar dari system atau dengan kata lain harus ada cukup cadangan pada unit-unit lain untuk menutupi kekurangan suplai daya yang hilang dalam periode waktu tertentu. *Spining reserve* harus dialokasikan untuk mentaati aturan-aturan tertentu, biasanya aturan yang di pakai adalah bahwa cadangan tersebut berupa sebuah presentase yang diberikan terhadap beban puncak yang

diperkirakan, atau bahwa cadangan berikut harus mampu menutupi kehilangan daya dari unit yang paling besar yang dibebani penuh dalam suatu periode tertentu.

Selain itu cadangan tersebut tidak hanya harus mencukupi untuk menutupi kegagalan pembangkit unit, tetapi cadangan-cadangan tersebut harus dialokasikan diantara unit-unit yang bereaksi lambat. Ini memungkinkan system kontrol pembangkit otomatis untuk mengembalikan frekwensi dan perputaran yang cepat pada saat unit pembangkit keluar dari system.

Besar cadangan berputar tersebut harus ditentukan secara hati-hati, sebab seringkali penentuan yang didasarkan untuk menjaga keandalan system berbenturan dengan biaya pengoperasian yang diusahakan seekonomis mungkin, misalnya cadangan berputar kecil dan unit yang terbesar mengalami gangguan dan trip sehingga unit tersebut keluar secara mendadak dari sistem, maka cadangan berputar tersebut tidak cukup untuk mengatasi kekurangan pembangkit yang terjadi dan untuk menghindari system *collaps*, maka perlu dilakukan pelepasan beban dan ini mengakibatkan keandalan sistem menurun. Jadi makin besar cadangan berputar dalam sistem, maka makin handal sistem tersebut dalam menghindari gangguan, tetapi makin besar biaya operasi terutama biaya bahan bakarnya karena besarnya cadangan berputarnya. Oleh karena itu perlu adanya suatu kesepakatan antara pemenuhan keandalan dan pengoptimalan biaya operasi. Dalam menentukan cadangan putaran di hitung menurut jumlah yang ditentukan atau sebagai presentasi dari permintaan beban yang diramalkan, misalnya :

$$\sum_{i=1}^N U_{ii} P \max_i \geq (P_D + R_t) ; 1 \leq t \leq T \dots\dots\dots(2)$$

### 2.3.2. Kendala Unit Termis [2]

Unit termis biasanya memerlukan “crew” untuk mengoperasikannya, terutama ketika dinyalakan dan dimatikan. Sebuah unit termis hanya dapat dijalankan dibawah perubahan temperatur yang gradual, dan ini diterjemahkan kedalam sebuah periode waktu dalam jam yang dibutuhkan untuk membawa unit tersebut on-line. Hal ini menyebabkan kendala-kendala antara lain :

#### a. Minimum up-time

Sekali unit dihidupkan, tidak boleh unit tersebut langsung dimatikan, harus ada tenggang waktu tertentu untuk dioperasikan sebelum boleh mematikannya.

$$Ton_i \geq MUT_i ; 1 \leq i \leq N \dots\dots\dots(3)$$

#### b. Minimum down-time

Sesaat unit dimatikan, ada tenggang waktu minimum untuk tidak beroperasi sebelum unit tersebut dapat dihidupkan kembali.

$$Toff_i \geq MDT_i ; 1 \leq i \leq N \dots\dots\dots(4)$$

#### c. Limit Pembangkitan

$$U_{it} P_{min,i} \leq P_{it} \leq P_{max,i} U_{it} ; 1 \leq t \leq T , 1 \leq i \leq N \dots\dots\dots(5)$$

Selain tiga kendala di atas, terdapat kendala lain yang mempengaruhi unit termis, yaitu : status unit awal (*initial status*), ketersediaan unit (misalnya run, tidak tersedia, tersedia atau output tetap).

### 2.3.3. Biaya star-up [2]

Biaya start-up adalah biaya yang diperlukan oleh pembangkit untuk start dari keadaan tidak beroperasi sampai pembangkit beroperasi (terhubung pada sistem tenaga listrik). Ada dua macam biaya *start-up* yaitu :

a. Biaya *start-up* pada kondisi dingin (*Cold Start*)

Kondisi ini terjadi pada saat pembangkit lepas dari sistem (tidak beroperasi), sedangkan temperatur *boiler* dibiarkan turun dari temperatur kerjanya. Sehingga pada saat beroperasi kembali perlu dilakukan pemanasan kembali.

b. Biaya *start-up* pada kondisi panas (*Hot Start*).

Kondisi ini terjadi karena saat pembangkit tersebut baru saja dimatikan kemudian dihidupkan kembali, maka penyalannya dari kondisi panas (*hot star*) karena temperature dari boiler masih tinggi, sehingga unit pembangkit tersebut dikenakan biaya panas (*hot star cost*).

Biaya start-up tergantung pada waktu down dari unit tersebut, yang di mana dapat berbeda-beda dari nilai maksimum, ketika unit tersebut di mulai dari keadaan cold, kepada nilai yang paling kecil, jika unit tersebut hanya baru dimatikan. Perhitungan biaya start-up juga tergantung pada metode perlakuan unit panas selama periode waktu down. Dalam hal ini biaya start-up untuk unit *i* pada waktu *t*, menggunakan bentuk umum sebagai berikut :

$$S_{it} = \tau_i + [1 - \sigma_i \exp(-T_{off_i} / MDT_i)] + \delta_i \dots\dots\dots(6)$$

Dimana:

$S_{it}$  : Biaya start-up untuk unit *I* pada jam *t*

- $\tau_i$  : Konstanta waktu pendinginan unit i
- $\sigma_i$  : Hot start-up cost unit pembangkit i
- $\delta_i$  : Cold start-up cost unit pembangkit i

**c. Fungsi Obyektif**

Biaya bahan bakar merupakan unsur biaya yang paling penting dalam operasi system pembangkit termal. Fungsi biaya bahan bakar  $F_i (P_i)$  untuk tiap unit pembangkit terhadap daya keluaran diekspresikan dalam bentuk fungsi kuadrat, yang dapat dinyatakan sebagai berikut :

$$F_i (P_i) = A_i P_{ii}^2 + B_i P_{ii} + C_i \dots\dots\dots(7)$$

Dengan demikian, fungsi sasaran kesulitan dari UPC adalah total biaya operasi minimum komitmen unit pembangkit dalam waktu t jam, dapat diformulasikan sebagai berikut :

$$= \min (\text{total biaya bahan bakar} + \text{total biaya star up})$$

$$= \min \sum_{i=1}^T \sum_{i=1}^N (U_{ii} F_i (P_i) + V_{ii} S_{ii})$$

**2.4. Formulasi Masalah Komitmen Unit**

Permasalahan dalam komitmen unit adalah bagaimana cara menjadwalkan unit pembangkit yang beroperasi untuk periode waktu tertentu guna melayani kebutuhan beban secara optimal, sehingga didapatkan biaya operasi minimum tanpa melanggar kendala-kendala yang dimasukkan.

Adapun parameter-parameter yang digunakan pada formulasi masalah komitmen unit adalah sebagai berikut :

$i$	:	Indeks untuk unit generator
$t$	:	Indeks untuk interval waktu
$N_u$	:	Jumlah unit generator
$N_t$	:	Jumlah interval waktu
$P$	:	Output power pada satu unit generator
$D$	:	Total kebutuhan power
$U$	:	Matrik kondisi komitmen unit
$C$	:	Biaya operasi
$F$	:	Biaya bahan bakar satu unit generator
$E$	:	Total biaya operasi
$C_{ST}$	:	Biaya start-up satu unit generator
$T$	:	Suhu dalam algoritma simulated annealing
$M$	:	Jumlah iterasi (pengulangan) di tiap suhu
$N$	:	Total jumlah langkah suhu yang disimulasi

## **2.5. Teori Dasar Simulated Annealing**

### **2.5.1. Simulated Annealing (SA)**

Optimisasi SA adalah suatu simulasi dari proses fisik pendinginan logam dengan pengotoran, di mana suatu partikel cair didinginkan pelan-pelan dari temperatur yang sangat tinggi sampai mengeras pada temperatur rendah. Jumlah iterasi di dalam metoda SA adalah penganalisa untuk tingkat temperatur.

Sejumlah percobaan dilakukan dalam suatu iterasi. Pada setiap percobaan, suatu solusi calon dihasilkan. Manakala solusi calon yang baru mempunyai biaya yang lebih rendah dibanding solusi yang sekarang akan diterima sebagai solusi yang ditemukan kemudian. Cara lainnya, jika solusi calon yang baru adalah lebih jelek, Solusi itu diterima dengan suatu kemungkinan penerimaan  $\Pr(\Delta)$  :

$$\Pr(\Delta) = [1/(1 + \exp(\Delta/T))] \dots\dots\dots(8)$$

di mana  $\Delta$  adalah jumlah solusi jelek solusi baru dan solusi lama, dan T adalah temperatur di mana solusi yang baru dihasilkan. Temperatur T dikurangi pada awal suatu iterasi menurut

$$T_k = r^{(k-1)}T_0 \dots\dots\dots(9)$$

Di mana  $T_0$  dan  $T_k$  adalah temperatur awal dan temperatur di iterasi berturut-turut, dan r adalah faktor pengurangan temperatur. Solusi yang diterima digunakan untuk menghasilkan suatu solusi calon baru di dalam percobaan yang berikutnya. Solusi yang baru dibentuk dengan mengubah solusi lama. Proses solusi berlanjut sampai jumlah maksimum percobaan dicapai dan banyaknya iterasi kemudian menjadi 1. Proses optimisasi diakhiri manakala jumlah maksimum generasi telah didapatkan atau solusi dengan biaya yang diinginkan telah ditemukan.

**2.5.2. Konsep Parallel Simulated Annealing**

Pada algoritma SA yang *konvensional*, membutuhkan banyak sekali gangguan (misal  $M$  ) yang dievaluasi secara berurutan pada setiap nilai temperatur.

Setiap evaluasi membutuhkan tiga *basic step*:

- (a) Bangkitkan gangguan secara *random*.
- (b) Evaluasi biaya yang baru.
- (c) Terima atau tolak gangguan.

Pada algoritma paralel SA, sebuah blok dari gangguan (misal  $m$  gangguan) dievaluasi secara simultan dan kebutuhan jumlah gangguan ( $M$ ) dicapai lebih cepat dari pada algoritma *konvensional* SA. Faktor percepatan diberikan:

$$\text{Faktor percepatan} = \frac{\text{round-up}(M / m)}{M} \dots\dots\dots(10)$$

Dimana fungsi dari *round-up* ( $x$ ) adalah didefinisikan sebagai nilai integer yang paling kecil dan lebih besar dari pada  $x$ .

### 2.5.3. Algoritma Simulated Annealing

*Step (1).* Mulai U, menentukan suhu awal T dan jumlah pengulangan M dan mencari total biaya operasi E untuk U yang ditentukan.

*Step (2).* Mengulangi langkah 2.1. dan 2.2. sampai suhunya cukup rendah (nilai yang baik bias ditentukan melalui eksperimen).

*Step 2.1.* Ulangi langkah 2.1.1. sampai langkah 2.1.5. selama M kali.

*Step 2.1.1.* Mengganggu U secara acak.

*Step 2.1.2.* Mencari nilai total biaya operasi yang baru  $E_{new}$

*Step 2.1.3.* Komputasi  $\Delta E : \Delta E = E_{new} - E$

*Step 2.1.4.* Menerima atau menolak U baru sesuai berdasarkan pada fungsi Boltzman.



*Step 2.1.5.* Jika U baru bisa diterima maka U dan E di-update

*Step 2.2.* Menurunkan suhu.

*Step 2.1.1:* Bangkitkan secara acak U gangguan

Sebuah acak U gangguan diberikan ke U dimana batas *up time*, *down time* tidak dilanggar. Hal ini dikerjakan dengan memilih unit pembangkit secara acak, memilih unit pembangkit yang akan di *off*-kan secara acak dan memilih durasi *shut down* secara acak, tetapi tidak melanggar batasan *minimum up time* dan *down time*. Prosedur diatas dijabarkan dibawah ini:

- (a) Pilihlah sebuah unit pembangkit i.
- (b) Set  $uc_{it}$  untuk seluruh jam t.
- (c) Secara acak pilih *interfal* waktu  $t_{shut}$ .
- (d) Secara acak pilih sebuah periode shut down  $t_{shut}$  sehingga batasan *minimum up time* dan *minimum down time* tidak dilanggar.
- (e) If  $t_{shut}$  tidak sama dengan nol set  $uc_{it_{shut}}$  to  $uc_{it_{shut}} + t_{shut}$  sama dengan nol.

*Step 2.1.2:* Temukan nilai baru dari total biaya operasi  $F_{new}$ .

Dalam  $uc$ , *sub problem Economic Dispatch* bisa didefinisikan:

$$\text{Minimum: } C_T = \sum_{i=1}^{nu} F_i U_{it} \text{ for } t=1 \text{ to } n_t \dots\dots\dots(1)$$

Dengan memperhatikan :

- (1) batasan keseimbangan daya

$$\sum_{i=1}^{nu} uc_{it} P_{it} = D_t \text{ for } t = 1 \text{ to } n_t \dots\dots\dots(2)$$

(2) batasan *spinning reserve*

$$\sum_{i=1}^{nu} uc_{it}P_i^{max} \geq R_t + D_t \text{ for } t=1 \text{ to } n_t \dots\dots\dots(3)$$

(3) batasan kapasitas unit pembangkit

$$P_i^{min} \leq P_{it} \leq P_i^{max} \text{ for } t=1 \text{ to } n_t \dots\dots\dots(4)$$

(4) kemungkinan pembangkitan sesuai yang dibutuhkan beban

$$\sum_{i=1}^{nu} P_i^{max} ut_{it} \leq D_t \dots\dots\dots(5)$$

(5) Batasan *minimum up- time* dan *minimum down- time*

Minimum *up- time* adalah minimum jumlah jam unit harus *on* sebelum bisa di *off*-kan. Sebagaimana diatas minimum *down time* adalah *minimum* jumlah jam unit harus *off* sebelum bisa di *on*-kan kembali.

Jika batasan-batasan diatas dilanggar maka  $C_t$  dikenakan pinalti  $\mu$ , dan nilai  $C_t$  dihitung dengan memperhatikan batasan (1) dan (3) dengan menyelesaikan masalah ekonomi dispatch. Nilai dari pinalti  $\mu$  dipilih bernilai 10% lebih besar dari pada biaya yang disuplai oleh beban puncak dengan kombinasi unit pembangkit yang paling mahal.

Total biaya  $F_{new}$  dihitung dengan:

$$F_{new} = \sum_{t=1}^{nt} C_t + \sum_{i=1}^{nu} C_{Sti} \dots\dots\dots(6)$$

Biaya *start Up* dihitung berdasarkan data yang diberikan oleh unit pembangkit masing-masing.

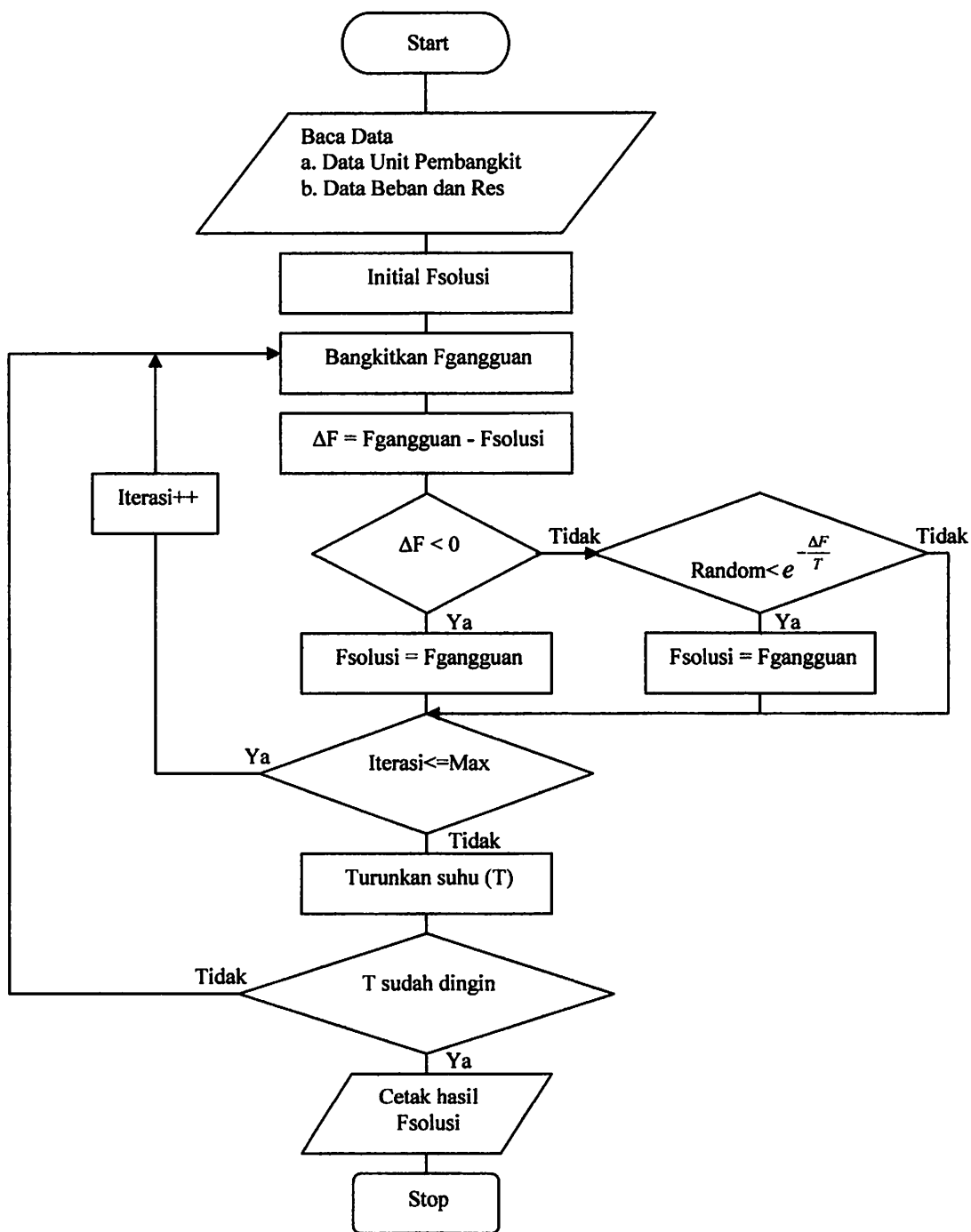
*Step 2.1.4: Terima atau tolak uc baru tergantung kepada fungsi Boltzman*

Pada algoritma *Simulated Annealing* kriteria dari penerimaan atau penolakan didasarkan pada fungsi *Boltzman*. Hal ini bisa dilihat dari:

JIKA  $\Delta F$  adalah *negatif* MAKA terima *configurasi* baru

JIKA TIDAK hitung probabilitas dari penerimaan =  $e^{-\Delta F/T}$ , dan bangkitkan nilai *acak* antara 0 dan 1.

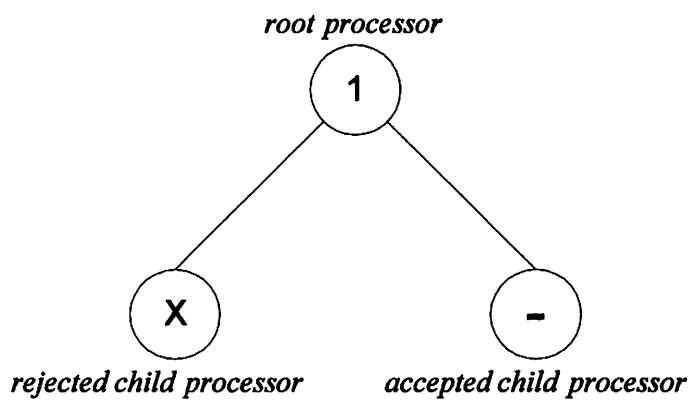
Jika probabilitas penerimaan adalah lebih besar dari nilai *acak* maka terima konfigurasi baru, jika tidak ditolak.



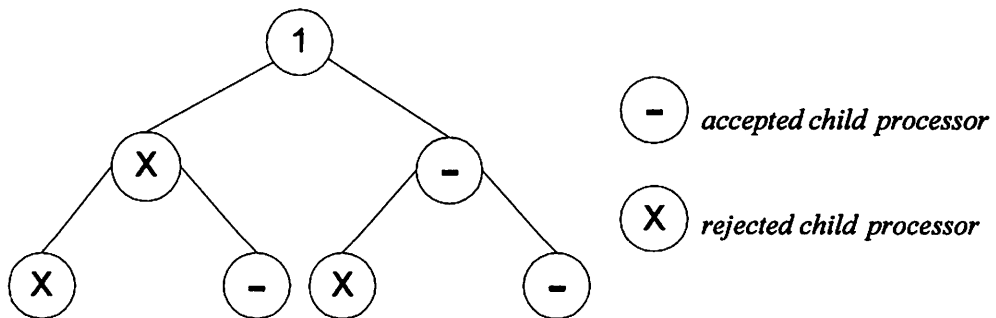
**Gambar 2.6.**  
**FLOWCHARTSTANDART SIMULATED ANNEALING**

#### 2.5.4. Pendekatan Komputasi Spekulatif (KS)

Konsep dari KS adalah melakukan pekerjaan sebelum tahu apakah hal itu diperlukan atau tidak. Satu processor diketahui sebagai *root processor* bisa melakukan pekerjaan yang diberikan oleh iterasi dan dua processor yang lain yang diketahui sebagai *child processor*, melakukan komputasi spekulatif dengan cara salah satu dari children processor berspekulasi bahwa gangguan yang dievaluasi oleh root processor akan ditolak dan kemudian diproses untuk iterasi selanjutnya. Sementara itu child processor yang lain berspekulasi untuk menerima gangguan dan dipakai untuk iterasi selanjutnya.



Gambar 2.7. Komputasi Spekulatif dengan dua iterasi



Gambar 2.8. Komputasi Spekulatif dengan tiga iterasi

## 2.6. Implementasi dari Paralel SA

Untuk mengimplementasikan metode ini, algoritma dimulai dengan random matrik  $U$ . Sebuah gangguan diaplikasikan untuk matrik  $U$  ini dan root processor mengevaluasi apakah menerima atau tidak. Jika diterima child processor akan timbul pada tree, hal ini diasumsikan bahwa root processor menerima gangguan dan child processor akan ditugasi untuk membangkitkan  $U$  matrik gangguan dari matrik  $U$  yang diterima oleh root processor. Jika child processor yang ditolak hadir pada tree, matrik  $U$  yang ditolak akan dievaluasi oleh root processor dan dikembalikan ke matrik  $U$  sebelumnya. Kemudian gangguan baru diberikan diberikan dan hasil matrik  $U$  akan dievaluasi. Prosedur ini akan dievaluasikan untuk seluruh processor yang lain pada tree dengan menggunakan setiap child processor sebagai root processor untuk menghitung pada setiap level.

Pada setiap akhir evaluasi, tree ditracking sampai kondisi berhenti dari tree dicapai. Jika jumlah node yang dikerjakan adalah  $m$  maka evaluasi paralel adalah sama dengan sejumlah  $m$  evaluasi dari konvensional algoritma SA (CSA). Pada state akhir setelah paralel evaluasi ditentukan dengan state dari node terakhir yang dihasilkan. Paralel evaluasi diulang sampai  $\Sigma m$  adalah sama atau lebih besar daripada  $M$  yang dibutuhkan.

## **BAB III**

### **PENERAPAN METODE *SIMULATED ANNEALING* DENGAN *KOMPUTASI SPEKULATIF* PADA PT. PEMBANGKIT JAWA BALI**

#### **3.1. Pendahuluan**

Penjadwalan dalam sistem tenaga listrik merupakan faktor yang paling penting, karena penjadwalan dalam sistem tenaga listrik saling terkait dengan biaya operasional pembangkit (biaya bahan bakar, biaya start up, biaya shut down dan biaya lainnya). Oleh karena itu, perencanaan penjadwalan unit operasi pembangkit harus komprehensif dan sistematis baik dalam perencanaan jangka panjang ataupun perencanaan operasional pembangkit, sehingga di capai suatu operasi pembangkit yang optimal.

Kebutuhan energi listrik fluktuatif perubahannya tiap jam selama 24 jam sehari, sehingga PT. PJB memerlukan suatu perkiraan pembebanan yang tepat dan akurat. Dengan demikian dapat ditentukan unit pembangkit mana yang beroperasi untuk mencukupi kebutuhan tersebut agar dicapai biaya operasional pembangkit listrik minimum.

Sebagai penyedia kebutuhan energi listrik bagi masyarakat luas, PT. PJB pada prinsipnya merupakan industri dalam bidang jasa dan perniagaan. Dengan demikian maka PT. PJB diharapkan mampu memberikan pelayanan yang memuaskan kepada masyarakat, selain itu suatu keharusan bagi perusahaan untuk berupaya meningkatkan penyediaan tenaga listrik yang handal, ekonomis dan bermutu.

### 3.2. Data Unit Pembangkit Termal

Jumlah unit pembangkit termal di PT. Pembangkit Jawa Bali berdasarkan survey yang dilakukan di tempat tersebut berjumlah 38 unit pembangkitan, masing-masing terdiri dari 11 unit Pembangkit Listrik Tenaga Uap (PLTU), 5 unit Pembangkit Listrik Tenaga Gas (PLTG), dan 22 unit Pembangkit Listrik Tenaga Gas dan Uap (PLTGU) yang tersebar di seluruh Jawa dan Bali.

Tabel 3.1.

Data Unit Termal Pada PT. PJB

No	Nama Pembangkit	Bahan Bakar	Kapasitas (MW)		Lama Wktu (Jam)			
			Min	Max	MUT	MDT	Cold Start	Hot Start
1	PLTU Paiton 1	Coal	225	370	72	48	17	4
2	PLTU Paiton 2	Coal	225	370	72	48	17	4
3	PLTGU Gersik GT 1.1.	Gas	53	102	36	10	1	0
4	PLTGU Gersik GT 1.2.	Gas	53	102	36	10	1	0
5	PLTGU Gersik GT 1.3.	Gas	53	102	36	10	1	0
6	PLTGU Gersik ST 1.0.	Gas	250	480	36	10	3	1
7	PLTGU Gersik GT 2.1.	Gas	53	102	36	10	1	0
8	PLTGU Gersik GT 2.2.	Gas	53	102	36	10	1	0
9	PLTGU Gersik GT 2.3.	Gas	53	102	36	10	1	0
10	PLTGU Gersik ST 1.0.	Gas	250	480	36	10	3	1
11	PLTGU Gersik GT 3.1.	Gas	53	102	36	10	1	0
12	PLTGU Gersik GT 3.2.	Gas	53	102	36	10	1	0
13	PLTGU Gersik ST 3.3.	Gas	53	102	36	10	1	0
14	PLTGU Gersik ST 1.0.	Gas	250	480	36	10	3	1
15	PLTU Gersik 1	Gas	43	85	48	10	9	1
16	PLTU Gersik 2	Gas	43	85	48	10	9	1
17	PLTU Gersik 3	Gas	90	175	48	10	9	2
18	PLTU Gersik 4	Gas	90	175	48	10	9	2
19	PLTG Gersik 1	Gas	5	16	3	1	1	0
20	PLTG Gersik 2	Gas	5	16	3	1	1	0
21	PLTG Gersik 3	Gas	5	16	3	1	1	0
22	PLTG Gilitimur 1	HSD	5	16	3	1	1	0
23	PLTG Gilitimur 2	HSD	5	16	3	1	1	0
24	PLTGU M. Karang GT 1.1	Gas	50	95	36	10	1	0
25	PLTGU M. Karang GT 1.2.	Gas	50	95	36	10	1	0
26	PLTGU M. Karang GT 1.3.	Gas	50	95	36	10	1	0
27	PLTGU M. Karang ST 1.0.	HSD	300	465	36	10	3	1
28	PLTGU M. Tawar GT 1.1.	HSD	72	138	36	10	0	0
29	PLTGU M. Tawar GT 1.2.	HSD	72	138	36	10	0	0



30	PLTGU M. Tawar GT 1.3.	HSD	72	138	36	10	0	0
31	PLTGU M. Tawar GT 2.1.	HSD	72	138	36	10	0	0
32	PLTGU M. Tawar GT 2.2.	HSD	72	138	36	10	0	0
33	PLTGU M. Tawar ST 1.0.	HSD	162	202	36	10	3	1
34	PLTU M. Karang 1	MFO	44	85	48	10	6	1
35	PLTU M. Karang 2	MFO	44	85	48	10	6	1
36	PLTU M. Karang 3	MFO	44	85	48	10	6	1
37	PLTU M. Karang 4	Gas	90	165	48	10	11	2
38	PLTU M. Karang 5	Gas	90	165	48	10	11	2

Sumber : Data Penawaran PT. PJB, Jl.Ketintang Baru No.11, Surabaya 6002

Keterangan : MUD = minimum up time.

Tabel 3.2.

Data Biaya Dan Parameter Unit Termal Pada PT. PJB

No	Nama Pembangkit	Biaya Start-Up (Juta Rp)		Koefisien Biaya Bahan Bakar		
		Cold Start-up	Hot Start-up	A	B	C
1	PLTU Paiton 1	682.98	149.68	3244978	111712.15	10.2971
2	PLTU Paiton 2	682.98	149.68	3244978	111712.15	10.2971
3	PLTGU Gersik GT 1.1.	7.82	0	5467532.4	217963.548	34.155
4	PLTGU Gersik GT 1.2.	7.82	0	5467532.4	217963.548	34.155
5	PLTGU Gersik GT 1.3.	7.82	0	5467532.4	217963.548	34.155
6	PLTGU Gersik ST 1.0.	57.68	31.46	10936203.3	72527.004	368.875
7	PLTGU Gersik GT 2.1.	7.82	0	5467532.4	217963.548	34.155
8	PLTGU Gersik GT 2.2.	7.82	0	5467532.4	217963.548	34.155
9	PLTGU Gersik GT 2.3.	7.82	0	5467532.4	217963.548	34.155
10	PLTGU Gersik ST 1.0.	57.68	31.46	10936203.3	72527.004	368.875
11	PLTGU Gersik GT 3.1.	7.82	0	5467532.4	217963.548	34.155
12	PLTGU Gersik GT 3.2.	7.82	0	5467532.4	217963.548	34.155
13	PLTGU Gersik ST 3.3.	7.82	0	5467532.4	217963.548	34.155
14	PLTGU Gersik ST 1.0.	57.68	31.46	10936203.3	72527.004	368.875
15	PLTU Gersik 1	143.74	40.59	1327126.68	217378.359	132.066
16	PLTU Gersik 2	143.74	40.59	1327126.68	217378.359	132.066
17	PLTU Gersik 3	229.5	95.52	5017369.5	169242.576	193.545
18	PLTU Gersik 4	229.5	95.52	5017369.5	169242.576	193.545
19	PLTG Gersik 1	6.13	0	352707.3	350680.77	903.969
20	PLTG Gersik 2	6.13	0	352707.3	350680.77	903.969
21	PLTG Gersik 3	6.13	0	352707.3	350680.77	903.969
22	PLTG Gilitimur 1	6.13	0	687181.85	683240.965	1762.3893
23	PLTG Gilitimur 2	6.13	0	687181.85	683240.965	1762.3893
24	PLTGU M. Karang GT 1.1	7.35	0	5730795	202052.97	108.045
25	PLTGU M. Karang GT 1.2.	7.35	0	5730795	202052.97	108.045

26	PLTGU M. Karang GT 1.3.	7.35	0	5730795	202052.97	108.045
27	PLTGU M. Karang ST 1.0.	54.22	29.67	11560815	53685.135	460.845
28	PLTGU M. Tawar GT 1.1.	0	0	14706521.3	433337.8	49.4605
29	PLTGU M. Tawar GT 1.2.	0	0	14706521.3	433337.8	49.4605
30	PLTGU M. Tawar GT 1.3.	0	0	14706521.3	433337.8	49.4605
31	PLTGU M. Tawar GT 2.1.	0	0	14706521.3	433337.8	49.4605
32	PLTGU M. Tawar GT 2.2.	0	0	14706521.3	433337.8	49.4605
33	PLTGU M. Tawar ST 1.0.	118.08	64.4	672630	144191.717	519.1757
34	PLTU M. Karang 1	122.58	31.08	2417820.7	473895.41	120.77935
35	PLTU M. Karang 2	122.58	31.08	2417820.7	473895.41	120.77935
36	PLTU M. Karang 3	122.58	31.08	2417820.7	473895.41	120.77935
37	PLTU M. Karang 4	215.34	89.29	2949187.5	205217.145	83.79
38	PLTU M. Karang 5	215.34	89.29	2949187.5	205217.145	83.79

Sumber Data Penawaran PT. PJB, Jl. Ketintang Baru No.11 Surabaya 60231

Catatan:	Harga Batubara	:	255	Rp/Kg
	Harga MFO	:	1595,5	Rp/Liter
	Harga HSD	:	1595,5	Rp/Liter
	Harga Gas UP. Gersik	:	2,53	US\$/MMBTU
	Harga Gas UP. M. Karang	:	2,45	US\$/MMBTU
	Nilai Tukar	:	9000	Rp/US\$

### 3.3. Aplikasi *Paralle Simulated Annealing* Pada PT. PJB

Perhitungan dan analisa ini dilakukan pada kebutuhan daya yang di tanggung PT. Pembangkit Jawa Bali tanggal 4, 6, dan 7 Desember 2003. Analisa dilakukan untuk ketiga hari tersebut, karena ketiga hari tersebut mewakili karakteristik kurva berlainan dengan keterangan sebagai berikut:

- Tanggal 4 Desember 2003 adalah hari kamis, mewakili beban pada hari kerja penuh di mana sebagian besar industri dan perkantoran melakukan aktifitasnya.

- Tanggal 6 Desember 2003 adalah hari sabtu, pada hari tersebut kebutuhan energi relative sedang, karena sebagian industri dan perkantoran melakukan aktifitas kerjanya setengah hari. Dengan kata lain bahwa hari tersebut mewakili beban pada hari kerja efektif setengah penuh.
- Tanggal 7 Desember 2003 adalah hari minggu, pada hari tersebut kebutuhan energi relative rendah karena banyak industri dan perkantoran tidak melakukan aktifitas. Oleh karena itu tepat sekali kalau hari tersebut mewakili beban pada hari libur.

Berdasarkan data unit termal yang terdapat dalam PT. Pembangkit Jawa Bali pada sistem tenaga pada table 3.1, ternyata saat dilakukan pengambilan data, semua unit pembangkit dalam kondisi siap beroperasi. Maka dapat di susun input dan unit pembangkit termal yang siap beroperasi pada tanggal 4, 6 dan 7 Desember 2003, yaitu sebanyak 38 unit pembangkit.

Dalam data beban harian system pada PT. Pembangkit Jawa Bali, Terdapat data hasil perhitungan mengenai jumlah total pembangkitan, beban total dan cadangan berputar pada tiap jam dalam tiap-tiap area. Data-data ini tidak di pakai dalam skripsi ini karena data tersebut menyangkut sistem secara keseluruhan dalam suatu area. Dalam suatu area biasanya terdapat lebih dari satu perusahaan penyediaan energi listrik. Misalnya pada area IV, Terdapat tiga perusahaan penyedia energi listrik, yaitu : PT. Pembangkit Jawa Bali, PT. Indonesia Power dan perusahaan swasta. Jadi tidak relevan jika data-data tersebut digunakan dalam skripsi ini, sedangkan aplikasi pada skripsi ini hanya pada PT. Pembangkit Jawa Bali (PJB).

### 3.4. Beban Sistem

Dalam wilayah Jawa-Bali, Pembangkit-pembangkit yang ada dikoordinasi oleh PT. Pembangkit Jawa Bali. Proses penjadwalan pembangkit dengan metode *parallel simulated annealing dengan komputasi spekulatif* bertujuan untuk membuat rencana penjadwakan pembangkit dalam biaya operasi yang seekonomis mungkin.

Untuk mengetahui seberapa besar efisiensi dari metode ini, maka dilakukan evaluasi dengan mengambil data unit pembangkit termal dan beban yang ditanggung oleh PT. Pembangkit Jawa Bali sebagai beban perbandingan. Sedangkan kombinasi jadwal dan daya output pembangkit tenaga listrik dalam sistem PT. Pembangkit Jawa Bali tanggal 4, 6 dan 7 Desember 2003, terdapat pada lampiran. Untuk beban sistem terdapat pada table 3.3 (beban sistem yang ditanggung oleh pembangkit termal saja).

PT. Pembangkit Jawa Bali tidak mempunyai dasar yang pasti untuk menentukan nilai dari *spining reserve* (cadang berputar) tiap periode jam, tapi PT. Pembangkit Jawa Bali menggunakan ketetapan bahwa nilai cadangan berputar di ambil dari daya terpasang terbesar dari unit pembangkit yang mengalami gagal operasi. Dalam hal ini pada PT. Pembangkit Jawa Bali daya terpasang adalah dari unit pembangkit PLTU Paiton yaitu sebesar 400 MW sebagai nilai cadangan berputar tiap periode jam.

Tabel 3.3.

## Data Beban Unit Termal Pada PT. PJB

Jam	Kamis 4 Desember 2003		Sabtu 6 Desember 2003		Minggu 7 Desember 2003	
	Beban Sistem (MW)	Cadangan Berputar (MW)	Beban Sistem (MW)	Cadangan Berputar (MW)	Beban Sistem (MW)	Cadangan Berputar (MW)
01.00	3205	400	3140	400	2956	400
02.00	3205	400	3000	400	2860	400
03.00	3205	400	3000	400	2860	400
04.00	3205	400	3000	400	2860	400
05.00	3290	400	3110	400	2869	400
06.00	2880	400	2712	400	2869	400
07.00	2790	400	2682	400	2640	400
08.00	3220	400	3020	400	2450	400
09.00	3275	400	3105	400	2520	400
10.00	3275	400	3105	400	2620	400
11.00	3275	400	3105	400	2620	400
12.00	3195	400	3025	400	2570	400
13.00	3210	400	2890	400	2570	400
14.00	3260	400	2849	400	2545	400
15.00	3357	400	2806	400	2587	400
16.00	3347	400	2804	400	2587	400
17.00	3525	400	2814	400	2790	400
18.00	3650	400	3700	400	3685	400
19.00	3820	400	3700	400	3685	400
20.00	3770	400	2685	400	3685	400
21.00	3540	400	3330	400	3370	400
22.00	3360	400	3120	400	3220	400
23.00	3345	400	3095	400	3195	400
24.00	3205	400	3080	400	3050	400

## **^BAB IV**

### **ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL DENGAN METODE SIMULATED ANNEALING PADA PT PJB**

#### **4.1. Penggunaan Program Komputer Metode Parallel SA**

Untuk memecahkan permasalahan yang ada pada metode *Parallel Simulated Annealing (PSA)*, untuk menentukan komitmen unit pembangkit termal digunakan bantuan program komputer. Dengan menggunakan program komputer ini, maka waktu komputasi yang dibutuhkan pada saat menganalisis data akan jauh lebih cepat dibandingkan dengan melakukan perhitungan secara manual. Karena data yang di analisis banyak dan metode parallel simulated annealing sendiri terdiri dari beberapa algoritma. Untuk memecahkan masalah pada skripsi ini, maka dibutuhkan ketelitian yang tinggi pada program komputer metode Parallel Simulated Annealing (PSA) sehingga hasil analisis yang ditampilkan sesuai dengan tujuan.

Program komputer ini menggunakan bahasa pemrograman *Borland Delphi versi 7.0.*, yang merupakan bahasa pemrograman yang terstruktur yang relatif mudah untuk dipelajari dan mudah penggunaannya.

#### **4.2. Algoritma Program.**

Urutan langkah-langkah dalam program computer ini dapat dilihat pada algoritma program berikut:

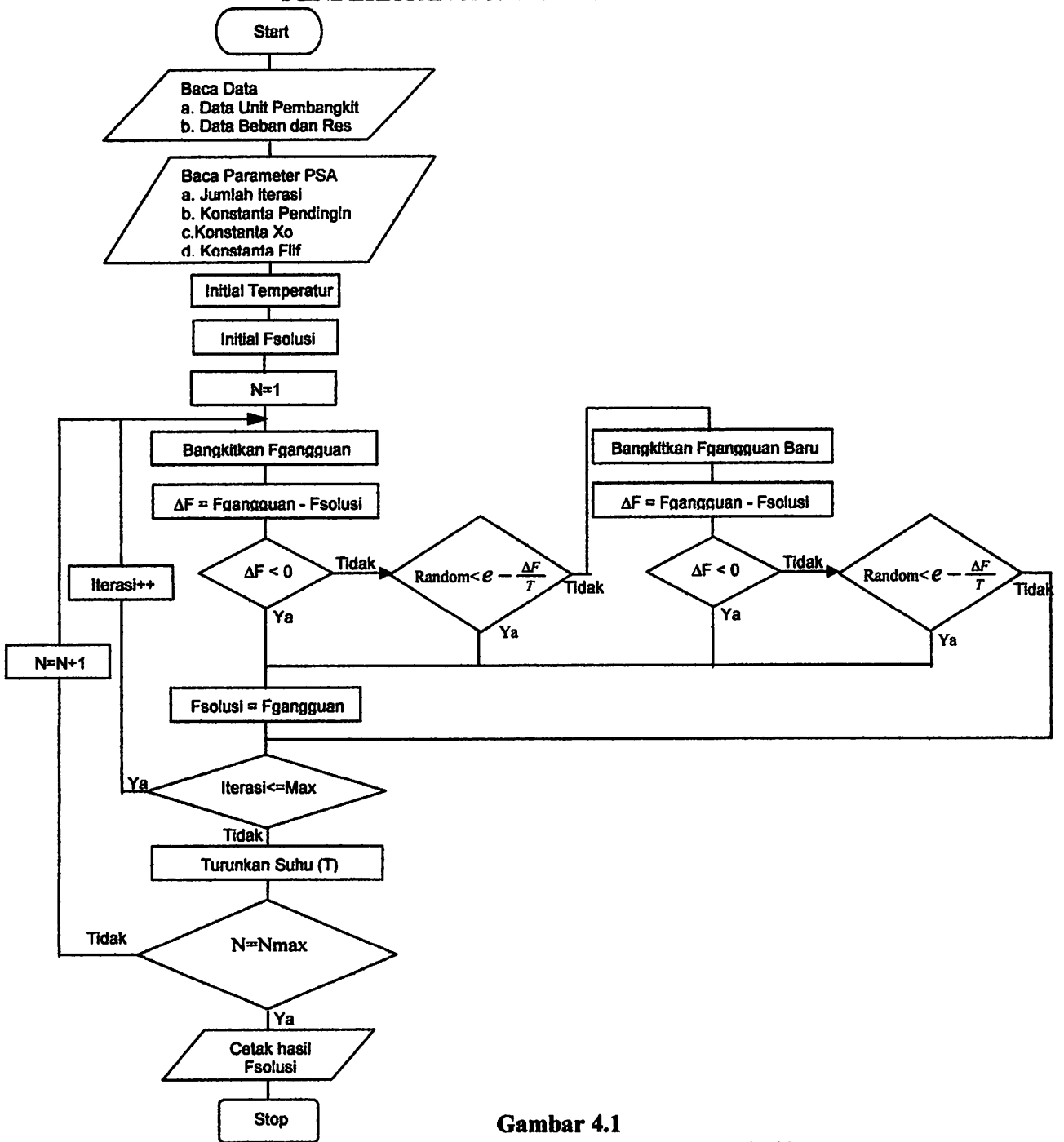
1. Masukkan jumlah dan data setiap unit pembangkit termal.

Data tiap-tiap unit pembangkit termal yang diperlukan adalah jumlah unit pembangkit, daya maksimum dan daya minimum, konstanta persamaan biaya bahan bakar, harga bahan bakar, *biaya start up time*, *minimum down time* dan beban sistem.

2. Masukkan parameter *parallel simulated annealing* dengan kombinasi *komputasi spekulatif* dan *subset serial*, yang meliputi banyaknya jumlah iterasi, nilai konstanta pendingin ( $r$ ), nilai konstanta  $X_0$  dan nilai konstanta *flip*.
3. Masukkan initial Fsolusi  
Inisialisasi U, tentukan inisialisasi temperatur T dan jumlah iterasi M dan tentukan total biaya operasi F yang diberikan.
4. Bangkitkan F gangguan
5. Apakah delta F lebih kecil dari nol jika ya Fsolusi = Fgangguan jika tidak,  $\text{random} < e^{-\Delta F/T}$ . Jika diterima Fsolusi = Fgangguan dan jika tidak maka bangkitkan gangguan baru seperti langkah 1,2,3,4
6. Apakah iterasi  $\leq \text{Max}$ , jika Ya maka kembali ke langkah 3 dan jika Tidak ke langkah selanjutnya
7. Cari Fsolusi yang paling max
8. Turunkan suhu (T)
9. Apakah T sudah dingin, Jika Tidak kembali ke langkah 3 dan jika Ya ke langkah berikutnya
10. Cetak hasil F solusi
11. Stop

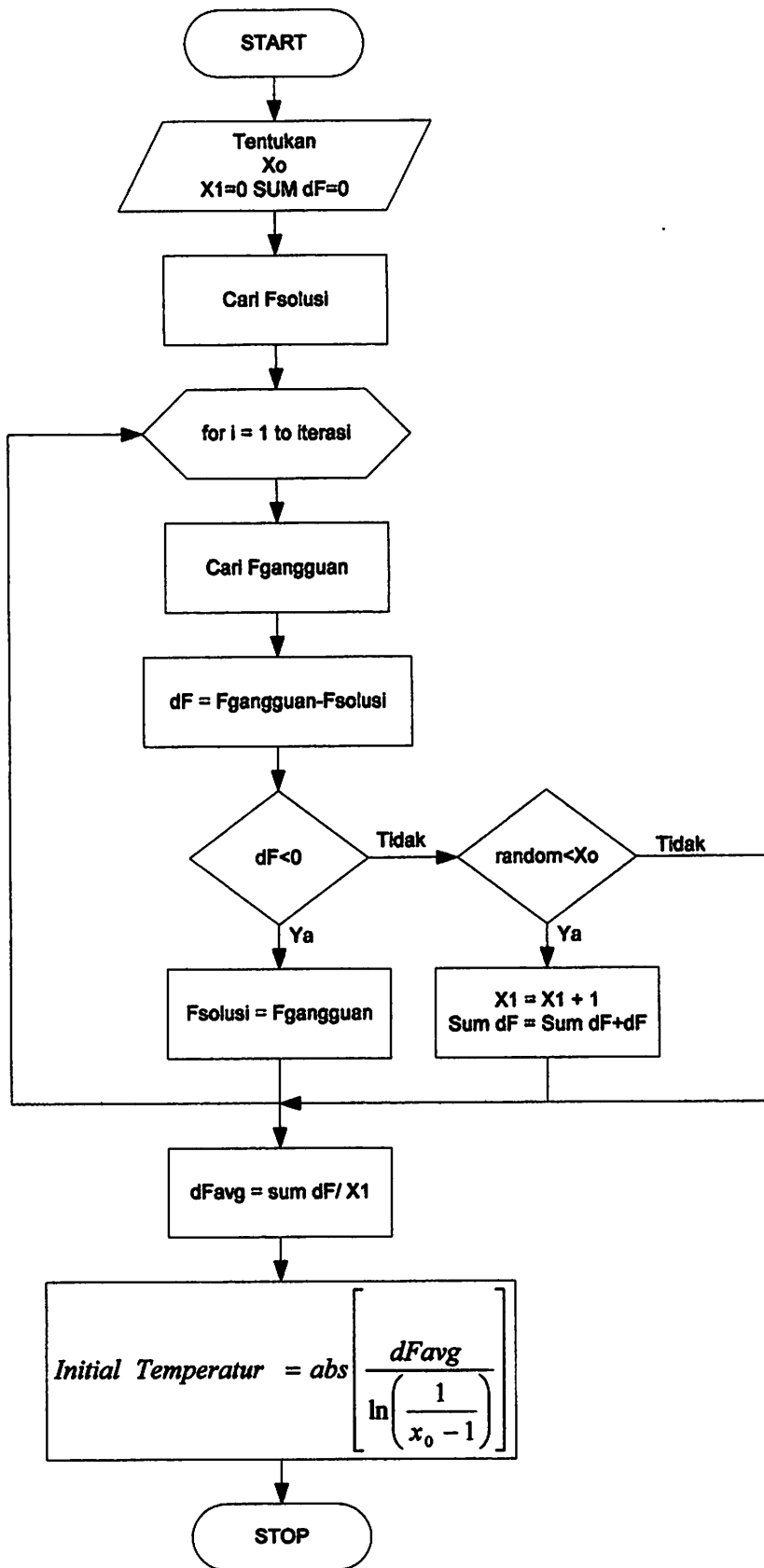
### 4.3 Flowchart Algoritma Program

#### FLOWCHART PARALLEL SIMULATED ANNEALING DENGAN PENDEKATAN KOMPUTASI SPEKULATIF



Gambar 4.1  
Flowchart SA dengan Komputasi Spekulatif





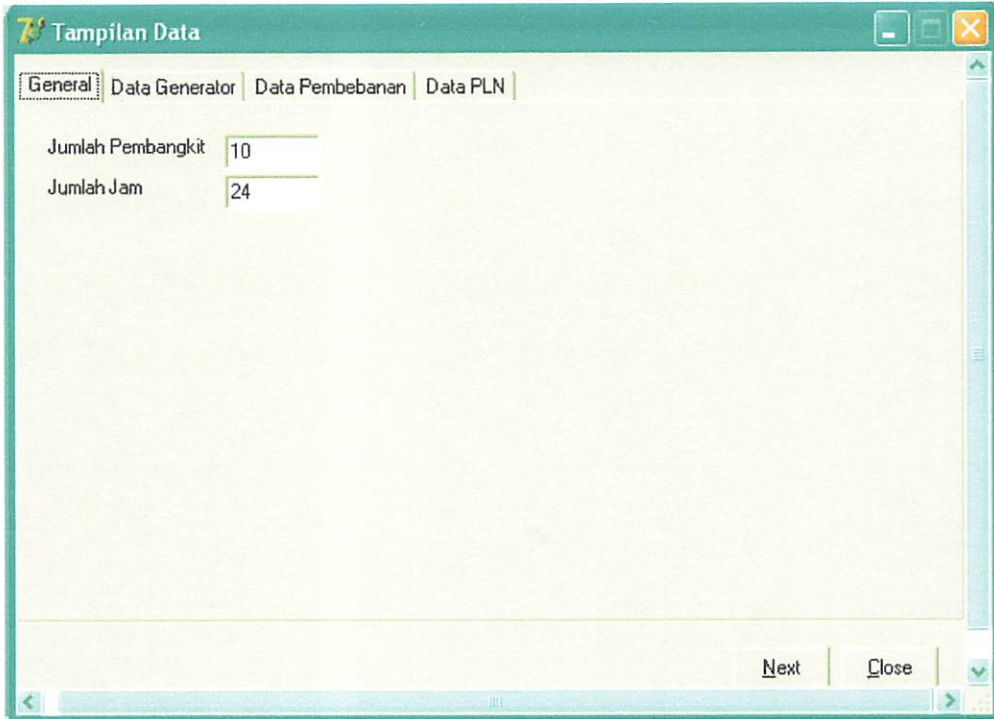
**Flowchart Untuk Initial Temperatur Pada SA**

#### 4.4. Validasi Program.

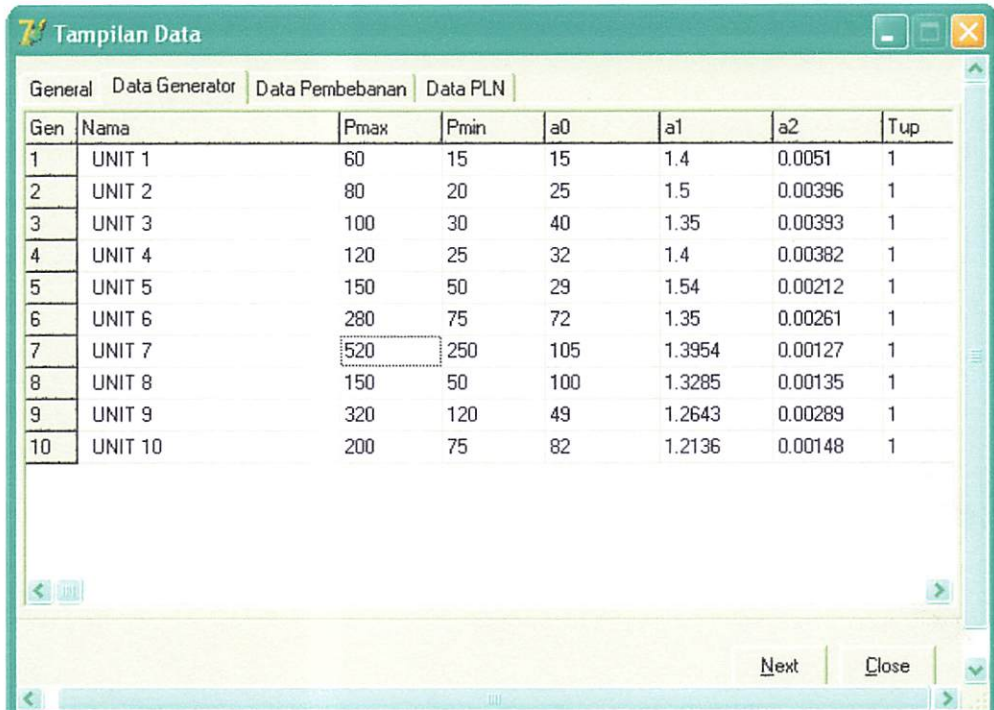
Uji validasi program dilakukan untuk mengetahui apakah program tersebut layak atau tidak untuk digunakan. Salah satu cara untuk mengujinya menggunakan program tersebut untuk mengolah data dari journal, dan hasilnya dibandingkan dengan hasil journal. Apabila hasilnya serupa atau mendekati dengan journal, maka program dinyatakan layak untuk digunakan.



Gambar 4.2. Menu Utama Hasil Uji Validasi Metode *PSA* Dengan *KS*



**Gambar 4.3. Tampilan Data Untuk Validasi**  
b



**Gambar 4.4. Data Generator Untuk Validasi**

Tampilan Data

General | Data Generator | Data Pembebanan | Data PLN

	Load	Res
1	1460	146
2	1390	139
3	1300	130
4	1300	130
5	1295	129.5
6	1350	135
7	1400	140
8	1350	135
9	1295	129.5
10	1260	126
11	1200	120
12	1195	119.5
13	1150	115
14	1120	112
15	1110	111

Next Close

Gambar 4.5. Data Pembebanan Untuk Validasi

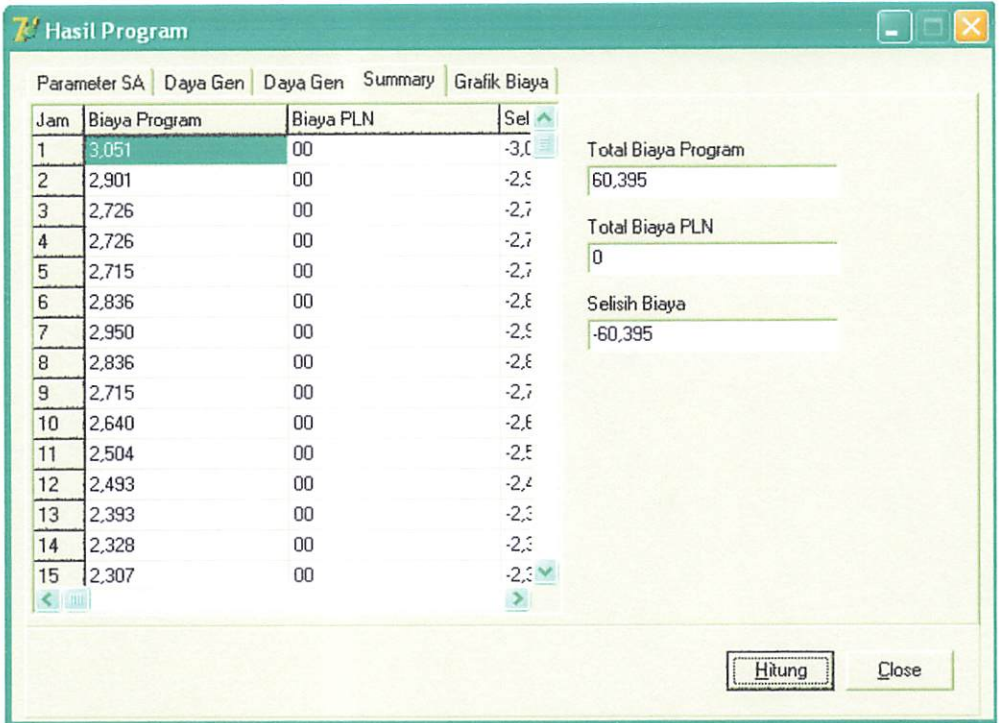
Hasil Program

Parameter SA | Daya Gen | Daya Gen | Summary | Grafik Biaya

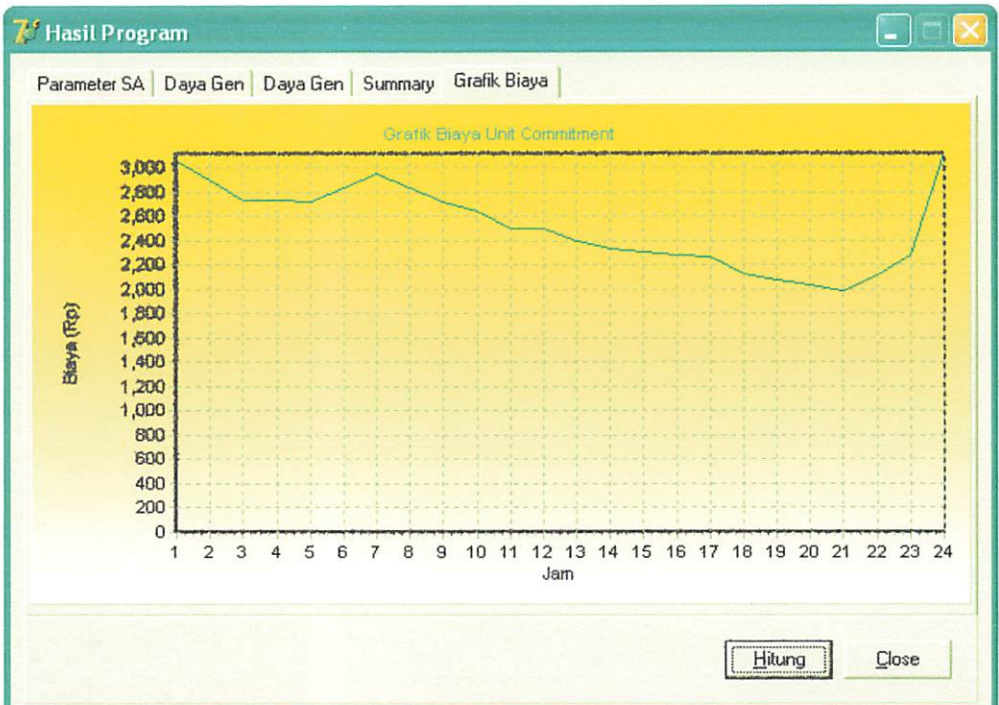
	Jam 1	Jam 2	Jam 3	Jam 4	Jam 5	Jam 6	Jam 7	Jam 8	Jam 9
Unit 1	1	1	1	1	1	1	1	1	1
Unit 2	1	1	1	1	1	1	1	1	1
Unit 3	1	1	1	1	1	1	1	1	1
Unit 4	1	1	1	1	1	1	1	1	1
Unit 5	1	1	1	1	1	1	1	1	1
Unit 6	1	1	1	1	1	1	1	1	1
Unit 7	1	1	1	1	1	1	1	1	1
Unit 8	0	0	0	0	0	0	0	0	0
Unit 9	1	1	1	1	1	1	1	1	1
Unit 10	1	1	1	1	1	1	1	1	1

Hitung Close

Gambar 4.6. Hasil Perhitungan Untuk Daya Generator Pada Validasi



Gambar 4.7. Hasil Perhitungan Biaya Validasi



Gambar 4.8. Kurva Biaya Oprasional Validasi

**Tabel 4.1.**

**Perbandingan Biaya Hasil Journal Dengan Hasil Validasi**

<b>Hasil Biaya Journal (\$)</b>	<b>Hasil Biaya Validasi (\$)</b>	<b>Selisih Hasil Biaya (%)</b>
59520	60395	1,4

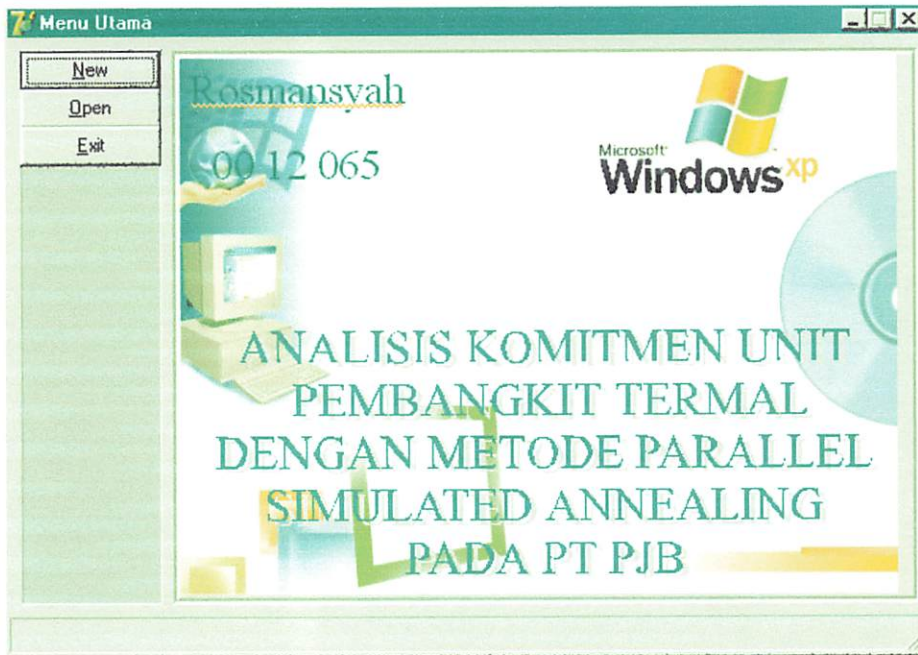
Setelah memperhatikan hasil biaya dari journal dengan biaya uji validasi, maka program metode *Parallel Simulated Annealing Dengan Kombinasi Komputasi Spekulatif Dan Subset Serial* dinyatakan layak digunakan.

**4.5. Tampilan Program**

Program dalam skripsi ini dijalankan dengan computer berprosesor Intel Pentium IV, dengan memori 256 Mb. Mengenai jalannya program ikuti prosedur program sebagai berikut :

1. Menu utama dari program.

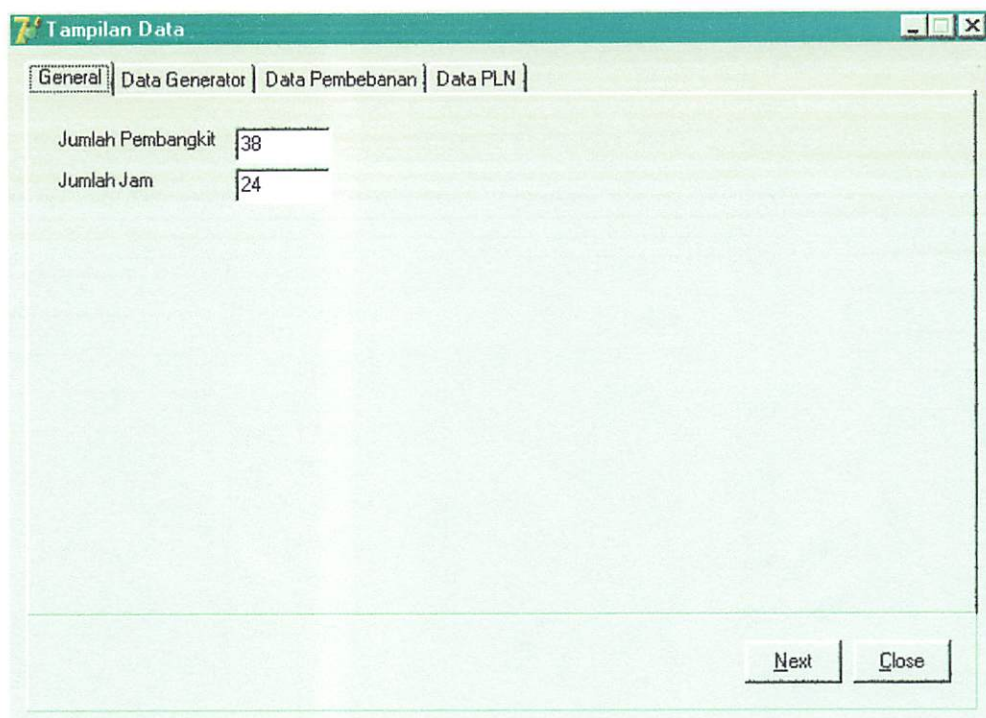
Tekan tombol open untuk membuka data base yang sudah tersimpan.



**Gambar 4.9. Menu Utama**

## 2. Tampilan data.

Berisi general data, data pembangkit, data pembebanan, data PLN. Kemudian tekan general data untuk mengetahui jumlah pembangkit dan jumlah jam.



The screenshot shows a software window titled "Tampilan Data" with a green header bar. Below the header, there are four tabs: "General", "Data Generator", "Data Pembebanan", and "Data PLN". The "General" tab is selected. The main area of the window displays two data entries: "Jumlah Pembangkit" with a value of 38, and "Jumlah Jam" with a value of 24. At the bottom right of the window, there are two buttons: "Next" and "Close".

Jumlah Pembangkit	38
Jumlah Jam	24

### a. Tampilan General Data



3. Tekan tombol data generator, untuk mengetahui data penawaran PT. Pembangkit Jawa Bali.

Gen	Nama	Pmax	Pmin	a0	a1	a2	Tup
1	PLTU Paiton 1	370	225	3244978	111712.2	10.2971	72
2	PLTU Paiton 2	370	225	3244978	111712.2	10.2971	72
3	PLTGU Gresik GT 1.1	102	53	5467532.4	217963.5	34.155	36
4	PLTGU Gresik GT 1.2	102	53	5467532.4	217963.5	34.155	36
5	PLTGU Gresik GT 1.3	102	53	5467532.4	217963.5	34.155	36
6	PLTGU Gresik ST 1.0	480	250	10936203	72527	368.875	36
7	PLTGU Gresik GT 2.1	102	53	5467532.4	217963.5	34.155	36
8	PLTGU Gresik GT 2.2	102	53	5467532.4	217963.5	34.155	36
9	PLTGU Gresik GT 2.3	102	53	5467532.4	217963.5	34.155	36
10	PLTGU Gresik ST 2.0	480	250	10936203	72527	368.875	36
11	PLTGU Gresik GT 3.1	102	53	5467532.4	217963.5	34.155	36
12	PLTGU Gresik GT 3.2	102	53	5467532.4	217963.5	34.155	36
13	PLTGU Gresik GT 3.3	102	53	5467532.4	217963.5	34.155	36
14	PLTGU Gresik ST 3.0	480	250	10936203	72527	368.875	36

### b. Tampilan Data Generator

4. Tekan tombol data pembebanan, untuk mengetahui beban tiap jam.

	Load	Res
1	3205	400
2	3035	400
3	3035	400
4	3085	400
5	3290	400
6	2880	400
7	2790	400
8	3220	400
9	3275	400
10	3275	400
11	3275	400
12	3195	400
13	3210	400
14	3260	400
15	3257	400

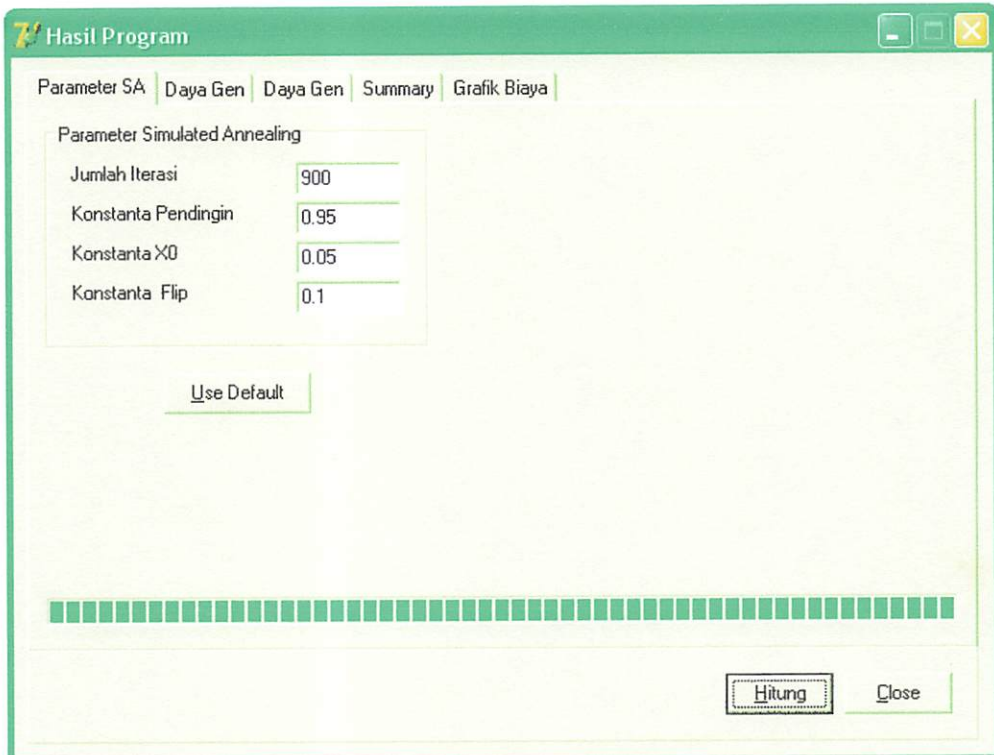
### c. Tampilan Data Pembebanan

5. Tekan tombol data PLN untuk mengetahui data dari PLN dan kemudian tekan tombol Next untuk memulai perhitungannya.

	Jam 1	Jam 2	Jam 3	Jam 4	Jam 5	Jam 6	Jam 7	Ja
Gen 1	370	370	370	370	370	325	325	37
Gen 2	370	370	370	370	370	325	325	37
Gen 3	0	0	0	0	0	0	0	0
Gen 4	0	0	0	0	0	0	0	0
Gen 5	0	0	0	0	0	0	0	0
Gen 6	300	300	300	300	300	250	250	30
Gen 7	0	0	0	0	0	0	0	0
Gen 8	0	0	0	0	0	0	0	0
Gen 9	0	0	0	0	0	0	0	0
Gen 10	300	250	250	300	350	250	250	25
Gen 11	0	0	0	0	0	0	0	0
Gen 12	0	0	0	0	0	0	0	0
Gen 13	0	0	0	0	0	0	0	0
Gen 14	300	250	250	250	300	250	250	27

d. Tampilan Data PLN

6. Setelah memasukkan *input* yang dibutuhkan, maka komputasi dapat dilakukan. Tampilan dari perhitungan tes program dapat di lihat pada gambar di bawah ini.



#### e. Tampilan Parameter Simulated Annealing

Di mana :

- Jumlah iterasi menunjukkan banyaknya iterasi yang mungkin dilakukan, jika pada nilai iterasi tersebut belum memperoleh hasil yang optimal maka dapat di rubah pada nilai yang lebih besar.
- Konstanta pendingin merupakan nilai yang berpengaruh terhadap factor kecepatan pendinginan.
- Konstanta X0 merupakan nilai kemungkinan diterimanya solusi yang lebih jelek / mahal di terima sebagai solusi yang optimal pada inisial X.

- Konstanta filp merupakan suatu nilai yang berfungsi untuk membangkitkan solusi lain yang sesuai apabila solusi percobaan ditolak.

7. Tekan tombol hitung, untuk mengetahui data yang akan di hitung.

Parameter SA Daya Gen | Daya Gen | Summary | Grafik Biaya |

	Jam 1	Jam 2	Jam 3	Jam 4	Jam 5	Jam 6	Jam 7	Jam 8	Jam 9
Unit 1	1	1	1	1	1	1	1	1	1
Unit 2	1	1	1	1	1	1	1	1	1
Unit 3	0	0	0	0	0	0	0	0	0
Unit 4	1	1	1	1	1	1	1	1	1
Unit 5	0	0	0	0	0	0	0	0	0
Unit 6	1	1	1	1	1	1	1	1	1
Unit 7	1	1	1	1	1	1	1	1	1
Unit 8	1	1	1	1	1	1	1	1	1
Unit 9	1	1	1	1	1	1	1	1	1
Unit 10	1	1	1	1	1	1	1	1	1
Unit 11	0	0	0	0	0	0	0	0	0
Unit 12	0	0	0	0	0	0	0	0	0
Unit 13	0	0	0	0	0	0	0	0	0
Unit 14	1	1	1	1	1	1	1	1	1
Unit 15	1	1	1	1	1	1	1	1	1

Hitung Close

**f. Tampilan Hasil Perhitungan Daya Generator**

8. Tekan tombol summary untuk melihat selisih biaya program dan biaya dari PLN.

The screenshot shows a software window titled "7 Hasil Program" with a green header. It contains a table with four columns: "Jam", "Biaya Program", "Biaya PLN", and "Sel". To the right of the table, there are summary rows for "Total Biaya Program", "Total Biaya PLN", and "Selisih Biaya". At the bottom right, there are two buttons: "Hitung" and "Close".

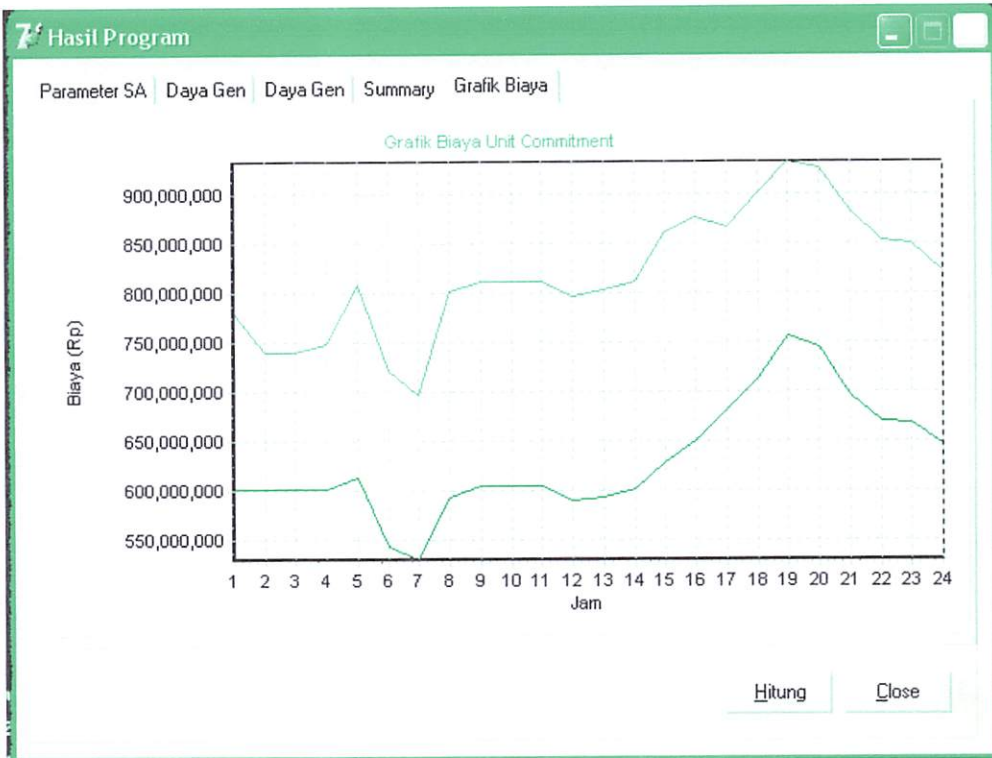
Jam	Biaya Program	Biaya PLN	Sel
1	601,741,741	781,204,639	17%
2	601,741,741	741,054,874	13%
3	601,741,741	741,054,874	13%
4	601,741,741	748,438,388	14%
5	614,440,422	809,418,829	19%
6	544,137,826	722,632,365	17%
7	530,790,394	696,745,625	16%
8	594,783,391	801,869,811	20%
9	604,973,620	812,383,597	20%
10	604,973,620	812,383,597	20%
11	604,973,620	812,383,597	20%
12	591,047,498	797,476,311	20%
13	593,288,806	803,923,951	21%
14	601,835,476	811,307,465	20%
15	628,257,745	861,165,315	23%

Total Biaya Program	15,155,177,773
Total Biaya PLN	19,668,568,812
Selisih Biaya	4,513,391,039

g. Tampilan Hasil Perhitungan Biaya Total Pembangkit

9. Tekan tombol grafik biaya untuk melihat grafik perhitungannya



**h. Tampilan Grafik Pembebanan Terhadap Biaya PT. PJB  
Dan Biaya Program**

**Gambar 4.10.**

**Tampilan Hasil Tes Program  
Menggunakan Metode *PSA Dengan KS***

10. Tekan close untuk keluar program.

#### **4.6. Analisa Program Dan Hasil Perhitungan.**

Program optimalisasi penjadwalan pembebanan unit termal pada sistem PT. Pembangkit Jawa Bali menggunakan metode *Parallel Simulated Annealing Dengan Komputasi Spekulatif*, terdiri dari empat tahap yang kesemuanya harus dilakukan secara berurutan :

1. Tahap input data dengan inialisasi masukkan data karakteristik unit pembangkit termal dan beban.
2. Proses inialisasi parameter *parallel simulated annealing*.
3. Melakukan proses *parallel simulated annealing dengan komputasi spekulatif* dimana akan menghasilkan komitmen unit dengan biaya operasional yang lebih murah.

Pada kombinasi penjadwalan unit termal pada PT. Pembangkit Jawa Bali unit 1 sampai unit 38 pada periode waktu 24 jam, dapat di lihat pada tabel 4.1, 4.2, dan 4.3. Hal tersebut dipergunakan sebagai bahan perbandingan dengan hasil dari PT. Pembangkit Jawa Bali dan hasil metode *parallel simulated annealing dengan komputasi spekulatif*, yang dapat di lihat pada tabel 4.4, 4.5, dan tabel 4.6.



Tabel 4.2

Kombinasi Penjadwalan Unit Termal Pada PT Pembangkit Jawa Bali

Kamis, 4 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
2	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
3	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
4	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
5	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
6	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
7	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
8	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
9	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
10	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
11	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
12	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
13	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
14	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
15	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
16	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
17	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
18	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
19	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
20	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
21	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
22	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
23	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
24	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1

Tabel 4.3

Kombinasi Penjadwalan Unit Termal Pada PT Pembangkit Jawa Bali

Sabtu, 6 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1 1
2	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
3	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
4	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
5	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
6	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
7	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
8	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
9	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
10	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
11	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
12	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
13	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
14	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
15	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
16	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
17	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
18	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
19	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
20	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
21	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
22	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
23	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
24	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1

Tabel 4.4

Kombinasi Penjadwalan Unit Termal Pada PT Pembangkit Jawa Bali

Minggu, 7 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1 1
2	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
3	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
4	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
5	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
6	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
7	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
8	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
9	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
10	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
11	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 1 0 0 1 1 1 1 1 1
12	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
13	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
14	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
15	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
16	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
17	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
18	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
19	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
20	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
21	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
22	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
23	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1
24	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 1 1 1 1 1

Tabel 4.5

Kombinasi Penjadwalan Unit Pembangkit Termal

Pada PT Pembangkit Jawa Bali Menggunakan Metode Parallel Simulated

Annealing Dengan Komputasi Spekulatif

Kamis, 4 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 1 0 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
2	1 1 0 1 0 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
3	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
4	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
5	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
6	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
7	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
8	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
9	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
10	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
11	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
12	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
13	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
14	1 1 0 1 0 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
15	1 1 0 1 0 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
16	1 1 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
17	1 1 0 1 0 1 1 1 1 1 1 1 1 0 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
18	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
19	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
20	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
21	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
22	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
23	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
24	1 1 0 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1

Tabel 4.6

Kombinasi Penjadwalan Unit Pembangkit Termal

Pada PT Pembangkit Jawa Bali Menggunakan Metode Parallel Simulated

Annealing Dengan Komputasi Spekulatif

Sabtu, 6 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 0 0 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
2	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
3	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
4	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
5	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
6	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
7	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
8	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
9	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
10	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
11	1 1 0 0 0 1 1 1 1 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
12	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
13	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
14	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
15	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
16	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
17	1 1 0 0 0 1 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
18	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
19	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
20	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
21	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
22	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
23	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1
24	1 1 0 1 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 1 0 0 0 1 1

Tabel 4.7

Kombinasi Penjadwalan Unit Pembangkit Termal

Pada PT Pembangkit Jawa Bali Menggunakan Metode Parallel Simulated

Annealing Dengan Komputasi Spekulatif

Minggu, 7 Desember 2003

Jam	Status ON dan OFF Pembangkit
1	1 1 0 0 0 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
2	1 1 0 0 0 1 1 0 0 1 1 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
3	1 1 0 0 0 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
4	1 1 0 0 0 1 1 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
5	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
6	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
7	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
8	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
9	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
10	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
11	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
12	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
13	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
14	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
15	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
16	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
17	1 1 0 0 0 1 0 0 0 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 1 1
18	1 1 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
19	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
20	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
21	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
22	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
23	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1
24	1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 1 0 0 0 1 1

Proses optimasi telah selesai dan dianggap sudah optimal karena sudah tidak dapat pelanggaran kendala-kendala dari unit pembangkit termal. Kemudian dilakukan perhitungan biaya total dari PT. Pembangkit Jawa Bali biaya total dengan metode *parallel simulated annealing dengan komputasi spekulatif* dalam tiap jam. Hasil perbandingan dapat dilihat pada tabel 4.7, 4.8 dan 4.9.

Tabel 4.8

Perbandingan Biaya Operasional Per Jam PT. Pembangkiti Jawa Bali  
 Dengan Metode *Parallel Simulated Annealing* Dengan *Komputasi Spekulatif*

Kamis, 4 Desember 2003

Jam	Biaya Program	Biaya PLN	Selisih Biaya
1	601.741.741	781.204.634	179.462.898
2	601.741.741	741.054.874	139.313.134
3	601.741.741	741.054.874	139.313.134
4	601.741.741	748.438.388	146.696.648
5	614.440.422	809.418.829	194.978.407
6	544.137.826	722.6312.365	178.494.539
7	530.790.394	696.746.625.	165.955.231
8	594.783.391	801.869.811	207.409.977
9	604.973.620	812.383.597	207.409.977
10	604.973.620	812.383.597	207.409.977
11	604.973.620	812.383.597	207.409.977
12	591.047.498	797.476.311	206.428.813
13	593.288.806	803.923.951	210.635.145
14	601.835.476	811.307.465	209.471.990
15	628.257.745	861.165.315	232.907.570
16	650.551.278	876.618.255	226.066.977
17	681.530.546	867.105.192	185.574.646
18	714.208.631	902.209.097	188.000.466
19	757.124.429	933.486.177	176.361.748
20	746.044.536	926.079.893	180.035.357
21	697.775.580	884.146.106	186.370.526
22	670.918.411	853.169.554	182.251.143
23	668.684.754	849.176.415	181.031.661
24	647.870.226	822.594.884	174.724.658

*Total Biaya Program* : 15,155,177,773

*Total Biaya PLN* : 19,668,568,812

*Selisih Biaya* : 4,513,391,039



Tabel 4.9

Perbandingan Biaya Operasional Per Jam PT. PJB

Dengan Metode *Parallel Simulated Annealing Komputasi Spekulatif*

Sabtu, 6 Desember 2003

Jam	Biaya Program	Biaya PLN	Selisih Biaya
1	574.196.990	733.672.317	159.475.327
2	553.313.181	694.079.607	140.766.427
3	553.313.181	694.079.607	140.766.427
4	553.313.181	694.079.607	140.766.427
5	569.716.879	714.089.019	144.372.140
6	510.539.317	650.750.134	140.210.817
7	506.098.188	646.344.618	140.246.430
8	556.292.939	701.787.618	145.494.877
9	568.970.459	714.846.727	145.876.268
10	568.970.459	714.846.727	145.876.268
11	568.970.459	714.846.727	145.876.268
12	548.389.978	700.858.929	152.468.451
13	528.277.636	681.580.859	153.303.223
14	522.180.397	617.888.666	149.708.269
15	515.791.214	660.923.317	145.132.103
16	515.494.180	661.794.495	146.300.315
17	516.979.474	663.268.965	146.288.991
18	725.175.585	860.233.798	135.058.213
19	725.175.539	860.233.798	135.058.259
20	563.536.835	856.780.659	293.243.824
21	657.229.140	755.513.065	98.283.925
22	626.005.436	720.214.886	94.209.451
23	622.297.246	716.537.361	94.240.114
24	620.073.243	713.084.222	93.010.978

*Total Biaya Program* : 13,770,301,135

*Total Biaya PLN* : 17,196,334,926

*Selisih Biaya* : 3,426,033,791

Tabel 4.10

Perbandingan Biaya Operasional Per Jam PT. Pembangkit Jawa Bali  
 Dengan Metode *Parallel Simulated Annealing* Dengan *Komputasi Spekulatif*  
 Minggu, 7 Desember 2003

Jam	Biaya Program	Biaya PLN	Selisih Biaya
1	573.917.028	686.090.040	112.173.012
2	515.150.193	673.676.463	157.526.270
3	515.150.193	672.676.463	157.526.270
4	515.150.193	680.535.993	165.385.800
5	516.489.841	683.936.632	167.446.791
6	511.975.187	628.773.489	116.798.301
7	473.396.236	592.701.983	119.305.746
8	445.249.473	600.979.539	155.730.067
9	455.606.582	617.779.194	162.172.612
10	470.428.258	617.779.194	147.350.936
11	470.428.258	617.779.194	147.350.936
12	463.013.625	610.372.910	147.359.285
13	463.013.625	610.372.910	147.359.285
14	459.309.154	606.372.910	147.391.922
15	465.533.748	644.121.019	178.587.270
16	465.533.748	644.121.019	178.587.270
17	504.739.186	656.039.778	151.295.592
18	716.462.750	855.916.773	139.354.022
19	716.462.697	855.916.773	139.354.076
20	716.462.697	855.916.773	139.354.076
21	653.984.016	765.949.349	111.965.332
22	631.625.356	758.497.525	126.872.168
23	627.905.554	737.978.111	110.072.557
24	606.368.121	706.281.975	99.913.854

*Total Biaya Program* : 12,953,355,720  
*Total Biaya PLN* : 16,379,589,172  
*Selisih Biaya* : 3,426,233,452

Bila di buat suatu perbandingan biaya untuk tiap jam (lihat table 4.7, 4.8 dan 4.9) urutan biaya tidak selalu tetap, Tetapi biaya perjam untuk metode *Parallel Simulated Annealing dengan Komputasi Spekulatif* lebih kecil dari biaya PT. Pembangkit Jawa Bali.

Biaya total setiap satu hari dibandingkan untuk kedua cara, yaitu pada PT. Pembangkit Jawa Bali dan metode *parallel simulated annealing dengan komputasi spekulatif* dapat di lihat pada tabel 4.10 dibawah ini.

Tabel 4.11  
Perbandingan Total Biaya Operasional PT. PJB  
Dengan Total Biaya Program

Periode Waktu (24 jam)	Total Biaya PT. PJB (Rupiah)	Total Biaya Program (Rupiah)	Selisih Biaya (%)
Kamis, 4 Desember 2003	19,668,568,812	15,155,177,773	22.95
Sabtu, 6 Desember 2003	17,196,334,926	13,770,301,135	19.92
Minggu, 7 Desember 2003	16,379,589,172	12,953,355,720	20.92

Tabel 4.10 menunjukkan bahwa dengan menggunakan metode *parallel simulated annealing dengan komputasi spekulatif* terdapat pengurangan total biaya operasional pembangkitan dalam tiap periode 24 jam (1 hari). Hal ini berarti bahwa proses pemrograman menyebabkan pengurangan biaya operasi pembangkit yang cukup besar di banding dengan biaya operasi PT. PJB.

Kurva perbandingan hasil optimasi biaya total dari PT. PJB dan biaya total pemrograman dapat dilihat pada grafik 4.1, 4.2, dan 4.3.

Bila di buat suatu perbandingan biaya untuk tiap jam (lihat table 4.7, 4.8 dan 4.9) urutan biaya tidak selalu tetap. Tetapi biaya perjam untuk metode Paralel simulated annealing dengan komposisi spektrum lebih kecil dari biaya PT. Pembangunan Jawa Bali.

Biaya total setiap satu hari dibandingkan untuk kedua cara yaitu pada PT. Pembangunan Jawa Bali dan metode Paralel simulated annealing dengan komposisi spektrum/gabar di lihat pada tabel 4.10 dibawah ini.

Tabel 4.11  
Perbandingan Total Biaya Operasional PT. PJB  
Dengan Total Biaya Program

Periode Waktu (24 jam)	Total Biaya PT. PJB (Rupiah)	Total Biaya Program (Rupiah)	Selisih Biaya (%)
Kamis 4 Desember 2003	19.668.268,812	12.122.177,773	38,92
Jabat 6 Desember 2003	17.190.334,020	13.770.301,132	19,92
Minggu 7 Desember 2003	16.370.280,172	12.023.322,720	26,92

Tabel 4.10 menunjukkan bahwa dengan menggunakan metode Paralel simulated annealing dengan komposisi spektrum terdapat pengurangan total biaya operasional (dalam tiap periode 24 jam) (lihat Hal ini berarti bahwa proses programan menghasilkan pengurangan biaya operasi pembangkit yang cukup besar di banding dengan biaya operasi PT. PJB. Kalau dibandingkan hasil optimasi biaya total dari PT. PJB dan biaya total programan gabar dilihat pada grafik 4.1, 4.2, dan 4.3.

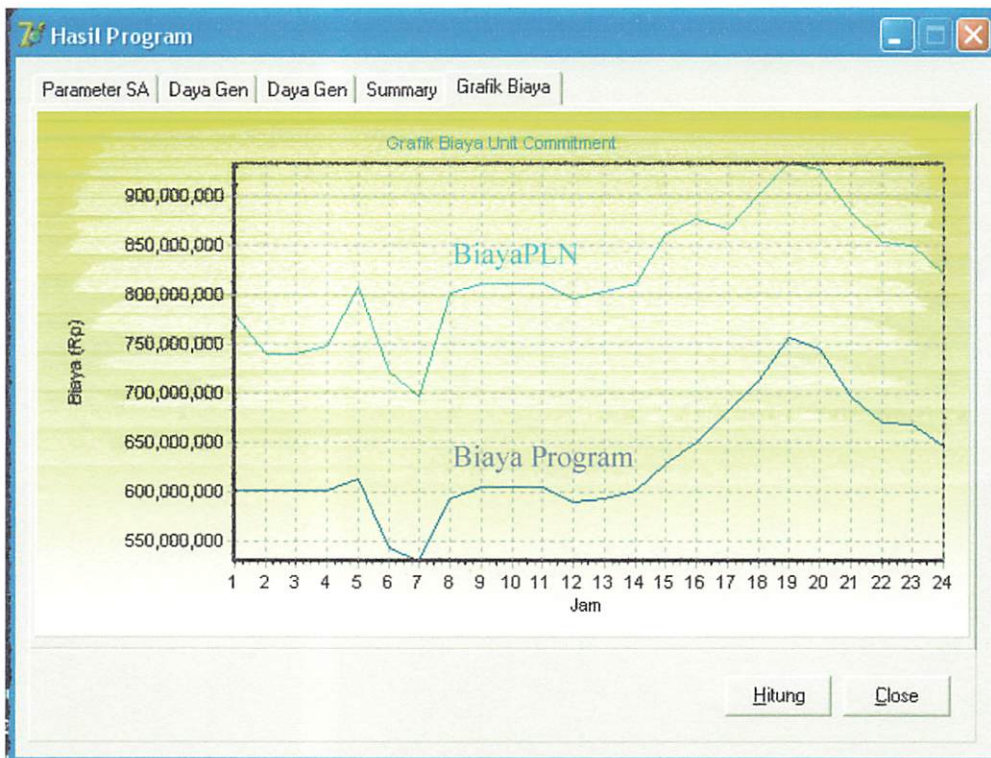
Grafik 4.1.

Kurva Biaya Operasional PT. Pembangkiti Jawa Bali

Metode *Parallel Simulated Annealing*

Dengan *Komputasi Spekulatif*

Kamis, 4 Desember 2003



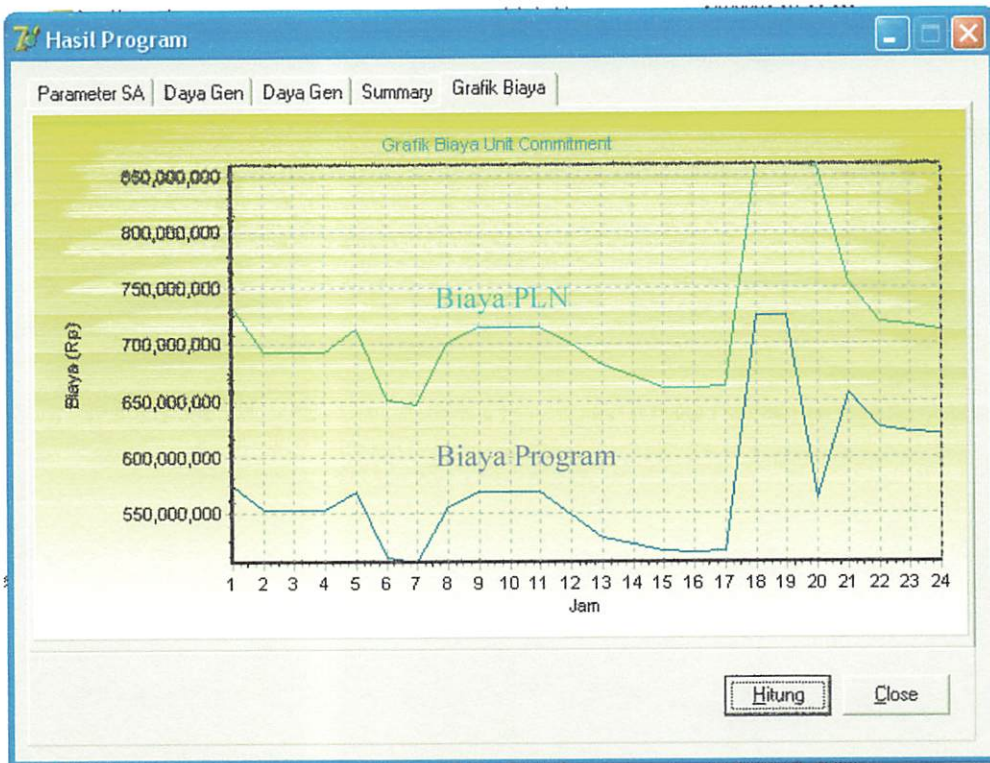
Grafik 4.2.

Kurva Biaya Operasional PT. Pembangkiti Jawa Bali

Metode *Parallel Simulated Annealing*

Dengan *Komputasi Spekulatif*

Sabtu, 6 Desember 2003



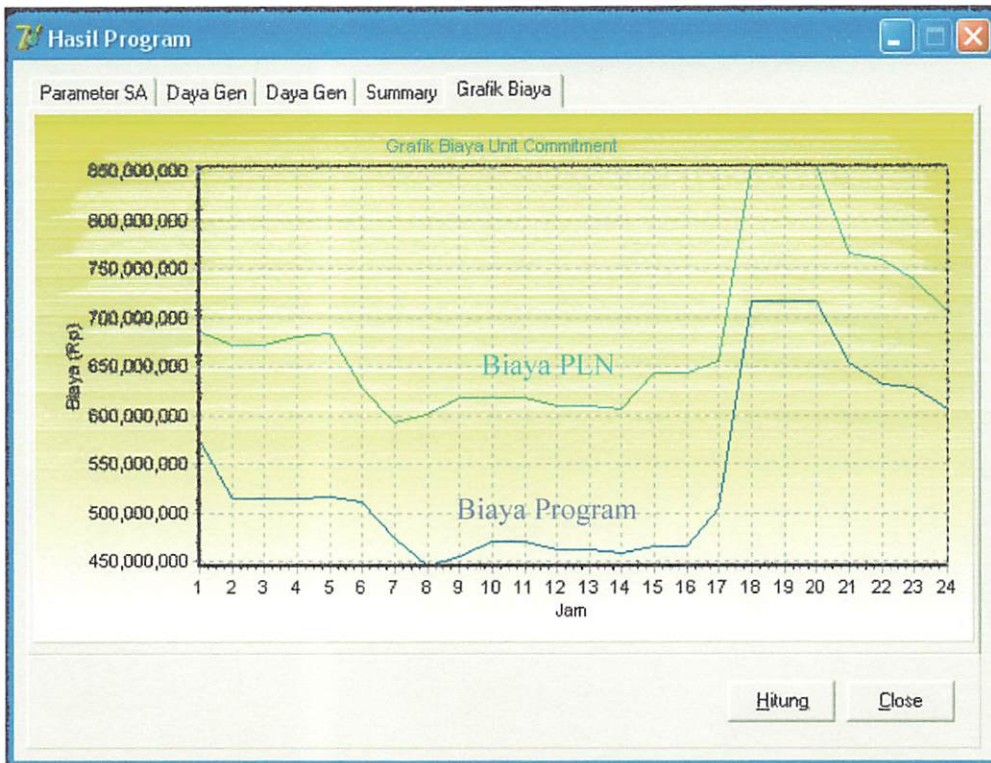
Grafik 4.3.

Kurva Biaya Operasional PT. Pembangkiti Jawa Bali

Metode *Parallel Simulated Annealing*

Dengan *Komputasi Spekulatif*

Minggu, 7 Desember 2003



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Dari hasil perhitungan dan analisa hasil perhitungan terhadap penggunaan metode *PSA* dengan *Komputasi Spekulatif* pada komitmen unit tenaga terhadap beban yang ditanggung PT. Pembangkitan Jawa Bali pada tanggal 4, 6, dan 7 Desember 2003, maka dapat disimpulkan :

- a. Proses metode *PSA* dengan kombinasi *KS* dan *SS* memberikan sebuah analisis penyelesaian yang cukup efektif (optimal) dalam mengoptimalkan pembebanan dan sekaligus menghasilkan penghematan biaya total operasional PT. PJB pada tanggal 4 Desember 2003 selisih biaya total PT. PJB dengan metode *PSA* dengan *KS* adalah sebesar Rp 4,513,391,039 (22.95%). Pada tanggal 6 Desember 2003 selisih biaya total sebesar Rp 3,426,033,791 (19.92%) dan pada tanggal 7 Desember 2003 selisih biaya total sebesar Rp 3,426,233,452 (20.92%)
- b. Metode *PSA* dengan *KS* memungkinkan untuk diterapkan pada PT. PJB karena biaya total yang dikeluarkan atau dihasilkan lebih ekonomis.

#### 5.2. Saran

Perlu adanya koordinasi antara mahasiswa dengan PT. Pembangkit Jawa-Bali agar nantinya didapatkan data yang sesuai dengan keperluan mahasiswa.



## DAFTAR PUSTAKA

- [1] Djiteng Marsudi, Ir., "*Operasi Sistem Tenaga Listrik*", ISTN 1990.
- [2] Allan J. Wood and Bruce F. Wollenberg, "*Power Generation Operation and control*", John Willey, Second Edition, 1996
- [3] U.D. Annakkage, T. Numnonda, N.C. Pahalawaththa. "*Unit Commitment By Parallel Simulated Annealing*", IEE Proc –Gener. Transm. Distrib. Vol 142, No. 6, November 1995
- [4] Sri Kusumadewi. "*Artificial intelligence (teknik dan aplikasinya)*", Graha Ilmu, Yogyakarta, 2003.

# LAMPIRAN

1



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
JURUSAN TEKNIK ELEKTRO

## BERITA ACARA UJIAN SKRIPSI FAKULTAS TEKNOLOGI INDUSTRI

1. Nama : ROSMANSYAH
2. NIM : 00.12.065
3. Jurusan : TEKNIK ELEKTRO
4. Konsentrasi : TEKNIK ENERGI LISTRIK
5. Judul Skripsi : ANALISIS KOMITMEN UNIT PEMBANGKIT  
TERMAL DENGAN METODE PARALLEL  
SIMULATED ANNEALING PADA PT PJB

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

Hari : SENIN  
Tanggal : 28 MARET 2005  
Dengan Nilai : 81.10 (A) *gum*

Ketua Majelis Penguji



Ir. Mochtar Asroni, MSME  
NIP.Y : 101 810 0036

Sekretaris Majelis Penguji



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

Penguji 1



Ir. I Made Wartana, MT  
NIP : 131 991 182

Anggota Penguji

Penguji 2




Ir. Widodo Pudji M, MT  
NIP : 102 870 0171



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

---

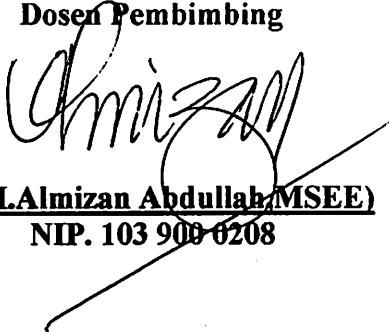
## LEMBAR BIMBINGAN SKRIPSI

1. Nama : ROSMANSYAH
2. NIM : 00.12.065
3. Jurusan : TEKNIK ELEKTRO
4. Konsentrasi : TEKNIK ENERGI LISTRIK
5. Judul Skripsi : ANALISIS KOMITMEN UNIT  
PEMBANGKIT TERMAL DENGAN  
METODE PARALLEL SIMULATED  
ANNEALING PADA PT PJB
6. Tanggal Mengajukan Skripsi : 27 Oktober 2004
7. Tanggal Menyelesaikan Skripsi : 28 MARET 2005
8. Dosen Pembimbing : Ir.H.Almizan Abdullah,MSEE
9. Telah dievaluasi dengan nilai : 85(delapan puluh lima) 

Mengetahui,  
Ketua Jurusan Teknik Elektro

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

Menyetujui,  
Dosen Pembimbing

  
(Ir. H. Almizan Abdullah, MSEE)  
NIP. 103 900 0208



FORMULIR BIMBINGAN SKRIPSI

Nama : ROSMANSYAH  
Nim : 00.12.006  
Masa Bimbingan : 7 Februari 2005 s/d 7 Agustus 2005  
Judul Skripsi : ANALISIS KOMITMEN UNIT PEMBANGKIT  
TERMAL DENGAN METODE PARALLEL  
SIMULATED ANNEALING PADA PT PJB

No.	Tanggal	Uraian	Paraf Pembimbing
1.	07-03-05	Sempurnakan Flow Chart & menunjukkan adanya proses paralel	
2.	14-03-05	1. Sprim; Recroe moment Journal 10% tapi PLN di sana ke unit pembangkit terbesar dlm lister (100 MW)	
3.		2. Jelaskan kenapa pada jam-jam tertentu ke. Supaya biaya PLN bisa lebih murah.	
4.	26-03-05	Bab IV. Kenapa Biaya PJB lebih murah d/p IWA	
5.	31-03-05	Bab IV Apa yang dimaksud dengan temperatur sudah dengan	
6.			
7.			
8.			
9.			
10.			

Malang, 2005

Dosen Pembimbing

Ir. H. Almizan Abdullah, MSEE

NIP : 1039000208

Form.S-4b



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

## PERSETUJUAN PERBAIKAN SKRIPSI

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

Hari : Senin  
Tanggal : 28 Maret 2005

Telah dilakukan perbaikan skripsi oleh :

1. Nama : ROSMANSYAH
2. NIM : 00.12.065
3. Jurusan : Teknik Elektro
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : ANALISIS KOMITMEN UNIT PEMBANGKIT TERMAL DENGAN METODE PARALLEL SIMULATED ANNEALING PADA PT PJB

Perbaikan meliputi :

No	Materi Perbaikan	Ket
1.	Bagaimana menentukan jumlah iterasi 100, padahal pada iterasi 50 biaya yang dibutuhkan lebih murah	

Penguji Pertama

Ir. I Made Wartana, MT  
NIP : 131 991 182

Anggota Penguji

Penguji Kedua

Ir. Widodo Pudji M, MT  
NIP : 102 870 0171

Dosen Pembimbing

Ir. H. Almizan Abdullah, MSEE  
NIP. 103 900 0208

PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai permohonan dari mahasiswa/i :

Nama : *Ramansyah*

Nim : *0012063*

Semester : *X*

Jurusan : Teknik Elektro S-1

Konsentrasi : Energi Listrik

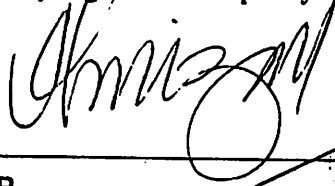
Dengan ini Menyatakan bersedia / tidak bersedia \*) Membimbing Skripsi dari mahasiswa tersebut, dengan judul :

*Analisis Efektifitas dan Pembangkit Termal dengan Metode Simulasi Anemometer pada PT PSB*

Demikian surat Pernyataan ini kami buat agar dapat dipergunakan seperlunya.

Malang,

Kami yang Membuat pernyataan,



NIP.

Catatan :

Setelah disetujui agar formulir ini  
Diserahkan mahasiswa/i yang bersangkutan  
Kepada Jurusan untuk diproses lebih lanjut.

\*) Coret yang tidak perlu

INSTITUT TEKNOLOGI NASIONAL  
Jl. Bendungan Sigurgura No. 2  
**MALANG**

Lampiran : 1 (satu) berkas  
Pembimbing Skripsi

Kepada : Yth. Bapak/Ibu Ir. H. Alimam A. MSEE  
Dosen Institut Teknologi Nasional  
MALANG

Yang bertanda tangan di bawah ini :

Nama : Rosmansyah  
Nim : 00.12.065  
Jurusan : Teknik Elektro S-1  
Konsentrasi : Energi Listrik

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing Utama / Pendamping \*), untuk penyusunan Skripsi dengan judul (proposal terlampir) :

Analisis Unit Demitmen pembangkit tenaga dengan metode simulasi anealing pada PT PIB

Adapun tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.

Demikian permohonan kami dan atas kesediaan Bapak/Ibu kami ucapkan terima kasih.

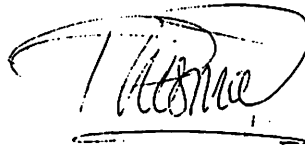
Malang,

Mengetahui  
Ketua Jurusan Teknik Elektro



Ir. I Made Wartana, MT  
NIP. 131 991 182

Hormat kami,



Rosmansyah.

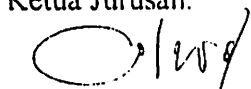

\*) coret yang tidak perlu





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

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika \*)

1	Nama Mahasiswa : <u>ROS MANSYAH</u>		Nim : <u>0012065</u>	
2	Waktu pengajuan	Tanggal :	Bulan :	Tahun :
		<u>27</u>	<u>10</u>	<u>2004</u>
Spesifikasi judul ( berilah tanda silang )				
3	<input type="checkbox"/> a. Sistem Tenaga Elektrik <input checked="" type="checkbox"/> b. Energi & Konversi Energi <input type="checkbox"/> c. Tegangan Tinggi & Pengukuran <input type="checkbox"/> d. Sistem Kendali Industri		<input type="checkbox"/> e. Elektronika & Komponen <input type="checkbox"/> f. Elektronika Digital & Komputer <input type="checkbox"/> g. Elektronika Komunikasi <input type="checkbox"/> h. lainnya .....	
4	Konsultasikan judul sesuai materi bidang ilmu kepada Dosen *) : <u>Ir. H. Alimuzan Abdullah</u>		Ketua Jurusan.  Ir. I Made Wartana, MT Nip: 131 991 182	
5	Judul yang diajukan mahasiswa :	<u>Analisis Unit Commitment pembangkit termal PT. PLN (Persero) PJB Dengan Metode Paralel Simulated Annealing.</u>		
6	Perubahan Judul yang disetujui Dosen sesuai materi bidang ilmu	<u>Analisis <del>Unit</del> Komitmen Unit pembangkit termal dengan Metode Paralel Simulated Annealing pada PT PTB</u>		
7	Catatan :			
	Persetujuan Judul Skripsi yang dikonsultasikan kepada Dosen materi bidang ilmu		Disetujui, <u>27 - 10 - 2004</u> . Dosen  <u>ALMUZAN ABDULLAH</u>	

**Perhatian :**

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

# LAMPIRAN

**2**



**DATA PENAWARAN  
PT PLN PEMBANGKITAN JAWA BALI  
AGUSTUS 2002**

No.	NAMA PEMBANGKIT	KAPASITAS			LAMA WAKTU (JAM)						YA START UP (JUTA)			EFISIEN BIAYA BAHAN BAKA		
		Daya Terpasang	MIN (MW)	MAX (MW)	MIN UP TIME	MIN DOWN TIME	COLD START UP	HOT START UP	COLD START UP	HOT START UP	a0	a1	a2			
1	UP. PAITON PLTU #1/2 (COAL)	2 x 400	225	370	72	48	17	4	682.98	149.68	3244978	111712.2	10.2971			
2	UP. GRESIK	9 x 112	53	102	36	10	1	0	7.82	0	5467532	217963.5	34.155			
	GT 1-9 OC (GAS)		115	143	36	10	3	1	57.68	31.46	10936203	72527	368.874			
	CC - 1.1.1 (GAS)		164	314	36	10	3	2	65.5	39.28	11795771	152515.7	6.831			
	CC - 2.2.1 (GAS)		250	480	36	10	3	2	73.32	47.1	17177460	145165.6	4.554			
	CC - 3.3.1 (GAS)	3 x 526	43	85	48	10	9	1	143.74	40.59	1327127	217378.4	132.066			
	PLTU # 1/2 (GAS)	100	90	175	48	10	9	2	229.5	92.52	5017370	169242.6	193.545			
	PLTU # 3/4 (GAS)	200	5	16	3	1	1	0	6.13	0	352707.3	350680.8	903.969			
	PLTG GRESIK 1-3 (GAS)	3 x 20	5	16	3	1	1	0	6.33	0	687181.9	683241	1762.389			
	PLTG GILITIMUR 1-2 (HSD)	2 x 20														
3	UP. MUARA KARANG	3 x 107	50	95	36	10	1	0	7.35	0	5730795	202053	108.045			
	GT 1/2/3 - OC	153	110	150	36	10	3	1	54.22	29.67	11560815	53685.14	460.845			
	CC - 1.1.1 (GAS)	317	200	300	36	10	3	2	61.57	36.92	16010064	127208.7	35.28			
	CC - 2.2.1 (GAS)	508	300	465	36	10	3	2	68.92	44.27	31017735	87825.15	57.33			
	CC - 3.3.1 (GAS)	2 x 140	72	138	36	10	0	0	0	0	14706521	433337.8	49.4605			
	MTW GT 1/2 - OC (HSD)	200	162	202	36	10	3	1	118.08	64.4	672630	144191.7	519.1757			
	MTW CC - 1.1.1 (HSD)	420	210	403	36	10	3	2	134.1	80.42	30123040	303208.8	11.64715			
	MTW CC - 2.2.1 (HSD)	640	315	605	36	10	3	2	160.1	96.42	43043399	288610	7.6584			
	MTW CC - 3.3.1 (HSD)	3 x 100	44	85	48	10	6	1	122.58	31.08	2417821	473895.4	120.7794			
	PLTU # 1/2/3 (MFO)	2 x 200	90	165	48	10	11	2	215.34	89.29	2949188	205217.1	83.79			
	PLTU # 4/5 (Gas)															

Catatan :

Harga Batubara  
Harga MFO  
Harga HSD  
Harga Gas UP. Gresik  
Harga Gas UP. M.Karang  
Nilai Tukar

253 Rp/Kg  
1595.5 Rp/liter  
1595.5 Rp/liter  
2.53 US\$/MMBTU  
2.45 US\$/MMBTU  
9000 Rp/US\$

SUB SISTEM REGION\_1

RENCANA : HARI/TANGGAL : KAMIS, 4 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

PLTU	MKRNGI0C	MKRNGI0C1	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	MKRNGI0C	Jan		
																						00.30	01.00	
PLTU MKRNG	#4	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150
	#5	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165
	#1	70	70	60	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55
	#2	70	70	60	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55
	#3	70	70	60	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55	55
PLTU MKRNGI0C	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400
	425	425	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400	400

PEMBELIAN DARI LUAR PLN

PLTU	Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel		Kabalau Steel	
	4700	4700	4845	4635	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830	4830
4214	4214	4185	4054	4062	4021	4040	4058	4152	4152	3984	3935	3937	4188	4546	4839	5033	5107	5153	5153	5164	5175	4986	4908	4908	4908	4908
486	486	460	581	568	609	590	588	609	609	588	588	609	588	588	609	588	588	609	609	588	588	609	588	588	609	588
37	37	55	28	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54	54
139	140	186	205	210	210	210	184	79	79	162	317	633	570	219	89	39	39	24	24	24	24	24	24	24	24	24

12/2003 14:12/2003/03/04

SUB SISTEM REGION\_1

RENCANA : HARITANGGAL : KAMIS, 4 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

		Jan	13.00	13.30	14.00	14.30	15.00	15.30	16.00	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30	21.00	21.30	22.00	22.30	23.00	23.30	24.00	RAMA-2	
PLTGU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG10CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU	MKRNG1CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG2CC	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	425	418	
	MKRNG3CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG4CC	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	
	MKRNG5CC	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	163	
MKRNG	#1	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	78	
	#2	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	78	
	#3	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	85	78	
PLTGU	MTWAR110C	0	0	0	0	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	72	0	
	MTWAR10CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	36	
	MTWAR20CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR130C	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	135	0	
	MTWAR110C	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	0	
	MTWAR11CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR12CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR13CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTWAR GT 2.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
MTWAR GT 2.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PEMBELIAN DARI LUAR PLN																											
PLTP	GSLAK #4	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	
	#5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
CIKARANG	#6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	#6	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	125	
PLTU	Kralatau Steel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	Pembangkitan Area-1	4833	4833	4833	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4905	4842	
(*)	Bahan Area-1	5100	5187	5262	5278	5271	5234	5242	5221	5316	5314	5385	5422	5411	5368	5302	5302	5302	5302	5302	5302	5302	5302	5302	5302	4795	
	Selish (*) - (**)	-367	-354	-429	-341	-372	-386	-349	-354	-283	-213	-141	-172	-199	-188	-145	-99	-99	-99	-99	-99	-99	-99	-99	-99	487	
Codangan Sekeloa		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	
		24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	27	
Codangan P U L O R		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	27	
		24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	27	

SUB SISTEM REGION\_2

RENCANA : HARI/TANGGAL : KAMIS, 4 DESEMBER 2003

PT. INDONESIA POWER

Area 2.	Jm	00:30 01:00 01:30 02:00 02:30 03:00 03:30 04:00 04:30 05:00 05:30 06:00 06:30 07:00 07:30 08:00 08:30 09:00 09:30 10:00 10:30 11:00 11:30 12:00 12:30																									
		00:30	01:00	01:30	02:00	02:30	03:00	03:30	04:00	04:30	05:00	05:30	06:00	06:30	07:00	07:30	08:00	08:30	09:00	09:30	10:00	10:30	11:00	11:30	12:00	12:30	
PLTA Area 2.	SGLNG	#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#2	127	127	119	103	108	108	108	100	109	134	109	105	80	80	82	91	108	121	124	124	124	124	111	90	81
		#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTP KMLING	#1	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
		#2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		#3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		#4	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTP DRJAT	#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG SRAGI	#1	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
		#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWABALI																											
PLTA CRAFA	#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
WYNDU DRJAT	#1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	
		#2	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
		#3	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	
		#4	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	
PEMBELIAN DARI LUAR PLN																											
Pembangkitan Area-2	Beban Area-2	574	574	566	550	555	555	555	547	536	581	536	552	527	527	609	659	742	766	816	816	816	816	748	686	670	
		1689	1706	1678	1707	1699	1718	1693	1693	1737	1730	1746	1674	1572	1449	1372	1505	1536	1624	1645	1655	1661	1661	1640	1566	1545	
		-1115	-1132	-1112	-1157	-1144	-1163	-1138	-1190	-1174	-1165	-1118	-1020	-922	-845	-895	-897	-882	-877	-839	-845	-845	-845	-892	-880	-875	
		13	13	12	10	11	11	11	10	11	11	13	11	11	8	8	18	21	30	32	25	25	25	25	30	24	22
Cadangan Selekta	Cadangan P u l e r	48	48	56	72	67	67	67	75	66	41	66	70	95	95	188	263	180	154	106	106	106	106	174	236	252	

SUB SISTEM REGION 2

RENCANA : HARTANGGAL : KAMIS, 4 DESEMBER 2003

PT. INDONESIA POWER

		Jum																								Rate-2
		13.00	13.30	14.00	14.30	15.00	15.30	16.00	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30	21.00	21.30	22.00	22.30	23.00	23.30	24.00		
PLTA	Area 2.	24	24	24	24	24	24	24	24	24	24	24	29	29	29	29	29	29	29	29	24	24	24	24	25	
PLTA	SGJUNG	#1	100	105	125	99	99	99	99	99	119	111	115	108	112	127	121	103	103	100	105	105	100	102	76	
		#2	101	105	125	99	99	99	99	99	120	110	115	108	112	127	121	103	103	100	100	105	100	100	102	97
		#3	0	0	0	0	0	0	0	0	0	0	80	107	112	128	121	103	103	100	106	100	100	99	103	30
		#4	0	0	0	0	0	0	0	0	0	0	80	100	104	119	113	103	102	83	0	0	0	0	0	17
PLTP	KMLJUNG	#1	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
		#2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		#3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		#4	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTP	DRJAT	#1	0	0	0	0	0	0	0	0	0	10	18	18	18	18	18	18	18	18	18	18	18	18	4	
		#2	0	0	0	0	0	0	0	0	0	0	18	18	18	18	18	18	18	18	18	18	18	18	4	
		#3	0	0	0	0	0	0	0	0	0	0	18	18	18	18	18	18	18	18	18	18	18	18	18	
		#4	0	0	0	0	0	0	0	0	0	10	18	18	18	18	18	18	18	18	18	10	0	0	0	4
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI																										
PLTA	GRAHA	#1	0	0	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	32	
		#2	100	100	100	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	53	
		#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		#4	0	0	0	0	0	0	0	0	30	60	60	60	80	80	80	80	80	80	80	80	80	80	13	
		#5	0	0	0	0	0	0	0	0	0	30	60	60	80	80	80	80	80	80	80	80	80	80	11	
		#6	0	0	0	0	0	0	0	0	0	0	30	60	60	80	80	80	80	80	80	80	80	80	8	
		#7	0	0	0	0	0	0	0	0	0	0	30	60	60	80	80	80	80	80	80	80	80	80	7	
		#8	0	0	0	0	0	0	0	0	0	0	30	60	60	80	80	80	80	80	80	80	80	80	7	
PEMBELIAN DARI LUAR PLN																										
WYNDU	#1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105		
		90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90		
		35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	37		
		35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	37		
DRJAT	#2	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105		
		90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90		
		35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	37		
		35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	37		
Pembangkitan Area-2																										
(*)	Beban Area-2	748	757	877	805	805	805	805	805	846	878	1153	1306	1443	1504	1479	1395	1394	1151	959	837	807	725	645	806	
		1605	1600	1626	1631	1647	1661	1640	1676	1707	1857	1857	2168	2214	2256	2245	2232	2210	2167	2057	1971	1947	1898	1827	1619	1762
		-857	-843	-749	-826	-842	-856	-835	-830	-829	-714	-652	-652	-771	-752	-766	-837	-816	-1016	-1098	-1134	-1140	-1173	-1182	-1167	-856
Cadangan Seketika		30	31	43	38	38	38	38	38	40	42	67	82	94	100	98	89	89	66	47	38	38	28	20	34	
Cadangan P u l a r		174	165	170	242	242	242	242	242	201	330	655	752	635	574	599	683	684	537	304	251	240	197	152	234	

RENCANA : HARI/TANGGAL : KAMIS , 4 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

Jam 00.30 01.00 01.30 02.00 02.30 03.00 03.30 04.00 04.30 05.00 05.30 06.00 06.30 07.00 07.30 08.00 08.30 09.00 09.30 10.00 10.30 11.00 11.30 12.00 12.30

PLTU	Area 4.	Jam																									
		00.30	01.00	01.30	02.00	02.30	03.00	03.30	04.00	04.30	05.00	05.30	06.00	06.30	07.00	07.30	08.00	08.30	09.00	09.30	10.00	10.30	11.00	11.30	12.00	12.30	
PLTA	SUTAMI	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
	BRANTAS	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
PLTU	PTON #1	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	
	PTON #2	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	
PLTG 2	GRSK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSK220C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSK20C2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSK10C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK20C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK310C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK310C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG 3	GRSK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK10C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSK20C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU	GRSK #3	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
	GRSK #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
	GRSK #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	GRSK #2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTG	GLTMR #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GLTMR #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRESIK #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG	GRESIK #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRESIK #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
(*) Pembangkitan Area-4	2054	2058	2054	2049	2044	2062	2065	2045	2108	2059	1959	1802	1877	1621	1788	1974	1802	2067	2071	2082	2081	2054	1992	1978	1987	1987	
	1223	1219	1223	1128	1133	1115	1112	1152	1164	1238	1058	965	1078	1134	1142	1281	1372	1463	1459	1448	1448	1476	1538	1462	1433	1433	
	57	57	57	52	52	52	52	53	56	58	87	94	93	93	50	87	55	37	37	37	37	37	37	37	85	82	
	608	609	609	609	609	609	609	688	814	788	1049	884	994	994	819	904	899	599	599	589	589	589	589	688	688	739	





RENCANA : HARI/TANGGAL : SABTU, 6 DESEMBER 2003

SUB SISTEM REGION\_1

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

		Jan																									
		00.30	01.00	01.30	02.00	02.30	03.00	03.30	04.00	04.30	05.00	05.30	06.00	06.30	07.00	07.30	08.00	08.30	09.00	09.30	10.00	10.30	11.00	11.30	12.00	12.30	
PLTGU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG12CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG22CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKRNG32CC	425	410	410	400	400	400	400	400	400	400	400	400	350	350	350	360	400	425	425	425	425	425	425	425	425	400
	PLTGU	MKRNG #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90
	MKRNG #5	165	150	150	150	150	150	150	150	165	165	150	150	150	150	150	155	165	165	165	165	165	165	165	165	150	
	MKRNG #1	70	65	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	MKRNG #2	70	65	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	MKRNG #3	70	65	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTGU	MTVAR110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR130C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR12CC	350	350	350	325	325	325	325	325	350	350	350	300	300	300	350	350	350	350	350	350	350	350	350	350	350	
	MTVAR13CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR13CC1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTVAR	GT 2.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GT 2.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PEMBELUAN DARI LUAR PLN																											
PLTP	GS/LAK #4	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	
	#5.	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	#6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	CIKARANG	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	
	PLTU	Kekatsu Steel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		Pembangkitan Area-1	4655	4553	4506	4453	4453	4453	4453	4453	4550	4550	4443	4343	4125	4125	4328	4493	4591	4591	4591	4591	4591	4591	4591	4573	4533
	Beban Area-1	4482	4436	4379	4323	4224	4180	4143	4105	4309	4228	4093	4093	4151	4207	4445	4718	4819	4780	4859	4888	4830	4885	4861	4708	4541	
	Selish (*) - (**)	173	117	129	130	229	273	310	348	241	322	350	250	-26	-82	-117	-225	-228	-189	-268	-297	-239	-294	-288	-175	-88	
	Cadangan Selekta	54	106	104	121	121	121	121	121	70	70	130	125	172	172	126	123	50	50	50	50	50	50	50	70	75	
	Cadangan P u t a r	169	272	318	372	372	372	372	372	275	275	382	482	675	675	472	342	244	244	244	244	244	244	244	244	284	

RENCANA :

# HARIANGLAL : SABTU, 6 DESEMBER 2003

SUB SISTEM REGION 1

PT PLN PEMBANGKITAN TERAGA LISTRIK JAWA-BALI

	Jam	13.00	13.30	14.00	14.30	15.00	15.30	15.30	16.00	16.30	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30	21.00	21.30	22.00	22.30	23.00	23.30	24.00	Rata-2							
PLTU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
	MKRNG10CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
	MKRNG20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	MKRNG20CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	MKRNG30CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
MKRNG	MKRNG3CC	360	370	360	360	360	350	350	350	350	350	350	400	425	425	425	425	425	425	425	400	400	400	400	400	400	400	400	395					
	#4	90	90	90	90	90	90	90	90	90	90	90	135	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	107				
	#5	150	150	150	150	150	150	150	150	150	150	150	150	165	165	165	165	165	165	165	160	150	150	150	150	150	150	150	156					
	#1	50	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	60	50	50	50	50	50	50	50	55					
	#2	50	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	60	50	50	50	50	50	50	50	55					
PLTGU	MTWAR110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	MTWAR10CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	MTWAR20CI	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	MTWAR120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	MTWAR130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	MTWAR110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR130C	350	360	325	325	292	292	300	300	300	300	300	365	390	390	390	390	390	390	390	350	350	350	350	350	350	350	350	403					
	MTWAR13CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
MTWAR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
MTWAR GT 2.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
MTWAR GT 2.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
PEMBELIAN DARI LUAR PLN																																		
PLTP	GSGLAK #4	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182				
	#5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	#6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	CIKARANG	110	110	110	110	110	110	110	110	110	110	110	110	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	100	100	114			
	Mkakuu Steel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	PLTU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
MKRNG10CI		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
MKRNG20C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
MKRNG20CI		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
MKRNG30C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
MKRNG30CI		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
(*) Pembangunan Area-1																																		
(**) Beban Area-1																																		
Selisih (*) - (**)																																		
Cadangan Sekeloa	73	74	72	72	69	69	70	70	70	76	87	87	44	44	44	44	44	44	44	44	53	-72	-53	-76	-95	-90	-166	-85	14	135	177	255	296	378
Cadangan P u l a r	404	384	429	429	599	599	591	591	591	544	329	329	114	114	114	114	114	114	114	189	309	319	319	319	319	319	319	382	382	382	382	382	382	382
4582																																		
4609																																		
-27																																		
86																																		
346																																		

SUB SISTEM REGION\_2

RENCANA : HARI/TANGGAL : SABTU, 6 DESEMBER 2003

PT. INDONESIA POWER

		Jum																																	
		00.30	01.00	01.30	02.00	02.30	03.00	03.30	04.00	04.30	05.00	05.30	06.00	06.30	07.00	07.30	08.00	08.30	09.00	09.30	10.00	10.30	11.00	11.30	12.00	12.30									
PLTA	Area 2.	#1	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24								
			#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
				#3	113	104	102	100	100	100	100	100	100	100	100	100	100	100	101	102	104	115	115	116	116	116	116	125	105	99					
					#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
PLTP	KUNJING	#1	113	103	103	100	100	100	100	100	100	100	100	100	100	100	100	101	103	103	114	114	116	116	116	125	105	99							
			#2	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25						
				#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
					#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
PLTP	DRJAT	#1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50							
			#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0						
				#3	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18					
					#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
PLTG	SRAGI	#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
			#2	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18						
				#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
					#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI																																			
PLTA	CRATA	#1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0							
			#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0					
				#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
					#4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
						#5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
							#6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
								#7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
									#8	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100
PEMBELIAN DARI LUAR PLN																																			
WYNDU	#1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105							
		90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90						
DRJAT	#2	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45							
		45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45						
PEMUN. OROKTA JALIHUR																																			
(*) Pembangkitan Area-2	Selisih (*) - (**)	Cadangan Sektika	733	714	712	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707	707							
			(**) Beban Area-2	1717	1695	1673	1651	1648	1707	1758	1769	1765	1719	1627	1590	1532	1339	1395	1473	1557	1576	1519	1520	1520	1520	1520	1520	1520	1520	1520					
				Cadangan Sektika	-883	-981	-851	-944	-941	-1000	-1051	-1082	-1033	-987	-929	-907	-855	-672	-687	-693	-773	-770	-713	-711	-711	-711	-711	-711	-711	-711	-711				
					Cadangan P u l a r	33	31	31	30	30	30	30	30	33	33	29	28	26	26	30	37	38	38	40	40	40	40	40	40	40	40	40			
		149	168	170	175	175	175	175	175	175	184	199	215	215	215	174	227	223	201	201	199	198	198	198	198	180	220	282							

SUB SISTEM REGION\_2

HARI/TANGGAL : SABTU, 6 DESEMBER 2003

RENCANA :

PT. INDONESIA POWER

		Jam																Rate-2							
		13.00	13.30	14.00	14.30	15.00	15.30	16.00	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30		21.00	21.30	22.00	22.30	23.00	23.30	24.00
PLTA	Area 2	24	24	24	24	24	24	24	24	24	24	24	24	33	33	33	33	33	33	33	24	24	24	24	24
PLTA	SGLNG #1	0	0	0	0	0	0	0	0	0	80	115	121	126	126	120	107	102	102	101	102	100	103	103	31
PLTA	SGLNG #2	102	117	87	87	80	80	81	80	86	101	115	121	126	126	120	108	102	102	100	0	0	0	0	92
PLTA	SGLNG #3	0	0	0	0	0	0	0	0	0	80	114	121	126	120	107	102	102	101	102	100	102	102	31	
PLTA	SGLNG #4	101	116	87	87	80	80	81	80	86	102	114	121	126	120	107	103	103	100	100	0	0	0	93	
PLTP	KUJUNG #1	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25
PLTP	KUJUNG #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTP	KUJUNG #3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
PLTP	DRJAT #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
PLTG	SRANGI #1	0	0	0	0	0	0	0	0	0	18	18	18	18	18	18	18	18	18	18	18	18	18	18	4
PLTG	SRANGI #2	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18
PLTG	SRANGI #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTG	SRANGI #4	0	0	0	0	0	0	0	0	0	18	18	18	18	18	18	18	18	18	18	18	18	18	18	4
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI																									
PLTA	CRATA #1	0	0	0	0	0	0	0	0	0	30	70	80	80	80	80	80	80	80	80	80	80	80	80	22
PLTA	CRATA #2	0	0	0	0	0	0	0	0	0	30	60	80	80	80	80	80	80	80	80	80	80	80	80	16
PLTA	CRATA #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	CRATA #4	0	0	0	0	0	0	0	0	0	30	60	80	80	80	80	80	70	0	0	0	0	0	0	11
PLTA	CRATA #5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	CRATA #6	0	0	0	0	0	0	0	0	0	0	30	60	80	80	80	70	60	60	60	60	60	60	60	13
PLTA	CRATA #7	70	70	70	70	70	70	70	70	70	70	80	80	80	80	80	80	80	80	80	80	80	80	80	41
PLTA	CRATA #8	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	76
PEMBELIAN DARI LUAR PLN																									
PLTA	WYRIDU #1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105
PLTA	DRJAT #2	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90
PLTA	DRJAT #2	45	45	45	45	45	45	45	45	45	50	50	50	50	50	50	50	50	50	50	50	50	50	50	46
Penum.Otorita Jeliuhur		760	790	731	731	717	717	717	717	717	747	1051	1266	1381	1441	1417	1336	1276	1069	971	836	687	692	692	844
(*) Pembangkitan Area-2		1441	1483	1410	1494	1402	1412	1444	1405	1391	1590	1961	2101	2155	2108	2083	2000	2000	1858	1820	1806	1759	1693	1693	1645
(**) Beban Area-2		-681	-693	-679	-763	-685	-695	-725	-688	-844	-529	-675	-720	-714	-691	-747	-724	-792	-799	-849	-970	-1072	-1001	-1001	-800
Selisih (*) - (**)		35	38	32	32	31	31	31	31	32	60	64	92	98	96	88	82	82	62	61	51	42	28	29	43
Cadangan Seketika		247	217	276	276	290	290	288	290	278	722	612	528	480	490	571	631	548	483	386	319	195	190	189	282
Cadangan Pulat																									

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

Jam

00.30 01.00 01.30 02.00 02.30 03.00 03.30 04.00 04.30 05.00 05.30 06.00 06.30 07.00 07.30 08.00 08.30 09.00 09.30 10.00 10.30 11.00 11.30 12.00 12.30

PLTA	Area 4.	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
PLTA	SUTAMI	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
PLTA	BRANTAS.	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	
PLTU	PITON #1	370	370	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	
PLTU	PITON #2	370	370	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	350	
PLTGU 2	GRSIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTU	GRSIK #3	165	165	165	150	150	150	150	150	150	150	150	150	150	150	150	165	165	165	165	165	165	150	150	
PLTU	GRSIK #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
PLTU	GRSIK #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTU	GRSIK #2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTG	GLTMR #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG	GLTMR #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG	GLTMR #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG	GLTMR #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTG	GRSIK #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTGU 3	GRSIK10C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK20C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK310C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	Pembangkitan Area-4		3517	3517	3477	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426	3426
Beban Area-4		2086	2049	2022	1999	2093	2096	2077	2073	2075	2182	1863	1757	1717	1665	1816	1892	1962	2082	2069	2030	2092	1996	2038	
Selishih (*)		1451	1468	1455	1427	1333	1330	1249	1333	1422	1315	1319	1217	1127	1179	1251	1380	1460	1440	1453	1492	1430	1528	1484	
Cadangan Sekeloa		65	65	100	108	108	108	108	108	72	72	87	93	92	92	98	59	61	61	61	61	61	61	61	
Cadangan P u i a r		604	604	644	695	695	695	695	695	624	624	639	1147	952	952	1054	854	704	604	604	604	604	604	644	

SUB SISTEM REGION\_1

RENCANA : HARI/TANGGAL : MINGGU , 7 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

Jam		00.30	01.00	01.30	02.00	02.30	03.00	03.30	04.00	04.30	05.00	05.30	06.00	06.30	07.00	07.30	08.00	08.30	09.00	09.30	10.00	10.30	11.00	11.30	12.00	12.30	
PLTGU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG1CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG2CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG3CC	400	396	350	350	350	350	344	319	346	346	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	
PLTU	MKRNG #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
	#5	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	
	MKRNG #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	#2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	#3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTGU	MTWAR10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR11CC	0	0	0	0	0	0	0	0	0	0	0	0	0	200	180	180	180	200	200	200	200	200	200	200	200	
	MTWAR12CC	350	350	350	350	350	350	350	390	390	390	320	250	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR13CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR GT 2.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR GT 2.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PEMBELIAN DARI LUAR PLN																											
PLTP	GSLAK #4	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	
	#5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	#6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU	CIKARANG	100	100	100	100	100	100	100	100	100	100	100	100	50	50	50	50	50	50	50	50	50	50	50	50	50	
	Krakatau Steel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
(*) Pembangkitan Area-1		4720	4639	4593	4555	4520	4514	4529	4556	4556	4207	3965	3845	3825	3825	3875	3945	3945	3945	3945	3945	3945	3945	3945	3937	3937	
(**) Beban Area-1		4260	4248	4152	4070	4096	4069	4138	4018	4071	4071	3946	3855	3826	3744	3771	3885	4026	4005	3945	3978	3997	4066	4052	4020	4075	
Selisih (*) - (**)		460	391	441	485	424	451	376	511	485	485	281	110	19	81	54	-10	-81	-60	0	-33	-52	-121	-115	-83	-188	
Cadangan Seketika		89	140	138	193	191	191	190	172	173	173	146	163	150	159	159	161	155	155	155	155	155	155	155	155	152	
Cadangan P u t s a r		299	381	428	465	500	500	506	491	464	464	651	893	775	795	795	745	675	675	675	675	675	675	675	675	725	

SUB SISTEM REGION\_1

RENCANA: HARI/TANGGAL : MINGGU , 7 DESEMBER 2003  
 PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

		Jam																								Rate-2
		13.00	13.30	14.00	14.30	15.00	15.30	16.00	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30	21.00	21.30	22.00	22.30	23.00	23.30	24.00		
PLTGU	MKRNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG10CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU	MKRNG2CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNG3CC	300	300	300	300	300	300	300	300	350	400	425	425	425	425	425	425	425	425	425	425	425	425	425	400	
	MKRNG #4	90	90	90	90	90	90	90	90	135	140	150	150	150	150	150	150	150	150	150	150	150	150	150	135	
	MKRNG #5	150	150	150	150	150	150	150	150	150	150	165	165	165	165	165	165	165	165	165	165	165	165	165	150	
	MKRNG #1	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85	50	
	MKRNG #2	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85	50	
PLTGU	MKRNG #3	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85	85	50	
	MTWAR10CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR130C	0	0	0	72	72	72	72	72	72	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR110CC	200	200	200	170	170	170	170	170	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	395	
	MTWAR110C	0	0	0	0	0	0	0	0	250	275	325	375	375	375	375	375	375	375	375	375	375	375	375	350	
	MTWAR12CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTWAR13CC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PEMBELIAN DARI LUAR PLN																										
PLTP	GSLAK #4	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	
	GSLAK #5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GSLAK #6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	CIKARANG	50	50	50	50	50	50	50	50	50	50	150	150	150	150	150	150	150	150	150	150	150	150	150	100	
	Makassar Steel	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	PLTGU Pembangunan Area-1	3887	3937	3937	3979	3928	3928	3979	4084	4294	4567	4993	4993	4993	4993	4993	4993	4993	4993	4993	4993	4993	4993	4993	4993	
(*) Beban Area-1	4051	4033	4159	4160	4108	4083	4070	4226	4395	4481	5159	5038	5025	5030	5020	5020	5020	5020	5020	5020	5020	5020	5020	5020		
Selish (*) - (**)	-164	-96	-222	-181	-179	-134	-91	-142	-101	88	-166	-45	-32	-37	-27	-27	-27	-27	-27	-27	-27	-27	-27	-27		
Cedangan Sektora	152	155	155	167	164	164	167	142	160	168	43	43	43	43	43	43	43	43	43	43	43	43	43	43		
Cedangan P u l a r	725	675	675	771	821	821	771	721	738	565	149	149	149	149	149	149	149	149	149	149	149	149	149	149		



SUB SISTEM REGION\_2

RENCANA : HARI/TANGGAL : MINGGU , 7 DESEMBER 2003

PT. INDONESIA POWER

Jenis	00.30 01.00 01.30 02.00 02.30 03.00 03.30 04.00 04.30 05.00 05.30 06.00 06.30 07.00 07.30 08.00 08.30 09.00 09.30 10.00 10.30 11.00 11.30 12.00 12.30																							
	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
PLTA Area 2.	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	
PLTA SGLNG #1	122	103	102	108	104	105	108	100	121	121	80	84	80	80	80	80	90	91	111	111	111	111	111	
PLTA SGLNG #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA SGLNG #3	122	103	102	109	104	104	107	100	121	121	81	84	80	80	80	80	90	92	111	111	111	111	111	
PLTA SGLNG #4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTP KMJNG #1	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
PLTP KMJNG #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTP KMJNG #3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTP DRJAT #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTP SRAGI #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTP SRAGI #2	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
PLTP SRAGI #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTP SRAGI #4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #1	100	100	100	100	100	100	100	100	110	110	100	90	80	80	80	80	80	80	100	100	100	100	100	
PLTA CRATA #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI																								
PLTA CRATA #1	100	100	100	100	100	100	100	100	110	110	100	90	80	80	80	80	80	80	100	100	100	100	100	
PLTA CRATA #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTA CRATA #8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PEMBELIAN DARI LUAR PLN																								
WYNDU #1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	
DRJAT #2	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
Penum.Otorita Jatituhur	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	
(*) Pembangkitan Area-2	751	713	711	724	715	716	722	707	759	759	668	665	647	647	647	667	670	729	729	729	729	729	668	
(**) Beban Area-2	1724	1674	1666	1677	1628	1625	1580	1675	1695	1651	1463	1362	1275	1322	1321	1319	1308	1333	1352	1396	1414	1356	1234	
Selishih (*) - (**)	-973	-961	-955	-953	-913	-909	-958	-968	-936	-892	-795	-697	-628	-675	-674	-672	-641	-663	-623	-667	-605	-627	-587	
Cadangan Sekeloka	34	31	30	32	31	31	32	30	35	35	26	26	24	24	24	24	26	26	32	32	32	32	26	
Cadangan P u l a r	131	169	171	158	167	166	160	175	123	123	214	217	235	235	235	235	215	212	153	153	153	204	215	

RENCANA : HARTANGGAL : MINGGU , 7 DESEMBER 2003  
 PT. INDONESIA POWER SUB SISTEM REGION\_2

		Jam																								Rate 2			
		13.00	13.30	14.00	14.30	15.00	15.30	16.00	16.30	17.00	17.30	18.00	18.30	19.00	19.30	20.00	20.30	21.00	21.30	22.00	22.30	23.00	23.30	24.00					
PLTA	Area 2.	#1	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	24	26			
			SQUNG	#2	86	98	113	102	87	87	87	92	113	115	100	103	102	103	103	103	105	105	103	104	104	104	104	91	
				#3	0	0	0	0	0	0	0	0	0	0	80	80	103	102	103	103	105	105	103	104	104	104	104	32	
				#4	0	0	0	0	0	0	0	0	0	0	0	80	103	102	103	103	105	105	103	104	104	104	104	61	
PLTP	KUMUNG	#1	86	88	113	102	87	87	87	91	112	116	101	104	102	103	103	105	105	104	104	104	104	80	0	0	58		
			DRJAT	#2	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
				#3	0	0	0	0	0	0	0	0	0	0	30	50	50	50	50	50	50	50	50	50	50	50	50	50	
				#4	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50
PLTG	SRAGI	#1	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	4		
			DRJAT	#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
				#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
				#4	0	0	0	0	0	0	0	0	0	0	10	18	18	18	18	18	18	18	18	18	18	18	18	18	0
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI																													
PLTA	CRATA	#1	80	80	100	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	80	87		
			DRJAT	#2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19
				#3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
				#4	0	0	0	0	0	0	0	0	0	0	30	60	80	80	80	80	80	80	80	80	80	80	80	80	0
				#5	0	0	0	0	0	0	0	0	0	0	30	80	80	80	80	80	80	80	80	80	80	80	80	80	15
				#6	0	0	0	0	0	0	0	0	0	0	30	80	80	80	80	80	80	80	80	80	80	80	80	80	11
				#7	0	0	0	0	0	0	0	0	0	0	30	80	80	80	80	80	80	80	80	80	80	80	80	80	7
				#8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
				PEMBELIAN DARI LUAR PLN																									
Pembangkitan Area-2	WYNDU	#1	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105			
			DRJAT	#2	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
Pembangkitan Area-2	Cadangan P u l a r	#1	659	683	733	691	661	661	670	712	846	1056	1341	1375	1389	1387	1340	1343	1283	1139	1015	873	768	751	734	825			
			#2	1265	1295	1249	1235	1266	1324	1350	1306	1532	1704	1971	2005	2121	2116	2073	2022	1948	1874	1801	1810	1815	1711	1692	1567		
Cadangan Seketika	Cadangan P u l a r	#1	-606	-612	-516	-544	-625	-673	-690	-594	-686	-648	-630	-630	-722	-729	-733	-679	-655	-735	-766	-937	-1047	-960	-958	-742			
			#2	25	28	33	28	25	25	26	31	42	60	84	87	89	88	83	84	78	63	51	38	29	29	28	39		
				199	149	191	221	221	212	170	354	742	582	558	570	617	614	549	443	392	225	182	181	197	271				

		Jam																							
		00.30	01.00	01.30	02.00	02.30	03.00	03.30	04.00	04.30	05.00	05.30	06.00	06.30	07.00	07.30	08.00	08.30	09.00	09.30	10.00	10.30	11.00	11.30	
PLTA	Area 4.	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	
	SUTANI	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
PLTA	BRANTAS	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	73	73	73	73	73	73	73	73	
		PLTU	PITON #1	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370
	PITON #2	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	
PLTU 2	GRSIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GRSIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	PLTU 3	GRSIK220C	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
GRSIK230C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK310C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK10C3		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK20C3		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK320C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK330C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GRSIK310C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK320C		250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250	250
GRSIK330C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK310C		0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTU		GRSIK #3	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90
	GRSIK #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
	GRSIK #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	GRSIK #2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTG	GLTMR #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GLTMR #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRESIK #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRESIK #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GRESIK #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
(*) Pembangkitan Area-4 (**) Beban Area-4 Selisih (*) - (**) Cadangan Sekeloa Cadangan P u t a r	2357	2357	2307	2172	2172	2162	2162	2162	2162	2162	2162	2112	2862	2792	2692	2692	2817	2847	2897	2897	2897	2897	2897	2897	
	2095	2044	2037	2064	2031	2016	1997	2006	2039	2064	2044	1878	1805	1601	1515	1566	1614	1623	1664	1732	1688	1655	1649	1637	
	1162	1213	1170	1108	1141	1146	1165	1156	1123	1098	1234	1237	1191	1117	1126	1203	1224	1233	1145	1209	1242	1248	1260	1260	
	65	65	63	61	61	61	61	61	61	61	58	58	90	87	84	84	88	89	92	92	92	92	92	92	
	799	799	849	884	884	894	894	894	894	894	894	794	1044	1114	889	889	1094	1064	1064	1014	1014	1014	1014	1014	



# LAMPIRAN

3

unit uAbout;

interface

uses

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

type

TfrmAbout = class(TForm)

private

{ Private declarations }

public

{ Public declarations }

end;

var

frmAbout: TfrmAbout;

implementation

{ \$R \*.dfm }

end.

unit uComplex;

interface

uses uUtils;

type

TComplex=class

private

FReal,FIImag:double;

public

constructor Create;overload;

constructor Create(const aReal:double);overload;

constructor Create(const aReal,aImag:double);overload;

constructor Create(const aComplex:TComplex);overload;

function GetAbs:double;

function GetAngleRad:double;

function GetAngleDeg:double;

function Add(const aReal:double):TComplex;overload;

function Add(const aComplex:TComplex):TComplex;overload;

function Subtract(const aReal:double):TComplex;overload;

function Subtract(const aComplex:TComplex):TComplex;overload;

function Multiply(const aReal:double):TComplex;overload;

function Multiply(const aComplex:TComplex):TComplex;overload;

function Divide(const aReal:double):TComplex;overload;

function Divide(const aComplex:TComplex):TComplex;overload;

function Conj:TComplex;

function Negative:TComplex;

function ToStringI(const rLen:integer):string;

function ToStringJ(const rLen:integer):string;

property Real:double read FReal write FReal;

property Imag:double read FIImag write FIImag;

end;

CArr1=array of TComplex;

CArr2=array of array of TComplex;

implementation

{ TComplex }

//constructor

constructor TComplex.Create;

begin

inherited Create;

FReal:=0.0;

FIImag:=0.0;

end;

constructor TComplex.Create(const aReal:double);

begin

inherited Create;

FReal:=aReal;

FIImag:=0.0;

end;

```

constructor TComplex.Create(const aReal,aImag:double);
begin
  inherited Create;
  FReal:=aReal;
  FImag:=aImag;
end;

constructor TComplex.Create(const aComplex:TComplex);
begin
  inherited Create;
  FReal:=aComplex.FReal;
  FImag:=aComplex.FImag;
end;

//data operation

function TComplex.Add(const aReal:double):TComplex;
begin
  result:=TComplex.Create((FReal+aReal),FImag);
end;

function TComplex.Add(const aComplex:TComplex):TComplex;
begin
  result:=TComplex.Create((FReal+aComplex.FReal),(FImag+aComplex.FImag));
end;

function TComplex.Subtract(const aReal:double):TComplex;
begin
  result:=TComplex.Create((FReal-aReal),FImag);
end;

function TComplex.Subtract(const aComplex:TComplex):TComplex;
begin
  result:=TComplex.Create((FReal-aComplex.FReal),(FImag-aComplex.FImag));
end;

function TComplex.Multiply(const aReal:double):TComplex;
begin
  result:=TComplex.Create((aReal*FReal),(aReal*FImag));
end;

function TComplex.Multiply(const aComplex:TComplex):TComplex;
begin
  result:=TComplex.Create((FReal*aComplex.FReal-FImag*aComplex.FImag),
  (FReal*aComplex.FImag+FImag*aComplex.FReal));
end;
function TComplex.Divide(const aReal:double):TComplex;
begin
  result:=TComplex.Create((FReal/aReal),(FImag/aReal));
end;

function TComplex.Divide(const aComplex:TComplex):TComplex;
var denote:double;
begin
  denote:=sqr(aComplex.FReal)+sqr(aComplex.FImag);

```



```
    result:=TComplex.Create(((FReal*aComplex.FReal+FImag*aComplex.FImag)/denote),
        ((FImag*aComplex.FReal-FReal*aComplex.FImag)/denote));
end;
```

```
function TComplex.GetAbs:double;
begin
    result:=sqrt(sqrt(FReal)+sqrt(FImag));
end;
```

```
function TComplex.GetAngleRad:double;
begin
    result:=arctan(FImag/FReal);
end;
```

```
function TComplex.GetAngleDeg:double;
var phi:double;
begin
    phi:=4*arctan(1);
    result:=GetAngleRad*180/phi;
end;
```

```
function TComplex.Conj:TComplex;
begin
    result:=TComplex.Create(FReal,-FImag);
end;
```

```
function TComplex.Negative:TComplex;
begin
    result:=TComplex.Create((FReal*-1),(FImag*-1));
end;
```

```
function TComplex.toStringI(const rLen:integer):string;
begin
    if FImag<0 then
        begin
            result:=RealToStr(FReal,rLen)+'-'+RealToStr(FImag,rLen)+'i';
        end
    else
        begin
            result:=RealToStr(FReal,rLen)+'+'+RealToStr(FImag,rLen)+'i';
        end;
end;
```

```
function TComplex.toStringJ(const rLen:integer):string;
begin
    if FImag<0 then
        begin
            result:=RealToStr(FReal,rLen)+'-' j'+RealToStr(FImag,rLen);
        end
    else
        begin
            result:=RealToStr(FReal,rLen)+'+' j'+RealToStr(FImag,rLen);
        end;
end;
end.
```

```
unit uMatrix;
```

```
interface
```

```
uses uUtils, SysUtils;
```

```
function MatrixAdd(const mat1:dArr2;const aValue:double):dArr2;overload;  
function MatrixAdd(const mat1,mat2:dArr2):dArr2;overload;  
function MatrixSub(const mat1:dArr2;const aValue:double):dArr2;overload;  
function MatrixSub(const mat1,mat2:dArr2):dArr2;overload;  
function MatrixMul(const mat1:dArr2;const aValue:double):dArr2;overload;  
function MatrixMul(const mat1:dArr2;const mat2:dArr1):dArr1;overload;  
function MatrixMul(const mat1,mat2:dArr2):dArr2;overload;  
function MatrixInvers(const mat1:dArr2):dArr2;  
function MatrixTranspose(const mat1:dArr2):dArr2;  
function MatrixNegative(const mat1:dArr2):dArr2;  
function EllGauss(const mat1:dArr2;const mat2:dArr1):dArr1;
```

```
implementation
```

```
function MatrixAdd(const mat1:dArr2;const aValue:double):dArr2;  
var ij:integer;  
begin  
  SetLength(result,high(mat1)+1,high(mat1[0])+1);  
  for i:=0 to high(mat1) do  
    begin  
      for j:=0 to high(mat1[0]) do  
        begin  
          result[i,j]:=mat1[i,j]+aValue;  
        end;  
      end;  
    end;  
end;
```

```
function MatrixAdd(const mat1,mat2:dArr2):dArr2;  
var ij:integer;  
begin  
  if (high(mat1) < high(mat2)) or (high(mat1[0]) < high(mat2[0])) then  
    begin  
      raise Exception.Create('row dan col kedua matrik tidak sama!');  
    end;  
  SetLength(result,high(mat1)+1,high(mat1[0])+1);  
  for i:=0 to high(mat1) do  
    begin  
      for j:=0 to high(mat1[0]) do  
        begin  
          result[i,j]:=mat1[i,j]+mat2[i,j];  
        end;  
      end;  
    end;  
end;
```

```
function MatrixSub(const mat1:dArr2;const aValue:double):dArr2;  
var ij:integer;  
begin  
  SetLength(result,high(mat1)+1,high(mat1[0])+1);  
  for i:=0 to high(mat1) do
```

```

begin
  for j:=0 to high(mat1[0]) do
    begin
      result[i,j]:=mat1[i,j]-aValue;
    end;
  end;
end;

```

```

function MatrixSub(const mat1,mat2:dArr2):dArr2;
var i,j:integer;
begin
  if (high(mat1) <> high(mat2)) or (high(mat1[0]) <> high(mat2[0])) then
    begin
      raise Exception.Create('Row dan Col kedua matrik tidak sama!');
    end;
  SetLength(result,high(mat1)+1,high(mat1[0])+1);
  for i:=0 to high(mat1) do
    begin
      for j:=0 to high(mat1[0]) do
        begin
          result[i,j]:=mat1[i,j]-mat2[i,j];
        end;
      end;
    end;
end;

```

```

function MatrixMul(const mat1:dArr2;const aValue:double):dArr2;
var i,j:integer;
begin
  SetLength(result,high(mat1)+1,high(mat1[0])+1);
  for i:=0 to high(mat1) do
    begin
      for j:=0 to high(mat1[0]) do
        begin
          result[i,j]:=mat1[i,j]*aValue;
        end;
      end;
    end;
end;

```

```

function MatrixMul(const mat1:dArr2;const mat2:dArr1):dArr1;
var i,j,k:integer;
    sum:double;
begin
  if high(mat1[0]) <> high(mat2) then
    begin
      raise Exception.Create('Jumlah kolom matrik1 dan jumlah baris matrik2 tidak sama');
    end;
  SetLength(result,high(mat2)+1);
  for i:=0 to high(mat1) do
    begin
      for j:=0 to high(mat2) do
        begin
          sum:=0.0;
          for k:=0 to high(mat1[0]) do
            begin
              sum:=sum+mat1[i,k]*mat2[k];
            end;
          end;
        end;
      end;
    end;
end;

```

```

    end;
    result[i]:=sum;
  end;
end;
end;
end;

```

```

function MatrixMul(const mat1,mat2:dArr2):dArr2;
var i,j,k:integer;
    sum:double;
begin
  if high(mat1[0])<>high(mat2) then
    begin
      raise Exception.Create('Jumlah kolom matrik1 dan jumlah baris matrik2 tidak sama');
    end;
  SetLength(result,high(mat1)+1,high(mat2[0])+1);
  for i:=0 to high(mat1) do
    begin
      for j:=0 to high(mat2[0]) do
        begin
          sum:=0;
          for k:=0 to high(mat1[0]) do
            begin
              sum:=sum+mat1[i,k]*mat2[k,j];
            end;
          result[i,j]:=sum;
        end;
      end;
    end;
  end;
end;

```

```

function MatrixInvers(const mat1:dArr2):dArr2;
var i,j,k:integer;
    A,D:double;
begin
  if high(mat1)<>high(mat1[0]) then
    begin
      raise Exception.Create('Bukan matrik bujur sangkat!');
    end;
  SetLength(result,high(mat1)+1,high(mat1[0])+1);
  for i:=0 to high(mat1) do
    begin
      for j:=0 to high(mat1[0]) do
        begin
          result[i,j]:=mat1[i,j];
        end;
      end;
    end;
  try
    for i:=0 to high(result) do
      begin
        D:=result[i,i];
        result[i,i]:=1;
        for j:=0 to high(result[0]) do
          begin
            if D=0 then
              begin
                D:=0.00001;

```

```

    end;
    result[i,j]:=result[i,j]/D;
end;
for k:=0 to high(result) do
begin
    if k <> i then
    begin
        A:=result[k,i];
        result[k,i]:=0;
        for j:=0 to high(result[0]) do
        begin
            result[k,j]:=result[k,j]-A*result[i,j];
        end;
    end;
end;
end;
except
    raise exception.Create('matrik tidak bisa diinvers!');
end;
end;

```

```

function MatrixTranspose(const mat1:dArr2):dArr2;
var i,j:integer;
begin
    SetLength(result,high(mat1)+1,high(mat1[0])+1);
    for i:=0 to high(mat1) do
    begin
        for j:=0 to high(mat1[0]) do
        begin
            result[j,i]:=mat1[i,j];
        end;
    end;
end;
end;

```

```

function MatrixNegative(const mat1:dArr2):dArr2;
var i,j:integer;
begin
    SetLength(result,high(mat1)+1,high(mat1[0])+1);
    for i:=0 to high(mat1) do
    begin
        for j:=0 to high(mat1[0]) do
        begin
            result[i,j]:=mat1[i,j]*-1;
        end;
    end;
end;
end;

```

```

function EllGauss(const mat1:dArr2;const mat2:dArr1):dArr1;
var i,j,k,Nmat:integer;
    konst,value,DE,AE,sum:double;
    tmp:dArr2;
begin
    if high(mat1) <> high(mat1[0]) then
    begin
        raise Exception.Create('Matrik 1 bukan matrik bujur sangkar!');
    end;

```

```

end;
if high(mat1[0]) <> high(mat2) then
begin
  raise Exception.Create('Jumlah kolom matrik 1 tidak sama dengan jumlah baris matrik 2!');
end;
Nmat:=high(mat1)+1;
SetLength(tmp,Nmat,Nmat+1);
for i:=0 to Nmat-1 do
begin
  for j:=0 to Nmat-1 do
  begin
    tmp[i,j]:=mat1[i,j];
  end;
end;
for i:=0 to Nmat-1 do
begin
  tmp[i,Nmat]:=mat2[i];
end;
for i:=0 to Nmat-1 do
begin
  if tmp[i,i]=0 then
  begin
    for k:=i+1 to Nmat-1 do
    begin
      if tmp[k,i] <> 0 then
      begin
        for j:=0 to Nmat do
        begin
          Konst:=tmp[i,j];
          value:=tmp[k,j];
          tmp[i,j]:=value;
          tmp[k,j]:=Konst;
        end;
      end;
    end;
  end;
end;
for i:=0 to Nmat-1 do
begin
  DE:=tmp[i,i];
  for j:=0 to Nmat do
  begin
    tmp[i,j]:=tmp[i,j]/DE;
  end;
  for k:=i to Nmat-1 do
  begin
    if k <> i then
    begin
      AE:=tmp[k,i];
      for j:=0 to Nmat do
      begin
        tmp[k,j]:=tmp[k,j]-AE*tmp[i,j];
      end;
    end;
  end;
end;
end;

```

```
end;  
SetLength(result,Nmat);  
for i:=Nmat-1 downto 0 do  
begin  
  if i<>Nmat-1 then  
  begin  
    Sum:=0;  
    for j:=i+1 to Nmat-1 do  
    begin  
      Sum:=Sum+tmp[i,j]*result[j];  
    end;  
    result[i]:=tmp[i,Nmat]-Sum;  
  end  
  else  
  begin  
    result[i]:=tmp[i,Nmat];  
  end;  
end;  
end;  
  
end.
```

unit uMenu;

interface

uses

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

type

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

var

frmMenu: TfrmMenu;

implementation

uses uComplex, uUtils, uInputGen, uObjFunc, uGenerator;

{ \$R \*.dfm }

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

```
procedure TfrmMenu.btnNewClick(Sender: TObject);
begin
  frmInput.Caption:='Input Data';
  frmInput.btnNext.Caption:='&Save';
  frmInput.Show;
end;
```

```
procedure TfrmMenu.btnOpenClick(Sender: TObject);
var NamaFile, Nama:string;
    output:TextFile;
    Pmin,Pmax,a2,a1,a0,Sh,Sc,Ramp,Load,Res:double;
    i,j,Ngen,Njam,Tup,Tdown,Tcold,InitSt:integer;
    aLoad,aRes:dArr1;
    aPLN:dArr2;
```



```

aGen:TGenArr;
begin
try
if OpenFileDialog1.Execute then
begin
NamaFile:=OpenDialog1.FileName;
AssignFile(output,NamaFile);
Reset(output);
Readln(output,Ngen);
Readln(output,Njam);
frmInput.edtNGen.Text:=IntToStr(Ngen);
frmInput.edtNjam.Text:=IntToStr(Njam);
frmInput.fgGen.RowCount:=Ngen+1;
SetLength(aGen,Ngen+1);
for i:=1 to Ngen do
begin
Readln(output,Pmax,Pmin,a0,a1,a2,Tup,Tdown,Sh,Sc,tcold,InitSt,
Ramp,Nama);
aGen[i]:=TPembangkit.Create(Nama,Pmin,Pmax,a2,a1,a0,Sh,Sc,Ramp,
Tup,Tdown,Tcold,InitSt);
frmInput.fgGen.Cells[1,i]:=Nama;
frmInput.fgGen.Cells[2,i]:=FloatToStr(Pmax);
frmInput.fgGen.Cells[3,i]:=FloatToStr(Pmin);
frmInput.fgGen.Cells[4,i]:=FloatToStr(a0);
frmInput.fgGen.Cells[5,i]:=FloatToStr(a1);
frmInput.fgGen.Cells[6,i]:=FloatToStr(a2);
frmInput.fgGen.Cells[7,i]:=IntToStr(Tup);
frmInput.fgGen.Cells[8,i]:=IntToStr(Tdown);
frmInput.fgGen.Cells[9,i]:=FloatToStr(Sh);
frmInput.fgGen.Cells[10,i]:=FloatToStr(Sc);
frmInput.fgGen.Cells[11,i]:=IntToStr(Tcold);
frmInput.fgGen.Cells[12,i]:=IntToStr(InitSt);
frmInput.fgGen.Cells[13,i]:=FloatToStr(Ramp);
end;
frmInput.fgLoad.RowCount:=Njam+1;
SetLength(aLoad,Njam+1);
SetLength(aRes,Njam+1);
for i:=1 to Njam do
begin
Readln(output,Load,Res);
aLoad[i]:=Load;
aRes[i]:=Res;
frmInput.fgLoad.Cells[1,i]:=FloatToStr(Load);
frmInput.fgLoad.Cells[2,i]:=FloatToStr(Res);
end;
frmInput.fgPLN.RowCount:=Ngen+1;
frmInput.fgPLN.ColCount:=Njam+1;
SetLength(aPLN,Ngen+1,Njam+1);
for i:=1 to Ngen do
begin
for j:=1 to Njam do
begin
Read(output,Load);
aPLN[i,j]:=Load;
frmInput.fgPLN.Cells[j,i]:=FloatToStr(Load);

```

```
end;  
  Readln(output);  
end;  
CloseFile(output);  
gObjFunc:=TObjFunc.Create(aLoad,aRes,aPLN,aGen);  
for i:=1 to Ngen do  
begin  
  aGen[i].Free;  
end;  
frmInput.Caption:='Tampilan Data';  
frmInput.btnNext.Caption:='&Next';  
frmInput.Show;  
end;  
except  
  MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);  
end;  
end;  
  
end.
```

```

unit uSA;
interface
uses uUtils,uObjFunc,uHasil;

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

TSA=class
private
  FNgen,FNjam,FIterasi:integer;
  Fr,FTheta,FX0,FPflip:double;
  FBest,FOpt,FCek:TIndividu;
  function GetChromHasil:bArr2;
  procedure InitSA;
  function UbahSopt(const rScek:TIndividu):TIndividu;
  procedure doHitung;
public
  constructor Create;overload;
  constructor Create(const rNgen,rNjam,rIterasi:integer;
    const rR,rX0,rPflip:double);overload;
  destructor Destroy;override;
  property Ngen:integer read FNgen write FNgen;
  property Njam:integer read FNjam write FNjam;
  property Iterasi:integer read FIterasi write FIterasi;
  property R:double read FR write FR;
  property Theta:double read FTheta write FTheta;
  property X0:double read FX0 write FX0;
  property Pflip:double read FPflip write FPflip;
  property ChromHasil:bArr2 read GetChromHasil;
end;

implementation

//constructor
constructor TSA.Create;
begin
  inherited Create;
  FNgen:=0;
  FNjam:=0;
  FIterasi:=100;
  FR:=0.95;
  FX0:=0.95;
  FPflip:=0.3;
end;

constructor TSA.Create(const rNgen,rNjam,rIterasi:integer;
  const rR,rX0,rPflip:double);
begin
  inherited Create;
  FNgen:=rNgen;
  FNjam:=rNjam;
  FIterasi:=rIterasi;

```

```

Fr:=rR;
FX0:=rX0;
FPflip:=rPflip;
end;

//data processing
function TSA.UbahSopt(const rScek:Tindividu):Tindividu;
var i,j:integer;
begin
  SetLength(result.chrom,FNgen+1,FNJam+1);
  for i:=1 to FNgen do
    begin
      for j:=1 to FNJam do
        begin
          result.chrom[i,j]:=rScek.chrom[i,j];
        end;
      end;
    result.fitness:=rScek.fitness;
  end;
  procedure TSA.InitSA;
  var i,X1:integer;
      ran,dC:double;
  begin
    X1:=0;
    dC:=0;
    FOpt.chrom:=gObjFunc.getRandomChrom(FPflip);
    gObjFunc.doHitungChrom(FOpt.chrom,FOpt.fitness);
    for i:=1 to FIterasi do
      begin
        FCek.chrom:=gObjFunc.getRandomChrom(FPflip);
        gObjFunc.doHitungChrom(FCek.chrom,FCek.fitness);
        if FCek.fitness<FOpt.fitness then
          begin
            Fopt:=UbahSopt(Fcek);
          end
          else
            begin
              ran:=random;
              if FX0>ran then
                begin
                  X1:=X1+1;
                  dC:=dC+(FCek.fitness-FOpt.fitness);
                end;
              end;
            end;
            dC:=dC/X1;
            FTheta:=abs(dC/ln(1/X0-1));
          end;
        procedure TSA.doHitung;
        var i,k:integer;
            ran,dF,PdF,the:double;
        begin
          InitSA;
          frmHasil.pbIterasi.StepBy(1);
          the:=FTheta;

```

```

FOpt.chrom:=gObjFunc.getRandomChrom(FPflip);
gObjFunc.setLocalSearch(FOpt.chrom);
gObjFunc.doHitungChrom(FOpt.chrom,FOpt.fitness);
FBest:=UbahSopt(FOpt);
for k:=1 to 10 do
begin
for i:=1 to FIterasi do
begin
FCek.chrom:=gObjFunc.getRandomChrom(FPflip);
gObjFunc.setLocalSearch(FCek.chrom);
gObjFunc.doHitungChrom(FCek.chrom,FCek.fitness);
if FCek.fitness<FBest.fitness then
begin
FBest:=UbahSopt(FCek);
end;
if FCek.fitness<FOpt.fitness then
begin
FOpt:=UbahSopt(FCek);
end
else
begin
dF:=FCek.fitness-FOpt.fitness;
ran:=random;
PdF:=1/(1+exp(dF/the));
if ran<PdF then
begin
FOpt:=UbahSopt(FCek);
end;
end;
end;
the:=pangkat(r,k-1)*Ftheta;
frmHasil.pbIterasi.StepBy(1);
end;
end;

//data output
function TSA.GetChromHasil:bArr2;
var i,j:integer;
begin
SetLength(result,FNgen+1,FNjam+1);
doHitung;
for i:=1 to FNgen do
begin
for j:=1 to FNjam do
begin
result[i,j]:=FBest.chrom[i,j];
end;
end;
end;
//destructor
destructor TSA.Destroy;
begin
inherited Destroy;
end;
end.

```

```

unit uUtils;

interface

uses SysUtils;

type
  TSort=(asc,dec);

  TBatas=record
    min,max:double;
  end;

  TBatasArr1=array of TBatas;
  TBatasArr2=array of array of TBatas;

  dArr1=array of double;
  dArr2=array of array of double;
  iArr1=array of integer;
  iArr2=array of array of integer;
  bArr1=array of boolean;
  bArr2=array of array of boolean;
  sArr1=array of String;

  TAlleleTCSC=record
    Lokasi,TypeAlat,Setting:double;
  end;

  TChromTCSC1=array of TAlleleTCSC;

  TAlleleUpfc=record
    Status:boolean;
    TypeAlat,Tap,Sudut:double;
  end;

  TChromUpfc1=array of TAlleleUpfc;

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

procedure Swap(var X,Y:byte);overload;
procedure Swap(var X,Y:integer);overload;
procedure Swap(var X,Y:word);overload;
procedure Swap(var X,Y:double);overload;
procedure Swap(var X,Y:extended);overload;
procedure Swap(var X,Y:string);overload;
procedure Swap(var X,Y:boolean);overload;

procedure BubleSort(var aData:dArr1;const aType:TSort);overload;
procedure BubleSort(var aData:iArr1;const aType:TSort);overload;
procedure BubleSort(var aData:sArr1;const aType:TSort);overload;

function DecodeBinToFloat1(const aData:bArr1):double;
function DecodeBinToFloat2(const aData:bArr2):dArr1;

```

```
function GetBatas(const aValue,aMin,aMax:double):double;  
function GetFlip(const aFlip:double):boolean;
```

```
function GetBatasToReal(const aValue,aMin,aMax:double):double;  
function GetRealToBatas(const aValue,aMin,aMax:double):double;
```

implementation

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

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

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

```
procedure Swap(var X,Y:byte);  
var tmp:byte;  
begin  
    tmp:=X;  
    X:=Y;  
    Y:=tmp;  
end;
```

```
procedure Swap(var X,Y:integer);  
var tmp:integer;  
begin  
    tmp:=X;  
    X:=Y;  
    Y:=tmp;  
end;
```

```
procedure Swap(var X,Y:word);  
var tmp:word;  
begin  
    tmp:=X;  
    X:=Y;  
    Y:=tmp;  
end;
```

```
procedure Swap(var X,Y:double);
var tmp:double;
begin
  tmp:=X;
  X:=Y;
  Y:=tmp;
end;
```

```
procedure Swap(var X,Y:extended);
var tmp:extended;
begin
  tmp:=X;
  X:=Y;
  Y:=tmp;
end;
```

```
procedure Swap(var X,Y:string);
var tmp:string;
begin
  tmp:=X;
  X:=Y;
  Y:=tmp;
end;
```

```
procedure Swap(var X,Y:boolean);
var tmp:boolean;
begin
  tmp:=X;
  X:=Y;
  Y:=tmp;
end;
```

```
procedure BubbleSort(var aData:dArr1;const aType:TSort);
var i,j:integer;
begin
  for i:=1 to (high(aData)-1) do
  begin
    for j:=i to high(aData) do
    begin
      if aType=asc then
      begin
        if aData[i]>aData[j] then
          begin
            Swap(aData[i],aData[j]);
          end;
        end
      else if aType=dec then
      begin
        if aData[i]<aData[j] then
          begin
            Swap(aData[i],aData[j]);
          end;
        end;
      end;
    end;
  end;
end;
```



end;

```
procedure BubleSort(var aData:iArr1;const aType:TSort);
var i,j:integer;
begin
  for i:=1 to (high(aData)-1) do
  begin
    for j:=i to high(aData) do
    begin
      if aType=asc then
      begin
        if aData[i]>aData[j] then
        begin
          Swap(aData[i],aData[j]);
        end;
      end
      else if aType=dec then
      begin
        if aData[i]<aData[j] then
        begin
          Swap(aData[i],aData[j]);
        end;
      end;
    end;
  end;
end;
```

```
procedure BubleSort(var aData:sArr1;const aType:TSort);
var i,j:integer;
begin
  for i:=1 to (high(aData)-1) do
  begin
    for j:=i to high(aData) do
    begin
      if aType=asc then
      begin
        if aData[i]>aData[j] then
        begin
          Swap(aData[i],aData[j]);
        end;
      end
      else if aType=dec then
      begin
        if aData[i]<aData[j] then
        begin
          Swap(aData[i],aData[j]);
        end;
      end;
    end;
  end;
end;
```

```
function DecodeBinToFloat1(const aData:bArr1):double;
var i:integer;
    powerof2,sa:double;
```

```

begin
  result:=0;
  powerof2:=1;
  sa:=pangkat(2,high(aData))-1;
  for i:=high(aData) downto 1 do
  begin
    if aData[i]=true then
    begin
      result:=result+powerof2;
    end;
    powerof2:=powerof2*2;
  end;
  result:=result/sa;
end;

```

```

function DecodeBinToFloat2(const aData:bArr2):dArr1;
var i,j:integer;
    Data:bArr1;
begin
  SetLength(Data,high(aData[0])+1);
  SetLength(result,high(aData)+1);
  for i:=1 to high(aData) do
  begin
    for j:=1 to high(aData[0]) do
    begin
      Data[j]:=aData[i,j];
    end;
    result[i]:=DecodeBinToFloat1(Data);
  end;
end;

```

```

function GetBatas(const aValue,aMin,aMax:double):double;
begin
  if aValue>1.0 then raise Exception.Create('Value tidak boleh lebih dari 1');
  if aValue<0.0 then raise Exception.Create('Value tidak boleh kurang dari 0');
  result:=aMin+aValue*(aMax-aMin);
end;

```

```

function GetFlip(const aFlip:double):boolean;
begin
  result:=false;
  if random<=aFlip then result:=true;
end;

```

```

function GetBatasToReal(const aValue,aMin,aMax:double):double;
begin
  result:=aMin+aValue*(aMax-aMin);
end;

```

```

function GetRealToBatas(const aValue,aMin,aMax:double):double;
begin
  result:=(aValue-aMin)/(aMax-aMin);
end;

```

```

end.

```

```
unit uGenerator;
```

```
interface
```

```
type
```

```
TPembangkit=class
```

```
private
```

```
FNama:string;
```

```
FPmax,FPmin,Fa2,Fa1,Fa0,FSh,FSc,FDaya,FRamp:double;
```

```
FTup,FTdown,FTcold,FInitSt:integer;
```

```
function GetAFLC:double;
```

```
public
```

```
constructor Create;overload;
```

```
constructor Create(const rNama:string;
```

```
const rPmin,rPmax,ra2,ra1,ra0,rSh,rSc,rRamp:double;
```

```
const rTup,rTdown,rTcold,rInitSt:integer);overload;
```

```
constructor Create(const rPembangkit:TPembangkit);overload;
```

```
procedure Assign(const rPembangkit:TPembangkit);
```

```
function GetBiaya(const rDaya:double):double;
```

```
function GetDaya(const rLamda:double):double;
```

```
function GetLamda(const rDaya:double):double;
```

```
property Nama:string read FNama write FNama;
```

```
property Pmax:double read FPmax write FPmax;
```

```
property Pmin:double read FPmin write FPmin;
```

```
property a2:double read Fa2 write Fa2;
```

```
property a1:double read Fa1 write Fa1;
```

```
property a0:double read Fa0 write Fa0;
```

```
property Sh:double read FSh write FSh;
```

```
property Sc:double read FSc write FSc;
```

```
property Ramp:double read FRamp write FRamp;
```

```
property Tup:integer read FTup write FTup;
```

```
property Tdown:integer read FTdown write FTdown;
```

```
property Tcold:integer read FTcold write FTcold;
```

```
property InitSt:integer read FInitSt write FInitSt;
```

```
property Daya:double read FDaya write FDaya;
```

```
property AFLC:double read GetAFLC;
```

```
end;
```

```
TGenArr=array of TPembangkit;
```

```
implementation
```

```
//constructor
```

```
constructor TPembangkit.Create;
```

```
begin
```

```
inherited Create;
```

```
end;
```

```
constructor TPembangkit.Create(const rNama:string;
```

```
const rPmin,rPmax,ra2,ra1,ra0,rSh,rSc,rRamp:double;
```

```
const rTup,rTdown,rTcold,rInitSt:integer);
```

```
begin
```

```
inherited Create;
```

```
FNama:=rNama;
```

```
FPmin:=rPmin;
```

```

FPmax:=rPmax;
Fa2:=ra2;
Fa1:=ra1;
Fa0:=ra0;
FSh:=rSh;
FSc:=rSc;
FRamp:=rRamp;
FTup:=rTup;
FTdown:=rTdown;
FTcold:=rTcold;
FInitSt:=rInitSt;
end;

constructor TPembangkit.Create(const rPembangkit:TPembangkit);
begin
  inherited Create;
  FNama:=rPembangkit>Nama;
  FPmin:=rPembangkit>Pmin;
  FPmax:=rPembangkit>Pmax;
  Fa2:=rPembangkit>a2;
  Fa1:=rPembangkit>a1;
  Fa0:=rPembangkit>a0;
  FSh:=rPembangkit>Sh;
  FSc:=rPembangkit>Sc;
  FRamp:=rPembangkit>Ramp;
  FTup:=rPembangkit>Tup;
  FTdown:=rPembangkit>Tdown;
  FTcold:=rPembangkit>Tcold;
  FInitSt:=rPembangkit>InitSt;
end;

function TPembangkit.GetAFLC:double;
begin
  Result:=fa0/fPmax+fa1+fa2*fPmax;
end;

procedure TPembangkit.Assign(const rPembangkit:TPembangkit);
begin
  FNama:=rPembangkit>Nama;
  FPmin:=rPembangkit>Pmin;
  FPmax:=rPembangkit>Pmax;
  Fa2:=rPembangkit>a2;
  Fa1:=rPembangkit>a1;
  Fa0:=rPembangkit>a0;
  FSh:=rPembangkit>Sh;
  FSc:=rPembangkit>Sc;
  FRamp:=rPembangkit>Ramp;
  FTup:=rPembangkit>Tup;
  FTdown:=rPembangkit>Tdown;
  FTcold:=rPembangkit>Tcold;
  FInitSt:=rPembangkit>InitSt;
end;

//data operation

```

```

function TPembangkit.GetBiaya(const rDaya:double):double;
begin
  result:=0;
  if rDaya<>0 then
  begin
    result:=Fa2*sqr(rDaya)+Fa1*rDaya+Fa0;
  end;
end;

function TPembangkit.GetDaya(const rLamda:double):double;
begin
  result:=(rLamda-Fa1)/(2*Fa2);
  if result>FPmax then result:=FPmax;
  if result<FPmin then result:=FPmin;
end;

function TPembangkit.GetLamda(const rDaya:double):double;
begin
  result:=2*Fa2*rDaya-Fa1;
end;

end.

```

unit uHasil;

interface

uses

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

type

```
TfrmHasil = class(TForm)
  TabSheet5: TTabSheet;
  TabSheet6: TTabSheet;
  TabSheet7: TTabSheet;
  Panel1: TPanel;
  btnClose: TButton;
  btnHitungEP: TButton;
  TabSheet1: TTabSheet;
  TabSheet2: TTabSheet;
  Chart1: TChart;
  Series1: TLineSeries;
  Series2: TLineSeries;
  TabSheet4: TPageControl;
  fgCostPerJam: TStringGrid;
  Label8: TLabel;
  edtTotalProgram: TEdit;
  Label9: TLabel;
  edtTotalPLN: TEdit;
  Label1: TLabel;
  edtSelisih: TEdit;
  fgDaya: TStringGrid;
  fgStatus: TStringGrid;
  gbParam: TGroupBox;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
  Label7: TLabel;
  edtIterasi: TEdit;
  edtR: TEdit;
  edtT0: TEdit;
  edtflip: TEdit;
  btnUseDefault: TButton;
  pbIterasi: TProgressBar;
  procedure btnCloseClick(Sender: TObject);
  procedure FormCreate(Sender: TObject);
  procedure btnHitungEPClick(Sender: TObject);
  procedure btnUseDefaultClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
```

var

frmHasil: TfrmHasil;

implementation

uses uObjFunc, uUtils, uSA;

{ \$R \*.dfm }

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

```
procedure TfrmHasil.FormCreate(Sender: TObject);
begin
  fgCostPerJam.Cells[0,0]:='Jam';
  fgCostPerJam.Cells[1,0]:='Biaya Program';
  fgCostPerJam.Cells[2,0]:='Biaya PLN';
  fgCostPerJam.Cells[3,0]:='Selisih Biaya';
end;
```

```
procedure TfrmHasil.btnHitungEPClick(Sender: TObject);
var i,j:integer;
    CostTotal, CostPLN:double;
    chrom:bArr2;
    PL:dArr2;
    CostPerJam, CostPerJamPLN:dArr1;
    sas:TSA;
begin
  //gObjFunc.doExecute(chrom, PL, CostPerJam, CostTotal);
  sas:=TSA.Create(gObjFunc.Ngen, gObjFunc.Njam, StrToInt(edtIterasi.Text),
    StrToFloat(edtR.Text), StrToFloat(edtT0.Text),
    StrToFloat(edtflip.Text));
  pblterasi.Max:=11;
  chrom:=sas.ChromHasil;
  sas.Free;
  gObjFunc.doHitungChrom(chrom, PL, CostPerJam, CostTotal);
  for i:=1 to high(chrom) do
  begin
    for j:=1 to high(chrom[0]) do
    begin
      fgDaya.Cells[j,i]:=RealToStr(PL[i,j],0);
      if chrom[i,j]=true then
      begin
        fgStatus.Cells[j,i]:='1';
      end
      else
      begin
        fgStatus.Cells[j,i]:='0';
      end;
    end;
  end;
  edtTotalProgram.Text:=FormatFloat('#,##0', CostTotal);
  gObjFunc.doHitungPLN(CostPerJamPLN, CostPLN);
  Series1.Clear;
  Series2.Clear;
```

```

for i:=1 to high(CostPerJam) do
begin
  fgCostPerJam.Cells[0,i]:=IntToStr(i);
  fgCostPerJam.Cells[1,i]:=FormatFloat('#,##0',CostPerJam[i]);
  fgCostPerJam.Cells[2,i]:=FormatFloat('#,##0',CostPerJamPLN[i]);
  fgCostPerJam.Cells[3,i]:=FormatFloat('#,##0',
    (CostPerJamPLN[i]-CostPerJam[i]));
  Series1.Add(CostPerJam[i],IntToStr(i));
  Series2.Add(CostPerJamPLN[i],IntToStr(i));
end;
edtTotalPLN.Text:=FormatFloat('#,##0',CostPLN);
edtSelisih.Text:=FormatFloat('#,##0',(CostPLN-CostTotal));
end;

procedure TfrmHasil.btnUseDefaultClick(Sender: TObject);
begin
  edtIterasi.Text:='100';
  edtR.Text:='0.95';
  edtT0.Text:='0.05';
  edtflip.Text:='0.1';
  btnHitungEP.Enabled:=true;
end;

end.

```



**unit uInputGen;**

**interface**

**uses**

**Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, ComCtrls, ExtCtrls, 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;  
  edtNGen: TEdit;  
  edtNjam: TEdit;  
  fgGen: TStringGrid;  
  fgLoad: TStringGrid;  
  TabSheet4: TTabSheet;  
  fgPLN: TStringGrid;  
  procedure btnCloseClick(Sender: TObject);  
  procedure FormCreate(Sender: TObject);  
  procedure edtNGenChange(Sender: TObject);  
  procedure edtNjamChange(Sender: TObject);  
  procedure btnNextClick(Sender: TObject);  
private  
  { Private declarations }  
public  
  { Public declarations }  
end;
```

**var**

**frmInput: TfrmInput;**

**implementation**

**uses uObjFunc, uHasil;**

**{ \$R \*.dfm }**

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

```
procedure TfrmInput.FormCreate(Sender: TObject);  
begin  
  fgGen.Cells[0,0] := 'Gen';  
  fgGen.Cells[1,0] := 'Nama';
```

```

fgGen.Cells[2,0] := 'Pmax';
fgGen.Cells[3,0] := 'Pmin';
fgGen.Cells[4,0] := 'a0';
fgGen.Cells[5,0] := 'a1';
fgGen.Cells[6,0] := 'a2';
fgGen.Cells[7,0] := 'Tup';
fgGen.Cells[8,0] := 'Tdown';
fgGen.Cells[9,0] := 'Sh';
fgGen.Cells[10,0] := 'Sc';
fgGen.Cells[11,0] := 'Tcold';
fgGen.Cells[12,0] := 'InitSt';
fgGen.Cells[13,0] := 'Ramp Rate';
fgLoad.Cells[0,0] := '';
fgLoad.Cells[1,0] := 'Load';
fgLoad.Cells[2,0] := 'Res';
end;

```

```

procedure TfrmInput.edtNgenChange(Sender: TObject);
var i:integer;
begin
  if edtNgen.Text="" then
  begin
    fgGen.RowCount:=2;
    fgPLN.RowCount:=2;
  end
  else
  begin
    fgGen.RowCount:=StrToInt(edtNgen.Text)+1;
    fgPLN.RowCount:=StrToInt(edtNgen.Text)+1;
    for i:=1 to StrToInt(edtNgen.Text) do
    begin
      fgGen.Cells[0,i]:=IntToStr(i);
      fgPLN.Cells[0,i]:='Gen '+IntToStr(i);
    end;
  end;
end;

```

```

procedure TfrmInput.edtNjamChange(Sender: TObject);
var i:integer;
begin
  if edtNjam.Text="" then
  begin
    fgLoad.RowCount:=2;
    fgPLN.ColCount:=2;
  end
  else
  begin
    fgLoad.RowCount:=StrToInt(edtNjam.Text)+1;
    fgPLN.ColCount:=StrToInt(edtNjam.Text)+1;
    for i:=1 to StrToInt(edtNjam.Text) do
    begin
      fgLoad.Cells[0,i]:=IntToStr(i);
      fgPLN.Cells[i,0]:='Jam '+IntToStr(i);
    end;
  end;
end;

```

```

end;

procedure TfrmInput.btnNextClick(Sender: TObject);
var input:TextFile;
    NamaFile,Nama:string;
    Pmin,Pmax,a2,a1,a0,Sh,Sc,Ramp,Load,Res:double;
    i,j,Tup,Tdown,Tcold,InitSt,Ngen,Njam:integer;
begin
if btnNext.Caption='&Save' then
begin
if SaveDialog1.Execute then
begin
NamaFile:=SaveDialog1.FileName;
AssignFile(input,NamaFile+'.txt');
Reset(input);
Ngen:=StrToInt(edtNgen.Text);
Njam:=StrToInt(edtNjam.Text);
Writeln(input,Ngen);
Writeln(input,Njam);
for i:=1 to Ngen do
begin
Nama:=fgGen.Cells[1,i];
Pmax:=StrToFloat(fgGen.Cells[2,i]);
Pmin:=StrToFloat(fgGen.Cells[3,i]);
a0:=StrToFloat(fgGen.Cells[4,i]);
a1:=StrToFloat(fgGen.Cells[5,i]);
a2:=StrToFloat(fgGen.Cells[6,i]);
Tup:=StrToInt(fgGen.Cells[7,i]);
Tdown:=StrToInt(fgGen.Cells[8,i]);
Sh:=StrToFloat(fgGen.Cells[9,i]);
Sc:=StrToFloat(fgGen.Cells[10,i]);
Tcold:=StrToInt(fgGen.Cells[11,i]);
InitSt:=StrToInt(fgGen.Cells[12,i]);
Ramp:=StrToFloat(fgGen.Cells[13,i]);
Writeln(input,Pmax:7:0,' ',Pmin:7:0,' ',
a0:9:4,' ',a1:9:4,' ',a2:9:5,' ',Tup,' ',Tdown,' ',
Sh:7:0,' ',Sc:7:0,' ',tcold,' ',InitSt,' ',Ramp:7:0,' ',Nama);
end;
for i:=1 to Njam do
begin
Load:=StrToFloat(fgLoad.Cells[1,i]);
Res:=StrToFloat(fgLoad.Cells[2,i]);
Writeln(input,Load,Res);
end;
fgPLN.RowCount:=Ngen+1;
fgPLN.ColCount:=Njam+1;
for i:=1 to Ngen do
begin
for j:=1 to Njam do
begin
Load:=StrToFloat(fgPLN.Cells[j,i]);
Write(input,Load:7:2,' ');
end;
Writeln(input,"");
end;
end;
end;

```

```
    CloseFile(input);
end;
end
else if btnNext.Caption='&Next' then
begin
    frmHasil.fgStatus.RowCount:=gObjFunc.Ngen+1;
    frmHasil.fgStatus.ColCount:=gObjFunc.Njam+1;
    frmHasil.fgDaya.RowCount:=gObjFunc.Ngen+1;
    frmHasil.fgDaya.ColCount:=gObjFunc.Njam+1;
    frmHasil.fgCostPerJam.RowCount:=gObjFunc.Njam+1;
    for i:=1 to gObjFunc.Ngen do
    begin
        frmHasil.fgStatus.Cells[0,i]:='Unit '+IntToStr(i);
        frmHasil.fgDaya.Cells[0,i]:='Unit '+IntToStr(i);
    end;
    for i:=1 to gObjFunc.Njam do
    begin
        frmHasil.fgStatus.Cells[i,0]:='Jam '+IntToStr(i);
        frmHasil.fgDaya.Cells[i,0]:='Jam '+IntToStr(i);
    end;
    frmHasil.Show;
end;
end;
end.
```

```

unit uObjFunc;

interface

uses uUtils,uGenerator,SysUtils;

type
  TObjFunc=class
  private
    FNgen,FNjam:integer;
    FBeban,FRes,FAFLC:dArr1;
    FSortAFLC:iArr1;
    FPLN:dArr2;
    FGen:TGenArr;
    function GetBeban:dArr1;
    function GetRes:dArr1;
    function GetPLN:dArr2;
    function GetGen:TGenArr;
    procedure SetGen(const rGen:TGenArr);
    procedure SetBeban(const rBeban:dArr1);
    procedure SetRes(const rRes:dArr1);
    procedure SetPLN(const rPLN:dArr2);
    function isON(const rFlip:double):boolean;
    function isServe(const rJam:integer;const rChrom:bArr1):boolean;
    function isRampRate(const rJam:integer;const rPL:dArr2):boolean;
    function FindAFLC(const ri:integer):integer;
    procedure RepairAFLC(var rChrom:bArr2);
    function CreateChromBase:bArr2;
    function CreateChromONOFF:bArr2;
    function HitungEcoDis(const rJam:integer;
      const rChrom1:bArr1):dArr1;
    function GetSortAFLC:iArr1;
    function GetSortChrom(const rRank:integer):bArr1;
    function HitungCostGen(const rPL:dArr2):dArr2;
    function HitungCostSUC(const rPL:dArr2):dArr2;
    function GantiChrom(const rChrom:bArr2):bArr2;
    function GetSwap(const rChrom:bArr2):bArr2;
    function doCariGreyZone(const rChrom:bArr2):bArr2;
  public
    constructor Create;overload;
    constructor Create(const rBeban,rRes:dArr1;
      const rPLN:dArr2;
      const rGen:TGenArr);overload;
    function getRandomChrom(const rFlip:double):bArr2;
    function getConstructSolution(const rFlip:double):bArr2;
    procedure setLocalSearch(var rChrom:bArr2);
    procedure doHitungChrom(const rChrom:bArr2;
      var rCostTotal:double);overload;
    procedure doHitungChrom(const rChrom:bArr2;
      var rPL:dArr2;
      var rCostPerJam:dArr1;
      var rCostTotal:double);overload;
    procedure doHitungPLN(
      var rCostPerJam:dArr1;

```

```

        var rCostTotal:double);
procedure doExecute(var rChrom:bArr2;
    var rPL:dArr2;
    var rCostPerJam:dArr1;
    var rCostTotal:double);
destructor Destroy;override;
property Ngen:integer read FNgen write FNgen;
property Njam:integer read FNjam write FNjam;
property Gen:TGenArr read GetGen write SetGen;
property Beban:dArr1 read GetBeban write SetBeban;
property PLN:dArr2 read GetPLN write SetPLN;
property Res:dArr1 read GetRes write SetRes;
property SortAFLC:iArr1 read GetSortAFLC;
end;

var gObjFunc:TObjFunc;

implementation

//constructor
constructor TObjFunc.Create;
begin
    inherited Create;
    FNgen:=0;
    FNjam:=0;
end;

constructor TObjFunc.Create(const rBeban,rRes:dArr1;
    const rPLN:dArr2;
    const rGen:TGenArr);
var i,j,Ncek:integer;
begin
    inherited Create;
    FNgen:=high(rGen);
    FNjam:=high(rBeban);
    Ncek:=high(rRes);
    if FNjam<Ncek then raise Exception.Create('Dimensi matrik tidak sama!');
    SetLength(FGen,FNgen+1);
    SetLength(FBeban,FNjam+1);
    SetLength(FRes,FNjam+1);
    SetLength(FAFLC,FNgen+1);
    for i:=1 to FNgen do
        begin
            FGen[i]:=TPembangkit.Create(rGen[i]);
            FAFLC[i]:=FGen[i].AFLC;
        end;
    for i:=1 to FNjam do
        begin
            FBeban[i]:=rBeban[i];
            FRes[i]:=rRes[i];
        end;
    SetLength(FPLN,FNgen+1,FNjam+1);
    for i:=1 to FNgen do
        begin
            for j:=1 to FNjam do

```

```
begin
  FPLN[i,j]:=rPLN[i,j];
end;
end;
FSortAFLC:=GetSortAFLC;
end;
```

```
//data accessing
function TObjFunc.GetBeban:dArr1;
var i:integer;
begin
  SetLength(result, FNjam+1);
  for i:=1 to FNjam do
    begin
      result[i]:=FBeban[i];
    end;
  end;
```

```
function TObjFunc.GetRes:dArr1;
var i:integer;
begin
  SetLength(result, FNjam+1);
  for i:=1 to FNjam do
    begin
      result[i]:=FRes[i];
    end;
  end;
```

```
function TObjFunc.GetPLN:dArr2;
var i,j:integer;
begin
  SetLength(result, FNgen+1, FNjam+1);
  for i:=1 to FNgen do
    begin
      for j:=1 to FNjam do
        begin
          result[i,j]:=FPLN[i,j];
        end;
      end;
    end;
end;
```

```
function TObjFunc.GetGen:TGenArr;
var i:integer;
begin
  SetLength(result, FNgen+1);
  for i:=1 to FNgen do
    begin
      result[i]:=TPembangkit.Create(FGen[i]);
    end;
  end;
```

```
procedure TObjFunc.SetGen(const rGen:TGenArr);
var i:integer;
begin
  FNgen:=high(rGen);
```

```

SetLength(FGen,FNgen+1);
SetLength(FAFLC,FNgen+1);
for i:=1 to FNgen do
begin
  FGen[i]:=TPembangkit.Create(rGen[i]);
  FAFLC[i]:=FGen[i].AFLC;
end;
FSortAFLC:=GetSortAFLC;
end;

```

```

procedure TObjFunc.SetBeban(const rBeban:dArr1);
var i,Ncek:integer;
begin
  if FNjam>0 then
  begin
    Ncek:=high(rBeban);
    if FNjam<>Ncek then raise Exception.Create('Dimensi matrik tidak sama!');
  end
  else
  begin
    FNjam:=high(rBeban);
  end;
  SetLength(FBeban,FNjam+1);
  for i:=1 to FNjam do
  begin
    FBeban[i]:=rBeban[i];
  end;
end;

```

```

procedure TObjFunc.SetRes(const rRes:dArr1);
var i,Ncek:integer;
begin
  if FNjam>0 then
  begin
    Ncek:=high(rRes);
    if FNjam<>Ncek then raise Exception.Create('Dimensi matrik tidak sama!');
  end
  else
  begin
    FNjam:=high(rRes);
  end;
  SetLength(FRes,FNjam+1);
  for i:=1 to FNjam do
  begin
    FRes[i]:=rRes[i];
  end;
end;

```

```

procedure TObjFunc.SetPLN(const rPLN:dArr2);
var i,j,Ncek:integer;
begin
  if FNgen>0 then
  begin
    Ncek:=high(rPLN);
    if FNgen<>Ncek then raise Exception.Create('Dimensi matrik tidak sama!');
  end;
end;

```



```

end
else
begin
  FNgen:=high(rPLN);
end;
if FNjam>0 then
begin
  Ncek:=high(rPLN[0]);
  if FNjam<>Ncek then raise Exception.Create('Dimensi matrik tidak sama!');
end
else
begin
  FNjam:=high(rPLN[0]);
end;
SetLength(FPLN,FNgen+1,FNjam+1);
for i:=1 to FNgen do
begin
  for j:=1 to FNjam do
  begin
    FPLN[i,j]:=rPLN[i,j];
  end;
end;
end;

//data processing
function TObjFunc.isON(const rFlip:double):boolean;
begin
  result:=false;
  if random<=rFlip then result:=true;
end;

function TObjFunc.isServe(const rJam:integer;const rChrom:bArr1):boolean;
var i:integer;
    load,sBebanMin,sBebanMax:double;
begin
  result:=true;
  sBebanMin:=0;
  sBebanMax:=0;
  for i:=1 to FNgen do
  begin
    if rChrom[i]=true then
    begin
      sBebanMin:=sBebanMin+FGen[i].Pmin;
      sBebanMax:=sBebanMax+FGen[i].Pmax;
    end;
  end;
  load:=FBeban[rJam]+FRes[rJam];
  if load<sBebanMin then result:=false;
  if load>sBebanMax then result:=false;
end;

function TObjFunc.isRampRate(const rJam:integer;const rPL:dArr2):boolean;
var i:integer;
    delta:double;
begin

```

```

result:=true;
for i:=1 to FNgen do
begin
  if rJam>1 then
  begin
    delta:=rPL[i,rJam]-rPL[i,rJam-1];
    if delta>0 then
    begin
      if delta>FGen[i].Ramp then
      begin
        result:=false;
        break;
      end;
    end;
  end;
end;
end;
end;

```

```

function TObjFunc.HitungEcoDis(const rJam:integer;
  const rChrom1:bArr1):dArr1;
var i,j:integer;
  Status:bArr1;
  LoadCek,Pa,Pb,Lmd,LoadSplit:double;
  aBeban,diffa2,diffa1,Cek,tes:double;
begin
  SetLength(Status,FNgen+1);
  for i:=1 to FNgen do
  begin
    Status[i]:=rChrom1[i];
    FGen[i].Daya:=0;
  end;
  aBeban:=FBeban[rJam];
  LoadCek:=aBeban;
  LoadSplit:=aBeban;
  for i:=1 to 15 do
  begin
    Pa:=0;
    Pb:=0;
    for j:=1 to FNgen do
    begin
      if Status[j] then
      begin
        diffa2:=FGen[j].a2*2;
        diffa1:=FGen[j].a1;
        Pa:=Pa+1/diffa2;
        Pb:=Pb+diffa1/diffa2;
      end;
    end;
  end;
  if Pa<=0 then
  begin
    Lmd:=(LoadSplit+Pb)/Pa;
  end
  else
  begin
    Lmd:=LoadSplit+Pb;
  end;
end;

```

```

end;
Cek:=0;
for j:=1 to FNgen do
begin
if Status[j] then
begin
dffa2:=2*FGen[j].a2;
dffa1:=FGen[j].a1;
FGen[j].Daya:=(Lmd-dffa1)/dffa2;
if FGen[j].Daya<FGen[j].Pmin then
begin
FGen[j].Daya:=FGen[j].Pmin;
end;
if FGen[j].Daya>FGen[j].Pmax then
begin
FGen[j].Daya:=FGen[j].Pmax;
end;
end;
end;
Cek:=Cek+FGen[j].Daya;
end;
tes:=LoadCek-Cek;
if (tes<0.0001) and (tes>-0.0001) then
begin
break;
end
else if tes>0 then
begin
for j:=1 to fNgen do
begin
if Status[j] then
begin
if FGen[j].Daya=FGen[j].PMax then
begin
Status[j]:=false;
LoadSplit:=LoadSplit-FGen[j].Daya;
if LoadSplit<0 then
begin
LoadSplit:=LoadSplit+fGen[j].Daya;
Status[j]:=true;
end;
end;
end;
end;
end;
end
else if tes<0 then
begin
for j:=1 to fNgen do
begin
if Status[j] then
begin
if FGen[j].Daya=FGen[j].Pmin then
begin
Status[j]:=false;
LoadSplit:=LoadSplit-FGen[j].Daya;
if LoadSplit<0 then

```

```

begin
  LoadSplit:=LoadSplit+FGen[j].Daya;
  Status[j]:=true;
end;
end;
end;
end;
end;
SetLength(result,FNgen+1);
for i:=1 to FNgen do
begin
  result[i]:=0;
  if rChrom 1[i] then
  begin
    result[i]:=FGen[i].Daya;
  end;
end;
end;

function TObjFunc.HitungCostGen(const rPL:dArr2):dArr2;
var i,j:integer;
begin
  SetLength(result,FNgen+1,FNjam+1);
  for i:=1 to FNgen do
  begin
    for j:=1 to FNjam do
    begin
      result[i,j]:=FGen[i].GetBiaya(rPL[i,j]);
    end;
  end;
end;

function TObjFunc.HitungCostSUC(const rPL:dArr2):dArr2;
var i,j,init,tcold:integer;
begin
  SetLength(result,FNgen+1,FNjam+1);
  for i:=1 to FNgen do
  begin
    init:=FGen[i].InitSt;
    tcold:=FGen[i].Tdown+FGen[i].Tcold;
    for j:=1 to FNjam do
    begin
      result[i,j]:=0;
      if rPL[i,j]<>0 then
      begin
        if init>0 then
        begin
          init:=init+1;
        end
        else if init<0 then
        begin
          if abs(init)<=tcold then
          begin
            result[i,j]:=FGen[i].Sh;
          end;
        end;
      end;
    end;
  end;
end;

```

```

    end
    else
    begin
        result[i,j]:=FGen[i].Sc;
    end;
    init:=1;
end;
end
else if rPL[i,j]=0 then
begin
    if init>0 then
    begin
        init:=-1;
    end
    else if init<0 then
    begin
        init:=init-1;
    end;
end;
end;
end;
end;
end;

procedure TObjFunc.doHitungChrom(const rChrom:bArr2;
    var rCostTotal:double);
var i,j:integer;
    PLa:dArr1;
    PL, CostGen, CostSUC:dArr2;
    chrom1:bArr1;
begin
    SetLength(PL, FNgen+1, FNjam+1);
    SetLength(chrom1, FNgen+1);
    SetLength(PLa, FNgen+1);
    for i:=1 to FNjam do
    begin
        for j:=1 to FNgen do
        begin
            chrom1[j]:=rChrom[j,i];
        end;
        PLa:=HitungEcoDis(i, chrom1);
        for j:=1 to FNgen do
        begin
            PL[j,i]:=PLa[j];
        end;
    end;
    CostGen:=HitungCostGen(PL);
    CostSUC:=HitungCostSUC(PL);
    rCostTotal:=0;
    for i:=1 to FNjam do
    begin
        for j:=1 to FNgen do
        begin
            rCostTotal:=rCostTotal+CostGen[j,i]+CostSUC[j,i];
        end;
    end;
end;
end;

```

end;

```
procedure TObjFunc.doHitungChrom(const rChrom:bArr2;
    var rPL:dArr2;
    var rCostPerJam:dArr1;
    var rCostTotal:double);
var i,j:integer;
    PLa:dArr1;
    CostGen,CostSUC:dArr2;
    chrom1:bArr1;
begin
    SetLength(rPL,FNgen+1,FNjam+1);
    SetLength(rCostPerJam,FNjam+1);
    SetLength(chrom1,FNgen+1);
    SetLength(PLa,FNgen+1);
    for i:=1 to FNjam do
        begin
            for j:=1 to FNgen do
                begin
                    chrom1[j]:=rChrom[j,i];
                end;
            PLa:=HitungEcoDis(i,chrom1);
            for j:=1 to FNgen do
                begin
                    rPL[j,i]:=PLa[j];
                end;
            end;
            CostGen:=HitungCostGen(rPL);
            CostSUC:=HitungCostSUC(rPL);
            SetLength(rCostPerJam,FNjam+1);
            rCostTotal:=0;
            for i:=1 to FNjam do
                begin
                    rCostPerJam[i]:=0;
                    for j:=1 to FNgen do
                        begin
                            rCostPerJam[i]:=rCostPerJam[i]+CostGen[j,i]+CostSUC[j,i];
                        end;
                    rCostTotal:=rCostTotal+rCostPerJam[i];
                end;
            end;
        end;
end;
```

```
procedure TObjFunc.doHitungPLN(var rCostPerJam:dArr1;
    var rCostTotal:double);
var i,j:integer;
    CostGen,CostSUC:dArr2;
begin
    SetLength(rCostPerJam,FNjam+1);
    CostGen:=HitungCostGen(FPLN);
    CostSUC:=HitungCostSUC(FPLN);
    rCostTotal:=0;
    for i:=1 to FNjam do
        begin
            rCostPerJam[i]:=0;
            for j:=1 to FNgen do
```

```

begin
  rCostPerJam[i]:=rCostPerJam[i]+CostGen[j,i]+CostSUC[j,i];
end;
rCostTotal:=rCostTotal+rCostPerJam[i];
end;
end;

```

```

//data output
function TObjFunc.GetSortAFLC:iArr 1;
var i,j,tmp:integer;
    tmpAFLC:double;
begin
  SetLength(result,FNgen+1);
  for i:=1 to FNgen do
    begin
      result[i]:=i;
    end;
    for i:=1 to fNgen-1 do
      begin
        for j:=i to fNgen do
          begin
            if FAFLC[i]>FAFLC[j] then
              begin
                tmpAFLC:=FAFLC[i];
                FAFLC[i]:=FAFLC[j];
                FAFLC[j]:=tmpAFLC;
                tmp:=result[i];
                result[i]:=result[j];
                result[j]:=tmp;
              end;
            end;
          end;
        end;
      end;
    for i:=1 to fNgen do
      begin
        fAFLC[i]:=fGen[i].AFLC;
      end;
    end;
end;

```

```

function TObjFunc.GetSortChrom(const rRank:integer):bArr 1;
var i:integer;
begin
  SetLength(result,FNgen+1);
  for i:=1 to FNgen do
    begin
      result[i]:=false;
    end;
    for i:=1 to rRank do
      begin
        result[FSortAFLC[i]]:=true;
      end;
    end;
end;

```

```

function TObjFunc.FindAFLC(const ri:integer):integer;
var i:integer;
begin

```

```

result:=1;
for i:=1 to FNgen do
begin
  if FSortAFLC[i]=ri then
  begin
    result:=i;
    break;
  end;
end;
end;

```

```

procedure TObjFunc.RepairAFLC(var rChrom:bArr2);
var i,j,pos,k:integer;
begin
  for i:=1 to FNjam do
  begin
    for j:=1 to Ngen do
    begin
      if rChrom[j,i]=true then
      begin
        pos:=FindAFLC(j);
        for k:=1 to pos do
        begin
          if rChrom[FSortAFLC[k],i]=false then
          begin
            rChrom[FSortAFLC[k],i]:=true;
            rChrom[j,i]:=false;
            break;
          end;
        end;
      end;
    end;
  end;
end;
end;

```

```

function TObjFunc.CreateChromBase:bArr2;
var i,j,k:integer;
    chrom1:bArr1;
    serve:boolean;
begin
  SetLength(result,FNgen+1,FNjam+1);
  SetLength(chrom1,FNgen+1);
  for i:=1 to FNjam do
  begin
    for j:=1 to FNgen do
    begin
      chrom1:=GetSortChrom(j);
      serve:=isServe(i,chrom1);
      if serve=true then
      begin
        for k:=1 to FNgen do
        begin
          result[k,i]:=chrom1[k];
        end;
        break;
      end;
    end;
  end;
end;

```



```
end;  
end;  
end;  
end;
```

```
function TObjFunc.CreateChromONOFF:bArr2;
```

```
var i,j,k:integer;  
    chrom1:bArr1;  
    serve:boolean;  
begin  
    SetLength(result,FNgen+1,FNjam+1);  
    SetLength(chrom1,FNgen+1);  
    for i:=1 to FNjam do  
    begin  
        for j:=FNgen downto 1 do  
        begin  
            chrom1:=GetSortChrom(j);  
            serve:=isServe(i,chrom1);  
            if serve=true then  
            begin  
                for k:=1 to FNgen do  
                begin  
                    result[k,i]:=chrom1[k];  
                end;  
                break;  
            end;  
        end;  
    end;  
end;  
end;
```

```
function TObjFunc.getRandomChrom(const rFlip:double):bArr2;
```

```
var i,j:integer;  
    chromBase,chromONOFF:bArr2;  
begin  
    chromBase:=CreateChromBase;  
    chromONOFF:=CreateChromONOFF;  
    SetLength(result,Ngen+1,Njam+1);  
    for i:=1 to Ngen do  
    begin  
        for j:=1 to Njam do  
        begin  
            if chromBase[i,j]=true then  
            begin  
                result[i,j]:=true;  
            end  
            else  
            begin  
                if chromONOFF[i,j]=true then  
                begin  
                    if isON(rFlip)=true then  
                    begin  
                        result[i,j]:=true;  
                    end  
                    else  
                    begin
```

```

        result[i,j]:=false;
    end;
end;
end;
end;
end;
RepairAFLC(result);
result:=GetSwap(result);
end;

```

```

function TObjFunc.getConstructSolution(const rFlip:double):bArr2;
var i,j:integer;
    chromBase,chromONOFF:bArr2;
begin
    chromBase:=CreateChromBase;
    chromONOFF:=CreateChromONOFF;
    SetLength(result,Ngen+1,Njam+1);
    for i:=1 to Ngen do
    begin
        for j:=1 to Njam do
        begin
            if chromBase[i,j]=true then
            begin
                result[i,j]:=true;
            end
            else
            begin
                if chromONOFF[i,j]=true then
                begin
                    if isON(rFlip)=true then
                    begin
                        result[i,j]:=true;
                    end
                    else
                    begin
                        result[i,j]:=false;
                    end;
                end;
            end;
        end;
    end;
end;
RepairAFLC(result);
result:=GetSwap(result);
end;

```

```

procedure TObjFunc.setLocalSearch(var rChrom:bArr2);
var i,j:integer;
    CostBest,CostCek:double;
    greyChrom:bArr2;
begin
    doHitungChrom(rChrom,CostBest);
    greyChrom:=doCariGreyZone(rChrom);
    for i:=1 to FNgen do
    begin
        for j:=1 to FNjam do

```

```

begin
  if greyChrom[i,j]=true then
    begin
      rChrom[i,j-1]:=true;
      doHitungChrom(rChrom, CostCek);
      if CostBest > CostCek then
        begin
          CostBest:=CostCek;
        end
      else
        begin
          rChrom[i,j-1]:=false;
        end;
      end;
    end;
  end;
end;

```

```

function TObjFunc.GetSwap(const rChrom:bArr2):bArr2;
var i,j,k,init,pos:integer;
begin
  SetLength(result, FNgen+1, FNjam+1);
  for i:=1 to FNgen do
    begin
      init:=FGen[i].InitSt;
      for j:=1 to FNjam do
        begin
          result[i,j]:=rChrom[i,j];
          if result[i,j]=true then
            begin
              if init<0 then
                begin
                  if abs(init)>=FGen[i].Tdown then
                    begin
                      init:=1;
                    end
                  else
                    begin
                      pos:=j+init;
                      if pos<1 then
                        begin
                          pos:=1;
                        end;
                      for k:=pos to j-1 do
                        begin
                          result[i,k]:=true;
                        end;
                      end;
                    end
                  else if init>0 then
                    begin
                      init:=init+1;
                    end;
                end
              else if result[i,j]=false then

```

```

begin
  if init<0 then
    begin
      init:=init-1;
    end
  else if init>0 then
    begin
      if init>=FGen[i].Tup then
        begin
          init:=-1;
        end
      else
        begin
          result[i,j]:=true;
          init:=init+1;
        end;
      end;
    end;
  end;
end;
end;
end;

```

```

//destructor
destructor TObjFunc.Destroy;
var i:integer;
begin
  try
    for i:=1 to FNgen do
      begin
        FGen[i].Free;
      end;
    finally
      inherited Destroy;
    end;
  end;
end;

```

```

function TObjFunc.GantiChrom(const rChrom:bArr2):bArr2;
var i,j:integer;
begin
  SetLength(result,FNgen+1,FNjam+1);
  for i:=1 to FNgen do
    begin
      for j:=1 to FNjam do
        begin
          result[i,j]:=rChrom[i,j];
        end;
      end;
    end;
  end;
end;

```

```

function TObjFunc.doCariGreyZone(const rChrom:bArr2):bArr2;
var i,j,init,tcold:integer;
begin
  SetLength(result,FNgen+1,FNjam+1);
  for i:=1 to FNgen do
    begin

```

```

for j:=1 to FNjam do
begin
  result[i,j]:=false;
end;
end;
for i:=1 to FNgen do
begin
  init:=FGen[i].InitSt;
  tcold:=FGen[i].Tdown+FGen[i].Tcold;
  for j:=1 to FNjam do
  begin
    if rChrom[i,j]=true then
    begin
      if init<0 then
      begin
        if abs(init)=(tcold+1) then
        begin
          result[i,j]:=true;
        end;
        init:=1;
      end
      else if init>0 then
      begin
        init:=init+1;
      end;
    end
    else if rChrom[i,j]=false then
    begin
      if init<0 then
      begin
        init:=init-1;
      end
      else if init>0 then
      begin
        init:=-1;
      end;
    end;
  end;
end;
end;

```

```

procedure TObjFunc.doExecute(var rChrom:bArr2;
  var rPL:dArr2;
  var rCostPerJam:dArr1;
  var rCostTotal:double);
var i,j:integer;
  CostBest,CostCek:double;
  chromBase,greYChrom:bArr2;
begin
  chromBase:=CreateChromBase;
  rChrom:=GetSwap(chromBase);
  doHitungChrom(rChrom,CostBest);
  greYChrom:=doCariGreyZone(rChrom);
  for i:=1 to FNgen do
  begin

```

```
for j:=1 to FNjam do
begin
  if greyChrom[i,j]=true then
  begin
    rChrom[i,j-1]:=true;
    doHitungChrom(rChrom, CostCek);
    if CostBest > CostCek then
    begin
      CostBest:=CostCek;
    end
    else
    begin
      rChrom[i,j-1]:=false;
    end;
  end;
end;
doHitungChrom(rChrom, rPL, rCostPerJam, rCostTotal);
end;

end.
```

## DAFTAR RIWAYAT HIDUP

1. N a m a : Rosmansyah
2. Tempat /tanggal lahir : Kotabaru, 19 September 1981
3. Alamat Asal : JL. Veteran Gang Karya Sari RT 06 No 77  
Kotabaru Kalimantan Selatan
4. Alamat Di Malang : JL. Raya karanglo No 2 Singosari
5. No telepon : (0341) 450177, (0341)7000149
6. Jenis kelamin : Laki-laki
7. Agama : Islam
8. Nama Ayah : Alm.H.Muhammad Rusmi
9. Nama Ibu : Hj. Kursiah
10. Kebangsaan : Indonesia
11. Kegemaran : Olahraga, Makan,main games.
12. Pendidikan :
  - TK Muhammadiyah Kotabaru ( lulus tahun 1988 )
  - SDN Baharu Utara 1 Kotabaru ( lulus tahun 1994 )
  - SMPN 5 Kotabaru ( lulus tahun 1997 )
  - SMUN 1 Kotabaru ( lulus tahun 2000 )
  - Sarjana Teknik (ST) Elektro Konsentrasi Energi Listrik  
ITN Malang (lulus tahun 2005)

# LAMPIRAN

4