

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



**PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL
DENGAN MENGGUNAKAN KOMBINASI METODE
EVOLUTIONARY PROGRAMMING DAN LAGRANGE
RELAXATION PADA PT. PEMBANGKITAN JAWA BALI**

SKRIPSI



Disusun Oleh :

INA SUNARYANTININGSIH

NIM. 99.12.094

APRIL 2005

MAKASUD DAN TUJUAN
KEMENTERIAN PERKULIAHAN
SARJANA
KEMENTERIAN PERKULIAHAN

REVISI
KEMENTERIAN PERKULIAHAN
SARJANA
KEMENTERIAN PERKULIAHAN
SARJANA

REVISI

Revisi

KEMENTERIAN PERKULIAHAN

REVISI

REVISI

LEMBAR PERSETUJUAN

**PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL
DENGAN MENGGUNAKAN KOMBINASI METODE
EVOLUTIONARY PROGRAMMING dan *LAGRANGE
RELAXATION* PADA PT. PEMBANGKITAN JAWA BALI**


SKRIPSI

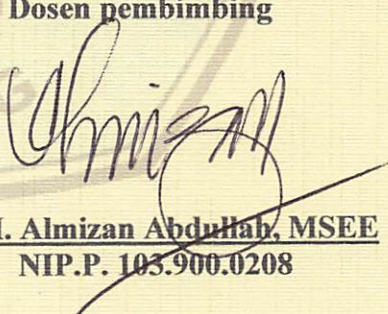
*Disusun Guna Melengkapi dan Memenuhi Syarat-Syarat
Guna Mencapai Gelar Sarjana Teknik*

**Disusun Oleh :
INA SUNARYANTININGSIH
NIM. 99.12.094**

**Mengetahui,
Ketua Jurusan Teknik Elektro**

**Diperiksa dan Disetujui,
Dosen pembimbing**


Ir. F. Yudi Limpraptono, MT
NIP. Y. 130.950.0274


Ir. H. Almizan Abdullah, MSEE
NIP.P. 103.900.0208

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

LEMBAR PERSEMBAHANLEMBAR PERSEMBAHAN

I'm nothing..... **GOD IS EVERYTHING**

But, the day of GOD will come like a thief
The heaven will disappear with a Roar
the elements will be destroyed by Fire
and earth and Everyting in it will be laid bare
GOD IS EVERYTHING

Get wisdom, get understanding
Get wisdom, get understanding

My friend,
keep yuor love with all diligence
for out of it springs the issues of life
put away from you adeceitful mouth
and put preserved lips far from you
let your eyes look straight ahead
and your eye lids look right before you
ponder the path of your feet
and let your ways be established
do not turn to the right or the left
remote your feet from evil

Mi dosa, mi celo
pero hasta a hora lejus...
No peuden ser el mios.....



Na SUNARYANTININGSIH

Tersesat.....
Mandikan aku ribuan tempaan keras
hujani aku pilihan hidup tersulit
tantanglah aku, lawanlah aku
dikumparan waktu yang paling terpuruk
terkuaklah seluruh tabir penyekat nurani
aku kan tetap melihat dengan mata yang
bijaksana
.....Hening
disitukah kau, wahai putri hujanku?
Hembuskanlah rona kasihmu yang tulus
tolonglah aku, tenangkanlah diriku
janganlah berpaling melihat kerapuhan ini
jinakkanlah hati lierku yang terhempas
di cakrawala tak terbatas.....

Trima
untukmu:

Bundaku Tersayang HARTATI,

Takkan cukup semua kemewahan di dunia ini untuk menganti setiap tetes air mata dan peluhmu untukku, Kalau ada dayaku akan kuberikan semua kebahagiaan untukmu demi melihat seutas senyuman dibibirmu. I LOVE YOU MOM.....

Ayahandaku ACHMAD SUHADI Alm, Aku percaya cinta kasihmu tercurah padaku, doa dan restumu slalu kau ucapkan untukku walau tanpa wujud dan warna, walau aku tak dapat memeluk jasadmu, tapi aku menyimpan kasihmu didalam hatiku SELALU.

I LOVE YOU PAP.....

ALWAYS

kasih

To My

Brother & Big Sister:

HADHI CAHYO WIBOWO,
NUR DWI SUKARUNIAWATI,
TYAS,

ACHMADI AGUNG SUTEJO: HAHHA..... AKU LULUS GA RUGI KAN,
THANKS ATAS OMELAN" YG SLALU MENYAKITKAN KUPINGKU.....,

TO DUA JAGOAN DI KELUARGA A.S : PRATAMA RIZKY
CAHYANTO, MUHAMAD FAIZ MAULANA IBRAHIM
JADI ANAK YANG SHOLEH YA.....

INA SAYANG KALIAN SEMUA
CUP-CUP MUAH

Big

Keluarg

Malang City:

Mbah Uti, Om"ku, Tante"ku,
Wiwik,
Dek Yuyun& dek AAN, belajar yang rajin ya.....+ Bandel dikit ga papa.....end semua sepupuku yang telah mengajariku banyak Hal.

Keluarga Besar Madiun: Mbah Favotitku "mbah YAS",
selalu ada dalam hatiku, ga ada mbah yang
segaul dikau

Om" & Tante", Paklek & Bulek dari
keluarga Madiun, Matur nuwun
sanget kagem
sedanten ipun.

a Besar

Special: Om Tatok & Tante

ELEKTRO B-INSPIRATION

Thanks to:

Bapak H. ALMIZAN ABDULLAH, MSEE

semua akan mengakui bahwa andalah yang terbaik disini, trima kasih telah menjadi guru yang konsisten dalam membimbing kami.

Dosen"ku

Terima kasih telah mengantarkan kami sampai dapat meraih gelar ini, maafkan bila ada salah, semoga Allah SWT membalas kebaikan bapak-ibu.

Konco" ST 3-4ku

Menjadi salah satu dari kalian adalah salah satu keberuntungan, kuliah bareng, nyontek bareng, bandel bareng, taruhan juga bareng, nimba ilmu yang kadang kita ga ngerti sama sekali tapi kita tetep semangat untuk kuliah. Ga popo sing penting ndang lulus cek ITN ga sesek.....

Geng gendengku

Eko, kalo ada tangan kanan kau tangan kiriku, matur suwun engkang ageng kokoPrabu. ARDI, cool men jangan mudah menyerah untuk hal yang lebih baik, kuncinya hanya satu yaitu istiQomah.

BOY, Tenang boy masih banyak cewek" cantik yang mau ama Lo, semangt boy jangan mau kalah ama cewek.

DANDUNG, waduh makin endut kok makin seksi, ayo kuliah yang bener biar cepet nikah, punya anak, punya cucu trus mati.

TONO & WIDIA, idih makin mesra aja, moga" awet.

FITRI, ayo kepasar besar lagi, opo ga kemeng se mba.

Fans Club

ComitmentUnit

Economic Dispatch

Peramalan Beban

By Babe Almizan

Hari

joko

mansyah

andi

Dedhy

Herman

aang

Hanafi

Wahyu

bobby

gandhi

dedhy 99

Mas thamrin

ina

Robert

Lingga

reza

pandu

irfan

AlmizanFClub

Dari perbedaan karakter, sifat, latar belakang, judul skripsi tetapi tetep kompak....

DESING BY IN

ABSTRAKSI

PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL MENGUNAKAN METODE KOMBINASI *EVOLUTIONARY PROGRAMMING* DAN *LAGRANGE RELAXATION* PADA PT. PJB

**(Ina Sunaryantiningsih, Nim. 99.12.094, Teknik Elektro S-1/ Teknik Energi
Listrik)**

(Dosen Pembimbing: Ir. H. Almizan Abdullah, MSEE.)

Kata Kunci: *Komitmen Unit, Ekonomi Dispatch, Evolutionary Programming,
Lagrange Relaxation, Optimasi biaya pembangkitan.*

Adanya persoalan dalam menghadapi kebutuhan 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 pada suatu operasi pada beban tertentu, perhitungan ekonomis harus tetap merupakan prioritas atau suatu nilai yang harus diperhitungkan disamping hal-hal lain sehingga nantinya diperlukan suatu rencana operasi yang optimum dengan tetap memenuhi beberapa persyaratan pengoperasian sistem tenaga listrik yaitu antara lain: Daya yang dibangkitkan cukup untuk memasok beban dan cadangan berputar (*Spinning Reserve*) pada sistem listrik Jawa-Bali.

Skripsi ini menganalisa permasalahan komitmen unit atau penentuan penjadwalan operasi unit-unit pembangkit dalam melayani beban sistem selama periode waktu tertentu dengan menggunakan metode kombinasi *Evolutionary Programming* dan *Lagrange Relaxation*, 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 data pembebanan tiap jam tiap unit pembangkit. Sedangkan hasil akhir dari program ini yaitu hasil perhitungan pembebanan ekonomis (*Economic Dispatch*), grafik biaya antara PT. PJB dan komitmen unit dengan kombinasi metode EP dan LR serta total biaya pembangkitan yang paling optimal.

Analisa dilakukan dengan bantuan program komputer, dengan menggunakan bahasa pemrograman Delphi versi 0.7 dan selesai dicoba pada PT. PJB. Yang terdiri dari 38 unit pembangkit, dimama telah berhasil dilakukan penghematan biaya pembangkitan untuk hari kamis, 4 Desember 2003 dengan kombinasi metode EP dan LR sebesar Rp. 3.987.765.201 Untuk tanggal 6 Desember 2003 sebesar Rp. 2.793.234.713 untuk tanggal 7 Desember 2003 sebesar Rp. 3.138.265.995 dengan waktu komputasi 1.31 detik.

KATA PENGANTAR

Dengan memanjatkan puji syukur kehadiran Allah S.W.T atas karunia dan hidayah-Nya skripsi ini dapat terselesaikan dengan baik, yang diajukan sebagai syarat untuk menyelesaikan studi di Fakultas Teknologi Industri Jurusan Teknik Eektro Konsentrasi Energi Listrik, Institut Teknologi Nasional Malang.

Pada kesempatan yang berbahagia ini, penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada :

1. Bapak Ir. F. Yudi Limpraptono MT, selaku ketua jurusan Teknik Elektro Konsentrasi Energi Listrik.
2. Bapak Ir. Almizan Abdullah, MSEE, selaku dosen pembimbing atas kesediaannya memberikan bimbingan, saran dan masukan dalam skripsi ini.
3. Bapak Yanto pada PT. Pembangkitan Jawa Bali, atas kesediaannya memberikan data yang ada.
4. Semua pihak yang telah membantu selama penulisan skripsi ini yang tidak dapat saya sebutkan satu persatu.

Penulis menyadari bahwa isi skripsi ini kemungkinan masih jauh dari sempurna. Untuk itu saran dan kritik yang membangun penulis harapkan.

Semoga skripsi ini dapat bermanfaat bagi penyusun maupun pembaca pada umumnya.

Malang, September 2004

Penulis

DAFTAR ISI

	Halaman
HALAMAN JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAKSI	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	x
DAFTAR GRAFIK	xii
BAB I PENDAHULUAN	
1.1. Latar Belakang	1
1.2. Rumusan Masalah	3
1.3. Tujuan Pembatasan	3
1.4. Batasan Masalah	4
1.5. Metodologi Pembahasan	4
1.6. Sistematika Pembahasan	6
1.7. Kontribusi	8

**BAB II TEORI KOMITMEN UNIT, *ECONOMIC DISPATCH*,
EVOLUTIONARY PROGRAMMING DAN *LAGRANGE
RELAXATION***

2.1. Sistem tenaga Listrik	7
2.2. Karakteristik Pembangkit Listrik	9
2.2.1. Karakteristik Masukan Keluaran.....	10
2.2.2. Karakteristik Laju Tambahan Biaya Bahan Bakar	12
2.3. Pembebanan Ekonomis pembangkit listrik	13
2.3.1. Penyelesaian <i>Economic Dispatch</i> Dengan Metode Pengali <i>Lagrange</i>	13
2.4. Komitmen Unit.....	16
2.4.1 Kendala Pada Komitmen Unit	17
2.4.2. Biaya Start	19
2.4.3. Fungsi Biaya Bahan Bakar	20
2.5. Metode <i>Lagrange Relaxation</i> dalam Penyelesaian Komitmen Unit	20
2.5.1. Fungsi Obyektif dan Constraint Pembangkit Tenaga Listrik.....	21
2.5.2. Prosedur <i>Lagrange Relaxation</i>	23
2.5.3. Pengaturan Pengali <i>Lagrange</i>	26
2.5.4. Relative Duality Gap	27
2.6. Prosedur <i>Evolutionay Programming</i>	28
2.6.1. Parameter <i>Evolutionary Programming</i>	30
2.6.2 Mekanisme <i>Evolutionary Programming</i>	31

2.7. Aplikasi Metode <i>Evolutionary Programming dan Lagrange Relaxation</i> dalam penyelesaian Komitmen Unit	35
2.7.1. Algoritma Penyelesaian kombinasi metode EP dan LR untuk masalah Komitmen Unit.....	37

**BAB III DATA SISTEM PEMBANGKIT TERMAL PADA PT.
PEMBANGKITAN JAWA BALI**

3.1. Pendaluan	42
3.2. Data Pembangkit Termal.....	43
3.3. Aplikasi Kombinasi Metode <i>Evolutionary Programming dan Lagrange Relaxation</i> Pada PT. Pembangkitan Jawa Bali.....	45
3.4. Beban Sistem.....	53
3.5. Validasi Program.....	53

**BAB IV SIMULASI DAN ANALISA KOMITMEN UNIT
MENGUNAKAN KOMBINASI METODE *EVOLUTIONARY
PROGRAMMING DAN LAGRANGE RELAXATION* PADA PT.
PEMBANGKITAN JAWA BALI**

4.1. Program Komputer Untuk Menyelesaikan Komitmen Unit	51
4.2. Algoritma Program	51
4.3 Flow Chart Program	56
4.4. Uji Validasi.....	61
4.5. Hasil Perhitungan dan Analisa Data	61

BAB V PENUTUP

5.1. Kesimpulan 83

5.2. Saran 84

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Elemen Pokok Sistem Tenaga Listrik	8
Gambar 2.2. Unit Boiler, Turbin dan Generator	10
Gambar 2.3. Kurva Karakteristik <i>Input-Output</i> Pembangkit Termal.....	11
Gambar 2.4. Kurva Karakteristik Laju Tambahan Biaya Bahan Bakar.....	12
Gambar 2.5. N Unit Pembangkit Termal Melayani Beban	14
Gambar 2.6. Two State Unit Pembangkit	25
Gambar 2.7. <i>Flow chart</i> Metode LR Update EP	41
Gambar 4.1. Flowchart Program EPLR	56
Gambar 4.2a. Tampilan Program	63
Gambar 4.2.b. Tampilan <i>Input</i> Data dan Jumlah Unit Pembangkit dan Jam	63
Gambar 4.3. Tampilan parameter EPLR untuk Validasi	63
Gambar 4.4 Tampilan status on-off untuk Uji Validasi	63
Gambar 4.5 Tampilan Perhitungan pembebanan untuk Uji Validasi.....	64
Gambar 4.6 Hasil validasi untuk biaya	64
Gambar 4.7. Hasil Optimasi Pada Tanggal 4 Desember 2003	75
Gambar 4.8. Hasil Optimasi Pada Tanggal 6 Desember 2003	75
Gambar 4.9. Hasil Optimasi Pada Tanggal 7 Desember 2003	76

DAFTAR TABEL

	Halaman
Tabel 3.1. Data Unit Termal Pada PT. Pembangkitan jawa Bali	43
Tabel 3.2. Data Biaya dan Parameter Unit Termal Pada PT. Pembangkitan jawa Bali	44
Tabel 3.3. Data Beban Unit Pembangkit Termal Pada PT. Pembangkitan Jawa Bali.	48
Tabel 3.4 Data Unit Comitment 10 Unit Untuk Validasi.....	49
Tabel 3.5 Data Beban Sistem Untuk Validasi	50
Tabel 4.1 Data Unit Comitment validasi.....	61
Tabel 4.2 Data beban untuk Validasi	62
Tabel 4.3. Kombinasi Unit Pembangkit Pada PT. Pembangkitan Jawa Bali Pada Hari Kamis, 4 desember 2003	68
Tabel 4.4. Kombinasi Unit Pembangkit Pada PT. Pembangkitan Jawa Bali Pada Hari Sabtu, 6 desember 2003	69
Tabel 4.5. Kombinasi Unit Pembangkit Pada PT. Pembangkitan Jawa Bali Pada Hari Minggu, 7 desember 2003	70
Tabel 4.6. Kombinasi Unit Pembangkit Pada PT. Pembangkitan Jawa Bali Dengan Kombinasi Metode EP dan LR Pada Hari Kamis, 4 desember 2003	71
Tabel 4.7. Kombinasi Unit Pembangkit Pada PT. Pembangkitan	

	Jawa Bali Dengan Kombinasi Metode EP dan LR	
	Pada Hari Sabtu, 6 desember 2003	72
Tabel 4.8.	Kombinasi Unit Pembangkit Pada PT. Pembangkitan	
	Jawa Bali Dengan Kombinasi Metode EP dan LR	
	Pada Hari Minggu, 7 desember 2003	73
Tabel 4.9.	Perbandingan Biaya Tiap jam Pada PT. Pembangkitan	
	Jawa Bali Dengan Metode EP dan LR	
	Pada Hari Kamis, 4 desember 2003	77
Tabel 4.10.	Perbandingan Biaya Tiap jam Pada PT. Pembangkitan	
	Jawa Bali Dengan Kombinasi Metode EP dan LR	
	Pada Hari Sabtu, 6 desember 2003	78
Tabel 4.11.	Perbandingan Biaya Tiap jam Pada PT. Pembangkitan	
	Jawa Bali Dengan Metode	
	Pada Hari Minggu, 7 desember 2003	79
Tabel 4.12.	Perbandingan Biaya Total Pada PT. Pembangkitan	
	Jawa Bali Dengan Kombinasi Metode EP dan LR	82
Tabel 5.1	Selisih hasil perhitungan biaya PT.PJB dan	
	kombinasi metode EP dan LR.....	84

DAFTAR GRAFIK

	Halaman
Grafik 4.1. Kurva Fungsi Pembebanan Terhadap Biaya Pada PT. Pembangkitan Jawa Bali Dengan Kombinasi Metode <i>Evolutionary Programming</i> dan <i>Lagrange Relaxation</i> Pada Hari Kamis, 4 desember 2003	80
Grafik 4.2. Kurva Fungsi Pembebanan Terhadap Biaya Pada PT. Pembangkitan Jawa Bali Dengan Kombinasi Metode <i>Evolutionary Programming</i> dan <i>Lagrange Relaxation</i> Pada Hari Kamis, 6 desember 2003	80
Grafik 4.3. Kurva Fungsi Pembebanan Terhadap Biaya Pada PT. Pembangkitan Jawa Bali Dengan Kombinasi Metode <i>Evolutionary Programming</i> dan <i>Lagrange Relaxation</i> Pada Hari Kamis, 7 desember 2003	81

BAB I

PENDAHULUAN

1.1 . Latar Belakang

Permasalahan yang harus dijawab dalam merencanakan suatu pengembangan jangka panjang sistem pembangkit tenaga listrik adalah bagaimana suatu investasi yang optimum dapat ditentukan untuk menghadapi kebutuhan pertambahan beban yang meningkat tajam dengan kebutuhan biaya yang sangat besar menghadapi berbagai kendala baik bersifat teknis-ekonomis maupun yang berupa keterbatasan sumber daya energi. Sedangkan tenaga listrik tidak dapat disimpan dalam bentuk tenaga listrik, karenanya harus disediakan pada saat dibutuhkan. Akibatnya timbul persoalan dalam menghadapi kebutuhan daya listrik yang tidak tetap dari waktu ke waktu. Bagaimana mengoperasikan suatu sistem tenaga listrik yang selalu dapat memenuhi kebutuhan daya pada setiap saat, dengan kualitas yang baik dan harga yang murah. Apabila daya yang dikirim dari bus-bus pembangkit jauh lebih besar dari pada permintaan daya pada bus-bus beban, maka akan timbul persoalan peningkatan frekwensi sistem dan pemborosan energi pada perusahaan listrik, terutama pada pembangkit termal. Sedangkan apabila daya yang dibangkitkan dan dikirimkan lebih rendah atau tidak memenuhi kebutuhan beban konsumen, maka akan terjadi penurunan frekwensi sitem dan pemadaman lokal pada bus-bus beban,yang akibatnya merugikan pihak konsumen. Oleh karena itu, diperlukan penyesuaian antara pembangkitan dengan permintaan daya setiap saat.

Banyak sekali kemungkinan atau alternatif gabungan (kombinasi) pembangkitan tenaga listrik yang dapat diikuti sertakan dalam perencanaan jangka panjang, karena sistem tenaga listrik terdiri dari sejumlah unit pembangkit yang saling terhubung atau interkoneksi. Penentuan penjadwalan pembangkit adalah faktor penting karena ini berhubungan dengan dengan biaya operasional pembangkit dan memegang peranan penting dalam perencanaan operasional pembangkitan sehingga didapat operasi pembangkitan yang optimum dalam sistem tenaga listrik secara keseluruhan. Perencanaan operasional pembangkitan sistem tenaga listrik sangat erat kaitannya dengan masalah komitmen unit, yaitu masalah pemilihan unit-unit pembangkit yang akan dioperasikan untuk memenuhi kebutuhan beban listrik. Komitmen unit ini nantinya akan berpengaruh pada biaya bahan bakar. Oleh karena itu, perlu diupayakan pengoptimalan operasi unit pembangkit agar dicapai biaya operasional pembangkitan yang minimum.

Ada banyak metode yang digunakan untuk memecahkan masalah komitmen unit, diantaranya: *metode Ekstensif Enumeration, priority List (skala prioritas)* , *dynamic programming, Lagrange Relaxation, Genetic Algorithm, Evolutionary programming*. Dalam skripsi ini akan dibahas suatu metode kombinasi dalam menyelesaikan masalah komitmen unit, yaitu kombinasi metode *Evolutionary programming (EP)* dan *Lagrange Relaxation*. Metode *Lagrange Relaxation* menyediakan solusi yang cepat tetapi tidak memberi solusi yang terbaik, selain itu mutu solusi LR tergantung dari metode yang memperbarui Pengali Lagrange, sehingga dalam skripsi ini dikaji metode yang mengkombinasikan *Lagrange Relaxation* dan *Evolutionary Programming*, dimana

EP digunakan untuk memperbaharui Pengali Lagrange untuk meningkatkan kualitas capaian LR.

1.2. Rumusan Masalah

Berdasarkan latar belakang diatas timbul pertanyaan seberapa besar kemampuan kombinasi metode *EP* dan *LR* dalam pemecahan masalah pemecahan komitmen unit sehingga dapat digunakan sebagai salah satu acuan dalam operasi pembangkit yang seekonomis mungkin serta untuk mengetahui kondisi sistem selama keadaan operasi. Sehubungan dengan itu maka judul skripsi ini adalah

**PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL
MENGUNAKAN KOMBINASI METODE EVOLUTIONARY
PROGRAMMING DAN LAGRANGE RELAXATION PADA
PT. PEMBANGKITAN JAWA BALI**

1.3. Tujuan

Berdasarkan pada permasalahan yang telah didiskripsikan diatas maka skripsi ini mempunyai tujuan membuat analisa dan mengevaluasi kesesuaian penerapan komitmen unit dengan kombinasi metode *EP* dan *LR* untuk mengoptimasi biaya operasional pembangkitan pada sistem tenaga listrik PT. Pembangkitan Jawa Bali, untuk mendapatkan suatu perencanaan dan penjadwalan pembangkitan yang optimal, ekonomis, efektif dan efisien .

1.4. Batasan Masalah

Agar pemecahan masalah mengarah pada tujuan, maka pembahasan skripsi ini dibatasi pada hal-hal berikut:

1. Analisa dilakukan dengan asumsi bahwa sistem berada dalam operasi normal.
2. Tidak membahas masalah rugi-rugi pada transmisi
3. Tidak membahas *combined cycle* pada PLTGU.
3. Tidak membahas masalah biaya cadangan berputar (*spinning reserve*), hanya memperhatikan kendala batasan cadangan berputar.

1.5. Metodologi Pembahasan

Metodologi yang digunakan dalam pembahasan skripsi ini adalah sebagai berikut:

1. Studi Literatur

Yaitu kajian pustaka yang mempelajari teori-teori yang terkait melalui literatur, yaitu teori tentang komitmen unit, pembebanan Ekonomis, algoritma, Evolutionary programming, Lagrange Relaxation.

2. Pengumpulan Data

Bentuk data yang digunakan:

- a. Data Kuantitatif, yaitu data yang dapat dihitung atau data yang berbentuk angka-angka, meliputi data harga penawaran, data pembebanan harian PT. Pembangkitan Jawa-Bali.

- b. Data Kualitatif, yaitu data yang berbentuk uraian, keterangan dan sebagainya , yang bisa juga diambil dari penelitian terdahulu.
- 3 Digunakan program komputer untuk melakukan hasil simulasi serta dilakukan pembahasan hasil simulasi, hasil simulasi digunakan untuk menentukan apakah metode yang digunakan dalam skripsi ini lebih ekonomis.

1.6. Sistematika Penulisan

Penulisan skripsi ini terdiri atas bab-bab dengan susunan sebagai berikut:

BAB I PENDAHULUAN

Terdiri dari latar belakang penulisan , rumusan masalah, maksud dan tujuan penulisan skripsi, ruang lingkup pembahasan, dan sistematika penulisan

BAB II TEORI DASAR

Penjelasan tentang teori dasar yang membahas Komitmen Unit, Economic Dispatch, Lagrange Relaxation dan Evolutionary Programming

BAB III Data sistem pembangkitan termal pada PT. Pembangkitan Jawa-Bali

BAB IV Simulasi Progran Untuk Pemecahan Masalah Komitmen Unit Dengan Menggunakan Pembebanan Ekonomis, Evolutionary Programming dan Lagrange Relaxation Di PT. Pembangkitan Jawa-Bali

BAB V PENUTUP

Berisikan tentang kesimpulan dari pembahasan skripsi dan saran

BAB II

TEORI KOMITMEN UNIT, *EKOMINIC DISPATCH, EVOLUTIONARY PROGRAMMING* DAN *LAGRANGE RELAXATION*

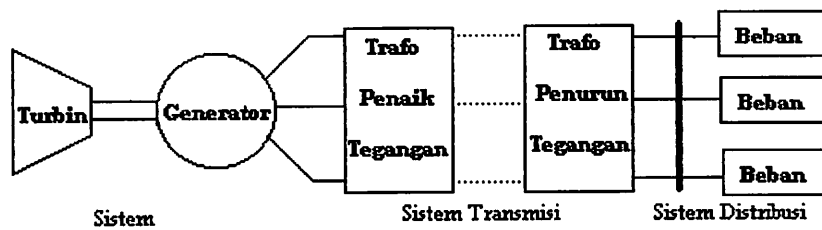
2.1 Sistem Tenaga Listrik ^{[1][3]}

Karena berbagai persoalan teknis, tenaga listrik hanya dapat dibangkitkan pada lokasi tertentu. Mengingat pemakai tenaga listrik atau pelanggan tenaga listrik tersebar diberbagai tempat, maka penyaluran tenaga listrik dari berbagai penanganan teknis.

Tenaga listrik dibangkitkan di pusat-pusat listrik seperti PLTA, PLTD, PLTU, PLTG dan PLTGU, kemudian disalurkan melalui saluran transmisi setelah tegangannya dinaikkan terlebih dahulu oleh Transformator Penaik Tegangan yang terdapat di pusat listrik.

Setelah tenaga listrik disalurkan melalui transmisi, maka sampailah tenaga listrik tersebut di Gardu Induk (GI) yang untuk kemudian tegangannya diturunkan oleh Trafo Penurun Tegangan menjadi tegangan menengah atau tegangan distribusi primer.

Jaringan keluar dari gardu induk umumnya disebut jaringan distribusi, sedangkan antara pusat listrik dengan gardu induk disebut jaringan transmisi. Setelah disalurkan melalui jaringan distribusi primer maka tenaga listrik kemudian diturunkan tegangannya oleh gardu distribusi menjadi tegangan distribusi sekunder atau tegangan rendah, dan baru kemudian disalurkan ke konsumen.



Gambar 2.1 Elemen Pokok Sistem Tenaga Listrik

Dari uraian diatas kiranya dapat mengerti bahwa besar kecilnya tenaga listrik ditentukan sepenuhnya oleh konsumen, yaitu tergantung bagaimana para konsumen akan menggunakan peralatan listriknya, kemudian PT. PLN (Persero) harus mengimbangi kebutuhan tenaga listrik tersebut dalam arti selalu menyesuaikan daya listrik yang dibangkitkan dari waktu ke waktu.

Untuk memenuhi kebutuhan tenaga listrik bagi para konsumen, diperlukan berbagai peralatan listrik. Berbagai peralatan listrik ini dihubungkan satu sama yang lain secara keseluruhan membentuk suatu sistem tenaga listrik. Sehingga yang disebut Sistem Tenaga Listrik disini adalah sekumpulan pusat listrik dan gardu induk yang satu dengan yang lain dihubungkan oleh jaringan transmisi sehingga merupakan sebuah kesatuan interkoneksi.

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

- a. Perkiraan beban.
- b. Syarat-syarat pemeliharaan peralatan.
- c. Keandalan yang diinginkan

- d. Pengaturan dan penyaluran beban
- e. Proses produksi tenaga listrik yang ekonomis

Kelima hal diatas masih harus sering dikaji ulang terhadap berbagai kendala seperti :

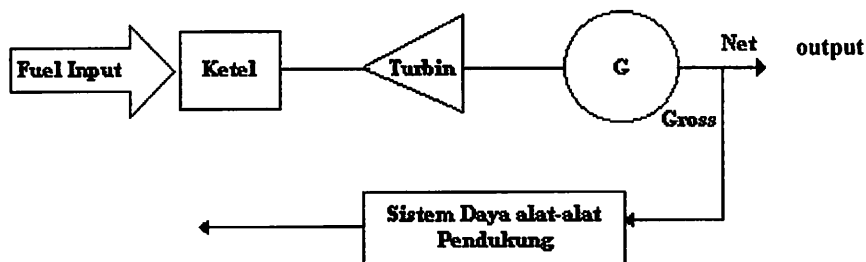
- a. Aliran beban dalam jaringan
- b. Daya hubung singkat dan gangguan yang sering menimpa peralatan
- c. Stabilitas sistem
- d. Penyediaan suku cadang dan dana

Dengan memperhatikan kendala-kendala diatas 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.

2.2 Karakteristik Pembangkit Listrik ^{[1][3]}

Hal yang paling mendasar dalam optimasi ekonomi dari sebuah pembangkit listrik 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 pembangkit listrik tersebut.

Model sebuah pembangkit listrik 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 dihasilkan tidak seluruhnya 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 – Turbin – Generator^[2]

2.2.1 Karakteristik Masukan – Keluaran^{[1][3]}

Masukan sebuah pembangkit listrik termal umumnya dinyatakan sebagai banyaknya energi per satuan waktu dari bahan bakar yang diberikan ke ketel untuk menghasilkan daya listrik yang merupakan keluaran dari pusat listrik tersebut. Terdapat dua notasi yang umum digunakan :

H dengan satuan [Mbtu/hour]

F dengan satuan [\$US/hour],

Dimana $F = H \times \$US \text{ Btu}$, dari $\$US/Btu$ menyatakan harga bahan bakar per satuan energi yang dikandung oleh bahan bakar tersebut.

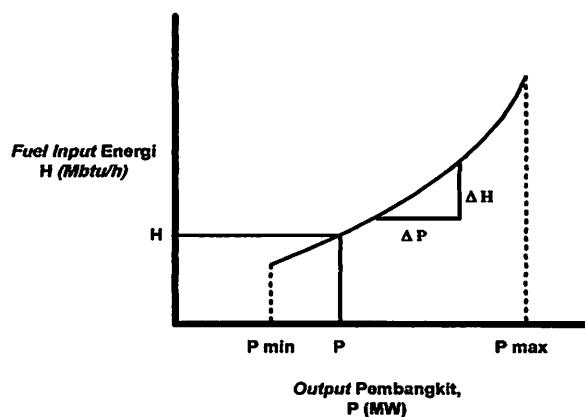
Sedangkan keluaran dari pembangkit listrik termal adalah daya nyata yang dihasilkan oleh generator dikurangi dengan daya nyata yang dipakai oleh pusat listrik tersebut. Notasi yang umum digunakan adalah :

P dengan satuan [MW]

Jadi dapat disimpulkan bahwa masukan pusat listrik merupakan fungsi terhadap keluarannya, maka hubungan tersebut dapat ditulis sebagai berikut :

$$H = f(P) \text{ [Mbtu/h] atau } F = f(P) \text{ [\$US/h]}$$

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



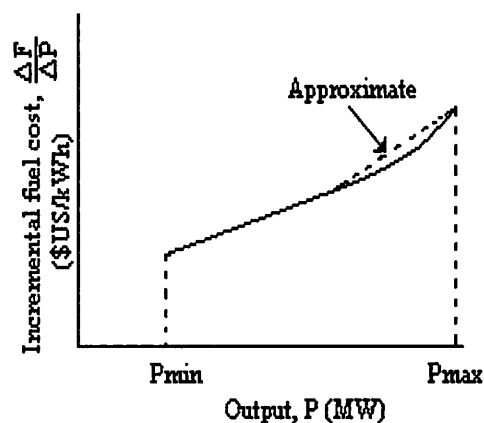
Gambar 2.3 Kurva Karakteristik Input – Output Pembangkit Termal [2]

Data yang diperlukan 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 ^{[1][3]}

Karakteristik laju tambahan biaya bahan bakar atau *Incremental Fuel Cost Characteristic* adalah turunan pertama dari fungsi biaya bahan bakar F [\$US/h] 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 ekonomis dari sebuah pembangkit listrik tenaga termal. Tampak pada gambar 2.4 adalah kurva laju tambahan biaya bahan bakar yang telah diidealkan dari sebuah pembangkit listrik termal.



Gambar 2.4 Kurva Karakteristik Laju Tambahan Biaya Bahan Bakar ^[1]

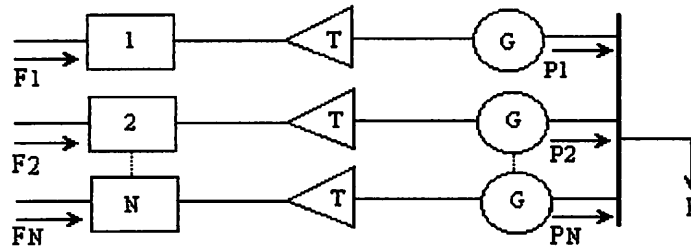
2.3 Pembebanan Ekonomis Pembangkit Listrik ^{[1][3]}

Pembebanan ekonomis atau *economic dispatch* adalah pembangian pembebanan pada pembangkit-pembangkit yang ada dalam sistem tenaga listrik, secara optimal dan ekonomis pada harga beban tertentu. Komponen terbesar dari biaya pembangkitan adalah biaya bahan bakar. Oleh sebab itu dengan dilakukannya *economic dispatch* berarti pula didapatkan biaya bahan bakar pembangkitan yang paling murah. Oleh karena beban yang harus ditanggung oleh sistem pembangkit selalu berubah setiap periode waktu tertentu, maka perhitungan *economic dispatch* ini dilakukan untuk setiap harga beban tertentu. Penyelesaian *economic dispatch* dapat dilakukan dengan beberapa cara yang akan dibahas pada sub bab di bawah ini.

2.3.1 Penyelesaian Economic Dispatch Dengan Metode Pengali Lagrange ^{[1][3]}

Sistem tenaga listrik dengan mengabaikan rugi-rugi transmisi dapat dilihat pada gambar 2.5. Sistem ini memperlihatkan pembangkit termal yang terdiri atas N buah unit yang dihubungkan pada sebuah bus bar untuk melayani total beban sebesar P_D . Masukan untuk setiap unit ke- i adalah F_i yang menyatakan tingkat biaya dari masing-masing unit, dan keluaran dari masing-masing unit P_i adalah daya listrik yang dibangkitkan oleh tiap-tiap unit.

Biaya total F_T yang ditanggung sistem adalah jumlah biaya dari tiap-tiap unit pembangkit. Dan batasan yang paling penting dari pengoperasian pembangkit termal tersebut adalah daya listrik yang dihasilkan harus sama dengan besarnya beban konsumen.



Gambar 2.5 N Unit Pembangkit termis Melayani Beban P_D ^[1]

Yang menjadi permasalahan adalah meminimumkan total biaya F_T dengan memperhatikan pembatas ϕ bahwa daya dihasilkan unit pembangkit sama dengan yang diterima beban. Secara matematis pernyataan tersebut diatas dapat dinyatakan dengan persamaan berikut :

$$\begin{aligned}
 F_T &= F_1 + F_2 + \dots + F_N \\
 &= \sum_{i=1}^N F_i(P_i) \dots\dots\dots (2.1)
 \end{aligned}$$

Dan daya listrik yang dihasilkan oleh setiap unit untuk melayani beban total adalah :

$$P_D = \sum_{i=1}^N P_i \dots\dots\dots (2.2)$$

$$P_D - \sum_{i=1}^N P_i = \phi = 0 \dots\dots\dots (2.3)$$

dimana P_D = kebutuhan beban, dan

P_i = jumlah daya yang dihasilkan

Penyelesaian permasalahan optimasi seperti ini dapat diselesaikan dengan metode yang menyangkut fungsi *lagrange* :

$$\mathcal{L} = F_T + \lambda\phi$$

atau

$$\mathcal{L} = \sum_{i=1}^N F_i(P_i) + \lambda (P_D - \sum_{i=1}^N P_i) \dots\dots\dots (2.4)$$

dimana $\lambda = \text{Lagrange Multipliers}$.

Untuk mencari harga optimal dari fungsi *lagrange* terhadap P_i , dapat diperoleh dengan operasi gradien dari persamaan *lagrange* sama dengan nol.

$$\nabla \mathcal{L} = 0$$

$$\frac{\partial \mathcal{L}}{\partial P_i} = \frac{\partial F_i}{\partial P_i} + \lambda \cdot \left(\frac{\partial P_D}{\partial P_i} - \frac{\partial P_i}{\partial P_i} \right) = 0 \quad \text{atau}$$

$$\frac{\partial F_i}{\partial P_i} + \lambda \cdot (0 - 1) = 0$$

$$\frac{\partial F_i}{\partial P_i} = \lambda \dots\dots\dots (2.5)$$

Persamaan diatas menunjukkan bahwa distribusi beban yang optimal terjadi apabila semua unit pembangkit beroperasi pada tingkat laju tambahan biaya bahan bakar yang sama, yang ternyata sama dengan nilai λ . Kondisi optimal ini tentunya memerlukan persamaan-persamaan pembatas (*constraint*) agar keluaran dari setiap unit pembangkit harus lebih besar atau sama dengan keluaran minimum, dan lebih kecil atau sama dengan keluaran maksimum yang diijinkan.

Dari N buah unit pembangkit yang telah dibahas, maka dapat diambil kesimpulan sebagai berikut :

$$\frac{\partial F_i}{\partial P_i} = \lambda \quad \text{ada } N \text{ buah persamaan}$$

$$P_{i \min} \leq P_i \leq P_{i \max} \quad \text{ada } 2N \text{ buah pertidaksamaan}$$

$$\sum_{i=1}^N P_i = P_D \quad \text{ada 1 buah pembatas}$$

Batasan-batasan diatas dapat diperluas menjadi :

$$\frac{\partial F_i}{\partial P_i} = \lambda \text{ untuk } P_{i \min} \leq P_i \leq P_{i \max}$$

$$\frac{\partial F_i}{\partial P_i} \leq \lambda \text{ untuk } P_i = P_{i \max}$$

$$\frac{\partial F_i}{\partial P_i} \geq \lambda \text{ untuk } P_i = P_{i \min}$$

2.4 Unit Commitment^{[1][3]}

Dalam suatu siklus waktu, misalnya siklus harian yang terbagi dalam interval waktu 1 jam selama 24 jam, beban listrik dalam sistem tenaga selalu berubah. Operasi pusat-pusat pembangkit di dalam sistem tenaga harus selalu dikoordinasikan dalam pembagian pembebanan secara optimal atau ekonomis pada setiap perubahan beban dalam interval waktu untuk siklus waktu tertentu. Oleh karena itu diperlukan penjadwalan operasi unit-unit pembangkit dalam melayani beban sistem selama periode waktu tertentu, atau dikenal dengan istilah *Unit Commitment*.

Pada masalah *unit commitment* diasumsikan ada sejumlah N unit pembangkit yang tersedia dan dioperasikan untuk memenuhi permintaan beban, sehingga dapat disimpulkan bahwa, bila tersedia N unit pembangkit yang diharapkan mencukupi permintaan beban, maka masalahnya adalah unit-unit pembangkit mana yang seharusnya dioperasikan dengan biaya operasi seekonomis mungkin.

Dalam skripsi ini akan dititik beratkan pada pengoptimalan pusat pembangkit termis saja, karena pada wilayah kerja PJB selain jumlah dan total

kapasitas dari unit hidro yang relatif kecil, unit hidro dalam pembangkitan dayanya tergantung dari SDA. Sedangkan unit termis tergantung dari pemakaian bahan bakar. Dalam pengoperasian pembangkit termis harus dicapai biaya operasi dan bahan bakar yang paling minimum, untuk itu komitmen unit merupakan metode tepat mengatasi permasalahan diatas guna mencari jadwal unit pembangkit yang harus beroperasi untuk periode tertentu agar dicapai biaya operasi yang seekonomis mungkin.

2.4.1 Kendala Pada Unit Commitment^{[1][3]}

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

a. Spinning Reserve

Merupakan cadangan daya yang harus diperhitungkan dari unit-unit pembangkit yang beroperasi, dimana apabila ada salah satu unit mengalami kegagalan operasi atau keluar dari sistem, harus ada cukup cadangan daya untuk mengganti atau menutupi berkurangnya suplai daya dalam periode waktu tertentu. Umumnya cadangan daya diperhitungkan untuk mampu mengganti apabila unit yang terbesar mengalami kegagalan operasi.

Spinning Reserve harus dialokasikan untuk menuruti aturan-aturan tertentu, biasanya aturan yang dipakai adalah bahwa cadangan tersebut berupa sebuah persentase yang diberikan terhadap beban puncak yang

diperkirakan, atau bahwa cadangan tersebut 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 memenuhi kegagalan pembangkitan unit, tetapi cadangan-cadangan tersebut harus dialokasikan diantara unit-unit yang dapat bereaksi cepat dan unit-unit yang bereaksi lambat. Ini memungkinkan sistem kontrol pembangkitan otomatis untuk mengembalikan suplai daya yang cepat pada saat unit pembangkit keluar dari sistem.

Besar cadangan berputar tersebut harus ditentukan secara hati-hati, sebab sering kali penentuan yang didasarkan untuk menjaga keandalan sistem berbenturan dengan biaya pengoperasian yang diusahakan seekonomis mungkin, misalnya jika cadangan berputar kecil dan unit yang terbesar mengalami gangguan dan trip unit keluar dari sistem secara mendadak, maka cadangan berputar tersebut tidak cukup untuk mengatasi kekurangan pembangkitan yang terjadi. Jika makin besar cadangan berputar dalam sistem maka makin handal sistem tersebut dalam menghadapi gangguan, tetapi semakin besar biaya operasi terutama biaya bahan bakarnya, oleh karena itu perlu adanya suatu kesepakatan antara pemenuhan keandalan dan pengoptimalan biaya operasi.

b. Kendala Unit Termal

Sebuah unit pembangkit termal dapat mengalami perubahan temperatur setahap demi setahap, dan ini diterjemahkan ke dalam sebuah

periode waktu dalam jam dimana dibutuhkan untuk membawa unit tersebut *on line*. Hal ini menyebabkan kendala-kendala antara lain :

1. Minimum Up Time

Minimum up time adalah interval waktu minimum dimana suatu unit yang dihidupkan (ON) tidak boleh dimatikan (OFF) kembali sebelum melewati batas waktunya (Up Time).

2. Minimum Down Time

Minimum down time adalah interval waktu minimum dimana suatu unit yang dimatikan (OFF) tidak boleh dihidupkan (ON) kembali sebelum melewati batas waktunya (Down Time).

2.4.2 Biaya Start ^[2]

Biaya start 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 yaitu :

a. Biaya start 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 pada kondisi panas (Hot Start)

Kondisi ini terjadi karena saat pembangkit dilepas dari sistem (tidak beroperasi), temperatur boiler tetap dijaga pada temperatur kerja.

2.4.3 Fungsi Biaya Bahan Bakar^[1]

Biaya bahan bakar merupakan unsur biaya yang penting dalam operasi sistem pembangkitan termal karena merupakan biaya terbesar dalam pembangkitan. 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 :

$$F_i(P_{it}) = a_i + b_i P_{it} + c_i P_{it}^2 \dots\dots\dots(2.6)$$

dimana : a_i, b_i, c_i = konstanta persamaan dari unit ke- i

P_{it} = daya keluaran dari unit ke- i pada jam t

2.5 Metode Lagrange Relaxation dalam Penyelesaian unit Commitment^{[1][3]}

Komitmen unitik adalah metode yang dipergunakan untuk menentukan kombinasi dari unit-unit pembangkit yang akan beroperasi tiap periode waktu dalam jam, dimana pembagian beban dari unit-unit pembangkit tersebut harus memenuhi kendala-kendala tertentu dan menghasilkan pembagian bebannya dalam keadaan optimal atau seekonomis mungkin. Dari jumlah unit pembangkit yang banyak, maka untuk menentukan unit mana yang harus beroperasi dan tidak beroperasi pada jam tertentu, dapat diperhitungkan dengan membuat kombinasi keadaan dari unit-unit yang ada dan berada pada keadaan operasi yang optimal, kemudian baru dipilih kombinasi mana yang termurah biaya operasinya.

Lagrange relaxation adalah suatu metode yang didasarkan pada pendekatan *dual optimization* (optimasi ganda) yang berhubungan dengan pengali *lagrange* (λ), yang dapat dipakai untuk menentukan kombinasi yang paling

optimal sehingga mendapatkan biaya operasi yang paling minimal. Pengali *lagrange* disini digunakan pada persamaan batasan pembebanan, dengan melalui proses subgradien, sehingga dapat memaksimumkan *dual value* (nilai ganda).

Dalam solusi *Lagrange relaxation* pada masalah *unit commitment*, digunakan bentuk penguraian yang berbeda untuk memudahkan dalam memutuskan komitmen dan jadwal pembangkitan tiap-tiap unit pembangkit dalam horizon perencanaan, kemudian tiap-tiap unit dapat dikombinasikan untuk melihat apakah sudah memenuhi batasan sistem.

2.5.1 Fungsi Obyektif dan Constraint Pembangkit Tenaga Listrik^{[1][3]}

Sasaran dari masalah *unit commitment* adalah meminimalkan total biaya operasi dalam penjadwalan unit pembangkit. Oleh karena itu, fungsi obyektif dinyatakan sebagai jumlah dari fungsi biaya bahan bakar dan biaya start up dari unit pembangkit. Dapat dinyatakan dengan :

$$F(P_{it}, U_{it}) = \sum_{t=1}^T \sum_{i=1}^N [F_i(P_{it}) + ST_i(1 - U_{i,t-1})] U_{it} \dots\dots\dots(2.7)$$

dimana : $F_i(P_{it})$ = fungsi biaya bahan bakar unit ke- i

ST_i = biaya start up dari unit ke- i

U_{it} = status on atau off dari unit ke- i pada jam t , $U_{it} = 0$ ketika unit off
dan $U_{it} = 1$ ketika unit on

N = jumlah unit

T = jumlah jam

Sehubungan dengan minimalisasi dari total biaya operasi, terdapat berbagai *constraint* atau syarat pembatas, sebagai berikut :

a. Batasan pembebanan

$$\sum_{i=1}^N P_{it} U_{it} = D_t \text{ untuk } t = 1 \dots T \dots\dots\dots (2.8)$$

dimana : D_t = kebutuhan beban pada jam t

b. Batasan cadangan berputar

$$\sum_{i=1}^N U_{it} P_i^{maks} \geq D_t + R_t \dots\dots\dots (2.9)$$

dimana : R_t = cadangan berputar pada jam t

c. Batasan pembangkitan

$$U_{it} P_i^{min} \leq P_{it} \leq U_{it} P_i^{maks} \text{ untuk } i = 1 \dots N \text{ dan } t = 1 \dots T \dots\dots\dots (2.10)$$

d. Minimum up time (up_i) dan minimum down time ($down$)

$$U_{it} = 1 \text{ untuk } \sum_{x=t-up_i}^{t-1} U_{ix} < T_i^{up}, \text{ dan } U_{it} = 0 \text{ untuk } \sum_{x=t-down_i}^{t-1} (1-U_{ix}) < T_i^{down} \dots (2.11)$$

Pada masalah optimasi total biaya operasi yang sehubungan dengan keputusan variabel-variabel U_{it} dan dibatasi dengan beberapa *constraint* diatas, dapat diselesaikan dengan penambahan *Lagrange multipliers* (λ_t, μ_t) untuk batasan pembebanan pada fungsi obyektif, dan membentuk fungsi *Lagrange* sebagai berikut :

$$\mathcal{L}(P, U, \lambda, \mu) = F(P_{it}, U_{it}) - \sum_{t=1}^T \lambda_t \left(\sum_{i=1}^N P_{it} U_{it} - D_t \right) - \sum_{t=1}^T \mu_t \left(\sum_{i=1}^N P_{i \max} U_{it} - D_t - R_t \right)$$

2.5.2 Prosedur Lagrange Relaxation ^[1]

Prosedur *Lagrange relaxation* dalam memecahkan masalah *unit commitment* adalah dengan “*relaxing*” atau pengabaian sementara batasan bersama dan menyelesaikannya dengan melakukan prosedur optimasi ganda. Prosedur tersebut dapat mencapai batasan yang optimum dengan memaksimalkan fungsi *Lagrange* (\mathcal{L}) dengan memperhatikan pengali *Lagrange* (λ, μ), sambil meminimalkan dengan memperhatikan variabel P_{it} dan U_{it} pada masalah *unit commitment*, sehingga didapatkan persamaan *dual value* (nilai ganda) sebagai berikut :

$$q^* = \text{maks } q(\lambda) \dots\dots\dots(2.13)$$

dimana

$$q(\lambda) = \min_{P_{it}, U_{it}} \mathcal{L}(P, U, \lambda) \dots\dots\dots(2.14)$$

q^* adalah nilai fungsi *Lagrange* yang berhubungan dengan P_{it} dan U_{it} dari hasil fungsi minimalisasi yang memiliki nilai minimum (nilai negatif) terbesar.

Ini dilakukan dengan dua langkah dasar sebagai berikut :

- a. Mencari nilai untuk masing-masing λ_t yang akan mengubah nilai $q(\lambda)$ ke nilai yang besar.
- b. Mengasumsikan nilai λ_t pada langkah a sudah tetap, sehingga dapat dicari nilai minimum dari \mathcal{L} dengan mengatur nilai dari P_t dan U_t .

Dari langkah diatas dapat diasumsikan bahwa nilai λ_t adalah sebuah nilai yang tetap, dan yang didasarkan pada batasan, pada persamaan (2.9)-(2.11), sehingga dapat meminimumkan fungsi *Lagrange*, dan diperoleh persamaan sebagai berikut :

$$\begin{aligned} \mathcal{L} &= \sum_{t=1}^T \sum_{i=1}^N [F_i(P_{it}) + ST_i(1 - U_{i,t-1})]U_{it} + \sum_{t=1}^T \lambda_t \left(D_t - \sum_{i=1}^N P_{it}U_{it} \right) \\ &= \sum_{t=1}^T \sum_{i=1}^N [F_i(P_{it}) + ST_i(1 - U_{i,t-1})]U_{it} + \sum_{t=1}^T \lambda_t D_t - \sum_{t=1}^T \sum_{i=1}^N \lambda_t P_{it}U_{it} \dots (2.15) \end{aligned}$$

Pada persamaan (2.15) diatas, D_t merupakan variabel kendala bersama sehingga dapat diabaikan atau mengalami proses *relaxing*, sehingga fungsi *Lagrange* dapat ditulis sebagai berikut :

$$\mathcal{L} = \sum_{i=1}^N \left(\sum_{t=1}^T \{ [F_i(P_{it}) + ST_i(1 - U_{i,t-1})]U_{it} - \lambda_t P_{it}U_{it} \} \right) \dots \dots \dots (2.16)$$

Dari persamaan (2.16) diatas, dapat dipisahkan tiap-tiap unit dengan unit lainnya, sehingga diperoleh persamaan sebagai berikut :

$$\sum_{t=1}^T \{ [F_i(P_{it}) + ST_i(1 - U_{i,t-1})]U_{it} - \lambda_t P_{it}U_{it} \} \dots \dots \dots (2.17)$$

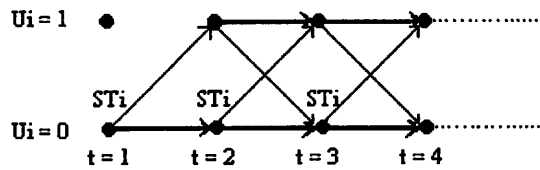
Persamaan (2.16) dan (2.17) diatas dapat dipecahkan secara terpisah untuk masing-masing unit pembangkit, tanpa memperhatikan apa yang terjadi pada unit pembangkit lainnya. Kemudian nilai minimum dari fungsi *Lagrange* dapat ditemukan dengan menyelesaikan minimalisasi untuk masing-masing unit pembangkit terhadap semua periode waktu, sehingga didapat persamaan sebagai berikut :

$$\min \mathcal{L} = \sum_{i=1}^N \min \sum_{t=1}^T \{ [F_i(P_{it}) + ST_i(1 - U_{i,t-1})]U_{it} - \lambda_t P_{it}U_{it} \} \dots \dots \dots (2.18)$$

dimana persamaan (2.18) diatas dibatasi oleh batas pembangkitan, minimum up time dan minimum down time.

Persamaan (2.18) diatas dapat mudah diselesaikan seperti masalah *dynamic programming* dalam satu variabel, dan dapat digambarkan pada gambar

(2.6), yang mana menunjukkan dua kemungkinan keadaan dari unit i yaitu $U_{it} = 0$ atau $U_{it} = 1$ dan ST_i adalah biaya start up untuk unit i .



Gambar 2.6 Two State Unit Pembangkit ^[2]

Pada $U_{it} = 0$, nilai dari fungsi minimalisasi adalah sama dengan 0, sedangkan pada $U_{it} = 1$, dengan mengeluarkan biaya *start up* pada minimalisasi yang berhubungan dengan (P_{it}) fungsi untuk meminimalkan adalah :

$$\min [F_i(P_{it}) - \lambda_i P_{it}] \dots\dots\dots (2.19)$$

Fungsi minimum dari persamaan diatas dapat diperoleh dengan mencari turunan pertama sebagai berikut :

$$\frac{d}{dP_{it}} [F_i(P_{it}) - \lambda_i P_{it}] = \frac{d}{dP_{it}} F_i(P_{it}) - \lambda_i = 0 \dots\dots\dots (2.20)$$

Persamaan (2.20) diatas dapat diselesaikan dengan persamaan sebagai berikut :

$$\frac{d}{dP_{it}} F_i(P_i^{opt}) = \lambda_i \dots\dots\dots (2.21)$$

Dari persamaan diatas dapat diketahui nilai P_i^{opt} yang merupakan daya keluaran optimal setiap unit ke- i .

Dari persamaan (2.21) diatas terdapat tiga ketentuan yang dapat diambil berkaitan hubungan antara P_i^{opt} dengan batas pembangkitan unit sebagai berikut :

1. Jika $P_i^{opt} \leq P_i^{min}$, maka :

$$\min [F_i(P_{it}) - \lambda_t P_{it}] = F_i(P_i^{min}) - \lambda_t P_i^{min} \dots\dots\dots (2.22)$$

2. Jika $P_i^{min} \leq P_i^{opt} \leq P_i^{maks}$, maka :

$$\min [F_i(P_{it}) - \lambda_t P_{it}] = F_i(P_i^{opt}) - \lambda_t P_i^{opt} \dots\dots\dots(2.23)$$

3. Jika $P_i^{opt} \geq P_i^{maks}$, maka :

$$\min [F_i(P_{it}) - \lambda_t P_{it}] = F_i(P_i^{maks}) - \lambda_t P_i^{maks} \dots\dots\dots (2.24)$$

Penyelesaian masalah program dinamis dua keadaan untuk tiap unit dilakukan dalam langkah yang biasa seperti cara penyelesaian untuk *Dynamic Programming* dalam menyelesaikan masalah komitmen unit dengan penambahan biaya *start up* (ST_i) yang sebelumnya diabaikan sementara. Apabila dilakukan pencarian untuk meminimalkan $[(F_i(P_{it}) + ST_i) - \lambda_t P_{it}]$ pada tiap stage dan ketika $U_{it} = 0$ nilai yang didapat akan selalu menjadi nol, oleh karena itu dilakukan suatu cara untuk mendapatkan nilai yang lebih kecil dengan cara sebagai berikut :

$$[(F_i(P_{it}) + ST_i) - \lambda_t P_{it}] < 0 \dots\dots\dots(2.25)$$

2.5.3 Pengaturan Pengali Lagrange^[2]

Untuk cara konvensional biasanya pengaturan pengali *Lagrange* (λ) untuk tiap-tiap periode waktu jam digunakan optimasi subgradien (ξ), dengan persamaan sebagai berikut :

$$\lambda_i^{k+1} = \lambda_i^k + \alpha \xi_i \dots\dots\dots(2.26)$$

dengan subyek pada $\lambda_t \geq 0$

dimana k adalah nomer iterasi, α merupakan pengali untuk perubahan λ_t yang diperoleh dari hasil pengujian, dan ξ_t adalah turunan dari fungsi *Lagrange* terhadap λ_t yang dapat dituliskan pada persamaan sebagai berikut :

$$\xi_t = \nabla f(\lambda_t) = D_t - \sum_{i=1}^N P_{it} U_{it} \dots\dots\dots(2.27)$$

2.5.4 Relative Duality Gap^[2]

Dalam menentukan hasil pemecahan yang mungkin pada solusi minimalisasi biaya pembangkitan j^* , dimana jika status $U_{it} = 0$, maka dapat menentukan nilai awal j^* pada nilai yang besar, sedangkan nilai awal q^* sama dengan 0. Nilai $j^* - q^*$ dinamakan *duality gap* dan dapat disebut *relative duality gap* pada persamaan sebagai berikut :

$$\frac{j^* - q^*}{q^*} \dots\dots\dots(2.28)$$

dimana :

- c. j^* (*primal value*) adalah nilai optimal fungsi obyektif $F(P_{it}, U_{it})$, yang merupakan total biaya pembangkitan dari semua periode jam terhadap daya keluaran (P_{it}^*) yang diperoleh dari hasil perhitungan *economic dispatch*. Dapat dituliskan pada persamaan sebagai berikut :

$$j^* = \sum_{t=1}^T \sum_{i=1}^N [F_i(P_{it}^*) + ST_i(1 - U_{i,t-1})] U_{it} \dots\dots\dots(2.29)$$

- d. q^* (*dual value*) adalah nilai dari fungsi *Lagrange* (£) dengan memperhatikan pengali *Lagrange* (λ_t) dan variabel P_{it} dan U_{it} .

Nilai yang didapat dari *relative duality gap* akan menentukan berhentinya proses dari prosedur *Lagrange relaxation*, dimana apabila nilai dari *relative duality gap* sudah cukup kecil maka prosedur *Lagrange relaxation* sudah cukup menghasilkan solusi yang optimal.

2.6 Prosedur *Evolutionary Programming* ^{[2] [4] [5] [6] [10] [11] [12]}

Evolutionary Programming (EP) merupakan metode stokastik yang bisa digunakan untuk memecahkan suatu pencarian nilai dalam sebuah masalah optimasi. Metode ini didasarkan pada proses evolusi yang ada dalam mahluk hidup yaitu perkembangan generasi dalam sebuah populasi yang alami, secara lambat laun mengikuti prinsip seleksi alam “siapa yang kuat, dia yang bertahan (*survive*)”. Dengan meniru proses ini, EP dapat digunakan untuk mencari solusi permasalahan-permasalahan dalam kehidupan nyata.

EP adalah suatu metode strategi optimasi yang merupakan cabang dari *Evolutionary Coputation* yang didalamnya terdiri dari *Algoritma Genetika*, *Genetic Programming*, *Evolutionary Stategies* dan *Evolutionary Programming*. Perbedaan yang paling mendasar antara *Evolutionary Programming* dengan *Algoritma Genetika* adalah pada proses operasi. Dalam metode *Evolutyonary Programming* tidak menggunakan operasi *crossover* melainkan operasi *competition* (kompetisi).

EP ditemukan oleh Lawrence.J. Fogel dilandasi oleh sifat-sifat evolusi alam. Fogel percaya bahwa ini sangat cocok digabungkan dengan sebuah algoritma komputer, menghasilkan sebuah teknik penyelesaian untuk permasalahan-permasalahan yang sulit dengan langkah alami yaitu melalui

evolusi. Fogel mulai bekerja dengan algoritma yang terbentuk dari string-string bilangan real yang disebut kromosom. Seperti halnya alam, metode ini menyelesaikan permasalahan-permasalahan dengan menentukan kromosom-kromosom yang baik dengan manipulasi materi dari sifat (gene) kromosom-kromosom. Algoritma ini tidak mengetahui tipe permasalahan yang akan diselesaikan. Sebelum EP dijalankan sebuah kode yang sesuai (*representasi*) untuk persoalan harus dirancang. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/*string* yang terdiri dari komponen genetic terkecil yaitu gen. Pemakaian bilangan biner (floating poin) sebagai allele (nilai gen) memungkinkan penerapan operator EP yaitu proses produksi (*reproduction*), mutasi (*mutation*), dan kompetisi (*competition*) untuk menciptakan himpunan titik-titik solusi untuk memeriksa hasil optimasi, kita membutuhkan fungsi fitness yang menandakan gambaran hasil (*solution*) yang sudah dikodekan. Selama proses, induk harus digunakan untuk reproduksi, mutasi dan kompetisi untuk menciptakan keturunan (*offspring*).

EP memiliki empat dasar kerja yaitu:

1. Bekerja dengan mengkodekan parameter-parameter permasalahan dan tidak bekerja secara langsung dengan parameter-parameter tersebut.
2. Mencari solusi masalah dari sejumlah populasi kandidat solusi, tidak hanya memproses satu solusi saja.
3. Hanya memperhitungkan fungsi fitness setiap kandidat solusi untuk mendapatkan hasil optimasi global.
4. Menggunakan aturan transisi secara *probabilistic* bukan *deterministic*.

2.6.1. Parameter Evolutionary Programming

Terdapat beberapa parameter yang digunakan dalam EP. Parameter tersebut digunakan untuk melihat kompleksitas dari EP. Parameter yang digunakan tersebut adalah:

a. *Jumlah Generasi (MAXGEN)*

Merupakan jumlah perulangan (interasi) dilakukan rekombinasi dan seleksi. Jumlah ini mempengaruhi kestabilan *output* dan lama intersi (waktu proses EP). Jumlah generasi yang besar dapat mengarah kearah solusi yang optimal, namun akan membutuhkan waktu yang lama. Sedangkan jika jumlah generasi terlalu sedikit maka solusi akan terjebak pada *local optimum solution*.

b. *Ukuran Populasi (POPSIZE)*

Ukuran populasi mempengaruhi kinerja dan efektifitas dari EP. Jika ukuran populasi kecil maka populasi tidak menyediakan kinerja EP menjadi buruk. Dalam hal ini dibutuhkan ruang yang lebih besar untuk mempresentasikan keseluruhan ruang permasalahan. Selain itu penggunaan populasi yang besar dapat mencegah terjadinya konvergensi pada wilayah local.

c. *Probabilitas Mutasi (Pm)*

Mutasi digunakan untuk meningkatkan variasi populasi yang dapat digunakan untuk menentukan tingkat mutasi yang terjadi, karena frekwensi terjadinya mutasi tersebut menjadi $Pm \times POPSIZE \times N$, dimana N adalah panjang struktur/gen dalam satu individu. Probabilitas mutasi yang rendah akan menyebabkan gen-gen yang berpotensi tidak dicoba. Dan sebaliknya, tingkat

mutasi yang tinggi akan menyebabkan keturunan akan semakin mirip dengan induknya. Dalam EP mutasi menjalankan aturan penting yaitu:

1. Menganti gen-gen yang hilang selama proses seleksi.
2. Menyediakan gen-gen yang tidak muncul pada saat inialisasi awal populasi.

d. Panjang Kromosom

Panjang kromosom berbeda-beda sesuai dengan model permasalahan. Titik solusi dalam ruang permasalahan dikodekan dalam bentuk kromosom/string yang terdiri dari komponen genetic terkecil yaitu gen.

2.6.2. Mekanisme *Evolutionary Programming* ^[12]

A. Pengkodean atau *Representasi*

Langkah pertama kali yang dilakukan dalam penggunaan EP adalah melakukan pengkodean atau representasi terhadap *permasalahan* yang akan dilakukan. Disini nilai yang di kodekan adalah nilai dari λ, μ untuk interval penjadwalan 24 jam. Dengan matrik yang dapat kita tuliskan sebagai berikut:

$$\lambda = [\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{24}]$$

$$\mu = [\mu_1, \mu_2, \mu_3, \dots, \mu_{24}]$$

Secara umum EP dibentuk serangkaian kromosom yang ditandai dengan $x_i (i=1,2,\dots,N)$. setiap elemen dalam kromosom ini adalah variable string yang disebut gen, berisi nilai-nilai atau allele. Variable-variabel ini dapat dinyatakan dalam bentuk bilangan interger, dari matrik diatas akan diterjemahkan atau dikodekan dalam bentuk EP dengan cara acak sebagai berikut:

$$\lambda = [\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{24}]$$



$$[1.3, 1.2, 0.2, \dots, N]$$

$$\mu = [\mu_1, \mu_2, \mu_3, \dots, \mu_{24}]$$

Selanjutnya beberapa kromosom dibentuk dan berkumpul membentuk populasi. Populasi inilah populasi awal bagi EP untuk awal melakukan pencarian.

B. Fungsi Fitness (Fungsi Evaluasi)

Dalam EP, sebuah fungsi fitness (*FIT*) harus dirancang untuk masing-masing permasalahan yang akan diselesaikan. Dengan menggunakan kromosom tertentu, fungsi objektif atau fungsi evaluasi akan mengevaluasi status masing-masing kromosom. Setiap gen x_i ($i=1,2,\dots,N$) dipergunakan untuk menghitung $f_k(x)$ ($k=1,2,\dots,POPSIZE$).

Pada permulaan optimasi, biasanya nilai fitness masing-masing individu masih mempunyai rentang yang lebar. Seiring dengan bertambah besar generasi, beberapa kromosom mendominasi populasi dan mengakibatkan rentang nilai fitness semakin kecil.

C. Mutation (mutasi)

Operator mutasi digunakan untuk memodifikasi satu atau lebih nilai gen dalam satu individu. Cara kerjanya dengan membangkitkan sebuah nilai random r_k dimana $k=1,2,\dots$ (panjang kromosom). Probabilitas mutasi (P_m) ditentukan dan digantikan untuk mengendalikan frekuensi operator mutasi. Apabila nilai random r_k , P_m maka gen ke- k kromosom tersebut terpilih untuk mengalami mutasi. Proses mutasi dalam *Evolutionary Programming* menggunakan operator *Gaussian*

mutation, dimana setiap individu akan terpilih secara acak untuk mengalami mutasi berdasarkan nomor acak Gaussian untuk menciptakan individu baru (*offspring*).

Fungsi dari operator mutasi adalah untuk menghindari agar solusi masalah yang diperoleh bukan merupakan solusi optimum local. Tipe dan implementasi dari operator mutasi bergantung pada jenis pengkodean dan permasalahan yang dihadapi. Seberapa sering mutasi dilakukan dinyatakan dalam suatu probabilitas mutasi, P_m . Posisi elemen pada kromosom yang akan mutasi ditentukan secara random. Mutasi dikerjakan dengan cara melakukan perubahan pada elemen tersebut.

D. Seleksi

Masalah yang paling mendasar pada proses ini adalah bagaimana proses penyeleksiannya. Menurut teori Darwin proses seleksi individu adalah: “*individu terbaik akan tetap hidup dan menghasilkan keturunan*”. Pada proses seleksi ini dapat menggunakan banyak metode seperti *roulette wheel selection*, *rank selection*, *elitesm* dan lain sebagainya.

1. Roulette Wheel Selection

Dimana setiap individu memiliki harga fitness sehingga didapatkan probabilitas individu tersebut dikopikan pada populasi yang baru. Untuk individu yang memiliki probabilitas 20% untuk jumlah populasi 10 maka kemungkinan individu tersebut dapat terpilih sebanyak dua kali.

Adapun algoritma dari *roulette-wheel* adalah sebagai berikut:

1. Menjumlahkan fitness dari seluruh anggota populasi.

2. Membangkitkan nilai k , nilai random antar 0 dan total fitnessnya.
3. Menjumlahkan fitness dari kromosom-kromosom dari populasi mulai 0 hingga total fitness lebih besar atau sama dengan nilai k lalu ambil kromosom tersebut.

2. Rank Selection

Apabila *fitness* yang dimiliki oleh suatu kromosom dalam populasi berbeda terlalu jauh dari kromosom lainnya maka hal ini dapat menjadi permasalahan. Misalnya bila kromosom terbaik mempunyai *fitness* yang menyebabkan besarnya tempat yang dimilikinya dalam *roulette wheel* sebesar 90% maka kromosom-kromosom yang lain akan mempunyai peluang yang terlalu kecil untuk diseleksi.

Rank selection pertama kali merangking populasi dan kemudian setiap kromosom diberi nilai *fitness* baru berdasarkan hasil rangking tersebut. Yang pertama yang akan mempunyai nilai *fitness* 1, yang kedua akan mempunyai *fitness* 2 dan seterusnya sampai yang terakhir akan mempunyai *fitness* N . dengan demikian semua kromosom akan mempunyai peluang untuk diseleksi.

E. Competition (kompetisi)

Dalam tahap kompetisi, mekanisme seleksi dipakai untuk menghasilkan populasi dari populasi yang ada. Melalui penggunaan skema kompetisi setiap individu dalam populasi baik orang tua (*peren*) maupun anak (*offspring*) akan dikompetisi satu sama lainnya. Kompetisi individu dengan lawannya didasarkan pada nilai *fitness* dari setiap individu tersebut. Agar optimal, solusi yang lebih pas atau yang lebih optimal seharusnya memiliki peluang seleksi yang

lebih besar. Individu yang memenangkan dari kompetisi akan digunakan sebagai individu baru bagi pembangkitan selanjutnya.

2.7. Aplikasi Metode *Evolutionary Programming dan Lagrange Relaxation* (EPLR) dalam penyelesaian Komitmen Unit^{[6][7][8][9]}

Ide dasar dari EPLR adalah EP digabungkan kedalam metode LR untuk meng-update Lagrange Multiplier dan meningkatkan performa dari LR. Metode LR memecahkan masalah komitmen unit dengan “relaxing” atau untuk sementara mengabaikan kendala-kendala bersama (*coupling constraints*) dan memecahkan masalah seperti mereka tidak ada. Prosedur penguraian LR, berdasarkan pada teori optimasi ganda (*dual optimization theory*), menciptakan masalah yang dapat dipecahkan dengan menggabungkan (*integrating*) beberapa kendala bersama ke dalam fungsi obyektif (*objective function*),. Dengan memecahkan masalah primal, ini dapat memecahkan masalah ganda dengan memaksimalkan fungsi lagrange terhadap factor pengali lagrange, selama minimasi terhadap variabel kontrol komitmen unit.

Prosedur penguraian LR tergantung pada perhitungan/pencarian nilai inisial/awal dari LM dan juga pada metode yang digunakan untuk mengupdate pengalinya (*multipliers*). Kesulitan pada metode LR adalah performa/keandalan perhitungan sangat tergantung pada metode dimana LM diperbarui. Melihat hal diatas, LM merupakan unsur penting dari metode LR dalam pencapaian solusi.

Oleh sebab itu pemilihan atau penentuan LM harus dilakukan secara hati-hati dan dicari suatu teknik yang cocok dalam memperbarui pengalinya. Sekarang ini terdapat berbagai teknik yang digunakan untuk menghitung LM mengacu dan mengandalkan pada algoritma *sub-gradient* atau *heuristic*, atau kombinasi. Penggunaan algoritma *sub-gradient* guna mengupdate LM dapat memberikan solusi hasil untuk cadangan berputar yang tidak layak untuk diterapkan. Hal inilah yang mendasari pemakaian EP dalam skripsi ini untuk memilih dan mengupdate nilai LM, sehingga dapat meningkatkan performa/keandalan dari metode LR. EP adalah metode pencarian paralel didasar pada mekanisme dari seleksi evolusi alam dan genetika alami.

Metode LREP memasukkan EP ke dalam LR untuk memperoleh LM dan meningkatkan performa dari LR. EP mengkombinasi sifat penyesuaian diri dari genetika alami atau prosedur evolusi dari organ dengan fungsi optimasi. Dengan mensimulasi “yang tersehatlah yang akan bertahan” dari evolusi Darwin diantara struktur kromosom, pencarian kromosom yang terbagus diperoleh dengan penukaran informasi secara acak. Empat operator yang berhubungan dengan EP adalah representasi, mutasi, seleksi, kompetisi. Metode EPLR terdiri dari dua keadaan putaran. Tingkat pertama untuk mencari nilai minimum dari fungsi Lagrange menurut LM yang konstan dengan DP dua keadaan (two-state). Tingkat yang kedua adalah dengan memaksimalkan fungsi Lagrange terhadap pengaturan LM oleh EP.

2.7.1 Algoritma penyelesaian kombinasi metode EP dan LR untuk masalah

Komitmen Unit ^{[6][7][8][9]}

Adapun langkah-langkah penyelesaian metode EPLR untuk masalah Komitmen Unit diberikan berikut ini:

1. Menentukan nilai awal pengali *Lagrange* λ dan μ
2. Mencari batas minimum dari fungsi lagrange L untuk setiap unit guna memperoleh P'_i dan U'_i menggunakan DP dua keadaan (two-state dynamic programming).

Minimum dari fungsi lagrange L (P, U, λ, μ) dapat dicari dengan DP dua keadaan (two-state DP) dalam dua variable untuk setiap unitnya. Skema dari keadaan DP dua keadaan dapat dilihat pada sub bab 2.5 gambar 2.6, dimana pada gambar hanya ada dua kemungkinan keadaan ($U'_i = 1$ atau 0) dan tidak lupa diberlakukannya constrain minimum up / down time dan start-up untuk unit ke-i.

3. Perhitungan nilai dual value q^*
4. Melakukan perhitungan *economic dispatch* tiap periode jam
5. Melakukam perhitungan *primal value*
6. Perhitungan *relatif duality gap*:

$$\varepsilon = \frac{j - q}{j}$$

7. Apabila terminating kriteria belum tercapai karena nilai duality gap besar maka pengali Lagrange akan diperbaharui oleh EP, dan kembali pada langkah ke-3.

8. Pada proses EP disini memiliki tahapan-tahapan proses yaitu:

- a. Memasukkan data parameter yang berupa nilai dari λ dan μ .
- b. Inisialisasi parameter-parameter EP
- c. Inisialisasi populasi dari solusi-solusi "parent" S_k , $k = 1, 2, 3, \dots, M$ (dimana M merupakan nilai parent), dihasilkan secara random. Diasini masing-masing λ dan μ akan dibentuk sesuai dengan inisialnya. Offspring yang layak mengganti beberapa inisial parent. Proses ini bisa diulang beberapa kali (yaitu 5 sampai 10 iterasi) untuk menghasilkan inisial populasi yang berbeda.
- d. Melakukan Perhitungan Fungsi Fitness

Metode EPLR menggunakan relatif duality gap yang berbasis aturan konvergensi (kesesuaian), nilai fitness di sini dikembalikan pada nilai duality gap.

$$FIT = \frac{1}{1 + K \bullet \left(\frac{f_{\max}}{f_i} - 1 \right)}$$

$$f_i = \frac{1}{relativeDualityGap} = \frac{1}{\epsilon} \quad f_{\max} = \text{maksimum dari populasi}$$

- e. Melakukan proses mutasi.

Populasi yang baru dari solusi Multiplier yang telah diperbarui dihasilkan dari populasi yang ada melalui operator mutasi. Individu yang baru S'_k dihasilkan dari setiap individu lama S_k , dimana variabel j , pada individu yang baru S'_k dihitung seperti:

$$x'_{jk} = x_{jk} + N(0, \sigma_{jk}^2)$$

dimana x'_{jk} menandakan nilai variable j dalam S'_k . x_{jk} adalah dari variable j dalam induk S'_k dan $N(0, \sigma_{jk}^2)$ adalah nomor acak Gaussian dengan nilai tengah nol dan standart deviasi (simpang baku) dari σ_{jk} , cara untuk merancang σ_{jk} adalah:

$$\sigma_{jk}^2 = (x_j^{\max} - x_j^{\min}) \left((f_{\max} - f_k) / f_{\max} + a^r \right)$$

dimana f_i adalah fitness dari individu k, f_{\max} adalah fitness maksimum di dalam populasi, x_j^{\max}, x_j^{\min} menandakan batasan atas dan bawah dari variable j, a adalah konstanta angka positif, dan r adalah intersi yang berulang-ulang.

f. Seleksi individu Dengan Kompetisi

Dalam tahap kompetisi, mekanisme kompetisi dipakai untuk menghasilkan populasi baru dari populasi yang sudah ada. Teknik seleksi yang dipakai adalah sekema tournament yang dapat dideskripsikan sebagai berikut:

K merupakan solusi induk (perent) S_k seiring dengan anak (offspring) S'_k terbentuk mutasi, $k= 1,2, \dots, k$ masing-masing mengalami serangkaian turnamen n_j dengan lawan yang terseleksi secara acak. Setiap individu K diberi suatu bobot W_k berdasarkan:

$$W_k = \sum_{j=k}^N n_j$$

$$n_j = \begin{cases} 1, & \text{jika } f_k > f_r \\ 0, & \text{jika lain} \end{cases}$$

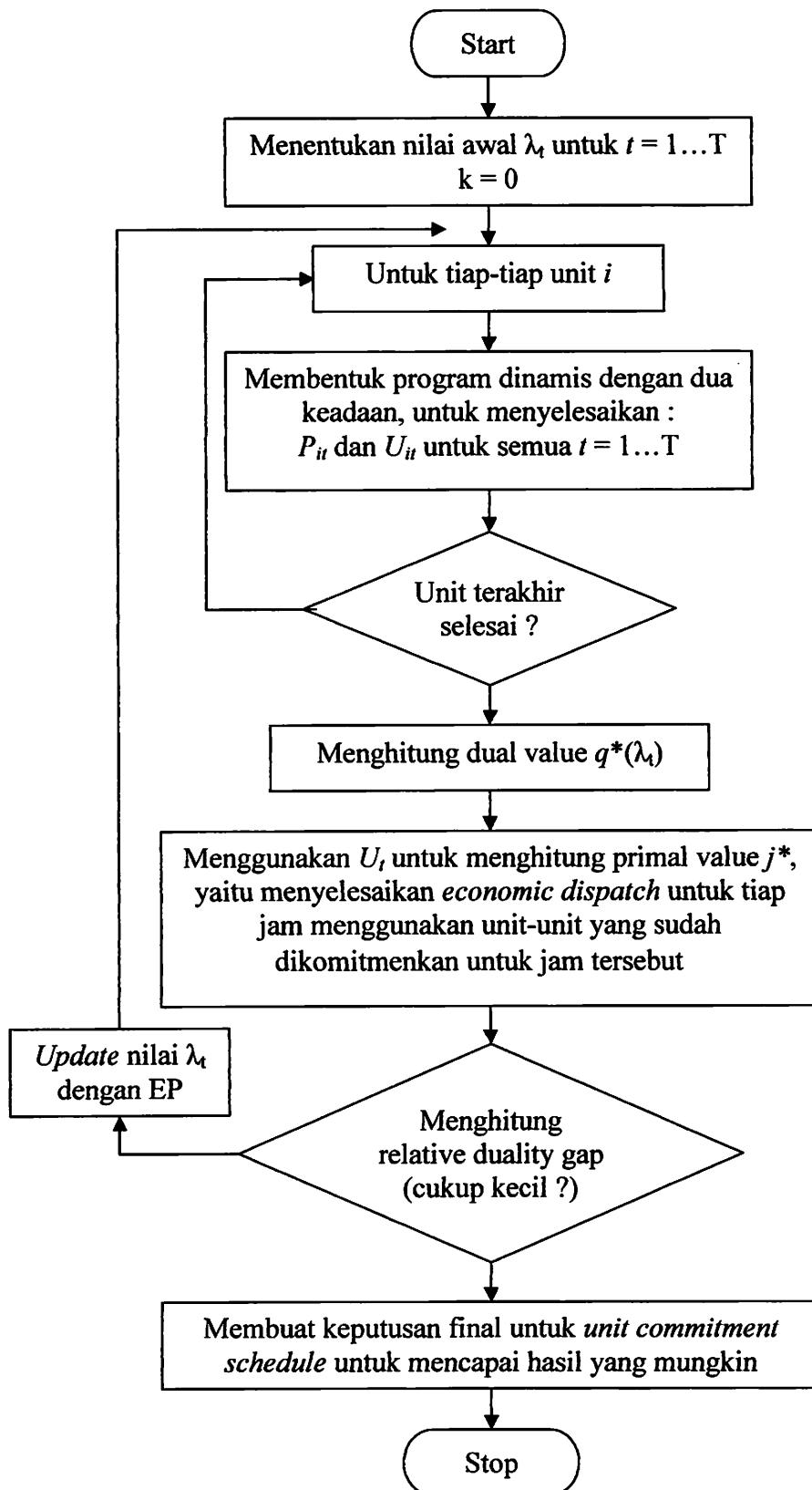
sedangkan f_k adalah fitness setiap individu k . Lawan r , terpilih acak dari setiap individu $2k$ berdasarkan $r = [2ku + 1]$. u adalah angka acak yang sama dengan interval $[0,1]$. K adalah solusi calon skor tertinggi diambil generasi selanjutnya.

- g. Competitor yang layak dirangking berdasarkan urutan menurun dari W_k . Solusi M yang pertama dapat bertahan dan dituliskan kembali bersama dengan elemennya sebagai dasar untuk generasi berikutnya.
- h. Proses (d sampai f) diulang sampai jumlah generasi yang ditentukan tercapai.

9. jumlah generasi = jumlah generasi + 1

10. Jika (jumlah generasi < max generasi atau nilai duality gap lebih besar dari ketetapan) kembali ke langkah 3, jika tidak lanjutkan kelangkah berikutnya.

11. Print hasil.



Gambar 2.9 Flowchart Metode LR Update EP

BAB III

DATA SISTEM PEMBANGKIT TERMAL

PADA PT. PEMBANGKITAN JAWA-BALI

3.1. Pendahuluan

Dalam menjalankan tugasnya sebagai penyedia kebutuhan akan energi listrik bagi masyarakat luas, PT. Pembangkitan Jawa-Bali dituntut untuk memberikan pelayanan yang memuaskan disamping juga harus memenuhi tujuan lainnya sebagai perusahaan yang bergerak dibidang jasa yaitu untuk mendapatkan keuntungan finansial.

PT. Pembangkitan Jawa-Bali dalam penyelenggaraan usaha ketenagalistrikan berdasarkan prinsip industri dalam perniagaan yang sehat, dituntut mampu bersaing dan mampu memanfaatkan sebesar-besarnya peluang pasar dalam bidang tenaga listrik. Dalam hal tersebut, PT. Pembangkitan Jawa-Bali harus menjaga efisiensi dan keandalan operasional penyediaan tenaga listrik dari pembangkit-pembangkit yang dimilikinya.

Dengan demikian merupakan suatu keharusan bagi seluruh jajaran PT. Pembangkitan Jawa-Bali agar selalu berupaya untuk meningkatkan kondisi penyediaan tenaga listrik dari pembangkit agar lebih ekonomis, bermutu dan didukung keandalan yang tinggi.

3.2. Data Pembangkit Termal

Pembangkit termal yang berada pada kawasan PT. Pembangkitan Jawa-Bali berjumlah 38 unit pembangkit yang terdiri dari 5 blok pembangkit listrik tenaga gas dan uap, 11 unit pembangkit listrik tenaga uap dan 5 unit pembangkit listrik tenaga gas. Adapun data-data lebih lengkapnya dapat dilihat pada table 3.1 dan 3.2, untuk harga bahan bakar berdasarkan statistik PLN tahun 2002 dimana dipakai nilai tukar Rp 9000 per dolar Amerika.

Tabel 3.1
Data Unit Termal Pada PT. Pembangkitan Jawa-Bali
Agustus 2002

No	Nama Pembangkit	Bahan Bakar	Kapasitas (MW)		Lama Waktu (Jam)			
			Min	Max	MU T	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 Gresik GT 1.1	Gas	53	102	36	10	1	0
4	PLTGU Gresik GT 1.2	Gas	53	102	36	10	1	0
5	PLTGU Gresik GT 1.3	Gas	53	102	36	10	1	0
6	PLTGU Gresik ST 1.0	Gas	250	480	36	10	3	1
7	PLTGU Gresik GT 2.1	Gas	53	102	36	10	1	0
8	PLTGU Gresik GT 2.2	Gas	53	102	36	10	1	0
9	PLTGU Gresik GT 2.3	Gas	53	102	36	10	1	0
10	PLTGU Gresik ST 2.0	Gas	250	480	36	10	3	1
11	PLTGU Gresik GT 3.1	Gas	53	102	36	10	1	0
12	PLTGU Gresik GT 3.2	Gas	53	102	36	10	1	0
13	PLTGU Gresik GT 3.3	Gas	53	102	36	10	1	0
14	PLTGU Gresik ST 3.0	Gas	250	480	36	10	3	1
15	PLTU Gresik 1	Gas	43	85	48	10	9	1
16	PLTU Gresik 2	Gas	43	85	48	10	9	1
17	PLTU Gresik 3	Gas	90	175	48	10	9	2
18	PLTU Gresik 4	Gas	90	175	48	10	9	2
19	PLTG Gresik 1	Gas	5	16	3	1	1	0
20	PLTG Gresik 2	Gas	5	16	3	1	1	0
21	PLTG Gresik 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 GT 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
MDT = minimum down time

Tabel 3.2
Data Biaya Dan Parameter Unit Termal Pada PT. Pembangkitan Jawa-Bali
Agustus 2002

No	Nama Pembangkit	Biaya Strat-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	149.48	3244978	111712.15	10.2971
3	PLTGU Gresik GT 1.1	7.82	0	5467532.4	217963.548	34.155
4	PLTGU Gresik GT 1.2	7.82	0	5467532.4	217963.548	34.155
5	PLTGU Gresik GT 1.3	7.82	0	5467532.4	217963.548	34.155
6	PLTGU Gresik ST 1.0	57.68	31.46	10936203.3	72527.004	368.875
7	PLTGU Gresik GT 2.1	7.82	0	5467532.4	217963.548	34.155
8	PLTGU Gresik GT 2.2	7.82	0	5467532.4	217963.548	34.155
9	PLTGU Gresik GT 2.3	7.82	0	5467532.4	217963.548	34.155
10	PLTGU Gresik ST 2.0	57.68	31.46	10936203.3	72527.004	368.875
11	PLTGU Gresik GT 3.1	7.82	0	5467532.4	217963.548	34.155
12	PLTGU Gresik GT 3.2	7.82	0	5467532.4	217963.548	34.155
13	PLTGU Gresik GT 3.3	7.82	0	5467532.4	217963.548	34.155
14	PLTGU Gresik ST 3.0	57.68	31.46	10936203.3	72527.004	368.875
15	PLTU Gresik 1	143.74	40.59	1327126.68	217378.359	132.066
16	PLTU Gresik 2	143.74	40.59	1327126.68	217378.359	132.066
17	PLTU Gresik 3	229.5	95.52	5017369.5	169242.579	193.545
18	PLTU Gresik 4	229.5	95.52	5017369.5	169242.579	193.545
19	PLTG Gresik 1	6.13	0	352707.3	350680.77	903.969
20	PLTG Gresik 2	6.13	0	352707.3	350680.77	903.969
21	PLTG Gresik 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 GT 1.0	54.22	29.67	11560815	53685.135	460.845
28	PLTGU M. Tawar GT 1.1	0	0	14706521.25	433337.8	49.4605
29	PLTGU M. Tawar GT 1.2	0	0	14706521.25	433337.8	49.4605
30	PLTGU M. Tawar GT 1.3	0	0	14706521.25	433337.8	49.4605
31	PLTGU M. Tawar GT 2.1	0	0	14706521.25	433337.8	49.4605
32	PLTGU M. Tawar GT 2.2	0	0	14706521.25	433337.8	49.4605
33	PLTGU M. Tawar GT 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. Gresik	2,53	US\$/MMBTU
	Harga Gas UP. M. Karang	2,45	US\$/MMBTU
	Nilai Tukar	9000	Rp/US\$

3.3. Aplikasi Kombinasi Metode Evolutionary Programming dan Lagrange

Relaxation Pada PT. PJB

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

- Tanggal 4 Desember 2003 adalah beban hari kerja penuh
- Tanggal 6 Desember 2003 adalah beban pada hari setengah penuh
- Tanggal 7 Desember 2003 adalah beban pada hari libur

Berdasarkan data unit yang terdapat dalam PT. Pembangkitan Jawa-Bali pada system tenaga pada tabel 3.1, ternyata pada saat dilakukan pengambilan data, semua unit pembangkit dalam kondisi siap beroperasi. Maka dapat disusun input data unit pembangkit termal yang siap operasi pada tanggal 4 Desember 2003 sampai dengan 7 Desember 2003, yaitu sebanyak 38 unit pembangkit.

Dalam data beban harian system yang diperoleh dari PT. Pembangkitan Jawa-Bali, terdapat data hasil perhitungan mengenai jumlah total pembangkitan, beban total dan cadangan berputar pada setiap jam dalam setiap area. Data-data ini tidak dipakai dalam skripsi ini karena data tersebut menyangkut sistem secara keseluruhan dalam suatu area. Dalam suatu area biasanya terdapat lebih dari satu perusahaan penyedia energi listrik. Misalnya pada area IV, terdapat tiga perusahaan penyedia energi listrik, yaitu: PT. Pembangkitan Jawa-Bali, PT. Indonesia Power dan perusahaan milik swasta. Jadi tidak relevan jika data-data tersebut digunakan dalam skripsi ini, sedangkan aplikasi pada skripsi ini hanya pada PT. Pembangkitan Jawa-Bali.

Model yang digunakan dalam melakukan perhitungan optimalisasi penjadwalan PLTU, PLTG maupun PLTGU menggunakan karakteristik tiap unit termal, meskipun PLTGU mempunyai karakteristik tiap blok yang saling tergantung antara unit gas (GT) dan unit uap (ST) atau yang sering disebut Combined Cycle. Untuk memudahkan perhitungan dilakukan dengan pendekatan per unit termis, dimana parameter unit tiap GT kita ambil dari parameter unit pembangkit sendiri sedangkan untuk unit ST diambil dari parameter kombinasi CC.1.1.1. (lihat tabel 3.2).

PT. Pembangkitan Jawa-Bali tidak mempunyai dasar yang pasti untuk menentukan nilai dari cadangan berputar (spining reserve) tiap periode jam, tetapi PT. Pembangkitan Jawa-Bali menggunakan asumsi bahwa nilai cadangan berputar diambil dari daya terpasang terbesar dari unit pembangkit yang beroperasi. Dalam hal ini PT. Pembangkitan Jawa-Bali menggunakan daya terpasang dari unit pembangkit PLTU Paiton yaitu sebesar 400 MW sebagai nilai cadangan berputar tiap periode jam

3.4. Beban Sistem

Dalam wilayah Jawa Bali, pembangkit-pembangkit yang ada dikoordinasi oleh PT. Pembangkitan Jawa-Bali. Proses penjadwalan pembangkit dengan kombinasi Metode Evolutionary Programming dan Lagrange Relaxation bertujuan untuk membuat rencana penjadwalan pembangkit dalam sistem tenaga listrik yang dapat memenuhi kebutuhan beban dengan 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. Pembangkitan Jawa-Bali sebagai bahan perbandingan. Sedangkan kombinasi jadwal dan daya output pembangkit tenaga listrik dalam sistem PT. Pembangkitan Jawa-Bali tanggal 4, 6 dan 7 Desember 2003, terdapat pada lampiran. Untuk beban sistem terdapat pada tabel 3.3 (beban sistem yang ditanggung oleh pembangkit termal saja).

Tabel 3.3
Data Beban Unit Termal Pada PT. Pembangkitan Jawa-Bali

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	3035	400	3000	400	2860	400
03.00	3035	400	3000	400	2860	400
04.00	3085	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	3447	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

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

3.5. Validasi Program

Pengujian validasi program dilakukan dengan menggunakan data jurnal

P. Attaviriyapap, H. Kita, E. Tanaka, J. Hasegawa. "Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method", 12th Annual Conference of Power & Energy Society, IEE of Japan. Data yang

digunakan pada kombinasi Evolutionary Programming dan Lagrange Relaxation terdiri dari 10 unit system. Adapun data tersebut sebagai berikut:

Tabel 3.4

Data Unit Commitment 10 unit untuk Validasi

	Unit 1	Unit 2	Unit 3	Unit 4	Unit 5
Pmax	455	455	130	130	162
Pmin	150	150	20	20	25
a	0,00048	0,00031	0,002	0,00211	0,00398
b	16,19	17,26	16,6	16,5	19,7
c	150	970	700	680	450
min up	8	8	5	5	6
min down	8	8	5	5	6
HST	4500	5000	550	560	900
CST	9000	10000	1100	1120	18000
CSH	5	5	4	4	4
Ini.	8	8	-5	-5	-6

	Unit 6	Unit 7	Unit 8	Unit 9	Unit 10
Pmax	80	85	55	55	55
Pmin	20	25	10	10	10
a	0,00712	0,00079	0,00431	0,00222	0,00173
b	22,26	27,74	25,92	27,27	27,79
c	370	480	660	665	670
min up	3	3	1	1	1
min down	3	3	1	1	1
HST	170	260	30	30	30
CST	340	520	60	60	60
CSH	2	2	0	0	0
Ini.	-3	-3	-1	-1	-1

Sumber : P. Attaviriyapap, H. Kita, E. Tanaka, J. Hasegawa. "Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method", 12th Annual Conference of Power & Energy Society, IEE of Japan.

Tabel 3.5
Data Beban Sistem Untuk Validasi

JAM	Beban Sistem (MW)	Reserve (MW)
01.00	700	70
02.00	750	75
03.00	850	85
04.00	950	95
05.00	1000	100
06.00	1100	110
07.00	1150	115
08.00	1200	120
09.00	1300	130
10.00	1400	140
11.00	1450	145
12.00	1500	150
13.00	1400	140
14.00	1300	130
15.00	1200	120
16.00	1050	105
17.00	1000	100
18.00	1100	110
19.00	1200	120
20.00	1400	140
21.00	1300	130
22.00	1100	110
23.00	900	90
24.00	800	80

Sumber : P. Attaviriyapap, H. Kita, E. Tanaka, J. Hasegawa. "Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method", 12th Annual Conference of Power & Energy Society, IEE of Japan.

BAB IV

SIMULASI PROGRAM DAN ANALISA KOMITMEN UNIT MENGUNAKAN KOMBINASI METODE *EVOLUTIONARY* *PROGRAMMING* DAN *LAGRANGE RELAXATION* PADA PT. PEMBANGKITAN JAWA- BALI

4.1 Pogram Komputer Untuk Menyelesaikan Komitmen Unit Pada PT. PEMBANGKITAN JAWA- BALI

Untuk pemecahan masalah unit komitmen digunakan bantuan program komputer. Program komputer ini sangat berguna untuk mempercepat proses perhitungan yang membutuhkan ketelitian tinggi dan sering melibatkan *iterasi* yang memerlukan waktu yang lama bila dikerjakan secara manual.

Program komputer ini menggunakan bahasa pemrograman Borland Delphi versi 7, yang merupakan bahasa pemrograman terstruktur yang relatif mudah untuk dipelajari dan mudah penggunaannya.

4.2 Algoritma Program

Dalam menyelesaikan masalah Komitmen Unit pada PT. PJB dengan menggunakan kombinasi metode EP dan LR digunakan urutan langkah-langkah yang dapat dilihat pada algoritma program berikut:

1. Memasukkan data parameter unit pembangkit termal dan data pembebanan harian untuk periode waktu 24 jam.

6. Melakukan perhitungan daya yang harus dibangkitkan (economic dispatch) tiap unit pembangkit untuk mendapatkan P^*_{it} dengan menggunakan status U'_i pada langkah no. 4.
7. Melakukan perhitungan primal value (J^*) dengan memperhatikan daya P^*_{it} yang dihasilkan dari perhitungan ekonomi dispatch pada langkah no. 6
8. Melakukan perhitungan relative duality gap.
9. Bila populasi telah maksimum lanjutkan ke langkah no. 10 bila belum kembali ke no.3
10. Perhitungan nilai fitness.
11. Melakukan proses statistik.

Dimana dalam proses ini kita menghitung total *fitness* dari semua individu

12. Melakukan proses seleksi

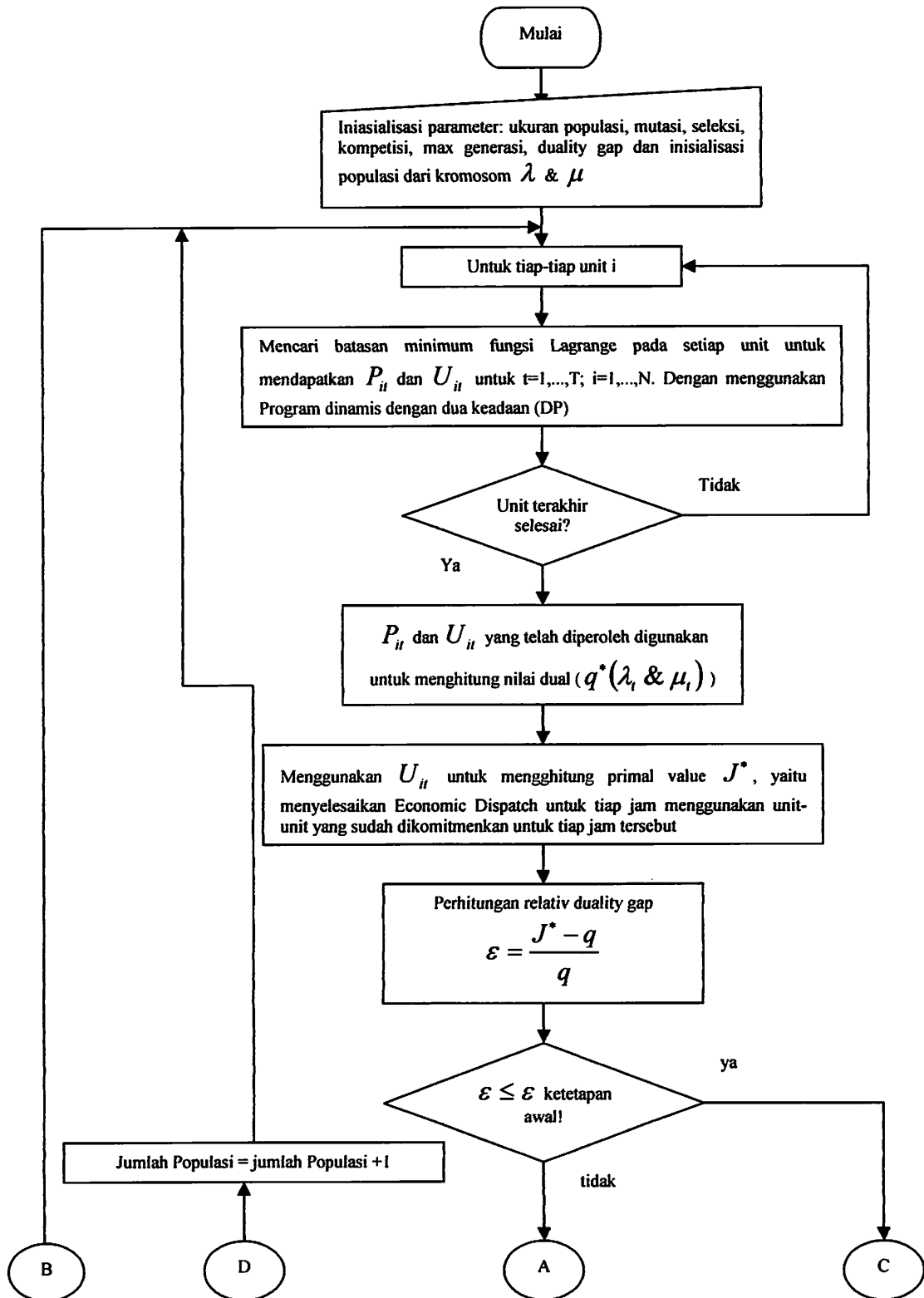
Pada proses ini digunakan proses roda rolet (roulette wheel), dimana nilai fitness masing-masing individu dibagi dengan total fitness dari seluruh individu, sehingga diperoleh nilai prosentase yang berlainan dari masing-masing individu, kemudian dilakukan proses pemutaran roda rolet sehingga secara logis individu yang memiliki prosentase lebih besar akan lebih sering terpilih untuk melakukan operasi genetika.

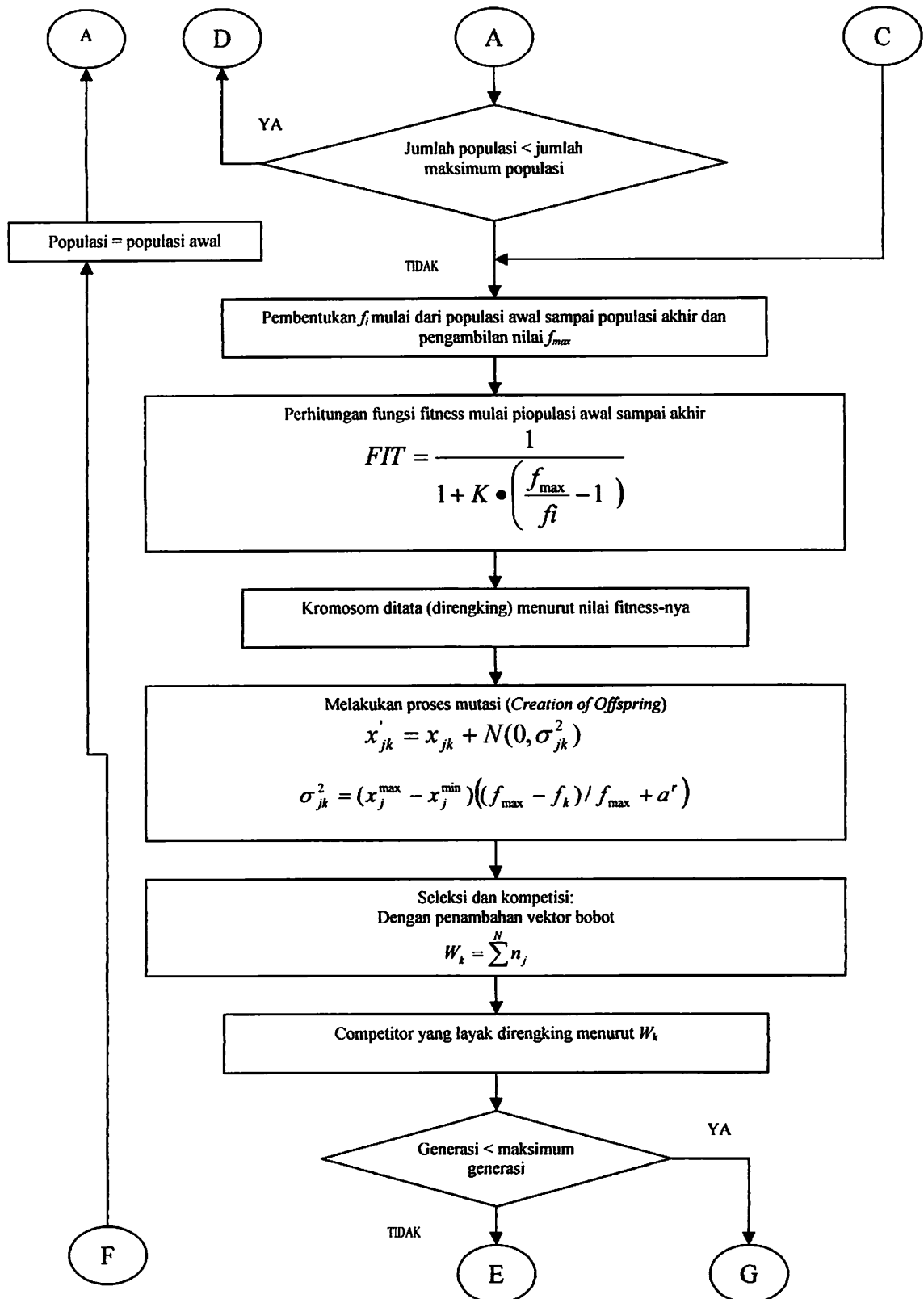
Algoritma penerapan pemilihan pada sistem roda rolet adalah sebagai berikut:

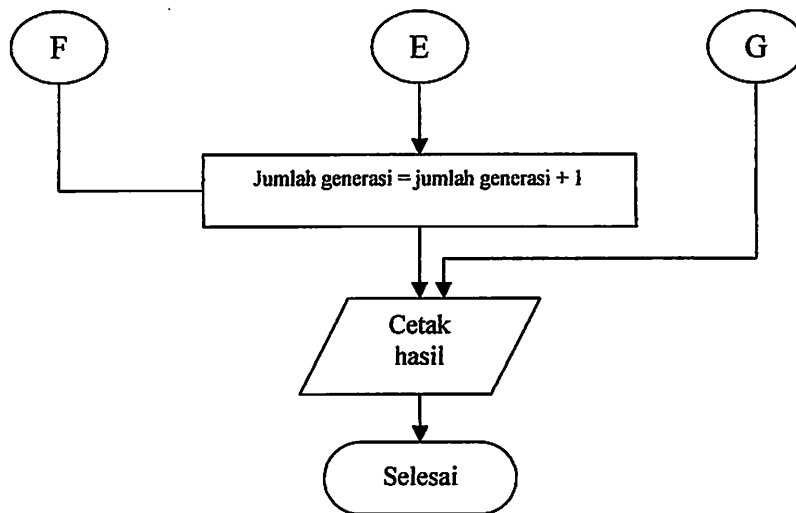
- a. Pembangkitan nilai random antara 0 dan 1.
- b. Perkalian antara total fitness yang telah diperoleh dengan nilai random yang telah dibangkitkan, kita asumsikan variabel hasil dari perkalian

- itu adalah “rand”. Rand ini pada roda rolet dapat disamakan dengan jarum penunjuk hasil perputaran, dimanapun dia berhenti disitulah nilai (dari fitness) akan diambil.
- c. Setelah rand diperoleh barulah akan dicari nilai fitness berapa yang dipilih, yaitu dengan cara sebagai berikut :
- Melakukan pencarian dengan perulangan.
 - Variabel yang diulang diantaranya adalah indeksasi untuk nilai fitness dari masing-masing populasi awal sampai akhir secara berurutan (sampai ditemukan titik rand).
 - Untuk populasi awal ditambahkan suatu variabel yang besarnya nol (kita set sebelum masuk perulangan) dan kita asumsikan nama variabel itu “pastrum”. Nama variabel ini juga kita pakai untuk hasil dari penjumlahan nilai partsum dengan populasi saat ini (populasi yang sedang dipilih).
 - Perulangan akan berhenti bila indeksasi telah menunjukkan nilai maksimum populasi atau hasil partsum lebih besar sama dengan (\geq) rand.
- d. Setelah perulangan selesai (berarti satu populasi telah terpilih), nilai indeksasi yang terakhir (dari perulangan terakhir) akan dipilih dan dimasukkan ke indeksasi nilai fitness populasi lalu proses kembali ke “a” sampai diperoleh jumlah populasi baru sama dengan populasi lama.

13. Melakukan proses mutasi
Populasi yang baru dari solusi multipliyer yang telah diperbaharui dihasilkan dari populasi yang ada melalui operator mutasi. (penjelasan dan rumus pada sub bab 2.7.1. poin 8e).
14. Melakukan prosaes kompetisi
Setelah individu-individu melakukan proses diatas maka akan dihasilkan individu yang baru. Setiap individu-individu yang terbentuk lama maupun baru akan mengalami proses kompetisi dengan penambahan vektor bobot (W_k). (penjelasan dan rumus pada sub bab 2.7.1. poin 8f).
15. Competitor yang layak akan direngking menurut W_k .
setelah melewati proses no.14 individu-individu yang masuk dalam rengking yang telah ditentukan disiapkan untuk proses genetika pada generasi selanjutnya.
16. Proses kembali pada langkah no.3 dan proses ini berulang terus seiring dengan tambahnya generasi dan akan berhenti hingga generasi menunjukkan jumlah maksimum generasi.
17. Penulisan hasil penjadwalan, daya keluaran dan total biaya pembangkitan selama 24 jam.

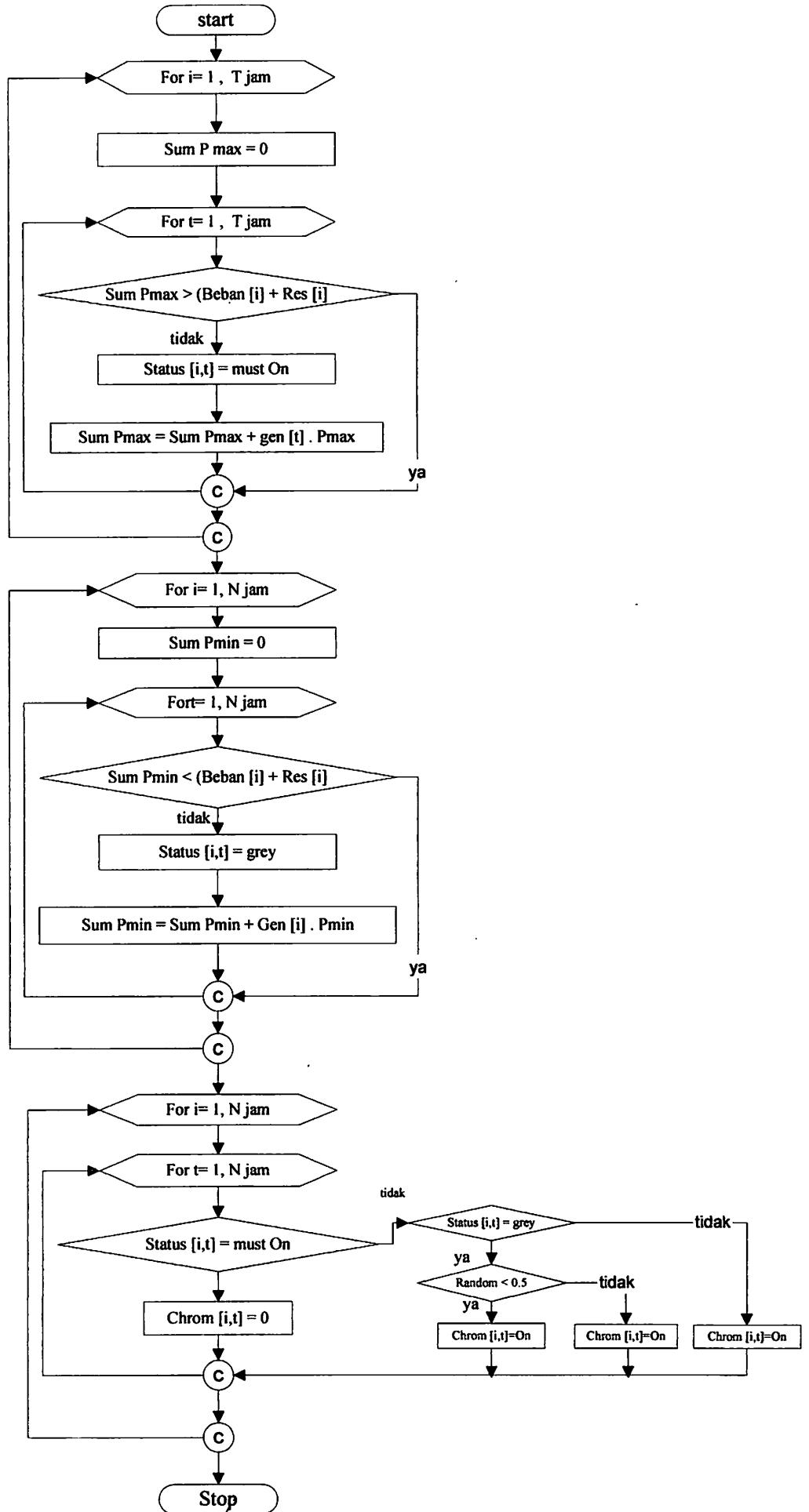




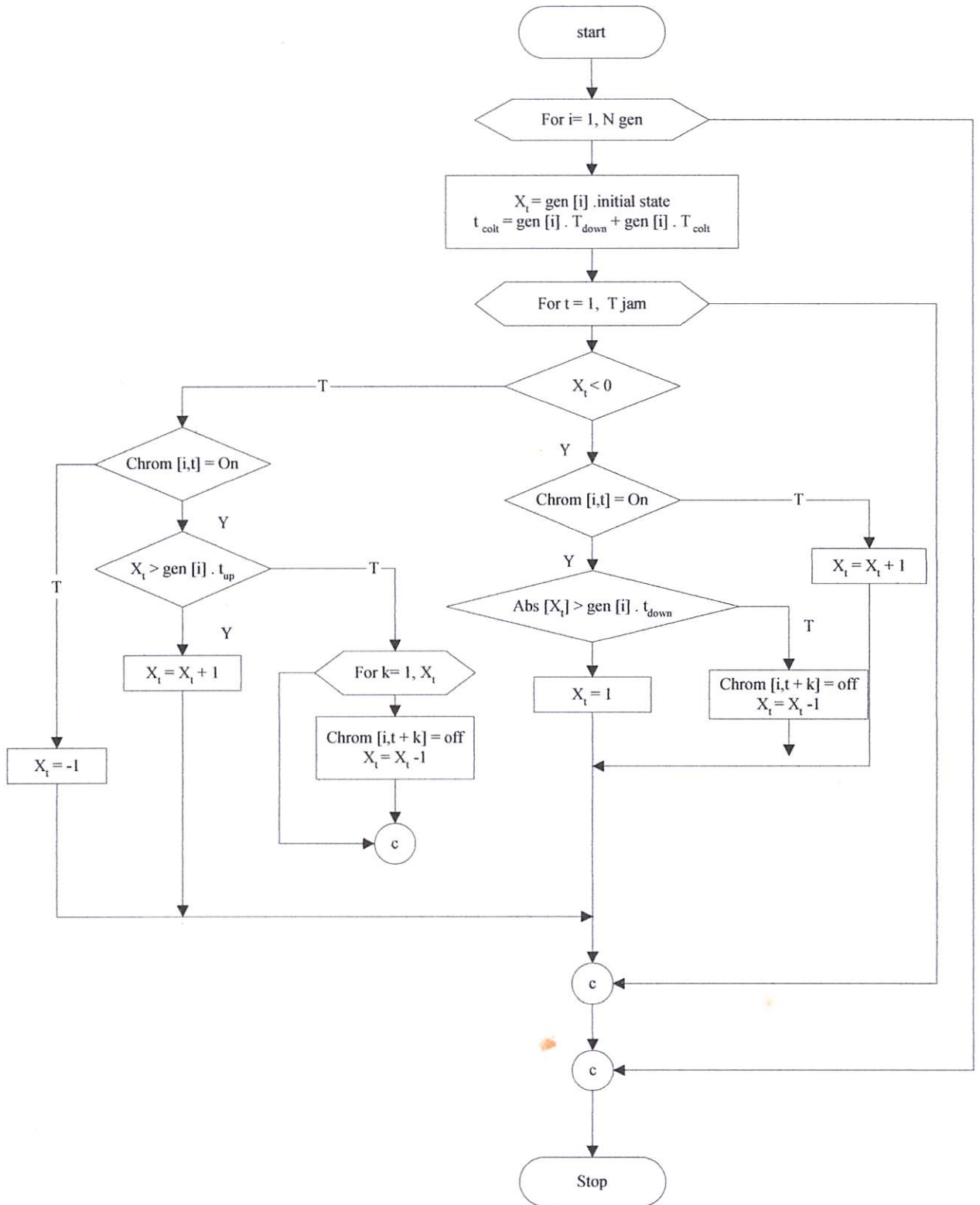


Gambar 4.1 Flowchart Pemrograman EPLR

Flow Chart Inisialisasi solusi parent



Flow Chart Repair Operation



4.4. Uji Validasi

Sebelum dilakukan optimasi perhitungan biaya total terlebih dahulu dilakukannya uji validasi program untuk melihat kelayakan dari program tersebut. Dalam program ini dilakukan pengujian berpedoman pada jurnal “*Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method*”, 12th Annual Conference of Power & Energy Society, IEE of Japan. Hasil pengujian dapat dilihat pada gambar 4.2. Kemudian dilakukan perhitungan biaya total dengan kombinasi metode *Evolutionary Programming* dan *Lagrange Relaxation* dalam tiap jam.

Tabel 4.1
Data Unit Commitment 10 Unit Untuk Validasi

No Unit	P_{\max} (MW)	P_{\min} (MW)	a	b	c	Min Up (h)	Min dn (h)	Hot Start Cost (\$)	Cold Start Cost (\$)	Cold Start Hrs (h)	Initial Status (h)
1	455	150	0.00048	16.19	150	8	8	4500	9000	5	8
2	455	150	0.00031	17.26	970	8	8	5000	10000	5	8
3	130	20	0.00200	16.60	700	5	5	550	1100	4	-5
4	130	20	0.00211	16.50	680	5	5	560	1120	4	-5
5	162	25	0.00398	19.70	450	6	6	900	1800	4	-6
6	80	20	0.00712	22.26	370	3	3	170	340	2	-3
7	85	25	0.00079	27.74	480	3	3	260	520	2	-3
8	55	10	0.00413	25.92	660	1	1	30	60	0	-1
9	55	10	0.00222	27.27	665	1	1	30	60	0	-1
10	55	10	0.00173	27.79	670	1	1	30	60	0	-1

Sumber : 12th Annual Conference of Power & Energy Society, IEE of Japan, “*Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method*”.

Tabel 4.2
Data Beban Sistem Untuk Validasi

JAM	Beban Sistem (MW)	Reserve (MW)
01.00	700	70
02.00	750	75
03.00	850	85
04.00	950	95
05.00	1000	100
06.00	1100	110
07.00	1150	115
08.00	1200	120
09.00	1300	130
10.00	1400	140
11.00	1450	145
12.00	1500	150
13.00	1400	140
14.00	1300	130
15.00	1200	120
16.00	1050	105
17.00	1000	100
18.00	1100	110
19.00	1200	120
20.00	1400	140
21.00	1300	130
22.00	1100	110
23.00	900	90
24.00	800	80

Sumber : 12th Annual Conference of Power & Energy Society, IEE of Japan, "Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method".

Gambar 4.2
Tampilan Software metode Evolutionary Programming dan Lagrange relaxation Untuk Uji Validasi



Gambar 4.5
 Hasil perhitungan pembebanan untuk Uji Validasi

	Jam 1	Jam 2	Jam 3	Jam 4	Jam 5	Jam 6	Jam 7	Jam 8
Gen 1	455	455	455	455	455	455	455	455
Gen 2	245	295	265	235	265	360	410	455
Gen 3	0	0	0	130	130	130	130	130
Gen 4	0	0	130	130	130	130	130	130
Gen 5	0	0	0	0	0	25	25	30
Gen 6	0	0	0	0	0	0	0	0
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	0	0	0	0	0	0	0	0

Perbandingan hasil perhitungan antara komputasi program dan perhitungan data adalah sebagai berikut :

Gambar 4.6
 Hasil validasi pada jurnal “12th Annual Conference of Power & Energy Society, IEE of Japan, “Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method”.

	Biaya PLN	Biaya UC	Selisi
1	0	13.683	-13.6
2	0	14.554	-14.5
3	0	17.452	-17.4
4	0	19.812	-19.8
5	0	20.133	-20.1
6	0	24.187	-24.1
7	0	23.262	-23.2
8	0	24.150	-24.1
9	0	28.115	-28.1
10	0	30.102	-30.1
11	0	31.961	-31.9
12	0	33.869	-33.8
13	0	30.042	-30.0
14	0	27.255	-27.2
15	0	24.150	-24.1

Pelanggaran Daya
 kWatt

Pelanggaran Up-Down Time
 Hour

Total Biaya EPLR
 566.244

Total Biaya PLN
 0

Selisi Total Biaya
 -566.244

Lama Eksekusi
 0:0:0.271
 (jam:menit:detik.mdetik)

Dari hasil pengujian disini dapat dilihat bahwa program tersebut layak untuk digunakan karena hasil dari perhitungan program tersebut mendekati dari hasil yang ada pada jurnal. Pada tampilan gambar dibawah ini dapat dilihat bahwa hasil perhitungan dari program EP menghasilkan biaya 566.442 \$ yang mendekati perhitungan dengan jurnal “12th Annual Conference of Power & Energy Society, IEE of Japan, “Unit Commitment using Evolutionary Programming Combine with Lagrange Relaxation Method”. yaitu 566.244 \$ dengan error 0,0003 atau 0.03% sehingga program ini valid untuk digunakan.

4.5. Hasil Perhitungan dan Analisa Data

Program optimasi pembebanan unit pembangkit termal pada sistem PT. Pembangkitan Jawa Bali dengan kombinasi metode *Evolutionary Programming* dan *Lagrange relaxation*, terdiri dari tiga tahap yang secara keseluruhan secara berurutan sebagai berikut :

1. Tahap *input* data dengan inialisasi data karakteristik tiap unit dan beban tiap jam.
2. Melakukan pencarian nilai yang paling minimum dari kombinasi ketiga fungsi keanggotaan yang masing-masing untuk biaya, beban sistem dan cadangan berputar.
3. Dari hasil tahap diatas, dilakukan pencarian nilai yang paling maksimum (pencarian nilai yang merupakan perpotongan dari ketiga fungsi

keanggotaan). Kemudian dilakukan pencarian jejak kebelakang dari setiap tahap yang sudah dilakukan, sehingga didapat kombinasi unit-unit pembangkit yang paling murah (optimal) untuk melayani beban sistem.

Seluruh unit termal yang siap beroperasi dalam PT. Pembangkitan Jawa Bali terdiri dari 38 unit pembangkit, yaitu sebagai berikut:

- Unit 1 PLTU Paiton 1
- Unit 2 PLTU Paiton 2
- Unit 3 PLTGU Gresik GT.1.1.
- Unit 4 PLTGU Gresik GT.1.2.
- Unit 5 PLTGU Gresik GT.1.3.
- Unit 6 PLTGU Gresik GT.1.0.
- Unit 7 PLTGU Gresik GT.2.1.
- Unit 8 PLTGU Gresik GT.2.2.
- Unit 9 PLTGU Gresik GT.2.3.
- Unit 10 PLTGU Gresik GT.1.0.
- Unit 11 PLTGU Gresik GT.3.1.
- Unit 12 PLTGU Gresik GT.3.2.
- Unit 13 PLTGU Gresik GT.3.3.
- Unit 14 PLTGU Gresik GT.1.0.
- Unit 15 PLTU Gresik 1
- Unit 16 PLTU Gresik 2

- Unit 17 PLTU Gresik 3
- Unit 18 PLTU Gresik 4
- Unit 19 PLTG Gresik 1
- Unit 20 PLTG Gresik 2
- Unit 21 PLTG Gresik 3
- Unit 22 PLTG Gilitimur 1
- Unit 23 PLTG Gilitimur 2
- Unit 24 PLTGU Muara karang GT 1.1.
- Unit 25 PLTGU Muara karang GT 1.2.
- Unit 26 PLTGU Muara karang GT 1.3.
- Unit 27 PLTGU Muara karang GT 1.0.
- Unit 28 PLTGU Muara Tawar GT 1.1.
- Unit 29 PLTGU Muara Tawar GT 1.2.
- Unit 30 PLTGU Muara Tawar GT 1.3.
- Unit 31 PLTGU Muara Tawar GT 2.1.
- Unit 32 PLTGU Muara Tawar GT 2.2.
- Unit 33 PLTGU Muara Tawar GT 1.0.
- Unit 34 PLTU Muara karang 1
- Unit 35 PLTU Muara karang 2
- Unit 36 PLTU Muara karang 3
- Unit 37 PLTU Muara karang 4

tersebut dipergunakan sebagai bahan perbandingan dengan hasil kombinasi metode *Evolutionary Programming dan Lagrange Relaxation*.

Tabel 4.3
Kombinasi Unit Pembangkit Pada PT. PJB
Kamis, 4 Desember 2003

Jam	Status <i>ON</i> dan <i>OFF</i>	Beban(MW)
1	11000100010001111100000000100100111111	3205
2	11000100010001111100000000100100111111	3035
3	11000100010001111100000000100100111111	3035
4	11000100010001111100000000100100111111	3085
5	11000100010001111100000000100100111111	3290
6	11000100010001111100000000100100111111	2880
7	11000100010001111100000000100100111111	2790
8	11000100010001111100000000100100111111	3220
9	11000100010001111100000000100100111111	3275
10	11000100010001111100000000100100111111	3275
11	11000100010001111100000000100100111111	3275
12	11000100010001111100000000100100111111	3195
13	11000100010001111100000000100100111111	3210
14	11000100010001111100000000100100111111	3260
15	11000100010001111100000000110100111111	3357
16	11000100010001111100000000110100111111	3447
17	11000100010001111100000000100100111111	3525
18	11000100010001111100000000100100111111	3650
19	11000100010001111100000000100100111111	3820
20	11000100010001111100000000100100111111	3370
21	11000100010001111100000000100100111111	3540
22	11000100010001111100000000100100111111	3360
23	11000100010001111100000000100100111111	3345
24	11000100010001111100000000100100111111	3205

Keterangan : 1 = *ON* dan 0 = *OFF*

Tabel 4.4
Kombinasi Unit Pembangkit Temal Pada PT. PJB
Sabtu, 6 Desember 2003

Jam	Status <i>ON</i> dan <i>OFF</i>	eban(MW)
1	11000100010001111100000000100100111111	3140
2	11000100010001111100000000100100111111	3000
3	11000100010001111100000000100100111111	3000
4	11000100010001111100000000100100111111	3000
5	11000100010001111100000000100100111111	3110
6	11000100010001111100000000100100111111	2712
7	11000100010001111100000000100100111111	2682
8	11000100010001111100000000100100111111	3020
9	11000100010001111100000000100100111111	3105
10	11000100010001111100000000100100111111	3105
11	11000100010001111100000000100100111111	3105
12	11000100010001111100000000100100111111	3025
13	11000100010001111100000000100100111111	2890
14	11000100010001111100000000100100111111	2849
15	11000100010001111100000000100100111111	2806
16	11000100010001111100000000100100111111	2804
17	11000100010001111100000000100100111111	2814
18	11000100010001111100000000100100111111	3700
19	11000100010001111100000000100100111111	3700
20	11000100010001111100000000100100111111	3300
21	11000100010001111100000000100100111111	3345
22	11000100010001111100000000100100111111	3135
23	11000100010001111100000000100100111111	3110
24	11000100010001111100000000100100111111	3095

Keterangan : 1 = *ON* dan 0 = *OFF*

Tabel 4.5
Kombinasi Unit Pembangkit Termal Pada PT. PJB
Minggu, 7 Desember 2003

Jam	Status <i>ON</i> dan <i>OFF</i>	Beban (MW)
1	11000100010001111100000000000100111111	2956
2	11000100010001111100000000000100111111	2860
3	11000100010001111100000000000100111111	2860
4	11000100010001111100000000000100111111	2869
5	11000100010001111100000000000100111111	2896
6	11000100010001111100000000000100111111	2640
7	11000100010001111100000000000100111111	2450
8	11000100010001111100000000000100111111	2520
9	11000100010001111100000000000100111111	2620
10	11000100010001111100000000000100111111	2620
11	11000100010001111100000000000100111111	2620
12	11000100010001111100000000000100111111	2570
13	11000100010001111100000000000100111111	2570
14	11000100010001111100000000000100111111	2545
15	11000100010001111100000000100100111111	2587
16	11000100010001111100000000100100111111	2587
17	11000100010001111100000000000100111111	2862
18	11000100010001111100000000000100111111	3757
19	11000100010001111100000000000100111111	3757
20	11000100010001111100000000000100111111	3757
21	11000100010001111100000000000100111111	3442
22	11000100010001111100000000000100111111	3392
23	11000100010001111100000000000100111111	3267
24	11000100010001111100000000000100111111	3122

Keterangan : 1 = *ON* dan 0 = *OFF*

Mengenai hasil kombinasi penjadwalan setelah komputasi optimasi unit pembangkit termal selengkapnya dapat dilihat pada table 4.6, 4.7, dan 4.8.

Tabel 4.6
 Kombinasi Penjadwalan Unit Termal Pada PT. PJB
 Dengan Kombinasi Metode *Evolutionary Programming dan Lagrange Relaxation*
 Kamis, 4 Desember 2003

Jam	Status <i>ON</i> dan <i>OFF</i> pembangkit	Beban (MW)
1	11111111111111111111111100111100000100111	3205
2	111111111111111111111111100111100000100111	3035
3	1111111111111111111111111100111100000100111	3035
4	111111111111111111111111100111100000100111	3085
5	1111111111111111111111111100111100000100111	3290
6	1111111111111111111111111100111100000100111	2880
7	1111111111111111111111111100111100000100111	2790
8	1111111111111111111111111100111100000100111	3220
9	1111111111111111111111111100111100000100111	3275
10	1111111111111111111111111100111100000100111	3275
11	1111111111111111111111111100111100000100111	3275
12	1111111111111111111111111100111100000100111	3195
13	1111111111111111111111111100111100000100111	3210
14	1111111111111111111111111100111100000100111	3260
15	1111111111111111111111111100111100000110111	3357
16	1111111111111111111111111100111100000111111	3447
17	1111111111111111111111111100111101000111111	3525
18	1111111111111111111111111100111101100111111	3650
19	1111111111111111111111111100111101110111111	3820
20	1111111111111111111111111100111101110111111	3820
21	1111111111111111111111111100111101110111111	3590
22	1111111111111111111111111100111101110111111	3410
23	1111111111111111111111111100111101110111111	3410
24	1111111111111111111111111100111101110111111	3270

Keterangan : 1 = *ON* dan 0 = *OFF*

Tabel 4.7
 Kombinasi Penjadwalan Unit Termal Pada PT. PJB
 Dengan Kombinasi Metode *Evolutionary Programming* dan *Lagrange Relaxation*
 Sabtu, 6 Desember 2003

Jam	Status ON dan OFF pembangkit	Beban (MW)
1	1111111111111111111100000111100000100011	3140
2	11111111111111111111100000111100000100011	3000
3	11111111111111111111100000111100000100011	3000
4	11111111111111111111100000111100000100011	3000
5	11111111111111111111100000111100000100011	3110
6	11111111111111111111100000111100000100011	2712
7	11111111111111111111100000111100000100011	2682
8	11111111111111111111100000111100000100011	3020
9	11111111111111111111100000111100000100011	3105
10	11111111111111111111100000111100000100011	3105
11	11111111111111111111100000111100000100011	3105
12	11111111111111111111100000111100000100011	3025
13	11111111111111111111100000111100000100011	2890
14	11111111111111111111100000111100000100011	2849
15	11111111111111111111100000111100000100011	2806
16	11111111111111111111100000111100000100011	2804
17	11111111111111111111100000111100000100011	2814
18	111111111111111111111100111101100111111	3700
19	111111111111111111111100111101100111111	3700
20	111111111111111111111100111101100111111	3300
21	111111111111111111111100111101100111111	3345
22	111111111111111111111100111101100111111	3135
23	111111111111111111111100111101100111111	3110
24	111111111111111111111100111101100111111	3095

Keterangan : 1 = ON dan 0 = OFF

Tabel 4.8
Kombinasi Penjadwalan Unit Termal Pada PT. PJB
 Dengan Kombinasi Metode *Evolutionary Programming dan Lagrange Relaxation*
 Minggu, 7 Desember 2003

Jam	Status ON dan OFF pembangkit	Beban (MW)
1	110111011111111111100000111100000100011	2956
2	110111011111111111100000111100000100011	2860
3	110111011111111111100000111100000100011	2860
4	110111011111111111100000111100000100011	2869
5	110111011111111111100000111100000100011	2896
6	11010101111001111100000111100000100011	2640
7	11010101111001111100000111100000100011	2450
8	11010101111001111100000111100000100011	2520
9	11010101111001111100000111100000100011	2620
10	11010101111001111100000111100000100011	2620
11	11010101111001111100000111100000100011	2620
12	11010101111001111100000111100000100011	2570
13	11010101111001111100000111100000100011	2570
14	11010101111001111100000111100000100011	2545
15	11010101111001111100000111100000100011	2587
16	11010101111001111100000111100000100011	2587
17	11111111111111111111100111101100111111	2862
18	111111111111111111111100111101100111111	3757
19	111111111111111111111100111101100111111	3757
20	111111111111111111111100111101100111111	3757
21	111111111111111111111100111101100111111	3442
22	111111111111111111111100111101100111111	3392
23	111111111111111111111100111101100111111	3267
24	111111111111111111111100111101100111111	3122

Keterangan : 1 = ON dan 0 = OFF

Gambar 4.7
Hasil optimasi pada tanggal 4 Desember 2003

	Biaya PLN	Biaya UC	Selisi
1	781.204.639	629.537.646	151.6
2	741.054.874	594.953.225	146.1
3	741.054.874	594.804.097	146.2
4	748.438.388	602.264.194	146.1
5	809.418.829	640.420.336	168.9
6	722.632.365	566.524.949	156.1
7	696.745.625	558.359.335	138.9
8	801.869.811	625.589.544	176.2
9	812.383.597	635.017.989	177.3
10	812.383.597	634.156.658	178.2
11	812.383.597	638.471.711	173.9
12	797.476.311	621.983.853	175.4
13	803.923.951	625.163.393	178.7
14	811.307.465	630.934.148	180.3
15	861.165.315	664.062.006	197.1

Pelanggaran Daya
 kWatt
 Pelanggaran Up-Down Time
 Hour
 Total Biaya EPLR
 15.680.803.610
 Total Biaya PLN
 19.668.568.812
 Selisih Total Biaya
 3.987.765.201
 Lama Eksekusi
 0:0:0:932
 (jam:menit:detik:mdetik)

Gambar 4.8
Hasil optimasi pada tanggal 6 Desember 2003

	Biaya PLN	Biaya UC	Selisi
1	733.672.317	602.147.331	131.5
2	694.079.607	571.135.285	122.9
3	694.079.607	571.135.285	122.9
4	694.079.607	571.135.285	122.9
5	714.089.019	589.397.321	124.6
6	650.750.134	527.960.048	122.7
7	646.344.618	522.024.881	124.3
8	701.787.816	570.090.117	131.6
9	714.846.727	582.534.918	132.3
10	714.846.727	584.592.176	130.2
11	714.846.727	586.668.789	128.1
12	700.858.429	573.375.432	127.4
13	681.580.859	553.983.521	127.5
14	671.888.666	545.645.893	126.2
15	660.923.317	539.250.249	121.6

Pelanggaran Daya
 kWatt
 Pelanggaran Up-Down Time
 Hour
 Total Biaya EPLR
 14.403.234.713
 Total Biaya PLN
 17.196.334.926
 Selisih Total Biaya
 2.793.100.213
 Lama Eksekusi
 0:0:0:981
 (jam:menit:detik:mdetik)

Gambar 4.9
Hasil optimasi pada tanggal 7 Desember 2003

	Biaya PLN	Biaya UC	Selisih
1	686.090.040	564.685.044	121.4
2	672.676.463	532.510.202	140.1
3	672.676.463	533.893.007	138.7
4	680.535.993	535.235.696	145.3
5	683.936.632	539.265.255	144.6
6	628.773.489	482.014.674	146.7
7	592.701.983	451.521.944	141.1
8	600.979.539	456.718.988	144.2
9	617.779.194	471.588.177	146.1
10	617.779.194	473.076.766	144.7
11	617.779.194	471.588.177	146.1
12	610.372.910	466.083.039	144.2
13	610.372.910	466.083.039	144.2
14	606.701.076	462.365.703	144.3
15	644.121.019	468.611.911	175.5

Summary Panel:

- Pelanggaran Daya: kWatt
- Pelanggaran Up-Down Time: Hour
- Total Biaya EPLR: 13.241.323.177
- Total Biaya PLN: 16.379.589.172
- Selisih Total Biaya: 3.138.265.995
- Lama Eksekusi: 0:0:952 (jam:menit:detik:mdetik)

Buttons: EPLR, Close

Setelah mendapatkan hasil optimal seperti yang sudah dijelaskan diatas, selanjutnya dilakukkann perhitungan total biaya operasi pembangkitan dalam tiap jam, dan juga perhitungan biaya operasi dalam sistem PT. PJB sebagai perbandingan, yang diperoleh dari hasil perhitungan fungsi obyektif dengan memperhatikan data pembebanan harian PT. PJB. Hasil perhitungan kedua biaya operasi tersebut untuk periode satu jam dapat dilihat pada table 4.9, 4.10 dan 4.11.

Tabel 4.9
Perbandingan Biaya Tiap Jam Pada PT. PJB Dan
Kombinasi Metode Evolutionary Programming dan Lagrange Relaxation
Kamis, 4 Desember 2003

Jam	PT. PJB	<i>Kombinasi Metode EP dan LR</i>
00.00-01.00	766.761.138	659.178.245
01.00-02.00	705.621.586	659.178.245
02.00-03.00	705.621.586	659.178.245
03.00-04.00	719.391.999	659.178.245
04.00-05.00	801.061.977	678.562.632
05.00-06.00	685.123.630	586.939.077
06.00-07.00	648.335.634	567.580.782
07.00-08.00	778.544.533	662.570.029
08.00-09.00	792.479.469	675.116.263
09.00-10.00	792.479.469	675.116.263
10.00-11.00	792.479.469	675.116.263
11.00-12.00	779.497.649	656.924.361
12.00-13.00	780.598.672	660.307.379
13.00-14.00	794.364.084	681.871.709
14.00-15.00	847.649.083	703.595.835
15.00-16.00	863.101.025	701.351.410
16.00-17.00	849.444.762	741.825.331
17.00-18.00	873.033.012	775.210.522
18.00-19.00	920.727.099	819.209.151
19.00-20.00	905.112.311	807.452.927
20.00-21.00	848.997.762	754.978.245
21.00-22.00	802.277.589	714.727.688
22.00-23.00	798.842.450	709.508.176
23.00-24.00	767.137.625	668.062.612

Tabel 4.10
Perbandingan Biaya Tiap Jam Pada PT. PJB Dan
Kombinasi Metode Evolutionary Programming dan Lagrange Relaxation
Sabtu, 6 Desember 2003

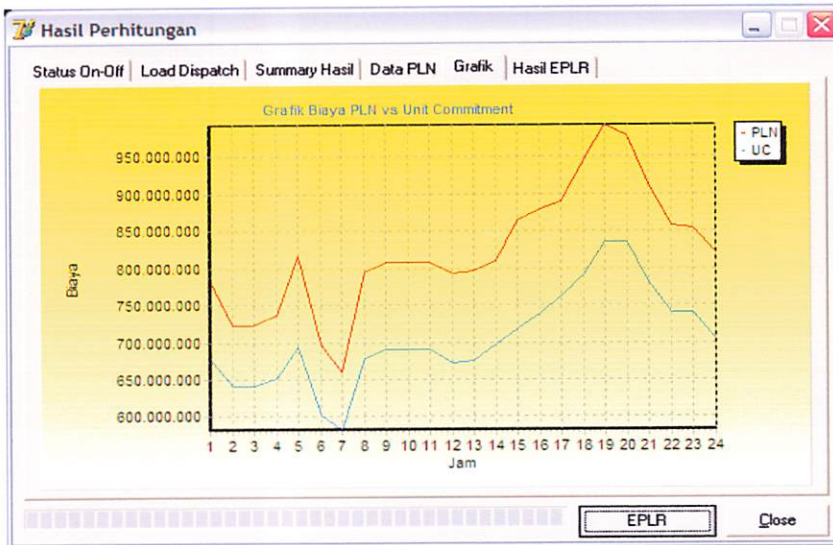
Jam	PT. PJB	<i>Kombinasi Metode EP dan LR</i>
00.00-01.00	691.638.642	649.822.174
01.00-02.00	650.798.164	598.083.333
02.00-03.00	650.798.164	598.083.333
03.00-04.00	650.798.164	598.083.333
04.00-05.00	671.406.787	623.769.318
05.00-06.00	580.200.580	533.637.457
06.00-07.00	573.930.257	527.151.491
07.00-08.00	651.574.541	602.705.636
08.00-09.00	676.667.742	622.583.041
09.00-10.00	676.667.742	622.583.041
10.00-11.00	676.667.742	622.583.041
11.00-12.00	652.700.100	603.863.744
12.00-13.00	610.941.757	572.984.013
13.00-14.00	603.311.599	563.081.798
14.00-15.00	593.737.356	554.256.305
15.00-16.00	592.592.377	556.855.506
16.00-17.00	595.125.135	565.498.660
1700 -18.00	860.662.437	789.865.254
18.00-19.00	860.662.437	786.877.214
19.00-20.00	857.209.298	583.596.250
20.00-21.00	738.355.865	703.376.277
21.00-22.00	672.056.559	605.772.928
22.00-23.00	666.324.087	649.164.306
23.00-24.00	662.870.948	644.797.068

Tabel 4.11
Perbandingan Biaya Tiap Jam Pada PT. PJB Dan
Kombinasi Metode Evolutionary Programming dan Lagrange Relaxation
Minggu, 7 Desember 2003

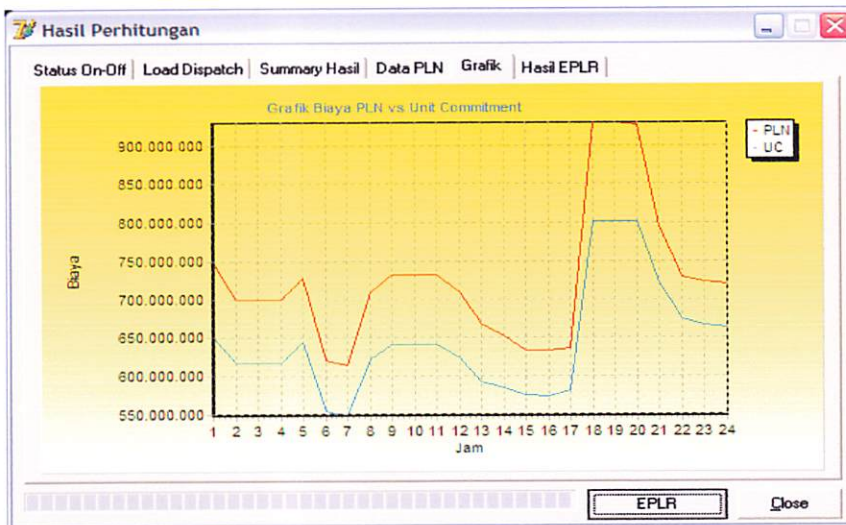
Jam	PT. PJB	<i>Kombinasi Metode EP dan LR</i>
00.00-01.00	634.728.535	603.612.604
01.00-02.00	600.829.874	562.394.570
02.00-03.00	600.829.874	563.394.295
03.00-04.00	596.839.874	563.381.289
04.00-05.00	596.839.869	560.927.613
05.00-06.00	606.563.839	532.897.613
06.00-07.00	557.453.338	503.197.413
07.00-08.00	525.667.561	461.431.993
08.00-09.00	533.942.121	476.354.405
09.00-10.00	556.382.614	498.665.098
10.00-11.00	556.382.614	498.665.098
11.00-12.00	556.382.614	487.436.592
12.00-13.00	540.767.826	487.436.592
13.00-14.00	540.767.826	481.877.132
14.00-15.00	535.496.448	491.237.890
15.00-16.00	576.784.931	491.237.890
16.00-17.00	596.703.277	531.807.113
1700 -18.00	857.932.002	783.371.909
18.00-19.00	857.932.002	783.371.909
19.00-20.00	857.932.002	783.371.909
20.00-21.00	748.621.005	711.101.901
21.00-22.00	729.317.467	678.252.080
22.00-23.00	692.381.047	672.696.822
23.00-24.00	656.068.702	638.082.205

Perbandingan hasil optimalisasi total biaya operasi pembangkitan sebagai fungsi daya beban dari proses Kombinasi Metode *Evolutionary Programming* dan *Lagrange Relaxation* pada PT. PJB dapat dilihat pada grafik 4.1, 4.2 dan 4.3.

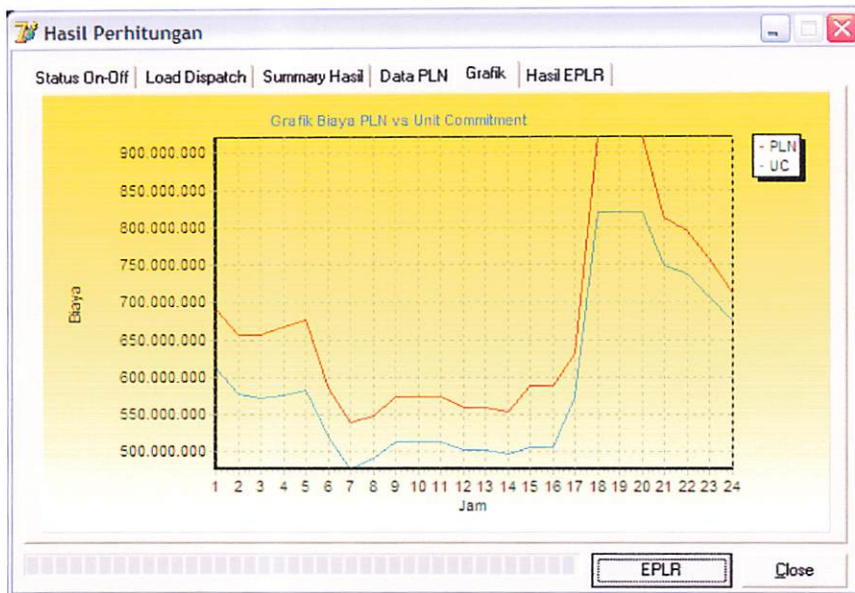
Grafik 4.1
Kurva Fungsi Pembebanan Terhadap Biaya PT. PJB Dan Kombinasi Metode *Evolutionary Programming* dan *Lagrange Relaxation* Kamis, 4 Desember 2003



Grafik 4.2
Kurva Fungsi Pembebanan Terhadap Biaya PT. PJB Dan Kombinasi Metode *Evolutionary Programming* dan *Lagrange Relaxation* Sabtu, 6 Desember 2003



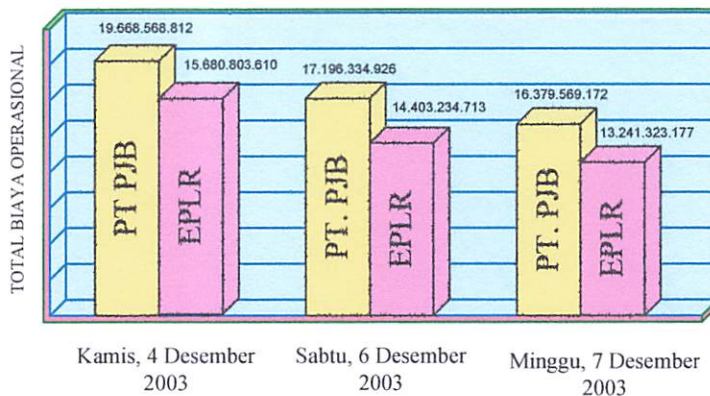
Grafik 4.3
Kurva Fungsi Pembebanan Terhadap Biaya PT. PJB Dan
Kombinasi Metode *Evolutionary Programming dan Lagrange Relaxation*
Minggu, 7 Desember 2003



Tabel 4.12
Perbandingan Total Biaya Dari PT.PJB Dan Total Biaya
Dengan Kombinasi Metode *Evolutionary Programming* dan *Lagrange relaxation*

Tanggal	Biaya PT. PJB (Rupiah)	Biaya EPLR (Rupiah)	Persentase selisih
4 Desember 2003	19.668.568.812	15.680.803.610	20%
6 Desember 2003	17.196.334.926	14.403.234.713	16%
7 Desember 2003	16.379.589.172	13.241.323.177	19%

Dari tabel 4.12 perbandingan biaya operasional untuk tiap jam selama 24 jam, dapat dibuat perbandingan biaya total operasional untuk periode 24 jam (satu hari) antara biaya total operasional selama 24 jam milik PT. PJB dengan biaya total operasional selama 24 jam hasil optimasi dengan metode EPLR.



Tabel 4.13
Selisih Hasil Perhitungan Optimasi PT.PJB Dengan
Kombinasi Metode *Evolutionary Programming* dan *Lagrange Relaxation*
Tanggal 4,6 dan 7 Desember 2003

Tanggal	Selisih Biaya PT. PJB dengan EPLR
4 Desember 2003	3.987.765.201
6 Desember 2003	2.793.234.713
7 Desember 2003	3.138.265.995

Bila diperhatikan tabel 4.13 diatas, tampak bahwa apabila dihitung dengan kedua cara, yaitu antara biaya yang dikeluarkan oleh PT. PJB dengan biaya hasil optimasi dengan menggunakan metode EPLR terdapat perbedaan, ini berarti hasil yang diperoleh optimasi dengan menggunakan metode EPLR menghasilkan penghematan yang relatif besar, sehingga tujuan utama dari optimasi bisa dikatakan sudah terpenuhi.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari hasil analisa program dan hasil perhitungan terhadap penggunaan Kombinasi Metode *Evolutionary Programming (EP) dan Lagrange Relaxation (LR)* pada komitmen unit atau penjadwalan unit pembangkit tenaga terhadap beban yang ditanggung PT. Pembangkitan Jawa Bali (PJB) pada tanggal 4, 6 dan 7 Desember 2003, maka dapat diambil kesimpulan :

Proses Kombinasi Metode EP dan LR memberikan sebuah analisis penyelesaian yang cukup efektif dalam mengoptimalkan pembebanan dan sekaligus menghasilkan penghematan biaya total operasional PT PJB. Adapun hasil perhitungan biaya total pembangkitan.

1. Total biaya operasional pada tanggal 4 Desember 2003 pada PT. PJB sebesar Rp 19.018.667.011 sedangkan pada metode EPLR sebesar Rp 16.552.747.696, dengan selisih Rp 2.465.921.316 atau penghematan sebesar 11%. Untuk tanggal 6 Desember 2003 pada PT. PJB sebesar Rp 176.117.698.478 sedangkan pada metode EPLR sebesar Rp 14.828.816.891 dengan selisih Rp 1.288.881.588 atau penghematan sebesar 11%. Untuk tanggal 7 Desember 2003 pada PT. PJB sebesar Rp 15.093.498.316 sedangkan dengan metode EPLR sebesar Rp 13.852.796.154 dengan selisih Rp 1.240.699.162 atau penghematan sebesar 10%.

2. Dari proses eksekusi perhitungannya komputer dengan Kombinasi Metode EPLR didapatkan waktu eksekusi sebagai berikut:
 - Pada tanggal 4 Desember 2003 waktu eksekusi 1.332 detik
 - Pada tanggal 6 Desember 2003 waktu eksekusi 1.222 detik
 - Pada tanggal 7 Desember 2003 waktu eksekusi 1.191 detik
3. Memungkinkan untuk menerapkan Kombinasi Metode EP dan LR pada PT. PJB karena biaya total yang dihasilkan lebih ekonomis dan mempercepat waktu perhitungannya.

5.2 Saran

Berdasarkan kesimpulan di atas, diajukan saran yang berhubungan dengan skripsi ini, antara lain :

1. Dengan melihat analisa Kombinasi Metode EPLR ini dapat diterapkan pada sistem tenaga listrik PT. PJB yang sudah terinterkoneksi seperti sistem tenaga listrik se-Jawa-Bali karena hasil perhitungannya yang dilakukan pada tanggal 4,6 dan 7 Desember 2003 memberikan biaya total pembangkitan yang lebih murah dari PT. PJB dengan waktu yang relatif singkat.
2. Sebagai studi lanjut, Kombinasi metode EP dan LR ini dapat dikembangkan dengan menambahkan masalah keandalan sistem pembangkitan, sistem pembangkitan hidro dan memasukkan proses *Combined Cycle* pada PLTGU secara lengkap.
3. Perlu dilakukan koordinasi antara mahasiswa dengan pihak PT. PJB agar didapat data yang bukan data per blok dan angkanya tidak dibulatkan.

DAFTAR PUSTAKA

- [1]. Allan J. Wood and Wollenberg B.F., "*Power Generation, Operation, and Control*", John Wiley & Sons, Inc, 1986.
- [2]. Brian T. Luke, "*Overview of Evolutionary Programming Methods*", btluke@aol.com 3/8/2004.
- [3]. Djiteng Marsudi, Ir, "*Operasi Sistem Tenaga Listrik*", Balai Penerbit dan Humas ISTN, 1990.
- [4]. David B. Fogel, "*Evolutionary Optimazion*", Orincon Corporation 9363 Town Centre Drive, San Diego, CA 92121, IEEE, 1992.
- [5]. Don Johnson, "*The Gaussian Random Variable*", By The Connexions Project
- [6]. K.A. Juste, "*An Evolutionary Programming Solution to the Commitment Problem*", IEEE Trans On Power Syst, Vol. 14, No. 4, Nov 1999.
- [7]. P. Attaviriyapap, H.Kita, E. Tanaka, J. Hasegawa, "*Unit Commitment Using Evolutionary Programming Combine With Lagrange Relaxation Method*", 12th Annual Conference of Power & Energy Society, IEE of Japan.
- [8]. P. Attaviriyapap, H.Kita, E. Tanaka, J. Hasegawa, "*A Hybrid Evolutionary Programming for Solving Thermal Unit Commitment*", 12th Annual Conference of Power & Energy Society, IEE of Japan.
- [9]. P. Attaviriyapap, H.Kita, E. Tanaka, J. Hasegawa, "*A Hybrid EP and SQP for Dynamic Economic Dispatch With Fuel Cost Function*", IEEE Trans On Power Syst, Vol. 17, No. 2, May 2002.
- [10]. Talib S. Hussain, "*An Introduction to Evolutionary Computation*", Departement of Computing and Information Science Queen's University, Kingston.
- [11]. William M. Spears, Kenneth A. De Jong, Thomas Back, David B. Fogel, Hugo de Garis, "*An Overview of Evolutionary Computation*", Europe Conference on Machine Learning.
- [12]. Y.J. Cao – Q.H. Wu – D.W. Shimming, "*Study of Initial Population In Evolutionary Programming*" Departement of EEE

- [13]. <http://zoolland.alife.org/hhg2ec/>. "What's Evolutionary Programming (EP)?"

LAMPIRAN



INSTITUT TEKNOLOGI NASIONAL MALANG
FAKULTAS TEKNOLOGI INDUSTRI
JURUSAN TEKNIK ELEKTRO

BERITA ACARA UJIAN SKRIPSI FAKULTAS TEKNOLOGI INDUSTRI

Nama : INA SUNARYANTININGSIH
NIM : 99.12.094
NIRM : -
Jurusan : TEKNIK ELEKTRO S-1
Konsentrasi : TEKNIK ENERGI LISTRIK
Judul Skripsi :

PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL
MENGUNAKAN METODE KOMBINASI EVOLUTIONARY
PROGRAMMING DAN LAGRANGE RELAXATION PADA
PT. PEMBANGKITAN JAWA BALI

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

Hari : Senin
Tanggal : 28 Maret 2005
Dengan Nilai : 86,25 (A) *Si ~*

Panitia Ujian Skripsi



Ketua Majelis Penguji

Ir. Mochtar Asroni, MSME
NIP. Y. 101 8100 036

Sekretaris Majelis Penguji

Ir. F. Yudi Limpraptono, MT
NIP. Y. 103 9500 274

Anggota Penguji

Penguji Pertama

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

Penguji Kedua

Ir. Widodo Pudji M. MT
NIP.P. 102 870 0171



LEMBAR BIMBINGAN SKRIPSI

1. Nama : INA SUNARYANTININGSIH
2. NIM : 99.12.094
3. NIRM : -
4. Jurusan : TEKNIK ELEKTRO
5. Konsentrasi : TEKNIK ENERGI LISTRIK
6. Judul Skripsi :

**PENENTUAN KOMITMEN UNIT PEMBANGKIT TERMAL
DENGAN MENGGUNAKAN KOMBINASI METODE
EVOLUTIONARY PROGRAMMING Dan LAGRANGE
RELAXATION PADA PT. PEMBANGKITAN JAWA BALI**

7. Tanggal Mengajukan Skripsi : 12 Mei 2004
8. Tanggal Menyelesaikan Skripsi : 28 Maret 2005
9. Dosen Pembimbing : Ir. H. Almizan Abdullah, MSEE
10. Telah dievaluasi dengan nilai : 90,00 (A)

Malang, April 2005

Mengetahui
Ketua Jurusan Teknik Elektro

Ir. F. Yudi Limpraptono, MT
NIP. Y. 103 9500 274

Disetujui
Dosen Pembimbing

Ir. H. Almizan Abdullah, MSEE.
NIP.P. 103 9000 208



FORMULIR PERBAIKAN SKRIPSI

Nama : INA SUNARYANTININGSIH
NIM : 99.12.094
Masa Bimbingan : 12 Mei 2004 – 28 Maret 2005
Judul Skripsi : PENENTUAN KOMITMEN UNIT PEMBANGKIT
TERMAL MENGGUNAKAN METODE
KOMBINASI *EVOLUTIONARY PROGRAMMING*
DAN *LAGRANGE RELAXATION* PADA PT.PJB

No.	Tanggal	Uraian	Paraf
1.	8-04-2005	Pada kesimpulan tidak perlu ada tabel.	
2.	8-04-2005	Tabel tersebut dimasukkan dalam analisis hasil	

Disetujui

Penguji Pertama

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

Penguji Kedua

Ir. Widodo Pudji M. MT
NIP. 102 8900 202

Mengetahui,
Dosen Pembimbing

Ir. H. Almizan Abdullah, MSEE
NIP. Y. 130 9000 208



FORMULIR BIMBINGAN SKRIPSI

Nama : INA SUNARYANTININGSIH
 Nim : 99.12.094
 Masa Bimbingan : 12 Mei 2004 s/d 12 Nopember 2004
 Judul Skripsi : PENENTUAN KOMITMEN UNIT PEMBANGKIT
 TERMAL MENGGUNAKAN KOMBINASI METODE
 EVOLUTIONARY PROGRAMMING dan LAGRANGE
 RELAXATION Pada PT. PLN PJB

No.	Tanggal	Uraian	Paraf Pembimbing
1.	14-06-04	1. Untuk PLGU data SI diambil dari CC 1.1.1 2. Notasi/kode harus konsisten 3. Purnus - Inunus tdk perlu diulang-ulang	
2.	27-07-04	1. Bab II : perlu diwaspadai bahwa optimasi dan di pada EP berbeda dg GA 2. Apakah dapat replikasi pengujian di bab III spt pada GA?	
3.	03-08-04	1. Dataran OC dalam (1,0). berarti harus jd proses merobak jadi integer dulu. 2. Perlu dipelajari lagi algoritmanya!	
4.		Masalahnya bagaimana pembentukan kromosom dari EP?	
5.	09-03-05	1. Notasi harus harus konsisten dan mudah standar di jurusan T.E. 2. Flow Chart pada Bab II adalah ber	
6.		dasar dari Algoritma metode/konsep selengkapnya di Bab III adalah benar sudah Algoritma Program! lengkap!	
7.		3. Apakah betul ada pengembangan keadaa ope- rasional oleh PLN? Jelaskan!	
8.	20-03-05	1. Kumpulan grafik PJB berbeda dg Jember 2. Tabel 4.12. tambahi kolom lebih (Rp 2%)	
9.		3. Dalam kesimpulan juga, lengkapi tabel tapi tulis kan urut dan bentuk kalimat	
10.			

Malang, 30-04-2005

Dosen Pembimbing,

Ir. H. ALMIZAN ABDULLAH, MSEE

Form.S-4b



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

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

PERSERO) MALANG
K NIAGA MALANG

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

Malang, 25 Mei 2004

Nomor : ITN-502/I.FTI/2/2004
Jumlah Lembaran : satu lembar
Perihal : **BIMBINGAN SKRIPSI**

Kepada : Yth. Sdr. Ir. H. ALMIZAN ABDULLAH, MSEE
Dosen Institut Teknologi Nasional
di -
Malang

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

Nama : INA SUNARYANTININGSIH
Nim : 9912094
Fakultas : Teknologi Industri
Jurusan : Teknik Elektro
Konsentrasi : T. Energi Listrik (S-1)

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

12 Mei 2004 s/d 12 November 2004

Adapun tugas tersebut merupakan salah satu syarat untuk memperoleh gelar Sarjana Teknik, Jurusan Teknik Elektro.

Demikian atas perhatian serta kerjasama yang baik kami ucapkan terima kasih,



Ketua
Jurusan Teknik Elektro

[Signature]
Ir. I Made Wartana, MT
Nip. 131 991 182

Dasaran :

2. Mahasiswa yang bersangkutan
2. Arsip.

Form. S-4a

RENCANA HARIAN: KAMIS, 4 DESEMBER 2003

SUB SISTEM REGION 1

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

Jenis	Jumlah	Jam																									
		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		
P-35	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P-36	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P-37	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
P-38	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

SUB SISTEM REGION 1

RENCANA : HARIANTANGCAL : 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	Rata-2	
PLTU	WARGA	4	153	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150
	MARING	4	16	155	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	163
	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
	MTWAR20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTWAR30C	0	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	WARGA	4	153	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150
	MARING	4	16	155	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	163
	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
	MTWAR20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTWAR30C	0	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	WARGA	4	153	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150
	MARING	4	16	155	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	163
	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
	MTWAR20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTWAR30C	0	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

PLTU

Rata-2

SUB SYSTEM REGION_2

RENCANA : HARI/TANGGAL : KAMIS, 4 DESEMBER 2003
PT. INDONESIA POWER

Unit	Jenis	Jam																								
		00.30	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	P1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P2	127	127	103	108	108	108	108	108	109	108	90	80	80	82	91	107	120	125	125	125	125	110	89	82
		P3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
		P4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTD	PMJNG	P1	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	25	
		P2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		P3	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
		P4	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTD	DIRJAT	P1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P2	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	
		P3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P4	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	P1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		P8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
WYNDU	URJAT	P1	135	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	105	
		P2	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
		P3	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	35	
		P4	374	574	565	550	555	555	547	556	581	556	552	527	527	609	659	742	788	816	816	816	816	748	686	670
Perumahan	Olarita Jaluhur	P1	1689	1756	1878	1707	1699	1718	1693	1737	1730	1874	1572	1449	1372	1505	1556	1624	1648	1655	1661	1361	1661	1640	1566	1545
		P2	-1115	1132	-1112	-1157	-1144	-1183	-1138	-1174	-1165	-1118	-1030	-922	-845	-886	-897	-882	-877	-839	-845	-845	-845	-892	-880	-875
		P3	13	13	12	10	11	11	11	11	13	11	11	8	8	16	21	30	32	25	25	25	25	30	24	22
		P4	48	48	56	72	87	67	67	65	11	66	70	95	95	108	263	180	154	106	124	106	106	174	236	-252

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

PEMBELIAN DARI LUAR PLN

(*) Perumahan Area 2
(**) Beban Area 2
Setelah (*) (*)
Cadangan Sehari
Cadangan Pulih

SUB SISTEM REGIONAL 2

HARITANGGAL : KAMIS, 4 DESEMBER 2003
PT. INDONESIA POWER

Peta	Area 2	Jam												Rata-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	24.30	Rata-2		
PETA	SGLNG	#1	100	105	125	99	99	99	119	115	108	112	127	121	103	103	103	100	105	105	100	99	99	99	99	102	76	
		#2	100	105	125	99	99	99	120	115	109	112	127	121	103	103	103	100	100	100	100	0	0	0	0	0	97	
		#3	0	0	0	0	0	0	0	80	107	112	128	121	103	103	103	100	100	100	100	100	99	99	99	99	103	30
		#4	0	0	0	0	0	0	0	0	80	100	104	119	113	103	103	102	83	0	0	0	0	0	0	0	0	17
PETA	KWJNG	#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	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	
		#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	
PETA	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	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	
		#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	

		Jam												Rata-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	24.30	Rata-2	
PETA	CRATA	#1	0	0	80	80	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	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	100	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	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
PETA	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
		#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	35	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	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
PETA	Pembangkitan Area 2	#1	148	157	177	805	805	805	805	805	846	876	1153	1306	1443	1479	1395	1394	1151	959	837	607	725	645	632	806	
		#2	1603	1600	1626	1621	1647	1661	1640	1676	1707	1867	2158	2234	2256	2245	2232	2210	2187	2057	1971	1947	1858	1827	1819	1762	
		#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	32	31	43	36	36	36	40	42	57	82	84	100	98	89	89	88	88	66	47	38	35	28	70	21	34
PETA	Cedangan Selain Cedangan Luar PLN	#1	174	165	170	742	242	242	242	242	201	330	657	752	635	574	599	683	584	537	304	251	240	197	152	144	
		#2	1603	1600	1626	1621	1647	1661	1640	1676	1707	1867	2158	2234	2256	2245	2232	2210	2187	2057	1971	1947	1858	1827	1819	1762	
		#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	32	31	43	36	36	36	40	42	57	82	84	100	98	89	89	88	88	66	47	38	35	28	70	21	34

(*) Pembangkitan Area 2

(**) Beban Area 2

(***) Selisih (+) (-)

Cedangan Selain Cedangan Luar PLN

SUB SISTEM REGION_4

HARI/TANGGAL: KAMIS , 4 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

RENCANA:

Jam	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	Res-b-2
PLTA Area 4, SUTAM	21	90	90	21	21	21	21	21	23	23	23	23	23	23	23	23	23	23	21	21	21	21	21	20
PLTA BRANTAS	93	93	93	93	93	93	93	93	116	116	116	116	116	116	116	116	118	86	86	86	86	86	86	84
PLTU PITON #1	350	350	350	350	350	350	370	370	370	370	370	370	370	370	370	370	350	350	350	350	350	350	350	357
PLTU PITON #2	350	350	350	350	350	350	350	370	370	350	370	370	370	370	370	370	350	350	350	350	350	350	350	356
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
GRSIK10C1	0	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	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
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
GRSIK11C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK12C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK13C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK10C2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK20C2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK220C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK230C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK21C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK22C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK23C	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	275	275	300	300	300	300	300	300	300	300	300	300	350	350	350	350	300	300	250	250	250	250	250	300
GRSIK10C3	0	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	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	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
GRSIK31C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK32C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GRSIK33C	275	275	300	300	300	300	300	300	300	300	300	300	350	350	350	300	300	300	250	250	250	250	250	300
PLTU #3	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	165	110
PLTU #4	90	90	90	90	90	90	140	140	140	140	140	140	165	165	165	165	165	165	165	165	165	165	165	100
PLTU #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	54
PLTU #2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	54
PLTG #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
PLTG #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
PLTG #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
Area 1: Membangkitkan Area-4	1654	1654	1654	1654	2864	2864	2904	2904	2904	2904	2904	2904	3204	3204	3204	3204	3204	3204	3204	3204	3204	3204	3204	3204
Area-4	1154	1321	1354	1354	1814	1814	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827	1827
Sekolah (*)	1375	1223	1270	1270	1250	1307	1298	1277	1270	1273	1127	1370	1456	1500	1512	1310	1424	1488	1457	1442	1487	1386	1263	1271
Cadangan Sektoral	78	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88	88
Cadangan PLTB	1044	1008	1089	1089	1144	1144	1144	1144	1094	1044	877	82	82	82	82	82	82	82	382	507	807	729	764	

12/10/2003 01:10:00 AM

**RENCANA :
HARJANGGAL: MINGGU, 7 DESEMBER 2003
PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI**

SUB SISTEM RESI0N_1

		12:00	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	Rata-2	
PLTU	MKRNGJCC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MKRNGJCC6	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-TU	MKRNGJCC7	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	300	
MNRNG	MNRNG	71	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	MNRNG1	42	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
	MNRNG2	73	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTU	MTRVAR10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	MTRVAR10C7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTRVAR10C8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTRVAR10C9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MTRVAR10C10	0	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																										
PLTU	MTRVAR11C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GSLAR	GSLAR	44	182	1,2	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	182	
	GSLAR1	75	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
CIBRANG	CIBRANG	50	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	CIBRANG1	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	Pemanggihan Aera-1	3887	3037	3037	3037	3039	3798	3798	3798	3078	4084	4284	4567	4893	4989	4992	4992	4992	4888	4888	4888	4881	4816	4737	4357	
	Bahan Aera-1	4031	4031	4159	4160	4128	4053	4070	4226	4395	4481	5158	5018	5025	5030	5010	4879	5002	4173	4842	4792	4576	4451	4398	4277	
	Salur-1 (1")	164	96	222	181	179	114	91	142	101	86	166	145	132	137	127	14	114	15	46	89	205	365	332	29	
	Salur-1 (1/2")	17	155	155	167	167	164	167	162	160	188	43	43	43	43	43	43	43	43	50	50	50	50	50	87	
	Salur-1 (2")																									
	Salur-1 (3")																									
	Salur-1 (4")																									
	Salur-1 (5")																									
	Salur-1 (6")																									
	Salur-1 (7")																									
	Salur-1 (8")																									
	Salur-1 (9")																									

SUB SISTEM REGION_2

RENCANA : HARI/TANGGAL : MINGGU , 7 DESEMBER 2003

PT. INDONESIA POWER

Jam	00.30	01.30	02.30	03.30	04.30	05.30	06.30	07.30	08.30	09.30	10.30	11.30	12.30
PLTA	24	24	24	24	24	24	24	24	24	24	24	24	24
PLTA	122	102	104	105	108	121	121	80	80	80	111	111	111
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	122	102	104	104	107	121	121	80	80	80	111	111	111
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	25	25	25	25	25	25	25	25	25	25	25	25	25
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	50	50	50	50	50	50	50	50	50	50	50	50	50
PLTA	50	50	50	50	50	50	50	50	50	50	50	50	50
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	18	18	18	18	18	18	18	18	18	18	18	18	18
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

PLTA	100	107	100	100	100	100	100	100	90	80	100	100	100
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0
PLTA	0	0	0	0	0	0	0	0	0	0	0	0	0

PEMBELIAN DARI LUAR PLN

WYNDU	105	105	105	105	105	105	105	105	105	105	105	105	105
DRJAT	90	90	90	90	90	90	90	90	90	90	90	90	90
Perum Listrik Jatinuhur	45	45	45	45	45	45	45	45	45	45	45	45	45

Perumahan Area 2	751	713	718	724	715	716	716	722	707	759	668	647	617
Area 2	1224	1674	1666	1677	1628	1625	1580	1675	1695	1651	1463	1362	1321
Sekeloa (*) (*)	-973	-963	955	953	-913	-909	-958	-928	-936	-992	-795	-697	-674
Cawanggi Sekeloa	34	31	30	32	31	31	32	30	35	35	26	24	24
Cabangan Pullar	131	166	171	158	167	166	170	175	172	173	214	217	235

SUB SYSTEM REGION_2

RENCANA : HARI/TANGGAL : MINGGU, 7 DESEMBER 2003

PT. INDONESIA POWER

Jam	12.00	13.30	15.00	16.30	18.00	19.30	21.00	22.30	23.30	24.00	24.00	
PLTA Area 2	24	24	24	24	24	33	33	33	24	24	24	26
PLTA ECLING	35	112	152	87	92	103	104	104	80	0	0	91
	0	0	0	0	0	103	104	104	104	107	107	32
	0	0	0	0	80	103	104	104	104	106	107	61
	36	93	102	87	91	103	104	104	80	0	0	58
PLTP KAMPUNG	25	25	25	25	25	25	25	25	25	25	25	25
	0	0	0	0	0	50	50	50	50	50	50	14
	50	50	50	50	50	50	50	50	50	50	50	50
PLTP DRJAT	50	50	50	50	50	50	50	50	50	50	50	50
PLTP SRACI	0	0	0	0	0	18	18	18	18	18	0	5
	18	18	18	18	18	18	18	18	18	18	18	18
	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	18	18	18	18	18	18	4
PLTA GRATA	80	100	80	80	80	80	80	80	80	80	80	87
	0	0	0	0	0	30	80	80	80	80	0	19
	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	80	80	80	60	0	0	15
	0	0	0	0	0	80	80	80	0	0	0	11
	0	0	0	0	0	80	80	80	0	0	0	11
	0	0	0	0	0	30	60	60	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
	105	105	105	105	105	105	105	105	105	105	105	105
	90	90	90	90	90	90	90	90	90	90	90	90
	45	45	45	45	45	50	50	50	50	45	45	46
WYNDU	0	0	0	0	0	0	0	0	0	0	0	0
DRJAT	0	0	0	0	0	0	0	0	0	0	0	0
PLTA Otomotif Jabaluhur	683	733	691	661	670	712	846	1056	1341	1475	1599	1387
	1295	1249	1235	1286	1334	1308	1532	1704	1911	2005	2121	2116
	617	515	544	625	673	690	686	648	640	630	722	729
	25	28	31	28	25	26	31	42	60	84	89	81
	221	193	193	221	212	212	354	442	607	582	558	570
	1106	1106	1106	1106	1106	1106	1106	1106	1106	1106	1106	1106
	1139	1139	1139	1139	1139	1139	1139	1139	1139	1139	1139	1139
	1874	1874	1874	1874	1874	1874	1874	1874	1874	1874	1874	1874
	-665	-665	-665	-665	-665	-665	-665	-665	-665	-665	-665	-665
	84	84	84	84	84	84	84	84	84	84	84	84
	443	443	443	443	443	443	443	443	443	443	443	443
	768	768	768	768	768	768	768	768	768	768	768	768
	1815	1815	1815	1815	1815	1815	1815	1815	1815	1815	1815	1815
	-1047	-1047	-1047	-1047	-1047	-1047	-1047	-1047	-1047	-1047	-1047	-1047
	29	29	29	29	29	29	29	29	29	29	29	29
	101	101	101	101	101	101	101	101	101	101	101	101
	724	724	724	724	724	724	724	724	724	724	724	724
	1682	1682	1682	1682	1682	1682	1682	1682	1682	1682	1682	1682
	-948	-948	-948	-948	-948	-948	-948	-948	-948	-948	-948	-948
	28	28	28	28	28	28	28	28	28	28	28	28
	107	107	107	107	107	107	107	107	107	107	107	107

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

PEMBELIAN DARI LUAR PLN

(*) Pembangkitan Area-2
 (**) Bahan Area-2
 Salusin (***)
 C. danjun Seterina
 C. danjun P. U. L. A. r

SUB SISTEM REGION_4

RENCANA : HARI/TANGGAL : MINGGU, 7 DESEMBER 2003

PT PLN PEMBANGKITAN TENAGA LISTRIK JAWA-BALI

Area	PLTA	Jam															
		01.00	02.00	03.00	04.00	05.00	06.00	07.00	08.00	09.00	10.00	11.00	12.00				
PLTA SUTAMI BRANTAS	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19	19
	50	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90
	63	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68	68
PLTU PITOH #1 PITOH #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	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	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	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	370	370
PLTU 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	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	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	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	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	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	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	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	370	370	370	370
PLTU 3	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200	200
PLTU GRSIF #3 #1 #2 #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	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
PLTC GLTR #1 #2 #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	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
 (**) Briban Area-4
 (***) Sekeloa (*) (***)
 Cendayan Sekeloa
 Cadangan Pulpar

PLTU	MKGNG10C	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		
PLTU	MKGNG10C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG20C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG30C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG40C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG50C	0	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	MKGNG60C	350	370	360	350	350	350	350	350	350	350	350	400	425	425	425	425	425	400	400	400	400	400	400	400	400
	MKGNG70C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG80C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG90C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	MKGNG100C	0	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	MKGNG110C	30	50	90	90	90	90	90	90	90	90	90	90	135	150	150	150	150	150	150	150	150	150	150	150	150
	MKGNG120C	150	150	150	150	150	150	150	150	150	150	150	150	150	165	165	165	165	165	165	165	165	165	165	165	165
	MKGNG130C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG140C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG150C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG160C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG170C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG180C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG190C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG200C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG210C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG220C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG230C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG240C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG250C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG260C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG270C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85
	MKGNG280C	50	50	50	50	50	50	50	50	50	50	50	50	85	85	85	85	85	85	85	85	85	85	85	85	85

PEMBELIAN DARI LUAR PLN

PLTU	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM	KGALUM
PLTU	KGALUM	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
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	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	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	KGALUM	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0

10/09/03 dan 01/12/03 manual

SUB SISTEM REGION_2

RENCANA : HARI/TANGGAL: SABTU, 6 DESEMBER 2003
PT. INDONESIA POWER

Jam	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	Rate-2	
PLTA Area 2	24	24	24	24	24	24	24	24	24	24	24	24	33	33	33	33	33	33	33	33	33	24	24	26	
PLTA SGLUNG #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
PLTA SGLUNG #2	102	117	57	37	90	30	31	32	35	101	115	121	126	120	102	107	102	102	101	102	102	100	103	103	31
PLTA SGLUNG #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	92
PLTA SGLUNG #4	101	115	37	87	80	80	31	30	35	102	114	121	126	120	107	103	101	100	100	100	100	100	102	102	31
PLTP KAJUNG #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
PLTP KAJUNG #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
PLTP KAJUNG #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
PLTP KAJUNG #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
PLTP KAJUNG #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
PLTP KAJUNG #6	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
PLTP KAJUNG #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
PLTP KAJUNG #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
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	22
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	0	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	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	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	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	0	0
PLTA CRATA #7	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	70	13
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	80	41
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
DRUJAT #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	90
Perum.Dionta Jatiukur	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	45	46
(*) Pembangunan Area 2	760	790	731	731	717	717	719	719	717	747	1051	1286	1381	1441	1417	1336	1276	1213	1066	971	836	687	692	692	844
(**) Babun Area 2	1441	1483	1410	1494	1402	1412	1444	1405	1391	1380	1361	2101	2115	2108	2083	2000	1975	1868	1820	1804	1759	1693	1693	1645	1645
Sedihan (*) - (**)	-681	-693	-679	-763	-685	-695	-725	-628	-614	-725	-675	-720	-714	-691	-747	-721	-762	-799	-849	-370	-1072	-1001	-1001	-800	-800
Cadangan Sotetiba	35	38	32	37	31	31	31	31	31	32	40	84	92	98	96	89	82	76	61	51	42	28	29	29	43
Cadangan Putar	247	217	276	274	290	290	238	250	28	727	612	526	466	490	511	631	565	463	386	319	195	150	189	189	247

	Jan	00.30	01.00	01.30	02.00	02.30	03.00	03.30	03.50	04.00	04.30	04.50	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						
PETA Area 4	ASIK10C	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	19	19	19				
	SUTAMI	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				
	BRANTAS	68	68	68	68	68	54	68	68	68	68	68	68	68	68	68	68	68	68	68	73	73	73	73	73	73	73	73	73				
	PLTN P1	370	370	350	350	350	350	350	350	350	350	350	350	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370				
PETA 2	ASIK110C	370	370	350	350	350	350	350	350	350	350	350	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370				
	ASIK10C1	370	370	350	350	350	350	350	350	350	350	350	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370	370				
	ASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	ASIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	ASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	ASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	ASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
PETA 3	ASIK130C	250	250	250	250	250	250	250	250	250	250	275	275	275	275	200	200	225	250	250	250	250	250	250	250	250	250	250	250				
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0				
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
	ASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0			
PETA 4	ASIK #1	165	165	165	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	165	165	165	165	165	165	165	165	165			
	ASIK #1	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	90			
	ASIK #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	50	50	50	50	50		
PETA 5	ASIK #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			
	ASIK #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		
	ASIK #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		
PETA 6	ASIK #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			
	ASIK #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		
	ASIK #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	
	ASIK #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	
	ASIK #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	
	ASIK #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	0
	ASIK #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	0
	ASIK #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	0
	ASIK #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	0
	ASIK #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	0

(*) Pembangkitan Area 4
 (*) Beban Area 4
 (*) Selisih (*) (*)
 Cadenang Saktela
 Cadenang Putera
 13/09/2003 08:27:36 AM/PLN

PT PLN PEMBAKANGKITAN TENAGA LISTRIK JAWA-BALI

Jum 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 4.	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20	
PLTA SUTAMAH	50	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
PLTA BRAWITAS	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	93	
PLTU PITON	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	
PITON #1	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	
PITON #2	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	360	
PLTU 2	GASIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK115C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK215C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK220C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK230C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK215C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK225C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK310C	180	180	164	164	164	164	164	164	164	164	164	164	164	164	164	164	164	164	164	164	164
GASIK315C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK30C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK20C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK310C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK320C	250	250	250	250	250	250	240	200	250	300	300	300	300	300	300	300	300	300	300	300	250	
GASIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU 3	GASIK110C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK10C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK20C1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK120C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK130C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK115C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK215C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK220C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK230C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK210C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK215C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK225C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	GASIK310C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
GASIK315C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK30C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK20C3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK320C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
GASIK330C	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
PLTU 4	GASIK #3	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	150	
	GASIK #4	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	90	
	GASIK #1	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
GASIK #2	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	50	
PLTU 5	GASIK #1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK #2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
	GASIK #3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	

```

unit TypDatUC;

interface

uses TypDatGA, LagrangeRelaxation, EvoP, Komplex;

type
  TString30=string[30];
  TGen=record
    Nama:string;
    Pmax, Pmin, a0, a1, a2, Sh, Sc:double;
    tup, tdown, tcold, iniSt:integer;
  end;
  TLoad=record
    Load, Res:double;
  end;
  GenArr=array of TGen;
  LoadArr=array of TLoad;

  TPembangkit=class
  private
    Nama:string;
    Pmax, Pmin, a0, a1, a2, Sh, Sc, Daya, AFLC:double;
    tup, tdown, tcold, iniSt:integer;
  public
    function GetNama:string;
    function GetPmax:double;
    function GetPmin:double;
    function Geta0:double;
    function Geta1:double;
    function Geta2:double;
    function GetSh:double;
    function GetSc:double;
    function GetTup:integer;
    function GetTdown:integer;
    function GetTcold:integer;
    function GetIniSt:integer;
    function GetBiaya(Daya:double):double;
    function GetAFLC:double;
    function GetDaya:double;
    procedure SetNama(dNama:string);
    procedure SetPmax(dPmax:double);
    procedure SetPmin(dPmin:double);
    procedure Seta0(da0:double);
    procedure Seta1(da1:double);
    procedure Seta2(da2:double);
    procedure SetSh(dSh:double);
    procedure SetSc(dSc:double);
    procedure SetTup(dTup:integer);
    procedure SetTdown(dTdown:integer);
    procedure SetTcold(dTcold:integer);
    procedure SetIniSt(dIniSt:integer);
    property GenNama:string read GetNama write SetNama;
    property GenPMax:double read GetPmax write SetPmax;
    property GenPMin:double read GetPmin write SetPmin;
    property Gena0:double read Geta0 write Seta0;
    property Genal:double read Geta1 write Seta1;
    property Genal2:double read Geta2 write Seta2;
    property GenSh:double read GetSh write SetSh;
    property GenSc:double read GetSc write SetSc;
    property GenTup:integer read GetTup write SetTup;
    property GenTDown:integer read GetTdown write SetTdown;
    property GenTCold:integer read GetTcold write SetTcold;
    property GenIniSt:integer read GetIniSt write SetIniSt;
    property GenDaya:double read GetDaya;
  end;

  ArrGen=array of TPembangkit;

  TUnitCommitment=class
    Generator:ArrGen;
    LoadHour:LoadArr;
    lmd, myu, Arrqlambda, ArrJkres, CostPerHour:Arr1;
    Daya, PL:Arr2;
    chrom:Tchromosome;
    Xs:iArr1;
    Ngene, Hour, PinHour, iterasi, ite:integer;

```

```

qlamda, Jkres, alpha, alpha2, err, PinLoad:double;
function GetNgen:integer;
function GetHour:integer;
function GetAlpha:double;
function GetAlpha2:double;
function GetErr:double;
function GetIterasi:integer;
function GetFuelCost(chrom:Tchromosome1):double;
function GetStartUpCost(PrevChrom,Chrom:Tchromosome1):double;
procedure SetNgen(dNgen:integer);
procedure SetHour(dHour:integer);
procedure SetAlpha(dAlpha:double);
procedure SetAlpha2(dAlpha2:double);
procedure SetErr(dErr:double);
procedure SetIterasi(dIterasi:integer);
procedure Inisialisasi;
procedure InisialisasiXs;
procedure InisialisasiLmdMyu;
procedure SetEcoDis(Load,Res:double;chrom:Tchromosome1;
var Status:boolean);
procedure HitungStartUpCost(NoGen,Xsb:integer;
var SUC:double);
procedure DynamicProgramming1(NoGen,Xsb:integer);
procedure DynamicProgramming;
procedure UpdateLamdaMyu;
procedure HitungQLamda;
procedure HitungJkres;
procedure LagrangeRelaxation;
procedure HitungPinalty;
procedure HitungBiayaPerJam;
property Ngenerator:integer read GetNgen write SetNgen;
property Nhour:integer read GetHour write SetHour;
property AlphaPositive:double read GetAlpha write SetAlpha;
property AlphaNegative:double read GetAlpha2 write SetAlpha2;
property Error:double read GetErr write SetErr;
end;

```

```

var NamaFile:string;
Ngen,Nhour:integer;
LoadAct,ResAct:double;
Gen:GenArr;
LoadH:LoadArr;
UComm:TUnitCommitment;
Xsb:iArr1;

```

```

implementation
//constructor
constructor TLagrange.Create;
begin
inherited create;
FNgen:=0;
FNjam:=0;
FPinalty:100000000;
FIntersiMax:=500;
FError:=0.01;
end;

```

```

function TPembangkit.GetNama:string;
begin
Result:=Nama;
end;

```

```

function TPembangkit.GetPmax:double;
begin
Result:=Pmax;
end;

```

```

function TPembangkit.GetPmin:double;
begin
Result:=Pmin;
end;

```

```

function TPembangkit.Geta0:double;
begin
Result:=a0;
end;

```

```

function TPembangkit.Geta1:double;
begin
    Result:=a1;
end;

function TPembangkit.Geta2:double;
begin
    Result:=a2;
end;

function TPembangkit.GetSh:double;
begin
    Result:=Sh;
end;

function TPembangkit.GetSc:double;
begin
    Result:=Sc;
end;

function TPembangkit.GetTup:integer;
begin
    Result:=Tup;
end;

function TPembangkit.GetTdown:integer;
begin
    Result:=Tdown;
end;

function TPembangkit.GetTcold:integer;
begin
    Result:=Tcold;
end;

function TPembangkit.GetIniSt:integer;
begin
    Result:=IniSt;
end;

function TPembangkit.GetBiaya(Daya:double):double;
begin
    Result:=a2*sqr(Daya)+a1*Daya+a0;
end;

function TPembangkit.GetAFLC:double;
begin
    Result:=a0/Pmax+a1+a2/Pmax;
end;

function TPembangkit.GetDaya:double;
begin
    Result:=Daya;
end;

procedure TPembangkit.SetNama(dNama:string);
begin
    Nama:=dNama;
end;

procedure TPembangkit.SetPmax(dPmax:double);
begin
    Pmax:=dPmax;
end;

procedure TPembangkit.SetPmin(dPmin:double);
begin
    Pmin:=dPmin;
end;

procedure TPembangkit.Seta0(da0:double);
begin
    a0:=da0;
end;

procedure TPembangkit.Seta1(da1:double);
begin

```



```

    a1:=da1;
end;

procedure TPembangkit.Seta2(da2:double);
begin
    a2:=da2;
end;

procedure TPembangkit.SetSh(dSh:double);
begin
    Sh:=dSh;
end;

procedure TPembangkit.SetSc(dSc:double);
begin
    Sc:=dSc;
end;

procedure TPembangkit.SetTup(dTup:integer);
begin
    Tup:=dTup;
end;

procedure TPembangkit.SetTdown(dTdown:integer);
begin
    Tdown:=dTdown;
end;

procedure TPembangkit.SetTcold(dTcold:integer);
begin
    Tcold:=dTcold;
end;

procedure TPembangkit.SetIniSt(dIniSt:integer);
begin
    IniSt:=dIniSt;
end;

function TUnitCommitment.GetNgen:integer;
begin
    Result:=Ngene;
end;

function TUnitCommitment.GetHour:integer;
begin
    Result:=Hour;
end;

function TUnitCommitment.GetAlpha1:double;
begin
    Result:=Alpha1;
end;

function TUnitCommitment.GetAlpha2:double;
begin
    Result:=Alpha2;
end;

function TUnitCommitment.GetErr:double;
begin
    Result:=Err;
end;

function TUnitCommitment.GetIterasi:integer;
begin
    Result:=iterasi;
end;

function TUnitCommitment.GetFuelCost(chrom:Tchromosomel):double;
var i:byte;
begin
    Result:=0;
    for i:=1 to Ngene do
        begin
            if chrom[i] then
                begin
                    Result:=Result+Generator[i].GetBiaya(Generator[i].Daya);
                end;
            end;
        end;
end;

```

```

    end;
  end;
end;

function TUnitCommitment.GetStartUpCost(PrevChrom,
    Chrom:Tchromosome1):double;
var i:integer;
begin
  Result:=0;
  for i:=1 to Ngene do
  begin
    if Chrom[i] then
    begin
      if not PrevChrom[i] then
      begin
        Result:=Result+Generator[i].Sh;
      end;
    end;
  end;
end;

procedure TUnitCommitment.SetNgen(dNgen:integer);
begin
  Ngene:=dNgen;
end;

procedure TUnitCommitment.SetHour(dHour:integer);
begin
  Hour:=dHour;
end;

procedure TUnitCommitment.SetAlpha1(dAlpha1:double);
begin
  Alpha1:=dAlpha1;
end;

procedure TUnitCommitment.SetErr(dErr:double);
begin
  Err:=dErr;
end;

procedure TUnitCommitment.SetAlpha2(dAlpha2:double);
begin
  Alpha2:=dAlpha2;
end;

procedure TUnitCommitment.SetIterasi(dIterasi:integer);
begin
  iterasi:=diterasi;
end;

procedure TUnitCommitment.Inisialisasi;
var i:integer;
begin
  SetLength(Generator,Ngene+1);
  SetLength(Xs,Ngene+1);
  SetLength(LoadHour,Hour+1);
  SetLength(Daya,Ngene+1,Hour+1);
  SetLength(chrom,Ngene+1,Hour+1);
  SetLength(PL,Ngene+1,Hour+1);
  SetLength(lmd,Hour+1);
  SetLength(myu,Hour+1);
  SetLength(CostPerHour,Hour+1);
  for i:=1 to Ngene do
  begin
    Generator[i]:=TPembangkit.Create;
  end;
end;

procedure TUnitCommitment.InisialisasiXs;
var i:integer;
begin
  for i:=1 to Ngene do
  begin
    Xs[i]:=Generator[i].iniSt;
    Generator[i].AFLC:=Generator[i].GetAFLC;
  end;
end;

```

```

end;

procedure TUnitCommitment.InisialisasiLmdMyu;
var i:integer;
begin
  SetLength(Lmd, Hour+1);
  SetLength(myu, Hour+1);
  SetLength(ArrQlamda, iterasi+1);
  SetLength(ArrJkres, iterasi+1);
  for i:=1 to Hour do
  begin
    Lmd[i]:=LoadHour[i].Load*Alpha1;
    myu[i]:=(LoadHour[i].Load+LoadHour[i].Res)*Alpha1;
  end;
end;

procedure TUnitCommitment.SetEcoDis (Load, Res:double;
  chrom:Tchromosomel;
  var Status:boolean);
var i, j:integer;
  StateCek:Tchromosomel;
  CekLoadMax, CekLoadMin, LoadCek, Pa, Pb, Lmd, LoadSplit:double;
  diffa2, diffal, Cek:double;
begin
  SetLength(StateCek, Ngene+1);
  Status:=true;
  for i:=1 to Ngene do
  begin
    StateCek[i]:=chrom[i];
    Generator[i].Daya:=0;
  end;
  CekLoadMax:=0;
  CekLoadMin:=0;
  for i:=1 to Ngene do
  begin
    if StateCek[i] then
    begin
      CekLoadMax:=CekLoadMax+Generator[i].Pmax;
      CekLoadMin:=CekLoadMin+Generator[i].Pmin;
    end;
  end;
  LoadCek:=Load+Res;
  if LoadCek>CekLoadMax then
  begin
    Status:=false;
  end
  else if LoadCek<CekLoadMin then
  begin
    Status:=false;
  end;
  if (LoadCek-Res)>CekLoadMax then
  begin
    Status:=false;
    for i:=1 to Ngene do
    begin
      if StateCek[i] then
      begin
        Generator[i].Daya:=Generator[i].Pmax;
      end;
    end;
  end
  else if (LoadCek-Res)<CekLoadMin then
  begin
    Status:=false;
    for i:=1 to Ngene do
    begin
      if StateCek[i] then
      begin
        Generator[i].Daya:=Generator[i].Pmin;
      end;
    end;
  end;
  end
  else
  begin
    LoadCek:=LoadCek-Res;
    LoadSplit:=LoadCek;
    for i:=1 to 15 do

```

```

begin
  Pa:=0;
  Pb:=0;
  Lmd:=0;
  for j:=1 to Ngene do
  begin
    if StateCek[j] then
    begin
      diffa2:=Generator[j].a2*2;
      diffal:=Generator[j].a1;
      Pa:=Pa+1/diffa2;
      Pb:=Pb+diffal/diffa2;
    end;
  end;
  if Pa<>0 then
  begin
    Lmd:=(LoadSplit+Pb)/Pa;
  end
  else
  begin
    Lmd:=LoadSplit+Pb;
  end;
  Cek:=0;
  for j:=1 to Ngene do
  begin
    if StateCek[j] then
    begin
      diffa2:=2*Generator[j].a2;
      diffal:=Generator[j].a1;
      Generator[j].Daya:=(Lmd-diffal)/diffa2;
      if Generator[j].Daya<Generator[j].Pmin then
      begin
        Generator[j].Daya:=Generator[j].Pmin;
      end;
      if Generator[j].Daya>Generator[j].Pmax then
      begin
        Generator[j].Daya:=Generator[j].Pmax;
      end;
    end;
    Cek:=Cek+Generator[j].Daya;
  end;
  Cek:=LoadCek-Cek;
  if Cek=0 then
  begin
    break;
  end;
  if Cek>0 then
  begin
    for j:=1 to Ngene do
    begin
      if StateCek[j] then
      begin
        if Generator[j].Daya=Generator[j].Pmax then
        begin
          StateCek[j]:=false;
          LoadSplit:=LoadSplit-Generator[j].Daya;
          if LoadSplit<0 then
          begin
            LoadSplit:=LoadSplit+Generator[j].Daya;
            StateCek[j]:=true;
          end;
        end;
      end;
    end;
  end;
  else if Cek<0 then
  begin
    for j:=1 to Ngene do
    begin
      if StateCek[j] then
      begin
        if Generator[j].Daya=Generator[j].Pmin then
        begin
          StateCek[j]:=false;
          LoadSplit:=LoadSplit-Generator[j].Daya;
          if LoadSplit<0 then
          begin

```

```

        LoadSplit:=LoadSplit+Generator[j].Daya;
        StateCek[j]:=true;
    end;
end;
end;
end;
end;
end;
end;
end;
end;

procedure TUnitCommitment.HitungStartupCost (NoGen,Xsb:integer;
    var SUC:double);
var i,tcold:integer;
begin
    SUC:=0;
    tcold:=Generator[NoGen].tdown+Generator[NoGen].tcold;
    for i:=1 to Hour do
    begin
        if chrom[NoGen,i] then
        begin
            if Xsb<0 then
            begin
                Xsb:=1;
                if abs(Xsb)>=tcold then
                begin
                    SUC:=SUC+Generator[NoGen].Sc;
                end
                else
                begin
                    SUC:=SUC+Generator[NoGen].Sh;
                end;
            end
            else
            begin
                Xsb:=Xsb+1;
            end;
        end
        else
        begin
            if Xsb>0 then
            begin
                Xsb:=-1;
            end
            else
            begin
                Xsb:=Xsb-1;
            end;
        end;
    end;
end;

procedure TUnitCommitment.DynamicProgramming1 (NoGen,Xsb:integer);
var i,tcold:integer;
    up,down,SUC,FCost,Gen:double;
begin
    tcold:=Generator[NoGen].tdown+Generator[NoGen].tcold;
    //Inisial State Generator
    if Xsb<0 then
    begin
        chrom[NoGen,0]:=false;
    end
    else if Xsb>0 then
    begin
        chrom[NoGen,0]:=true;
    end;
    for i:=1 to Hour do
    begin
        up:=0;
        SUC:=0;
        //Hitung Startup Cost bila State berubah status
        if not chrom[NoGen,i-1] then
        begin
            if abs(Xsb)>tcold then
            begin
                SUC:=Generator[NoGen].Sc;
            end

```

```

else
begin
SUC:=Generator[NoGen].Sh;
end;
end;
//Hitung Daya Pembangkitan dengan constraints  $P_{min}$  dan  $P_{max}$ 
Gen:=(lmd[i]-Generator[NoGen].a1)/(2*Generator[NoGen].a2);
if Gen<Generator[NoGen].Pmin then
begin
Gen:=Generator[NoGen].Pmin;
end
else if Gen>Generator[NoGen].Pmax then
begin
Gen:=Generator[NoGen].Pmax;
end;
FCost:=Generator[NoGen].GetBiaya(Gen);
up:=FCost+SUC-lmd[i]*Gen-myu[i]*Generator[NoGen].Pmax;
FCost:=Generator[NoGen].GetBiaya(Generator[NoGen].Pmin);
down:=FCost-lmd[i]*Generator[NoGen].Pmin-myu[i]*Generator[NoGen].Pmax;
//Constraint uptime dan down time
if up<down then
begin
if Xsb<0 then
begin
if abs(Xsb)>=Generator[NoGen].tdown then
begin
chrom[NoGen,i]:=true;
Xsb:=1;
Daya[NoGen,i]:=Gen;
end
else
begin
chrom[NoGen,i]:=false;
Xsb:=Xsb-1;
Daya[NoGen,i]:=0;
end;
end
else if Xsb>0 then
begin
chrom[NoGen,i]:=true;
Xsb:=Xsb+1;
Daya[NoGen,i]:=Gen;
end;
end
else
begin
if Xsb>0 then
begin
if Xsb>=Generator[NoGen].tup then
begin
chrom[NoGen,i]:=false;
Xsb:=-1;
Daya[NoGen,i]:=0;
end
else
begin
chrom[NoGen,i]:=true;
Xsb:=Xsb+1;
Daya[NoGen,i]:=Gen;
end;
end
else if Xsb<0 then
begin
chrom[NoGen,i]:=false;
Xsb:=Xsb-1;
Daya[NoGen,i]:=0;
end;
end;
end;
end;
end;
end;

procedure TUnitCommitment.DynamicProgramming;
var i,j:integer;
begin
for i:=1 to Ngene do
begin
for j:=1 to Hour do

```

```

begin
    chrom[i,j]:=false;
    Daya[i,j]:=0;
    PL[i,j]:=0;
end;
end;
for i:= 1 to Ngene do
begin
    DynamicProgramming1(i,Xs[i]);
end;
end;

procedure TUnitCommitment.UpdateLamdaMyu;
var i,j:integer;
    diffQLamda,diffQmyu,Load,Loadmyu:double;
begin
    for i:=1 to Hour do
    begin
        diffQLamda:=0;
        Load:=0;
        Loadmyu:=0;
        for j:=1 to Ngene do
        begin
            Load:=Load+Daya[j,i];
            if Daya[j,i]>0 then
            begin
                Loadmyu:=Loadmyu+Generator[j].Pmax;
            end;
        end;
        diffQLamda:=LoadHour[i].Load-Load;
        diffQmyu:=LoadHour[i].Load+LoadHour[i].Res-Loadmyu;
        if diffQLamda>0 then
        begin
            lmd[i]:=lmd[i]+diffQLamda*alpha1;
            myu[i]:=myu[i]+diffQmyu*alpha1;
        end
        else
        begin
            lmd[i]:=lmd[i]+diffQLamda*alpha2;
            myu[i]:=myu[i]+diffQmyu*alpha2;
        end;
    end;
end;

procedure TUnitCommitment.HitungQLamda;
var i,j,tcold:integer;
    TempSUC,SUC,FuelCost,SumLamda,SumMyu,SumPi,SumPimax:double;
    Xsa:iArr1;
begin
    qlamda:=0;
    SetLength(Xsa,Ngene+1);
    SUC:=0;
    FuelCost:=0;
    SumLamda:=0;
    SumMyu:=0;
    for i:=1 to Ngene do
    begin
        Xsa[i]:=Xs[i];
        tcold:=Generator[i].tdown+Generator[i].tcold;
        HitungStartUpCost(i,Xsa[i],TempSUC);
        SUC:=SUC+TempSUC;
        for j:=1 to Hour do
        begin
            if chrom[i,j] then
            begin
                FuelCost:=FuelCost+Generator[i].GetBiaya(Daya[i,j]);
            end;
        end;
    end;
    for i:=1 to Hour do
    begin
        SumPi:=0;
        SumPimax:=0;
        for j:=1 to Ngene do
        begin
            if chrom[j,i] then
            begin

```

```

        SumPi:=SumPi+Daya[j,i];
        SumPimax:=SumPimax+Generator[j].Pmax;
    end;
end;
SumLamda:=SumLamda+lmd[i]*(LoadHour[i].Load-SumPi);
SumMyu:=SumMyu+myu[i]*(LoadHour[i].Load+LoadHour[i].Res-SumPimax);
end;
qlamda:=FuelCost+SUC+SumLamda+SumMyu;
end;

procedure TUnitCommitment.HitungJkres;
var i,j:integer;
    diffQLamda,diffQmyu,Load,LoadMyu,FuelCost,SUC,SUCi:double;
    Keren:double;
    chrom1:Tchromosomel;
    Xsa:iArr1;
    Status:boolean;
begin
    SetLength(Xsa,Ngene+1);
    Jkres:=0;
    FuelCost:=0;
    SUC:=0;
    Keren:=15000000000/Hour;
    for i:=1 to Hour do
    begin
        SetLength(chrom1,Ngene+1);
        for j:=1 to Ngene do
        begin
            chrom1[j]:=chrom[j,i];
        end;
        diffQLamda:=0;
        Load:=0;
        Loadmyu:=0;
        for j:=1 to Ngene do
        begin
            Load:=Load+Daya[j,i];
            if Daya[j,i]<>0 then
            begin
                Loadmyu:=Loadmyu+Generator[j].Pmax;
            end;
        end;
        diffQLamda:=LoadHour[i].Load-Load;
        diffQmyu:=LoadHour[i].Load+LoadHour[i].Res-Loadmyu;
        if (diffQLamda<0) and (diffQmyu<0) then
        begin
            SetEcoDis(LoadHour[i].Load,LoadHour[i].Res,chrom1,Status);
            for j:=1 to Ngene do
            begin
                PL[j,i]:=Generator[j].Daya;
                FuelCost:=FuelCost+Generator[j].GetBiaya(Generator[j].Daya);
            end;
        end
        else
        begin
            FuelCost:=FuelCost+Keren;
            for j:=1 to Ngene do
            begin
                PL[j,i]:=0;
            end;
        end;
        end;
        for i:=1 to Ngene do
        begin
            Xsa[i]:=Xs[i];
            HitungStartUpCost(i,Xsa[i],SUCi);
            SUC:=SUC+SUCi;
        end;
        Jkres:=FuelCost+SUC+Keren;
    end;

//Program Utama
procedure TUnitCommitment.LagrangeRelaxation,EvoP;
var i:integer;
    dsa:double;
begin
    InisialisasiXs;
    InisialisasiLmdMyu;

```



```

ite:=0;
for i:=1 to iterasi do
begin
ite:=ite+1;
DynamicProgramming;
HitungQLamda;
HitungJkres;
Arrqlamda[i]:=qlamda;
ArrJkres[i]:=Jkres;
dsa:=(Jkres-qlamda)/qlamda;
if abs(dsa)<=err then break;
UpdateLamdaMyu;
end;
HitungPinalty;
HitungBiayaPerJam;
end;

procedure TUnitCommitment.HitungPinalty;
var i,j:integer;
    CekLoad:double;
begin
PinHour:=0;
PinLoad:=0;
for i:=1 to Hour do
begin
CekLoad:=0;
for j:=1 to Ngene do
begin
CekLoad:=CekLoad+PL[j,i];
end;
PinLoad:=PinLoad+(LoadHour[i].Load-CekLoad);
end;
end;

procedure TUnitCommitment.HitungBiayaPerJam;
var i,j:integer;
    SumCost:double;
    Xsb:iArr1;
begin
SetLength(Xsb,Hour+1);
for i:=1 to Ngene do
begin
Xsb[i]:=Xs[i];
if Xs[i]>0 then
begin
chrom[i,0]:=true;
end
else
begin
chrom[i,0]:=false;
end;
end;
for i:=1 to Hour do
begin
SumCost:=0;
for j:=1 to Ngene do
begin
if chrom[j,i] then
begin
if chrom[j,i-1] then
begin
Xsb[j]:=Xsb[j]+1;
end
else
begin
Xsb[j]:=-1;
end;
end
else
begin
if chrom[j,i-1] then
begin
Xsb[j]:=1;
end
else
begin
Xsb[j]:=Xsb[j]-1;

```

```
        end;
    end;
    SumCost:=SumCost+Generator[j].GetBiaya(PL[j,i]);
    if (chrom[j,i-1]=false) and (chrom[j,i]=true) then
    begin
        if Abs(Xsb[j])>(Generator[j].tdown+Generator[j].tcold) then
        begin
            SumCost:=SumCost+Generator[j].Sh;
        end;
    end;
    end;
    CostPerHour[i]:=SumCost;
end;
end;
end.
```

```

unit EvolutonariProgramming;

interface
uses Komplex;

type
  TGA=class
  private
    fMaxgen, fPopsiz, fLchrom, fNparam:integer;
    fPmutat, fSelecComp, fPflip, fKa:double;
    function GetMaxgen:integer;
    function GetPopsiz:integer;
    function GetLchrom:integer;
    function GetNparam:integer;
    function GetPmutat:double;
    function GetSelecComp:double;
    function GetPflip:double;
    function GetKa:double;
    procedure SetMaxgen(dMaxgen:integer);
    procedure SetPopsiz(dPopsiz:integer);
    procedure SetLchrom(dLchrom:integer);
    procedure SetNparam(dNparam:integer);
    procedure SetPmutat(dPmutat:double);
    procedure SetSelecComp(dPselecComp:double);
    procedure SetPflip(dPflip:double);
    procedure SetKa(dKa:double);
  public
    constructor Create;
    function GetFlip(const param:double):boolean;
    function GetRandom(const min,max:integer):integer;
    destructor Destroy;override;
    property Maxgen:integer read GetMaxgen write SetMaxgen;
    property Popsiz:integer read GetPopsiz write SetPopsiz;
    property Lchrom:integer read GetLchrom write SetLchrom;
    property Nparam:integer read GetNparam write SetNparam;
    property Pmutat:double read getPmutat write SetPmutat;
    property PselecComp:double read getPselecComp write SetPselecComp;
    property Pflip:double read GetPflip write SetPflip;
    property Ka:double read GetKa write SetKa;
  end;

implementation

//constructor
constructor TEvoP.Create;
begin
  inherited Create;
  fMaxgen:=1;
  fPopsiz:=1;
  fNparam:=1;
  fKa:=1;
end;

//data accessing
function TEvoP.GetMaxgen:integer;
begin
  result:=fMaxgen;
end;

function TEP.GetPopsiz:integer;
begin
  result:=fPopsiz;
end;

function TEP.GetLchrom:integer;
begin
  result:=fLchrom;
end;

function TEP.GetNparam:integer;
begin
  result:=fNparam;
end;

function TEP.GetPmutat:double;
begin

```

```

    result:=fPmutat;
end;

function TEP.GetPselecComp:double;
begin
result:=fPselecComp;
end;

function TEP.GetPflip:double;
begin
    result:=fPflip;
end;

function TEP.GetKa:double;
begin
    result:=fKa;
end;

procedure TEP.SetMaxgen(dMaxgen:integer);
begin
    fMaxgen:=dMaxgen;
end;

procedure TEP.SetPopsize(dPopsize:integer);
begin
    fPopsize:=dpopsize;
end;

procedure TEP.SetLchrom(dLchrom:integer);
begin
    fLchrom:=dLchrom;
end;

procedure TEP.SetNparam(dNparam:integer);
begin
    fNparam:=dNparam;
end;

procedure TEP.SetPmutat(dPmutat:double);
begin
    fPmutat:=dPmutat;
end;

procedure TEP.SetPselecComp(dPselecComp:double);
begin
    fPselecComp:=dPselecComp;
end;

procedure TEP.SetPflip(dPflip:double);
begin
    fPflip:=dPflip;
end;

procedure TEP.SetKa(dKa:double);
begin
    fKa:=dka;
end;

//data processing
function TEP.GetFlip(const param:double):boolean;
var rand:double;
begin
    rand:=random;
    if rand<=param then
        begin
            result:=true;
        end
    else
        begin
            result:=false;
        end;
end;

function TEP.GetRandom(const min,max:integer):integer;
var rand:double;
begin
    rand:=random;
    result:=round(min+rand*(max-min));
end;

```

end;

destructor TEP.Destroy;

begin

 inherited Destroy;

end;

end.

```

unit HitungChrom;

interface

uses Komplex, TypDatEP, UnitGenerator, EcoDispath, SysUtils;

type
  THitungChrom=class
  private
    fNgen, fNjam, fPinTime:integer;
    fGen:TGenArr;
    fBeban, fRes:Arr1;
    fPL:Arr2;
    fFUC, fPinDaya:double;
    fCostPerJam:Arr1;
    fChrom:TChromBin2;
    fEco:TEcoDis;
    function getNgen:integer;
    function getNjam:integer;
    function getGen:TGenArr;
    function getBeban:Arr1;
    function getRes:Arr1;
    function getPL:Arr2;
    function getChrom:TChromBin2;
    procedure setNgen(const dNgen:integer);
    procedure setNjam(const dNjam:integer);
    procedure setGen(const dGen:TGenArr);
    procedure setBeban(const dBeban:Arr1);
    procedure setRes(const dRes:Arr1);
    procedure setPL(const dPL:Arr2);
    procedure setChrom(const dChrom:TChromBin2);
    function getFUC:double;
    function getPinTime:integer;
    function getPinDaya:double;
    function getCostPerJam:Arr1;
  public
    constructor Create;overload;
    constructor Create(const dGen:TGenArr;const dBeban,dRes:Arr1);overload;
    procedure Assign(const dGen:TGenArr;const dBeban,dRes:Arr1);overload;
    procedure doHitung;
    destructor Destroy;override;
    property Ngen:integer read getNgen write setNgen;
    property Njam:integer read getNjam write setNjam;
    property Gen:TGenArr read getGen write setGen;
    property Beban:Arr1 read getBeban write setBeban;
    property Res:Arr1 read getRes write setRes;
    property PL:Arr2 read getPL write setPL;
    property Chrom:TChromBin2 read getChrom write setChrom;
    property FUC:double read getFUC;
    property PinTime:integer read getPinTime;
    property PinDaya:double read getPinDaya;
    property CostPerJam:Arr1 read getCostPerJam;
  end;

implementation

//constructor
constructor THitungChrom.Create;
begin
  inherited Create;
  fNgen:=1;
  fNjam:=1;
end;

constructor THitungChrom.Create(const dGen:TGenArr;const dBeban,dRes:Arr1);
var i,j:integer;
begin
  inherited Create;
  fNgen:=high(dGen);
  fNjam:=high(dBeban);
  SetLength(fGen, fNgen+1);
  SetLength(fBeban, fNjam+1);
  SetLength(fRes, fNjam+1);
  SetLength(fPL, fNgen+1, fNjam+1);
  SetLength(fCostPerJam, fNjam+1);
  for i:=1 to fNgen do
  begin

```

```

    fGen[i]:=TPembangkit.Create(dGen[i]);
end;
for i:=1 to fNjam do
begin
    fBeban[i]:=dBeban[i];
    fRes[i]:=dRes[i];
    fCostPerJam[i]:=0;
end;
fEco:=TEcoDis.Create;
fEco.Gen:=fGen;
fPinTime:=0;
fPinDaya:=0;
fFUC:=0;
for i:=1 to fNgen do
begin
    for j:=1 to fNjam do
    begin
        fPL[i,j]:=0;
    end;
end;
end;

//data accessing
function THitungChrom.getNgen:integer;
begin
    result:=fNgen;
end;

function THitungChrom.getNjam:integer;
begin
    result:=fNjam;
end;

function THitungChrom.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;

function THitungChrom.getBeban:Arr1;
var i:integer;
begin
    SetLength(result,fNjam+1);
    for i:=1 to fNjam do
    begin
        result[i]:=fBeban[i];
    end;
end;

function THitungChrom.getRes:Arr1;
var i:integer;
begin
    SetLength(result,fNjam+1);
    for i:=1 to fNjam do
    begin
        result[i]:=fRes[i];
    end;
end;

function THitungChrom.getPL:Arr2;
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]:=fPL[i,j];
        end;
    end;
end;

function THitungChrom.getChrom:TChromBin2;

```

```

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]:=fChrom[i,j];
        end;
      end;
    end;
end;

procedure THitungChrom.setNgen(const dNgen:integer);
begin
  fNgen:=dNgen;
end;

procedure THitungChrom.setNjam(const dNjam:integer);
begin
  fNjam:=dNjam;
end;

procedure THitungChrom.setGen(const dGen:TGenArr);
var i:integer;
begin
  fNgen:=high(dGen);
  SetLength(fGen,fNgen+1);
  for i:=1 to fNgen do
    begin
      fGen[i]:=TPembangkit.Create(dGen[i]);
    end;
  if high(fBeban)>0 then
    begin
      SetLength(fPL,fNgen+1,high(fBeban)+1);
    end;
  fEco:=TEcoDis.Create;
  fEco.Gen:=fGen;
end;

procedure THitungChrom.setBeban(const dBeban:Arr1);
var i:integer;
begin
  fNjam:=high(dBeban);
  SetLength(fBeban,fNjam+1);
  for i:=1 to fNjam do
    begin
      fBeban[i]:=dBeban[i];
    end;
  if high(fGen)>0 then
    begin
      SetLength(fPL,high(fGen)+1,fNjam+1);
    end;
end;

procedure THitungChrom.setRes(const dRes:Arr1);
var i:integer;
begin
  fNjam:=high(dRes);
  SetLength(fRes,fNjam+1);
  for i:=1 to fNjam do
    begin
      fRes[i]:=dRes[i];
    end;
  if high(fGen)>0 then
    begin
      SetLength(fPL,high(fGen)+1,fNjam+1);
    end;
end;

procedure THitungChrom.setPL(const dPL:Arr2);
var i,j,cekNgen,cekNjam:integer;
begin
  cekNgen:=high(dPL);
  cekNjam:=high(dPL[0]);
  if (cekNgen<>fNgen) or (cekNjam<>fNjam) then
    begin
      raise Exception.Create('Ukuran Matrik tidak sama dengan data!');
    end;
end;

```



```

    end;
    SetLength(fPL, fNgen+1, fNjam+1);
    for i:=1 to fNgen do
    begin
        for j:=1 to fNjam do
        begin
            fPL[i, j]:=dPL[i, j];
        end;
    end;
end;

procedure THitungChrom.setChrom(const dChrom:TChromBin2);
var i, j, cekNgen, cekNjam:integer;
    dPL:Arr1;
    aChrom:TChromBin1;
begin
    cekNgen:=high(dChrom);
    cekNjam:=high(dChrom[0]);
    if (cekNgen<>fNgen) or (cekNjam<>fNjam) then
    begin
        raise Exception.Create('Ukuran Matrik tidak sama dengan data!');
    end;
    SetLength(fChrom, fNgen+1, fNjam+1);
    for i:=1 to fNgen do
    begin
        for j:=1 to fNjam do
        begin
            fChrom[i, j]:=dChrom[i, j];
        end;
    end;
    SetLength(dPL, fNgen+1);
    SetLength(aChrom, fNgen+1);
    for i:=1 to fNjam do
    begin
        for j:=1 to fNgen do
        begin
            aChrom[j]:=fChrom[j, i];
            fPL[j, i]:=0;
        end;
        fEco.Chrom:=aChrom;
        fEco.Beban:=fBeban[i];
        dPL:=fEco.PL;
        for j:=1 to fNgen do
        begin
            fPL[j, i]:=dPL[j];
        end;
    end;
end;

procedure THitungChrom.Assign(const dGen:TGenArr;const dBeban, dRes:Arr1);
var i:integer;
begin
    fNgen:=high(dGen);
    fNjam:=high(dBeban);
    SetLength(fGen, fNgen+1);
    SetLength(fBeban, fNjam+1);
    SetLength(fRes, fNjam+1);
    SetLength(fPL, fNgen+1, fNjam+1);
    for i:=1 to fNgen do
    begin
        fGen[i]:=TPembangkit.Create(dGen[i]);
    end;
    fEco:=TEcoDis.Create;
    fEco.Gen:=fGen;
    for i:=1 to fNjam do
    begin
        fBeban[i]:=dBeban[i];
        fRes[i]:=dRes[i];
    end;
end;

//data processing
procedure THitungChrom.doHitung;
var i, j, Xs, tcold:integer;
    sPmin, sPmax, RampCon, sRampCon:double;
    costGen:Arr2;
    costSUC:Arr2;

```

```

SetLength(fNgen, fNgen+1, fNJam+1);
SetLength(costSUC, fNgen+1, fNJam+1);
SetLength(fCostPerJam, fNJam+1);
fPinTime:=0;
sRampCon:=0;
//mencari Biaya Pembangkitan dan StartUpcost per jam per Pembangkit
for i:=1 to fNgen do
begin
  Xs:=fGen[i].IniSt;
  tcold:=Gen[i].Tdown+fGen[i].Tcold;
  for j:=1 to fNJam do
  begin
    costGen[i,j]:=0;
    costSUC[i,j]:=0;
    if j>1 then
    begin
      RampCon:=fPL[i,j]-fPL[i,j-1];
      if RampCon>0 then
      begin
        if RampCon>fGen[i].Ramp then
        begin
          sRampCon:=sRampCon+(RampCon-fGen[i].Ramp);
        end;
      end;
    end;
    if fPL[i,j]<>0 then
    begin
      costGen[i,j]:=Gen[i].GetBiaya(fPL[i,j]);
      if Xs>0 then
      begin
        Xs:=Xs+1;
      end
      else if Xs<0 then
      begin
        if abs(Xs)<Gen[i].Tdown then
        begin
          fpinTime:=fpinTime+1;
        end;
        if abs(Xs)>tcold then
        begin
          costSUC[i,j]:=Gen[i].Sc;
        end
        else
        begin
          costSUC[i,j]:=Gen[i].Sh;
        end;
        Xs:=1;
      end;
    end
    else if fPL[i,j]=0 then
    begin
      if Xs>0 then
      begin
        if Xs<Gen[i].Tup then
        begin
          fPinTime:=fPinTime+1;
        end;
        Xs:=-1;
      end
      else if Xs<0 then
      begin
        Xs:=Xs-1;
      end;
    end;
  end;
end;
end;
//mencari Total Biaya dan Biaya per jam
fFUC:=0;
for i:=1 to fNJam do
begin
  fCostPerJam[i]:=0;
  for j:=1 to fNgen do
  begin
    fCostPerJam[i]:=fCostPerJam[i]+costGen[j,i]+costSUC[j,i];
  end;
  fFUC:=fFUC+fCostPerJam[i];
end;

```

```

end;
fPinDaya:=0;
for i:=1 to fNJam do
begin
  sPmin:=0;
  sPmax:=0;
  for j:=1 to fNgen do
  begin
    if fPL[j,i]<>0 then
    begin
      sPmin:=sPmin+Gen[j].Pmin;
      sPmax:=sPmax+Gen[j].Pmax;
    end;
  end;
  if (fBeban[i]+fRes[i])<sPmin then
  begin
    fPinDaya:=fPinDaya+abs(sPmin-fBeban[i]+fRes[i]);
  end
  else if (fBeban[i]+fRes[i])>sPmax then
  begin
    fPinDaya:=fPinDaya+abs(fBeban[i]+fRes[i]-sPmax);
  end;
end;
fPinDaya:=fPinDaya+sRampCon;
end;

//data output
function THitungChrom.getFUC:double;
begin
  result:=fFUC;
end;

function THitungChrom.getPinTime:integer;
begin
  result:=fPinTime;
end;

function THitungChrom.getPinDaya:double;
begin
  result:=fPinDaya;
end;

function THitungChrom.getCostPerJam:Arr1;
var i:integer;
begin
  SetLength(result,fNjam+1);
  for i:=1 to fNjam do
  begin
    result[i]:=fCostPerJam[i];
  end;
end;

//destructor
destructor THitungChrom.Destroy;
var i:integer;
begin
  try
    for i:=1 to fNgen do
    begin
      fGen[i].Free;
    end;
    fEco.Free;
  finally
    inherited Destroy;
  end;
end;
end.

```