

# SKRIPSI

## OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE *REINFORCEMENT LEARNING (RL)* PADA PLN P3B REGION IV



*Disusun oleh;*

**TOMMY NUGROHO**  
**NIM 00.12.079**



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

**SEPTEMBER 2007**

THE UNIVERSITY OF

THE UNIVERSITY OF  
THE UNIVERSITY OF  
THE UNIVERSITY OF  
THE UNIVERSITY OF

THE UNIVERSITY OF  
THE UNIVERSITY OF  
THE UNIVERSITY OF

THE UNIVERSITY OF  
THE UNIVERSITY OF

THE UNIVERSITY OF

# LEMBAR PERSETUJUAN

OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE  
*REINFORCEMENT LEARNING (RL)* PADA PLN P3B REGION IV

## SKRIPSI

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

Disusun Oleh :

**TOMMY NUGROHO**  
00.12.079



Mengetahui,  
Ketua Jurusan Teknik Elektro

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

Menyetujui,  
Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT  
NIP.Y.101 8800 189

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



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

**Nama Mahasiswa** : TOMMY NUGROHO  
**N.I.M.** : 00.12.079  
**Jurusan** : Teknik Elektro S-1  
**Konsentrasi** : Teknik Energi Listrik  
**Judul Skripsi** : OPTIMASI DAYA REAKTIF MENGGUNAKAN  
METODE *REINFORCEMENT LEARNING (RL)*  
PADA PLN P3B REGION IV

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

**Hari** : Senin  
**Tanggal** : 3 September 2007  
**Dengan Nilai** : 80,35 (A) *Buy*



( Ir. Mochtar Asroni, MSME )  
Ketua

**Panitia Ujian**

( Ir. F. Yudi Limpraptono, MT )  
Sekretaris

**Anggota Penguji**

( Ir. Teguh Herbasuki, MT )  
Penguji Pertama

( Ir. Eko Nurcahyo )  
Penguji Kedua



# ABSTRAKSI

## ***OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE REINFORCEMENT LEARNING PADA PT. PLN P3B REGION IV***

(Tommy Nugroho, 00.12.079, Teknik Elektro S-1/T. Energi Listrik)  
(Dosen Pembimbing : Ir. Yusuf Ismail Nakhoda, MT)

**Kata Kunci :** Daya Reaktif, *Reinforcement Learning*, *Q-Learning*

Pada penelitian ini mempresentasikan model *Reinforcement Learning*. Dasar dari metode tersebut digambarkan dan diaplikasikan untuk permasalahan minimalisasi rugi-rugi dan kontrol tegangan dengan hasil yang sangat baik. Untuk uji validasi digunakan IEEE 14-bus sistem dan diaplikasikan pada sistem Region IV 150 Kv dengan 72 bus dan 86 saluran. Kontrol tegangan dan daya reaktif bertujuan untuk meningkatkan profil tegangan serta meminimalisasi rugi-rugi daya pada saluran transmisi 150 kV Region IV Jawa Timur

Untuk memenuhi pelaksanaan dari masalah *Reinforcement Learning*, masalah aliran beban terkendala dirumuskan sama seperti suatu ketetapan masalah banyaknya tingkatan. Seting kontrol optimal diketahui dengan pengembangan pengaturan suatu ketentuan *close-loop* kontrol, yang mana menggambarkan keadaan (solusi aliran beban) untuk tindakan pengontrolan (seting kontrol *offline*).

Dari hasil perhitungan *load flow* awal (sebelum optimasi) menggunakan metode Newton Raphson, maka total daya yang dibangkitkan sebesar 1485.119MW, total pembebanan 1417.500MW dan rugi-ruginya sebesar 67.619MW. Dan setelah dilakukan menggunakan metode RL maka *loadflow* akhirnya (setelah optimasi), untuk total daya pembangkitan sebesar 1483.335MW, total pembebanan sebesar 1417.500MW dan rugi-ruginya sebesar 65.835 MW

## KATA PENGANTAR

Dengan memanjatkan puji syukur kehadirat Allah SWT, berkat limpahan Rahmat dan Hidayah-Nya, sehingga penyusun dapat menyelesaikan skripsi ini.

Skripsi ini disusun sebagai salah satu persyaratan dalam menyelesaikan studi program strata satu (S-1) jurusan Teknik Elektro, Program Studi Teknik Energi Listrik, Fakultas Teknologi Industri, Institut Teknologi Nasional Malang

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

1. Bapak Prof. Dr. Ir. Abraham Lomi, MSEE, selaku Rektor Institut Teknologi Nasional Malang.
2. Bapak Ir. Mochtar Asroni, MSME, selaku Dekan Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
3. Bapak Ir. F. Yudi Limpraptono, MT, selaku Ketua Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Nasional Malang.
4. Bapak Ir. Yusuf Ismail Nakhoda, MT, selaku Dosen pembimbing.
5. Kedua orang tuaku, yang selalu memberikan do'a, kasih sayang dan dukungan hingga terselesaikannya skripsi ini.
6. Serta semua pihak yang tidak dapat penulis sebutkan satu – persatu, yang turut serta membantu menyelesaikan skripsi ini.

Penyusun menyadari akan segala kekurangan yang ada dalam skripsi ini, maka dengan kerendahan hati penyusun mengharapkan kritik dan saran demi penyempurnaan skripsi ini. Akhirnya, kepada semua pihak yang telah bekerja keras dan bersungguh-sungguh hingga terwujudnya skripsi ini , saya menyampaikan penghargaan dan terima kasih.

Malang,September 2007

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL</b> .....	i
<b>LEMBAR PERSETUJUAN</b> .....	ii
<b>ABSTRAKSI</b> .....	iii
<b>KATA PENGANTAR</b> .....	iv
<b>DAFTAR ISI</b> .....	vi
<b>DAFTAR GAMBAR</b> .....	ix
<b>DAFTAR TABEL</b> .....	x
<b>DAFTAR GRAFIK</b> .....	xii

### **BAB I PENDAHULUAN**

1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	3
1.3. Tujuan.....	4
1.4. Batasan Masalah.....	4
1.5. Metodologi Penelitian .....	4
1.6. Sistematika Penulisan.....	5
1.7. Kontribusi .....	6

### **BAB II TEORI DASAR**

2.1. Sistem Tenaga Listrik.....	7
2.2. Saluran Transmisi.....	7
2.3. Konstanta Pada Saluran Transmisi.....	8



2.3.1. Resistansi.....	8
2.3.2 Induktansi.....	9
2.3.3 Kapasitansi.....	9
2.4. Tegangan Transmisi .....	9
2.5. Daya Dalam Sistem Tenaga Listrik.....	10
2.6. Daya Reaktif.....	11
2.7. Pengaruh Daya Reaktif Terhadap Tegangan.....	12
2.8. Pengaturan Tegangan dan Daya Reaktif .....	14
2.8.1 Pengaturan Eksitasi Generator.....	15
2.8.2 Pengaturan Dengan Peralatan Kompensasi Daya Reaktif..	16
2.8.2.1 Kapasitor Shunt.....	16
2.8.2.2 Reaktor Shunt.....	18
2.9. Operasi Sistem 150 kV.....	20
2.10. Analisa aliran Daya.....	21
2.10.1 Persamaan Aliran daya Dengan Menggunakan Metode Newton – Rhapson.....	22
2.10.2 Klasifikasi Bus.....	26
2.11. Analisa Sensitivitas.....	28

**BAB III APLIKASI METODE REINFORCEMENT LEARNING DALAM  
PENGONTROLAN DAYA REAKTIF**

3.1. Saluran Transmisi.....	32
3.2. Studi Aliran Beban.....	32
3.3. Masalah <i>CLF</i> ( <i>Constrained Load flow</i> ).....	34

3.4. <i>Reinforcement Learning (RL)</i> .....	35
3.5. Pengaplikasian <i>Q-learning</i> Untuk Pengontrolan Daya Reaktif ..	37
3.6. Vektor Keadaan ( <i>State</i> ) .....	38
3.7. Vektor Tindakan ( <i>Action</i> ).....	38
3.8. Rewards.....	39
3.9. Parameter <i>Discount</i> ( $\gamma$ ) dan <i>Learning Step</i> ( $\alpha$ ).....	40

## **BAB IV HASIL DAN ANALISA HASIL**

4.1. Program Komputer Optimasi Daya Reaktif Menggunakan Metode <i>Reinforcement Learning</i> .....	42
4.1.1. Algoritma Program.....	42
4.1.2. Flowchart Program.....	45
4.2. Validasi Program Menggunakan Data Sistem IEEE 14-Bus.....	48
4.2.1. Data Sistem IEEE 14-Bus.....	48
4.2.2. Hasil PerhitunganValidasi IEEE 14 Dengan Menggunakan Metode <i>Reinforcement Learning</i> .....	49
4.3. Data Pembangkitan dan Pembebanan Pada PLN P3B Region IV.....	53
4.4. Data Saluran Transmisi Pada PLN P3B Region IV.....	58
4.5. Prosedur pelaksanaan program perhitungan.....	60
4.6. Hasil dan Analisis Hasil Perhitungan Optimasi Daya Reaktif Menggunakan Metode <i>Reinforcement Learning</i> Pada PLN P3B Region IV.....	67

4.6.1. Hasil Perhitungan Sebelum Optimasi.....	67
4.6.2. Hasil Perhitungan Setelah Optimasi.....	71
4.7. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Daya Reaktif Menggunakan Metode <i>Reinforcement Learning</i> .....	75
4.7.1. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus.....	75
4.7.2. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran....	77

## **BAB V KESIMPULAN DAN SARAN**

5.1. Kesimpulan.....	79
5.2. Saran.....	80

## **DAFTAR PUSTAKA**

## **LAMPIRAN**

## DAFTAR GAMBAR

Gambar 2-1	Rangkaian ekivalen Saluran transmisi.....	8
Gambar 2-2	Reprentasi segitiga daya.....	11
Gambar 2-3	Aliran daya Melalui saluran Pendek.....	13
Gambar 2-4	Perubahan Tegangan Sebagai Fungsi aliran Daya Reaktif.....	14
Gambar 2-5	Eksitasi Generator .....	16
Gambar 2-6	Kapasitor Pada Simpul Beban.....	17
Gambar 2-7	Rangkaian Ekivalen Pemasangan Kapasitor Dalam Bentuk Thevenin.....	17
Gambar 2-8	Diagram Kapasitor Sebelum dihubungkan Dengan saklar tertutup	18
Gambar 4-1	<i>Flowchart</i> Perhitungan Aliran Daya Metode Newton Rapshon.....	45
Gambar 4-2	Flowchart Program Reinforcement learning untuk Pengontrolan Daya Reaktif.....	47
Gambar 4-3	Tampilan program IEEE-14 Bus.....	49
Gambar 4-4	Tampilan input data IEEE-14 Bus.....	50
Gambar 4-5	Tampilan Parameter RL.....	50
Gambar 4-6	Hasil load flow.....	51
Gambar 4-7	Hasil perhitungan program.....	51
Gambar 4-8	Tampilan Utama Program P3B Region IV.....	61
Gambar 4-9	Tampilan Masukan Data.....	61
Gambar 4-10	Tampilan Data Bus.....	62



Gambar 4-11	Tampilan Data Saluran.....	62
Gambar 4-12	Tampilan Data Generator.....	63
Gambar 4-13	Tampilan hasil <i>Load Flow</i> pada Kondisi Awal (Sebelum Optimasi).....	63
Gambar 4-14	Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi).....	64
Gambar 4-15	Tampilan <i>Summary Load Flow</i> Sebelum Optimasi.....	64
Gambar 4-16	Tampilan Parameter RL.....	65
Gambar 4-17	Tampilan Hasil Program pada Kondisi Akhir (Setelah Optimasi).....	65
Gambar 4-18	Tampilan Hasil <i>Load Flow</i> Kondisi Akhir (Setelah Optimasi).....	66
Gambar 4-19	Tampilan Aliran Daya Pada Kondisi Akhir (Setelah Optimasi).....	66
Gambar 4-20	Tampilan <i>Summary Load Flow</i> Setelah Optimasi.....	67

## DAFTAR TABEL

Tabel 4-1	Data Generator IEEE 14-Bus.....	48
Tabel 4-2	Data Saluran IEEE 14-Bus.....	48
Tabel 4-3	Perbandingan Hasil Data Referensi Jurnal dengan Data Optimasi..	52
Tabel 4-4	Perbandingan Total Rugi-Rugi.....	53
Tabel 4-5	Penomoran Bus.....	53
Tabel 4-6	Data Pembangkitan dan Pembebanan Pada PLN P3B Region IV...	56
Tabel 4-7	Data saluran transmisi 150 kV pada PLN P3B Region IV.....	58
Tabel 4-8	Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sebelum Optimasi.....	67
Tabel 4-9	Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi..	69
Tabel 4-10	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran sebelum Optimasi.....	71
Tabel 4-11	Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sesudah Optimasi.....	71
Tabel 4-12	Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi....	73
Tabel 4-13	Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi.....	75
Tabel 4-14	Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus.....	75

## DAFTAR GRAFIK

Grafik 4-1	Perbandingan Tegangan Sebelum Optimasi dan Setelah Optimasi.....	78
Grafik 4-2	Perbandingan Total Pembangkitan, Pembebanan dan Rugi-Rugi Untuk Daya Aktif Sebelum dan Sesudah Optimasi.....	78

# BAB I

## PENDAHULUAN

### 1.1. Latar Belakang

Seirama dengan perkembangan pemakaian tenaga listrik oleh para pelanggan, berkembang pula perkembangan pemakaian tenaga listrik yang dilayaninya. Karena berbagai persoalan teknis, tenaga listrik hanya dibangkitkan pada tempat-tempat tertentu. Sedangkan pemakai tenaga listrik atau pelanggan tenaga listrik tersebar diperbagai tempat, maka penyampaian tenaga listrik dari tempat dibangkitkan sampai ke tempat pelanggan memerlukan berbagai penanganan teknis.

Banyak permasalahan optimisasi sistem daya dapat dinyatakan sebagai masalah optimisasi kombinasi, seperti masalah aliran beban yang terkendala (*Constrained Load Flow*). Masalah *CLF* mencakup penyesuaian *offline* dari variabel pengendali sistem daya untuk memenuhi kendala fisik dan pengoperasian.

Seperti telah kita sadari, bahwa didalam sistem tenaga listrik tidak mungkin menyalurkan energi listrik secara terus menerus ke konsumen secara mutlak tanpa gangguan. Peluang terjadinya gangguan pada saluran transmisi maupun distribusi; baik itu gangguan kawat putus, hubung singkat, tegangan lebih dan lain-lain, tetap tidak bisa dihilangkan. Bahkan dengan disain suatu peralatan penyaluran yang mahalpun, terjadinya gangguan tidak bisa dihindari dan tetap ada, walaupun kecil kemungkinan terjadinya gangguan tersebut. Yang bisa



dilakukan oleh kita, adalah bagaimana caranya agar gangguan yang terjadi berakibat sekecil mungkin, baik bagi peralatan maupun konsumen.

Dengan demikian studi aliran beban sangat penting dalam merencanakan perluasan sistem tenaga listrik dan dalam menentukan operasi terbaik untuk sistem yang telah ada. Mutu dan keandalan diukur dengan frekwensi, tegangan dan jumlah gangguan. Masalah mutu tenaga listrik tidak semata-mata merupakan masalah operasi Sistem Tenaga Listrik tetapi erat kaitannya dengan pemeliharaan instalasi tenaga listrik dan juga erat kaitannya dengan masalah pengembangan Sistem Tenaga Listrik, mengingat bahwa konsumsi tenaga listrik oleh para pelanggan selalu bertambah dari waktu ke waktu. Keterangan utama yang diperoleh dari suatu studi aliran beban adalah besar dan sudut fasa tegangan pada setiap rel dan daya nyata maupun daya reaktif yang mengalir pada setiap saluran. Dalam penyediaan tenaga listrik bagi para pelanggan, tegangan yang konstan seperti halnya frekwensi yang konstan, merupakan salah satu syarat utama yang harus dipenuhi. Oleh karenanya masalah pengaturan tegangan merupakan masalah operasi sistem tenaga listrik yang perlu mendapatkan penanganan tersendiri. Pengaturan tegangan erat kaitannya dengan pengaturan daya reaktif dalam sistem. Alokasi daya reaktif yang tepat dalam sistem bisa menghasilkan kondisi optimum, yaitu kondisi dengan rugi-rugi transmisi yang minimum.

Pada skripsi ini akan dibahas metode alternatif untuk masalah aliran beban terkendala dengan meminimalisasi rugi-rugi daya reaktif dan daya aktif melalui penerapan studi aliran beban menggunakan metode *Reinforcement Learning*. Dan yang paling utama metode *Reinforcement Learning* bila dibandingkan dengan

metode yang lain, misalnya seperti *Conventional Probabilistic method* yang dikenalkan oleh *Ernst et al.* menunjukkan hasil yang lebih baik dalam meminimalisasi rugi-rugi daya reaktif dan daya aktif<sup>[1]</sup>.

Untuk memenuhi pelaksanaan dari masalah *Reinforcement Learning*, masalah aliran beban terkendala dirumuskan sama seperti suatu ketetapan masalah banyaknya tingkatan. Seting kontrol optimal diketahui dengan pengembangan pengaturan suatu ketentuan *close-loop* kontrol, yang mana menggambarkan keadaan (solusi aliran beban) untuk tindakan pengontrolan (seting kontrol *offline*). Seting kontrol berdasarkan pada *reward*, menunjukkan bagaimana tindakan bekerja diatas seluruh periode perencanaan, yaitu bagaimana batasan pengoperasian dari variabel perencanaan terpenuhi. Di dalam makalah ini *model-free learning algorithm (Q-learning)* diaplikasikan untuk pengontrolan daya reaktif, bagaimanapun, merupakan algoritma umum dan dapat diaplikasikan untuk suatu jenis lebar dari masalah optimasi terkendala di dalam perencanaan atau pengoperasi perencanaan.

## **1.2. Permasalahan**

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

Bagaimana menentukan dan menghitung tegangan, arus, faktor daya atau daya reaktif yang terdapat pada berbagai titik dalam suatu jaringan listrik pada keadaan

pengoperasian normal, baik yang sedang berjalan maupun yang diharapkan akan terjadi dimasa yang akan datang.

Berdasarkan deskripsi permasalahan tersebut. maka proposal ini diberi judul :

**“ OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE  
REINFORCEMENT LEARNING PADA PT. PLN PJB REGION IV ”**

### **1.3. Tujuan**

Tujuan dari penelitian skripsi dengan menggunakan metode *reinforcement learning (RL)* ini adalah:

Pengoptimasian pengontrolan daya reaktif untuk memperkecil rugi-rugi daya nyata yang sesuai dengan kendala operasi selama seluruh periode perencanaan.

### **1.4. Batasan Masalah**

Agar permasalahan mengarah sesuai dengan tujuan, maka pembahasan skripsi ini dibatasi sebagai berikut :

1. Analisis optimasi hanya menyangkut masalah aliran beban yang terkendala (*Constrained Load Flow*).
2. Analisa perhitungan aliran beban pada PT. PLN PJB REGION IV menggunakan metode *Reinforcement Learning*.

### **1.5. Metode Penelitian**

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

## 1. Studi Literatur

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

## 2. Pengumpulan Data

Bentuk data yang digunakan :

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

## 3. Simulasi dan pembahasan masalah

Menggunakan analisa perhitungan yang disimulasi dengan menggunakan pendekatan *Reinforcement Learning* pada komputer dengan bahasa pemrograman *Borland Delphi* versi 07.

### 1.6. Sistematika Penulisan

Adapun sistematika pembahasan pada skripsi ini adalah sebagai berikut :

#### BAB I : PENDAHULUAN

Menguraikan latar belakang, rumusan masalah, tujuan, batasan masalah, metode penelitian, sistematika penulisan dan kontribusi penelitian.

#### BAB II : TEORI DASAR

Meliputi teori-teori dasar yang mendukung studi aliran beban dan metode *Reinforcement Learning*.



**BAB III : ANALISA OPTIMASI DAYA REAKTIF  
MENGUNAKAN METODE REINFORCEMENT  
LEARNING**

Menguraikan teori dasar dari *reinforcement learning*, ke permasalahan optimasi daya reaktif.

**BAB IV : HASIL DAN ANALISIS HASIL**

Menguraikan alur program, hasil validasi, serta hasil perhitungan *reinforcement learning*

**BAB V : KESIMPULAN DAN SARAN**

Memuat intisari dan hasil pembahasan, yang berisikan kesimpulan dan saran yang dapat digunakan sebagai pertimbangan untuk pengembangan selanjutnya.

**1.6. Kontribusi**

Kontribusi dari skripsi ini adalah menggunakan *Reinforcement Learning (RL)* untuk dapat diterapkan oleh pemerintah khususnya PT. PLN PJB REGION IV, sehingga bisa menambah keuntungan bagi PLN sebagai perusahaan penyedia energi listrik di Indonesia.

## **BAB II**

### **TEORI DASAR**

Sistem tenaga listrik terdiri dari tiga bagian utama yaitu sistem pembangkit, sistem transmisi dan sistem beban. Setiap bagian dari sistem tersebut dibagi menjadi beberapa komponen yang saling berhubungan. Sistem pembangkit umumnya terletak jauh dari tempat dimana energi listrik itu digunakan atau pusat – pusat beban (*load centers*). Karena itu, tenaga listrik yang dibangkitkan harus disalurkan melalui kawat – kawat transmisi.

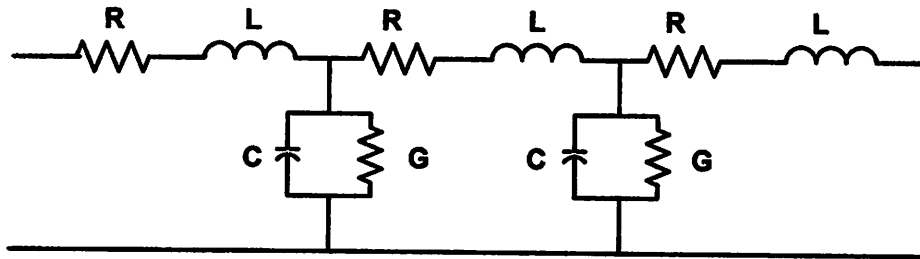
#### **2.1. Sistem Tenaga Listrik<sup>[4]</sup>**

Menurut jenis arusnya dikenal sistem arus bolak – balik (AC) dan sistem arus searah (DC) . Di dalam sistem AC kenaikan dan penurunan tegangan mudah dilakukan yaitu dengan menggunakan transformator. Itulah sebabnya maka sampai sekarang ini saluran transmisi di dunia sebagian besar adalah saluran AC. Di dalam sistem AC ada sistem satu fasa dan tiga fasa. Di Indonesia sistem yang digunakan adalah sistem tiga fasa karena memiliki banyak kelebihan dibandingkan dengan sistem satu fasa. Kelebihannya yaitu daya yang dibangkitkan lebih besar nilai sesaatnya konstan dan medan putarnya mudah diadakan.

#### **2.2. Saluran Transmisi<sup>[4]</sup>**

Saluran transmisi tenaga listrik mempunyai empat parameter yang mempengaruhi kemampuannya untuk berfungsi sebagai bagian dari suatu sistem tenaga listrik yaitu resistansi, induktansi, kapasitansi dan konduktansi yang

dinamakan konstanta saluran (*line konstan*). Keempat parameter saluran tersebut tersebar merata disepanjang saluran transmisi. Nilai – nilai dari parameter tersebut sangat tergantung pada panjang saluran.



Gambar 2.1

Rangkaian ekivalen Saluran transmisi

Sumber : Ir. I Made Wartana, "Transmisi Daya Arus Bolak-balk," diktat Kuliah ITN Malang.

Untuk mempresentasikan suatu saluran transmisi secara umum kedalam bentuk rangkaian setaranya, resistansi (R) dan induktansi (L) dinyatakan sebagai impedansi seri, sedangkan kapasitansi (C) dan konduktansi (G) dinyatakan sebagai admitansi *shunt* (simpang). Rangkaian ekivalen saluran transmisi secara umum dapat dilihat pada gambar diatas.

### 2.3. Konstanta Pada Saluran Transmisi<sup>[4]</sup>

#### 2.3.1. Resistansi

Resistansi pada sebuah penghantar sebanding dengan panjang kawat dan terbalik dengan luas penampangnya, hal ini dapat dilihat dari persamaan dibawah ini :

$$R = \rho \frac{l}{A} \Omega \dots\dots\dots (2.1)$$

Dimana : R = Resistansi saluran (ohm)

$\rho$  = Resistivitas saluran (ohm-m)

$l$  = Panjang saluran (m)

$A$  = Luas penampang ( $m^2$ )

### 2.3.2. Induktansi

Induktansi rata – rata per fasa adalah :

$$L = 2.10^{-7} \ln \frac{D_{eq}}{D_s} \text{ H/m} \dots\dots\dots (2.2)$$

Dimana :  $D_{eq}$  = Jarak rata – rata geometris penghantar

$D_s$  = Radius rata – rata geometris penghantar

### 2.3.3. Kapasitansi

Kapasitansi antara dua penghantar pada saluran dua kawat didefinisikan sebagai muatan pada penghantar itu per unit beda potensial diantara keduanya. Akibat beda potensial antara penghantar (konduktor) dapat diambil persamaan sebagai berikut :

$$C = \frac{q}{v} \text{ F/m} \dots\dots\dots (2.3)$$

Dimana :  $q$  = Muatan pada saluran (C/m)

$V$  = Beda potensial antara kedua penghantar (Volt)

### 2.4. Tegangan Transmisi<sup>[4]</sup>

Untuk daya yang sama, daya guna penyaluran akan naik apabila tegangan transmisi dinaikan. Oleh karena itu, pemilihan tegangan transmisi dilakukan dengan memperhitungkan daya yang disalurkan, jumlah rangkaian, jarak penyaluran, keandalan, biaya peralatan untuk tegangan tertentu, serta tegangan yang sekarang ada dan yang akan direncanakan. Penentuan tegangan juga harus

dilihat dari segi standarisasi peralatan yang ada. Penentuan tegangan merupakan bagian dari perencanaan sistem secara keseluruhan.

Di Indonesia, telah menyeragamkan tegangan pada saluran transmisi sebagai berikut :

Tegangan nominal (kv) : 30 – 66 – 150 – 220 – 380 – 500

Tegangan maksimum perlengkapan (kv) : 36 – 72.5 – 170 – 245 – 420 – 525

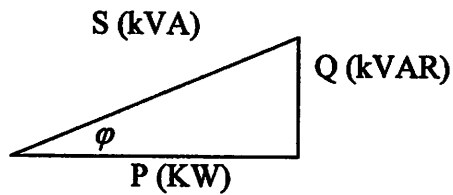
Tegangan nominal 30 kv hanya diperkenankan untuk daerah asuhan (tertentu) dimana tegangan distribusi primer 20 kv tidak diperkenankan.

## **2.5. Daya Dalam Sistem Tenaga Listrik<sup>[2]</sup>**

Dalam sistem tenaga listrik, pembangkit – pembangkit tenaga listrik harus mampu menyediakan tenaga listrik kepada pelanggan sesuai dengan permintaan beban listrik yang ada. Dan hal yang harus diperhatikan adalah sistem yang tetap konstan, dalam hal ini tegangan dan frekuensi harus tetap konstan karena berhubungan dengan daya. Daya listrik yang di bangkitkan dikenal dengan istilah:

- Daya aktif adalah daya terpakai, yaitu daya yang dapat dikonversikan ke bentuk lain (KW).
- Daya reaktif adalah daya yang timbul karena adanya pembentukan medan magnet pada beban – beban induktif (KVAR).
- Daya semu merupakan penjumlahan secara vektoris antara daya aktif dan daya reaktif (KVA).

Selain diatas dikenal juga istilah faktor daya (*power faktor*) atau *cosphi* faktor daya adalah merupakan perbandingan antara daya aktif dan daya semu yang dalam bentuk vektor dapat digambarkan sebagai berikut :



Gambar 2.2  
Representasi segitiga daya

Sumber : William D. Stevenson 1996, "Anlisis sistem Tenaga Listrik," Edisi IV Erlangga

Hubungan antara ketiganya dapat ditunjukkan dengan persamaan matematika sebagai berikut :

$$P = V \times I \times \cos \varphi$$

$$Q = V \times I \times \sin \varphi$$

$$S = V \times I$$

$$\begin{aligned} \cos \varphi &= \frac{P}{S} \\ &= \frac{P}{\sqrt{P^2 + Q^2}} \end{aligned}$$

## 2.6. Daya Reaktif

Kita mengenal dua macam daya reaktif yaitu daya reaktif induktif dan daya reaktif kapasitif, dimana keduanya memiliki tanda yang berlawanan. Daya reaktif kapasitif adalah daya listrik yang dibutuhkan kapasitor, kapasitansi kawat transmisi tegangan tinggi dan lainnya. Dari uraian diatas daya reaktif ini tidak menghasilkan kerja, tetapi tersimpan dalam bentuk energi magnetis atau energi kapasitif.

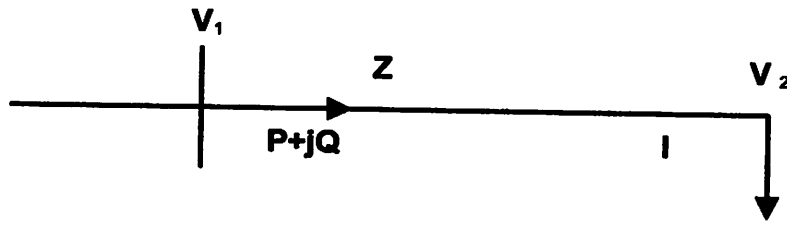
Daya reaktif induktif adalah daya listrik yang dibutuhkan untuk menghasilkan medan magnet yang dibutuhkan oleh alat – alat induksi seperti motor induksi, transformator dan sebagainya. Tanpa daya reaktif induktif ini daya tidak dapat ditransfer kesisi sekunder dalam suatu trafo atau melalui celah udara pada motor listrik.

Keseimbangan daya reaktif dalam pembangkitan harus dijaga karena hal ini akan mempengaruhi harga tegangan yang harus dijaga pada batas – batas tertentu dari harga nominalnya. Misalkan pada peralihan dari keadaan beban tinggi ke beban rendah ada suatu kecenderungan tegangan sistem akan naik. Hal ini harus diperbaiki dengan mengatur tap trafo, menurunkan penguatan generator atau menghubungkan dengan reaktor atau kapasitor.

## **2.7. Pengaruh Daya Reaktif Terhadap Tegangan**

Keseimbangan daya reaktif yang dihasilkan dan digunakan harus dapat dipertahankan agar tidak terjadi perubahan tegangan. Bila besaran tegangan ini mengalami perubahan, maka daya reaktif pada simpul tersebut tidak dapat dipertahankan lagi. Ini perlu aliran daya reaktif pada sistem.

Dari gambar 2-3 dibawah ini terlihat bahwa pembangkit memberikan daya sebesar  $P + jQ$  ke beban. Misalkan  $R$  dianggap jauh lebih kecil dari  $X$ , maka resistansi saluran dapat diabaikan. Sehingga impedansi saluran dituliskan  $Z = jX$ .



Gambar 2.3

Aliran daya Melalui saluran Pendek

Sumber : William D. Stevenson 1996, "Analisis sistem Tenaga Listrik," Edisi IV Erlangga

Jatuh tegangan yang terjadi sepanjang saluran disebabkan oleh adanya impedansi sehingga hubungan yang terjadi :

$$V_2 = V_1 - I.Z \dots\dots\dots(2.4)$$

Karena daya yang mengalir pada simpul I adalah  $P + jQ$ , maka :

$$I = \frac{P + jQ}{V_1^*}$$

Jika  $V_1$  dipilih sebagai referensi dan dianggap mempunyai sudut nol maka persamaan menjadi :

$$I = \frac{P - jQ}{V_1} \dots\dots\dots(2.5)$$

Substitusi persamaan (2-4) ke persamaan (2-5) diperoleh :

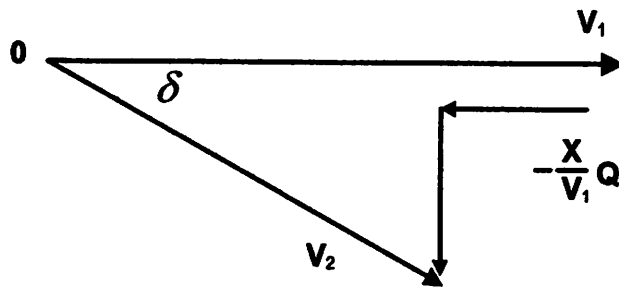
$$V_2 = V_1 - \left( \frac{P - jQ}{V_1} \cdot Z \right)$$

$$V_2 = V_1 - \left( \frac{jXP}{V_1} + \frac{XQ}{V_1} \right)$$

$$V_2 = V_1 - \frac{XQ}{V_1} - j \frac{X}{V_1} P \dots\dots\dots(2.6)$$

Persamaan diatas dapat dilihat dalam bentuk gambar sebagai berikut :





Gambar 2 .3

Perubahan Tegangan Sebagai Fungsi aliran Daya Reaktif

Sumber : William D. Stevenson 1996, "Anlisis sistem Tenaga Listrik," Edisi IV Erlangga

Dari persamaan diatas dapat dijelaskan :

1. Perubahan daya aktif  $\Delta P$  pada dasarnya hanya mempengaruhi sudut fasa tegangan sedangkan magnitudo tegangan hampir tidak berubah.
2. Perubahan daya reaktif  $\Delta Q$  pada dasarnya hanya mempengaruhi magnitudo tegangan [V] dan hampir tidak berpengaruh pada sudut fasa tegangan

## 2.8. Pengaturan Tegangan dan Daya Reaktif <sup>[3]</sup>

Pengaturan tegangan pada dasarnya adalah pengaturan daya reaktif pada saluran dan simpul. Sehingga untuk mempertahankan tegangan didalam batas operasi yang diinginkan perlu keseimbangan daya reaktif pada simpul tersebut.

Sebagaimana diketahui masalah tegangan dapat diakibatkan oleh pelanggaran batas atas maupun batas bawah. Jika terlalu banyak beban daya reaktif induktif (positif), misalnya pada daerah industri, maka tegangan simpul yang terlalu rendah akan jadi masalah. Ini dapat diatasi dengan memasang kapasitor shunt, *Static Var Compensation* dan sebagainya. Sebaliknya pada saat beban ringan karena pengaruh kapasitansi shunt dari saluran transmisi, maka akan

ada kelebihan daya reaktif pada saluran sehingga tegangan simpul terlalu tinggi. Hal ini dapat diatasi dengan memasang kapasitor shunt, *Static Var Compensation* dan sebagainya.

Dengan menginjeksikan daya reaktif kedalam sistem, pengatur – pengatur baik untuk mengatur pembangkitan maupun aliran daya reaktif. Dengan cara ini profil tegangan simpul yang diinginkan dapat diperoleh.

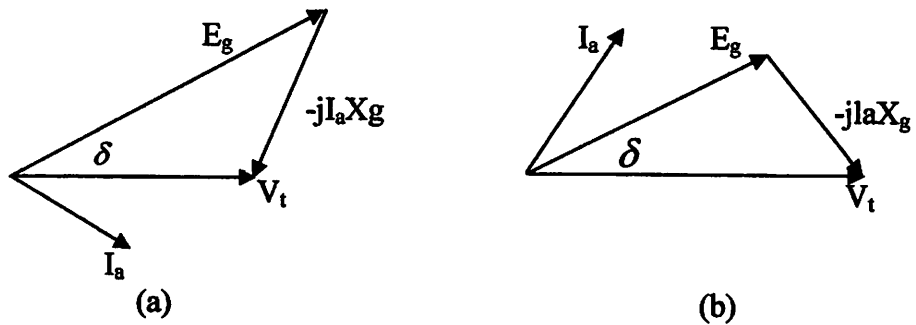
### 2.8.1. Pengaturan Eksitasi Generator<sup>[3]</sup>

Pegaturan eksitasi generator mempunyai fungsi utama mengatur tegangan pada sisi terima didalam suatu harga yang telah ditetapkan dengan mengirimkan sejumlah daya reaktif. Tegangan terminal generator dapat berubah sesuai dengan perubahan eksitasi generator.

Jika suatu generator dihubungkan kesuatu sistem tenaga yang sangat besar, maka tegangan terminal generator  $V_t$  relatif konstan meskipun ada perubahan eksitasi generator. Simpul tempat generator itu dihubungkan disebut simpul tak terhingga (*infinite bus*). Jika masukan daya aktif ke sistem  $P = |V_t| |I_a| \cos \theta$  konstan dan dengan mengubah – ubah eksitasi untuk mengubah  $|E_g|$  yang tinggi dan yang rendah, diagram pashornya dapat dilihat pada gambar dibawah ini.

Sudut  $\delta$  dinamakan sudut daya dari generator, eksitasi yang normal didefinisikan sebagai berikut :

$$|E_g| \cos \delta = V_t \dots\dots\dots(2.7)$$



Gambar 2.5

Eksitasi Generator

Sumber : William D. Stevenson 1996, "Anlisis sistem Tenaga Listrik," Edisi IV Erlangga

Untuk keadaan pada gambar 2-5 (a) generator terlalu diperkuat (*over ekcited*) dan mencatu arus tertinggal *lagging* ke sistem. Gambar 2-5 (b) adalah generator yang kurang diperkuat (*under ekcited*) yang mencatu arus mendahului (*leading*) ke sistem. Generator ini menyerap daya reaktif dari sistem.

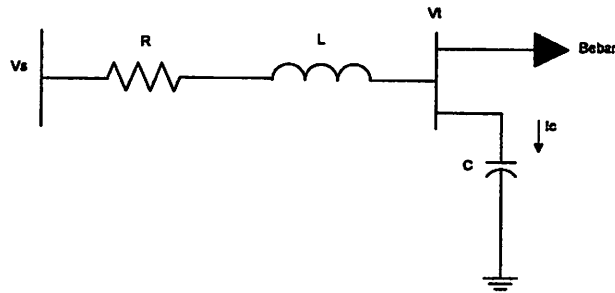
2.8.2. Pengaturan Dengan Peralatan Kompensasi Daya Reaktif<sup>[3]</sup>

Rendahnya faktor daya beban mengakibatkan naiknya kebutuhan daya reaktif dan arus saluran. Karena itu diperlukan peralatan yang dapat mengkompensasi kebutuhan daya reaktif ini, baik *lagging* maupun *leading*. Kapasitor shunt digunakan untuk sirkit faktor daya *lagging* sedang reaktor shunt untuk sirkit *leading*.

2.8.2.1. Kapasitor Shunt<sup>[3]</sup>

Pada dasarnya kapasitor suatu alat untuk mencatu daya reaktif pada titik pemasanganya. Dengan pemasangan kapasitor secara pararel pada sisi terima seperti pada gambar 2-6, rugi – rugi tegangan dan rugi – rugi lain pada saluran

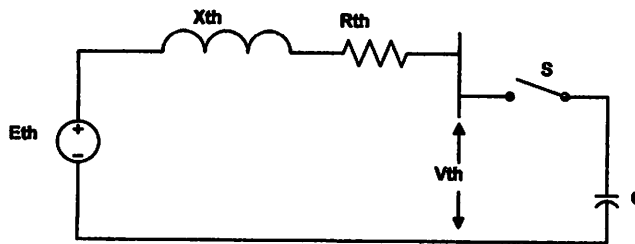
berkurang dan tegangan pada titik tersebut akan naik. Kapasitor – kapasitor shunt dapat dihubungkan secara tetap, tetapi sebagai pengatur tegangan dapat juga dihubungkan dan diputuskan dari sistem melalui saklar sesuai dengan perubahan – perubahan permintaan tegangan dan daya reaktif.



Gambar 2.6  
Kapasitor Pada Simpul Beban

Sumber : William D. Stevenson 1996, "Anlisis sistem Tenaga Listrik," Edisi IV Erlangga

Jika pada suatu simpul tertentu dipasang kapasitor, kenaikan tegangan pada simpul dapat ditentukan dengan dalil thevenin.



Gambar 2.7

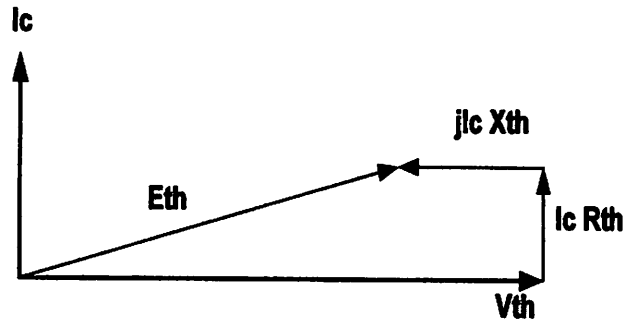
Rangkaian Ekuivalen Pemasangan Kapasitor Dalam Bentuk Thevenin

Sumber : William D. Stevenson 1996, "Anlisis sistem Tenaga Listrik,"

maka akan mengalir arus kapasitif sebesar :

$$I_c = \frac{E_{th}}{Z_{th} - jX_c} \dots\dots\dots (2.8)$$

Dagram fasornya diperlihatkan pada gambar 2 – 8, dimana terlihat bahwa kenaikan  $V_t$  yang disebabkan oleh penambahan kapasitor hampir sama dengan  $j|I_c|X_{th}$ .



**Gambar 2.8**  
**Diagram Kapasitor Sebelum dihubungkan Dengan saklar tertutup**  
 Sumber : William D. Stevenson 1996, "Analisis Sistem Tenaga Listrik," Edisi IV Erlangga

Untuk menentukan rating VAR dari kapasitor yang dibutuhkan dapat diperoleh dengan memperhitungkan daya reaktif beban.

### 2.8.2.2. Reaktor Shunt<sup>[3]</sup>

Pada saluran transmisi panjang pengaruh kapasitansi pada saluran dapat menimbulkan arus pengisian (*line charging*) yang cukup besar. Bila ada pengaturan gejala ini sangat menguntungkan karena saat beban puncak kebutuhan daya reaktif cukup besar, sehingga kebutuhan daya reaktif ini dapat dipenuhi oleh saluran transmisi itu sendiri.

Sebaliknya pada saat beban ringan arus pengisian ini dapat menyebabkan kelebihan daya reaktif pada sistem yang menyebabkan kenaikan tegangan pada saluran transmisi. Untuk menyerap kelebihan daya reaktif kapasitif pada sistem ini dapat digunakan reaktor shunt yang berupa suatu kumparan dan dipasang paralel dengan sistem.

Untuk tujuan mengatur aliran daya reaktif, pada umumnya reaktor ini dikombinasikan dengan kapasitor sehingga kombinasi ini mempunyai

kemampuan untuk membangkitkan atau menyerap daya reaktif dari sistem. Dari persamaan saluran transmisi panjang diperoleh :

$$V_s = V_r \cosh \gamma l + I_r Z_c \sinh \gamma l \dots\dots\dots(2.9)$$

Pada beban nol, harga  $I_r$  sangat kecil sehingga dapat diabaikan. Dengan demikian persamaan diatas menjadi :

$$V_s = V_{r0} \cosh \gamma l \dots\dots\dots (2.10)$$

Dimana :

$V_{r0}$  = tegangan sisi terima pada sisi nol

$$\gamma l = (\alpha + j\beta)l$$

Karena arus mengalir pada saluran sangat kecil, maka rugi – rugi daya  $I^2R$  dapat diabaikan ( $\alpha = 0$ ), sehingga :

$$V_s = V_{r0} \cosh \beta l \dots\dots\dots(2.11)$$

Dimana :

$$\beta = \frac{2\Delta}{\lambda} \qquad \lambda = \frac{v}{f} = \frac{1}{f\sqrt{LC}}$$

$\lambda$  = Panjang gelombang

$v$  = Cepat rambat gelombang

$f$  = Frekuensi gelombang tegangan

Jadi :

$$V_s = V_m \cosh (2\Delta f \sqrt{LC} l) \dots\dots\dots (2.12)$$

Dari persamaan diatas dapat dijelaskan bahwa jika susceptansi kapasitif saluran dikompensasi dengan reaktor shunt, maka harga  $\text{Cosh}(2\Delta f \sqrt{LC})l$  akan naik. Dengan demikian kenaikan tegangan disisi terima dapat dikurangi.

## 2.9. Operasi Sistem 500 kV<sup>[2]</sup>

Sistem 500 kV yang telah beroperasi di Indonesia baru ada dipulau Jawa. Masalah isolasi peralatan dan kapasitansi saluran merupakan hal – hal yang perlu mendapat perhatian lebih besar pada operasi sistem 500 kV dibandingkan dengan operasi sistem yang mempunyai tegangan lebih rendah.

Untuk pertama kalinya sistem 500 kV dioperasikan di pulau Jawa dalam tahun 1984. Tegangan 500 kV sudah tergolong tegangan ekstra tinggi, dalam arti bahwa surja karena *switching* (*switching surge*) menjadi lebih menonjol daripada surja karena petir (*lightning surge*).

Selama sistem 500 kV beroperasi belum pernah terjadi karena petir. Sistem 500 kV di pulau Jawa menggunakan SUTT dengan konduktor rangkap 4 untuk setiap fasa. Masalah yang menonjol secara operasional pada SUTT 500 kV adalah besarnya arus kapasitif, sehingga diperlukan reaktor – reaktor untuk mengkompensir arus kapasitif ini disamping kemampuan generator – generator ( yang langsung berhubungan dengan SUTT 500 kV ) untuk menyerap arus kapasitif ini. Untuk ini generator didesain secara khusus.

## 2.10. Analisa aliran Daya<sup>[2]</sup>

Analisa aliran daya perlu dilakukan dalam melayani kebutuhan beban dan pengoperasian sistem tenaga listrik, agar sistem tenaga listrik tersebut dapat dioperasikan baik memenuhi persyaratan teknis maupun ekonomis. Didalam analisa daya dilakukan perhitungan tegangan, arus, daya dan daya reaktif yang terdapat dalam berbagai titik pada suatu jaringan tenaga listrik pada keadaan pengoperasian normal, baik yang sedang berjalan maupun yang diharapkan terjadi dimasa yang akan datang. Analisa aliran daya sangat penting dalam perencanaan perkembangan suatu sistem untuk masa yang akan datang.

Di dalam analisa aliran daya terdapat tiga jenis variabel yaitu :

1. Variabel bebas yang diatur, misalnya  $|V|$  dan  $\delta$  pada bus beban atau  $\delta$  dan  $Q$  pada bus generator.
2. Variabel tidak bebas (yang mengatur) misalnya  $P$  dan  $|V|$  pada bus generator.
3. Variabel yang tidak dapat diatur (parameter tetap), misalnya kebutuhan konsumen.



### 2.10.1. Persamaan Aliran daya Dengan Metode Newton – Rhapson<sup>[2]</sup>

Dalam menyelesaikan persamaan aliran daya metode yang umum digunakan adalah Newton – Rhapson karena metode ini mempunyai waktu hitung yang lebih cepat dalam mencapai konvergensi atau membutuhkan jumlah iterasi yang lebih sedikit bila dibandingkan dengan metode lain. Penyelesaian aliran daya dengan menggunakan metode *Newton – Rhapson* dibentuk berdasarkan matriks admitsnsi simpul ( $Y_{bus}$ ) yang mana admitansi simpul dibuat dengan prosedur yang langsung dan sederhana.

Dengan membalikkan matrik impedansi rel  $Y_{bus}$  dan menamakan matriks yang dihasilkan matrik impedansi rel  $Z_{bus}$ . Menurut definisi :

$$Z_{bus} = Y_{bus}^{-1} \dots\dots\dots(2.13)$$

Untuk dapat memahami pentingnya bermacam – macam impedansi pada matriks tersebut secara fisik, kita akan membandingkan dengan admitansi simpul. Hal ini dapat kita lakukan dengan melihat persamaan pada suatu simpul tertentu.

$$I = Y_{bus} V \dots\dots\dots(2.14)$$

Dari simpul 2 dari tiga simpul yang berdiri bebas, kita mempunyai

$$I_2 = Y_{21}V_1 + Y_{22} V_2 + Y_{23} V_3 \dots\dots\dots (2.15)$$

Admitansi sendiri pada simpul 2 adalah

$$Y_{22} = \frac{I_2}{V_2} \Big|_{V_1=V_3=0} \dots\dots\dots(2.16)$$

Secara sistematis persamaan aliran daya metode Newton – Rhapson dapat menggunakan koordinat rectanguler, koordinat polar atau bentuk hibrid (gabungan

antara bentuk kompleks dengan bentuk polar). Dalam pembahasan ini digunakan bentuk polar.

Hubungan antara simpul  $I_p$  dan tegangan simpul  $V_q$  pada suatu jaringan dengan  $n$  simpul dapat ditulis :

$$I_p = \sum_{q=1}^n Y_{pq} V_q \dots\dots\dots (2.17)$$

Injeksi daya pada simpul  $p$  adalah :

$$S_p = P_p + jQ_p = V_p I_p^* \dots\dots\dots (2.18)$$

$$= V_p \sum_{q=1}^n Y_{pq}^* V_q^* \dots\dots\dots (2.19)$$

Penyelesaian aliran daya Newton – Rhapson menggunakan persamaan bentuk polar adalah :

$$V_p = |V_p| e^{j\delta_p}$$

$$V_p^* = |V_p| e^{-j\delta_p} \quad Q_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \text{Sin}(\delta_p - \delta_q - \theta_{pq})$$

$$Y_{pq}^* = |Y_{pq}| e^{-j\theta_{pq}}$$

Maka persamaan injeksi daya pada simpul  $P$  dapat ditulis :

$$P_p + jQ_p = \sum_{q=1}^n |V_p V_q Y_{pq}| e^{j(\delta_p - \delta_q - \theta_{pq})} \dots\dots\dots (2.20)$$

Selanjutnya daya nyata dan daya reaktif dapat dinyatakan sebagai berikut :

$$P_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \text{Cos}(\delta_p - \delta_q - \theta_{pq}) \dots\dots\dots (2.21)$$

$$Q_p = \sum_{q=1}^n |V_p V_q Y_{pq}| \text{Sin}(\delta_p - \delta_q - \theta_{pq}) \dots\dots\dots (2.22)$$

Persamaan diatas akan menghasilkan suatu kumpulan persamaan serempak (simultan) yang tidak linear untuk setiap sistem tenaga listrik. Untuk magnitudo tegangan  $|V|$  dan sudut fasa ( $\delta$ ) disetiap simpul dapat diselesaikan dengan persamaan aliran daya (2.21) dan (2.22) yang dilinearakan dengan metode Newton – Rhapson seperti persamaan dibawah ini :

$$\begin{bmatrix} \Delta P \\ \Delta Q \end{bmatrix} = \begin{bmatrix} H & N \\ M & L \end{bmatrix} \begin{bmatrix} \Delta \delta \\ \Delta |V| \end{bmatrix} \dots\dots\dots (2.23)$$

Dimana:

$\Delta P$  = Selisih injeksi daya aktif dengan penjumlahan aliran daya aktif tiap transmisi yang menghubungkan simpul dengan  $|V|$  yang didapat dari perhitungan iterasi ke-k.

$\Delta Q$  = Selisih injeksi daya reaktif dengan penjumlahan aliran daya reaktif tiap transmisi yang menghubungkan simpul dengan  $|V|$  yang didapat dari perhitungan iterasi ke-k.

$\Delta \delta$  = Vektor koreksi sudut fasa tegangan.

$\Delta |V|$  = Vektor koreksi magnitudo tegangan.

H, L, M, N adalah elemen – elemen off-diagonal dan dari sub-matriks Jacobian yang dibentuk dengan mendiferensialkan persamaan – persamaan (2.21) dan (2.22), dimana :

$$H_{pq} = \frac{\partial P_p}{\partial \delta_q} ; \quad N_{pq} = \frac{\partial P_p}{\partial |V_q|} ; \quad M_{pq} = \frac{\partial Q_p}{\partial \delta_q} ; \quad L_{pq} = \frac{\partial Q_p}{\partial |V_q|}$$

Rumus dari elemen matriks Jacobian adalah :

Untuk H adalah :

$$\frac{\partial P_p}{\partial \delta_q} = |V_p V_q V_{pq}| \sin(\delta_p - \delta_q - \theta_{pq}) \quad q \neq p$$

$$\frac{\partial P_p}{\partial \delta_p} = - \sum_{q=1}^n |V_p V_q Y_{pq}| \sin(\delta_p - \delta_q - \theta_{pq})$$

Untuk N adalah

$$\frac{\partial P_p}{\partial |V_q|} = |V_p V_{pq}| \cos(\delta_p - \delta_q - \theta_{pq}) \quad q \neq p$$

$$\frac{\partial P_p}{\partial |V_p|} = 2|V_p V_{pq}| \cos \theta_{pp} + \sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q - \theta_{pq})$$

Untuk M adalah :

$$\frac{\partial Q_p}{\partial \delta_q} = |V_p V_q V_{pq}| \cos(\delta_p - \delta_q - \theta_{pq}) \quad q \neq p$$

$$\frac{\partial Q_p}{\partial \delta_p} = - \sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q - \theta_{pq})$$

Untuk L adalah

$$\frac{\partial Q_p}{\partial |V_q|} = |V_p Y_{pq}| \sin(\delta_p - \delta_q - \theta_{pq}) \quad q \neq p$$

$$\frac{\partial P_p}{\partial |V_p|} = 2|V_p V_{pq}| \sin \theta_{pp} + \sum_{q=1}^n |V_p V_q Y_{pq}| \cos(\delta_p - \delta_q - \theta_{pq})$$

Untuk selisih daya, maka pertama ditentukan harga awal dari tegangan simpul dan sudut fasanya. Kemudian daya aktif dan daya reaktif dihitung dengan menggunakan persamaan (2.21) dan (2.22). Selisih daya antara daya yang telah

ditentukan dengan daya hasil perhitungan ini merupakan perubahan daya yang terjadi pada simpul.

$$\Delta p = -P_G + P_L - P_p \dots\dots\dots (2.24)$$

$$\Delta Q = -Q_G + Q_L - Q_p \dots\dots\dots (2.25)$$

Magnitude tegangan dan sudut fasa yang diasumsikan ( $|V_p|$  dan  $\delta_p$ ) serta selisih daya yang dihitung ( $\Delta P_p$  dan  $\Delta Q_p$ ) digunakan untuk memperoleh elemen elemen matriks Jacobian. Persamaan (2.23) digunakan untuk menghitung vektor koreksi magnitude tegangan ( $\Delta|V|$ ) dan sudut fasa tegangan ( $\Delta\delta$ ) yang baru.

$$|V|^{k+1} = |V|^k + \Delta|V|^k$$

$$\delta^{k+1} = \delta^k + \Delta\delta^k$$

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

### 2.10.2. Klasifikasi Bus

Tujuan analisa daya pada sistem tenaga listrik adalah untuk menghitung besar magnitude tegangan  $|V|$  dan sudut fasa tegangan  $\delta$  pada semua bus, sehingga dengan diketahuinya parameter – parameter tersebut akan dapat dihitung besar daya yang mengalir pada saluran transmisi beserta rugi – ruginya.

Pada setiap bus pada jaringan terdapat parameter – parameter sebagai berikut :

- Injeksi netto daya nyata, dinyatakan dengan P satuannya Megawatt (MW).
- Injeksi netto daya reaktif, dinyatakan dengan Q satuannya Mega Volt Ampere Reaktif (MVAR).
- Besar magnitude tegangan mempunyai simbol  $|V|$  dengan satuan kiloVolt (kV).
- Sudut fasa tegangan mempunyai simbol  $\delta$  dengan satuan radian.

Dengan analisa aliran daya, pada setiap bus perlu diketahui dua parameter dari keseluruhan empat parameter yang diperhitungkan. Dengan melihat kedua parameter yang diketahui, setiap bus diklasifikasikan menjadi 3 yaitu :

#### 1. Bus Beban (*load bus*)

Daya aktif (P) dan daya reaktif (Q) diketahui dimana pada bus ini hanya terdapat kebutuhan daya untuk beban, sementara  $|V|$  dan  $\delta$  berubah – ubah sesuai kebutuhan.

#### 2. Bus Generator

Bus ini hanya terdapat daya pembangkitan dimana tegangan diatur menggunakan regulator tegangan dan P diatur dengan Governor. Sehingga untuk bus ini, P dan  $|V|$  diketahui. Sementara Q (daya reaktif) dan  $\delta$  (sudut fasa) dicari. Harga daya reaktif Q mempunyai nilai – nilai yang bervariasi tetapi masih dalam batas – batas tertentu. Dalam perhitungan iterasinya, bila nilai – nilai Q kembali memenuhi batas – batas yang ditentukan maka bus tersebut diubah kembali menjadi bus generator.

### 3. Bus Slack

Pada sistem tenaga listrik jumlah daya yang dibangkitkan harus sama dengan jumlah daya permintaan ditambah kerugian pada saluran transmisi. Jika kerugian transmisi belum dapat ditentukan sebelumnya, maka jumlah daya pembangkit belum dapat ditentukan. Oleh karena itu digunakan metode dimana salah satu bus generator dipakai untuk memenuhi tiap daya tambahan sesuai dengan kebutuhan diluar parameter yang ditentukan untuk bus – bus generator lainnya guna mengimbangi beban sistem ditambah kerugian saluran transmisi. Hal ini diberikan karena kerugian daya pada saluran transmisi baru bisa ditentukan setelah proses perhitungan pada bus – bus selesai dilakukan. Pada bus slack ini tegangan  $|V|$  dan sudut fasa  $\delta$  sudah ditentukan besarnya, sementara daya aktif P dan daya reaktif Q dihitung. Sedangkan sudut fasa tegangan berharga nol, karena pada bus ini phasor tegangan dari bus dipakai sebagai referensi.

#### 2.11. Analisa Sensitivitas<sup>[6]</sup>

Pada perencanaan dan perancangan sistem tenaga listrik, penting untuk mengetahui pengaruh perubahan beberapa variabel sistem terhadap performance sistem. Apabila terjadi perubahan pada harga suatu variabel terhadap variabel yang lain, ketergantungan antara variabel – variabel tersebut dapat ditentukan dengan apa yang dinamakan analisa sensitivitas. Dengan analisa sensitivitas tersebut dapat diketahui variabel mana yang perubahannya paling berpengaruh

pada perubahan variabel yang lain dan besarnya perubahan yang terjadi pada variabel yang berpengaruh tersebut.

Bila  $Y = F(x_i)$  adalah suatu fungsi rangkaian dan  $x$  menyatakan parameter fisis yang berhubungan dengan rangkaian itu, sensitivitas dari fungsi rangkaian  $Y$  terhadap perubahan  $x_i$  asalkan semua perubahan itu kecil secara matematis dapat ditulis sebagai :

$$S_{x_i} = \frac{\partial y}{\partial x_i} \dots\dots\dots (2.25)$$

Untuk memperoleh  $S_{x_i}$  dilakukan analisa rangkaian dengan nilai nominal dari  $x_1, x_2, \dots, x_n$  seperti yang diuraikan dibawah ini.

Dianggap bahwa kondisi operasi sistem dalam keadaan tetap (*Steady-State*). Untuk harga – harga nominal  $X_0, U_0$  dan  $P_0$  dalam keadaan ini dapat dinyatakan sebagai fungsi yang terdiri dari Variabel – Variabel yang diatur, variabel pengganggu, dan variabel pengatur persamaan ini dapat ditulis :

$$F(X_0, U_0, P_0) = 0 \dots\dots\dots (2.26)$$

Bila vektor variabel  $U$  dan variabel  $P$  mengalami perubahan sebesar  $\Delta U$  dan  $\Delta P$  dari harga nominalnya maka vektor variabel tak bebas akan berubah dari  $X_0$  menjadi  $X_0 + \Delta X$ , agar sistem itu tetap dalam keadaan keseimbangan yang secara matematis memenuhi persamaan (2.26), dan dapat dinyatakan dengan :

$$F ( X_0 + \Delta X, U_0 + \Delta U, P_0 + \Delta P) = 0 \dots\dots\dots (2.27)$$

$$= F(X_0, U_0, P_0) + F_X \Delta X + F_U \Delta U + F_P \Delta P \dots\dots\dots (2.28)$$

Mengingat sistem dalam keseimbangan sebelum perubahan – perubahan itu terjadi maka persamaan (2.28) menjadi :

$$F_X \Delta X + F_U \Delta U + F_P \Delta P = 0 \dots\dots\dots (2.29)$$



Dimana  $F_X, F_U, F_P$  adalah elemen – elemen matriks jacobian. Kemudian elemen – elemen tersebut dapat dihitung dengan persamaan sebagai berikut :

$$F_X = \frac{\partial F}{\partial X} \mid X_0, U_0, P_0 \dots\dots\dots (2.30)$$

$$F_U = \frac{\partial F}{\partial U} \mid X_0, U_0, P_0 \dots\dots\dots (2.31)$$

$$F_P = \frac{\partial F}{\partial P} \mid X_0, U_0, P_0 \dots\dots\dots (2.32)$$

$$\begin{bmatrix} \frac{\partial F_1}{\partial X_1} & \frac{\partial F_1}{\partial X_2} & \dots\dots\dots & \frac{\partial F_1}{\partial P_{2n}} \\ \frac{\partial F_2}{\partial X_1} & \frac{\partial F_2}{\partial X_2} & \dots\dots\dots & \frac{\partial F_2}{\partial X_{2n}} \\ \frac{\partial F_{2n}}{\partial X_1} & \frac{\partial F_{2n}}{\partial X_2} & \dots\dots\dots & \frac{\partial F_{2n}}{\partial X_{2n}} \end{bmatrix} \begin{bmatrix} 0 \\ \Delta X_2 \\ \Delta X_{2n} \end{bmatrix} + \begin{bmatrix} \frac{\partial F_1}{\partial U_1} & \frac{\partial F_1}{\partial U_2} & \dots\dots\dots & \frac{\partial F_1}{\partial U_{2n}} \\ \frac{\partial F_2}{\partial U_1} & \frac{\partial F_2}{\partial U_2} & \dots\dots\dots & \frac{\partial F_2}{\partial U_{2n}} \\ \frac{\partial F_{2n}}{\partial U_1} & \frac{\partial F_{2n}}{\partial U_2} & \dots\dots\dots & \frac{\partial F_{2n}}{\partial U_{2n}} \end{bmatrix} \begin{bmatrix} \Delta U_1 \\ \Delta U_2 \\ \Delta U_{2n} \end{bmatrix} + \begin{bmatrix} \frac{\partial F_1}{\partial P_1} & \frac{\partial F_1}{\partial P_2} & \dots\dots\dots \\ \frac{\partial F_2}{\partial P_1} & \frac{\partial F_2}{\partial P_2} & \dots\dots\dots \\ \frac{\partial F_{2n}}{\partial P_1} & \frac{\partial F_{2n}}{\partial P_2} & \dots\dots\dots \end{bmatrix} \begin{bmatrix} \Delta P_1 \\ \Delta P_2 \\ \Delta P_{2n} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \dots\dots\dots (2.33)$$

Secara singkat dapat ditulis:

$[F_x] [X] + [F_U] [U] + [F_P] [\Delta P] = 0$ ; Dimana:  $[F_X], [F_L], [F_P]$  adalah matriks jacobian. Dan  $[X], [U], [P]$  adalah vektor *increment* dari variabel yang diatur X, variabel pengatur P dan variabel Pengganggu U.

Agar variabel yang diatur merupakan hasil pengaturan dinyatakan secara eksplisit, untuk itu perlu dirubah sebagai berikut:

$$\Delta X = -F_X^{-1} \cdot F_U \Delta U - F_X^{-1} \cdot F_P \Delta P \dots\dots\dots (2.34)$$

Selanjutnya :

$$S_U = -F_X^{-1} \cdot F_U \dots\dots\dots (2.35)$$

$$S_P = -F_X^{-1} \cdot F_P \dots\dots\dots (2.36)$$

Maka sensitivitas dari variabel tak bebas terhadap perubahan variabel kontrol  $\Delta U$  dan perubahan parameter tetap  $\Delta P$  dapat dihitung dengan persamaan dibawah ini :

$$\Delta X = S_U \cdot \Delta U + S_P \cdot \Delta P \dots\dots\dots (2.37)$$

Apabila hanya terjadi perubahan nilai pada variabel kontrol maka  $\Delta P=0$ , dengan demikian hanya perlu menurunkan persamaan mismatch daya terhadap variabel-variabel kontrol.

Persamaan sensitivitas untuk keadaan ini adalah:

$$\Delta X = S_U \cdot \Delta U \dots\dots\dots (2.38)$$

## **BAB III**

### **APLIKASI METODE REINFORCEMENT LEARNING DALAM PENGONTROLAN DAYA REAKTIF**

#### **3.1 Saluran Transmisi**

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

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

1. Saluran transmisi pendek dengan panjang saluran kurang dari 80 km.
2. Saluran transmisi menengah dengan panjang saluran antara 80 km–240 km.
3. Saluran transmisi panjang dengan panjang saluran lebih dari 240 km.

#### **3.1 Studi Aliran Beban <sup>[2]</sup>**

Studi beban adalah penentuan atau perhitungan tegangan, arus, daya, dan faktor daya atau daya reaktif yang terdapat pada berbagai titik dalam suatu jaringan listrik pada keadaan pengoperasian normal, baik yang sedang berjalan maupun yang diharapkan akan terjadi dimasa yang akan datang. Studi beban sangat penting dalam perencanaan pengembangan suatu sistem untuk masa yang

akan datang., karena pengoperasian yang baik dari sistem tersebut banyak tergantung pada diketahuinya efek interkoneksi dengan sistem tenaga yang lain, beban yang baru, stasiun pembangkit baru, serta saluran transmisi baru, sebelum semuanya itu dipasang. Betapa pentingnya studi aliran beban dalam merencanakan perluasan sistem tenaga dan dalam menentukan operasi terbaik untuk sistem yang telah ada. Keterangan utama yang diperoleh dari suatu studi aliran beban adalah besar dan sudut fasa tegangan pada setiap rel dan daya nyata dan reaktif yang mengalir pada setiap saluran.

Disamping membahas bagaimana studi-studi aliran beban dilakukan pada sebuah komputer, bab ini telah pula memyajikan beberapa metode pengaturan tegangan dan aliran daya dilihat dari segi pengertian bagaimana pengaturan ini dapat dilakukan. Studi aliran beban pada sebuah komputer adalah cara yang terbaik untuk memperoleh jawaban-jawaban kuantitatif tentang pengaruh operasi-operasi pengaturan tertentu.

Tujuan studi aliran beban ialah :

- Untuk memeriksa tegangan-tegangan pada setiap rel (bus) yang ada dalam sistem yang dipelajari, biasanya variasi tegangan yang diijinkan adalah  $\pm 5\%$ .
- Memeriksa kapasitas semua peralatan yang ada dalam sistem apakah cukup besar untuk menyalurkan daya yang diinginkan.
- Untuk memperoleh kondisi mula studi selanjutnya yaitu:  
Studi hubung singkat, studi stabilitas, dan juga studi rugi-rugi transmisi.

Informasi-informasi diatas sangat penting untuk menganalisa keadaan sekarang dari sistem dan guna perancangan pengembangan sistem selanjutnya.

### 3.1 Masalah CLF (Constrained Load flow) [1]

Masalah aliran beban dapat dinyatakan dengan dua himpunan persamaan tidak linier sebagai berikut :

$$Y = g(X, U)$$

$$Z = h(X, U) \dots \dots \dots (1)$$

Dimana

Vektor Inputan Y, meliputi node (simpul) injeksi daya

Vektor Inputan Z, meliputi aliran daya, daya reaktif pada bus PV, dsb.

Vektor keadaan X, meliputi besaran dan tegangan sudut

Vektor kontrol U, meliputi posisi tap setting trafo, besarnya kompensasi *shunt*, Daya dan tegangan pada bus PV beban yang harus dibuang, dsb.

Sasaran aliran beban terkendala (CLF) adalah untuk memelihara beberapa atau semua elemen vektor X dan Z dengan batas operasi yang diberikan dalam ketersediaan unit pembangkit yang tidak pasti dan beban yang tidak pasti.

kendala seperti itu biasanya merupakan himpunan besaran tegangan pada bus beban, daya reaktif dan daya aktif yang diinjeksikan (dimasukkan) pada bus generator, aliran daya nyata pada saluran, dsb. Operasi dalam batas kendala dapat dicapai dengan variasi yang sesuai dari variabel kontrol U, yang juga terkendala, yaitu batasan fisik yang ada didalam variasi tap *transformator*, peralatan kompensasi *shunt*, tegangan pada bus PV, dsb.

Secara umum, suatu solusi aliran beban yang tidak terkendala akan menghasilkan suatu bilangan variabel X dan Z diluar *interval* variasi yang diijinkan untuk membatasi variabel-variabel itu, tindakan pada variabel kontrol yang dibutuhkan. Tindakan ini akan menjadi dasar dari hasil analisa sensitifitas,

yaitu menghitung faktor sensitifitas pada setiap variabel yang perlu dibatasi terhadap variabel kontrol.

### 3.2 Reinforcement Learning (RL) <sup>[1]</sup>

Teknik *Reinforcement Learning* adalah iterative algoritma sederhana yang belajar untuk bertindak sebagai suatu cara optimal sampai mengalami yang diperoleh dengan menyelidiki suatu sistem yang tidak dikenal. *RL* berasumsi bahwa “world” dapat diuraikan oleh satu set keadaan dan suatu agen dapat memilih satu tindakan dari satu set tindakan  $A$ . rentang operasi adalah dibagi menjadi langkah pembelajaran diskrit. pada masing-masing yang langkah pembelajaran, agen mengamati keadaan sekarang “world” ( $s \in S$ ), dan memilih suatu tindakan  $a \in A$  itu yang cenderung untuk memaksimalkan suatu nilai jangka panjang yang diharapkan berfungsi . setelah mengambil tindakan ( $a$ ), agen diberi suatu *immediate reward*  $r \in R$ , menyatakan efektivitas tindakan dan pengamatan hasil keadaan “world” . tujuan *RL* adalah untuk mengambil pengalaman ini tuples ( $s, a, r, s'$ ), dan belajar suatu pemetaan dari keadaan ( atau keadaan dan tindakan, tergantung pada penggunaan algoritma tertentu) untuk suatu ukuran nilai jangka panjang mengambil tindakan ini, di kenal sebagai nilai fungsi optimal.

Algoritma *RL* tertentu digunakan di dalam pekerjaan ini adalah algoritma Q-learning. nilai fungsi Q-learning optimal digambarkan atas pertolongan persamaan *Bellman*, sebagai berikut :

$$Q^*(s,a) = E(r_{(s,a)} + \gamma \max_a Q^*(s',a')) \dots\dots\dots(2)$$

Ini menghadirkan jumlah *rewards* yang diharapkan, ketika permulaan dari suatu keadaan awal (*s*) pengambilan tindakan (*a*), dan melakukan tindakan optimal (*a'*) di dalam pencarian berikutnya, sampai nilai Q-Function yang optimal, ( $Q^*(s, a)$ ) dicapai.  $E$  menghadirkan jumlah perhitungan yang di harapan di atas nilai-nilai. Potongan parameter  $\gamma$  ( $0 \leq \gamma \leq 1$ ) digunakan untuk mengurangi bobot eksponensial *reward* di dalam pencarian berikutnya.

Sekali kita mempunyai nilai yang optimal  $Q^*(s, a)$ , adalah mudah untuk menentukan tindakan yang optimal  $a^*$  menggunakan suatu kebijakan *greedy*. Suatu cara sederhana adalah untuk melihat pada seluruh tindakan yang memungkinkan (*a*) dari keadaan yang ditentukan (*s*) dan memilih satu dengan nilai yang paling besar.

$$a^* = \arg \max_a Q_a^*(s, a) \dots \dots \dots (3)$$

Fungsi-Q ( Q-Memory) secara khas disimpan di dalam suatu tabel, indeks keadaan dan tindakan. Mulai dengan nilai-nilai sembarang, kita dapat mengulangi pendekatan Function-Q yang optimal berdasarkan pada kriteria pengoptimalan kami. Tabel masukan untuk keadaan (*s*) dan tindakan (*a*) kemudian di perbaharui menurut:

$$Q(s, a) = (1 - \alpha) \cdot Q(s, a) + \alpha \left( r + \gamma \max_{a'} Q(s', a') \right) \dots \dots \dots (4)$$

Merupakan penting untuk dicatat bahwa nilai baru untuk  $Q(s, a)$  memori didasarkan kedua-duanya pada nilai sekarang  $Q(s, a)$ , dan nilai-nilai (*immediate reward*) dari tindakan pengendalian yang diperoleh dari pencarian berikutnya. Maka, parameter  $\alpha$  ( $0 \leq \alpha \leq 1$ ), menggunakan suatu peran kritis yang mewakili

jumlah  $Q$ -memory yang di perbaharui dan mempengaruhi banyaknya jumlah iterasi. Parameter  $(1 - \alpha)$  menghadirkan jumlah total nilai- $Q$  menenangkan yang tersisa sebagai memori di (dalam) fungsi- $Q$ .

Di bawah beberapa kondisi-kondisi layak, pers. (4) dijamin konvergen untuk nilai  $Q^*$  yang optimal ( $Q$ -memori yang terbaik) karena masing-masing penghembus  $(s, a)$  di dalam suatu batas jumlah iterasi. Ini diasumsikan yang berhubungan dengan *nonchaotic* alam (yaitu tiap-tiap penghembus  $(s, a)$  akan mengunjungi dengan jumlah waktu yang terbatas) dan jumlah dari tindakan kontrol dan / atau keadaan solusi dari sistem *learning* (metode diskret). Satu dari beberapa fitur terpenting *Q-learning* adalah apa yang perlu diketahui sama seperti algoritma *off-policy*. Rata-rata distribusi kemungkinan ini dimana dari contoh training  $(s, a)$  yang mana terpengaruh, di dalam batasan, pada proses *Q-learning* pers. (4). Ini akan dikembangkan untuk menjadi sangat penting untuk masalah kami karena tidak selalu digunakan pada banyaknya kebijakan eksplorasi untuk algoritma *Q-learning* kami.

### 3.4 Pengaplikasian *Q-learning* Untuk Pengontrolan Daya Reaktif<sup>[1]</sup>

Untuk kepentingan analisa kami, dua kelas klasifikasi diasumsikan. "*World*" tentang solusi keadaan  $RL$  ( $s \in S$ ) adalah biner, berisi karakteristik titik operasi dengan kepuasan dari seluruh terkendala dan tak dapat diterima, ketika manapun terkendala menyimpang. vektor kontrol yang dikombinasikan nilai-nilai diskrit dari penyesuaian kendali yang terpisah adalah tindakan ( $a \in A$ ), dan algoritma *Q-Learning* adalah "agen" proses algoritmanya sebagai berikut : Suatu titik operasi berisikan suatu beban dan pola pembangkitan yang mencakup satu himpunan tindakan pengendalian diciptakan secara acak. Agen mengamati



keadaan dari sistem (s), seperti diperoleh dengan solusi aliran beban, dan memilih satu tindak pengendali (a) dari vektor kendali. Suatu aliran beban baru dieksekusi. Agen mengamati hasil keadaan dari solusi (s') dan menyediakan suatu *immediate reward* (r) :  $S \times A \rightarrow \mathfrak{R}$  yang menyatakan derajat tingkat kepuasan batas operasi dari variabel terkendala. Suatu tindak pengendalian baru terpilih pada berikutnya, mendorong ke arah suatu solusi aliran beban baru dan *reward* baru. Pemilihan suatu tindakan pengendalian baru diulangi sampai tidak ada perubahan di dalam nilai *reward* yang dapat dicapai. Sasaran agen adalah untuk pembelajaran Fungsi-Q yang optimal ( $Q^*(s, a)$ ) Penggunaan pemetaan keadaan untuk suatu tindakan ( $S \rightarrow A$ ) seperti *reward* jangka panjang dimaksimalkan. Prosedur diulangi untuk sejumlah besar operasi keadaan yang mencakup keseluruhan periode perencanaan. Agen menemukan setting kontrol *greedy* optimal ( $a^*$ ) menggunakan kebijakan yang diuraikan oleh pers (3).

**3.5 Vektor Keadaan (State) <sup>[1]</sup>**

Dalam rangka membatasi variabel terkendala  $z_j$  di dalam nya beroperasi batasan ( $z_{j \min}, z_{j \max}$ ), keadaan sistem diskret adalah sebagai berikut : Ketika salah satu dari variabel  $z_j$  berada di luar batasan operasi, tingkat-keadaan diperlakukan sebagai keadaan -1, jika tidak dipertimbangkan maka tingkat-keadaan nol. Sebagai konsekwensi, jika kita mempunyai variabel terkendala-n, total bilangan keadaan adalah :

$$\bar{S} = 2^n \dots\dots\dots(5)$$

**3.6 Vektor Tindakan (Action) <sup>[1]</sup>**

Jika masing-masing tingkatan variabel pengendali  $u_i$ , diskret  $d_{u_i}$ , jumlah total vektor tindakan mengakibatkan aliran beban adalah

$$\bar{A} = \prod_{i=1}^m d_{u_i} \dots\dots\dots(6)$$

Di mana  $m$  menyatakan jumlah total variabel kontrol.

Dalam hal variabel kontrol berlanjut, *finer* adalah interval diskret, pencapaian *Q-Learning* makin lebih baik, dan langkah-langkah *Learning* lebih besar.

### 3.7 Rewards<sup>[1]</sup>

Aplikasi algoritma *Q-Learning* ke dalam masalah pengontrolan daya reaktif dihubungkan kepada pemilihan dari suatu *immediate reward* ( $r$ ), seperti nilai iterasi dari Fungsi-Q pers. (4) dimaksimalkan. Sedangkan batas variabel terkendala dicukupi untuk seluruh periode perencanaan. Suatu strategi empirik yang dikuatkan adalah untuk mempertimbangkan variasi variabel terkendala dekat dengan rata-rata operasi interval sebagai ukuran *reward*. *immediate reward* ( $r$ ) dihitung dengan rata-rata dari seluruh variabel terkendala, yang dinormalisir di dalam interval :  $[0, 1]$ , sebagai berikut :

$$r = -\frac{1}{n} \cdot \sum_{j=1}^n \left| \frac{2z_j - z_{j \max} - z_{j \min}}{z_{j \max} - z_{j \min}} \right| \dots\dots\dots(7)$$

Di mana  $n$  menyatakan banyaknya variabel terkendala, nilai  $z_j$  dari variabel terkendala- $j$  dan  $(z_{j \min}, z_{j \max})$  adalah batas bawah dan batas atas. Sekali kita mengoptimalkan tindakan (3) untuk sejumlah besar titik operasi, salah satu memaksimalkan penjumlahan *reward* di atas seluruh periode perencanaan digambarkan sebagai greedy-optimal tindakan ( $a^*$ ) dari *Q-Learning*.

$$\text{mtr}(a^*) = \text{max total reward}$$

$$= \max_{\pi^*(s)} \left( \sum_{\text{diatas seluruh periode perencanaan}} \cdot r \right) \dots\dots\dots(8)$$

Dimana  $a^*$  : arg max  $(\pi^*(s))$  diatas seluruh periode perencanaan.

### 3.8 Parameter *Discount* ( $\gamma$ ) dan *Learning Step* ( $\alpha$ ) <sup>[1]</sup>

Sebagai tambahan terhadap definisi di atas, dua parameter  $\gamma$  dan  $\alpha$  untuk penerapan algoritma *Q-Learning* perlu untuk dipilih.

Parameter  $\gamma$ , memperkenalkan di pers. (2), adalah faktor kontrol yang mana reward berikutnya adalah *discounted* dan itu harus antara 0 dan 1. didalam pengaplikasian kami, reward berikutnya tidaklah terlalu penting karena disini tidak ada saling ketergantungan antar solusi aliran beban yang diproduksi oleh langkah-langkah *Q-Learning* ; oleh karena itu, nilai  $\gamma$  dipilih mendekati 0. pengalaman kami menunjukkan bahwa semua nilai di dalam batasan 0 - 0.01 pekerjaan dengan baik. Untuk yang berikutnya, suatu nilai 0.005 adalah digunakan.

Parameter  $\alpha$  kritis menggunakan pers. (4), menyatakan jumlah *Fungsi-Q* yang di perbaharui. Suatu parameter  $\alpha$  cukup besar ( mendekati 1) di iijinkan konvergen dengan cepat dari algoritma *Q-Learning*, sedangkan suatu nilai yang terkecil (mendekati 0) menghindari ketidak stabilan dari *Q-Learning*. Karena *Q-Learning* digunakan didalam masalah aliran beban terkendala yang tidak tergantung pada langkah-langkah *Q-Learning* sebelumnya seperti dinyatakan di

atas, parameter ini bekerja dengan baik mendekati 1. Di dalam pengaplikasiannya, dipilih suatu nilai 0,995.

## BAB IV

### HASIL DAN ANALISA HASIL

#### 4.1. Program Komputer Optimasi Daya Reaktif Menggunakan Metode *Reinforcment Learning*

Dalam penyelesaian masalah ini diperlukan bantuan program komputer dalam perhitungan yang membutuhkan ketelitian dan keakuratan. Program komputer dalam skripsi ini di jalankan dengan menggunakan bahasa Pemrograman Bordland Delphi versi 7.0 dan merupakan program yang mudah diaplikasikan .

##### 4.1.1. Algoritma Program

Algoritma aliran daya dengan metode *Newton Raphson* adalah sebagai berikut:

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

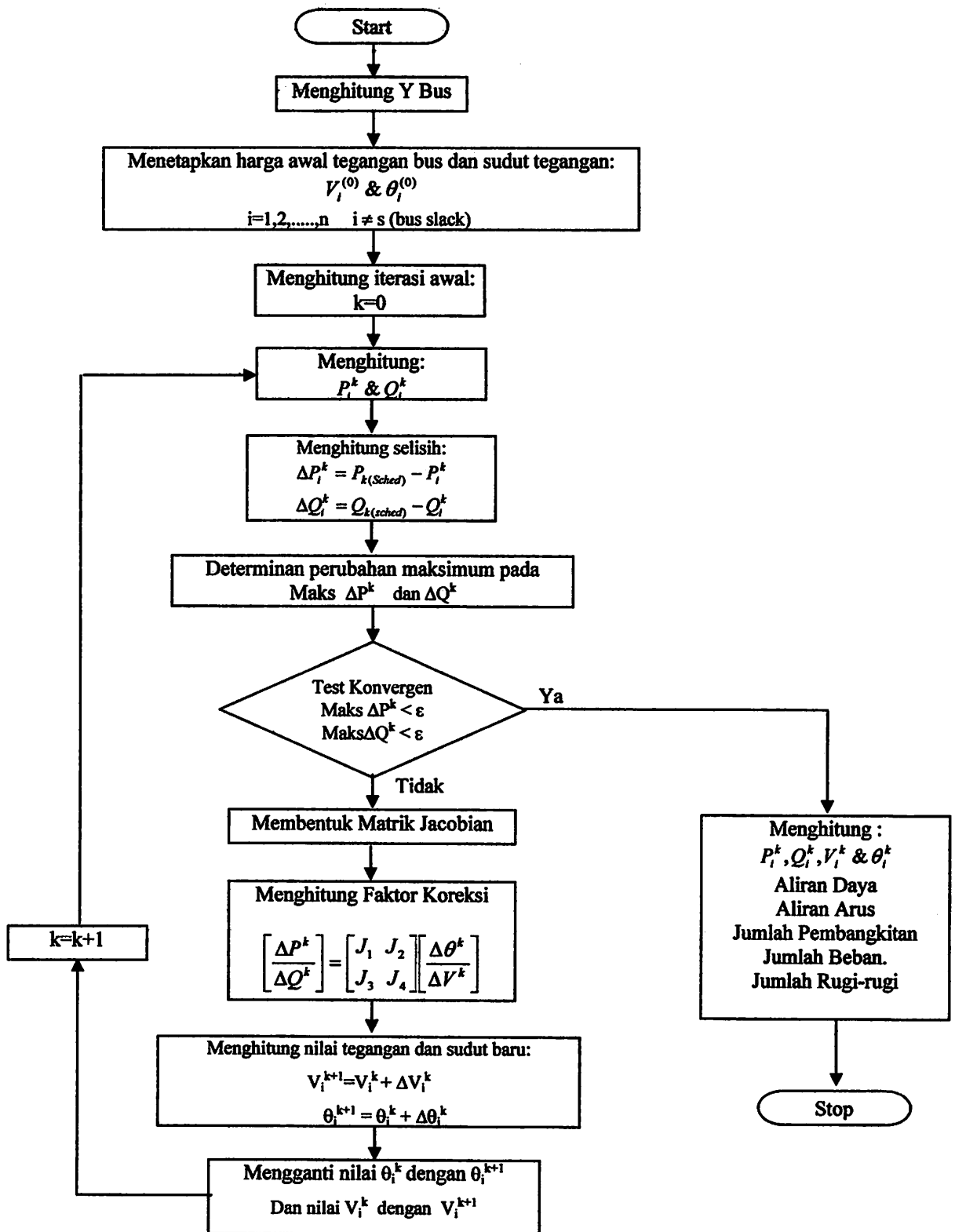
7. Membandingkan apakah selisih daya sudah sama atau lebih kecil dari  $\varepsilon$
8. Jika “ ya” hitung daya aktif dan reaktif , tegangan dan sudut fasa tegangan pada setiap bus, serta aliran daya pada saluran dan perhitungan selesai, jika “tidak” lanjutkan ke langkah berikutnya.
9. Membentuk element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

Selanjutnya untuk Algoritma program Optimasi Daya Reaktif menggunakan metode *RL* dilakukan dengan langkah-langkah sebagai berikut:

1. Mulai
2. Masukkan data aliran beban
3. Memilih tindakan kontrol (a) dari *vektor kontrol*
4. Eksekusi aliran beban
5. Memilih hasil keadaan ( $s'$ ) dari solusi aliran beban menggunakan pers. (6)
6. Menyediakan *immediate reward* menggunakan pers. (7)

7. Inisialisasi  $0 \leq \gamma \leq 1$
8. Memilih tindakan berikutnya menggunakan pers. (3)
9. Menghitung nilai fungsi  $Q$  – *learning optimal* menggunakan pers. (2)
10. Apakah nilai  $Q$  fungsi sudah mencapai nilai optimal , jika tidak kembali kelangkah 7 dengan  $1 - \gamma$
11. Jika Ya simpan nilai fungsi  $Q$  kedalam  $Q$  *memory*
12. Inisialisasi  $0 \leq \alpha \leq 1$
13. Update nilai  $Q$  fungsi menggunakan pers. (4)
14. Set  $s \leftarrow s'$
15. Apakah nilai *reward*  $\mathfrak{R}$  tidak ada perubahan, jika tidak kembali kelangkah 12 dengan  $1 - \alpha$
16. Jika Ya, hitung menggunakan  $a^* : \arg \max ( \pi *(s) )$
17. Tindakan sudah mencapai *Gready Optimal* ( $a^*$ ) dari  $Q$  – *Learning* dengan menggunakan pers. (8)
18. Cetak hasil optimal
19. Selesai

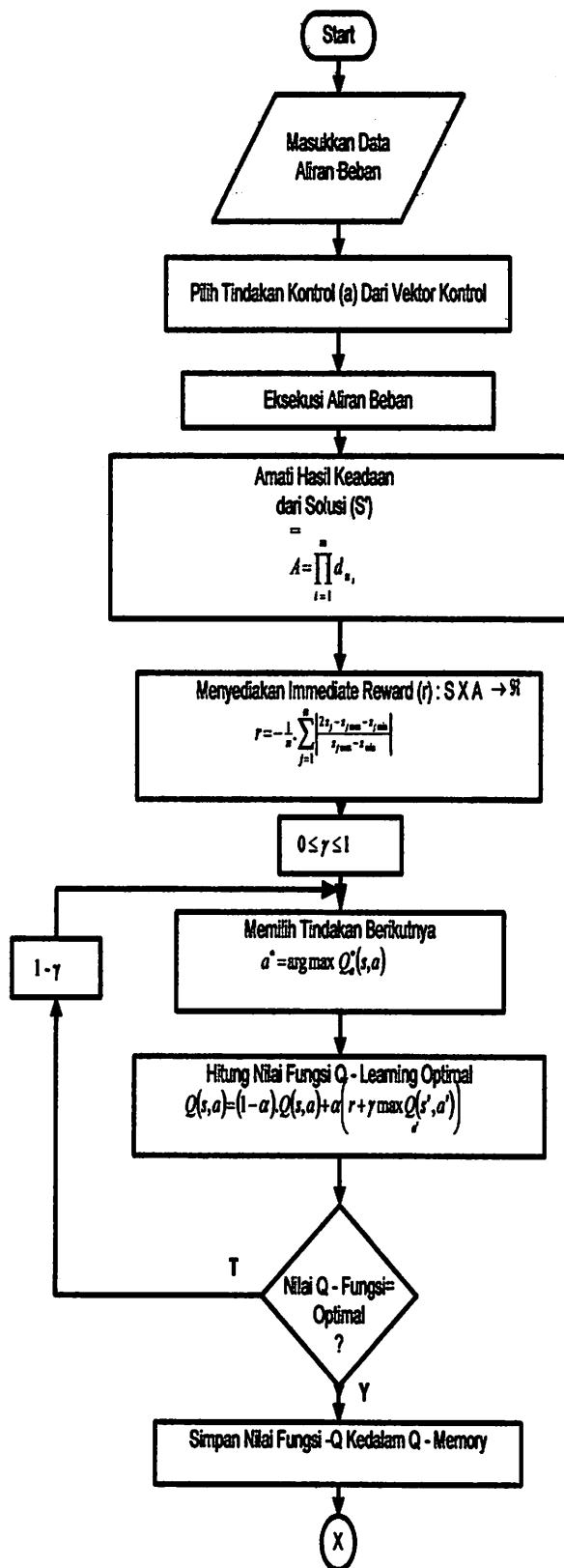
#### 4.1.2. Flowchart Program.

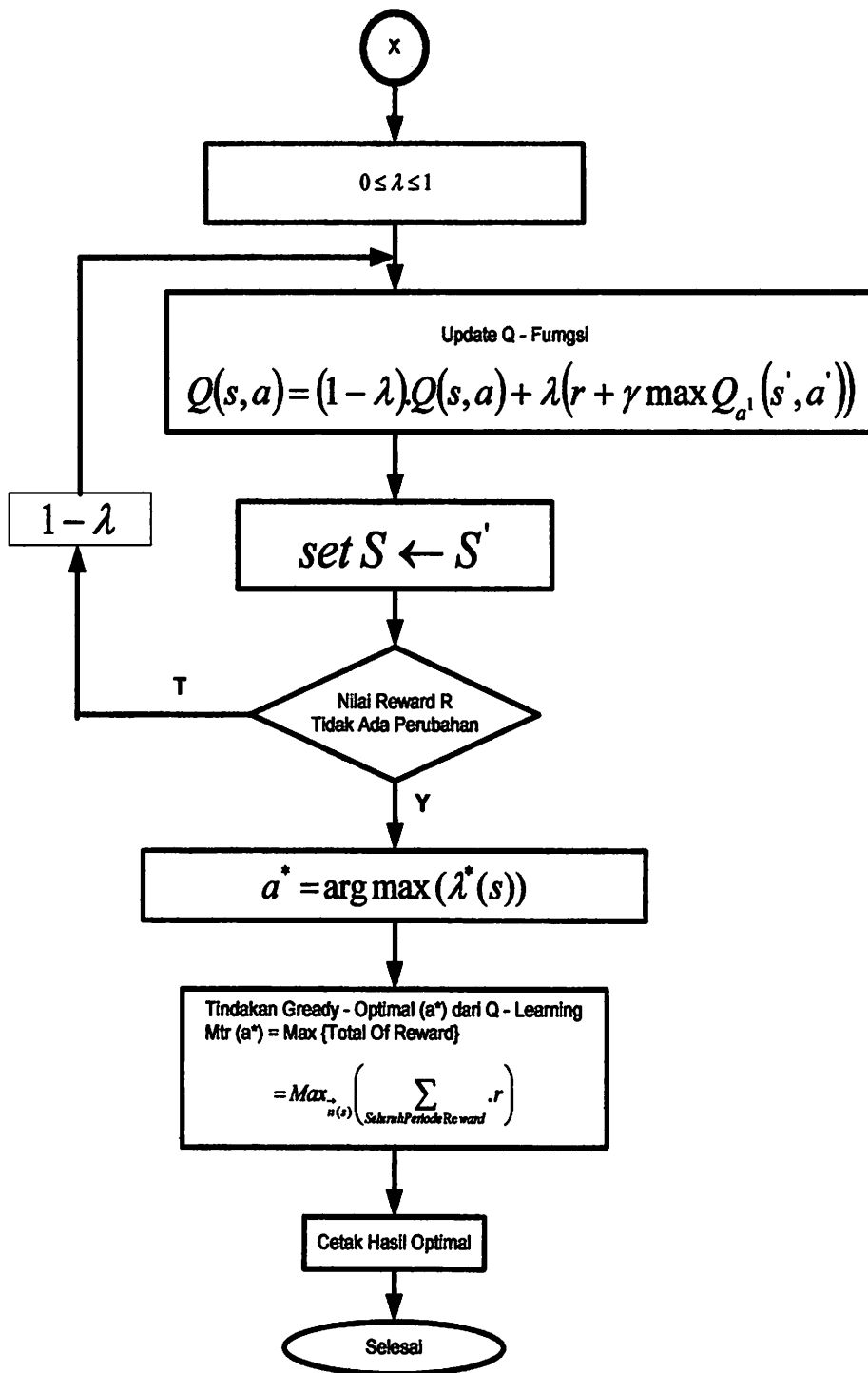


Gambar 4.1

Flow Chart Perhitungan Aliran Daya dengan Metode Newton Raphson







Gambar 4.2  
 Flowchart Program Reinforcement learning  
 untuk Pengontrolan Daya Reaktif

## 4.2. Validasi Program Menggunakan Data Sistem IEEE 14-Bus

### 4.2.1. Data Sistem IEEE 14-Bus

Tabel 4.1<sup>[1]</sup>  
Data Generator IEEE 14-Bus

No	Bus	Jenis	Pembangkitan		Pembebanan	
			P (MW)	Q (MVAR)	P (MW)	Q (MVAR)
1	1	Swing	35	0	0	0
2	2	Generator	15	30	-10	5
3	3	Generator	20	5	20	0
4	4	Beban	0	0	25.800	2
5	5	Beban	0	0	-5.	1
6	6	Generator	10	5	8	5
7	7	Beban	0	0	0	0
8	8	Beban	0	0	0	0
9	9	Beban	0	0	7	8
10	10	Beban	0	0	3	4
11	11	Beban	0	0	1	1
12	12	Beban	0	0	5.1	0.5
13	13	Beban	0	0	7.5	3.5
14	14	Beban	0	0	8.9	2

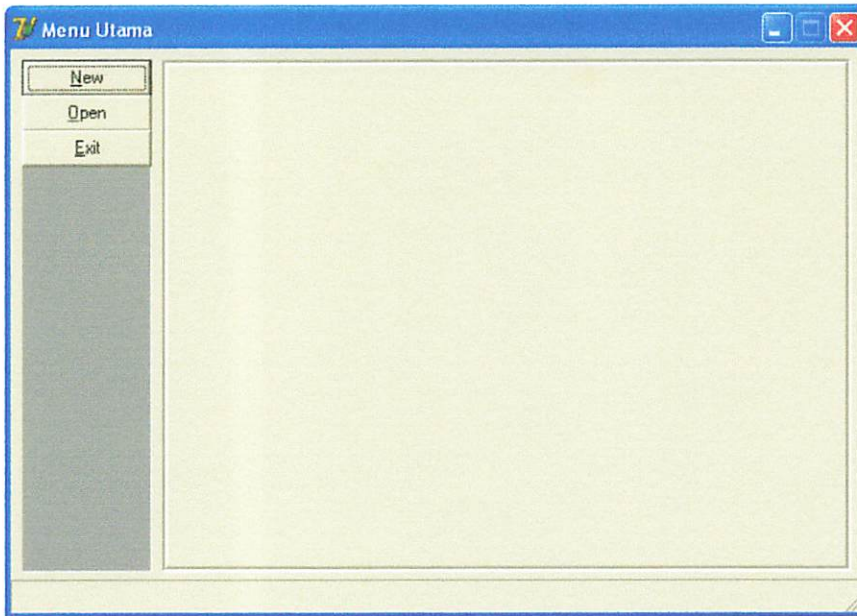
Tabel 4.2<sup>[1]</sup>  
Data Saluran IEEE 14-Bus

No	Saluran	R (pu)	X (pu)	Lc (pu)
1	1 – 2	0.02538	0.06717	0.03340
2	2 – 3	0.05679	0.11797	0.03190
3	2 – 4	0.06864	0.18632	0.02870
4	1 – 5	0.06403	0.23304	0.03460
5	2 – 5	0.06695	0.20388	0.02700
6	3 – 4	0.07705	0.18103	0.02730
7	4 – 5	0.02335	0.05211	0.00640
8	5 – 6	0.00000	0.26202	0.00000
9	4 – 7	0.00000	0.21912	0.00000
10	7 – 8	0.00000	0.18615	0.00000
11	4 – 9	0.00000	0.56618	0.00000
12	7 – 9	0.00000	0.12001	0.00000
13	9 – 10	0.04181	0.09450	0.00000
14	6 – 11	0.10498	0.20890	0.00000

15	6 – 12	0.13291	0.26581	0.00000
16	6 – 13	0.07615	0.14027	0.00000
17	9 – 14	0.13711	0.28038	0.00000
18	10 – 11	0.09205	0.20207	0.00000
19	12 – 13	0.23092	0.20988	0.00000
20	13 – 14	0.18093	0.35802	0.00000

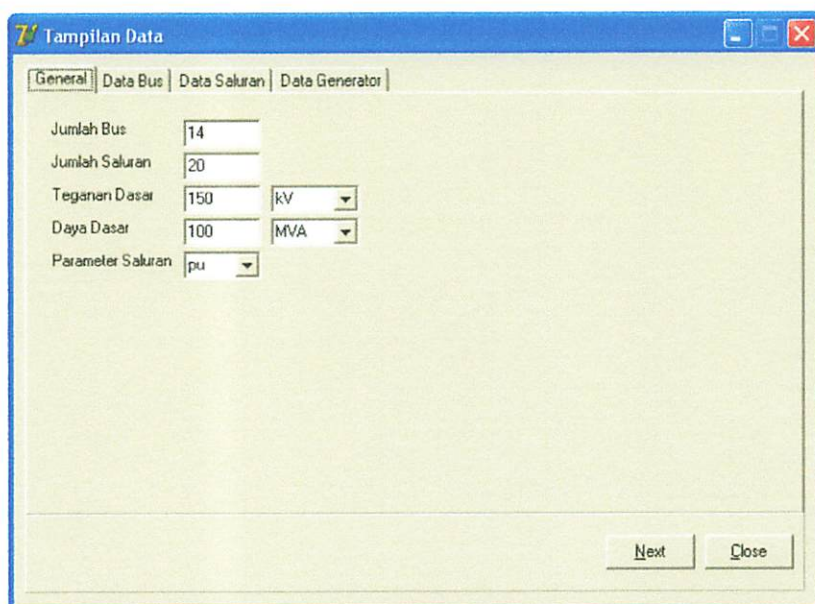
#### 4.2.2 Hasil Perhitungan Validasi IEEE 14 Dengan Menggunakan Metode *Reinforcement Learning*

Tampilan utama dari program computer ini dapat dilihat pada gambar dibawah ini:

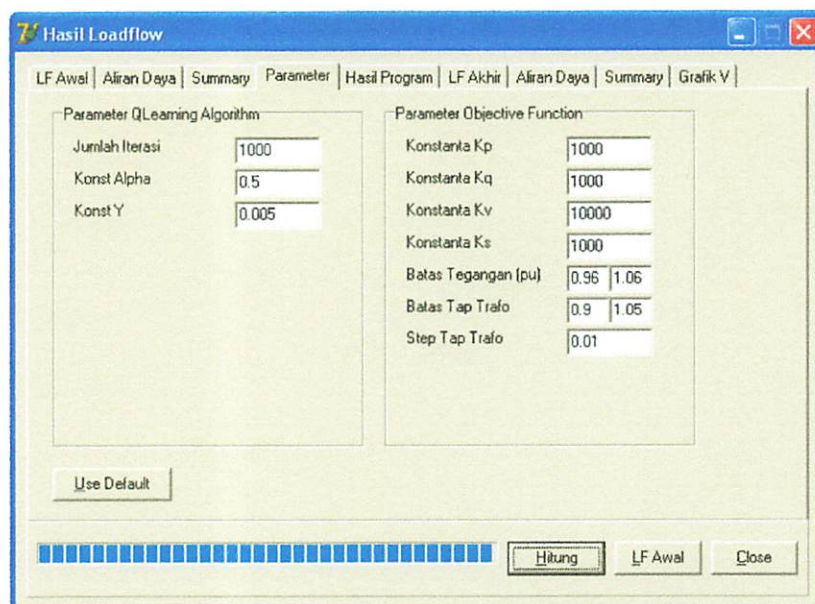


Gambar 4.3  
Tampilan program

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



Gambar 4.4  
Tampilan input data IEEE-14 Bus



Gambar 4.5  
Tampilan Parameter RL



Setelah menetapkan parameter tersebut, maka komputasi dapat dilakukan

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.02000	0.00000	27.219	3.036	0.000	0.000	0.000	1
2	1.01000	-0.00606	15.000	-31.262	-10.000	5.000	0.000	2
3	1.01000	-0.02017	20.000	-3.997	20.000	0.000	0.000	2
4	1.00768	-0.03984	0.000	0.000	25.800	2.000	0.000	3
5	1.01198	-0.03191	0.000	0.000	-5.000	1.000	0.000	3
6	1.01000	-0.06789	10.000	8.121	8.000	5.000	0.000	2
7	1.01689	-0.05205	0.000	0.000	0.000	0.000	0.000	3
8	1.01699	-0.06205	0.000	0.000	0.000	0.000	0.000	3
9	1.02047	-0.07410	0.000	0.000	7.000	8.000	0.200	3
10	1.01388	-0.07410	0.000	0.000	3.000	4.000	0.000	3
11	1.01042	-0.07164	0.000	0.000	1.000	1.000	0.000	3
12	1.00092	-0.08026	0.000	0.000	5.100	0.500	0.000	3
13	1.00017	-0.07955	0.000	0.000	7.500	3.500	0.000	3
14	1.00145	-0.08879	0.000	0.000	8.900	2.000	0.000	3

Gambar 4.6  
Hasil load flow

No	Bus	Pgen
1	Vg1	1.020
2	Vg2	1.010
3	Vg3	1.010
4	Vg6	1.010
5	Tr56	1.004
6	Tr47	0.997
7	Tr49	0.988

Rekapitulasi Losses		
Sebelum Optimasi	0.920	MW
Sesudah Optimasi	0.919	MW
Selalih Optimasi	0.001	MW

Gambar 4.7  
Hasil perhitungan program

Tabel 4.3  
Perbandingan Hasil Data Referensi Jurnal dengan Data Optimasi

Tegangan Keluaran	Data Referensi	Hasil Program
V4 (pu)	0.9951	1.00768
V5 (pu)	1.0030	1.01198
V7 (pu)	1.0274	1.01689
V8 (pu)	1.0274	1.01689
V9 (pu)	1.0284	1.02047
V10 (pu)	1.0193	1.01388
V11 (pu)	1.0067	1.01042
V12 (pu)	0.9915	1.00092
V13 (pu)	0.9903	1.00017
V14 (pu)	0.9977	1.00145

Hasil tegangan keluaran pada program menunjukkan bahwa hasil program masih di dalam batas  $V_{\min}$  dan  $V_{\max}$ . Menurut hasil program jurnal John G. Vlachogiannis, Nikos D. Hatziargyriou, *Senior Member, IEEE*, "Reinforcement Learning For Reactive Power Control," *IEEE Transaction On Power System*, Vol. 19, No. 3, August 2004 <sup>[4]</sup> Untuk *IEEE 14-bus System*.

Untuk proses validasi didapatkan selisih total rugi-rugi daya aktif yang berdasarkan hasil program dan hasil referensi, sehingga dapat dihitung persentase error sebagai berikut :

$$\begin{aligned}
 \text{Persentasi Error} &= \frac{\text{Hasil Re ferensi} - \text{Hasil Pr ogram}}{\text{Hasil Re ferensi}} \times 100\% \\
 &= \frac{0.920 - 0.919}{0.920} \times 100\% \\
 &= 0.10869\%
 \end{aligned}$$

Dengan tingkat *presentasi error*-nya masing-masing sebesar 0.10869 %  
Dapat dijelaskan pada Tabel 4.4 di bawah ini :

Tabel 4.4  
Perbandingan Total Rugi-Rugi

Total	Hasil Program	Hasil Referensi	Error (%)
Rugi-Rugi (MW)	0.919	0.920	0.10869 %

Dari hasil pengujian disini dapat dilihat bahwa program tersebut layak untuk digunakan, karena hasil perhitungan program tersebut mendekati hasil yang ada pada jurnal referensi. Pada tampilan program diatas dapat dilihat bahwa hasil perhitungan tidak menyimpang dari batasan variabel terkendala diatas, yang tercantum pada data hasil program dan data referensi didalam Tabel 4.4. Yang mendekati perhitungan pada jurnal John G. Vlachogiannis and Nikos D. Hatziargyriou, “Reinforement Learning for Reactive Power Control” , *IEEE Transaction on Power System* , Vol, 19, No.3, August 2004<sup>[4]</sup>. sehingga program ini dapat dikatakan valid untuk digunakan

#### 4.3 Data Pembangkitan dan Pembebanan Pada PLN P3B Region IV

Tabel 4.5<sup>[1]</sup>  
Penomoran Bus

No Bus	Nama Bus
1	<b>GRESIK LAMA</b>
2	<b>PITON</b>
3	<b>GDTAN 5</b>
4	<b>REJOSO</b>
5	<b>SITUBONDO</b>
6	<b>BANYUWANGI</b>
7	<b>KRAKSAAN</b>
8	<b>PROBOLINGGO</b>
9	<b>GENDING</b>
10	<b>LUMAJANG</b>
11	<b>BONDOWOSO</b>



12	<b>TANGGUL</b>
13	<b>JEMBER</b>
14	<b>GENTENG</b>
15	<b>PIER 5</b>
16	<b>BANGIL</b>
17	<b>BCKRO 5</b>
18	<b>BLKDN 5</b>
19	<b>LAWANG</b>
20	<b>KBAGN 5</b>
21	<b>SUTAMI</b>
22	<b>SENGKALING</b>
23	<b>PAKIS</b>
24	<b>WLINGI</b>
25	<b>JAKER 5</b>
26	<b>SURABAYA BARAT</b>
27	<b>ALTAP 5</b>
28	<b>DRIYOREJO</b>
29	<b>BBDAN 5</b>
30	<b>KASIH 5</b>
31	<b>CERME</b>
32	<b>MANYAR</b>
33	<b>GRESIK BARU</b>
34	<b>BANGKALAN</b>
35	<b>SAMPANG</b>
36	<b>PAMEKASAN</b>
37	<b>SUMENEP</b>
38	<b>GRATI</b>
39	<b>TANDES</b>
40	<b>PERAK</b>
41	<b>DARMO</b>
42	<b>SAWAHAN</b>
43	<b>UNDAAN</b>
44	<b>KUPANG</b>
45	<b>WARU</b>
46	<b>KLANG 5</b>
47	<b>RUNGKUT</b>
48	<b>SUKOLILO</b>
49	<b>KENJERAN</b>
50	<b>UJUNG</b>
51	<b>NGAGEL</b>
52	<b>SIMPANG</b>
53	<b>WONOKROMO</b>
54	<b>MKBAN 5</b>
55	<b>GLTMR 5</b>

<b>56</b>	<b>BUDURAN</b>
<b>57</b>	<b>SGMDU 5</b>
<b>58</b>	<b>PKMIA 5</b>
<b>59</b>	<b>LAMONGAN</b>
<b>60</b>	<b>BABAT</b>
<b>61</b>	<b>BOJONEGORO</b>
<b>62</b>	<b>TUBAN</b>
<b>63</b>	<b>KEREK</b>
<b>64</b>	<b>MLWNG 5</b>
<b>65</b>	<b>BLBND 5</b>
<b>66</b>	<b>SKTIH 5</b>
<b>67</b>	<b>NGORO</b>
<b>68</b>	<b>KERTOSONO</b>
<b>69</b>	<b>BNRAN 5 (A)</b>
<b>70</b>	<b>MJAGN 5</b>
<b>71</b>	<b>TUBAN 35</b>
<b>72</b>	<b>ISPDO5</b>

Sebelum memasukkan nilai data pada tabel 4.6, nilai-nilai tegangan pada data diubah satuannya menjadi satuan perunit. Berikut perhitungan konversi satuan dari KV menjadi pu:

- Pada Bus 1:  $V_{(pu)} = \frac{V_{daridata}(KV)}{V_{dasar}(KV)} = \frac{145.5}{150} = 0.97 pu$

Pada bus-bus selanjutnya, dilakukan hal yang sama seperti di atas dan hasilnya dapat dilihat pada tabel 4.6.

Untuk selanjutnya dilakukan perhitungan seperti langkah di atas, sehingga didapatkan hasil sebagai berikut :

Tabel 4.6.<sup>[1]</sup>  
 Data Pembangkitan dan Pembebanan  
 Pada PLN P3B Region IV

No Bus	Tegangan (pu)	P <sub>gen</sub> (MW)	Q <sub>gen</sub> (MVAR)	P <sub>load</sub> (MW)	Q <sub>load</sub> (MVAR)	Tipe Bus
1	0.97000	204.335	-69.111	0,000	0,000	1
2	1.05000	563.100	30.033	17,800	8,800	2
3	1.00587	0,000	0,000	27,000	12,100	3
4	1.00427	0,000	0,000	23,400	10,800	3
5	0.96162	0,000	0,000	20,300	1,300	3
6	0.95603	0,000	0,000	34,500	5,400	3
7	1.02771	0,000	0,000	2,200	1,000	3
8	1.00282	0,000	0,000	35,600	14,300	3
9	1.02331	0,000	0,000	17,500	8,700	3
10	0.97362	0,000	0,000	41,400	15,600	3
11	0.95764	0,000	0,000	17,000	3,700	3
12	0.96014	0,000	0,000	25,000	7,300	3
13	0.95505	0,000	0,000	50,500	22,800	3
14	0.95106	0,000	0,000	35,500	3,700	3
15	0.99255	0,000	0,000	12,000	2,700	3
16	0.98882	0,000	0,000	15,300	5,400	3
17	0.98754	0,000	0,000	20,600	4,500	3
18	0.99063	0,000	0,000	20,200	3,000	3
19	0.99416	0,000	0,000	20,600	6,900	3
20	1.00107	0,000	0,000	39,900	0,500	3
21	1.02000	30,000	41.884	0,000	0,000	2
22	0.99895	0,000	0,000	20,800	9,700	3
23	0.99873	0,000	0,000	30,000	12,000	3
24	1.03000	19,000	33.768	20.800	9.500	2
25	1.01000	0,000	120.122	15.700	13.000	3
26	0.98702	0,000	0,000	-500.400	-300.000	3
27	0.97387	0,000	0,000	7,000	4,000	3
28	0.98497	0,000	0,000	35,900	10,900	3
29	0.98215	0,000	0,000	35,800	15,600	3
30	0.97564	0,000	0,000	16,400	10,000	3
31	0.96689	0,000	0,000	12,000	6,000	3
32	0.96517	0,000	0,000	37,000	27,500	3
33	0.99000	410,300	0,000	0,000	0,000	2
34	0.96432	0,000	0,000	26,600	9,700	3
35	0.95313	0,000	0,000	20,500	8,000	3

36	0.95133	0,000	0,000	28,200	10,900	3
37	0.95086	0,000	0,000	31,900	11,000	3
38	1.03000	200,400	0,000	11,600	3,200	2
39	0.98280	0,000	0,000	57,900	22,100	3
40	0.99000	59,200	0,000	0,000	0,000	2
41	0.97931	0,000	0,000	38,300	25,000	3
42	0.98121	0,000	0,000	33,500	18,700	3
43	0.98120	0,000	0,000	12,700	6,400	3
44	0.98099	0,000	0,000	21,200	10,600	3
45	0.97470	0,000	0,000	87,000	44,000	3
46	0.98240	0,000	0,000	28,900	11,700	3
47	0.97019	0,000	0,000	100,300	32,100	3
48	0.96896	0,000	0,000	67,100	32,700	3
49	0.96793	0,000	0,000	48,300	24,100	3
50	0.96635	0,000	0,000	22,500	13,700	3
51	0.96879	0,000	0,000	14,400	7,400	3
52	0.96857	0,000	0,000	32,000	15,600	3
53	0.96880	0,000	0,000	32,800	14,800	3
54	0.97797	0,000	0,000	36,900	19,100	3
55	0.96891	0,000	0,000	6,200	2,700	3
56	0.97740	0,000	0,000	73,700	42,500	3
57	0.97321	0,000	0,000	17,200	7,800	3
58	0.97128	0,000	0,000	31,700	17,500	3
59	0.96850	0,000	0,000	27,500	20,600	3
60	0.95815	0,000	0,000	28,600	18,800	3
61	0.95106	0,000	0,000	39,000	27,300	3
62	0.95740	0,000	0,000	31,200	19,100	3
63	0.95902	0,000	0,000	41,600	-39,700	3
64	0.95836	0,000	0,000	11,300	7,000	3
65	0.98669	0,000	0,000	46,800	6,500	3
66	0.98777	0,000	0,000	50,700	23,800	3
67	0.98569	0,000	0,000	13,600	7,400	3
68	1.00196	0,000	0,000	0,000	0,000	3
69	1.00181	0,000	0,000	39,800	23,400	3
70	0.99713	0,000	0,000	30,400	15,600	3
71	0.95882	0,000	0,000	11,300	7,000	3
72	0.97462	0,000	0,000	16,000	2,700	3

Sumber : PT.PLN (Persero) P3B Region Jawa Timur Bali

Keterangan : 1. Bus Slack  
2. Bus Generator  
3. Bus Beban

#### 4.4 Data Saluran Transmisi Pada PLN P3B Region IV

Pada PLN P3B Region IV terdiri dari 72 bus dan 86 saluran transmisi, dalam hal ini saluran transmisi yang dibahas adalah saluran transmisi 150 kV.

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

$$\text{* saluran 2 (hubungan bus 1-45): } R_{\text{paralel}} = \frac{R_1 \cdot R_2}{R_1 + R_2} = \frac{0,0064 \cdot 0,0064}{0,0064 + 0,0064} = 0,0032$$

untuk X dan B serta R berikutnya yang jumlah sirkuitnya 2 dilakukan hal yang sama, hasil selengkapnya dapat dilihat pada tabel 4.7.

Untuk selanjutnya dilakukan perhitungan seperti langkah di atas, sehingga didapatkan hasil sebagai berikut :

Tabel 4.7<sup>[1]</sup>  
Data saluran transmisi 150 kV pada PLN P3B Region IV.

No Saluran	Hubungan Bus	Jumlah sirkuit	R (pu)	X (pu)	B (pu)
1	1 - 27	1	0.0013	0.0062	0.0025
2	1 - 45	2	0.0032	0.01545	0.0226
3	1 - 55	2	0.0008	0.0024	0.0017
4	1 - 57	2	0.0010	0.0035	0.0025
5	2 - 5	2	0.0477	0.0346	0.0502
6	2 - 7	2	0.0053	0.0180	0.0130
7	3 - 4	2	0.0028	0.0094	0.0067
8	3 - 8	2	0.0088	0.0301	0.0217
9	3 -15	2	0.0099	0.0072	0.0104
10	3 - 16	2	0.0044	0.0150	0.0108
11	3 - 38	2	0.0093	0.0067	0.0097
12	5 - 6	2	0.0097	0.0464	0.0680
13	5 - 11	2	0.0091	0.0310	0.0233
14	6 - 13	2	0.0430	0.1469	0.0265
15	6 - 14	1	0.0177	0.0604	0.0109
16	7 - 8	2	0.0079	0.0269	0.0194
17	7 - 9	1	0.0100	0.0340	0.0062
18	8 - 10	2	0.0135	0.0460	0.0332
19	10 - 12	1	0.0167	0.0571	0.0103



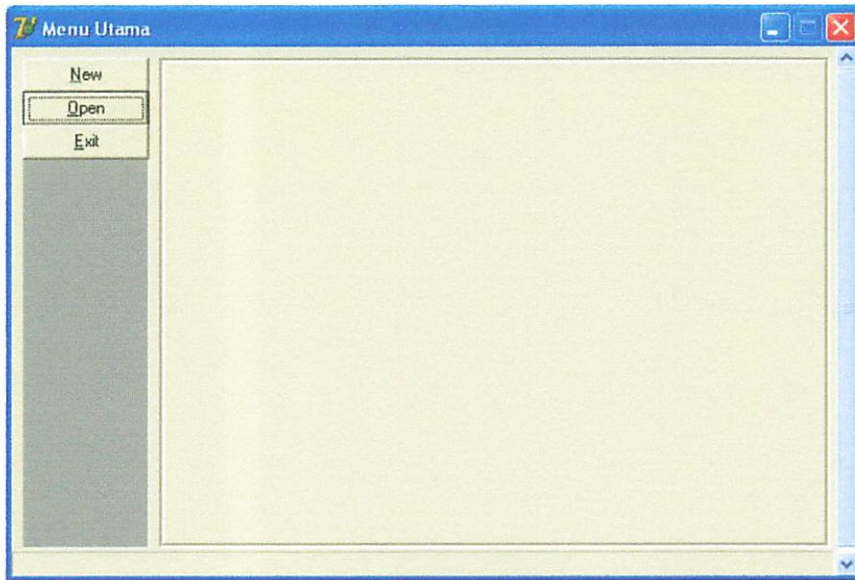
20	10 - 13	1	0.0315	0.1077	0.0195
21	11 - 13	2	0.0104	0.0353	0.0255
22	12 - 13	1	0.0148	0.0506	0.0092
23	13 - 14	1	0.0337	0.1151	0.0208
24	15 - 16	2	0.0045	0.0032	0.0047
25	15 - 23	2	0.0048	0.0343	0.0495
26	16 - 17	2	0.0054	0.0039	0.0056
27	16 - 18	1	0.0145	0.0494	0.0089
28	16 - 19	1	0.0181	0.0617	0.0112
29	16 - 45	1	0.0051	0.0174	0.0032
30	16 - 56	1	0.0051	0.0174	0.0032
31	18 - 19	1	0.0091	0.0309	0.0056
32	19 - 20	2	0.0067	0.0230	0.0166
33	20 - 21	2	0.0073	0.0249	0.0179
34	20 - 22	2	0.0046	0.0133	0.0098
35	20 - 23	2	0.0011	0.0081	0.0117
36	21 - 24	1	0.0224	0.0428	0.0074
37	25 - 68	1	0.0048	0.0144	0.0198
38	25 - 69	1	0.0083	0.0274	0.0053
39	26 - 27	2	0.0024	0.0113	0.0165
40	26 - 28	1	0.0028	0.0094	0.0017
41	26 - 29	2	0.0054	0.0185	0.0034
42	26 - 30	1	0.0620	0.0095	0.0035
43	26 - 31	2	0.0053	0.0527	0.0094
44	26 - 39	2	0.0039	0.0132	0.0095
45	26 - 42	2	0.0012	0.0101	0.0147
46	26 - 46	2	0.0009	0.0043	0.0063
47	26 - 65	1	0.0031	0.0149	0.0055
48	26 - 66	1	0.0063	0.0301	0.0110
49	27 - 57	1	0.0003	0.0013	0.0045
50	31 - 32	2	0.0010	0.0048	0.0070
51	33 - 39	2	0.0020	0.0079	0.0136
52	34 - 35	2	0.0290	0.0841	0.0155
53	34 - 55	2	0.0050	0.0144	0.0106
54	35 - 36	2	0.0161	0.0466	0.0086
55	35 - 37	1	0.0157	0.0454	0.0084
56	36 - 37	1	0.0279	0.0807	0.0149
57	39 - 40	2	0.0023	0.0079	0.0057
58	39 - 41	2	0.0012	0.0041	0.0029
59	39 - 42	2	0.0004	0.0020	0.0029
60	41 - 45	2	0.0028	0.0095	0.0069
61	42 - 43	2	0.0003	0.0009	0.0991
62	42 - 44	2	0.0008	0.0012	0.0665
63	42 - 45	2	0.0014	0.0067	0.0098
64	42 - 54	2	0.0060	0.0051	0.0036
65	45 - 46	2	0.0014	0.0067	0.0098
66	45 - 47	2	0.0006	0.0030	0.0044
67	45 - 72	2	0.0003	0.0011	0.0008

<b>68</b>	47 - 48	2	0.0003	0.0011	0.0008
<b>69</b>	48 - 49	2	0.0012	0.0004	0.0008
<b>70</b>	48 - 51	2	0.0004	0.0007	0.0759
<b>71</b>	48 - 53	2	0.0004	0.0013	0.1380
<b>72</b>	49 - 50	2	0.0025	0.0073	0.0053
<b>73</b>	51 - 52	2	0.0005	0.0009	0.0983
<b>74</b>	57 - 58	2	0.0021	0.0071	0.0051
<b>75</b>	57 - 59	2	0.0086	0.0248	0.0183
<b>76</b>	59 - 60	2	0.0064	0.0185	0.0136
<b>77</b>	60 - 61	2	0.0108	0.0311	0.0229
<b>78</b>	60 - 62	2	0.0042	0.0200	0.0293
<b>79</b>	62 - 63	2	0.0037	0.0125	0.0090
<b>80</b>	63 - 64	2	0.0024	0.008	0.0058
<b>81</b>	63 - 71	2	0.0006	0.0018	0.0013
<b>82</b>	65 - 66	1	0.0017	0.0083	0.0031
<b>83</b>	66 - 67	2	0.0066	0.0191	0.0141
<b>84</b>	66 - 70	2	0.0055	0.0187	0.0135
<b>85</b>	68 - 69	1	0.0083	0.0274	0.0053
<b>86</b>	69 - 70	2	0.0054	0.0182	0.0132

#### **4.5 Prosedur pelaksanaan program perhitungan.**

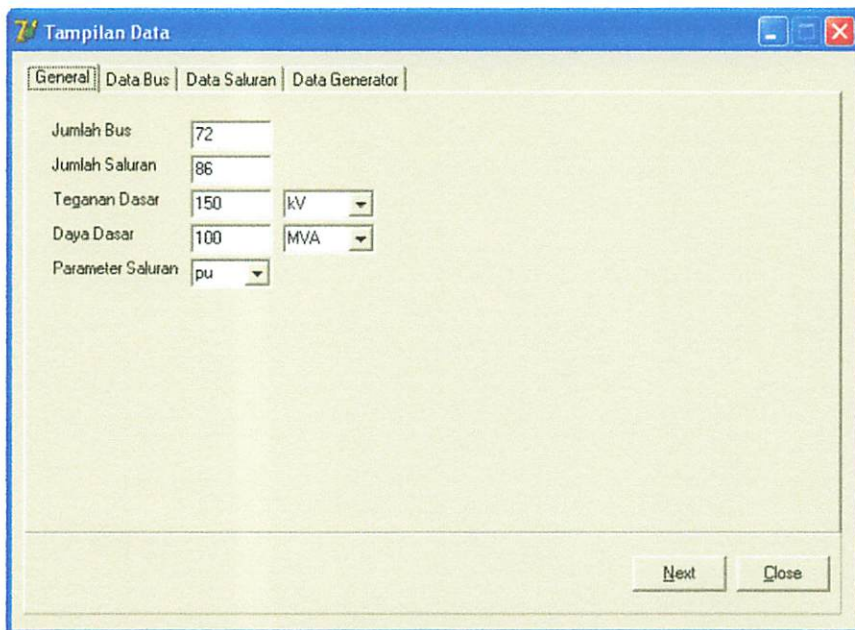
Prosedur menjalankan program perhitungan dilakukan setelah memasukkan seluruh data ke dalam program dengan menggunakan bahasa Pemrograman Borland Delphi versi 7.0. Mengenai prosedur jalannya program dapat dilakukan sebagai berikut:

1. Tampilan utama program.



Gambar 4.8  
Tampilan Utama Program

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



Gambar 4.9  
Tampilan Masukan Data.



3. Kemudian tekan tombol Data Bus

Bus	absv' (pu)	sudv' (deg)	Pg (MW)	Qg (MVA/R)	PL (MW)	QL (MVA/R)	Cap (pu)	Type Bus
1	1.04	0	0	0	0	0	0	1
2	1.03	0	563.1	0	17.8	8.8	0	2
3	1	0	0	0	27	12.1	0	3
4	1	0	0	0	23.4	10.8	0	3
5	1	0	0	0	20.3	1.3	0	3
6	1	0	0	0	34.5	5.4	0	3
7	1	0	0	0	2.2	1	0	3
8	1	0	0	0	35.6	14.3	0	3
9	1	0	0	0	17.5	8.7	0	3
10	1	0	0	0	41.4	15.6	0	3
11	1	0	0	0	17	3.7	0	3
12	1	0	0	0	25	7.3	0	3
13	1	0	0	0	50.5	22.8	0	3
14	1	0	0	0	35.5	3.7	0	3

Gambar 4.10  
Tampilan Data Bus

4. Kemudian tekan data saluran.

No	Dari	Ke	R (pu)	X (pu)	Lc (pu)	Tr	Tu	Su (deg)	Kap (MVA)
1	1	27	0.0013	0.0062	0.0025	0	0	0	10000
2	1	45	0.0032	0.01545	0.0226	0	0	0	10000
3	1	55	0.001	0.0035	0.0025	0	0	0	10000
4	1	59	0.0008	0.0024	0.0017	0	0	0	10000
5	2	5	0.0477	0.0346	0.0502	0	0	0	10000
6	2	7	0.0053	0.018	0.013	0	0	0	10000
7	3	4	0.0028	0.0094	0.0067	0	0	0	10000
8	3	8	0.0088	0.0301	0.0217	0	0	0	10000
9	3	15	0.0099	0.0072	0.0104	0	0	0	10000
10	3	16	0.0044	0.015	0.0108	0	0	0	10000
11	3	38	0.0093	0.0067	0.0097	0	0	0	10000
12	5	6	0.0097	0.0464	0.068	0	0	0	10000
13	5	11	0.0071	0.021	0.0133	0	0	0	10000
14	6	12	0.0024	0.01369	0.00245	0	0	0	10000

Gambar 4.11  
Tampilan Data Saluran

5. Tekan tombol data generator

No	Bus	Qmin (MVAR)	Qmax (MVAR)	a2	a1	a0	Fix Cost	Var Cost	Pmin
1	1	-600.00	600.00	6.18000	1306.15000	388144.168	0.73	5.10	150.
2	2	-600.00	600.00	1.40690	1599.00000	87435.0000	1.03	7.20	150.
3	21	-200.00	200.00	14.29320	1670.00000	88960.0000	1.03	7.20	35.0
4	24	-60.00	60.00	113.90000	5297.00000	139560.000	1.03	7.20	15.0
5	26	-500.00	500.00	113.90000	5297.00000	139560.000	1.03	7.20	150.
6	33	-500.00	500.00	113.90000	5297.00000	139560.000	1.03	7.20	150.
7	38	-70.00	70.00	113.90000	5297.00000	139560.000	1.03	7.20	100.
8	40	-85.00	85.00	113.90000	5297.00000	139560.000	1.03	7.20	25.0

Gambar 4.12  
Tampilan Data Generator

6. Tekan tombol *Load flow* awal untuk melihat hasil perhitungan pada kondisi awal (sebelum optimasi).

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	1.04000	0.00000	206.119	347.606	0.000	0.000	0.000	1
2	1.03000	0.29484	563.100	10.040	17.800	8.800	0.000	2
3	1.00136	0.07580	0.000	0.000	27.000	12.100	0.000	3
4	0.99975	0.07389	0.000	0.000	23.400	10.800	0.000	3
5	0.94114	0.21325	0.000	0.000	20.300	1.300	0.000	3
6	0.93600	0.17047	0.000	0.000	34.500	5.400	0.000	3
7	1.01048	0.23775	0.000	0.000	2.200	1.000	0.000	3
8	0.99031	0.15385	0.000	0.000	35.600	14.300	0.000	3
9	1.00600	0.23269	0.000	0.000	17.500	8.700	0.000	3
10	0.95798	0.14358	0.000	0.000	41.400	15.600	0.000	3
11	0.93768	0.19107	0.000	0.000	17.000	3.700	0.000	3
12	0.94267	0.14551	0.000	0.000	25.000	7.300	0.000	3
13	0.93612	0.16066	0.000	0.000	50.500	22.800	0.000	3
14	0.93103	0.16538	0.000	0.000	35.500	3.700	0.000	3
15	0.99359	0.05330	0.000	0.000	12.000	2.700	0.000	3

Gambar 4.13  
Tampilan hasil *Load Flow* pada Kondisi Awal (Sebelum Optimasi)



7. Tekan tombol Aliran Daya untuk melihat kondisi aliran daya pada kondisi awal (sebelum optimasi).

No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	27	-29.073	120.894	-186.367	774.960	27	1	29.260	-1
2	1	45	2.270	176.982	14.551	1134.499	45	1	-1.317	-1
3	1	55	77.287	6.110	495.429	39.165	55	1	-77.231	.
4	1	59	155.636	43.621	997.665	279.621	59	1	-155.442	.
5	2	5	208.879	-19.436	1257.069	-513.240	5	2	-189.172	;
6	2	7	336.421	20.677	2122.411	-504.699	7	2	-330.743	.
7	3	4	23.418	9.520	160.261	51.390	4	3	-23.400	.
8	3	8	-224.062	110.118	-1431.918	843.983	8	3	229.575	;
9	3	15	160.607	-110.361	1010.546	-813.603	15	3	-156.880	1.
10	3	16	199.577	-1.587	1324.092	-111.158	16	3	-197.829	.
11	3	38	-186.540	-19.789	-1248.321	-37.324	38	3	189.800	;
12	5	6	80.225	-10.640	539.459	-193.936	6	5	-79.518	.
13	5	11	88.647	-14.618	591.803	-234.104	11	5	-88.003	.
14	6	13	5.922	-3.626	37.188	-32.610	13	6	-5.908	.

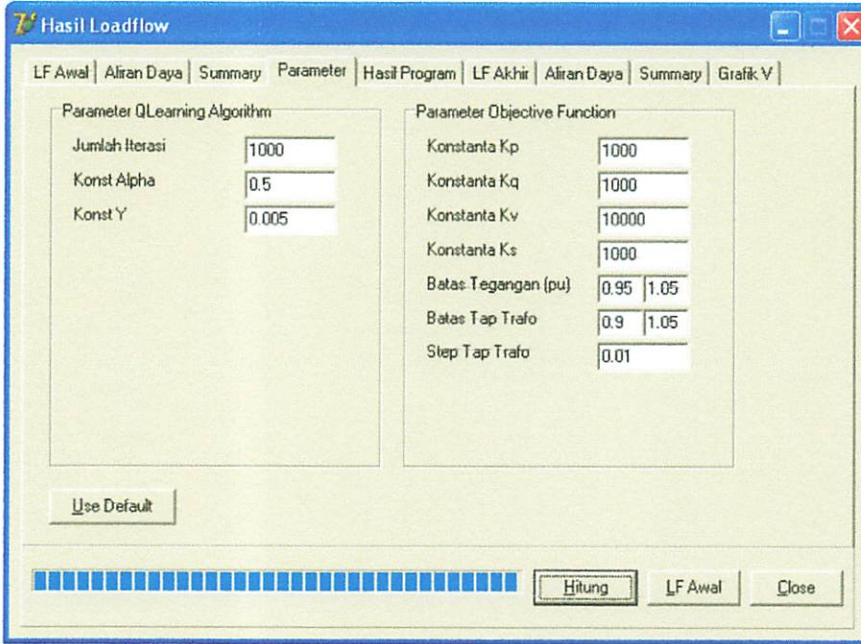
Gambar 4.14  
Tampilan Hasil Aliran Daya Pada Kondisi Awal (Sebelum Optimasi)

8. Tekan tombol *Summary* untuk melihat *summary load flow* (sebelum optimasi)

Summary Loadflow		
Jumlah Pembangkitan	1485.119+ j 325.977	MVA
Jumlah Pembebanan	1417.500+ j 445.900	MVA
Jumlah Rugi-Rugi	67.619- j 119.923	MVA
Iterasi	3	
Waktu Hitung	0.0.0.93	

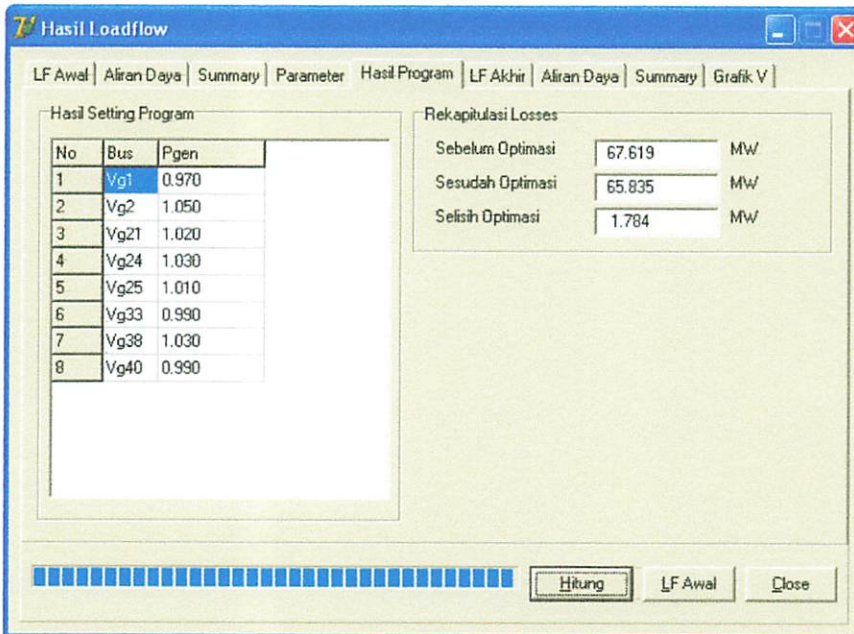
Gambar 4.15  
Tampilan *Summary Load Flow* Sebelum Optimasi

9. Tekan tombol next, kemudian tekan tombol *Use Default* untuk memasukkan parameter IEP atau berikan parameter yang sudah ditentukan.



Gambar 4.16  
Tampilan Parameter RL.

10. Tekan tombol Hasil Program untuk melihat Rekapitulasi *Losses*.



Gambar 4.17  
Tampilan Hasil Program pada Kondisi Akhir (Setelah Optimasi)



11. Tekan tombol *Load Flow* akhir untuk melihat hasil perhitungan pada kondisi akhir (setelah optimasi).

Bus	absV (pu)	sudV (deg)	Pg (MW)	Qg (MVAR)	PL (MW)	QL (MVAR)	Sups (pu)	Type Bus
1	0.97000	0.00000	204.335	-69.111	0.000	0.000	0.000	1
2	1.05000	0.26510	563.100	30.033	17.800	8.800	0.000	2
3	1.00587	0.05609	0.000	0.000	27.000	12.100	0.000	3
4	1.00427	0.05419	0.000	0.000	23.400	10.800	0.000	3
5	0.96162	0.18777	0.000	0.000	20.300	1.300	0.000	3
6	0.95603	0.14672	0.000	0.000	34.500	5.400	0.000	3
7	1.02771	0.21099	0.000	0.000	2.200	1.000	0.000	3
8	1.00282	0.13103	0.000	0.000	35.600	14.300	0.000	3
9	1.02331	0.20609	0.000	0.000	17.500	8.700	0.000	3
10	0.97362	0.12094	0.000	0.000	41.400	15.600	0.000	3
11	0.95764	0.16652	0.000	0.000	17.000	3.700	0.000	3
12	0.96014	0.12276	0.000	0.000	25.000	7.300	0.000	3
13	0.95505	0.13731	0.000	0.000	50.500	22.800	0.000	3
14	0.95106	0.14194	0.000	0.000	35.500	3.700	0.000	3
15	0.99255	0.03692	0.000	0.000	12.000	2.700	0.000	3

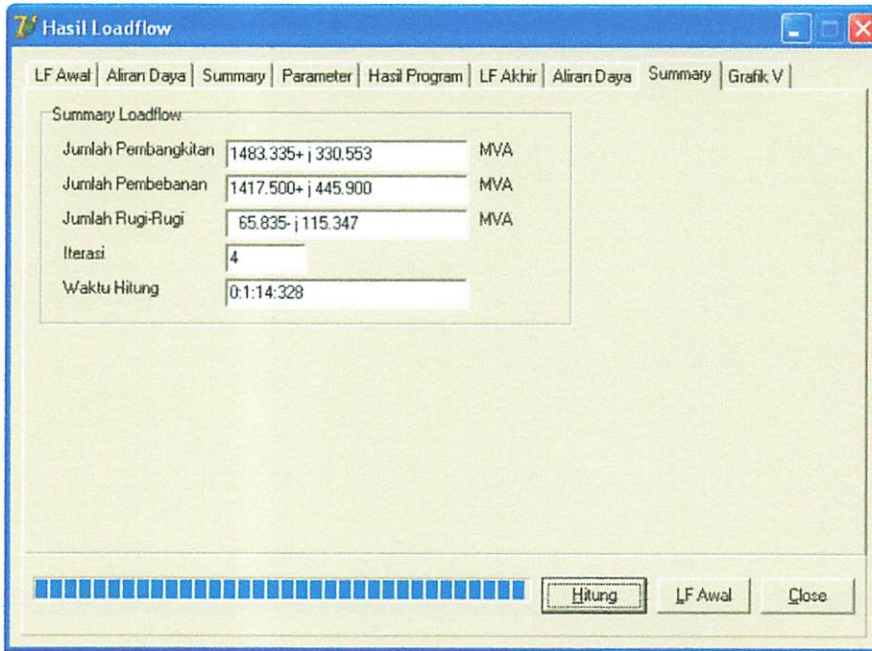
Gambar 4.18  
Tampilan Hasil *Load Flow* pada Kondisi Akhir (Setelah Optimasi)

12. Tekan tombol *Aliran Daya* untuk melihat hasil perhitungan pada kondisi akhir (setelah optimasi).

No	Dari	Ke	P (MW)	Q (MVAR)	Arus re (A)	Arus im (A)	Dari	Ke	P (MW)	Q
1	1	27	-25.832	-55.414	-177.541	-380.853	27	1	25.884	
2	1	45	0.793	-31.778	5.448	-218.402	45	1	-0.763	
3	1	55	77.425	7.998	532.129	54.972	55	1	-77.360	
4	1	59	151.950	10.082	1044.329	69.291	59	1	-151.753	
5	2	5	209.330	-17.210	1254.022	-453.672	5	2	-190.313	
6	2	7	335.970	38.442	2122.576	-323.336	7	2	-330.467	
7	3	4	23.418	9.507	158.497	54.210	4	3	-23.400	
8	3	8	-225.873	83.463	-1463.663	636.220	8	3	230.948	
9	3	15	181.664	-62.162	1179.033	-478.843	15	3	-170.069	
10	3	16	179.419	62.585	1210.528	347.484	16	3	-177.843	
11	3	38	-185.629	-105.492	-1267.559	-629.109	38	3	189.800	1
12	5	6	80.558	-9.875	535.890	-171.509	6	5	-79.876	
13	5	11	89.454	-12.253	593.404	-199.218	11	5	-88.831	
14	6	13	6.095	-2.988	39.002	-26.828	13	6	-6.082	

Gambar 4.19  
Tampilan *Aliran Daya* pada Kondisi Akhir (Setelah Optimasi)

13. Kemudian tekan tombol *summary* untuk melihat *summary load flow* setelah optimasi.



Gambar 4.20  
Tampilan *Summary Load Flow* Setelah Optimasi

#### 4.6. Hasil dan Analisis Hasil Perhitungan Optimasi Daya Reaktif Menggunakan Metode *Reinforcement Learning* Pada PLN P3B Region IV

##### 4.6.1. Hasil Perhitungan Sebelum Optimasi

Tabel 4.8  
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sebelum Optimasi

Bus No.	Tegangan		Pembangkit		Pembebanan	
	Magnitude (pu)	Sudut (deg)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1	1.04000	0.00000	206.119	347.606	0.000	0.000
2	1.03000	0.29484	563.100	10.040	17.800	8.800
3	1.00136	0.07580	0.000	0.000	27.000	12.100
4	0.99975	0.07389	0.000	0.000	23.400	10.800
5	0.94114	0.21325	0.000	0.000	20.300	1.300
6	0.93600	0.17047	0.000	0.000	34.500	5.400

7	1.01048	0.23775	0.000	0.000	2.200	1.000
8	0.99031	0.15385	0.000	0.000	35.600	14.300
9	1.00600	0.23269	0.000	0.000	17.500	8.700
10	0.95798	0.14358	0.000	0.000	41.400	15.600
11	0.93768	0.19107	0.000	0.000	17.000	3.700
12	0.94267	0.14551	0.000	0.000	25.000	7.300
13	0.93612	0.16066	0.000	0.000	50.500	22.800
14	0.93103	0.16538	0.000	0.000	35.500	3.700
15	0.99359	0.05330	0.000	0.000	12.000	2.700
16	0.99312	0.04567	0.000	0.000	15.300	5.400
17	0.99184	0.04507	0.000	0.000	20.600	4.500
18	0.99010	0.03344	0.000	0.000	20.200	3.000
19	0.99061	0.03199	0.000	0.000	20.600	6.900
20	0.99233	0.03039	0.000	0.000	39.900	0.500
21	1.00000	0.03580	30.000	19.121	0.000	0.000
22	0.99019	0.02798	0.000	0.000	20.800	9.700
23	0.99185	0.03282	0.000	0.000	30.000	12.000
24	1.00000	0.03482	19.000	9.703	20.800	9.500
25	1.00000	-0.02533	0.000	20.927	15.700	13.000
26	1.02222	0.01664	0.000	0.000	-500.400	-300.000
27	1.03315	0.00314	0.000	0.000	7.000	4.000
28	1.02025	0.01369	0.000	0.000	35.900	10.900
29	1.01754	0.01106	0.000	0.000	35.800	15.600
30	1.01125	0.02092	0.000	0.000	16.400	10.000
31	1.00293	-0.00695	0.000	0.000	12.000	6.000
32	1.00127	-0.00845	0.000	0.000	37.000	27.500
33	1.02000	0.05276	410.300	-28.589	0.000	0.000
34	1.03499	-0.01173	0.000	0.000	26.600	9.700
35	1.02544	-0.04749	0.000	0.000	12.500	0.800
36	1.02384	-0.05113	0.000	0.000	12.200	0.700
37	1.02341	-0.05185	0.000	0.000	18.900	0.800
38	1.02000	0.08632	200.400	22.356	10.600	2.200
39	1.01457	1.01457	0.000	0.000	58.900	20.100
40	1.01000	0.02692	56.200	-75.189	0.000	0.000
41	1.01317	0.01528	0.000	0.000	38.300	25.000
42	1.01485	0.01581	0.000	0.000	33.500	18.700
43	1.01463	0.01560	0.000	0.000	12.700	6.400
44	1.01429	0.02349	0.000	0.000	21.200	10.600
45	1.01329	0.00512	0.000	0.000	87.000	44.000
46	1.01471	0.01555	0.000	0.000	28.900	11.700
47	1.00911	-0.00376	0.000	0.000	100.300	32.100
48	1.00798	-0.00598	0.000	0.000	67.100	32.700
49	1.00699	-0.00583	0.000	0.000	48.300	24.100
50	1.005483	-0.00712	0.000	0.000	22.500	13.700
51	1.00783	-0.00632	0.000	0.000	14.400	7.400
52	1.00762	-0.00657	0.000	0.000	32.000	15.600
53	1.00784	-0.00640	0.000	0.000	32.800	14.800
54	1.01171	0.01507	0.000	0.000	36.900	19.100



55	1.03905	-0.00244	0.000	0.000	6.200	2.700
56	0.98175	0.03472	0.000	0.000	73.700	42.500
57	1.03300	0.00236	0.000	0.000	17.200	7.800
58	1.03119	0.00058	0.000	0.000	31.700	17.500
59	1.03780	-0.00314	0.000	0.000	25.500	18.600
60	1.02866	-0.02807	0.000	0.000	20.600	10.800
61	1.02216	-0.03484	0.000	0.000	27.000	14.300
62	1.02817	-0.04688	0.000	0.000	31.200	17.100
63	1.02973	-0.05559	0.000	0.000	41.600	-41.700
64	1.02912	-0.05634	0.000	0.000	11.300	5.000
65	1.01396	0.00045	0.000	0.000	46.800	4.500
66	1.01056	-0.00495	0.000	0.000	50.700	23.800
67	1.00853	-0.00711	0.000	0.000	13.600	7.400
68	0.99840	-0.02559	0.000	0.000	0.000	0.000
69	0.99826	-0.02555	0.000	0.000	39.800	23.400
70	1.00173	-0.01968	0.000	0.000	30.400	15.600
71	1.02954	-0.05574	0.000	0.000	11.300	7.000
72	1.013218	0.00495	0.000	0.000	16.000	2.700

Tabel 4.9  
 Hasil Perhitungan Aliran Daya Antar Saluran Sebelum Optimasi

No.	Saluran		Daya	
	Dari	Ke	Aktif (MW)	Reaktif (MVAR)
1	1	27	29.073	120.894
2	1	45	2.270	176.982
3	1	55	77.287	6.110
4	1	57	155.636	43.621
5	2	5	208.879	-19.436
6	2	7	336.421	20.677
7	3	4	23.418	9.520
8	3	8	224.062	110.118
9	3	15	160.607	-110.361
10	3	16	199.577	1010.546
11	3	38	186.540	-19.789
12	5	6	80.225	-10.640
13	5	11	88.647	-14.618
14	6	13	5.922	-3.626
15	6	14	39.096	-3.816
16	7	8	311.006	-4.466
17	7	9	17.537	7.564
18	8	10	38.347	55.200
19	10	12	4.108	23.545
20	10	13	7.833	20.065
21	11	13	71.003	-17.877
22	12	13	21.004	17.722
23	13	14	3.368	4.116
24	15	16	86.409	-106.286



25	15	23	58.472	-7.416
26	16	17	20.624	3.414
27	16	18	24.087	-1.740
28	16	19	21.215	-3.139
29	16	45	128.079	-158.174
30	16	56	74.082	43.178
31	18	19	3.802	-3.281
32	19	20	4.332	-10.309
33	20	21	28.104	-24.030
34	20	22	20.824	7.843
35	20	23	28.295	8.620
36	21	24	1.081	-1.681
37	25	68	4.979	8.460
38	25	69	20.679	-0.533
39	26	27	100.783	-121.076
40	26	28	35.938	10.672
41	26	29	35.879	15.163
42	26	30	16.619	9.310
43	26	31	49.199	15.163
44	26	39	14.745	62.675
45	26	42	17.122	71.114
46	26	46	61.084	165.190
47	26	65	119.436	32.192
48	26	66	79.085	22.723
49	27	57	63.963	-3.689
50	31	32	37.021	26.194
51	33	39	410.300	-28.589
52	34	35	44.197	-4.341
53	34	55	-70.797	-5.359
54	35	36	13.766	-1.690
55	35	37	17.399	-1.700
56	36	37	1.541	-1.262
57	39	40	-56.003	74.697
58	39	41	139.430	-5.969
59	39	42	249.818	-63.334
60	41	45	100.903	-31.148
61	42	43	12.701	-14.011
62	42	44	21.204	-3.090
63	42	45	162.213	-10.466
64	42	54	37.000	-18.466
65	45	46	157.512	11.238
66	45	47	318.283	78.463
67	45	72	16.001	2.539
68	47	48	217.355	44.121
69	48	49	70.892	36.638
70	48	51	46.414	-12.362
71	48	53	32.804	-13.225
72	49	50	22.517	12.676
73	51	52	32.005	-4.356
74	57	58	31.726	16.500
75	57	59	15.026	-27.078
76	59	60	144.899	1.454
77	60	61	27.090	9.743

78	60	62	95.961	-19.793
79	62	63	64.384	-32.489
80	63	64	11.303	3.782
81	63	71	11.301	6.727
82	65	66	72.181	26.645
83	66	67	13.614	4.567
84	66	70	86.442	21.463
85	68	69	0.000	-0.000
86	69	70	34.827	-9.950

Tabel 4.10  
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Sebelum Optimasi

Pembangkitan		Pembebanan		Rugi-rugi	
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1485.119	325.977	1417.500	445.900	67.619	-119.923

#### 4.6.2. Hasil Perhitungan Setelah Optimasi

Tabel 4.11  
Hasil Perhitungan Tegangan, Sudut Tegangan, Pembangkitan dan Pembebanan Sesudah Optimasi

Bus No.	Tegangan		Pembangkit		Pembebanan	
	Magnitude (pu)	Sudut (deg)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)
1	0.97000	0.00000	204.335	-69.111	0.000	0.000
2	1.05000	0.26510	563.100	30.033	17.800	8.800
3	1.00587	0.05609	0.000	0.000	27.000	12.100
4	1.00427	0.05419	0.000	0.000	23.400	10.800
5	0.96162	0.18777	0.000	0.000	20.300	1.300
6	0.95603	0.14672	0.000	0.000	34.500	5.400
7	1.02771	0.21099	0.000	0.000	2.200	1.000
8	1.00282	0.13103	0.000	0.000	35.600	14.300
9	1.02331	0.20609	0.000	0.000	17.500	8.700
10	0.97362	0.12094	0.000	0.000	41.400	15.600
11	0.95764	0.16652	0.000	0.000	17.000	3.700
12	0.96014	0.12276	0.000	0.000	25.000	7.300
13	0.95505	0.13731	0.000	0.000	50.500	22.800
14	0.95106	0.14194	0.000	0.000	35.500	3.700
15	0.99255	0.03692	0.000	0.000	12.000	2.700
16	0.98882	0.03184	0.000	0.000	15.300	5.400
17	0.98754	0.03124	0.000	0.000	20.600	4.500
18	0.99063	0.03999	0.000	0.000	20.200	3.000
19	0.99416	0.01575	0.000	0.000	20.600	6.900

20	1.00107	0.01263	0.000	0.000	39.900	0.500
21	1.02000	0.01452	30.000	41.884	0.000	0.000
22	0.99895	0.01026	0.000	0.000	20.800	9.700
23	0.99873	0.01531	0.000	0.000	30.000	12.000
24	1.03000	0.00844	19.000	33.768	20.800	9.500
25	1.01000	-0.04603	0.000	120.122	15.700	13.000
26	0.98702	0.01050	0.000	0.000	-500.400	-300.000
27	0.97387	0.00094	0.000	0.000	7.000	4.000
28	0.98497	0.00734	0.000	0.000	35.900	10.900
29	0.98215	0.00452	0.000	0.000	35.800	15.600
30	0.97564	0.01511	0.000	0.000	16.400	10.000
31	0.96689	-0.01482	0.000	0.000	12.000	6.000
32	0.96517	-0.01644	0.000	0.000	37.000	27.500
33	0.99000	0.04762	410.300	-8.135	0.000	0.000
34	0.96432	-0.01341	0.000	0.000	26.600	9.700
35	0.95313	-0.05443	0.000	0.000	12.500	0.800
36	0.95133	-0.05860	0.000	0.000	12.200	0.700
37	0.95086	-0.05944	0.000	0.000	18.900	0.800
38	1.03000	0.05871	200.400	108.687	10.600	2.200
39	0.98280	0.01416	0.000	0.000	58.900	20.100
40	0.99000	0.01697	56.200	73.306	0.000	0.000
41	0.97931	0.00864	0.000	0.000	38.300	25.000
42	0.98121	0.00918	0.000	0.000	33.500	18.700
43	0.98120	0.00905	0.000	0.000	12.700	6.400
44	0.98099	0.00895	0.000	0.000	21.200	10.600
45	0.97470	-0.00113	0.000	0.000	87.000	44.000
46	0.98240	0.00875	0.000	0.000	28.900	11.700
47	0.97019	-0.01070	0.000	0.000	100.300	32.100
48	0.96896	-0.01309	0.000	0.000	67.100	32.700
49	0.96793	-0.01292	0.000	0.000	48.300	24.100
50	0.96635	-0.01432	0.000	0.000	22.500	13.700
51	0.96879	-0.01344	0.000	0.000	14.400	7.400
52	0.96857	-0.01372	0.000	0.000	32.000	15.600
53	0.96880	-0.01353	0.000	0.000	32.800	14.800
54	0.97797	0.00839	0.000	0.000	36.900	19.100
55	0.96891	-0.00280	0.000	0.000	6.200	2.700
56	0.97740	0.02080	0.000	0.000	73.700	42.500
57	0.97321	0.00011	0.000	0.000	17.200	7.800
58	0.97128	-0.00189	0.000	0.000	31.700	17.500
59	0.96850	-0.00379	0.000	0.000	25.500	18.600
60	0.95815	-0.03233	0.000	0.000	20.600	10.800
61	0.95106	-0.04010	0.000	0.000	27.000	14.300
62	0.95740	-0.05398	0.000	0.000	31.200	17.100
63	0.95902	-0.06400	0.000	0.000	41.600	-41.700
64	0.95836	-0.06487	0.000	0.000	11.300	5.000
65	0.98669	-0.00853	0.000	0.000	46.800	4.500
66	0.98777	-0.01519	0.000	0.000	50.700	23.800
67	0.98569	-0.01745	0.000	0.000	13.600	7.400

68	1.00196	-0.04419	0.000	0.000	0.000	0.000
69	1.00181	-0.04415	0.000	0.000	39.800	23.400
70	0.99713	-0.03587	0.000	0.000	30.400	15.600
71	0.95882	-0.06418	0.000	0.000	11.300	7.000
72	0.97462	-0.00131	0.000	0.000	16.000	2.700

Tabel 4.12  
Hasil Perhitungan Aliran Daya Antar Saluran Setelah Optimasi

No.	Saluran		Daya	
	Dari	Ke	Aktif (MW)	Reaktif (MVAR)
1	1	27	25.832	-55.414
2	1	45	0.793	-31.778
3	1	55	77.425	7.998
4	1	57	151.950	10.082
5	2	5	209.330	-17.210
6	2	7	335.970	38.442
7	3	4	23.418	9.507
8	3	8	225.873	83.463
9	3	15	181.664	-62.162
10	3	16	179.419	62.585
11	3	38	185.629	-105.492
12	5	6	80.558	-9.875
13	5	11	89.454	-12.253
14	6	13	6.095	-2.988
15	6	14	39.281	-3.046
16	7	8	310.732	14.044
17	7	9	17.535	7.516
18	8	10	36.943	49.567
19	10	12	3.449	21.010
20	10	13	8.466	17.538
21	11	13	71.831	-15.348
22	12	13	21.638	15.338
23	13	14	3.561	3.175
24	15	16	107.179	-35.081
25	15	23	33.760	-20.187
26	16	17	20.624	3.424
27	16	18	24.144	-11.393
28	16	19	21.345	-15.714
29	16	45	128.944	-0.123
30	16	56	74.085	43.195
31	18	19	3.841	-12.999
32	19	20	4.447	-32.784
33	20	21	27.674	-69.770
34	20	22	20.823	7.808
35	20	23	28.669	31.752
36	21	24	1.933	-25.568



37	25	68	5.293	53.674
38	25	69	20.993	53.448
39	26	27	101.283	92.065
40	26	28	35.941	10.705
41	26	29	35.885	15.231
42	26	30	16.636	9.363
43	26	31	49.215	32.418
44	26	39	16.110	35.377
45	26	42	19.167	52.999
46	26	46	59.071	93.085
47	26	65	119.879	-22.126
48	26	66	79.434	-19.117
49	27	57	67.930	33.843
50	31	32	37.022	26.301
51	33	39	410.300	-8.135
52	34	35	44.288	-2.960
53	34	55	-70.888	-6.740
54	35	36	13.770	-1.340
55	35	37	17.406	-1.346
56	36	37	1.541	-1.081
57	39	40	55.998	-73.721
58	39	41	141.493	42.279
59	39	42	245.846	29.287
60	41	45	103.370	16.905
61	42	43	12.701	-12.681
62	42	44	21.204	-2.196
63	42	45	160.306	61.754
64	42	54	37.007	18.500
65	45	46	156.677	-79.295
66	45	47	318.362	83.796
67	45	72	16.001	2.551
68	47	48	217.378	48.104
69	48	49	70.900	36.739
70	48	51	46.415	-9.671
71	48	53	32.805	-11.094
72	49	50	22.518	12.762
73	51	52	32.006	-2.837
74	57	58	31.729	16.633
75	57	59	18.983	10.183
76	59	60	145.190	4.711
77	60	61	27.105	10.428
78	60	62	96.045	-18.158
79	62	63	64.412	-31.944
80	63	64	11.304	3.947
81	63	71	13.301	6.764
82	65	66	72.607	-27.824
83	66	67	13.615	4.698
84	66	70	87.192	-75.253

85	68	69	0.000	-0.000
86	69	70	-36.654	34.903

Tabel 4.13  
Total Pembangkitan, Pembebanan dan Rugi-rugi Saluran Setelah Optimasi

Pembangkitan		Pembebanan		Rugi-rugi		Waktu komputasi
Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	Aktif (MW)	Reaktif (MVAR)	
1483.335	330.553	1417.500	445.900	65.835	-115.347	0:1:43:0

#### 4.7. Perbandingan Hasil Perhitungan Sebelum dan Setelah Optimasi Daya Reaktif Menggunakan Metode *Reinforcement Learning*.

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

##### 4.7.1. Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

Berikut ini adalah tabel perbandingan tingkat tegangan dan sudut tegangan pada tiap bus.

Tabel 4.14  
Perbandingan Tingkat Tegangan dan Sudut Tegangan Pada Tiap Bus

Bus No	Sebelum Optimasi		Sesudah Optimasi	
	Tegangan (pu)	Sudut Tegangan (deg)	Tegangan (pu)	Sudut Tegangan (deg)
1	1.04000	0.00000	0.97000	0.00000
2	1.03000	0.29484	1.05000	0.26510
3	1.00136	0.07580	1.00587	0.05609
4	0.99975	0.07389	1.00427	0.05419
5	0.94114	0.21325	0.96162	0.18777
6	0.93600	0.17047	0.95603	0.14672
7	1.01048	0.23775	1.02771	0.21099
8	0.99031	0.15385	1.00282	0.13103
9	1.00600	0.23269	1.02331	0.20609
10	0.95798	0.14358	0.97362	0.12094
11	0.93768	0.19107	0.95764	0.16652
12	0.94267	0.14551	0.96014	0.12276

13	0.93612	0.16066	0.95505	0.13731
14	0.93103	0.16538	0.95106	0.14194
15	0.99359	0.05330	0.99255	0.03692
16	0.99312	0.04567	0.98882	0.03184
17	0.99184	0.04507	0.98754	0.03124
18	0.99010	0.03344	0.99063	0.03999
19	0.99061	0.03199	0.99416	0.01575
20	0.99233	0.03039	1.00107	0.01263
21	1.00000	0.03580	1.02000	0.01452
22	0.99019	0.02798	0.99895	0.01026
23	0.99185	0.03282	0.99873	0.01531
24	1.00000	0.03482	1.03000	0.00844
25	1.00000	-0.02533	1.01000	-0.04603
26	1.02222	0.01664	0.98702	0.01050
27	1.03315	0.00314	0.97387	0.00094
28	1.02025	0.01369	0.98497	0.00734
29	1.01754	0.01106	0.98215	0.00452
30	1.01125	0.02092	0.97564	0.01511
31	1.00293	-0.00695	0.96689	-0.01482
32	1.00127	-0.00845	0.96517	-0.01644
33	1.02000	0.05276	0.99000	0.04762
34	1.03499	-0.01173	0.96432	-0.01341
35	1.02544	-0.04749	0.95313	-0.05443
36	1.02384	-0.05113	0.95133	-0.05860
37	1.02341	-0.05185	0.95086	-0.05944
38	1.02000	0.08632	1.03000	0.05871
39	1.01457	1.01457	0.98280	0.01416
40	1.01000	0.02692	0.99000	0.01697
41	1.01317	0.01528	0.97931	0.00864
42	1.01485	0.01581	0.98121	0.00918
43	1.01463	0.01560	0.98120	0.00905
44	1.01429	0.02349	0.98099	0.00895
45	1.01329	0.00512	0.97470	-0.00113
46	1.01471	0.01555	0.98240	0.00875
47	1.00911	-0.00376	0.97019	-0.01070
48	1.00798	-0.00598	0.96896	-0.01309
49	1.00699	-0.00583	0.96793	-0.01292
50	1.005483	-0.00712	0.96635	-0.01432
51	1.00783	-0.00632	0.96879	-0.01344
52	1.00762	-0.00657	0.96857	-0.01372
53	1.00784	-0.00640	0.96880	-0.01353
54	1.01171	0.01507	0.97797	0.00839
55	1.03905	-0.00244	0.96891	-0.00280
56	0.98175	0.03472	0.97740	0.02080
57	1.03300	0.00236	0.97321	0.00011
58	1.03119	0.00058	0.97128	-0.00189
59	1.03780	-0.00314	0.96850	-0.00379
60	1.02866	-0.02807	0.95815	-0.03233

61	1.02216	-0.03484	0.95106	-0.04010
62	1.02817	-0.04688	0.95740	-0.05398
63	1.02973	-0.05559	0.95902	-0.06400
64	1.02912	-0.05634	0.95836	-0.06487
65	1.01396	0.00045	0.98669	-0.00853
66	1.01056	-0.00495	0.98777	-0.01519
67	1.00853	-0.00711	0.98569	-0.01745
68	0.99840	-0.02559	1.00196	-0.04419
69	0.99826	-0.02555	1.00181	-0.04415
70	1.00173	-0.01968	0.99713	-0.03587
71	1.02954	-0.05574	0.95882	-0.06418
72	1.013218	0.00495	0.97462	-0.00131

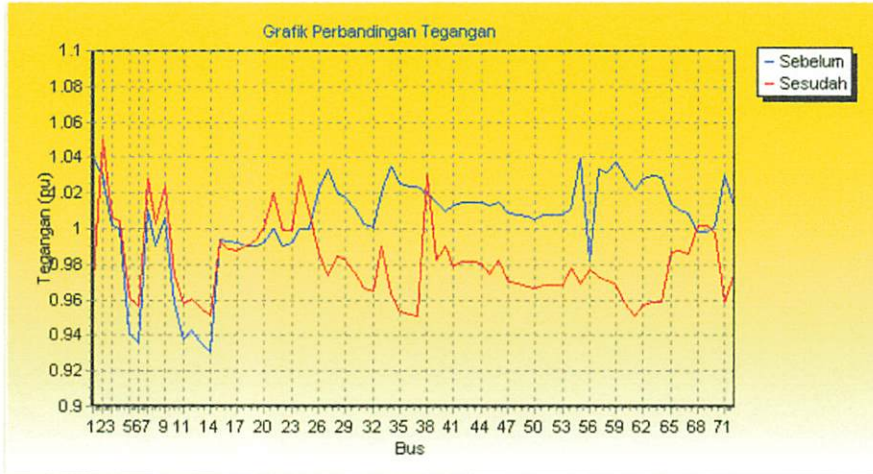
Dari tabel diatas dapat di lihat bahwa tegangan pada tiap bus masih dalam batas yang di perbolehkan yaitu antara 0.95 pu sampai 1.05 pu.

#### **4.8.2. Perbandingan Tingkat Rugi-rugi Daya Pada Saluran**

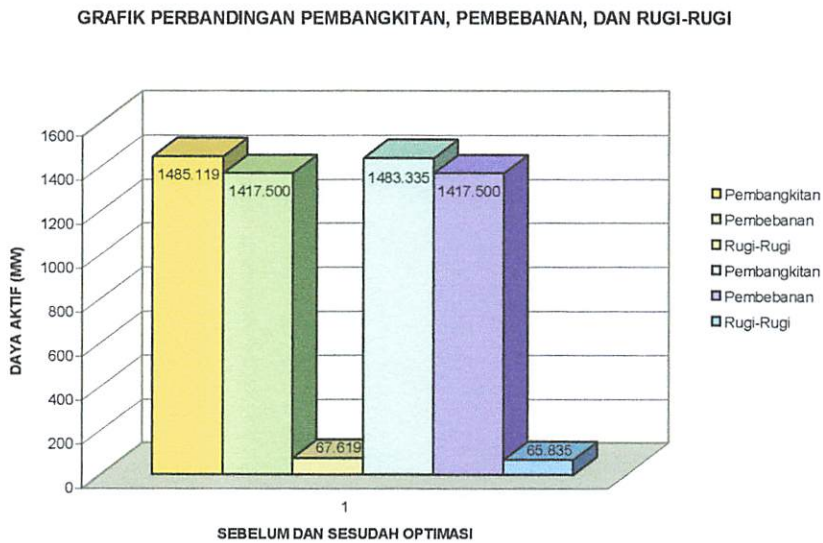
Untuk rugi-rugi daya pada saluran sebelum dioptimasi adalah 67.619 MW , sedangkan perhitungan rugi-rugi daya setelah optimasi adalah 65.835 MW dan. Dari hasil perhitungan dapat dilihat bahwa rugi-rugi daya yang ada sebelum dan sesudah optimasi terjadi penurunan yaitu sebesar 1.784 MW .



Grafik 4.1  
Perbandingan Tegangan Sebelum Optimasi dan Setelah Optimasi



Grafik 4.2  
Perbandingan Total Pembangkitan, Pembebanan dan Rugi-Rugi Untuk Daya Aktif Sebelum dan Sesudah Optimasi



## BAB V

### KESIMPULAN DAN SARAN

#### 5.1. Kesimpulan

Setelah dilakukan analisis perhitungan Optimasi Daya Reaktif dengan menggunakan metode *Reinforcement Learning* (RL) maka dapat diambil kesimpulan sebagai berikut:

1. Untuk rugi-rugi daya nyata pada saluran sebelum dioptimasi adalah 67.619 MW , sedangkan perhitungan rugi-rugi daya nyata setelah optimasi adalah 65.835 MW dan. Dari hasil perhitungan dapat dilihat bahwa rugi-rugi daya nyata yang ada sebelum dan sesudah optimasi terjadi penurunan yaitu sebesar 1.784 MW .
2. Setelah dilakukan pengoptimasian pengontrolan daya reaktif dengan menggunakan metode *Reinforcement Learning*, maka dapat diperoleh rugi-rugi daya nyata yang semakin kecil dan sesuai dengan kendala operasi selama seluruh periode perencanaan.
3. Dari hasil perhitungan *load flow* setelah optimasi daya reaktif menggunakan metode *Reinforcement Learning*, mampu memperbaiki profil tegangan sehingga tidak meyimpang dari batas yang sudah ditetapkan yaitu 0.95 – 1.05 pu.

## 5.2 Saran

Penggunaan metode *Reinforcement Learning* untuk menyelesaikan permasalahan optimasi daya reaktif masih perlu dikembangkan lagi terhadap sistem jaringan yang lebih luas minimal mengganti area penelitian, serta membahas kapasitas saluran, biaya *start up* dan *shut down*, Serta masalah *Spinning Reserve* sehingga hasil yang diperoleh dari penelitian lebih lengkap dan akurat, karena menyertakan seluruh komponen yang berhubungan dengan masalah optimalisasi dalam penyaluran daya listrik.

## DAFTAR PUSTAKA

- [1] Data pembangkitan dan Pembebanan Sub-Sistem 150 KV Region IV Jawa Timur tanggal 30 Juni 2006 Pukul 19.00 WIB, PT. PLN(Persero) P3B Region Jawa Timur dan Bali, Waru.
- [2] John Peschon, "*Sensitivity On Power System*", IEEE Trans. On Power App. And System, Vol, Pas – 87, No. 8 August 1968.
- [3] J. G. Vlachogiannis and N. D. Hatziargyriou, "*Reinforcement learning for reactive power control,*" *IEEE Trans. Power Syst., vol.19, no. 3, pp. 1317 – 1325, Aug. 2004.*
- [4] Made, Wartana, I, "*Transmisi Daya Arus Bolak – Balik*", Diktat Kuliah ITN Malang.
- [5] Marsudi Djiteng, Ir, "*Operasi system Tenaga Listrik*", Balai Penerbit dan Humas ISTN, 1990.
- [6] William D. Stevenson, "*Analisis Sistem Tenaga Listrik*", Edisi keempat penerbit Erlangga, 1996.

**unit uMenu;**

interface

uses

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

type

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

var

```
frmMenu: TfrmMenu;
```

implementation

uses uInputLFChild,uComplex, uUtils, uLoadflow, uHasil;

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

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

```
procedure TfrmMenu.btnNewClick(Sender: TObject);
begin
  try
    if frmInput=nil then
      begin
        frmInput:=TfrmInputLFChild.Create(Application);
      end;
    frmInput.Caption:='Input Data';
    frmInput.btnNext.Caption:='&Save';
    frmInput.ShowModal;
```

```
finally
  frmInput.Free;
end;
end;
```

```
procedure TfrmMenu.btnOpenClick(Sender: TObject);
var NamaFile,Nama:string;
    output:TextFile;
    i,j,Typ,dari,ke,Nbus,Nsal,Param,Ngen,NCable:integer;
    Cap,absV,sudV,Pg,Qg,PL,QL,CapSal,Pmin,Pmax,Harga,Length:double;
    R,X,Lc,Tr,Tu,Su,VKonst,PKonst,Pbase,Vbase,dia:double;
begin
  try
    if OpenFileDialog1.Execute then
      begin
        NamaFile:=OpenDialog1.FileName;
        AssignFile(output,NamaFile);
        Reset(output);
        Readln(output,Nbus);
        Readln(output,Nsal);
        Readln(output,Vbase);
        Readln(output,VKonst);
        Readln(output,Pbase);
        Readln(output,PKonst);
        Readln(output,param);
        gParamLF.Vbase:=Vbase;
        gParamLF.VKonst:=VKonst;
        gParamLF.Pbase:=Pbase;
        gParamLF.PKonst:=PKonst;
        if Param=1 then
          begin
            gParamLF.ParamBranch:=pbPu;
          end
        else if Param=2 then
          begin
            gParamLF.ParamBranch:=pbOhm;
          end;
        gParamLF.MaxIterasi:=15;
        gParamLF.Toleransi:=0.0001;
        try
          frmInput:=TfrmInputLFChild.Create(Application);
          frmInput.edtNbus.Text:=IntToStr(Nbus);
          frmInput.edtNsal.Text:=IntToStr(Nsal);
          frmInput.edtVbase.Text:=FloatToStr(Vbase);
          if VKonst=1 then
            begin
              frmInput.cmbVKonst.Text:='V';
            end
          else if VKonst=1000 then
            begin
```

```

    frmInput.cmbVKonst.Text:='kV';
end
else if VKonst=1000000 then
begin
    frmInput.cmbVKonst.Text:='MV';
end;
frmInput.edtPbase.Text:=FloatToStr(Pbase);
if PKonst=1 then
begin
    frmInput.cmbPKonst.Text:='VA';
    frmInput.fgBus.Cells[3,0]:='Pg (W)';
    frmInput.fgBus.Cells[4,0]:='Qg (VAR)';
    frmInput.fgBus.Cells[5,0]:='PL (W)';
    frmInput.fgBus.Cells[6,0]:='QL (VAR)';
    frmInput.fgBranch.Cells[9,0] :='Kap (VA)';
end
else if PKonst=1000 then
begin
    frmInput.cmbPKonst.Text:='kVA';
    frmInput.fgBus.Cells[3,0]:='Pg (kW)';
    frmInput.fgBus.Cells[4,0]:='Qg (kVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (kW)';
    frmInput.fgBus.Cells[6,0]:='QL (kVAR)';
    frmInput.fgBranch.Cells[9,0] :='Kap (kVA)';
end
else if PKonst=1000000 then
begin
    frmInput.cmbPKonst.Text:='MVA';
    frmInput.fgBus.Cells[3,0]:='Pg (MW)';
    frmInput.fgBus.Cells[4,0]:='Qg (MVAR)';
    frmInput.fgBus.Cells[5,0]:='PL (MW)';
    frmInput.fgBus.Cells[6,0]:='QL (MVAR)';
    frmInput.fgBranch.Cells[9,0] :='Kap (MVA)';
end;
if param=1 then
begin
    frmInput.cmbParam.Text:='pu';
    frmInput.fgBranch.Cells[3,0]:='R (pu)';
    frmInput.fgBranch.Cells[4,0]:='X (pu)';
    frmInput.fgBranch.Cells[5,0]:='Lc (pu)';
    frmInput.fgBus.Cells[7,0]:='Cap (pu)';
end
else if param=2 then
begin
    frmInput.cmbParam.Text:='ohm';
    frmInput.fgBranch.Cells[3,0]:='R (ohm)';
    frmInput.fgBranch.Cells[4,0]:='X (ohm)';
    frmInput.fgBranch.Cells[5,0]:='Lc (ohm)';
    frmInput.fgBus.Cells[7,0]:='Cap (ohm)';
end;

```

```

SetLength(gBus,Nbus);
for i:=0 to Nbus-1 do
begin
  Readln(output,absV,sudV,Pg,Qg,PL,QL,Cap,Typ);
  gBus[i].absV:=absV;
  gBus[i].sudV:=sudV;
  gBus[i].Pgen:=Pg;
  gBus[i].Qgen:=Qg;
  gBus[i].PL:=PL;
  gBus[i].QL:=QL;
  gBus[i].Cap:=Cap;
  gBus[i].typeBus:=Typ;
  frmInput.fgBus.Cells[0,i+1]:=IntToStr(i+1);
  frmInput.fgBus.Cells[1,i+1]:=FloatToStr(absV);
  frmInput.fgBus.Cells[2,i+1]:=FloatToStr(sudV);
  frmInput.fgBus.Cells[3,i+1]:=FloatToStr(Pg);
  frmInput.fgBus.Cells[4,i+1]:=FloatToStr(Qg);
  frmInput.fgBus.Cells[5,i+1]:=FloatToStr(PL);
  frmInput.fgBus.Cells[6,i+1]:=FloatToStr(QL);
  frmInput.fgBus.Cells[7,i+1]:=FloatToStr(Cap);
  frmInput.fgBus.Cells[8,i+1]:=IntToStr(typ);
end;
SetLength(gBranch,Nsal);
for i:=0 to Nsal-1 do
begin
  Readln(output,dari,ke,R,X,Lc,Tr,Tu,Su,CapSal);
  gBranch[i].dari:=dari;
  gBranch[i].ke:=ke;
  gBranch[i].R:=R;
  gBranch[i].X:=X;
  gBranch[i].Lc:=Lc;
  gBranch[i].Tr:=Tr;
  gBranch[i].Tu:=Tu;
  gBranch[i].Su:=Su;
  gBranch[i].KapSal:=CapSal;
  frmInput.fgBranch.Cells[0,i+1]:=IntToStr(i+1);
  frmInput.fgBranch.Cells[1,i+1]:=IntToStr(dari);
  frmInput.fgBranch.Cells[2,i+1]:=IntToStr(ke);
  frmInput.fgBranch.Cells[3,i+1]:=FloatToStr(R);
  frmInput.fgBranch.Cells[4,i+1]:=FloatToStr(X);
  frmInput.fgBranch.Cells[5,i+1]:=FloatToStr(Lc);
  frmInput.fgBranch.Cells[6,i+1]:=FloatToStr(Tr);
  frmInput.fgBranch.Cells[7,i+1]:=FloatToStr(Tu);
  frmInput.fgBranch.Cells[8,i+1]:=FloatToStr(Su);
  frmInput.fgBranch.Cells[9,i+1]:=FloatToStr(CapSal);
end;
Readln(output,Ngen);
if Ngen < 0 then
begin
  frmInput.fgGen.RowCount:=Ngen+1;

```



```

SetLength(gGenLF,Ngen);
for i:=0 to Ngen-1 do
begin
  Readln(output,dari,R,X,Lc,Tr,Tu,Su,CapSal,Pmin,Pmax);
  gGenLF[i].bus:=dari;
  gGenLF[i].Qmin:=R;
  gGenLF[i].Qmax:=X;
  gGenLF[i].a2:=Lc;
  gGenLF[i].a1:=Tr;
  gGenLF[i].a0:=Tu;
  gGenLF[i].FixCost:=Su;
  gGenLF[i].VarCost:=CapSal;
  gGenLF[i].Pmin:=Pmin;
  gGenLF[i].Pmax:=Pmax;
  frmInput.fgGen.Cells[0,i+1]:=IntToStr(i+1);
  frmInput.fgGen.Cells[1,i+1]:=IntToStr(gGenLF[i].bus);
  frmInput.fgGen.Cells[2,i+1]:=RealToStr(gGenLF[i].Qmin,2);
  frmInput.fgGen.Cells[3,i+1]:=RealToStr(gGenLF[i].Qmax,2);
  frmInput.fgGen.Cells[4,i+1]:=RealToStr(gGenLF[i].a2,5);
  frmInput.fgGen.Cells[5,i+1]:=RealToStr(gGenLF[i].a1,5);
  frmInput.fgGen.Cells[6,i+1]:=RealToStr(gGenLF[i].a0,5);
  frmInput.fgGen.Cells[7,i+1]:=RealToStr(gGenLF[i].FixCost,2);
  frmInput.fgGen.Cells[8,i+1]:=RealToStr(gGenLF[i].VarCost,2);
  frmInput.fgGen.Cells[9,i+1]:=RealToStr(gGenLF[i].Pmin,2);
  frmInput.fgGen.Cells[10,i+1]:=RealToStr(gGenLF[i].Pmax,2);
end;
end
else
begin
  frmInput.fgGen.RowCount:=2;
end;
CloseFile(output);
frmInput.Caption:='Tampilan Data';
frmInput.btnNext.Caption:='&Next';
frmInput.ShowModal;
finally
  frmInput.Free;
end;
end;
except
  MessageDlg('File Corrupt atau Error Program!',mtWarning,[mbOK],0);
end;
end;
end.

```

**unit uAbout;**

**interface**

**uses**

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

**type**

TfrmAbout = class(TForm)

private

{ Private declarations }

public

{ Public declarations }

end;

**var**

frmAbout: TfrmAbout;

**implementation**

{ \$R \*.dfm }

**end.**

**unit uHasil;**

interface

uses

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

type

```
TfrmHasil = class(TForm)
  PageControl1: TPageControl;
  TabSheet3: TTabSheet;
  TabSheet4: TTabSheet;
  TabSheet9: TTabSheet;
  Panel1: TPanel;
  btnClose: TButton;
  btnLFAwal: TButton;
  TabSheet1: TTabSheet;
  GroupBox6: TGroupBox;
  Label8: TLabel;
  Label9: TLabel;
  Label10: TLabel;
  Label11: TLabel;
  Label12: TLabel;
  lblGen: TLabel;
  lblLoad: TLabel;
  lblLoss: TLabel;
  edtSumGen: TEdit;
  edtSumLoad: TEdit;
  edtSumLoss: TEdit;
  edtIterasi: TEdit;
  edtTime: TEdit;
  TabSheet2: TTabSheet;
  btnHitung: TButton;
  TabSheet5: TTabSheet;
  TabSheet6: TTabSheet;
  TabSheet7: TTabSheet;
  TabSheet8: TTabSheet;
  Chart1: TChart;
  Series1: TLineSeries;
  Series2: TLineSeries;
  fgBus2: TStringGrid;
  fgBranch: TStringGrid;
  fgBranch2: TStringGrid;
  GroupBox1: TGroupBox;
  Label1: TLabel;
  Label2: TLabel;
  Label3: TLabel;
  Label4: TLabel;
```

```
Label5: TLabel;  
lblGen2: TLabel;  
lblLoad2: TLabel;  
lblLoss2: TLabel;  
edtSumGen2: TEdit;  
edtSumLoad2: TEdit;  
edtSumLoss2: TEdit;  
edtIterasi2: TEdit;  
edtTime2: TEdit;  
fgBus: TStringGrid;  
GroupBox3: TGroupBox;  
fgHasil: TStringGrid;  
pbGen: TProgressBar;  
GroupBox4: TGroupBox;  
btnUseDefault: TButton;  
Label21: TLabel;  
Label22: TLabel;  
Label23: TLabel;  
edtKp: TEdit;  
edtKq: TEdit;  
edtKv: TEdit;  
Label24: TLabel;  
edtVmin: TEdit;  
edtVmax: TEdit;  
GroupBox7: TGroupBox;  
Label26: TLabel;  
Label27: TLabel;  
Label28: TLabel;  
lblLoss21: TLabel;  
lblLoss22: TLabel;  
lblLoss23: TLabel;  
edtBaseLoss: TEdit;  
edtOptLoss: TEdit;  
edtSelisihLoss: TEdit;  
Label25: TLabel;  
edtTapTrafoMin: TEdit;  
edtTapTrafoMax: TEdit;  
Label29: TLabel;  
edtStepTrafo: TEdit;  
Label30: TLabel;  
edtKs: TEdit;  
GroupBox2: TGroupBox;  
Label17: TLabel;  
Label15: TLabel;  
Label16: TLabel;  
Label19: TLabel;  
edtMaxIterasi: TEdit;  
edtAlpha: TEdit;  
edtY: TEdit;  
procedure btnCloseClick(Sender: TObject);
```

```

procedure btnLFAwalClick(Sender: TObject);
procedure btnHitungClick(Sender: TObject);
procedure FormCreate(Sender: TObject);
procedure btnUseDefaultClick(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;

var
  frmHasil: TfrmHasil;

implementation

uses uUtils, uLoadflow, uNewtonRaphson, uComplex,
  uSensitivitas,uObjFunc,uPSO, uQLearning;

{$R *.dfm}

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

procedure TfrmHasil.btnLFAwalClick(Sender: TObject);
var i,Nbus,Nsal,ia,ja:integer;
    V,Sg,SL:CxArr1;
    Cap:dArr1;
    Lc,Tr:dArr2;
    TypBus:iArr1;
    Z,Tp,Alir,Arus:CxArr2;
    mulai,selesai,selang:TDateTime;
    jam,menit,detik,mdetik:word;
begin
  DecodeCommDataToLFData(gBus,Nbus,Nsal,V,Sg,SL,Cap,TypBus,
  gBranch,Z,Tp,Lc,Tr);
  mulai:=time;
  NewtonRaphson(gParamLF,V,Sg,SL,Cap,TypBus,Z,Tp,Alir,Arus,
  Lc,Tr);
  selesai:=time;
  selang:=selesai-mulai;
  Series1.Clear;
  Series2.Clear;
  for i:=0 to high(gBus) do
  begin
    fgBus.Cells[0,i+1]:=IntToStr(i+1);
    fgBus.Cells[1,i+1]:=RealToStr(V[i].real,5);
    fgBus.Cells[2,i+1]:=RealToStr(V[i].imag,5);
    fgBus.Cells[3,i+1]:=RealToStr(Sg[i].real,3);
  end;
end;

```

```

fgBus.Cells[4,i+1]:=RealToStr(Sg[i].imag,3);
fgBus.Cells[5,i+1]:=RealToStr(SL[i].real,3);
fgBus.Cells[6,i+1]:=RealToStr(SL[i].imag,3);
fgBus.Cells[7,i+1]:=RealToStr(Cap[i],3);
fgBus.Cells[8,i+1]:=IntToStr(TypBus[i]);
Series1.Add(V[i].real,IntToStr(i+1));
end;
for i:=0 to high(gBranch) do
begin
ia:=gBranch[i].dari-1;
ja:=gBranch[i].ke-1;
fgBranch.Cells[0,i+1]:=IntToStr(i+1);
fgBranch.Cells[1,i+1]:=IntToStr(gBranch[i].dari);
fgBranch.Cells[2,i+1]:=IntToStr(gBranch[i].ke);
fgBranch.Cells[3,i+1]:=RealToStr(Alir[ia,ja].real,3);
fgBranch.Cells[4,i+1]:=RealToStr(Alir[ia,ja].imag,3);
fgBranch.Cells[5,i+1]:=RealToStr(Arus[ia,ja].real,3);
fgBranch.Cells[6,i+1]:=RealToStr(Arus[ia,ja].imag,3);
fgBranch.Cells[7,i+1]:=IntToStr(gBranch[i].ke);
fgBranch.Cells[8,i+1]:=IntToStr(gBranch[i].dari);
fgBranch.Cells[9,i+1]:=RealToStr(Alir[ja,ia].real,3);
fgBranch.Cells[10,i+1]:=RealToStr(Alir[ja,ia].imag,3);
fgBranch.Cells[11,i+1]:=RealToStr(Arus[ja,ia].real,3);
fgBranch.Cells[12,i+1]:=RealToStr(Arus[ja,ia].imag,3);
end;
edtSumGen.Text:=toStringJ(gParamLF.SumGen,3);
edtSumLoad.Text:=toStringJ(gParamLF.SumLoad,3);
edtSumLoss.Text:=toStringJ(gParamLF.SumLoss,3);
edtBaseLoss.Text:=RealToStr(gParamLF.SumLoss.real,3);
edtIterasi.Text:=IntToStr(gParamLF.Iterasi);
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime.Text:=IntToStr(jam)+':'+IntToStr(menit)+':'+
IntToStr(detik)+':'+IntToStr(mdetik);
end;

procedure TfrmHasil.btnHitungClick(Sender: TObject);
var sa,i,Iterasi,c1,c2,PopSize,Length:integer;
    BatasV,BatasTrafo,w:TBatas;
    mulai,selesai,selang:TDateTime;
    jam,menit,detik,mdetik:word;
    BestChrom,Min,Avg,Max:dArr1;
    LBus:TBusArr1;
    LBranch:TBranchArr1;
    CostPLN,CostOpt,Kp,Kq,Kv,Ks,StepTrafo,tho:double;
    Pmutasi,Ka,Pr,Lamda,A,dV:double;
    pso:TPSO;
    BatasChrom:TBatasArr1;
    chromQLA:dArr1;
    qla:TQLearning;
begin

```

```

BatasV.min:=StrToFloat(edtVmin.Text);
BatasV.max:=StrToFloat(edtVmax.Text);
BatasTrafo.min:=StrToFloat(edtTapTrafoMin.Text);
BatasTrafo.max:=StrToFloat(edtTapTrafoMax.Text);
StepTrafo:=StrToFloat(edtStepTrafo.Text);
Kp:=StrToFloat(edtKp.Text);
Kq:=StrToFloat(edtKq.Text);
Kv:=StrToFloat(edtKv.Text);
Ks:=StrToFloat(edtKs.Text);
mulai:=time;
NewtonRaphson(gBus,gBranch,gParamLF);
gObjFunc:=TObjFunc.Create(Kp,Kq,Kv,Ks,StepTrafo,
    BatasV,BatasTrafo);
CostPLN:=gObjFunc.FindTotalCostGen(gBus);
Length:=gObjFunc.FindLength;
BatasChrom:=gObjFunc.FindBatasChrom;
Iterasi:=StrToInt(edtMaxIterasi.Text);
pbGen.Max:=Iterasi;
qla:=TQLearning.Create(Iterasi,0.5,0.005);
BestChrom:=qla.Bestchrom;
{ pbGen.Max:=Iterasi;
PopSize:=StrToInt(edtPopSize.Text);
c1:=StrToInt(edtC1.Text);
c2:=StrToInt(edtC2.Text);
dV:=StrToFloat(edtdV.Text);
w.min:=StrToFloat(edtwmin.Text);
w.max:=StrToFloat(edtwmax.Text);
pso:=TPSO.Create(Iterasi,PopSize,c1,c2,Length,dV,tho,w,BatasChrom);
BestChrom:=PSO.BestSwarm;
Min:=PSO.Global;
Series3.Clear;
for i:=0 to high(Min) do
begin
    Series3.Add(Min[i],IntToStr(i));
end;
pso.Free;    }
fgHasil.RowCount:=high(BestChrom)+2;
sa:=0;
for i:=0 to high(gBus) do
begin
    if gBus[i].typeBus<>3 then
    begin
        fgHasil.Cells[0,sa+1]:=IntToStr(sa+1);
        fgHasil.Cells[1,sa+1]:='Vg'+IntToStr(i+1);
        fgHasil.Cells[2,sa+1]:=FormatFloat('#,##0.000',BestChrom[sa]);
        inc(sa);
    end;
end;
for i:=0 to high(gBus) do
begin

```

```

if gBus[i].typeBus=3 then
begin
  if gBus[i].Qgen>0 then
  begin
    fgHasil.Cells[0,sa+1]:=IntToStr(sa+1);
    fgHasil.Cells[1,sa+1]:='Qg'+IntToStr(i+1);
    fgHasil.Cells[2,sa+1]:=FormatFloat('#,##0.000',BestChrom[sa]);
    inc(sa);
  end;
end;
end;
for i:=0 to high(gBranch) do
begin
  if gBranch[i].Tr<>0 then
  begin
    fgHasil.Cells[0,sa+1]:=IntToStr(sa+1);
    fgHasil.Cells[1,sa+1]:='Tr'+IntToStr(gBranch[i].dari)+
      IntToStr(gBranch[i].ke);
    fgHasil.Cells[2,sa+1]:=FormatFloat('#,##0.000',BestChrom[sa]);
    inc(sa);
  end;
end;
gObjFunc.doHitungAkhir(BestChrom,LBus,LBranch,CostOpt);
gObjFunc.Free;
selesai:=time;
selang:=selesai-mulai;
Series2.Clear;
for i:=0 to high(gBus) do
begin
  fgBus2.Cells[0,i+1]:=IntToStr(i+1);
  fgBus2.Cells[1,i+1]:=RealToStr(LBus[i].absV,5);
  fgBus2.Cells[2,i+1]:=RealToStr(LBus[i].sudV,5);
  fgBus2.Cells[3,i+1]:=RealToStr(LBus[i].Pgen,3);
  fgBus2.Cells[4,i+1]:=RealToStr(LBus[i].Qgen,3);
  fgBus2.Cells[5,i+1]:=RealToStr(LBus[i].PL,3);
  fgBus2.Cells[6,i+1]:=RealToStr(LBus[i].QL,3);
  fgBus2.Cells[7,i+1]:=RealToStr(LBus[i].Cap,3);
  fgBus2.Cells[8,i+1]:=IntToStr(LBus[i].typeBus);
  Series2.Add(LBus[i].absV,IntToStr(i+1));
end;
for i:=0 to high(LBranch) do
begin
  fgBranch2.Cells[0,i+1]:=IntToStr(i+1);
  fgBranch2.Cells[1,i+1]:=IntToStr(LBranch[i].dari);
  fgBranch2.Cells[2,i+1]:=IntToStr(LBranch[i].ke);
  fgBranch2.Cells[3,i+1]:=RealToStr(LBranch[i].Sij.real,3);
  fgBranch2.Cells[4,i+1]:=RealToStr(LBranch[i].Sij.imag,3);
  fgBranch2.Cells[5,i+1]:=RealToStr(LBranch[i].Aij.real,3);
  fgBranch2.Cells[6,i+1]:=RealToStr(LBranch[i].Aij.imag,3);
  fgBranch2.Cells[7,i+1]:=IntToStr(LBranch[i].ke);

```



```

fgBranch2.Cells[8,i+1]:=IntToStr(LBranch[i].dari);
fgBranch2.Cells[9,i+1]:=RealToStr(LBranch[i].Sji.real,3);
fgBranch2.Cells[10,i+1]:=RealToStr(LBranch[i].Sji.imag,3);
fgBranch2.Cells[11,i+1]:=RealToStr(LBranch[i].Aji.real,3);
fgBranch2.Cells[12,i+1]:=RealToStr(LBranch[i].Aji.imag,3);
end;
edtSumGen2.Text:=toStringJ(gParamLF.SumGen,3);
edtSumLoad2.Text:=toStringJ(gParamLF.SumLoad,3);
edtSumLoss2.Text:=toStringJ(gParamLF.SumLoss,3);
edtOptLoss.Text:=RealToStr(gParamLF.SumLoss.real,3);
edtSelisihLoss.Text:=RealToStr((strToFloat(edtBaseLoss.Text)-
StrToFloat(edtOptLoss.Text)),3);
edtIterasi2.Text:=IntToStr(4);
DecodeTime(selang,jam,menit,detik,mdetik);
edtTime2.Text:=IntToStr(jam)+':'+IntToStr(menit)+':'+
IntToStr(detik)+':'+IntToStr(mdetik);
//
end;

```

```

procedure TfrmHasil.FormCreate(Sender: TObject);
begin
fgHasil.Cells[0,0]:='No';
fgHasil.Cells[1,0]:='Bus';
fgHasil.Cells[2,0]:='Pgen';
end;

```

```

procedure TfrmHasil.btnUseDefaultClick(Sender: TObject);
begin
edtMaxIterasi.Text:='1000';
edtAlpha.Text:='0.5';
edtY.Text:='0.005';
edtKp.Text:='1000';
edtKv.Text:='10000';
edtKq.Text:='1000';
edtKs.Text:='1000';
edtVmin.Text:='0.95';
edtVmax.Text:='1.05';
edtTapTrafoMin.Text:='0.9';
edtTapTrafoMax.Text:='1.1';
edtStepTrafo.Text:='0.001';
btnHitung.Enabled:=true;
end;

```

```

end.

```

**unit uInputLFChild;**

interface

uses uInputLF, uHasil, SysUtils, Forms;

type

TfrmInputLFChild=class(TfrmInputLF)

protected

procedure ShowHasil;override;

end;

var frmInput:TfrmInputLFChild;

implementation

uses uLoadflow;

procedure TfrmInputLFChild.ShowHasil;

begin

try

if frmHasil=nil then

begin

frmHasil:=TfrmHasil.Create(Application);

end;

frmHasil.fgBus.Cells[0,0]:='Bus';

frmHasil.fgBus.Cells[1,0]:='absV (pu)';

frmHasil.fgBus.Cells[2,0]:='sudV (deg)';

frmHasil.fgBus.Cells[7,0]:='Sups (pu)';

frmHasil.fgBus.Cells[8,0]:='Type Bus';

frmHasil.fgBranch.Cells[0,0]:='No';

frmHasil.fgBranch.Cells[1,0]:='Dari';

frmHasil.fgBranch.Cells[2,0]:='Ke';

frmHasil.fgBranch.Cells[5,0]:='Arus re (A)';

frmHasil.fgBranch.Cells[6,0]:='Arus im (A)';

frmHasil.fgBranch.Cells[7,0]:='Dari';

frmHasil.fgBranch.Cells[8,0]:='Ke';

frmHasil.fgBranch.Cells[11,0]:='Arus re (A)';

frmHasil.fgBranch.Cells[12,0]:='Arus im (A)';

frmHasil.fgBus2.Cells[0,0]:='Bus';

frmHasil.fgBus2.Cells[1,0]:='absV (pu)';

frmHasil.fgBus2.Cells[2,0]:='sudV (deg)';

frmHasil.fgBus2.Cells[7,0]:='Sups (pu)';

frmHasil.fgBus2.Cells[8,0]:='Type Bus';

frmHasil.fgBranch2.Cells[0,0]:='No';

frmHasil.fgBranch2.Cells[1,0]:='Dari';

frmHasil.fgBranch2.Cells[2,0]:='Ke';

frmHasil.fgBranch2.Cells[5,0]:='Arus re (A)';

frmHasil.fgBranch2.Cells[6,0]:='Arus im (A)';

frmHasil.fgBranch2.Cells[7,0]:='Dari';

```

frmHasil.fgBranch2.Cells[8,0]:='Ke';
frmHasil.fgBranch2.Cells[11,0]:='Arus re (A)';
frmHasil.fgBranch2.Cells[12,0]:='Arus im (A)';
if gParamLF.PKonst=1 then
begin
frmHasil.fgBus.Cells[3,0]:='Pg (W)';
frmHasil.fgBus.Cells[4,0]:='Qg (VAR)';
frmHasil.fgBus.Cells[5,0]:='PL (W)';
frmHasil.fgBus.Cells[6,0]:='QL (VAR)';
frmHasil.fgBranch.Cells[3,0]:='P (W)';
frmHasil.fgBranch.Cells[4,0]:='Q (VAR)';
frmHasil.fgBranch.Cells[9,0]:='P (W)';
frmHasil.fgBranch.Cells[10,0]:='Q (VAR)';
frmHasil.lblGen.Caption:='VA';
frmHasil.lblLoad.Caption:='VA';
frmHasil.lblLoss.Caption:='VA';
//
frmHasil.fgBus2.Cells[3,0]:='Pg (W)';
frmHasil.fgBus2.Cells[4,0]:='Qg (VAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (W)';
frmHasil.fgBus2.Cells[6,0]:='QL (VAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (W)';
frmHasil.fgBranch2.Cells[4,0]:='Q (VAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (W)';
frmHasil.fgBranch2.Cells[10,0]:='Q (VAR)';
frmHasil.lblGen2.Caption:='VA';
frmHasil.lblLoad2.Caption:='VA';
frmHasil.lblLoss2.Caption:='VA';
frmHasil.lblLoss21.Caption:='W';
frmHasil.lblLoss22.Caption:='W';
frmHasil.lblLoss23.Caption:='W';
//
end
else if gParamLF.PKonst=1000 then
begin
frmHasil.fgBus.Cells[3,0]:='Pg (kW)';
frmHasil.fgBus.Cells[4,0]:='Qg (kVAR)';
frmHasil.fgBus.Cells[5,0]:='PL (kW)';
frmHasil.fgBus.Cells[6,0]:='QL (kVAR)';
frmHasil.fgBranch.Cells[3,0]:='P (kW)';
frmHasil.fgBranch.Cells[4,0]:='Q (kVAR)';
frmHasil.fgBranch.Cells[9,0]:='P (kW)';
frmHasil.fgBranch.Cells[10,0]:='Q (kVAR)';
frmHasil.lblGen.Caption:='kVA';
frmHasil.lblLoad.Caption:='kVA';
frmHasil.lblLoss.Caption:='kVA';
//
frmHasil.fgBus2.Cells[3,0]:='Pg (kW)';
frmHasil.fgBus2.Cells[4,0]:='Qg (kVAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (kW)';

```

```

frmHasil.fgBus2.Cells[6,0]:='QL (kVAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (kW)';
frmHasil.fgBranch2.Cells[4,0]:='Q (kVAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (kW)';
frmHasil.fgBranch2.Cells[10,0]:='Q (kVAR)';
frmHasil.lblGen2.Caption:='kVA';
frmHasil.lblLoad2.Caption:='kVA';
frmHasil.lblLoss2.Caption:='kVA';
frmHasil.lblLoss21.Caption:='kW';
frmHasil.lblLoss22.Caption:='kW';
frmHasil.lblLoss23.Caption:='kW';
//
end
else if gParamLF.PKonst=1000000 then
begin
frmHasil.fgBus.Cells[3,0]:='Pg (MW)';
frmHasil.fgBus.Cells[4,0]:='Qg (MVAR)';
frmHasil.fgBus.Cells[5,0]:='PL (MW)';
frmHasil.fgBus.Cells[6,0]:='QL (MVAR)';
frmHasil.fgBranch.Cells[3,0]:='P (MW)';
frmHasil.fgBranch.Cells[4,0]:='Q (MVAR)';
frmHasil.fgBranch.Cells[9,0]:='P (MW)';
frmHasil.fgBranch.Cells[10,0]:='Q (MVAR)';
frmHasil.lblGen.Caption:='MVA';
frmHasil.lblLoad.Caption:='MVA';
frmHasil.lblLoss.Caption:='MVA';
//
frmHasil.fgBus2.Cells[3,0]:='Pg (MW)';
frmHasil.fgBus2.Cells[4,0]:='Qg (MVAR)';
frmHasil.fgBus2.Cells[5,0]:='PL (MW)';
frmHasil.fgBus2.Cells[6,0]:='QL (MVAR)';
frmHasil.fgBranch2.Cells[3,0]:='P (MW)';
frmHasil.fgBranch2.Cells[4,0]:='Q (MVAR)';
frmHasil.fgBranch2.Cells[9,0]:='P (MW)';
frmHasil.fgBranch2.Cells[10,0]:='Q (MVAR)';
frmHasil.lblGen2.Caption:='MVA';
frmHasil.lblLoad2.Caption:='MVA';
frmHasil.lblLoss2.Caption:='MVA';
frmHasil.lblLoss21.Caption:='MW';
frmHasil.lblLoss22.Caption:='MW';
frmHasil.lblLoss23.Caption:='MW';
//
end;
frmHasil.fgBus.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasil.fgBranch.RowCount:=StrToInt(edtNsal.Text)+1;
frmHasil.fgBus2.RowCount:=StrToInt(edtNbus.Text)+1;
frmHasil.fgBranch2.RowCount:=StrToInt(edtNsal.Text)+1;
frmHasil.ShowModal;
finally
frmHasil.Free;

```



## FORMULIR BIMBINGAN SKRIPSI

Nama : **TOMMY NUGROHO**  
Nim : **00.12.079**  
Masa Bimbingan : **04 Januari 2007 s/d 04 Juli 2007**  
Judul Skripsi : **OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE  
REINFORCEMENT LEARNING PADA PT. PLN PJB REGION IV**

No.	Tanggal	Uraian	Parap Pembimbing
1.	27-06-2007	Konsultasi bab I, II	
2.	03-07-2007	Konsultasi perbaikan bab I & II sekaligus pengajuan bab III	
3.	08-07-2007	Konsultasi perbaikan bab III & melakukan test uji validasi	
4.	11-07-2007	Cek kembali error hasil uji validasi dengan jurnal	
5.	15-07-2007	Konsultasi hasil error uji validasi dan pengajuan bab IV	
6.	18-07-2007	Konsultasi perbaikan bab IV dan pengajuan makalah seminar	
7.	23-07-2007	Periksa kembali format penulisan	
8.	14-08-2007	Acc makalah seminar hasil	
9.	15-08-2007	Konsultasi seluruh bab dan Acc Ujian Skripsi	

Malang, Agustus 2007  
Dosen Pembimbing,

**Ir. Yusuf Ismail Nakhoda, MT**  
Nip. Y. 101 8800 189



Malang, 12 Oktober 2006

Lampiran : 1 ( satu ) berkas  
**Pembimbing Skripsi**

Kepada : Yth. Bapak/Ibu. Jr Yusuf Ismail Nakhoda, MT

Dosen Jurusan Elektro/ T.Energi Listrik  
Institut Teknologi Nasional Malang  
di-  
Malang

Yang bertanda tangan di bawah ini :

Nama : Tommy Nugroho  
Nim : 00.12.079  
Semester : XIII (Tiga Belas )  
Jurusan : Teknik Elektro  
Konsentrasi : Teknik Energi Listrik S-I

Dengan ini mengajukan permohonan, kiranya Bapak/Ibu bersedia menjadi Dosen Pembimbing Utama, untuk penyusunan Skripsi dengan judul :

**OPTIMASI PENGONTROLAN DAYA REAKTIF MENGGUNAKAN  
METODE REINFORCEMENT LEARNING  
PADA PT. PLN PJB REGION IV**

Adapun Tugas tersebut sebagai salah satu syarat untuk menempuh Ujian Akhir Sarjana Teknik.  
Demikian permohonan kami, atas kesediaan Bapak / ibu kami ucapkan terimakasih.

Mengetahui,  
Ketua Jurusan  
Teknik Elektro S-I

Ir. F. Yudi Eimpraptono  
Nip. 1039500274

\*) coret yang tidak perlu

Malang, 12 - 10 - 2006

Hormat kami,

Tommy Nugroho  
Nim : 00.12.079

Form. S - 3a



INSTITUT TEKNOLOGI NASIONAL MALANG  
FAKULTAS TEKNOLOGI INDUSTRI  
JURUSAN TEKNIK ELEKTRO

## PERNYATAAN KESEDIAAN DALAM PEMBIMBINGAN SKRIPSI

Sesuai dengan Permohonan Mahasiswa/i :

Nama : Tommy Nugroho  
Nim : 00.12.079  
Semester : XIII (Tiga Belas )  
Jurusan : Teknik Elektro  
Konsentrasi : Teknik Energi Listrik S-I

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

### OPTIMASI PENGONTROLAN DAYA REAKTIF MENGGUNAKAN METODE *REINFORCEMENT LEARNING* PADA PT. PLN PJB REGION IV

Demikian surat pernyataan ini kami buat agar dapat dipergunakan  
seperlunya.

Malang, 12 Oktober 2006  
Yang Membuat Pernyataan,

Ir Yusuf Ismail Nakhoda, MT  
NIP. P. 101 8800 189

Catatan :

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

\*)Coret yang tidak perlu

Form. S - 3b



**FORMULIR PENGAJUAN JUDUL SKRIPSI**  
JURUSAN TEKNIK ELEKTRO/T. ENERGI LISTRIK S-1

1.	Nama Mahasiswa : Tommy Nugroho	Nim : 00.12.079		
2.	Waktu Pengajuan :	Tanggal <i>16</i>	Bulan <i>10</i>	Tahun <i>2006</i>
3.	Spesifikasi Judul *)			
	<input checked="" type="radio"/> a. Sistem Tenaga Elektrik b. Mesin-Mesin Elektrik & Elda c. Sistem Pemb.Energi Elektrik		d. Sistem Kendali e. Teknik Tegangan Tinggi f. Lainnya .....	
4.	Konsultasikan judul sesuai materi bidang ilmu kepada Kelompok Dosen Keahlian **)  <i>Z. Yusuf Ismail N, MT</i>		Ketua Jurusan,   <b>Ir. F. Yudi Limpraptono, MT</b> Nip. 103 950 0274	
5.	Judul yang diajukan mahasiswa	<b>OPTIMASI PENGONTROLAN DAYA REAKTIF MENGGUNAKAN METODE REINFORCEMENT LEARNING PADA PT. PLN (Persero) PEMBANGKITAN JAWA BALI (PJB)</b>		
6.	Perubahan Judul yang Diusulkan oleh Kelompok Dosen Keahlian			
7.	<b>Cacatan :</b>			
8.	Persetujuan Judul Skripsi yang dikonsultasikan kepada Kelompok Dosen Keahlian		Disetujui, Kelompok Dosen Keahlian   <b>Ir. Yusuf Ismail N, MT</b>	

**Perhatian :**

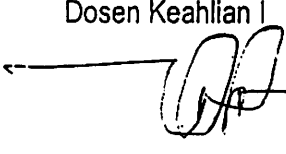
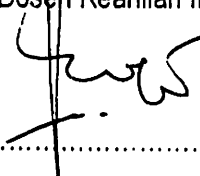
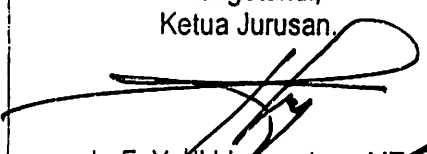
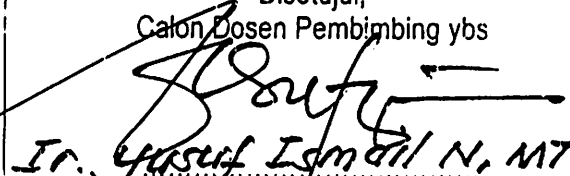
1. Formulir Pengajuan ini harap dikembalikan ke Jurusan paling lambat *satu minggu* setelah disetujui Kelompok Dosen Keahlian dengan dilampirkan Proposal Skripsi beserta persyaratan Skripsi sesuai **Form. S-1**.
2. \*) dilingkari a, b, c, ....atau f, sesuai bidang Keahlian.
3. \*\*) diisi oleh Jurusan.





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

Konsentrasi : Teknik Energi Listrik/Teknik Elektronika\*)

1.	Nama Mahasiswa: <b>TOMMY NUGROHO</b>		Nim: <b>00.12079</b>	
2.	Keterangan	Tanggal	Waktu	Tempat
	Pelaksanaan	<b>01-01-2007</b>		Ruang:
Spesifikasi Judul (berilah tanda silang)**)				
3.	a. Sistem Tenaga Elektrik		e. Elektronika & Komponen	
	b. Energi & Konversi Energi		f. Elektronika Digital & Komputer	
	c. Tegangan Tinggi & Pengukuran		g. Elektronika Komunikasi	
	d. Sistem Kendali Industri		h. lainnya .....	
4.	Judul Proposal yang diseminarkan Mahasiswa	<b>OPTIMASI PENGONTROLAN DAYA REAKTIF MENGGUNAKAN METODE REINFORCEMENT LEARNING PADA PT. PLN PJB REGION IV</b>		
5.	Perubahan Judul yang diusulkan oleh Kelompok Dosen Keahlian	<b>OPTIMASI DAYA REAKTIF MENGGUNAKAN METODE REINFORCEMENT LEARNING PADA PT. PLN PJB REGION IV</b>		
6.	Catatan: .....			
	.....			
7.	Persetujuan Judul Skripsi			
	Disetujui, Dosen Keahlian I		Disetujui, Dosen Keahlian II	
				
	Mengetahui, Ketua Jurusan.		Disetujui, Calon Dosen Pembimbing ybs	
				
<b>Ir. F. Yudi Limpraptono, MT</b> NIP. P. 1039500274		<b>Ir. Yusuf Ismail N, MT</b>		

Perhatian:

1. Keterangan: \*) Coret yang tidak perlu

\*\*) dilingkari a, b, c, ..... atau g sesuai bidang keahlian



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

## PERSETUJUAN PERBAIKAN SKRIPSI

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

Hari : Senin  
Tanggal : 3 September 2007

Telah dilakukan perbaikan skripsi oleh :

1. Nama : Tommy Nugroho
2. NIM : 00.12.079
3. Jurusan : Teknik Elektro S-1
4. Konsentrasi : Teknik Energi Listrik
5. Judul Skripsi : Optimasi Daya Reaktif Menggunakan Metode *Reinforcement Learning* Pada PLN P3B Region IV

Perbaikan meliputi :

No	Materi Perbaikan	Paraf
1.	Bab V, Kesimpulan Disesuaikan Dengan Tujuan	

Anggota Penguji I

Ir. Teguh Herbasuki, MT  
NIP. Y. 1038900209

Dosen Pembimbing

Ir. Yusuf Ismail Nakhoda, MT  
NIP. Y. 101 8800 189

# Reinforcement Learning for Reactive Power Control

John G. Vlachogiannis and Nikos D. Hatziargyriou, *Senior Member, IEEE*

**Abstract**—This paper presents a Reinforcement Learning (RL) method for network constrained setting of control variables. The RL method formulates the constrained load flow problem as a multistage decision problem. More specifically, the model-free learning algorithm (Q-learning) learns by experience how to adjust a closed-loop control rule mapping states (load flow solutions) to control actions (offline control settings) by means of reward values. Rewards are chosen to express how well control actions cause satisfaction of operating constraints. The Q-learning algorithm is applied to the IEEE 14 busbar and to the IEEE 136 busbar system for constrained reactive power control. The results are compared with those given by the probabilistic constrained load flow based on sensitivity analysis demonstrating the advantages and flexibility of the Q-learning algorithm. Computing times with another heuristic method is also compared.

**Index Terms**—Constrained load flow, Q-learning algorithm, reinforcement learning.

## I. INTRODUCTION

THE Constrained Load Flow (CLF) problem deals with the adjustment of the power system control variables in order to satisfy physical and operating constraints. A number of algorithms have been developed in order to solve this problem, which are based on modification of the Jacobian matrix formed in the standard load flow method using sensitivity or injection-changing-error feedback control [1], [2] or evolutionary computation techniques [3], [4]. The CLF problem is also expressed as a constrained optimization problem falling within the general class of Optimal Power Flow (OPF) problems [5]–[8].

In most published methods, one or more known sets of generation and loads are assumed, i.e., the input variables are assumed deterministically known. These methods [1]–[8], however, are inefficient in providing offline settings of control variables that must remain optimal for a whole planning period.

The problem of offline control settings has been tackled by the probabilistic CLF formulation [9], [10]. The method developed in [9] takes into account load uncertainties and generating unit unavailabilities modeled as probability density functions and provides control settings satisfying constraints over a whole planning period. Dispatching effects and topological variations are considered using probabilistic techniques. The control variable settings are based on sensitivity analysis of the constrained variables with respect to the control variables. Although this

method has shown to provide settings satisfying the required constraints, it provides suboptimal solutions dependent on the initial values of the control settings.

In this paper, the constrained load flow problem is solved by means of machine learning methods. Reinforcement Learning (RL) methods belong to this category [11]–[13]. They originate from optimal control theory and dynamic programming and aim at approximating by experience solutions to problems of unknown dynamics [14]. From a theoretical point of view, many breakthroughs have been realized concerning the convergence of the RL approach and their application to nonlinear systems [12], [13], leading to very efficient algorithms. Also the rapid increase in computer capacities makes RL methods feasible and attractive in the power system community [12], [13].

In order to apply the RL problem, the constrained load flow problem is formulated as a multistage decision problem. Optimal control settings are learnt by experience adjusting a closed-loop control rule, which is mapping states (load flow solutions) to control actions (offline control settings). The control settings are based on rewards, expressing how well actions work over the whole planning period, i.e., how well the operating limits of constrained variables are satisfied. In this paper, the model-free learning algorithm (Q-learning) [11] is applied to reactive power control; however, the algorithm is general and can be applied to a wide variety of constrained optimization problems in planning or operational planning.

The paper is organized into six sections. Section II formulates the constrained load flow problem and Section III describes the Reinforcement Learning approach. In Section IV, Q-learning algorithm is implemented to reactive power control. In Section V, the results obtained by the application of the Q-learning algorithm to the IEEE 14 bus system and to the IEEE 136 bus system are presented and discussed. The results are compared with those obtained by the probabilistic constrained load flow [9], showing the superiority of the proposed method. Computing times with another heuristic method (enhanced genetic algorithm) is also compared. In Section VI, the main advantages and disadvantages of the Q-learning algorithm on the optimal power flow method are highlighted. Finally, in Section VII, general conclusions are drawn.

## II. CONSTRAINED LOAD FLOW PROBLEM

The load flow problem can be expressed by the two sets of nonlinear equations

$$\begin{aligned} Y &= g(X, U) \\ Z &= h(X, U) \end{aligned} \quad (1)$$

Manuscript received February 13, 2004.  
J. G. Vlachogiannis is with the Informatics and Computer Technology Department, Technological Educational Institute of Lamia, Lamia 35100, Greece (e-mail: vlachogiannis@usa.com).

N. D. Hatziargyriou is with the Department of Electrical and Computer Engineering, National Technical University of Athens, Athens 10682, Greece (e-mail: nhr@power.ece.ntua.gr).

Digital Object Identifier 10.1109/TPWRS.2004.831259

here

nodal power injections vector;  
 constrained variables vector (power flows, reactive powers of PV buses, etc.);  
 state vector (voltage angles and magnitudes);  
 control vector (transformer tap position, shunt compensation, voltage and production at PV buses, etc.)

The objective of constrained load flow is to maintain some all the elements of  $X$  and  $Z$  vectors within given operating limits under load variations and generating unit unavailabilities. This can be achieved by selecting appropriate (robust) values of control variables under random variations of loads and generations (noise factors) within their operating range.

The next sections of this paper propose a technique to maintain constrained variables within operating limits over the whole planning period using Q-learning algorithm.

### III. REINFORCEMENT LEARNING

Reinforcement Learning (RL) techniques are simple iterative algorithms that learn to act in an optimal way through experience gained by exploring an unknown system [11]–[13]. RL assumes that the “world” can be described by a set of states and an “agent” can choose one action from a set of actions. The operating range is divided into discrete learning-steps. At each learning-step, the agent observes the current state  $s$  of the “world” ( $s \in S$ ), and chooses an action  $a \in A$  that tends to maximize an expected long-term value function [11]–[13]. After taking action ( $a$ ), the agent is given an immediate reward  $r \in \mathcal{R}$ , expressing the effectiveness of the action and observing the resulting state of the “world”  $s' \in S$ . The goal of RL is to take these experience tuples,  $(s, a, r, s')$  and learn a mapping from states (or states and actions, depending on the particular algorithm used) to a measure of the long-term value of taking this action, known as the optimal value function [11]–[13].

The particular RL algorithm used in this work is the Q-learning algorithm [11]. The Q-learning optimal value function is defined by means of the Bellman equation, as

$$Q^*(s, a) = E \left( r(s, a) + \gamma \max_{a'} Q^*(s', a') \right). \quad (2)$$

This represents the expected sum of rewards, when starting from an initial state ( $s$ ) taking action ( $a$ ), and performing optimal actions ( $a'$ ) in next searches, until the optimal value of  $Q$ -function, ( $Q^*(s, a)$ ) is reached.  $E$  represents the sum of mathematical expectation of above values. The discount parameter ( $0 \leq \gamma \leq 1$ ) is used to exponentially decrease the weight of rewards received in next searches [11]–[13].

Once we have the optimal value  $Q^*(s, a)$ , it is easy to determine the optimal action  $a^*$  using a greedy policy [11]–[13]. A simple way is to look at all possible actions ( $a$ ) from a given state ( $s$ ) and select the one with the largest value

$$a^* = \arg \max_a Q_a^*(s, a). \quad (3)$$

The Q-function (Q-memory) is typically stored in a table, indexed by state and action. Starting with arbitrary values, we can iteratively approximate the optimal Q-function based on our op-

timality criteria. The table entry for state ( $s$ ) and action ( $a$ ) is then updated according to [11]

$$Q(s, a) = (1 - \alpha) \cdot Q(s, a) + \alpha(r + \gamma \max_{a'} Q(s', a')). \quad (4)$$

It is important to note that the new value for  $Q(s, a)$  memory is based both on the current value of  $Q(s, a)$ , and the values (immediate rewards) of control actions obtained by next searches. So, the parameter  $\alpha$  ( $0 \leq \alpha \leq 1$ ) plays a critical role representing the amount of the updated Q-memory (4) and affects the number of iterations. The parameter  $(1 - \alpha)$  represents the total amount of Q-values still remaining as a memory in the Q-function [11]–[13].

Under some reasonable conditions, (4) is guaranteed to converge to the optimal  $Q^*$  values (the best Q-memories) for each pair of ( $s, a$ ) in a finite number of iterations. These assumptions concern the nonchaotic nature (i.e., each pair of ( $s, a$ ) has to be visited an infinite number of times) and the number of the control actions and/or solution states of the learning system (discretization method) [11]–[13]. One of the most important features of Q-learning is that it is what is known as an off-policy algorithm [12], [13]. This means that the probability distribution from which the training samples ( $s, a$ ) are derived has no effect, in the limit, on the Q-learning process (4). This will prove to be very important for our problem because there is no need to use any exploration policy for our Q-learning algorithm [12], [13].

### IV. Q-LEARNING APPLIED TO REACTIVE POWER CONTROL

For the purpose of our analysis, a two-class classification is assumed. The “world” of RL solution states ( $s \in S$ ) is binary, comprising acceptable operating points characterized by satisfaction of all constraints and unacceptable, when any constraint is violated. The control vectors that combine discrete values of control adjustments are the actions ( $a \in A$ ), and the Q-learning algorithm is the “agent.” The algorithm proceeds as follows: An operating point comprising a load and generation pattern including a set of control actions is randomly created. The agent observes the state ( $s$ ) of the system, as obtained by the load flow solution, and chooses one control action ( $a$ ) from the control vector. A new load flow is executed. The agent observes the resulting state of the solution ( $s'$ ) and provides an immediate reward ( $r$ ):  $S \times A \rightarrow \mathcal{R}$  expressing the degree of satisfaction of the operating limits of the constrained variables. A new control action is selected next, leading to a new load flow solution and a new reward. Selection of new control actions is repeated until no more changes in the reward value can be achieved. The goal of the agent is to learn the optimal Q-function ( $Q^*(s, a)$ ) using the mappings of states to actions ( $S \rightarrow A$ ) such that the long-term reward is maximized. The procedure is repeated for a large number of operating states covering the whole planning period. The agent finds the greedy-optimal control settings ( $a^*$ ) [11]–[13] using the policy described by (3). Table I shows the execution steps of the Q-learning as applied to reactive power control.

#### A. State Vectors

In order to confine the constrained variable  $z_j$  within its operating limits ( $z_{j \min} \leq z_j \leq z_{j \max}$ ), the states of the system are dis-

TABLE I  
Q-LEARNING ALGORITHM APPLIED TO REACTIVE POWER CONTROL

- 1) Initialize memory  $Q(s,a)$  and immediate rewards  $r(s,a)=0.0$ ,  $\forall s \in S, \forall a \in A$
- 2) Repeat for a given number of operating points over the whole planning period
  - 2.1) Repeat
    - 2.1.1) Observe the state ( $s$ ) of load flow solution
    - 2.1.2) Choose an action vector ( $a$ )
    - 2.1.3) Execute load flow
    - 2.1.4) Observe state ( $s'$ ) resulting from load flow and calculate the reward using (7)
    - 2.1.5) Update Q-function using (4)
    - 2.1.6) Set  $s \leftarrow s'$
 until optimal Q-function is achieved (no change in reward)
  - 2.2) Enforce each of the optimal control actions over the whole planning period and calculate (8)
- 3) Choose as a greedy-optimal action ( $a^*$ ) the one that maximizes (8)

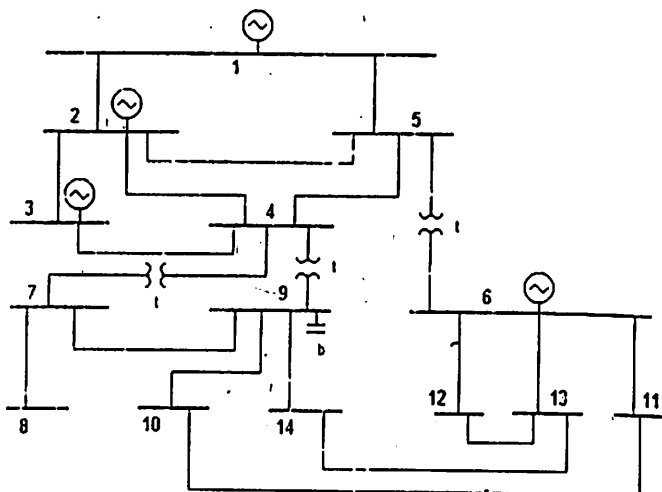


Fig. 1. One-line diagram of IEEE 14-busbar system.

cretized as follows: When one of the variables  $z_j$  lies outside its operating limits, the state is considered as  $-1$  level-state, otherwise it is considered at the zero level-state. Consequently, if we have  $n$ -constrained variables, the total number of states is

$$\bar{S} = 2^n \tag{5}$$

B. Action Vectors

If each control variable  $u_i$  is discretized in  $d_{u_i}$  levels, the total number of action-vectors effecting the load flow is

$$\bar{A} = \prod_{i=1}^m d_{u_i} \tag{6}$$

where  $m$  expresses the total number of control variables.

In case of continuous control variables, the finer is the interval discretization, the better is the performance of Q-learning, and larger the learning steps.

C. Rewards

Application of the Q-learning algorithm to reactive power control problem is linked to the choice of an immediate reward ( $r$ ), such that the iterative value of Q-function (4) is maximized, while the limits of the constrained variables are satisfied for the whole planning period. An enforced empirical strategy is to consider the variations of constrained variables close to the means of their operating intervals as a measure of reward. The immediate reward ( $r$ ) is computed by the average of all constrained variables, normalized in the interval:  $[0, 1]$ , as follows:

$$r = -\frac{1}{n} \cdot \sum_{j=1}^n \left| \frac{2z_j - z_{j \max} - z_{j \min}}{z_{j \max} - z_{j \min}} \right| \tag{7}$$

where  $n$  expresses the number of constrained variables,  $z_j$  the value of  $j$ -constrained variable and ( $z_{j \min}, z_{j \max}$ ) are its lower and upper limits, respectively. According to Table I, once we have the optimal actions (3) for a large number of operating points, the one that maximizes the sum of rewards over the whole planning period is defined as greedy-optimal action ( $a^*$ ) of Q-learning

$$\begin{aligned} \text{intr}(a^*) &= \max \{ \text{total of rewards} \} \\ &= \max_{\pi^*(s)} \left( \sum_{\text{over whole planning period}} r \right) \end{aligned} \tag{8}$$

where  $a^* = \arg \max(\pi^*(s))$  over the whole planning period.

D. Discount ( $\gamma$ ) and Learning Step ( $\alpha$ ) Parameters

In addition to the above definitions, two parameters  $\gamma$  and  $\alpha$  for implementing the Q-learning algorithm need to be chosen. RL theory gives generic guidelines for these parameters [11]–[13].

Parameter  $\gamma$ , introduced in (2), is the control factor by which later rewards are discounted and it must be between 0 and 1. In our application, later rewards are not important because there is no interdependence among load flow solutions produced by Q-learning steps: therefore, the value of  $\gamma$  is chosen close to 0. Our experience shows that any value in the range 0–0.01 works well. In the following, a value of 0.005 is used.

The critical parameter  $\alpha$  used in (4), expresses the amount of the updated Q-function, in other words the rate of learning [11]–[13]. A large enough parameter  $\alpha$  (close to 1) allows fast convergence of the Q-learning algorithm, while a small value (close to 0) avoids instability of Q-learning. Since the Q-learning enforced in constrained load flow problem does not depend on previous Q-learning steps as stated above, this parameter works well close to 1. In our application, a value of 0.995 is selected.

V. PERFORMANCE RESULTS

The Q-learning algorithm is applied to reactive control variable adjustment of the IEEE 14-bus test system shown in Fig. 1.

The IEEE 14-bus test system consists of the slack bus (node 1), three PV (nodes 2, 3, and 6), ten PQ buses, and 20 branches. It has been used in many probabilistic studies. The network data and load probabilistic data are the same data used in [9], [15] and are not repeated here. They comprise six discrete distributions for the active load (at nodes 3, 6, 9, 10, 11, and 14), four discrete distributions for the reactive load (at nodes 9, 10, 11, and 14), with 3 to 5 impulses each and 8 normal distributions for active and reactive loads at the remaining buses. The total installed capacity is equal to 4.9 pu and comprises 14 capacity blocks at node 1, 4 blocks at node 2, 2 blocks at node 3, and 2 blocks at node 6. The voltage at all PV buses is taken equal to 1.0 pu and the slack bus equal to 1.02 pu. A fixed network topology is assumed [9], [15].

TABLE II  
LIMITS AND DISCRETIZATION OF ACTIONS AND  
LIMITS OF CONSTRAINED VARIABLES

Control Actions	Umin	Umax	Step
t56	0.90	1.05	0.01
t49	0.90	1.05	0.01
t47	0.90	1.05	0.01
b9	0.00	0.27	0.03
Constrained Variables	Wmin	Wmax	
Qg2	0.00	0.30	
Qg3	0.00	0.70	
Qg6	0.00	0.45	
T23	0.00	0.75	
T56	0.00	0.50	
V4	0.96	1.05	
V5	0.96	1.05	
V7	0.96	1.05	
V8	0.96	1.05	
V9	0.96	1.05	
V10	0.96	1.05	
V11	0.96	1.05	
V12	0.96	1.05	
V13	0.96	1.05	
V14	0.96	1.05	

The control variables comprise all transformer taps (t) and reactive compensation (b) at bus 9 (Fig. 1). The upper part of Table II shows the limits of the control variables Umin and Umax and the discrete steps of variation. In the lower part of Table II the upper and lower limits of all constrained variables Wmax and Wmin are shown. These include reactive powers at generation buses (Qgi), apparent power flows (Tij) and voltages at load buses (Vi). Since the constrained variables are 15 [9], the total number of solution states by means of (5) is calculated  $2^{15} = 32768$ . The control variables are 3 transformer taps, discretized at 16 positions and 1 reactive compensator at 9 blocks [9], the total number action vectors, by means of (6) is calculated  $16 \times 16 \times 16 \times 9 = 36864$ .

The Q-learning algorithm (Table I) can be implemented in a large number of load combinations (operating points) selected over the whole planning period. In our study, the algorithm learns the optimum control settings for each of 41 operating points selected from the whole planning period. These operating points are obtained assuming uniform load variations at all buses in 40 equal percentage steps around their average values obtained by

$$\text{Load step} = \left( \mu \pm k \cdot \frac{3\sigma}{20} \right) \cdot 100\% \quad (9)$$

where  $k = 0, 1, 2, 3, \dots, 20$ . Factors  $\mu$  and  $\sigma$  are the average values and the standard deviations for normal distributions of loads given in [15]. The  $\mu$  and  $\sigma$  for the discrete distributions of loads are calculated using the formulae given by [16].

For each of the 41 operating points the actions stored in the action vector (36864) are simulated. Fig. 2(A)–(B)–(C) shows the obtained immediate reward (7) at every Q-learning step, corresponding to the average, upper, and lower values of load, respectively. Each Q-learning step corresponds to an iteration of the Q-learning algorithm. The average values correspond to the mean values of the load. The upper and lower load values correspond to the 1% confidence limit that all load values are lower and higher than these values, respectively.

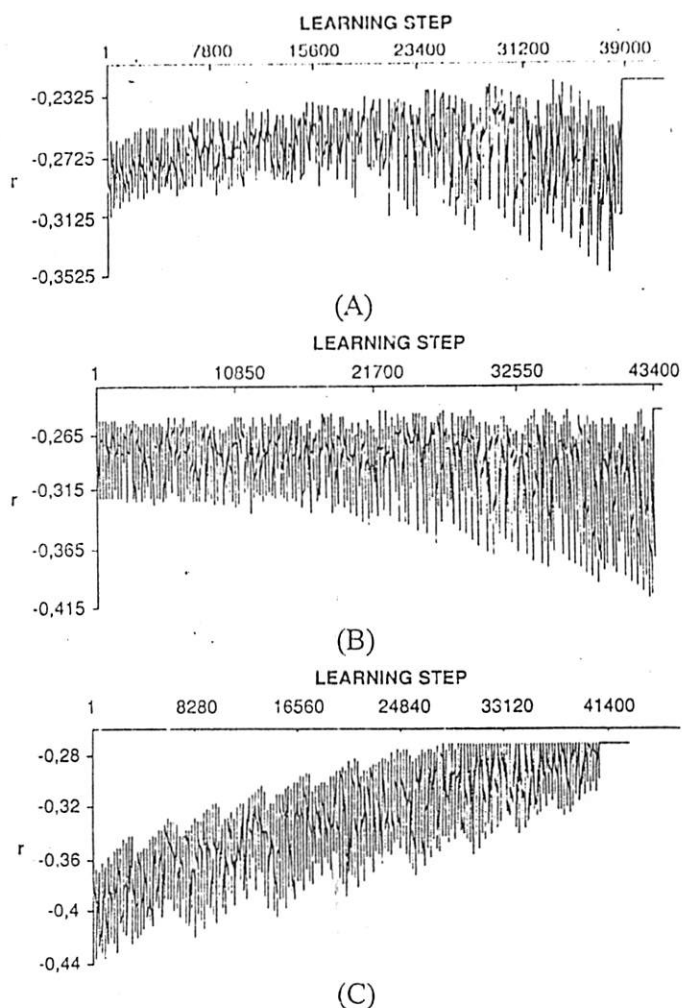


Fig. 2. Immediate rewards of Q-learning procedure in (A) average, (B) upper, and (C) lower values of loads.

These figures also depict the convergence of Q-learning algorithm in a maximum reward value, mapping the optimum control settings to the best solution state for each of the 3 operating points (average, lower, and upper load values). The agent made approximately 39 000 Q-learning steps in Fig. 2(A), 43 400 in Fig. 2(B), and 41 200 in Fig. 2(C) to find the optimum control actions. The total computing times were 65–70 s on a 1.4-GHz Pentium IV PC. It should be noted that the above steps were necessary because zero initial learning memory was assumed ( $Q(s,a) = 0.0$ ). The learning process, however, provides the Q-values for each of the 3 operating points that map each control action to each solution state. This Q-function table can be used as initial learning memory achieving optimum control actions in less Q-learning steps at the next runs. So, under this gained experience (knowledge), convergence of the Q-learning algorithm in the other 38 operating points is achieved within 400 to 1000 Q-learning steps only. This is further shown in the following applications. Among the 41 optimal control settings, the overall (greedy) optimal control settings are those that provide the maximum total reward calculated from (8). Tests have shown that the calculation of (8) in the average, upper, and lower operating points is sufficient to provide the maximum total reward.

TABLE III  
RESULTS OF Q-LEARNING AND PROBABILISTIC CLF ON IEEE 14-BUS

Q-Learning algorithm mtr = -0.804		Probabilistic CLF		
Actions	Greedy-Optimal Settings (a*)	Optimal Settings		
t56	1.03	0.94		
t49	0.97	0.97		
t47	0.90	0.98		
b9	0.12	0.12		
Constrained Variables	Wmin	Wmax	Wmin	Wmax
Qg2	-0.5250	-0.0527	0.2069	0.3160
Qg3	0.1751	0.6183	0.6420	0.6812
Qg6	0.0891	0.3787	0.3065	0.4161
T23	0.2284	0.6868	0.7223	0.7799
T56	0.1675	0.4528	0.4144	0.4931
V4	0.9742	0.9951	0.9654	0.9731
V5	0.9864	1.0030	0.9682	0.9734
V7	0.9501	1.0274	0.9710	0.9857
V8	0.9901	1.0274	0.9710	0.9833
V9	0.9840	1.0284	0.9656	0.9853
V10	0.9775	1.0193	0.9644	0.9803
V11	0.9831	1.0067	0.9828	0.9912
V12	0.9802	0.9915	0.9907	0.9936
V13	0.9725	0.9903	0.9832	0.9878
V14	0.9541	0.9977	0.9508	0.9663

Results on IEEE 14-bus show that the greedy-optimal action (a\*) for the whole planning period is the optimal action obtained for the average values of load [Fig. 2(A)]. Table III shows the maximum total reward calculated at  $-0.804$ , the greedy optimal-action (a\*) and the operating space of constrained variables when the greedy-optimal action (a\*) is enforced over the whole planning period. It can be seen that even when applying the greedy-optimal control settings, reactive production at node 2 (Qg2) violates its limits and the voltage at node 14 (V14) violates its lower limit. In Table III, these results are also compared to the results of the probabilistic constrained load flow (CLF) obtained for the same network [9]. It can be seen that the Q-learning algorithm provides improved results, since the upper limit of the apparent power flow (T23) in line 2-3 is not violated. However, the violations at Qg2 and V14 are due to the increased reactive demands of IEEE 14-bus system and the available control actions cannot enforce the violated limits.

One way of enforcing violated limits is to relax the constant voltage limitation at node 2, considering it as PQ and allowing the voltages at nodes 6 and 1 to be set at 1.021 pu and 1.03 pu, respectively [9]. Rerunning the Q-learning algorithm without initial learning memory ( $Q(s, a) = 0.0$ ), under these new considerations it provides the immediate rewards (7) shown in Fig. 3(A)-(C), in the average, upper, and lower values of loads, respectively.

The maximum final rewards correspond to optimum control actions at the above 3 operating points. The same optimum control actions can be obtained through exploiting the experience (Q-learning memory,  $Q(s, a) \neq 0.0$ ) previously gained and presented in Fig. 2(A)-(C).

Fig. 4(A)-(C) depicts the immediate reward (7) at every Q-learning step, corresponding to the average, upper, and lower load values, respectively. In this case, the agent makes only 760 Q-learning steps at average load values [Fig. 4(A)], 900 at upper load values [Fig. 4(B)], and 790 at lower load values [Fig. 4(C)] to find the optimum control actions. Among them the greedy optimal action (a\*) that achieves the maximum total

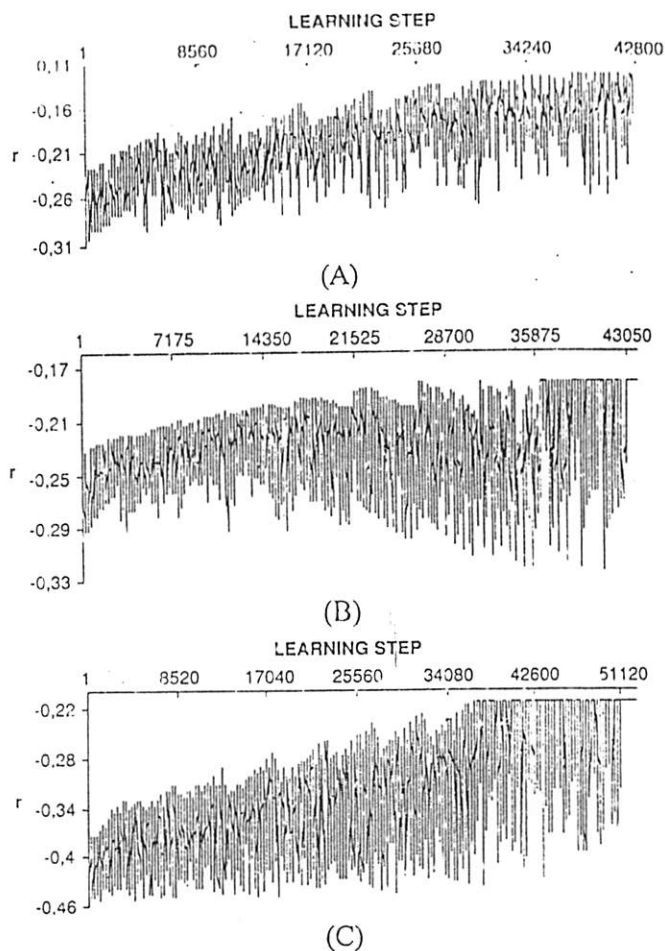


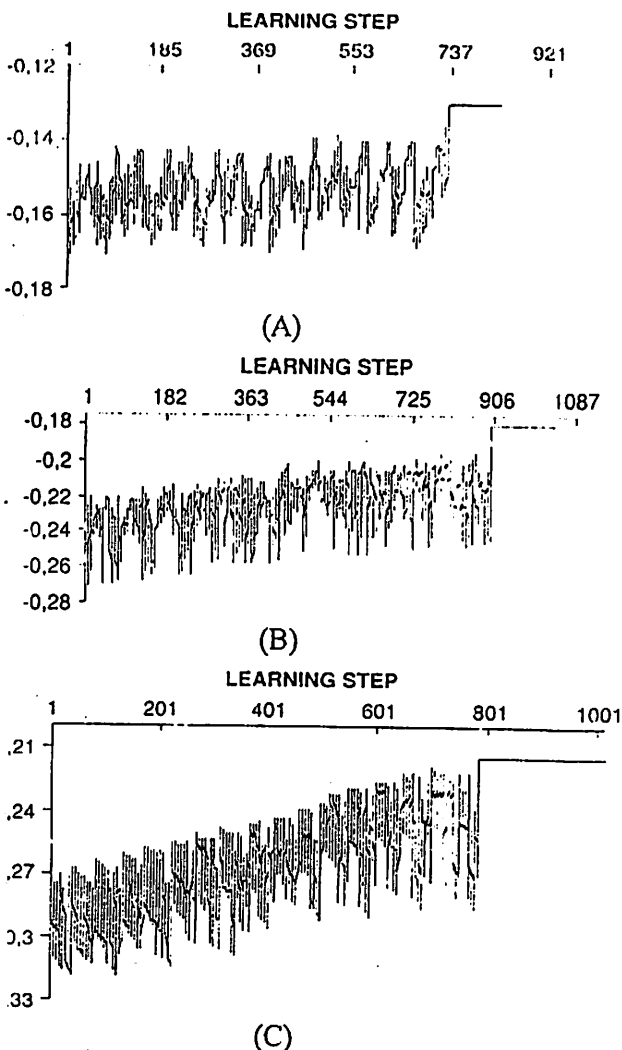
Fig. 3. Immediate rewards of Q-learning procedure when Cutoff Qg2 in the (A) average, (B) upper, and (C) lower values of loads ( $Q(s, a) = 0.0$ ).

reward (8) over the whole planning period is once again the optimal action obtained for the average values of load [Fig. 4(A)]. Table IV shows the maximum total reward calculated at  $-0.565$ , the greedy optimal-action (a\*) and the operating limits of the constrained variables when the greedy-optimal action (a\*) is enforced. In Table IV, the greedy-optimal control settings are compared to the results of the probabilistic CLF method [9]. In contrast to [9], it is seen that Q-learning algorithm provides better results, since all constraints including the upper limit of apparent power flow (T23) on line 2-3 are satisfied.

Table V shows the greedy-optimal control actions (a\*), proposed by Q-learning algorithm at 5 operating points (9) corresponding to the average load values ( $k = 0$ ) and the two adjacent pairs ( $k = \pm 1, \pm 2$ ) together with the maximum total rewards of the optimal actions. The last two columns of Table V represent the Q-learning steps that agent made, with zero Q-learning memory and using previous Q-learning memory, respectively.

A key advantage of the proposed Q-learning algorithm is its flexibility in providing control actions that can satisfy further criteria and thus solve multicriteria optimization problems. For example, if the cost of VAR compensation should be taken into account, then as greedy-optimal action, the action that minimizes compensation at node 9 could be chosen. Table VI shows the operating space of constrained variables when the





Immediate rewards of Q-learning procedure when Cutoff  $Qg2$  in the range, (B) upper, and (C) lower values of loads ( $Q(s,a) \neq 0.0$ ).

greedy-optimal actions ( $a^*$ ) are enforced over the whole planning period. Fig. 5(A) and (B) shows the convergence of the Q-learning algorithm to optimal actions in the case of minimum VAR compensation corresponding to operating points -2 and  $k = -1$  in Table V, respectively. These results show that the Q-learning algorithm provides control settings over the whole planning period and can be more effective than the probabilistic CLF method [9] since it satisfies constraints on minimum compensation.

Another example demonstrates the ability of the Q-learning algorithm to provide optimal reactive power control employing discrete control actions. In this case, the generator's 2, 3, and 6 outputs discretized at 20 positions (0–200 MW) and the compensation at bus 9 at 8 blocks are chosen. The dimension of the action vector is equal to  $20 \times 20 \times 20 \times 8 = 64\,000$ . The total number of solution states is  $2^{15} = 32\,768$ .

Due to the exploitation of the gained memory of previous planning runs at the average load values of this system, the Q-learning algorithm made only 400 iterations to converge (Fig. 5). The greedy optimal control settings and the range of the solution vectors are given in the upper and lower part

TABLE IV  
RESULTS OF Q-LEARNING AND PROBABILISTIC CLF ON  
IEEE 14-BUS (CUTOFF  $Qg2$ )

Q-Learning algorithm		Probabilistic CLF		
mtr = -0.565				
Actions	Greedy-Optimal settings ( $a^*$ )	Optimal Settings		
t56	0.99	0.94		
t49	0.91	0.97		
t47	0.97	0.98		
b9	0.03	0.12		
V6	1.021	1.021		
V1	1.03	1.03		
Constrained Variables	Wmin	Wmax	Wmin	Wmax
Qg3	0.0100	0.5595	0.5120	0.5695
Qg6	0.0462	0.3794	0.3066	0.4214
T23	0.2180	0.6767	0.7063	0.7665
T56	0.0330	0.2569	0.4126	0.4936
V4	0.9772	1.0044	0.9726	0.9822
V5	0.9888	1.0118	0.9763	0.9843
V7	0.9891	1.0311	0.9797	0.9954
V8	0.9891	1.0311	0.9797	0.9954
V9	0.9807	1.0289	0.9749	0.9933
V10	0.9786	1.0235	0.9737	0.9903
V11	0.9940	1.0192	0.9926	1.0013
V12	0.9998	1.0112	1.0008	1.0038
V13	0.9905	1.0086	0.9932	0.9979
V14	0.9603	1.0062	0.9605	0.9765

of Table VII, respectively. Comparing the results of Table III with those given in Table VII it is seen that the latter is more efficient, since all constraints are satisfied. That means the latter control actions are more effective since they can offset the violations of reactive production at node 2 ( $Qg2$ ) and voltage at node 14 ( $V14$ ).

The Q-learning algorithm is also applied to reactive power control of the IEEE 136-bus system. This system consists of 136 buses (33 PV and 103 load buses), 199 lines, 24 transformers, and 17 reactive compensations. In our study the Q-learning algorithm learns the optimum control settings for each of 41 operating points selected from the whole planning period similarly to the IEEE 14-bus case (9). The control variables selected comprise voltages at 4 and 21 PV buses (discrete variation 0.99 to 1.02, step 0.01), taps at transformers 28, 41, and 176 (discrete variation of 0.92 to 1.00, step 0.02) and reactive compensation (b) at buses 3 and 52 (discrete variation of 6 blocks). The total number of actions is  $5^2 \times 5^3 \times 6^2 = 112\,500$ . The constrained variables include voltages at all PQ buses (from 0.96 pu to 1.05 pu) and 3 apparent power-flows at the most heavily loaded lines 156, 177 (upper limit 4.6 p.u.), and 179 (upper limit 3.4 p.u.). The initial control settings violate the power flow limits of all the above lines and upper limit of the voltages of 18, 19, and 23 buses. The Q-learning algorithm learns the greedy-optimal control action resulting in the satisfaction of the constrained variables limits over the whole planning period as shown in Table VIII.

In this case, the agent found the optimum control action at the average load values after 112 980 Q-learning steps (Fig. 7) without using initial memory ( $Q(s,a) = 0.0$ ). The total computing time is 160 s on a 1.4 GHz Pentium-IV PC, compared to the 710 s achieved by other heuristic techniques such as the enhanced genetic algorithms for similar problem on smaller network (24-bus test system), reported in [17]. The time superiority



TABLE V  
GREEDY OPTIMAL ACTIONS OVER THE WHOLE PLANNING PERIOD (CUTOFF Q<sub>g2</sub>)

k	Operating points	Greedy-Optimal actions (a*)					Q-learning steps	
		t56	t49	t47	b9	mtr	With zero initial Q-memory	Using Q-memory
-2	$\mu - 6\sigma/20$	1.01	0.96	0.96	0.00	-0.570	41327	1600
-1	$\mu - 3\sigma/20$	0.99	0.90	0.97	0.00	-0.566	41078	1550
0	$\mu$	0.99	0.91	0.97	0.03	-0.565	42688	1400
1	$\mu + 3\sigma/20$	0.99	0.90	0.97	0.03	-0.567	40236	1150
2	$\mu + 6\sigma/20$	0.99	0.92	0.98	0.06	-0.569	40451	1200

TABLE VI  
OPERATING SPACE OF CONSTRAINED VARIABLES WHEN THE CRITERION IS MINIMUM VAR COMPENSATION AT BUS 9

Actions	Greedy-Optimal settings (a*)			
	k = -2	k = -1		
t56	1.01	0.99		
t49	0.96	0.90		
t47	0.96	0.97		
b9	0.00	0.00		
V6	1.021	1.021		
V1	1.03	1.03		
Constrained Variables	Wmin	Wmax	Wmin	Wmax
Qg3	0.0001	0.5266	0.0102	0.5713
Qg6	0.1544	0.4872	0.0577	0.3908
T23	0.2186	0.6731	0.2177	0.6732
T56	0.0620	0.2889	0.0321	0.2598
V4	0.9814	1.0087	0.9756	1.0028
V5	0.9935	1.0168	0.9877	1.0107
V7	0.9814	1.0234	0.9872	1.0291
V8	0.9814	1.0234	0.9872	1.0291
V9	0.9667	1.0149	0.9787	1.0268
V10	0.9670	1.0119	0.9769	1.0217
V11	0.9881	1.0133	0.9932	1.0184
V12	0.9987	1.0101	0.9996	1.0110
V13	0.9884	1.0066	0.9902	1.0083
V14	0.9599	0.9972	0.9601	1.0048

The Q-learning algorithm results from the fact that only one power flow solution is executed at each learning step. Further applications on larger power systems show that the learning algorithm requires much more computational time and larger stored memory, but drastically reduced, when the experience gained at these learning steps is exploited.

VI. DISCUSSION

Interpreting the previous results, the following advantages and disadvantages of the Q-learning algorithm, as applied to reactive power control problem, can be summarized. The Q-learning algorithm converges to optimal values independently of how the "agent" (Q-learning algorithm) explores the power system operating states. In our application, this means that discrete control actions are optimized independently of the initial values of the control settings and of the probabilistic distributions of loads. In addition, the Q-learning algorithm provides increased flexibility in choosing control actions, ... assigning their priorities or even dealing with multi-objective functions. Regarding execution times, it should be noted that, when the Q-learning algorithm is applied for the first time to control a large system, it converges quite slowly to the optimum power flow solution. Moreover, increasing control actions also increases the time needed by the Q-learning algorithm to converge. This is, however, a general feature of all heuristic methods. In fact Q-learning algorithm proves to be faster than

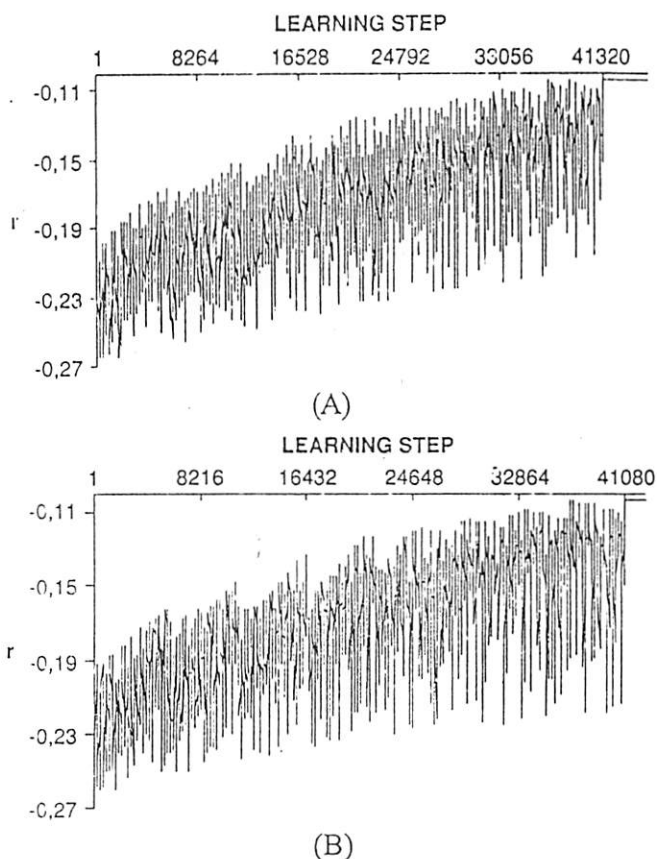


Fig. 5. Immediate rewards of Q-learning procedure when Cutoff Q<sub>g2</sub> in the (A)  $\mu - 6\sigma/20$  and (B)  $\mu - 3\sigma/20$  load values.

the enhanced genetic algorithm [17]. The main advantage of the Q-learning algorithm is that it can use the experience gained as initial memory for next searches of the system, leading to subsequent convergence at short computing times. One disadvantage of the Q-learning method is its increased storage needs, i.e., in order to exploit the greedy search strategy results of previous searches need to be stored. With the present availability of computer storage space, however, this is not considered a major drawback.

VII. CONCLUSION

In this paper, the Reinforcement Learning method is applied to the solution of the constrained load flow problem. An iterative novel Q-learning algorithm is implemented in order to provide the optimal action (offline control settings) over a planning period satisfying all physical and operating limits of the constrained variables. Optimal control settings are learnt by experience adjusting a closed-loop control rule, which is mapping

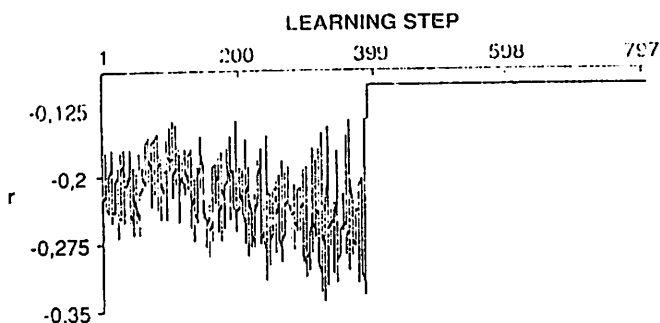


Fig. 6. Immediate rewards of Q-learning procedure in the average values of loads with initial  $Q(s, a) \neq 0.0$ .

TABLE VII  
OPERATING SPACE OF CONSTRAINED VARIABLES ON IEEE 14-BUS WITH ALTERNATIVE CONTROL ACTIONS

Q-Learning algorithm		
mtr = -0.804		
Actions	Greedy-Optimal Settings (n*)	
Pg2	0 MW	
Pg3	60 MW	
Pg6	30 MW	
b9	15 MVar	
Constrained Variables	Wmin	Wmax
Qg2	0.0029	0.1794
Qg3	0.1408	0.5741
Qg6	0.0085	0.2236
T23	0.1333	0.5896
T56	0.2042	0.3818
V4	0.9713	0.9925
V5	0.9757	0.9930
V7	0.9802	1.0153
V8	0.9802	1.0183
V9	0.9750	1.0204
V10	0.9740	1.0166
V11	0.9919	1.0160
V12	0.9993	1.0105
V13	0.9898	1.3076
V14	0.9601	1.0009

TABLE VIII  
RESULTS OF Q-LEARNING ON IEEE 136-BUS

Q-Learning algorithm				
mtr = -0.61				
Actions	Greedy-Optimal settings (n*) (p.u.)		Initial Control Settings (p.u.)	
V4	1.00		1.03	
V21	0.99		1.03	
T28	0.92		1.00	
T41	0.92		0.98	
T176	0.92		1.00	
b3	0.15		0.16	
b52	0.17		0.18	
Constrained Variables	Wmin	Wmax	Wmin	Wmax
V18	3.987	1.021	1.032	1.066
V19	0.998	1.028	1.034	1.064
V23	1.010	1.049	1.021	1.052
T156	3.987	4.500	4.234	5.358
T177	3.948	4.501	3.567	6.601
T179	2.675	3.234	2.933	3.687

operating states to control actions by means of reward values. The Q-learning algorithm is applied to the IEEE 14-bus system and to the IEEE 136-bus system for constrained reactive power

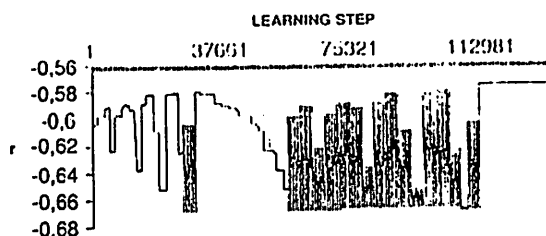


Fig. 7. Immediate rewards of Q-learning in the average values of loads of IEEE-136 bus without: initial memory  $Q(s, a) = 0.0$ .

control. Compared to probabilistic CLF methods, results show that the Q-learning algorithm is able to provide optimal control settings. Moreover, the approach is very flexible allowing its application to multicriteria optimization problems, e.g., assigning priorities to control actions. Further applications show that the Q-learning algorithm is faster than the enhanced genetic algorithm and previous gained experience of a power system can be exploited.

## REFERENCES

- [1] W. Kellermann, H. M. Z. El-Din, C. E. Graham, and G. A. Maria, "Optimization of fixed tap transformer settings in bulk electricity systems," *IEEE Trans. Power Syst.*, vol. 9, pp. 1126-1132, Aug. 1991.
- [2] J. G. Vlachogiannis, "Control adjustments in fast decoupled load flow," *Elect. Power Syst. Res.*, vol. 31, no. 3, pp. 185-194, Dec. 1994.
- [3] H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Nakanishi, "A practical swarm optimization for reactive power and voltage control considering voltage security assessment," *IEEE Trans. Power Syst.*, vol. 15, pp. 1232-1239, Nov. 2000.
- [4] P. K. Satpathy, D. Das, and P. B. Dutta Gupta, "A novel fuzzy index for steady state voltage stability analysis and identification of critical busbars," *Elect. Power Syst. Res.*, vol. 63, no. 2, pp. 127-140, Sept. 2002.
- [5] B. Cova, N. Losignore, P. Marannino, and M. Montagna, "Contingency constrained optimal reactive power flow procedures for voltage control in planning and operation," *IEEE Trans. Power Syst.*, vol. 10, pp. 602-608, May 1995.
- [6] E. Vahedi, Y. Mansour, J. Tamby, W. Li, and D. Sun, "Large scale voltage stability constrained optimal VAR planning and voltage stability applications using existing OPF/optimal VAR planning tools," *IEEE Trans. Power Syst.*, vol. 14, pp. 65-74, Feb. 1999.
- [7] D. Gan, R. J. Thomas, and R. D. Zimmerman, "Stability constrained optimal power flow," *IEEE Trans. Power Syst.*, vol. 15, pp. 535-540, May 2000.
- [8] R. A. Jabr and A. H. Coonick, "Homogeneous interior point method for constrained power scheduling," *Proc. Inst. Elect. Eng. Gener. Transm. Distrib.*, vol. 147, no. 4, pp. 239-244, July 2000.
- [9] T. S. Karakatsani and N. D. Hatziargyriou, "Probabilistic constrained load flow based on sensitivity analysis," *IEEE Trans. Power Syst.*, vol. 9, pp. 1853-1860, Nov. 1994.
- [10] L. Jeeng-Mia, L. Min-E, and H. Ching-Lien, "Investigation of fuzzy real power flow modeling with probabilistic-heuristic based information," *Elect. Power Syst. Res.*, vol. 32, no. 3, pp. 175-186, Mar. 1995.
- [11] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, no. 3, pp. 279-292, Aug. 1992.
- [12] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, pp. 237-285, Jan.-June 1996.
- [13] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *Adaptive Computations and Machine Learning*, 1998.
- [14] D. P. Bertsekas and J. N. Tsitsiklis, *Neuro-Dynamic Programming*. Belmont, MA: Athena Scientific, 1996.
- [15] R. N. Allan and M. R. G. Al-Shakarchi, "Probabilistic techniques in a.c. load-flow analysis," in *IEE Proc. Gener. Transm. Distrib.*, vol. 124, Feb. 1977, pp. 154-160.
- [16] J. F. Kennedy and E. S. Keeping, *Mathematics in Statistics*. Princeton, NJ: Nostrand, 1962.
- [17] A. G. Bakirtzis, P. N. Biskas, C. E. Zoumas, and V. Petridis, "Optimal power flow by enhanced genetic algorithm," *IEEE Trans. Power Syst.*, vol. 17, pp. 229-236, May 2002.



John G. Vlachogiannis received the Diploma in electrical engineering and the Ph.D. degree both from Aristotle University of Thessaloniki, Greece, in 1990 and 1994, respectively.

He is currently an Assistant Professor and the Head of Laboratories for Industrial and Energy Informatics (IEI-Lab.) and Total Quality Management (TQM-Lab.) in the Informatics and Computer Technology Department of the Technological Educational Institute (TEI) of Lamia, Lamia, Greece. His research interests include control and management

strategies and artificial intelligence techniques in planning and operation of power and industrial systems.

Dr. Vlachogiannis is a member of the Greek Computer Society (member of IFIP and CEPIS), the Hellenic Artificial Intelligence Society (EETN), and the Technical Chamber of Greece.



Nikos D. Hatzargyriou (S'80 M'82 SM'90) was born in Athens, Greece, in 1954. He received the Diploma in electrical and mechanical engineering from National Technical University of Athens (NTUA), Greece, in 1976 and M.Sc. and Ph.D. degrees from the University of Manchester Institute of Science and Technology (UMIST), Manchester, U.K., in 1979 and 1982, respectively.

He is currently a Professor with the Power Division of the Electrical and Computer Engineering Department, NTUA. His research interests include dis-

persed generation and renewable energy sources, dynamic security, and artificial intelligence techniques.

Dr. Hatzargyriou is a member of CIGRE SCC6 and the Technical Chamber of Greece.



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

Jalan Suningrat No. 45 Taman Sidoarjo 61257

Telepon : (031) 7882113, 7882114

Kotak Pos : 4119 SBS

Facsimile : (031) 7882578, 7881024

E-mail : region4@pln-jawa-bali.co.id

Website : www.pln-jawa-bali.co.id

Nomor : 054/1330/RJTB/2007  
Surat Sdr. No. : ITN-663/III.TA-2/2/07  
Lampiran : 1 (satu) lampiran.  
Perihal : Ijin Survey / Pengambilan Data.

02 MAR 2007

**Kepada Yth**

**INSTITUT TEKNOLOGI NASIONAL MALANG**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**DI**  
**MALANG**

Menunjuk surat Saudara nomor : ITN-663/III.TA-2/2/07 tanggal 08 Pebruari 2007 perihal : Survey / Permintaan Data, dengan ini diberitahukan bahwa kami tidak keberatan untuk memberikan ijin kepada Mahasiswa Saudara, bernama :

➤ **TOMMY NUGROHO**

**Nim : 00.12.079**

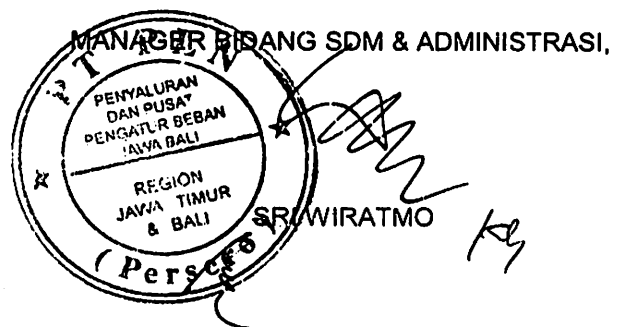
Untuk melakukan Pengambilan Data pada PT. PLN (Persero) P3B Region Jawa Timur dan Bali Bidang OPHAR, dengan persyaratan sebagai berikut :

1. Mahasiswa tersebut diatas supaya mengisi dan menanda tangani Surat Pernyataan 1 (satu) lembar bermeterai Rp. 6.000,-
2. Mahasiswa yang bersangkutan agar mematuhi peraturan/ketentuan yang berlaku di PT. PLN (PERSERO) sehingga faktor-faktor kerahasiaan harus benar-benar diutamakan.
3. Semua biaya perjalanan, penginapan, makan dan lain sebagainya tidak menjadi tanggungan PT. PLN (Persero) P3B Region Jawa Timur dan Bali.
4. Buku Laporan Kerja Praktek Mahasiswa tersebut agar dikirimkan kepada PT. PLN (Persero) P3B Region Jawa Timur dan Bali 1 (satu) buah.
5. Untuk informasi lebih lanjut dapat menghubungi PT. PLN (Persero) P3B Region Jawa Timur dan Bali Cq. Bidang SDM & ADMINISTRASI.

Demikian harap maklum dan terima kasih atas perhatian saudara.

Tembusan Yth. :

1. M.SDMO PLN P3B JB.
2. M.OPHAR PLN P3B RJTB.
3. Sdr. Tommy Nugroho



## SURAT PERNYATAAN



Yang bertanda tangan dibawah ini, saya :

Nama : TOMMY NUGROHO  
Pria/ Wanita : PRIA  
Tempat / Tanggal lahir : SAMBOJA 30 - OKTOBER - 1981  
Alamat / No. telepon : Jl. KH. SIMPANG YUSUF A 12  
0856 4646 3009  
Pekerjaan : MAHASISWA ITU MALANG

Dengan ini saya menerangkan bahwa :

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

Surabaya, 23 - 02 - 2007  
Yang membuat pernyataan

  
  
( TOMMY NUGROHO )