



UNIVERSITAS INDONESIA

PENGEMBANGAN MODEL DAN SIMULASI
SISTEM PEMBANGKIT LISTRIK TENAGA HIBRIDA (PLTH)
DENGAN METODE PETRI NET

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar Magister Teknik

SUKMA ABADI
0906578176

FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK ELEKTRO
KEKHUSUSAN TEKNIK KONTROL INDUSTRI
DEPOK
JUNI 2011

HALAMAN PERNYATAAN ORISINALITAS

Tesis ini adalah hasil karya saya sendiri,
dan semua sumber, baik yang dikutip maupun dirujuk
telah saya nyatakan dengan benar.

Nama : SUKMA ABADI

NPM : 0906578176

Tanda Tangan :

Tanggal : 30 Juni 2011

HALAMAN PENGESAHAN

Seminar ini diajukan oleh :

Nama : SUKMA ABADI
NPM : 0906578176
Program Studi : Teknik Elektro
Judul Seminar : Pengembangan Model dan Simulasi Sistem Pembangkit Listrik Tenaga Hibrida (PLTH) dengan Metode Petri Net.

telah berhasil dipertahankan di hadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk mendapatkan gelar Master Teknik pada Program Studi Teknik Elektro, Fakultas Teknik, Universitas Indonesia.

DEWAN PENGUJI

Pembimbing : Dr. Abdul Halim, M.Eng. ()
Penguji : Dr. Ir. Feri Yusivar, M.Eng. ()
Penguji : Ir. Ariès Subiantoro, M.SEE. ()
Penguji : Dr.-Ing. Eko Adhi Setiawan ()

Ditetapkan di : Depok
Tanggal : 30 Juni 2011

KATA PENGANTAR

Segala puji bagi Allah swt saya panjatkan karena berkat hidayah dan pertolongan-Nyalah, sehingga saya dapat menyelesaikan tesis ini. Penulisan tesis ini saya buat dalam rangka memenuhi salah satu syarat untuk memperoleh gelar master teknik pada Departemen Teknik Elektro Fakultas Teknik Universitas Indonesia.

Saya menyadari bahwa tesis ini dapat dibuat dengan bantuan banyak pihak. Oleh karena itu, saya menghaturkan rasa terima kasih yang sebesar-besarnya kepada:

- 1) Dr. Abdul Halim, M.Eng. selaku dosen pembimbing yang telah menyediakan waktu, tenaga, dan pikiran untuk mengarahkan saya dalam penyusunan tesis ini.
- 2) Orangtua, istri, dan keluarga saya yang telah memberikan bantuan dan dukungan moril dan materil.
- 3) Para sahabat yang telah banyak membantu saya dalam menyelesaikan tesis ini.

Akhir kata, saya berharap semoga Allah swt berkenan memberi balasan atas semua kebaikan pihak-pihak yang telah membantu. Semoga tesis ini dapat memberi manfaat bagi pengembangan ilmu, khususnya dalam bidang teknik elektro.

Depok, Juni 2011
Penulis

**HALAMAN PERNYATAAN PERSETUJUAN PUBLIKASI
TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIS**

Sebagai sivitas akademik Universitas Indonesia, saya yang bertanda tangan di bawah ini:

Nama : Sukma Abadi
NPM : 0906578176
Program Studi : Pascasarjana (Magister)
Departemen : Teknik Elektro
Fakultas : Teknik
Jenis Karya : Tesis

demikian pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Indonesia ***Hak Bebas Royalti Noneksklusif (Non-exclusive Royalty Free Right)*** atas karya ilmiah saya yang berjudul:

*Pengembangan Model dan Simulasi
Sistem Pembangkit Listrik Tenaga Hibrida (PLTH)
dengan Metode Petri Net*

beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Noneksklusif ini Universitas Indonesia berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di : Depok
Pada tanggal : 30 Juni 2011

Yang menyatakan

(Sukma Abadi)

ABSTRAK

Nama : Sukma Abadi
Program Studi : Teknik Elektro
Judul : Pengembangan Model dan Simulasi Sistem Pembangkit Listrik
Tenaga Hibrida (PLTH) dengan Metode Petri Net.

Tesis ini membahas tentang model Petri Net dari sistem Pembangkit Listrik Tenaga Hibrida (PLTH) yang kompleks, baik dari segi jenis dan jumlah sumber energi listriknya, jenis beban yang disuplai, maupun dari komponen-komponen lain yang digunakan. Sistem PLTH yang dibuat modelnya meliputi 1 unit photovoltaic, 2 unit turbin angin, 1 unit generator diesel, baterai (energy storage), konverter bidirectional, ac bus, dc bus, beban ac, dan beban dc. Penelitian bertujuan untuk menerapkan strategi *load-following*, *cycle-charging*, dan strategi *battery state of charge* dalam pengoperasian model sistem PLTH yang memadukan antara sumber-sumber energi terbarukan dan sumber energi tidak terbarukan. Analisis dilakukan dengan mengamati hasil simulasi model yang diperoleh dari ketiga macam strategi yang diterapkan kemudian menghitung estimasi biaya yang diperlukan dari ketiga strategi tersebut. Hasil simulasi menunjukkan bahwa biaya terendah diperoleh dengan menerapkan strategi *battery state of charge*. Model sistem PLTH dibuat dengan metode Petri Net, sedangkan simulasinya dilakukan dengan menggunakan software GPenSIM yang dioperasikan dalam platform MATLAB. Hasil simulasi menunjukkan bahwa ketiga strategi penyaluran daya listrik dapat digunakan pada model sistem PLTH yang dibuat.

Kata kunci:

PLTH, Energi Terbarukan, load-following, cycle-charging, battery state of charge, Petri Net, GPenSIM.

ABSTRACT

Name : Sukma Abadi
Study Program : Electrical Engineering
Title : Model Development and Simulation of Hybrid Power Generation Systems Using Petri Net Method.

This thesis discusses the Petri Net model of the Hybrid Power Generation Systems (HPGS) complex, both in the type and the amount of electrical energy sources, types of loads supplied, as well as from other components used. The system modeled the HPGS which includes a photovoltaic unit, 2 units of wind turbine, a diesel generator unit, the battery (energy storage), bidirectional converter, ac bus, dc bus, ac load, and dc load. The study aims to apply the load-following strategy, cycle-charging, and battery state of charge strategy in the operation model of the HPGS that combine renewable energy sources and non-renewable energy sources. The analysis was performed by observing the simulation results obtained by the model with three kinds of strategy that are applied then to calculate the estimated cost required of the three strategies. The simulation results show that the lowest cost is obtained by applying the battery state of charge strategy. The HPGS model created by Petri Net method, while the simulation is done by using software GPenSIM operated in MATLAB platform. The simulation results show that all of the strategy can be used in the model of the HPGS.

Keywords:

Hybrid Power Generation System, load-following, cycle-charging, and battery state of charge, Petri Net, GPenSIM.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN ORISINALITAS	ii
HALAMAN PENGESAHAN	iii
KATA PENGANTAR	iv
HALAMAN PERSETUJUAN PUBLIKASI KARYA ILMIAH	v
ABSTRAK	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR GAMBAR	x
DAFTAR TABEL	xi
DAFTAR LAMPIRAN	xii
1. PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Perumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	3
1.5 Batasan Masalah	3
1.6 Sistematika Penulisan	4
2. TINJAUAN PUSTAKA	5
2.1 Pembangkit Listrik Tenaga Hibrida (PLTH)	5
2.1.1 Photovoltaic	7
2.1.2 Wind Turbine	8
2.1.3 Generator Diesel	9
2.2 Petri Net	10
2.2.1 Struktur Petri Net	10
2.2.2 Penggambaran Struktur PN	11
2.2.3 Struktur Proses PN	12
2.2.4 Eksekusi Petri Net	13
2.2.5 Properti Petri Net	14
2.3 GPenSIM	17
2.3.1 Model dan Simulasi	18
2.3.2 Petri net Definition Files (PDF)	18
2.3.3 Transition Definition Files (PDF)	19
2.3.4 Main Simulation File (MSF)	21
2.3.5 Hasil Simulasi	21
2.3.6 Contoh Sederhana	22
3. PEMODELAN SISTEM PEMBANGKIT LISTRIK TENAGA HIBRIDA	25
3.1 Prosedur Pemodelan	25
3.1.1 Mekanisme Proses Sistem PLTH	26
3.1.2 Komponen-komponen Sistem PLTH	28
3.1.3 Blok Diagram Model Sistem PLTH	29
3.1.4 Model PN Kebutuhan Beban	29
3.1.5 Model PN Photovoltaic	30
3.1.6 Model PN Wind Turbine	31

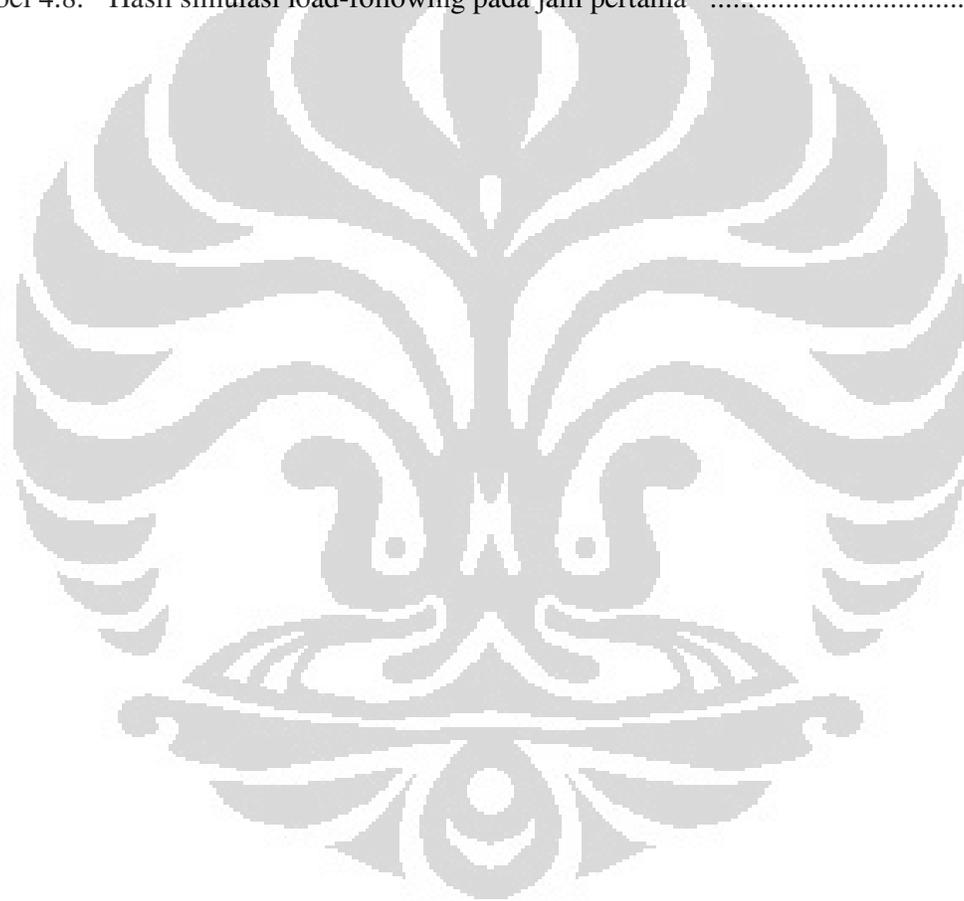
3.1.7 Model PN Generator Diesel	32
3.1.8 Model PN Baterai (Energy Storage)	32
3.1.9 Model PN Konverter Bidirectional	33
3.1.10 Model PN Bus dan Beban	33
3.1.11 Struktur PN Model Sistem PLTH	34
3.2 Prosedur Simulasi	35
3.2.1 Kode Program PDF	37
3.2.2 Kode Program TDF	37
3.2.3 Kode Program MSF	37
3.2.4 Running Program Simulasi Model Petri Net Sistem PLTH	38
3.2.5 Validasi Hasil Simulasi	39
4. SIMULASI DAN PEMBAHASAN	40
4.1 Desain Simulasi	40
4.2 Hasil Simulasi dengan Strategi Load-following	45
4.3 Hasil Simulasi dengan Strategi Cycle-charging	46
4.4 Hasil Simulasi dengan Strategi Battery state of charge	47
4.5 Analisis Hasil Simulasi	48
4.6 Analisis Sifat Reachability	50
4.7 Analisis Deadlock/Liveness	54
5. KESIMPULAN	55
DAFTAR REFERENSI	57

DAFTAR GAMBAR

Gambar 2.1.	Kombinasi PV dan diesel	5
Gambar 2.2.	Kombinasi PV dan wind turbine (WT)	5
Gambar 2.3.	Kombinasi PV, WT, dan diesel pada sistem stand-alone	6
Gambar 2.4.	Contoh panel photovoltaic	8
Gambar 2.5.	Contoh wind turbine	9
Gambar 2.6.	Struktur dasar Petri Net	11
Gambar 2.7.	Contoh struktur Petri Net	11
Gambar 2.8.	Proses tersebar	12
Gambar 2.9.	Proses sinkronisasi	12
Gambar 2.10.	Proses konflik	13
Gambar 2.11.	Proses berurutan	13
Gambar 2.12.	Perpindahan token struktur PN	14
Gambar 2.13.	Arsitektur GPenSIM	17
Gambar 2.14.	Pemisahan MSF dan PDF	18
Gambar 2.15.	Kedudukan TDF di antara MSF dan PDF	19
Gambar 2.16.	Model PN contoh sederhana	22
Gambar 2.17.	PDF untuk model PN contoh sederhana	22
Gambar 2.18.	MSF untuk model PN contoh sederhana	23
Gambar 2.19.	Print statespace untuk model PN contoh sederhana	23
Gambar 2.20.	Plot untuk model PN contoh sederhana	24
Gambar 3.1.	Flowchart prosedur pemodelan sistem PLTH	26
Gambar 3.2.	Sistem PLTH kombinasi pv, wt, dan diesel	27
Gambar 3.3.	Blok diagram model sistem PLTH	28
Gambar 3.4.	Model PN kebutuhan beban	29
Gambar 3.5.	Model PN photovoltaic	30
Gambar 3.6.	Model PN wind turbine	31
Gambar 3.7.	Model PN Generator diesel	31
Gambar 3.8.	Model PN baterai	32
Gambar 3.9.	Model PN konverter bidirectional	33
Gambar 3.10.	Model PN bus dan beban	33
Gambar 3.11.	Struktur Petri Net model sistem PLTH	34
Gambar 3.12.	Flowchart prosedur simulasi model sistem PLTH	36
Gambar 4.1.	Kurva beban selama 1 hari	40
Gambar 4.2.	Grafik radiasi surya 1 hari	41
Gambar 4.3.	Grafik kecepatan angin 1 hari	41
Gambar 4.4.	Ketersediaan daya listrik pv 1 hari	42
Gambar 4.5.	Ketersediaan daya listrik angin 1 hari	43
Gambar 4.6.	Jumlah token kebutuhan beban ac dan dc 1 hari	43
Gambar 4.7.	Jumlah token pv,wt1,wt2 1 hari	44
Gambar 4.8.	Grafik ketersediaan pasokan daya yang selalu cukup	49
Gambar 4.9.	Grafik ketersediaan pasokan daya yang kadang tidak cukup	50
Gambar 4.15.	Kondisi awal struktur PN model sistem PLTH	52
Gambar 4.16.	Kondisi yang diinginkan dari struktur PN model sistem PLTH	53

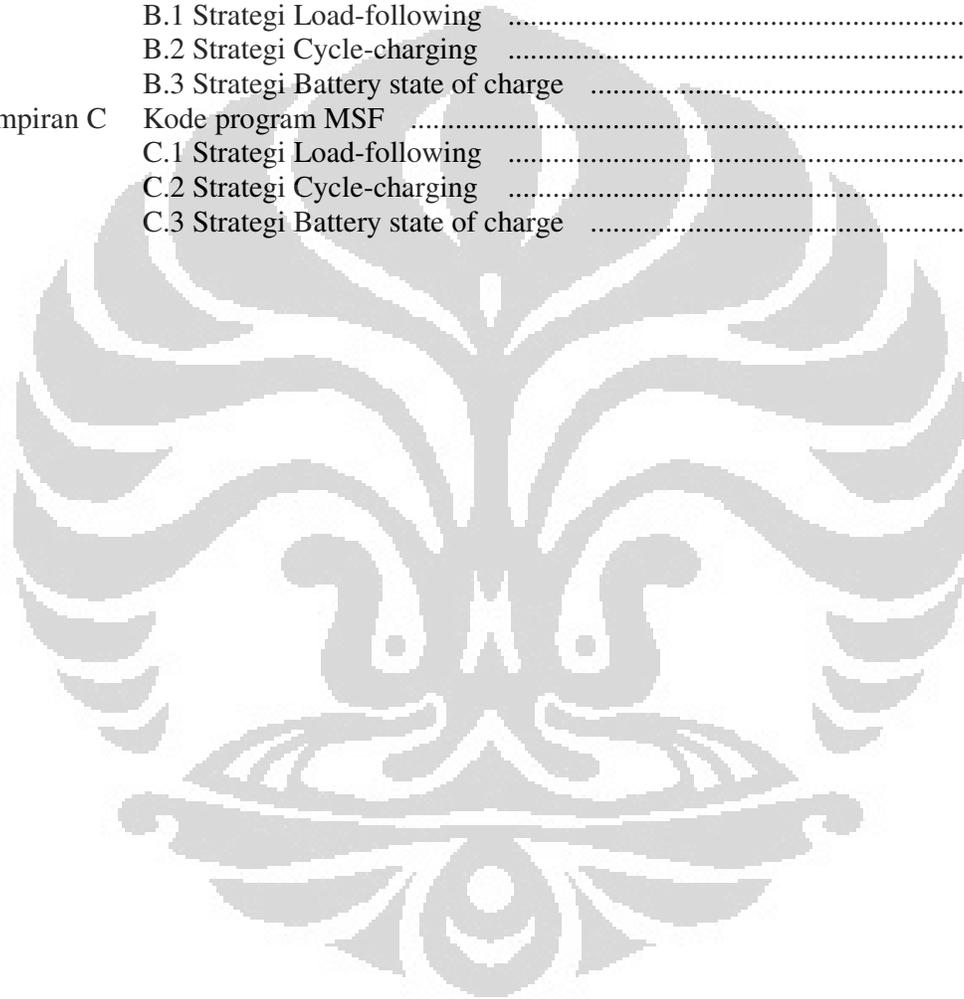
DAFTAR TABEL

Tabel 4.1.	Hasil simulasi strategi load-following	45
Tabel 4.2.	Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada LF	45
Tabel 4.3.	Hasil simulasi strategi cycle-charging	46
Tabel 4.4.	Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada CC	46
Tabel 4.5.	Hasil simulasi strategi battery state of charge	47
Tabel 4.6.	Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada BSC	47
Tabel 4.7.	Perhitungan biaya dengan tiga macam strategi	49
Tabel 4.8.	Hasil simulasi load-following pada jam pertama	54



DAFTAR LAMPIRAN

Lampiran A	Kode program PDF	59
	A.1 Strategi Load-following	59
	A.2 Strategi Cycle-charging	60
	A.3 Strategi Battery state of charge	61
Lampiran B	Kode program TDF	62
	B.1 Strategi Load-following	62
	B.2 Strategi Cycle-charging	63
	B.3 Strategi Battery state of charge	64
Lampiran C	Kode program MSF	65
	C.1 Strategi Load-following	65
	C.2 Strategi Cycle-charging	69
	C.3 Strategi Battery state of charge	73



BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Pemenuhan kebutuhan listrik masyarakat saat ini tidak hanya mengandalkan sumber listrik dari pembangkit listrik konvensional, namun sudah banyak memanfaatkan sumber daya alam yang tersedia, seperti air, angin, dan cahaya matahari (surya). Pemanfaatan sumber energi listrik juga tidak hanya memandang aspek kecilnya biaya investasi dan biaya operasional suatu jenis pembangkit listrik semata, namun juga melihat aspek yang lain seperti rendahnya emisi (dampak lingkungan) yang ditimbulkan dan potensi ketersediaannya di alam.

Listrik dari sumber energi fosil tidak bergantung pada kondisi cuaca dan biaya pembangunannya relatif lebih murah dibandingkan dengan listrik dari sumber energi terbarukan. Namun demikian, ketersediaan bahan bakar untuk pengoperasian sumber listrik dari fosil sudah sangat terbatas, sehingga diperlukan alternatif lain berupa sumber energi terbarukan seperti energi surya (photovoltaic) dan energi angin (wind energy). Energi listrik dari sumber energi terbarukan juga menimbulkan dampak lingkungan yang lebih rendah dibandingkan dengan sumber energi fosil. Oleh karena itu, pemanfaatan energi listrik gabungan dari sumber energi fosil dan sumber energi terbarukan semakin dikembangkan. Perpaduan kedua sumber energi tersebut dikenal sebagai sistem Pembangkit Listrik Tenaga Hibrida (PLTH).

Pada proses penyuplai daya listrik dari sumber-sumber daya listrik ke beban pada sistem PLTH dikenal adanya tiga macam strategi, yaitu load-following, cycle-charging, dan battery state of charge. Strategi mana yang akan digunakan sangat bergantung pada kurva beban (load profile) atau fluktuasi pemakaian energi listrik serta ketersediaan sumber-sumber daya listrik yang akan digunakan.

Oleh karena itu, penulis memandang perlunya dilakukan analisis terhadap penerapan ketiga macam strategi tersebut pada sistem PLTH. Agar penelitian menjadi lebih efisien, diperlukan model atau prototipe yang mewakili sistem PLTH sebenarnya sebagai obyek penelitian. Selain prototipe fisik sistem PLTH, diperlukan pula model simulasi yang dapat menggambarkan mekanisme kerja sistem PLTH tersebut.

Atas dasar itu, penulis bermaksud untuk membuat penelitian dengan judul “*Pengembangan Model dan Simulasi Sistem Pembangkit Listrik Tenaga Hibrida (PLTH) dengan Metode Petri Net*”. Pada penelitian ini, penulis akan membuat pemodelan sistem PLTH perpaduan antara photovoltaic, wind turbine, dan generator diesel sebagai sistem yang berdiri sendiri (stand alone). Pemodelan akan dibuat dengan menggunakan metode Petri Net. Sebagai perangkat simulasi, penulis menggunakan GPenSIM (General Purpose Petri Net Simulator) yang dijalankan dengan platform MATLAB. Salah satu tujuan dilakukannya simulasi model adalah untuk mengetahui strategi mana yang paling efektif digunakan dari sistem PLTH untuk suatu kondisi tertentu (kurva beban dan ketersediaan pasokan listrik dari sumber).

1.2 Perumusan Masalah

Berdasarkan atas rencana penelitian yang telah dikemukakan pada bagian latar belakang masalah di atas, maka dapat dibuat rumusan berikut:

- Bagaimana menerapkan tiga macam strategi penyuplaian daya listrik dari sumber ke beban pada sistem Pembangkit Listrik Tenaga Hibrida (PLTH) yang memadukan sumber-sumber energi terbarukan dan sumber energi tidak terbarukan?
- Bagaimana membuat model sistem PLTH yang kompleks (dengan 3 macam sumber daya listrik, baterai, konverter bidirectional, ac bus, dc bus, beban ac dan beban dc) menggunakan metode Petri Net?
- Bagaimana melakukan simulasi model Petri Net dari PLTH dengan GPenSIM?

1.3 Tujuan Penelitian

Tujuan penelitian ini adalah:

- untuk menerapkan tiga macam strategi penyuplaian daya listrik dari sumber ke beban pada sistem Pembangkit Listrik Tenaga Hibrida (PLTH) yang memadukan sumber-sumber energi terbarukan dan sumber energi tidak terbarukan

- untuk membuat model sistem PLTH yang kompleks (dengan 3 macam sumber daya listrik, baterai, konverter bidirectional, ac bus, dc bus, beban ac dan beban dc) menggunakan metode Petri Net
- untuk melakukan simulasi model Petri Net dari PLTH dengan GPenSIM.

1.4 Manfaat Penelitian

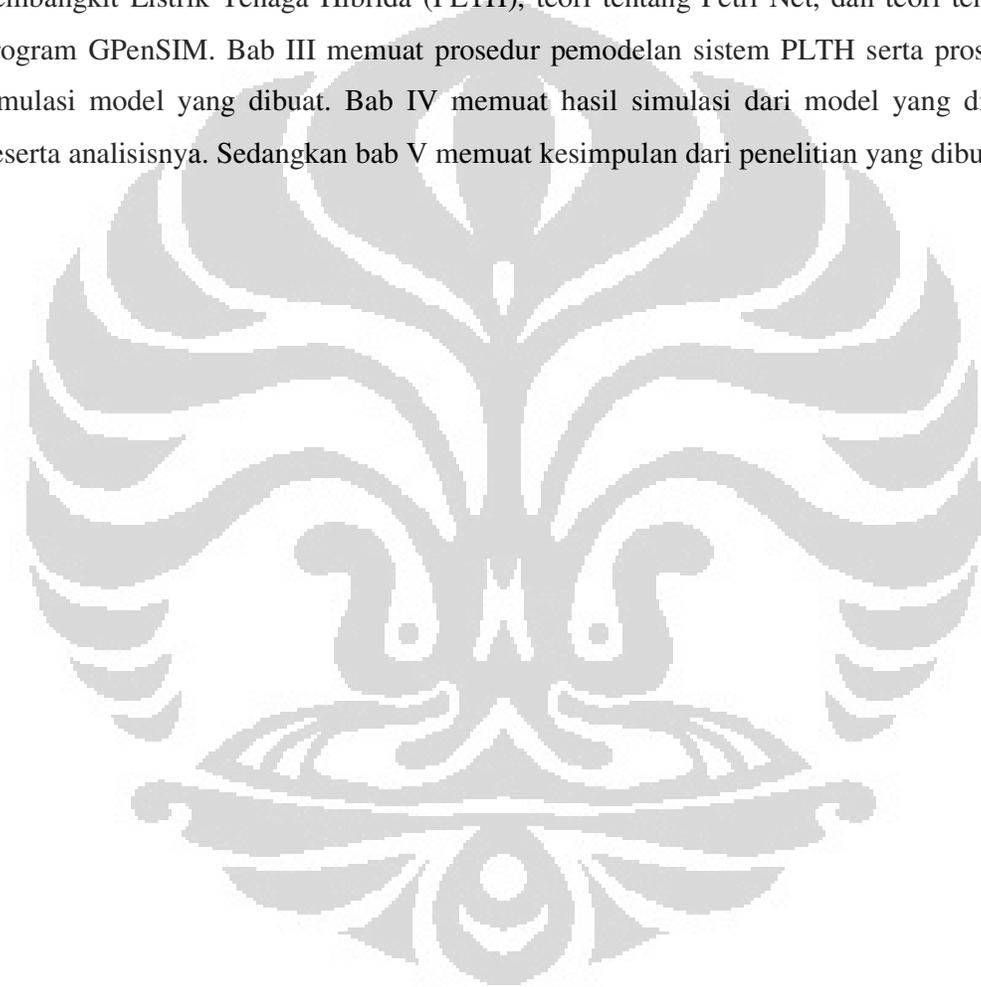
- Hasil penelitian dapat digunakan sebagai analisis tentang mekanisme kerja dari sistem Pembangkit Listrik Tenaga Hibrida (PLTH). Selanjutnya, hasil analisis tersebut dapat digunakan untuk menentukan strategi yang paling tepat diterapkan pada suatu daerah yang menggunakan sistem PLTH.
- Hasil penelitian ini juga dapat dijadikan sebagai referensi tentang keunggulan metode Petri Net dibandingkan dengan metode sistem kejadian diskrit lainnya.

1.5 Batasan Masalah

- Sistem PLTH yang digunakan merupakan perpaduan antara photovoltaic (kapasitas 3 kW), 2 unit wind turbine (masing-masing dengan kapasitas 1 kW), dan generator diesel (kapasitas 4 kW) sedangkan kapasitas beban ac maksimum sebesar 6 kW dan kapasitas beban dc maksimum sebesar 1 kW. Kapasitas energy storage (baterai) maksimum diasumsikan sebesar 10 kW.
- Konverter yang digunakan bersifat bidirectional (dapat mengubah listrik ac menjadi dc atau sebaliknya) dan diasumsikan ideal (tidak ada energi yang terbuang dalam proses konversi listrik). Kapasitas input konverter diasumsikan sama dengan output dari photovoltaic, wind turbine, generator diesel, dan baterai.
- Pada model Petri Net yang dibuat, nilai 1 token disetarakan dengan 1 kW.

1.6 Sistematika Penulisan

Tulisan ini terdiri dari lima bab yang disusun secara sistematis. Bab I memuat bagian pendahuluan mencakup latar belakang masalah, perumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah, dan sistematika penulisan. Bab II dari tulisan ini menguraikan tinjauan pustaka yang memuat teori dasar mengenai Pembangkit Listrik Tenaga Hibrida (PLTH), teori tentang Petri Net, dan teori tentang program GPenSIM. Bab III memuat prosedur pemodelan sistem PLTH serta prosedur simulasi model yang dibuat. Bab IV memuat hasil simulasi dari model yang dibuat beserta analisisnya. Sedangkan bab V memuat kesimpulan dari penelitian yang dibuat.



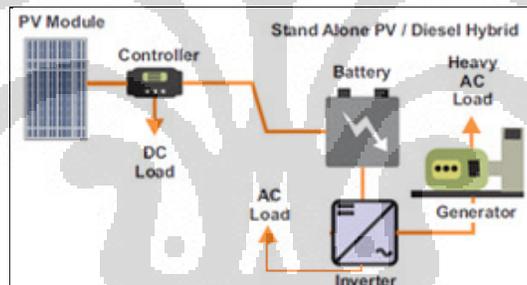
BAB 2

TINJAUAN PUSTAKA

1.7 Pembangkit Listrik Tenaga Hibrida (PLTH)

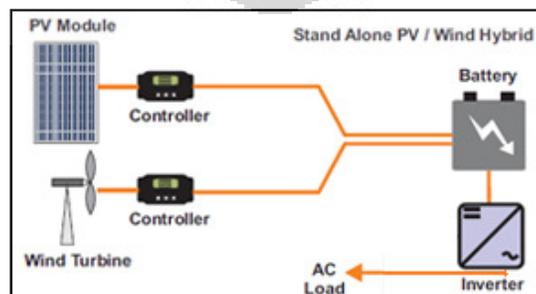
Pembangkit Listrik Tenaga Hibrida (PLTH) adalah sistem pembangkit yang terdiri dari beberapa jenis pembangkit listrik yang terintegrasi. Sistem PLTH bisa berupa gabungan antara pembangkit listrik dari sumber energi terbarukan (*renewable energy*) dan pembangkit listrik dari sumber energi konvensional atau tidak terbarukan (*nonrenewable energy*) atau gabungan antar pembangkit listrik terbarukan^[6,8].

Sistem PLTH dengan kombinasi antara pembangkit listrik energi terbarukan dan energi tidak terbarukan antara lain berupa perpaduan antara pembangkit listrik tenaga surya dan pembangkit listrik tenaga diesel seperti yang diperlihatkan pada Gambar 2.1 di bawah ini.



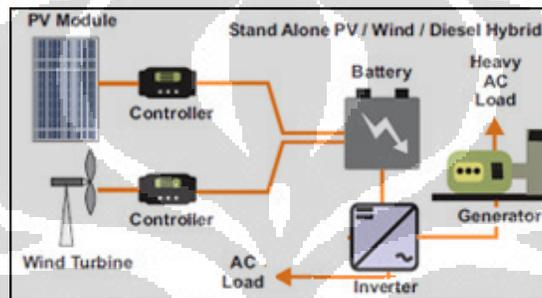
Gambar 2.1 Kombinasi PV dan diesel^[9]

Sedangkan kombinasi antar pembangkit listrik energi terbarukan antara lain berupa gabungan pembangkit listrik tenaga surya dan tenaga angin dan perpaduan photovoltaic dan fuel cell^[8,9]. Gabungan pembangkit listrik tenaga surya dan tenaga angin dapat dilihat pada Gambar 2.2 berikut:



Gambar 2.2 Kombinasi PV dan wind turbin (WT)^[9]

Pada sistem *stand-alone* (tidak terhubung ke jala-jala), biasanya digunakan energi terbarukan sebagai penghasil energi listriknya, seperti energi matahari atau energi angin. Namun demikian, penggunaan energi terbarukan masih memiliki kendala berupa ketersediaan yang masih sangat rendah. Oleh karena itu, sistem PLTH berupa perpaduan antara energi listrik tidak terbarukan dan beberapa sumber energi terbarukan menjadi solusinya^[9]. Pada Gambar 2.3 di bawah ini diperlihatkan perpaduan photovoltaic (PV), wind turbine (WT), dan generator diesel.



Gambar 2.3 Kombinasi PV, WT, dan diesel pada sistem *stand-alone*^[9]

Pembangkit listrik tenaga hibrida sudah lama dikembangkan di Indonesia. Pada tahun 2006, pemerintah Indonesia telah merencanakan pembangunan PLTH berupa gabungan sel surya (*solar cell*) dengan generator diesel. PLTH ini telah dibangun di 25 lokasi yang tersebar di berbagai wilayah di Indonesia^[9]. Pada tahun 2010, pemerintah Indonesia membangun proyek percontohan pengembangan energi hibrida terbesar di Indonesia di perkampungan nelayan Pantai Pandansimo Kabupaten Bantul, Yogyakarta. PLTH di Pandansimo tersebut dibangun dengan memanfaatkan energi kincir angin dan solar panel^[6].

Pada proses penyaluran daya listrik dari sumber-sumber daya listrik hingga ke beban dikenal adanya tiga macam strategi, yaitu^[13]:

- Strategi load-following
- Strategi cycle-charging
- Strategi battery state of charge

Pada strategi *load-following*, suplai daya listrik dari sumber-sumber energi baru dan terbarukan (photovoltaic dan wind turbine) lebih diutamakan daripada suplai dari generator diesel. Selain itu, hanya daya listrik dari photovoltaic dan wind turbine

yang disimpan ke baterai, sedangkan dari generator diesel tidak disimpan ke baterai. Pada strategi *cycle-charging*, suplai daya listrik dari generator diesel lebih diutamakan daripada suplai dari sumber-sumber energi baru dan terbarukan (photovoltaic dan wind turbine). Oleh karena itu, kelebihan daya listrik dari generator diesel juga disimpan ke baterai seperti halnya kelebihan daya listrik dari photovoltaic dan wind turbine. Pada strategi *battery state of charge*, daya listrik yang tersimpan dalam baterai diatur batas minimumnya. Misalnya, kapasitas minimum baterai disetting pada nilai 70%. Artinya, baterai dipakai menyuplai listrik jika tersimpan daya listrik minimal 70% dari kapasitas maksimumnya.

2.1.1 Photovoltaic

Radiasi surya dapat dimanfaatkan menjadi sumber energi listrik melalui sel surya. Sel surya terbuat dari potongan silikon yang sangat kecil dengan dilapisi bahan kimia khusus untuk membentuk dasar dari sel surya. Tiap sel surya biasanya menghasilkan tegangan 0,5 volt. Sel surya merupakan elemen aktif (semikonduktor) yang memanfaatkan efek photovoltaik untuk mengubah energi surya menjadi energi listrik. Untuk mendapatkan daya yang cukup besar diperlukan banyak sel surya. Biasanya sel-sel surya itu sudah disusun sehingga berbentuk panel, dan dinamakan panel photovoltaic (pv) seperti yang ditunjukkan pada Gambar 2.4.

Photovoltaic mengubah radiasi surya dalam satuan kWh/m²/hari menjadi daya listrik berdasarkan rumus perhitungan berikut:

$$P = I \cdot A \quad (2.1)$$

dimana:

P = daya listrik yang dihasilkan (kW)

I = intensitas radiasi surya (kWh/m²/hari)

A = luas bidang photovoltaic (m²)



Gambar 2.4 Contoh panel photovoltaic

Setelah mendapatkan output dari solar sel yang berupa arus listrik dapat langsung digunakan untuk beban yang dimanfaatkan. Tetapi juga arus listrik tersebut dapat digunakan sebagai pengisian dengan cara disimpan ke dalam baterai agar dapat dipergunakan pada saat yang diperlukan. Apabila solar sel tersebut digunakan untuk penyimpanan ke baterai, maka besarnya tegangan yang dihasilkan harus di atas spesifikasi baterai tersebut. Misalnya baterai yang digunakan adalah 12 Volt, maka tegangan yang dihasilkan solar sel harus di atas 12 Volt untuk dapat melakukan pengisian.

2.1.2 Turbin Angin (Wind Turbine)

Energi angin merupakan salah satu potensi energi terbarukan yang dapat memberikan kontribusi signifikan terhadap kebutuhan energi listrik, khususnya wilayah terpencil. Energi angin dapat dimanfaatkan menjadi energi listrik dengan turbin angin (wind turbine) seperti yang dapat dilihat pada Gambar 2.5.



Gambar 2.5 Contoh wind turbine

Turbin angin (wind turbine) mengubah energi angin menjadi daya listrik berdasarkan rumus perhitungan berikut^[12]:

$$P = k \cdot A \cdot v^3 \quad (2.2)$$

dimana:

P = daya listrik yang dihasilkan (kW)

k = konstanta karena faktor gesekan dan efisiensi sistem = $1,37 \cdot 10^{-5}$

A = luas penampang sudu-sudu turbin (m^2)

v = kecepatan angin (km/jam)

2.1.3 Generator Diesel

Pembangkit listrik tenaga diesel (generator diesel) merupakan sumber energi listrik konvensional. Generator diesel mengubah energi hasil pembakaran bahan bakar solar menjadi energi listrik, sehingga pembangkit listrik tenaga diesel biasa juga disebut sebagai sumber energi fosil.

Dalam mengoptimalkan kinerja generator diesel dalam sistem PLTH, perlu diperhatikan karakteristik operasi diesel terutama yang berkaitan dengan pemakaian bahan bakar solar. Efisiensi penggunaan bahan bakar solar pada generator diesel dilihat dari karakteristik Specified Fuel Consumption (SFC) dari kerja diesel tersebut.

1.8 Petri Net

Petri Net adalah salah satu metode pemodelan untuk sistem kejadian diskrit (*discrete event systems*). Metode ini menggunakan kombinasi antara model grafik dan aljabar^[4]. Sebagai model grafik, Petri Net dianggap sebagai model visual yang unggul karena adanya mekanisme transisi kondisi sistem (*system state*) secara visual yang dinyatakan dengan *token* yang bergerak. Adanya pergerakan *token* ini menjadi keunggulan tersendiri dibandingkan dengan, misalnya, metode flowchart, diagram blok, atau topologi jaringan^[4]. Adanya mekanisme pergerakan token ini juga merupakan kelebihan metode Petri Net dalam hal analisis suatu sistem dibandingkan dengan analisis sistem dengan menggunakan sistem digital (aljabar boolean).

Petri Net pertama kali diperkenalkan oleh **Carl Adam Petri** dalam disertasi doktornya yang berjudul '*Communication with automata*' pada tahun 1962. Disertasi tersebut menjelaskan tentang hubungan sebab-akibat pada suatu sistem komputer. Disertasi tersebut kemudian dikembangkan secara mendalam oleh para ahli lainnya untuk mendapatkan teori dan aplikasi dari Petri Net sebagai alat untuk mempelajari suatu sistem kejadian diskrit secara umum^[3].

2.2.1 Struktur Petri Net

Secara simbolik, struktur Petri Net (PN) dapat dibentuk oleh lima unsur utama (5-tuple) yang dapat ditulis sebagai^[4]:

$$PN = (P, T, I, O, M_0) \quad (2.3)$$

Keterangan:

$P = \{p_1, p_2, p_3, \dots, p_n\}$	adalah himpunan terbatas dari <i>place</i>
$T = \{t_1, t_2, t_3, \dots, t_n\}$	adalah himpunan terbatas dari <i>transisi</i> .
	Berlaku hubungan: $P \cup T \neq \emptyset$ dan $P \cap T = \emptyset$
$I: (P \times T) \rightarrow \mathbb{N}$	menyatakan fungsi <i>input</i> yang mendefinisikan tanda panah yang mengarah dari suatu <i>place</i> menuju ke suatu <i>transisi</i> . \mathbb{N} adalah bilangan integer bukan negatif
$O: (P \times T) \rightarrow \mathbb{N}$	menyatakan fungsi <i>output</i> yang

M_0

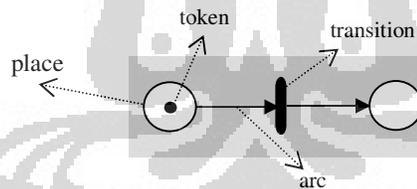
mendefinisikan tanda panah yang mengarah dari suatu transisi menuju ke suatu place adalah *marking awal (initial marking)* dari semua place

2.2.2 Penggambaran Struktur PN

Sebuah struktur Petri Net (PN) terdiri atas empat komponen, yaitu^[4]:

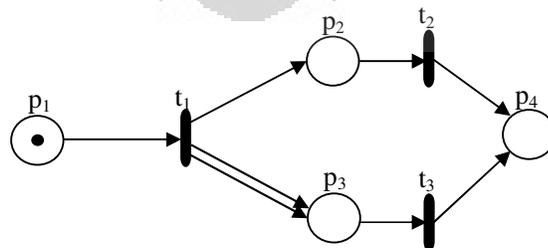
- *Place* (P) atau tempat yang digambarkan dengan lingkaran
- *Transition* (T) atau transisi yang digambarkan dengan persegi panjang atau bar
- *Token* yang digambarkan dengan titik atau dot hitam. Pada eksekusi PN, token ini akan berpindah dari suatu place ke place lainnya. Perpindahan token ini menunjukkan dinamisasi/perubahan kondisi sistem.
- *Arc* yang digunakan sebagai penghubung dari place menuju ke transisi atau sebaliknya, dari transisi menuju ke place. Arc memiliki arah dan bobot.

Struktur dasar Petri Net dapat dilihat pada Gambar 2.7 berikut:



Gambar 2.7 Struktur dasar Petri Net

Berikut ini akan disajikan sebuah contoh struktur PN:



Gambar 2.8 Contoh struktur Petri Net

Pada Gambar 2.8 di atas, diperoleh:

$$P = \{p_1, p_2, p_3, p_4\};$$

$$T = \{t_1, t_2, t_3\};$$

$$M_0 = (1,0,0,0,0); \text{ } p_1 \text{ memiliki 1 token}$$

$$I(p_1xt_1) = 1; \quad I(p_2xt_2) = 1; \quad I(p_3xt_3) = 1;$$

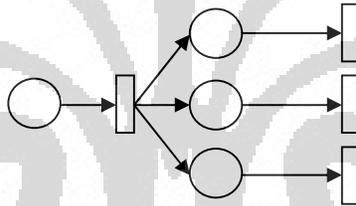
$$O(p_2xt_1) = 1; \quad O(p_3xt_1) = 1; \quad O(p_4xt_2) = 1; \quad O(p_4xt_3) = 1;$$

2.2.3 Struktur Proses PN

Petri Net (PN) dapat dipakai untuk menggambarkan beberapa proses yang terintegrasi. Proses-proses tersebut adalah sebagai berikut^[3]:

a. Proses tersebar

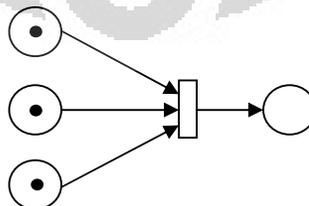
Model PN proses tersebar adalah proses digunakannya suatu proses pada beberapa proses berikutnya seperti ditampilkan pada Gambar 2.9.



Gambar 2.9 Proses tersebar pada model PN

b. Proses sinkronisasi

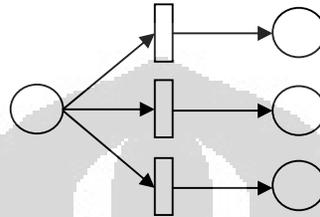
Model PN proses sinkronisasi adalah proses penyatuan kembali suatu proses dari beberapa proses yang terjadi sebelumnya seperti ditampilkan pada Gambar 2.10.



Gambar 2.10 Proses sinkronisasi pada model PN

c. Proses konflik

Model PN proses konflik adalah susunan beberapa proses yang mungkin untuk dilaksanakan, tetapi hanya satu proses saja yang dapat dilaksanakan, seperti ditampilkan pada Gambar 2.11.



Gambar 2.11 Proses konflik pada model PN

d. Proses berurutan

Model PN proses berurutan adalah susunan proses dimana proses berikutnya terjadi setelah selesainya proses sebelumnya sebagaimana ditampilkan pada Gambar 2.12.



Gambar 2.12 Proses berurutan pada model PN

2.2.4 Eksekusi Petri Net

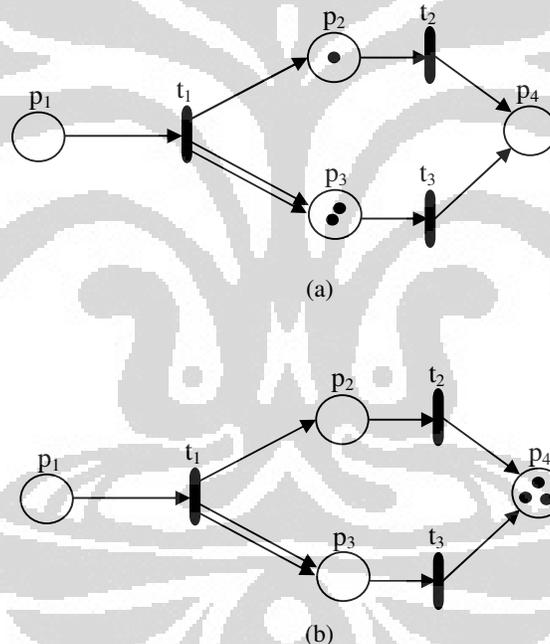
Suatu struktur Petri Net dapat dieksekusi jika transisi yang ada di dalam struktur itu yang mengalami proses firing. Proses *firing* pada sebuah transisi dapat terjadi bilamana transisi tersebut dalam kondisi *enabled*. Aturan *enabled* dan *proses firing* dari transisi dapat dijelaskan sebagai berikut^[4]:

- *Aturan enabled*; suatu transisi t dikatakan *enabled* jika input place dari transisi t mempunyai jumlah token paling sedikit sama dengan bobot dari arc (tanda panah) yang menuju ke transisi t tersebut.
- *Proses firing*; firing dari suatu transisi t yang *enabled* merupakan proses perpindahan token dari input place sebanyak bobot arc yang menuju transisi t

tersebut, kemudian didepositkan ke output place sebanyak bobot arc yang keluar dari transisi t menuju output place tersebut.

Pada contoh struktur PN Gambar 2.5 di atas, jumlah token p_1 sama dengan bobot arc dari p_1 menuju t_1 . Kondisi ini mengakibatkan transisi t_1 *enabled*. Jika transisi t_1 *firing* (dieksekusi), maka terjadi perpindahan token ke p_2 dan p_3 seperti diperlihatkan pada Gambar 2.13a berikut.

Selanjutnya, transisi t_2 dan t_3 memenuhi aturan *enabled*. Karena itu, t_2 dan t_3 mengalami proses *firing*, sehingga token kembali berpindah dari p_2 ke p_4 dan dari p_3 ke p_4 sesuai bobot arc masing-masing output dari transisi tersebut, seperti yang nampak pada Gambar 2.13b.



Gambar 2.13 Perpindahan token struktur PN

2.2.5 Properti Petri Net^[3]

Petri Net memiliki kehandalan dalam memodelkan beraneka ragam jenis sistem, dimana dapat digambarkan dengan baik tiap interaksi antara berbagai aksi yang mungkin terjadi. Setelah sebuah sistem dapat dimodelkan dengan Petri Net, langkah berikutnya adalah menganalisis sistem yang dimodelkan tersebut. Dengan

analisis ini diharapkan dapat diketahui lebih dalam lagi tentang tingkah laku sistem dan dapat dirancang tindakan yang tepat untuk memperbaiki kinerja sistem.

a. Reachability

Terhadap suatu model Petri Net akan timbul sebuah pertanyaan: apakah mungkin keadaan tertentu yang diwakili oleh suatu marking dapat dicapai dari initial marking (keadaan awal) yang sudah ditentukan? Hal ini sangat penting untuk mengetahui apakah ada suatu urutan firing yang dapat mencapai keadaan yang diinginkan dari keadaan awal tertentu. Dari sini muncul konsep yang disebut *reachability*. Terhadap suatu struktur Petri Net, suatu marking M_i dikatakan *reachable* dari M_0 bila urutan firing yang terjadi dapat menghasilkan kondisi seperti yang diinginkan, yaitu M_i .

Himpunan dari semua marking yang dapat dicapai dari kondisi awal M_0 disebut sebagai *himpunan reachability (reachability set)* yang dilambangkan dengan $\mathfrak{R}(M_0)$. Kumpulan firing yang berurutan dari beberapa transition untuk mencapai suatu marking baru yang diinginkan disebut dengan *firing sequence* (urutan firing).

Analisis sifat reachability dari struktur Petri Net model sistem PLTH diawali dengan mengubah nama-nama place dalam struktur tersebut menjadi p_1, p_2, \dots dan nama-nama transition menjadi t_1, t_2, \dots . Setelah itu, kondisi awal (initial marking) dan kondisi yang diinginkan pada akhir proses ditentukan.

Kondisi awal dinyatakan dengan M_0 yang dirumuskan sebagai:

$$M_0 = [x(p_1), x(p_2), x(p_3), \dots, x(p_n)] \quad (2.4)$$

dimana:

$$x(p_n) = \text{jumlah token awal pada place } p_n \text{ (n=banyaknya place)}$$

Sedangkan kondisi yang diinginkan dinyatakan dengan M_i yang dirumuskan sebagai:

$$M_i = [x'(p_1), x'(p_2), x'(p_3), \dots, x'(p_n)] \quad (2.5)$$

dimana:

$$x'(p_n) = \text{jumlah token yang diinginkan pada place } p_n \text{ (n= banyaknya place)}$$

b. Bounded dan safeness

Bounded berarti terbatas. Yang terbatas adalah jumlah token pada tiap place pada semua himpunan reachability. Bila suatu Petri Net dikatakan memiliki sifat yang *k-bounded* berarti pada semua marking yang reachable oleh initial marking tidak ada satu place pun yang memiliki jumlah token melebihi k .

Kasus khusus dari boundeness adalah *safeness*. Suatu Petri Net dikatakan *safe* bila memiliki sifat k -bounded dimana nilai dari $k=1$.

c. Liveness dan deadlock

Deadlock merupakan keadaan yang dapat saja terjadi pada suatu Petri Net dimana dihasilkan satu atau beberapa transition yang tidak akan pernah dapat diaktifkan, pada saat jaringan tersebut dijalankan atau dieksekusi.

Liveness merupakan keadaan yang terjadi pada suatu Petri Net dengan sifat yang berlawanan dengan deadlock. Suatu Petri Net dikatakan memiliki sifat liveness bila tidak pernah dijumpai suatu keadaan deadlock pada saat pengoperasiannya. Suatu Petri Net dikatakan *live* untuk sebuah marking M_0 , jika untuk semua marking yang berada pada himpunan reachability $\mathcal{R}(M_0)$, pada akhirnya menjalankan semua transition yang ada pada jaringan tersebut.

d. Reversibility

Hal penting dalam sebuah sistem terutama pada sistem kontrol adalah kemampuan sistem tersebut untuk mengatasi kesalahan. Sistem tersebut diharapkan mampu mengembalikan keadaan yang salah kepada kondisi sebelumnya atau kondisi awal.

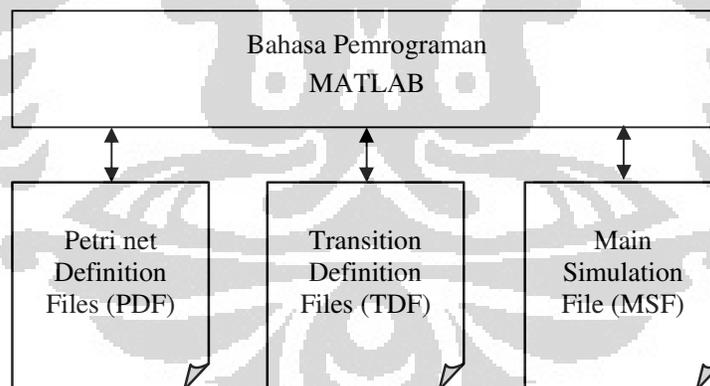
Kemampuan mengembalikan suatu marking M_i kepada marking awal M_0 disebut dengan *reversibility*. Suatu Petri Net disebut *reversible* jika marking awal dapat dicapai dari setiap marking yang lain.

1.9 GPenSIM

Ada sejumlah perangkat lunak (*software*) yang dapat digunakan untuk mensimulasikan model Petri Net (PN), baik untuk keperluan akademik maupun untuk kebutuhan komersial. Perangkat tersebut cukup bagus untuk memodelkan sistem kompleks dan besar. Salah satu perangkat lunak yang digunakan untuk memodelkan sistem kejadian diskrit (Petri Net) adalah GPenSIM (General Purpose Petri Net Simulator). GPenSIM dijalankan dalam lembar kerja MATLAB. Yang akan digunakan dalam penelitian ini adalah GPenSIM versi 4.0 (*last update*: September 2010)^[1].

2.3.1 Model dan Simulasi^[1]

Sebuah model Petri Net yang dikembangkan dengan GPenSIM terdiri dari tiga jenis file, yaitu Petri net Definition Files (PDF), Transition Definition Files (TDF), dan Main Simulation File (MSF) seperti yang ditunjukkan pada Gambar 2.14. File PDF memuat rincian statis dari model Petri Net, sedangkan MSF berisi informasi dinamisnya. Main Simulation File (MSF) adalah file yang akan dijalankan langsung oleh platform MATLAB.



Gambar 2.14 Arsitektur GPenSIM

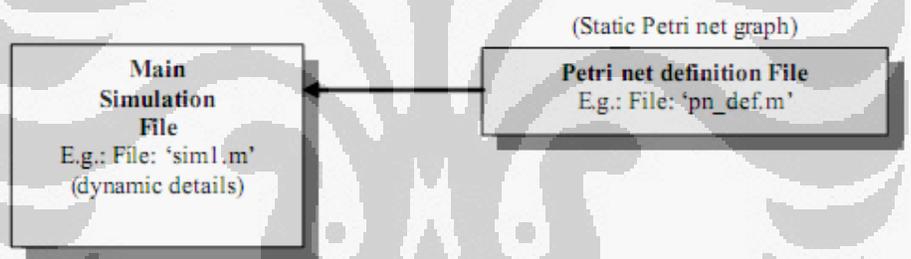
Selain MSF dan PDF, arsitektur GPenSIM juga dilengkapi dengan satu atau beberapa File Definition Transition (TDF). Sebuah TDF menentukan apakah suatu *transition* yang bersifat *enabled* dapat *firing* atau tidak. Setiap *transition* bisa memiliki file TDF tersendiri dalam model Petri Net.

2.3.2 Petri net Definition Files (PDF)

Dalam GPenSIM, definisi dari sebuah PN graph (*static details*) dituliskan dalam bagian Petri net Definition Files (PDF), terpisah dari bagian dinamis (*dynamic details*) yang dibuat di Main Simulation File (MSF) seperti yang diperlihatkan pada Gambar 2.15. Umumnya, sebuah model yang dibuat dalam GPenSIM hanya memiliki satu file PDF. Namun demikian, jika sebuah model PN dibagi ke dalam beberapa modul, maka dibuat lebih dari satu PDF, dimana setiap modul didefinisikan dengan PDF tersendiri^[1].

PDF adalah bagian statik dari sebuah model yang akan dibuat. Bagian ini terdiri dari tiga sub-bagian, yaitu^[1]:

- a) identifikasi semua place dari model PN
- b) identifikasi semua transition dari model PN
- c) penjabaran semua hubungan antara place dan transition dengan arc



Gambar 2.15 Pemisahan MSF dan PDF^[1]

Pemberian nama place dan transition dalam GPenSIM diatur sedemikian rupa, sehingga hanya 10 karakter pertama dari nama-nama itu yang disimpan^[1]. Misalnya, nama place “*pTurbin_Angin_1*” dan place “*pTurbin_Angin_2*” dianggap sama karena place yang pertama hanya diambil 10 karakter menjadi “*pTurbin_An*” sedangkan nama place yang kedua juga hanya diambil 10 karakter menjadi “*pTurbin_An*”. Hal ini menyebabkan simulasi model PN yang tidak sesuai yang kita harapkan.

2.3.3 Transition Definition File (TDF)

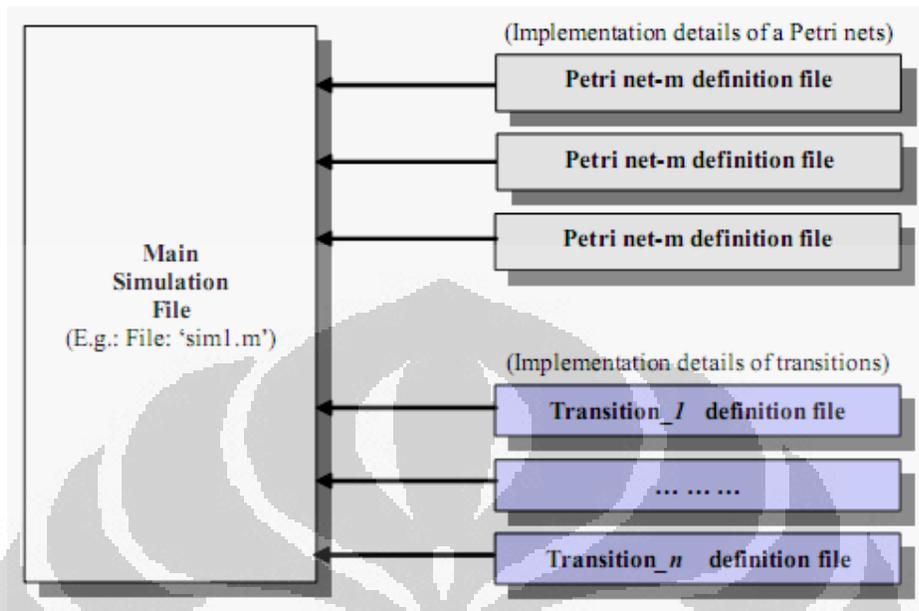
Transition Definition File (TDF) adalah bagian dalam arsitektur GPenSIM yang dipakai untuk mengatur terjadinya firing dari beberapa transition yang enabled. Kedudukan TDF di antara PDF dan MSF dapat dilihat pada Gambar 2.16.

Setiap transition dapat memiliki file TDF tersendiri dalam model Petri Net. Oleh karena itu, sebuah model PN bisa memuat satu atau lebih TDF seperti ditunjukkan pada Gambar 2.13. File TDF ini diklasifikasikan menjadi dua jenis, yaitu TDF_PRE dan TDF_POST. File TDF_PRE dijalankan sebelum terjadinya firing dari sebuah transition, sedangkan TDF_POST file dijalankan setelah terjadinya firing transition tersebut^[1].

Sesuai dengan teori Petri Net, sebuah transisi bisa mengalami firing (*enabled transisi*) ketika ada cukup token pada input place. Namun, dalam situasi yang sebenarnya, sebuah *event* yang mewakili transition dapat memiliki aturan tambahan untuk mengalami firing. Misalnya, *event-2* memiliki prioritas lebih tinggi daripada *event-1*, maka *event-2* diperbolehkan untuk firing meskipun *event-1* dan *event-2* sama-sama *enabled firing*. Dalam literatur GPenSIM, aturan tambahan disebut "*kondisi user-defined*"^[1].

Kondisi user-defined untuk firing dari sebuah transition disimpan dalam file TDF_PRE. Setelah sebuah transition mengalami firing, mungkin ada beberapa pencatatan yang perlu dilakukan. Hal ini dapat dikodekan ke dalam sebuah file TDF_POST^[1].

Nama dari file TDF harus mengikuti kaidah penamaan, karena nama file itu akan dipilih dan berlaku secara otomatis. Misalnya, TDF_PRE untuk transition 'trans1' harus diberi nama 'trans1_pre.m'. Dengan cara yang sama, TDF_POST untuk transition 'trans1' harus dinamai 'trans1_post.m'^[1].



Gambar 2.16 Kedudukan TDF di antara MSF dan PDF^[1]

2.3.4 Main Simulation File (MSF)^[1]

Bagian Main Simulation File (MSF) memuat informasi dinamis dari sebuah model PN. Informasi dinamis tersebut antara lain berupa token awal dari satu atau beberapa place, waktu terjadinya firing dari satu atau beberapa transition (*firing time of transition*), dan tampilan grafik dari model PN yang dibuat.

Selain informasi dinamis, MSF juga memuat *function* untuk melakukan proses simulasi dan beberapa *function* untuk menampilkan hasil simulasi tersebut. Proses simulasi akan dilakukan jika bagian statis dan bagian dinamis telah dibaca. Function untuk membaca bagian statis adalah “*petrinetgraph(nama_fungsi_PDF)*”. Nama_fungsi_PDF ditetapkan pada bagian PDF. Nama_fungsi_PDF tersebut sekaligus menjadi nama file PDF saat disimpan. Beberapa function yang digunakan untuk menampilkan hasil simulasinya terdiri dari:

- function “*gpensim*”
- function “*print_statespace*”
- function “*plotp*”

Function “*gpensim*” digunakan untuk melakukan proses transfer token dari satu place ke place lainnya berdasarkan struktur PN yang telah dibuat pada bagian PDF.

Function “print_statespace” digunakan untuk menampilkan data selama proses simulasi berlangsung, seperti transition yang enabled dan yang firing pada state tertentu serta komposisi token pada semua place pada tiap state. Function “plotp” digunakan untuk menampilkan hasil simulasi dalam bentuk grafik.

2.3.5 Hasil Simulasi

Hasil simulasi dari suatu model Petri Net pada GPenSIM dapat dilihat dengan menuliskan perintah tertentu. Perintah tersebut dapat dituliskan pada bagian MSF atau pada bagian Command Window MATLAB. Bila dituliskan pada bagian MSF, maka perintah tersebut langsung dieksekusi pada saat dilakukan proses “running”.

Secara umum, hasil simulasi dapat berbentuk *output grafis* dan *output print statespace*. Output berbentuk grafis menginformasikan data berbentuk gambar memuat jumlah token per tahapan proses atau per satuan waktu pada tiap place dari model Petri Net. Hasil berbentuk print statespace memperlihatkan urutan tahapan simulasi dan hasilnya, berupa transition yang enabled, transition yang firing, dan jumlah token tiap place. Untuk melihat hasil berbentuk grafikal, maka dituliskan perintah berikut^[1]:

```
plotp (Sim_Results, {'namaPlace-1', 'namaPlace-2', ..., 'namaPlace-n'});
```

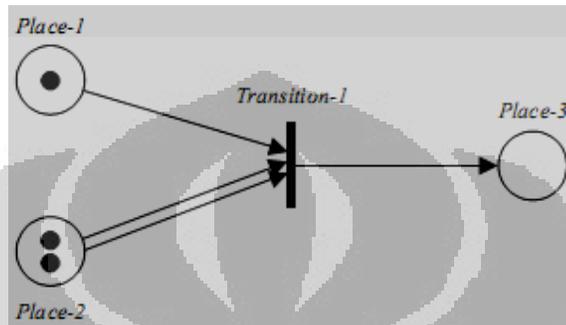
Sedangkan untuk melihat hasil berbentuk print statespace, dituliskan dengan perintah berikut^[1]:

```
print_statespace (Sim_Results);
```

Selain menampilkan hasil simulasi dalam bentuk grafis dan print statespace, GPenSIM juga dapat menampilkan hasil perhitungan yang dituliskan dalam bagian MSF. Misalnya, bila diperlukan untuk menampilkan jumlah token tiap place pada akhir simulasi kemudian menghitung prosentase besarnya token suatu place dibandingkan jumlah token semua place, maka rumus perhitungannya bisa dituliskan pada bagian MSF.

2.3.6 Contoh Sederhana^[1]

Untuk memudahkan pemahaman tentang simulasi model dengan GPenSIM, maka berikut ini akan diberikan sebuah contoh model PN sederhana sebagaimana diperlihatkan pada Gambar 2.17.



Gambar 2.17 Model PN contoh sederhana

Model PN pada Gambar 2.17 di atas terdiri dari 3 place, 1 transition, dan 4 arc. Pendefinisian place, transition, dan arc tersebut diatur dalam bagian statis. Bagian statis model PN tersebut dibuat dalam PDF seperti pada Gambar 2.18 berikut:

```
% Example-01: A Simple Example
% file: 'simple_pn_def.m'
% this file defines the simple petri net graph
function [PN_name, set_of_places, set_of_trans, set_of_arcs] ...
    = simple_pn_def(global_info)

PN_name = 'A Simple Petri Net';
set_of_places = {'Place-1', 'Place-2', 'Place-3'};
set_of_trans = {'Transition-1'};
set_of_arcs = {'Place-1', 'Transition-1', 1, ...
    'Place-2', 'Transition-1', 2, ...
    'Transition-1', 'Place-3', 1};
```

Gambar 2.18 PDF untuk model PN contoh sederhana

Pada Gambar 2.18 di atas nampak bahwa nama_fungsi_PDF yang diberikan adalah “*simple_pn_def*”. Nama_fungsi_PDF tersebut akan dipanggil pada bagian MSF. Proses pemanggilannya dapat dituliskan dengan perintah berikut:

pn = petrinetgraph ('simple_pn_def');

Perintah tersebut ditempatkan sebelum penulisan bagian dinamis dari model PN sebagaimana yang terdapat pada bagian MSF yang nampak pada Gambar 2.19.

```

pn = petrinetgraph('simple_pn_def');
dynamic_info.initial_markings = {'Place-1',1, 'Place-2',2};

Sim_Results = gpensim(pn, dynamic_info);
print_statespace(Sim_Results);
plotp(Sim_Results, {'Place-1','Place-2','Place-3'});

```

Gambar 2.19 MSF untuk model PN contoh sederhana

Seperti yang terlihat pada Gambar 2.19 di atas, perintah untuk melakukan proses simulasi dapat dituliskan dengan:

```

Sim_Results = gpensim (pn, dynamic_info);

```

Hasil simulasi yang dituliskan pada bagian MSF pada Gambar 2.16 di atas berbentuk print statespace. Hasilnya nampak pada Gambar 2.20 berikut:

```

Time: 0
State:0 (Initial State)
Place-1  Place-2  Place-3
  1      2      0
At time: 0
  Enabled transtions are:
  t1
At time: 0
  Firing transtions are:
  t1

Time: 0
State: 1
Fired Transition: t1
Current State:
Place-1  Place-2  Place-3
  0      0      1
At time: 0
  Enabled transtions are:

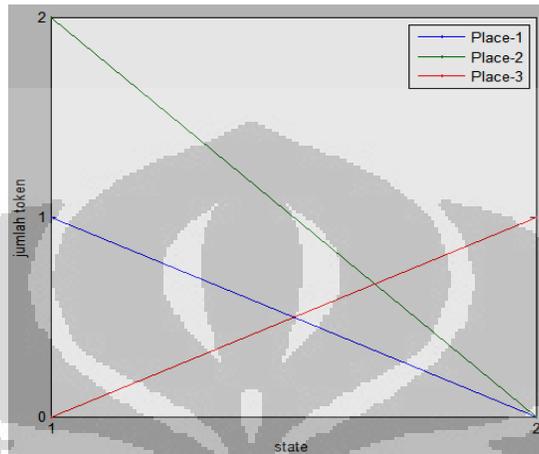
```

Gambar 2.20 Print statespace untuk model PN contoh sederhana

Hasil simulasi print statespace di atas selanjutnya dapat dibuat dalam bentuk tabel agar lebih memudahkan dalam pengamatan data, seperti pada tabel 2.1 berikut:

Tabel 2.1 Hasil simulasi contoh sederhana

Penyajian data hasil simulasi berbentuk tabel seperti di atas dapat dilengkapi dengan tampilan data berbentuk grafis. Hasil berbentuk grafis dapat dilihat pada Gambar 2.21 di bawah ini.



Gambar 2.21 Plot untuk model PN contoh sederhana

BAB 3

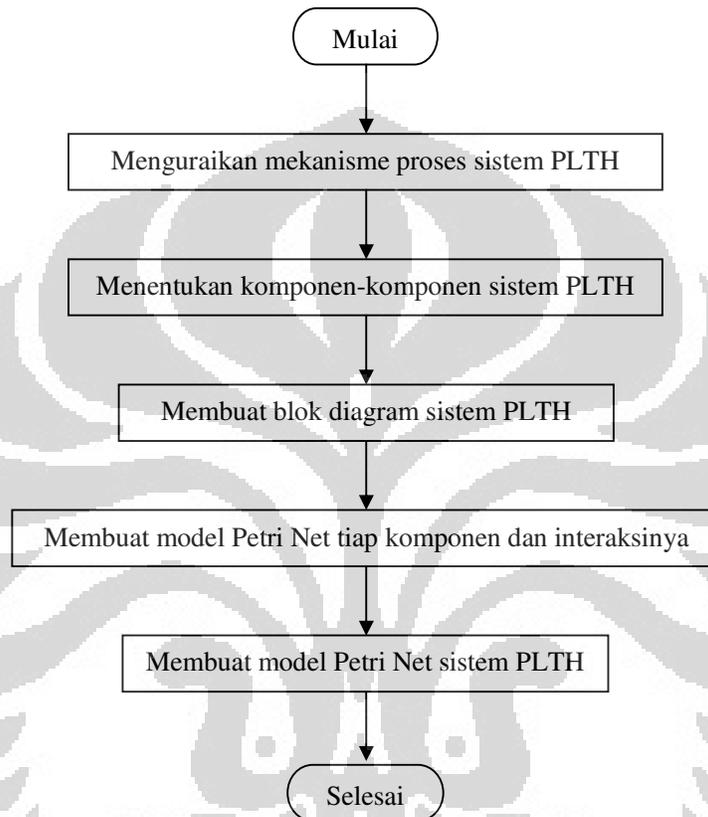
PEMBUATAN MODEL PETRI NET DAN MODEL SIMULASI SISTEM PEMBANGKIT LISTRIK TENAGA HIBRIDA (PLTH)

1.10 Prosedur Pemodelan

Secara umum langkah-langkah untuk membangun struktur petri net dilakukan dengan tahapan berikut ini:

- Langkah 1: *Menguraikan mekanisme proses dari sistem PLTH*
Langkah ini menentukan aliran energi dari input proses menuju outputnya. Langkah ini akan memperjelas proses yang terjadi dan komponen-komponen yang terlibat agar output sesuai dengan yang diharapkan.
- Langkah 2: *Menentukan komponen-komponen sistem PLTH*
Setelah proses diidentifikasi, maka langkah berikutnya adalah menentukan komponen-komponen sistem yang menjalankan proses tersebut. Pada bagian ini akan diperjelas komponen-komponen yang memberikan pengaruh besar pada proses.
- Langkah 3: *Membuat blok diagram sistem PLTH*
Pada langkah ini dibuat diagram yang menggambarkan proses yang dapat terjadi pada sistem PLTH yang akan dimodelkan. Semua kemungkinan aliran energi diperlihatkan dalam diagram ini untuk memastikan prosesnya dapat menghasilkan output yang dikehendaki dari input yang diberikan.
- Langkah 4: *Membuat model petri net tiap komponen dan interaksinya*
Setelah blok diagram ditentukan, maka langkah selanjutnya adalah membuat model petri net dari tiap bagian/komponen sistem PLTH.
- Langkah 5: *Membuat model petri net sistem PLTH*
Langkah berikutnya adalah membangun model petri net termasuk interaksi dengan lingkungannya, sehingga didapatkan model keseluruhan.

Berikut ini adalah diagram alir (flowchart) yang menggambarkan prosedur di atas:



Gambar 3.1 flowchart prosedur pemodelan sistem PLTH

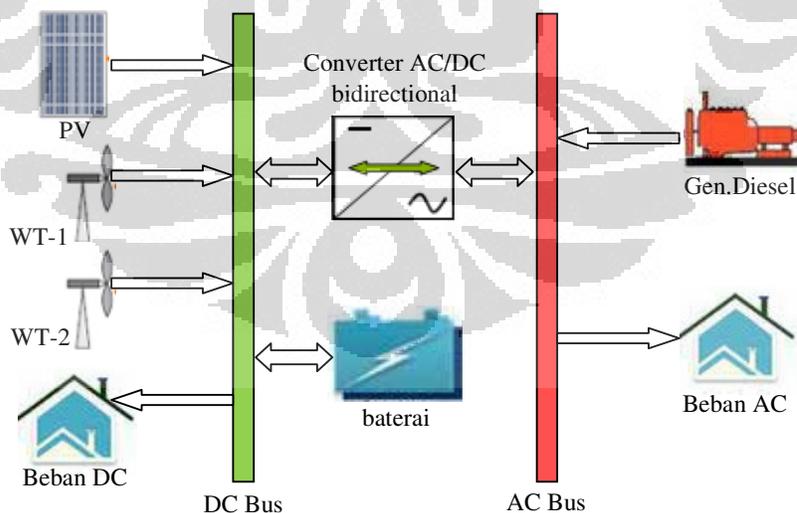
3.1.1 Mekanisme Proses Sistem PLTH

Sistem PLTH dengan perpaduan antara photovoltaic (pv), wind turbine (wt), dan generator diesel dioperasikan secara paralel untuk menyuplai energi listrik ke beban. Data potensi daya listrik dari photovoltaic diperoleh dari data besarnya radiasi sinar matahari yang dikonversi menjadi daya listrik dc, sedangkan data potensi daya listrik dari wind turbine diperoleh dari data besarnya kecepatan angin. Secara umum, mekanisme penyuplaian daya listrik dari sumber-sumber listrik ke beban dijabarkan dalam ketentuan berikut:

- Kapasitas dan jenis beban yang akan disuplai ditentukan berdasarkan perkiraan data kebutuhan beban (load profile).
- Suplai listrik lebih diprioritaskan untuk beban ac yang digunakan sebagai beban utama.
- Kelebihan daya listrik setelah disuplai ke beban akan disimpan di dalam baterai. Daya listrik baterai akan disalurkan ke beban bila pasokan dari pv, wt, dan/atau generator diesel tidak mencukupi.
- Suplai daya listrik dari pv, wt, dan baterai ke beban ac melalui konverter dc/ac, ac bus, kemudian ke beban ac. Sedangkan untuk menyuplai beban dc dilakukan melalui dc bus dan diteruskan ke beban dc.
- Suplai daya listrik dari generator diesel ke beban ac melalui ac bus, sedangkan untuk ke beban dc melalui konverter ac/dc dan dc bus.

3.1.11 Komponen-komponen Sistem PLTH

Berdasarkan uraian pada subbab 3.1.1 di atas, dapat dirumuskan komponen-komponen dari sistem PLTH yang akan dimodelkan.



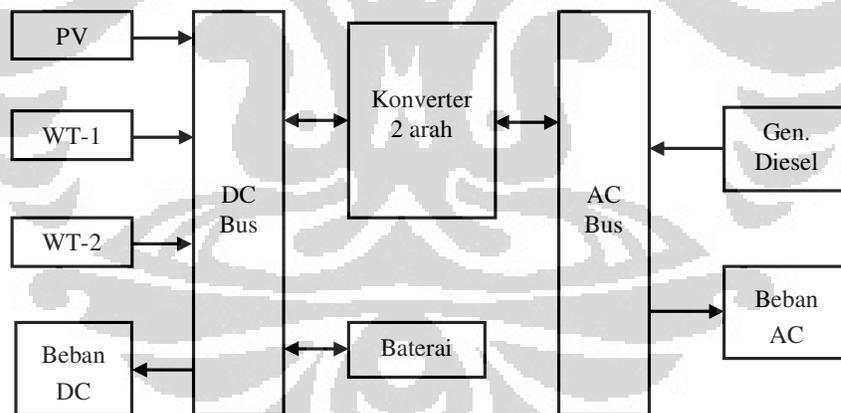
Gambar 3.2 Konfigurasi Sistem PLTH kombinasi PV,WT,DIESEL

Komponen-komponen yang dapat dilihat pada Gambar 3.2 di atas meliputi:

- a. photovoltaic (PV)
- b. wind turbine-1 (WT-1)
- c. wind turbine-2 (WT-2)
- d. generator diesel (DIS)
- e. baterai (BAT)
- f. konverter bidirectional (AC to DC / DC to AC)
- g. ac bus
- h. dc bus
- i. beban ac (ac load)
- j. beban dc (dc load)

3.1.12 Blok Diagram Model Sistem PLTH

Berikut ini adalah blok diagram dari model sistem PLTH gabungan antara photovoltaic, turbin angin, dan energi diesel:



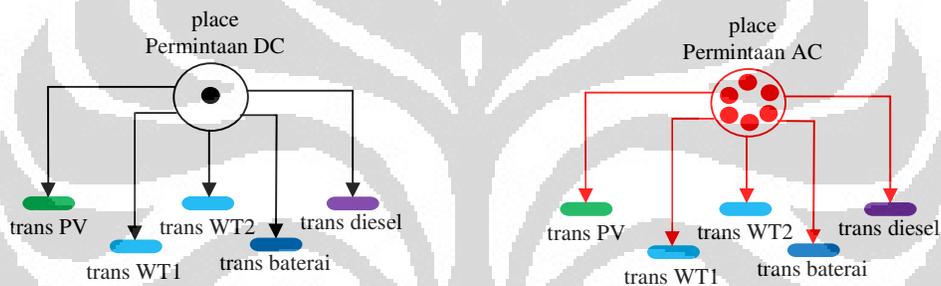
Gambar 3.3 Blok diagram model sistem PLTH

Dari blok diagram di atas, selanjutnya akan dibuat model Petri Net dari tiap komponen. Berikut ini adalah model Petri Net dari permintaan beban, photovoltaic (pv), wind turbin (wt), energy storage (baterai), inverter, ac bus, dan beban ac.

3.1.4 Model Petri Net Permintaan Beban

Permintaan beban merupakan sekumpulan data yang memuat besarnya daya listrik yang dibutuhkan oleh beban pada periode waktu tertentu. Dalam penelitian ini diasumsikan bahwa permintaan beban ac maksimum sebesar 30 kW dan beban dc maksimum sebesar 5 kW.

Pada model PN yang akan dibuat, permintaan beban akan dimodelkan sebagai sebuah *place* yang berisi sejumlah *token* dan beberapa *arc* yang menuju ke beberapa *transition*. *Token* dalam “*place* permintaan beban” mewakili jumlah daya listrik yang diminta oleh beban.



Gambar 3.4 Model PN permintaan beban

Pada Gambar 3.4 di atas terlihat model PN untuk permintaan beban. Pada penelitian ini, nilai 1 token diasumsikan setara dengan 5 kW, sehingga daya listrik sebesar 30 kW setara dengan 6 token. Dengan demikian “*place* permintaan beban ac” minimal berisi 0 *token* dan maksimal berisi 6 *token*, sedangkan “*place* permintaan beban dc” minimal berisi 0 *token* dan maksimal berisi 1 *token*.

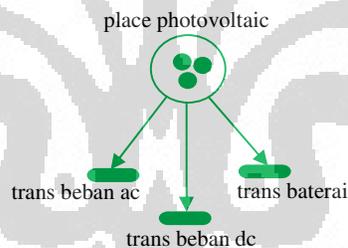
Pada Gambar 3.4 juga terlihat adanya 5 *arc* (busur) yang keluar dari “*place* permintaan beban”. Kelima *arc* tersebut menuju ke *transition* pv, wt1, wt2, baterai, dan diesel.

3.1.5 Model Petri Net Photovoltaic

Photovoltaic (pv) merupakan komponen dari sistem PLTH yang berfungsi untuk mengubah energi matahari (surya) menjadi energi listrik dc. Pada penelitian ini, daya listrik yang dihasilkan oleh photovoltaic (pv) disajikan dalam periode waktu tertentu.

Gambar 3.5 di bawah ini memperlihatkan model PN dari photovoltaic (pv). Pada model PN yang akan dibuat, photovoltaic dimodelkan sebagai sebuah place dengan 3 *transition* dan beberapa *arc*. Satu *transition* berfungsi untuk mengeksekusi daya listrik dari pv ke beban ac, satu *transition* ke beban dc, dan yang lainnya untuk ke baterai.

Pada penelitian ini, diasumsikan bahwa kapasitas maksimum dari photovoltaic adalah sebesar 15 kW, sehingga “*place pv*” minimum berisi 0 *token* dan maksimum 3 *token*.



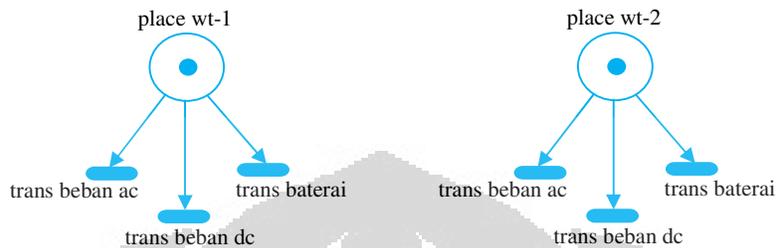
Gambar 3.5 Model PN dari photovoltaic

3.1.6 Model Petri Net Wind Turbine

Wind turbine (wt) merupakan komponen dari sistem PLTH yang berfungsi untuk mengubah energi angin menjadi energi listrik dc. Pada penelitian ini, daya listrik yang dihasilkan oleh wind turbine (wt) disajikan dalam periode waktu tertentu.

Gambar 3.6 di bawah ini memperlihatkan model PN dari wind turbine (wt1 dan wt2). Pada model PN yang akan dibuat, tiap wind turbine dimodelkan sebagai sebuah place dengan 3 *transition* dan beberapa *arc*. Satu *transition* berfungsi untuk

mengeksekusi daya listrik dari wt ke beban ac, satu transition ke beban dc, dan yang lainnya untuk ke baterai.



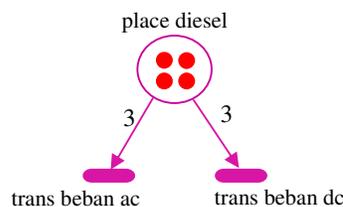
Gambar 3.6 Model PN dari wind turbine

Gambar 3.6 di atas menunjukkan model PN dari sumber listrik wind turbine (wt). Diasumsikan bahwa kapasitas maksimum dari wt sebesar 5 kW, sehingga “*place wt*” minimum berisi 0 *token* dan maksimum 1 *token*.

3.1.7 Model Petri Net Generator Diesel

Pada sistem PLTH ini, diesel difungsikan paralel dengan sumber energi lain yang lebih diprioritaskan penggunaannya (pv, wt, dan cadangan energi dari baterai). Generator diesel digunakan bilamana listrik dari pv, wt1, dan wt2 tidak mencukupi untuk memenuhi permintaan beban. Pada penelitian ini, diasumsikan bahwa diesel memiliki kapasitas maksimum sebesar 20 kW.

Pembangkit listrik tenaga diesel (generator diesel) dimodelkan dengan sebuah *place*, dua buah *transition*, dan beberapa *arc* seperti yang diperlihatkan pada Gambar 3.7. *Place* berisi sejumlah *token* yang merepresentasikan besarnya daya listrik yang dibangkitkan oleh diesel. Dengan asumsi di atas, “*place diesel*” selalu tersedia dengan 4 *token*.



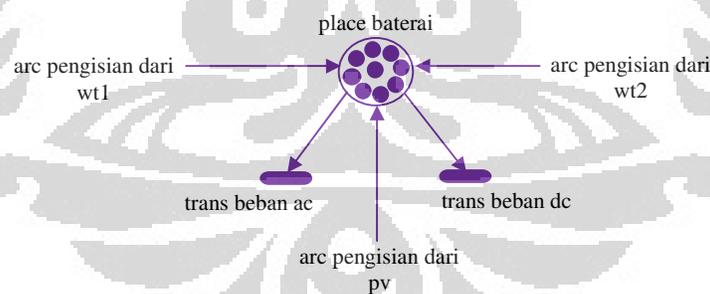
Gambar 3.7 Model PN dari generator diesel

Pada model PN untuk generator diesel yang diperlihatkan pada Gambar 3.7 di atas, tanda panah (arc) diberi bobot “3”. Hal ini dimaksudkan agar token yang keluar dari place diesel setiap kali salah satu transitionnya firing selalu berjumlah 3. Ini berarti, setiap kali listrik dari diesel terpakai, yang dikeluarkan adalah 3 token dari maksimal 4 token yang dimilikinya, atau sebesar 75% dari kapasitas maksimumnya.

3.1.8 Model Petri Net Baterai

Jika daya dari pv dan wt melebihi permintaan beban, maka kelebihan daya tersebut akan disimpan di dalam baterai. Daya yang tersimpan di dalam baterai ini pada saat yang lain akan digunakan untuk menyuplai beban bilamana listrik dari pv dan wt tidak mencukupi permintaan beban. Kapasitas baterai diasumsikan sebesar 200 kW, sehingga jumlah token maksimalnya adalah 40.

Pada Gambar 3.8 terlihat model PN untuk energy storage (baterai) yang terdiri dari sebuah place, dua buah transition, dan beberapa arc. Tiga arc diantaranya untuk pengisian baterai dari pv, wt1, dan wt2, sedangkan sisanya untuk penyuplaian daya listrik ke beban ac dan beban dc.

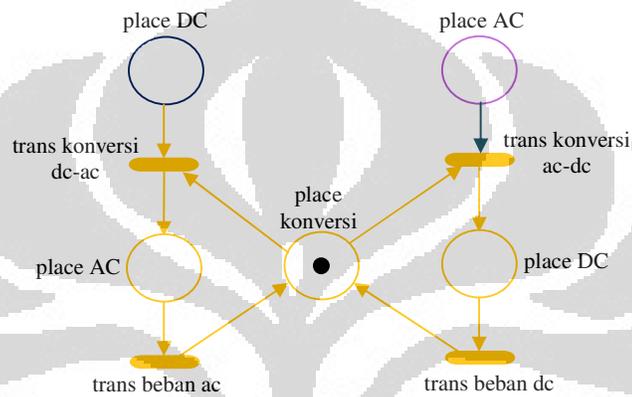


Gambar 3.8 Model PN dari baterai

3.1.9 Model Petri Net Konverter Bidirectional

Daya listrik dari sumber dc (pv, wt1, dan wt2) yang digunakan untuk menyuplai beban ac akan diteruskan ke beban ac melalui komponen konverter *dc to ac*. Sebaliknya, daya listrik dari sumber ac (generator diesel) yang digunakan untuk menyuplai beban dc akan diteruskan ke beban dc melalui komponen konverter *ac to*

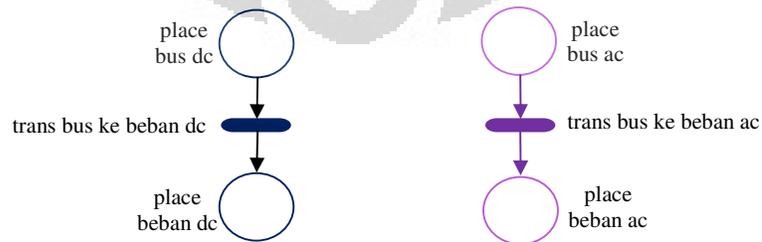
dc. Oleh karena itu, komponen konverter yang digunakan bersifat birectional (2 arah), dapat mengkonversi ac ke dc dan sebaliknya. Model PN konverter dilengkapi dengan sebuah place tambahan (*place konversi*) yang selalu berisi 1 token agar proses konversi ac ke dc tidak terjadi bersamaan dengan proses konversi dc ke ac. Model PN untuk konverter nampak pada Gambar 3.9.



Gambar 3.9 Model PN dari konverter bidirectional

3.1.10 Model Petri Net Bus dan Beban

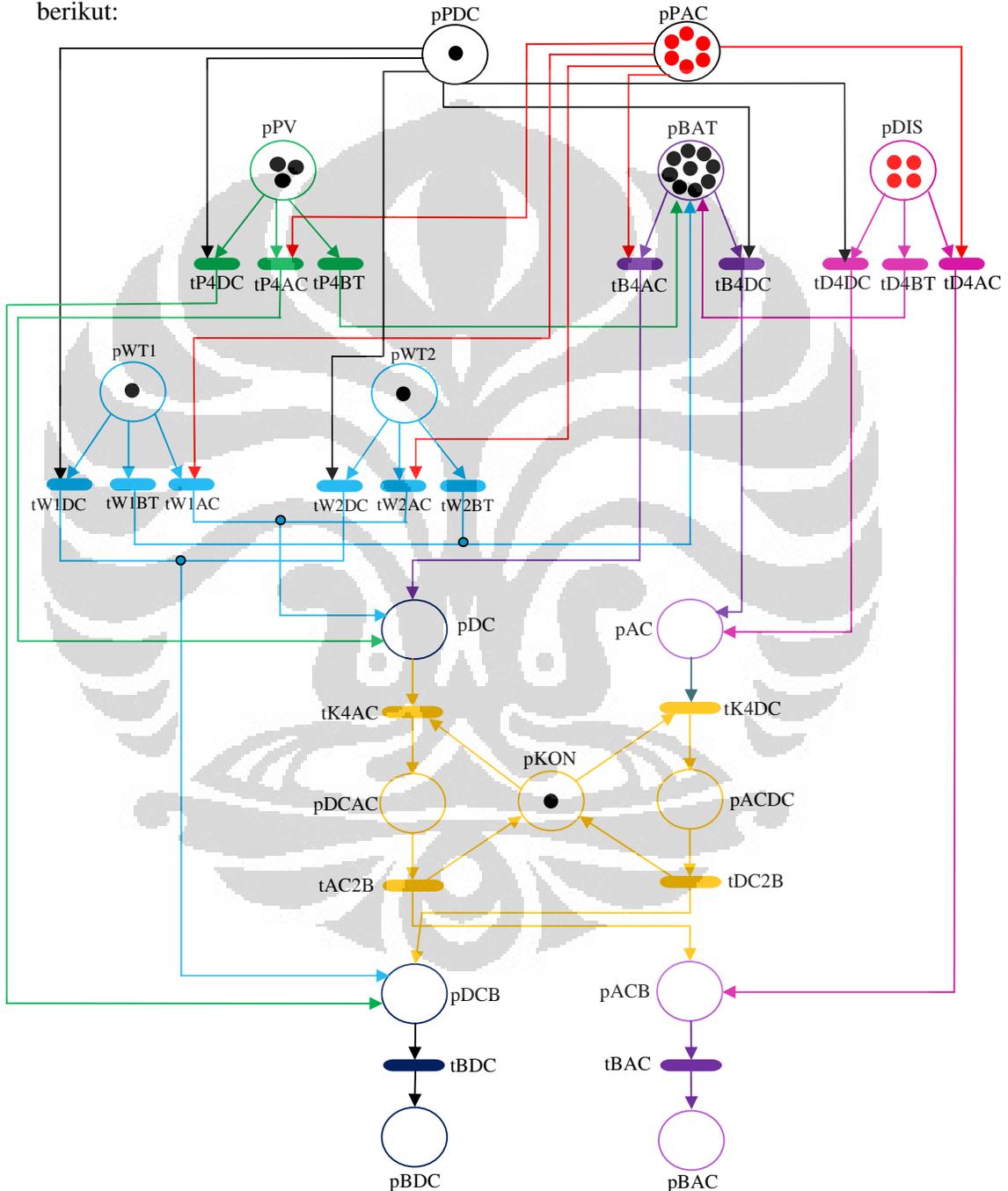
Daya listrik dari konverter diteruskan ke bus. Daya listrik ac diteruskan ke bus ac sedangkan daya listrik dc dilanjutkan ke bus dc. Dari bus ac dan bus dc kemudian diteruskan ke beban masing-masing. Model PN untuk bus dan beban merupakan struktur proses berurutan. Struktur PN untuk bus ac dan beban ac terpisah dengan struktur PN untuk bus dc dan beban dc sebagaimana diperlihatkan pada Gambar 3.10.



Gambar 3.10 Model PN dari bus dan beban (ac dan dc)

3.1.11 Struktur Petri Net Model Sistem PLTH

Berdasarkan pada ketiga macam strategi penyuplaian daya listrik yang akan disimulasikan, maka dapat dibuat struktur Petri Net model sistem PLTH sebagai berikut:



Gambar 3.11 Struktur Petri Net model sistem PLTH

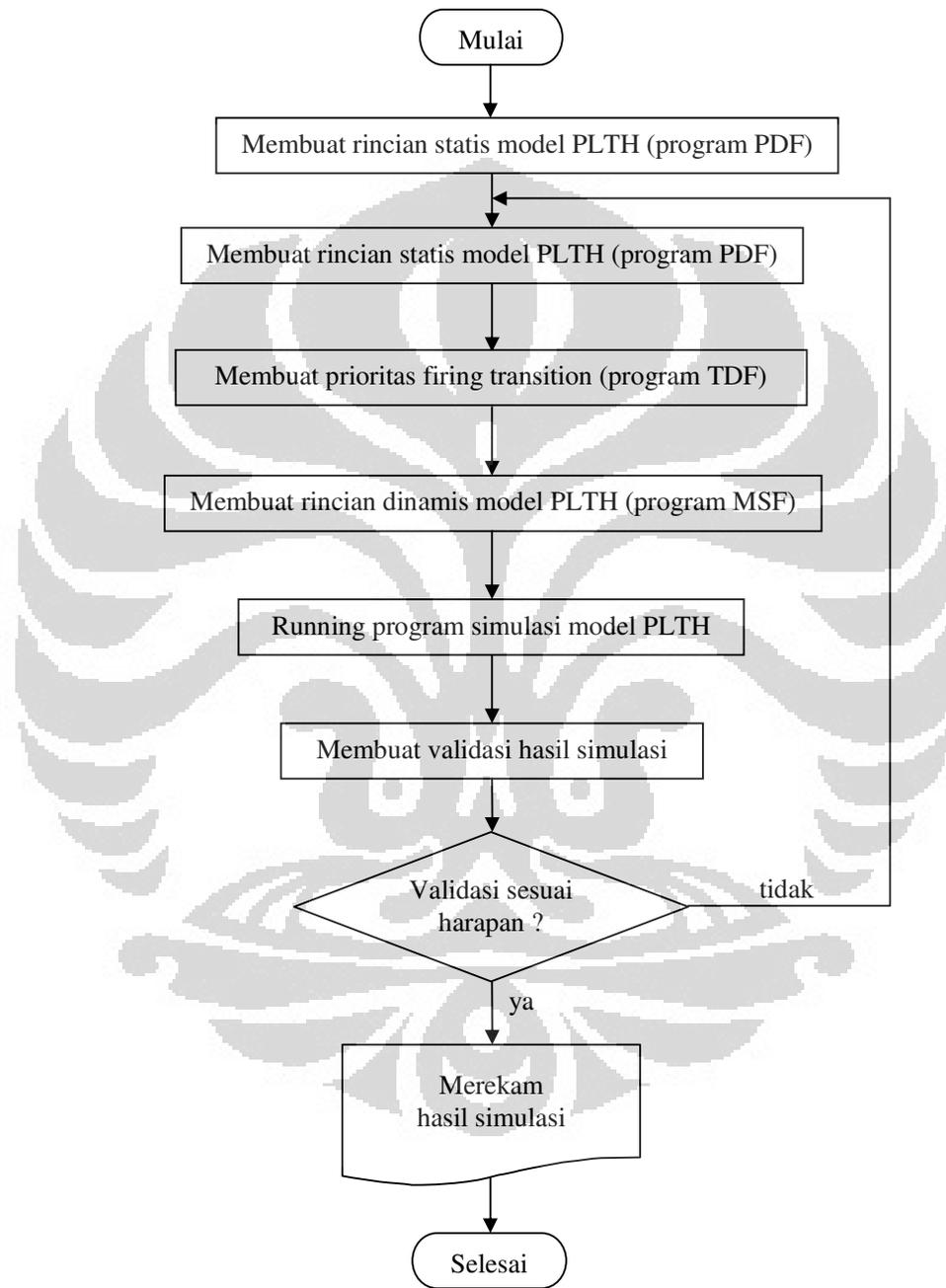
3.2 Prosedur Simulasi

Model Petri Net sistem PLTH yang telah dibuat kemudian akan disimulasikan dengan GPenSIM (General Purpose Simulator), sebuah software yang dijalankan dengan platform MATLAB. Model simulasi yang akan dibuat dengan GPenSIM terdiri dari tiga jenis m-file, yaitu Petri net Definition Files (PDF), Transition Definition Files (TDF), dan Main Simulation File (MSF). Rincian statis dari model akan ditulis pada bagian PDF, pengaturan prioritas firing transition akan diatur melalui TDFs, sedangkan rincian dinamis dan hasil simulasi yang akan ditulis pada bagian MSF.

Adapun prosedur pembuatan model simulasi dari sistem PLTH diuraikan dalam beberapa langkah berikut:

- Langkah 1: *Menetapkan strategi yang akan disimulasikan*
Rincian statis model PLTH dibuat dalam bagian PDF dari GPenSIM. Rincian ini memuat rincian place, transition, dan arc dalam model PLTH tersebut.
- Langkah 2: *Membuat rincian statis model simulasi*
Rincian statis model PLTH dibuat dalam bagian PDF dari GPenSIM. Rincian ini memuat rincian place, transition, dan arc dalam model PLTH tersebut.
- Langkah 3: *Membuat prioritas firing beberapa transition*
Agar simulasi berjalan sesuai dengan mekanisme sistem PLTH dengan strategi yang dipilih, maka pada langkah ini dibuat prioritas firing dari transition yang enabled. Pengaturan ini dibuat pada bagian TDF.
- Langkah 4: *Membuat rincian dinamis model simulasi*
Rincian dinamis model PLTH dibuat dalam bagian MSF dari program GPenSIM. Rincian dinamis ini mencakup token awal dari satu atau beberapa place, waktu firing dari transition, dan bentuk tampilan hasil simulasi.
- Langkah 5: *Running program simulasi dari model sistem PLTH*
Hasil yang diperoleh dari proses running program adalah jumlah token terpakai, baik pada diesel, baterai, maupun pada photovoltaic, dan wind turbine. Hasil running lainnya adalah grafik kurva beban dan grafik daya yang sampai ke beban.
- Langkah 6: *Melakukan validasi terhadap hasil simulasi*
Jika grafik kurva beban dan grafik daya yang sampai ke beban sudah sesuai, maka hasil simulasi sudah valid dan bisa direkam.

Berikut ini adalah diagram alir (flowchart) yang menggambarkan prosedur di atas:



Gambar 3.12 flowchart prosedur simulasi model sistem PLTH

3.2.1 Kode Program PDF

Kode program PDF memuat pendefinisian place, transition, dan arc yang digunakan pada struktur PN dari model sistem PLTH yang telah dibuat seperti yang telah diuraikan pada subbab 2.3.2.

Terdapat sedikit perbedaan pada kode program PDF antara strategi load-following dan cycle-charging dan battery state of charge. Pada strategi load-following, tidak ada arc (tanda panah) dari transition generator diesel (tD4AC dan tD4DC) ke place baterai (pBAT). Hal ini dikarenakan pada strategi load-following, daya listrik dari generator diesel tidak disimpan ke baterai. Adapun listing program PDF tersebut dapat dilihat pada bagian lampiran A.

3.2.2 Kode Program TDF

Pada kode program TDF dilakukan pengaturan *firing* transisi-transisi berdasarkan prioritasnya. Secara umum, pengaturan *firing* dari transisi-transisi pada tiap strategi yang akan diterapkan berbeda-beda. Pada strategi load-following, file-file TDF dibuat agar penggunaan sumber-sumber daya listrik dari pv dan wt lebih diprioritaskan daripada sumber dari generator diesel. Pada strategi cycle-charging, file-file TDF dibuat agar suplai daya listrik dari generator diesel lebih diprioritaskan daripada suplai dari pv dan wt. Sedangkan pada strategi battery state of charge, file-file TDF dibuat untuk mengatur daya listrik minimal yang tersimpan di dalam baterai. Dalam hal ini, disetting pada nilai minimal 70% dari kapasitas maksimumnya. Listing program TDF dapat dilihat pada bagian lampiran B.

3.2.3 Kode Program MSF

Pengaturan bentuk statis, bentuk dinamis, dan tampilan hasil simulasi dari struktur Petri Net dibuat melalui program MATLAB dari kode program MSF sebagaimana yang telah diuraikan pada subbab 2.3.4. Listing program MSF dapat dilihat pada bagian lampiran C.

3.2.4 Running Program Model Simulasi Petri Net Sistem PLTH

Running program dari model simulasi ini bertujuan untuk mengamati proses yang berlangsung pada model Petri Net sistem PLTH yang dibuat. Simulasi dibagi menjadi tiga bagian, yaitu:

- a. Simulasi untuk strategi load-following.
- b. Simulasi untuk strategi cycle-charging.
- c. Simulasi untuk strategi battery state of charge.

Proses simulasi dimulai dengan memasukkan data perkiraan kebutuhan beban (kurva beban) serta data potensi surya dan angin pada bagian dinamis program GPenSIM (MSF). Data kebutuhan beban yang dimasukkan berupa besaran daya listrik dengan satuan kW . Data potensi surya yang dimasukkan berupa besaran intensitas radiasi surya dalam satuan $kWh/m^2/hari$ sedangkan data potensi angin yang dimasukkan berupa besaran kecepatan angin dalam satuan m/dtk . Pada proses selanjutnya, data potensi surya (dalam satuan $kWh/m^2/hari$) dan data potensi angin (dalam satuan m/dtk) diubah menjadi data ketersediaan daya listrik photovoltaic dan wind turbine dalam satuan kW berdasarkan rumus (2.1) dan (2.2) seperti yang telah diuraikan pada subbab 2.1.1 dan 2.1.2. Proses simulasi pada bagian MSF dilanjutkan dengan konversi dari besaran daya listrik (dalam satuan kW) menjadi besaran token, sehingga kebutuhan beban ac, kebutuhan beban dc, ketersediaan pv, ketersediaan wt1, dan ketersediaan wt2 sudah dalam besaran token. Proses simulasi diakhiri dengan dilakukannya mekanisme aliran daya listrik (dalam token) dari sumber-sumber daya listrik hingga ke beban ac dan/atau beban dc. Besarnya daya listrik yang dikeluarkan oleh sumber-sumber listrik disesuaikan dengan data kebutuhan beban (dalam token). Proses simulasi ini melibatkan semua bagian dari program GPenSIM, yaitu PDF, TDF, dan MSF.

Simulasi dilanjutkan dengan menghitung estimasi biaya dengan ketiga macam strategi yang diterapkan. Adapun asumsi biaya yang digunakan pada perhitungan tersebut adalah sebagai berikut (*sumber: CV.Diariona*):

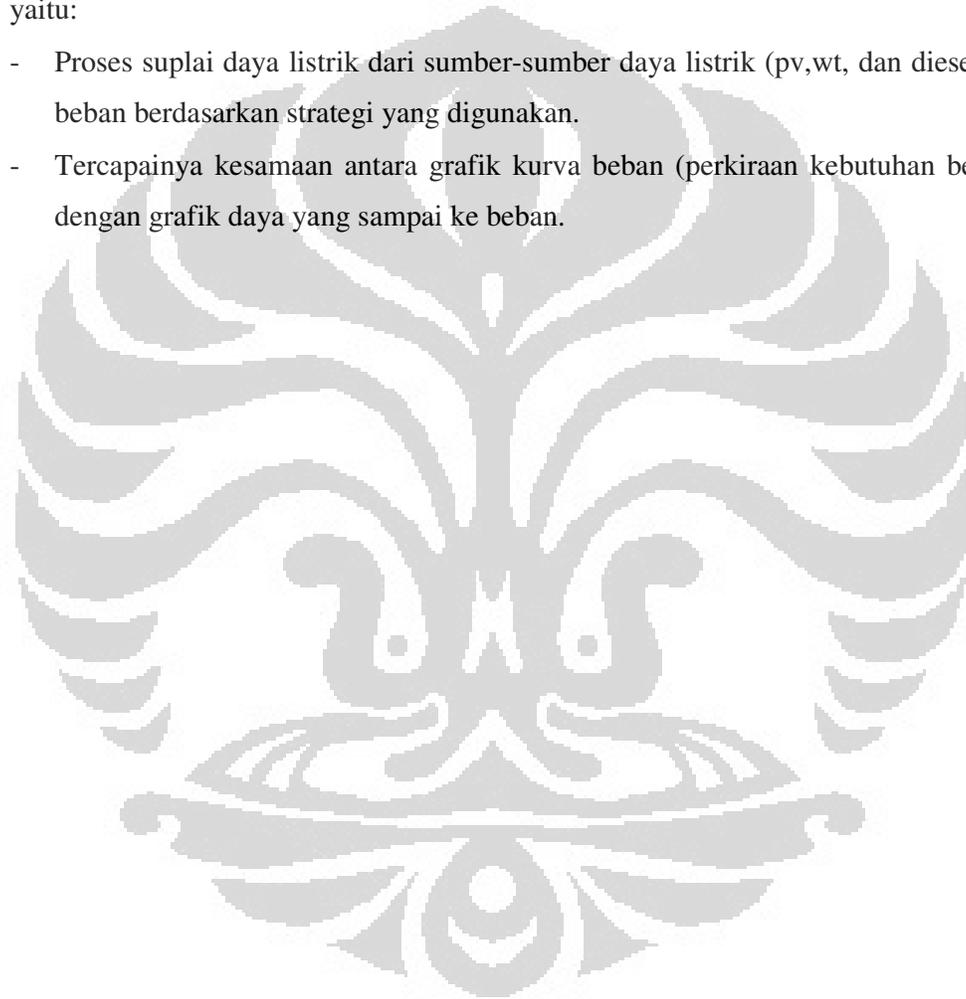
- biaya penyaluran daya listrik dari photovoltaic = Rp 19.500.000,-/kW
- biaya penyaluran daya listrik dari wind turbine = Rp 17.500.000,-/kW

- biaya penyaluran daya listrik dari generator diesel = Rp 10.800.000,-/kW

3.2.5 Validasi Hasil Simulasi

Setelah simulasi dilakukan, maka hasilnya divalidasi. Tujuannya adalah untuk membandingkan hasil simulasi tersebut dengan mekanisme kerja sistem PLTH yang diharapkan. Dalam hal ini, ada 2 mekanisme kerja yang ingin diamati, yaitu:

- Proses suplai daya listrik dari sumber-sumber daya listrik (pv,wt, dan diesel) ke beban berdasarkan strategi yang digunakan.
- Tercapainya kesamaan antara grafik kurva beban (perkiraan kebutuhan beban) dengan grafik daya yang sampai ke beban.



BAB 4

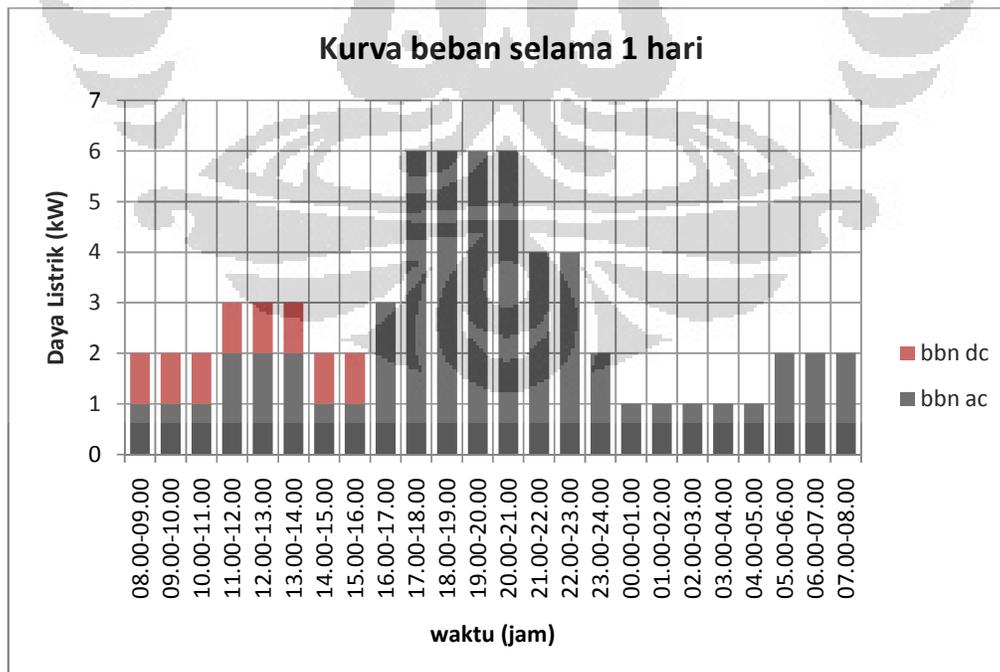
HASIL SIMULASI DAN PEMBAHASAN

1.13 Desain Simulasi

Tujuan simulasi ini adalah untuk memperlihatkan proses suplai energi listrik dari sumber-sumber tenaga listrik ke beban dengan tiga strategi yang berbeda; strategi *load-following* (LF), strategi *cycle-charging* (CC), dan strategi *battery state of charge* (BSC). Sumber tenaga listriknya berasal dari kombinasi photovoltaic (pv), wind turbine (wt), dan diesel. Mekanisme proses pada PLTH yang ingin diamati dengan simulasi ini adalah proses penyaluran energi listrik ke beban dengan ketiga strategi tersebut serta estimasi perhitungan biayanya.

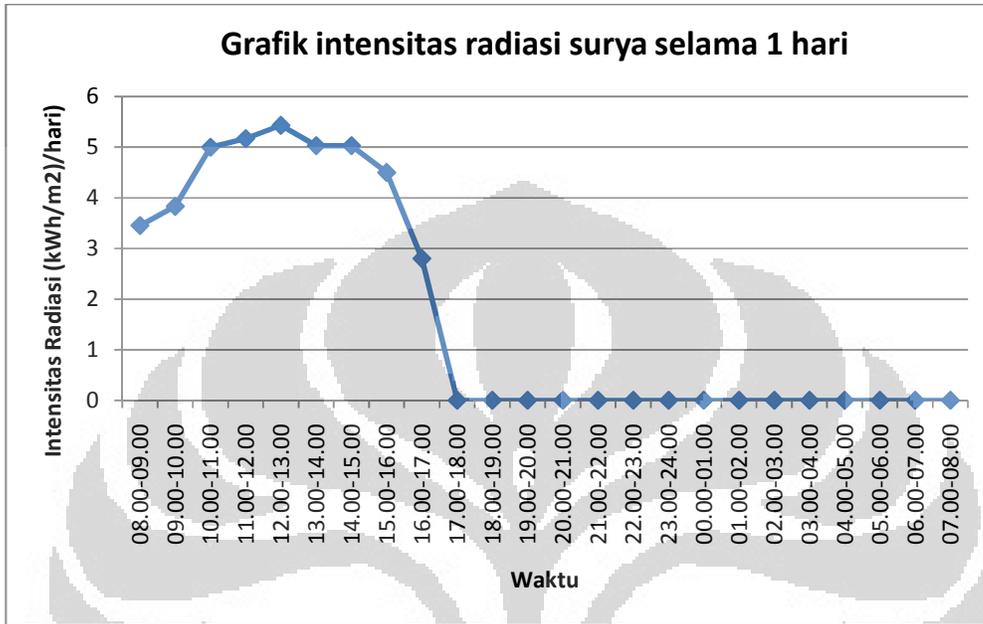
Pada simulasi yang dilakukan, besarnya kebutuhan listrik dari beban ditampilkan dalam bentuk kurva beban (load profil), pasokan listrik dari photovoltaic (PV) dihitung dari data intensitas radiasi cahaya matahari, pasokan dari wind turbine (WT1 dan WT2) dihitung dari data kecepatan angin, sedangkan diesel tersedia maksimum sebesar 4 kW.

Semua data disajikan dalam bentuk data harian yang dibagi menjadi 24 siklus (1 siklus untuk 1 jam). Kurva beban harian diperlihatkan pada Gambar 4.1 berikut:



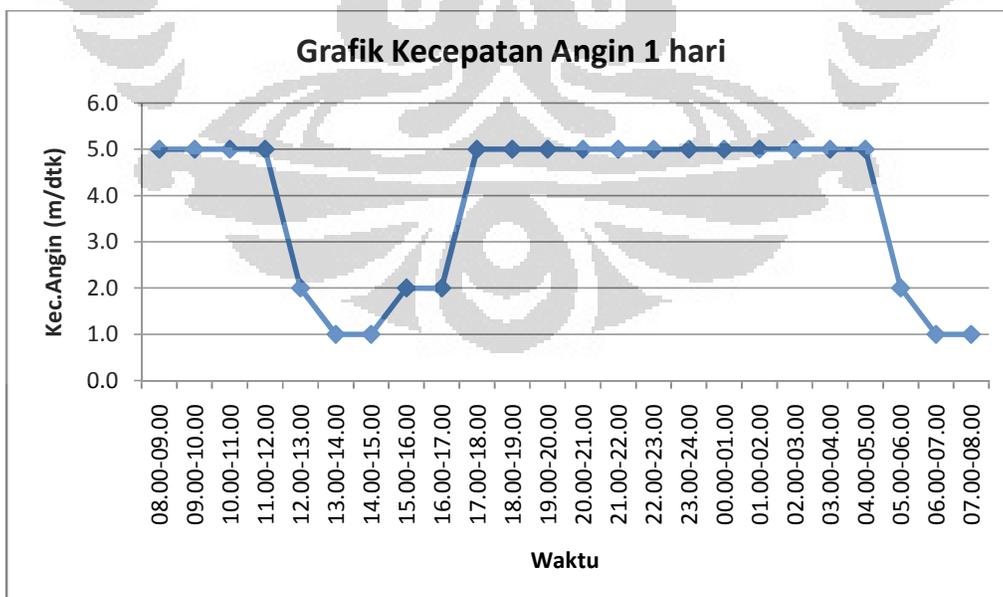
Gambar 4.1 Kurva beban selama 1 hari

Data radiasi surya selama 1 hari diperlihatkan pada Gambar 4.2 berikut:



Gambar 4.2 Grafik radiasi surya dalam 1 hari

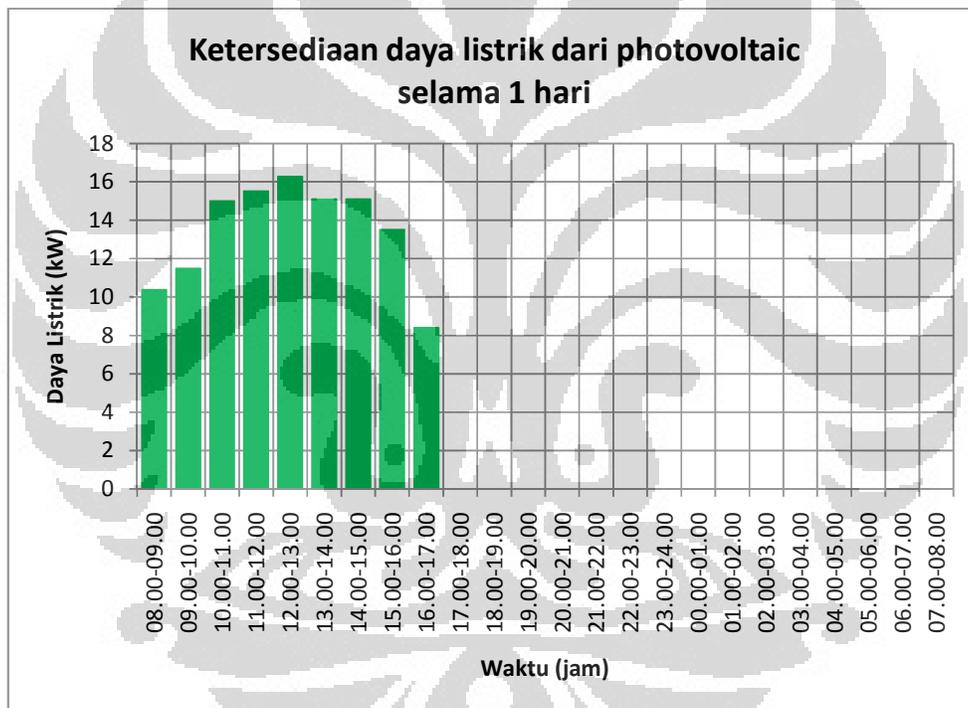
Sedangkan data kecepatan angin dalam 1 hari diperlihatkan pada Gambar 4.3 berikut:



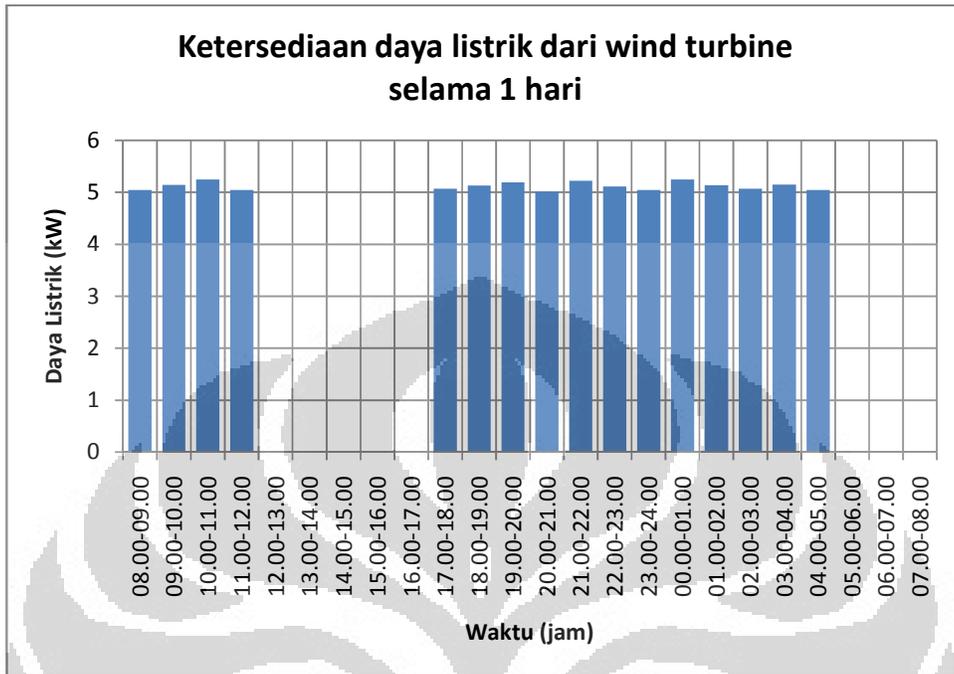
Gambar 4.3 Grafik kecepatan angin dalam 1 hari

Data potensi listrik dan kurva beban di atas akan digunakan dalam proses simulasi, baik untuk strategi load-following, strategi cycle-charging, maupun strategi state of charge the battery. Diharapkan agar model sistem PLTH dapat disimulasikan dengan ketiga strategi tersebut.

Sebelum simulasi dilakukan, data potensi surya dan potensi angin seperti yang ditunjukkan pada Gambar 4.2 dan Gambar 4.3 diubah menjadi ketersediaan daya listrik dari surya (photovoltaic) dan ketersediaan daya listrik dari angin (wind turbine) dalam satuan kW terlebih dahulu seperti diperlihatkan pada Gambar 4.4 dan 4.5 berikut:

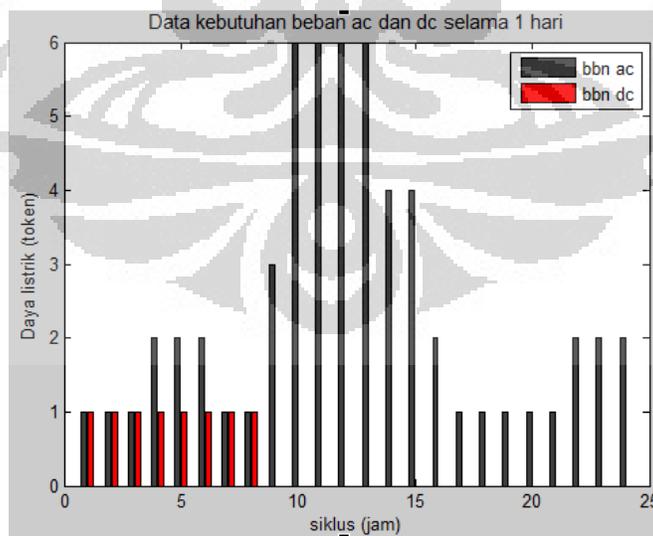


Gambar 4.4 Ketersediaan daya listrik dari pv selama 1 hari

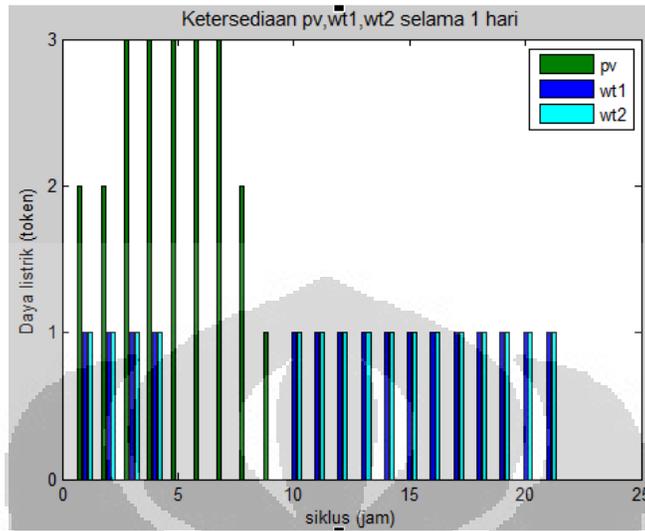


Gambar 4.5 Ketersediaan daya listrik dari angin selama 1 hari

Selanjutnya, data daya listrik dari kurva beban serta daya listrik dari photovoltaic (pv) dan wind turbine (wt) dikonversi menjadi token. Data tersebut ditampilkan pada Gambar 4.6 dan Gambar 4.7 berikut:

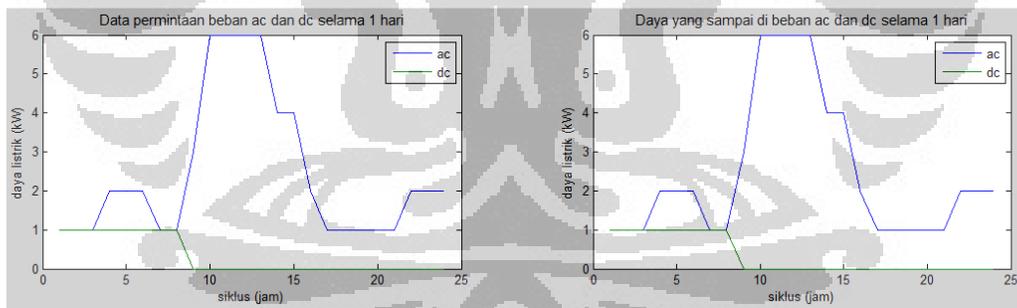


Gambar 4.6 Jumlah token kebutuhan beban ac dan dc selama 1 hari

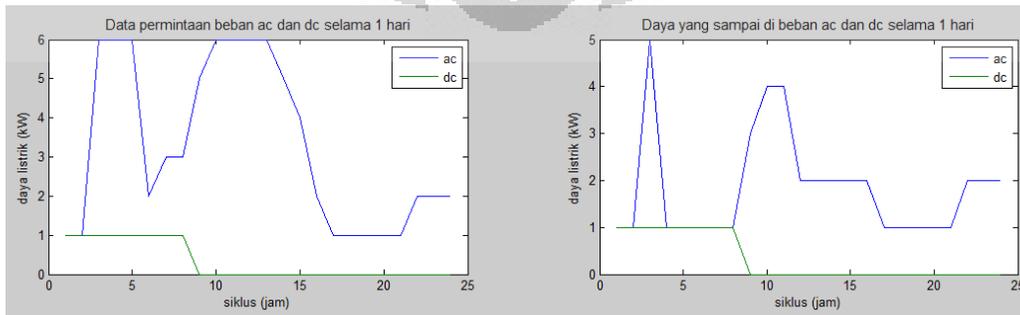


Gambar 4.7 Jumlah token pv, wt1, dan wt2 selama 1 hari

Pada proses simulasi, akan dilakukan validasi hasil simulasi. Jika terjadi kesesuaian antara estimasi kebutuhan beban dan daya yang tersuplai ke beban, maka hasil simulasi sudah tepat dan dapat dicatat/direkam. Kesesuaian dan ketidaksesuaian tersebut dapat dilihat pada Gambar 4.8 dan Gambar 4.9 berikut.



Gambar 4.8 Grafik kesesuaian antara kebutuhan dan suplai ke beban



Gambar 4.9 Grafik ketidaksesuaian antara kebutuhan dan suplai ke beban

1.14 Hasil Simulasi dengan Strategi Load-Following (LF)

Pada strategi ini, suplai daya listrik dari sumber-sumber energi baru dan terbarukan (photovoltaic dan wind turbine) lebih diutamakan daripada suplai dari generator diesel. Selain itu, hanya daya listrik dari photovoltaic dan wind turbine yang disimpan ke baterai, sedangkan dari generator diesel tidak disimpan ke baterai. Setelah disimulasikan, diperoleh hasil seperti terlihat pada tabel 4.1 dan tabel 4.2 berikut:

Tabel 4.1 Hasil simulasi strategi *load-following*

Jam ke-	Keb. Beban		Daya listrik tersedia					Daya listrik terpakai			Tersimpan di baterai
	ac	dc	pv	wt1	wt2	bat	diesel	pv	wt	diesel	
1	1	1	2	1	1	7	4	0	1	0	9
2	1	1	2	1	1	9	4	0	1	1	10
3	1	1	3	1	1	10	4	1	0	1	10
4	2	1	3	1	1	10	4	2	0	1	10
5	2	1	3	0	0	10	4	0	0	2	10
6	2	1	3	0	0	10	4	2	0	1	10
7	1	1	3	0	0	10	4	0	0	1	10
8	1	1	2	0	0	10	4	1	0	1	10
9	3	0	1	0	0	10	4	1	0	1	9
10	6	0	0	1	1	9	4	0	2	2	7
11	6	0	0	1	1	7	4	0	2	2	5
12	6	0	0	1	1	5	4	0	2	2	3
13	6	0	0	1	1	3	4	0	2	2	1
14	4	0	0	1	1	1	4	0	2	1	0
15	4	0	0	1	1	0	4	0	2	2	0
16	2	0	0	1	1	0	4	0	1	1	1
17	1	0	0	1	1	1	4	0	0	1	3
18	1	0	0	1	1	3	4	0	1	0	4
19	1	0	0	1	1	4	4	0	1	0	5
20	1	0	0	1	1	5	4	0	1	0	6
21	1	0	0	1	1	6	4	0	0	0	7
22	2	0	0	0	0	7	4	0	0	1	6
23	2	0	0	0	0	6	4	0	0	1	5
24	2	0	0	0	0	5	4	0	0	1	4
jumlah	59	8	22	16	16			7	18	25	

Tabel 4.2 Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada LF

PV	WT	Baterai	Diesel	Jumlah
7	18	17	25	67

*) Daya dari baterai bersumber dari pv dan wt.

1.15 Hasil Simulasi dengan Strategi Cycle-Charging (CC)

Pada strategi ini, suplai daya listrik dari generator diesel lebih diutamakan daripada suplai dari sumber-sumber energi baru dan terbarukan (photovoltaic dan wind turbine). Oleh karena itu, kelebihan daya listrik dari generator diesel juga disimpan ke baterai seperti halnya kelebihan daya listrik dari photovoltaic dan wind turbine. Pada simulasi ini, kapasitas maksimum generator diesel adalah 4 kW atau setara dengan 4 token. Setelah disimulasikan, diperoleh hasil seperti terlihat pada tabel 4.3 dan tabel 4.4 berikut:

Tabel 4.3 Hasil simulasi strategi *cycle-charging*

Jam ke-	Keb. Beban		Daya listrik tersedia					Daya listrik terpakai			Tersimpan di baterai
	ac	dc	pv	wt1	wt2	bat	diesel	pv	wt	diesel	
1	1	1	2	1	1	7	4	0	0	4	7
2	1	1	2	1	1	10	4	0	0	2	10
3	1	1	3	1	1	10	4	0	0	2	10
4	2	1	3	1	1	10	4	0	0	2	10
5	2	1	3	0	0	10	4	0	0	3	10
6	2	1	3	0	0	10	4	0	0	3	10
7	1	1	3	0	0	10	4	0	0	2	10
8	1	1	2	0	0	10	4	0	1	2	10
9	3	0	1	0	0	10	4	0	0	2	10
10	6	0	0	1	1	10	4	1	0	4	10
11	6	0	0	1	1	10	4	1	0	4	10
12	6	0	0	1	1	10	4	1	0	4	10
13	6	0	0	1	1	10	4	1	1	4	10
14	4	0	0	1	1	10	4	1	1	2	10
15	4	0	0	1	1	10	4	0	1	3	10
16	2	0	0	1	1	10	4	0	1	2	10
17	1	0	0	1	1	10	4	0	0	0	10
18	1	0	0	1	1	10	4	0	0	1	10
19	1	0	0	1	1	10	4	0	0	0	10
20	1	0	0	1	1	10	4	0	0	1	10
21	1	0	0	1	1	10	4	0	1	1	10
22	2	0	0	0	0	10	4	0	0	2	10
23	2	0	0	0	0	10	4	0	1	2	10
24	2	0	0	0	0	10	4	0	0	2	10
jumlah	59	8	22	16	16			5	10	54	

Tabel 4.4 Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada CC

PV	WT	Baterai	Diesel	Jumlah
5	10	0	54	67

1.16 Hasil Simulasi dengan Strategi Battery State of Charge (BSC)

Pada strategi ini, daya listrik yang tersimpan dalam baterai diatur batas minimumnya. Pada model simulasi yang dibuat, kapasitas minimum baterai disetting pada nilai 70%. Artinya, baterai dipakai menyuplai listrik jika tersimpan daya listrik minimal 70% dari kapasitas maksimumnya.

Pada model yang dibuat, kapasitas maksimum baterai sebesar 10 kW atau 10 token. Jadi, pada baterai harus selalu tersimpan daya listrik berkisar antara 7-10 token. Setelah disimulasikan, diperoleh hasil seperti terlihat pada tabel 4.5 dan 4.6 berikut:

Tabel 4.5 Hasil simulasi strategi *battery state of charge*

Jam ke-	Keb. Beban		Daya listrik tersedia					Daya listrik terpakai			Tersimpan di baterai
	ac	dc	pv	wt1	wt2	bat	diesel	pv	wt	diesel	
1	1	1	2	1	1	7	4			0	9
2	1	1	2	1	1	9	4			0	10
3	1	1	3	1	1	10	4			0	10
4	2	1	3	1	1	10	4			1	10
5	2	1	3	0	0	10	4			1	10
6	2	1	3	0	0	10	4			2	10
7	1	1	3	0	0	10	4			1	10
8	1	1	2	0	0	10	4			2	10
9	3	0	1	0	0	10	4			1	9
10	6	0	0	1	1	9	4			2	7
11	6	0	0	1	1	7	4			4	7
12	6	0	0	1	1	7	4			4	7
13	6	0	0	1	1	7	4			4	7
14	4	0	0	1	1	7	4			2	7
15	4	0	0	1	1	7	4			2	7
16	2	0	0	1	1	7	4			1	8
17	1	0	0	1	1	8	4			1	10
18	1	0	0	1	1	10	4			0	10
19	1	0	0	1	1	10	4			1	10
20	1	0	0	1	1	10	4			1	10
21	1	0	0	1	1	10	4			0	10
22	2	0	0	0	0	10	4			1	9
23	2	0	0	0	0	9	4			1	8
24	2	0	0	0	0	8	4			1	7
jumlah	59	8	22	16	16			16		33	

Tabel 4.6 Pemakaian daya listrik dari pv,wt,baterai, dan diesel pada BSC

PV	WT	Baterai	Diesel	Jumlah
4	12	18	33	67

*) Daya dari baterai bersumber dari pv dan wt.

1.17 Analisis Hasil Simulasi

Hasil simulasi dari ketiga strategi yang diujicobakan menunjukkan adanya kesesuaian dengan hasil yang diinginkan dari penerapan ketiga strategi tersebut. Pada strategi load-following, sebagian besar daya yang terpakai untuk menyuplai beban berasal dari photovoltaic (pv) dan wind turbine (wt), sedangkan pada strategi cycle-charging, daya yang terpakai sebagian besar berasal dari generator diesel. Di samping itu, pada strategi battery state of charge, kapasitas daya listrik yang tersimpan di dalam baterai tidak pernah lebih rendah daripada settingnya, yaitu minimal 70% dari kapasitas maksimumnya.

Pada simulasi dengan strategi load-following, penggunaan daya listrik dari diesel hanya sebanyak 25 token dari total beban 67 token, atau sebesar 37,3% dari total daya yang disuplai ke beban, sedangkan daya listrik dari pv dan wt yang digunakan sebanyak 42 token atau sebesar 62,7% dari total kebutuhan beban. Hal ini menunjukkan bahwa sebagian besar daya listrik yang disuplai ke beban berasal dari sumber-sumber energi baru dan terbarukan (pv dan wt).

Pada simulasi dengan penerapan strategi cycle-charging, besarnya daya listrik dari generator diesel yang terpakai adalah 54 token atau 80,6% dari total daya tersuplai ke beban, sedangkan yang berasal dari pv dan wt hanya sebesar 19,4% dari total daya terpakai. Hal ini menunjukkan tersuplainya beban oleh sebagian besar daya listrik yang bersumber dari generator diesel.

Pada simulasi yang didasarkan pada penggunaan strategi battery state of charge, besarnya daya listrik yang tersimpan di dalam baterai tidak pernah kurang dari 70% dari kapasitas maksimumnya (7 token dari maksimum 10 token). Hal ini sesuai dengan setting baterai dimana suplai daya listrik ke beban dari baterai hanya dilakukan bila terdapat sekurang-kurangnya 8 token di dalam baterai. Jika jumlah token di dalam baterai hanya 7 token atau lebih kecil, maka baterai tidak akan menyuplai daya listrik ke beban.

Berdasarkan asumsi biaya pemakaian listrik dari pv, wt, dan diesel seperti yang dicantumkan pada subbab 3.2.4 (masing-masing sebesar Rp 19.500.000,-/kW, Rp 17.500.000,-/kW, dan Rp 10.800.000,-/kW), maka total biaya yang dibutuhkan dengan ketiga strategi yang digunakan dapat dilihat pada tabel 4.7. Dari perhitungan total biaya yang dibutuhkan pada ketiga macam strategi tersebut, nampak bahwa biaya terendah diperoleh jika menggunakan strategi *battery state of charge*. Rangkuman hasil simulasi juga dapat dilihat pada tabel 4.7 berikut:

Tabel 4.7 Perhitungan biaya dengan tiga macam strategi

Strategi	Tok PV terpakai	Tok WT terpakai	Tok Dis terpakai	Token minimum Tersimpan di baterai	Biaya (Rupiah)
Load-following	7	18	25	0	721.500.000
Cycle-charging	5	10	54	7	855.700.000
Battery state of charge	4	12	33	7	644.400.000

Kekurangan Pasokan Daya

Bila pasokan daya dari sumber-sumber daya listrik tidak cukup untuk menyuplai beban, maka hasil simulasi akan menampilkan informasi tersebut dan menampilkan kekurangan dayanya (dalam kW).

Contoh: Bila ketersediaan daya (dalam token) meliputi photovoltaic (0), wind turbine-1 (0), wind turbine-2 (0), diesel (4), baterai (0), sedangkan kebutuhan beban terdiri dari beban ac (6) dan dc (1), maka simulasi akan menampilkan informasi berikut:

```

Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

Ketersediaan diesel = 4 kW
Ketersediaan pv = 0 kW
Ketersediaan wt1 = 0 kW
Ketersediaan wt2 = 0 kW
Ketersediaan di baterai = 0 kW
Kebutuhan daya beban = 7 kW
PASOKAN DAYA TIDAK CUKUP
TERDAPAT KEKURANGAN SEBESAR 3 kW
    
```

1.18 Analisis Sifat Reachability

Analisis sifat reachability dari struktur Petri Net model sistem PLTH sebagaimana yang diperlihatkan pada Gambar 3.11 diawali dengan mengubah nama-nama place dalam struktur tersebut menjadi p_1, p_2, \dots dan nama-nama transition menjadi t_1, t_2, \dots

- pPDC=place kebutuhan beban ac $\rightarrow p_1$
- pPAC=place kebutuhan beban dc $\rightarrow p_2$
- pPV=place listrik photovoltaic (pv) $\rightarrow p_3$
- pWT1=place listrik wind turbine1 (wt1) $\rightarrow p_4$
- pWT2=place listrik wind turbine2 (wt2) $\rightarrow p_5$
- pBAT=place listrik dalam baterai $\rightarrow p_6$
- pDIS=place listrik generator diesel $\rightarrow p_7$
- pAC=place listrik ac dari generator diesel yang akan diubah menjadi dc $\rightarrow p_8$
- pDC=place listrik dc dari pv,wt,baterai yang akan diubah menjadi ac $\rightarrow p_9$
- pDCAC=place listrik ac keluaran dari konverter dc ke ac $\rightarrow p_{10}$
- pKON=place pengatur kerja konverter ac ke dc dan dc ke ac $\rightarrow p_{11}$
- pACDC=place listrik dc keluaran dari konverter ac ke dc $\rightarrow p_{12}$
- pACB=place listrik di ac bus $\rightarrow p_{13}$
- pDCB=place listrik di dc bus $\rightarrow p_{14}$
- pBDC=place listrik yang sampai ke beban ac $\rightarrow p_{15}$
- pBAC=place listrik yang sampai ke beban dc $\rightarrow p_{16}$
- tP4AC=trans penyaluran listrik pv ke konverter $\rightarrow t_1$
- tP4DC=trans penyaluran listrik pv beban dc $\rightarrow t_2$
- tP4BT=trans penyimpanan daya pv ke bat $\rightarrow t_3$
- tW1AC=trans penyaluran listrik wt1 ke konverter $\rightarrow t_4$
- tW1DC=trans penyaluran listrik wt1 beban dc $\rightarrow t_5$
- tW1BT=trans penyimpanan daya wt1 ke bat $\rightarrow t_6$
- tW2AC=trans penyaluran listrik wt2 ke konverter $\rightarrow t_7$
- tW2DC=trans penyaluran listrik wt2 beban dc $\rightarrow t_8$
- tW2BT=trans penyimpanan daya wt2 ke bat $\rightarrow t_9$
- tB4AC=trans penyaluran daya bat ke konverter dc ke ac $\rightarrow t_{10}$
- tB4DC=trans penyaluran daya bat ke konverter ac ke dc $\rightarrow t_{11}$
- tD4AC=trans penyaluran listrik diesel ke beban ac $\rightarrow t_{12}$

tD4DC=trans penyaluran listrik diesel ke konverter → t13
tK4AC=trans konversi dc ke ac → t14
tK4DC=trans konversi ac ke dc → t15
tAC2B=trans penyaluran listrik ac ke ac bus → t16
tDC2B=trans penyaluran listrik dc ke dc bus → t17
tBAC=trans penyaluran daya listrik dari ac bus ke beban ac → t18
tBDC=trans penyaluran daya listrik dari dc bus ke beban dc → t19

Kondisi awal dinyatakan dengan M_0 dengan menggunakan rumus (2.4) sebagai:

$$M_0 = [x(p1), x(p2), x(p3), \dots, x(p16)]$$

dimana:

$$x(pn) = \text{jumlah token awal pada place } p_n \text{ (n=1,2,\dots,16)}$$

Sedangkan kondisi yang diinginkan dinyatakan dengan M_i dengan menggunakan rumus (2.5) sebagai:

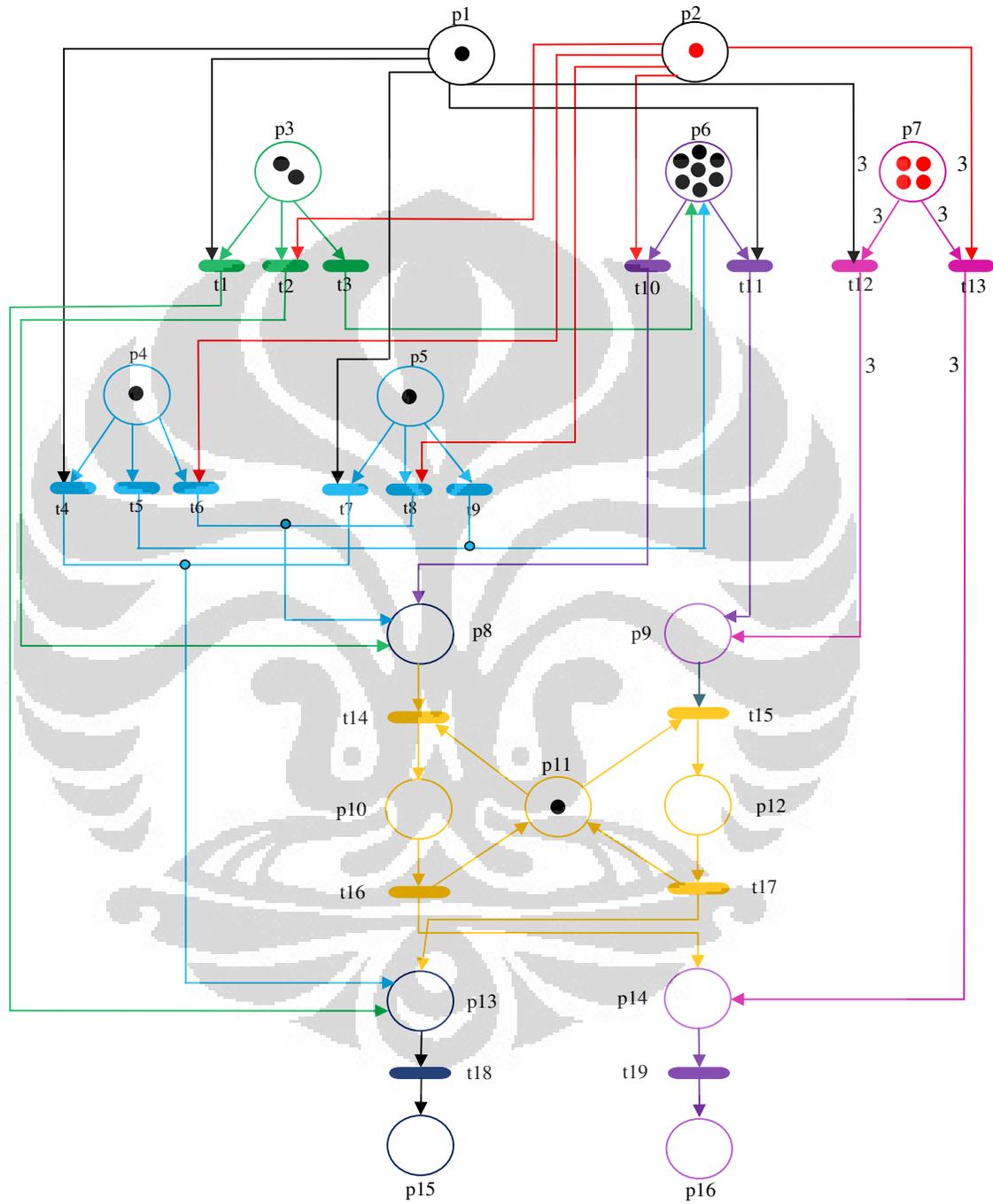
$$M_i = [x'(p1), x'(p2), x'(p3), \dots, x'(p16)]$$

dimana:

$$x'(pn) = \text{jumlah token yang diinginkan pada place } p_n \text{ (n=1,2,\dots,16)}$$

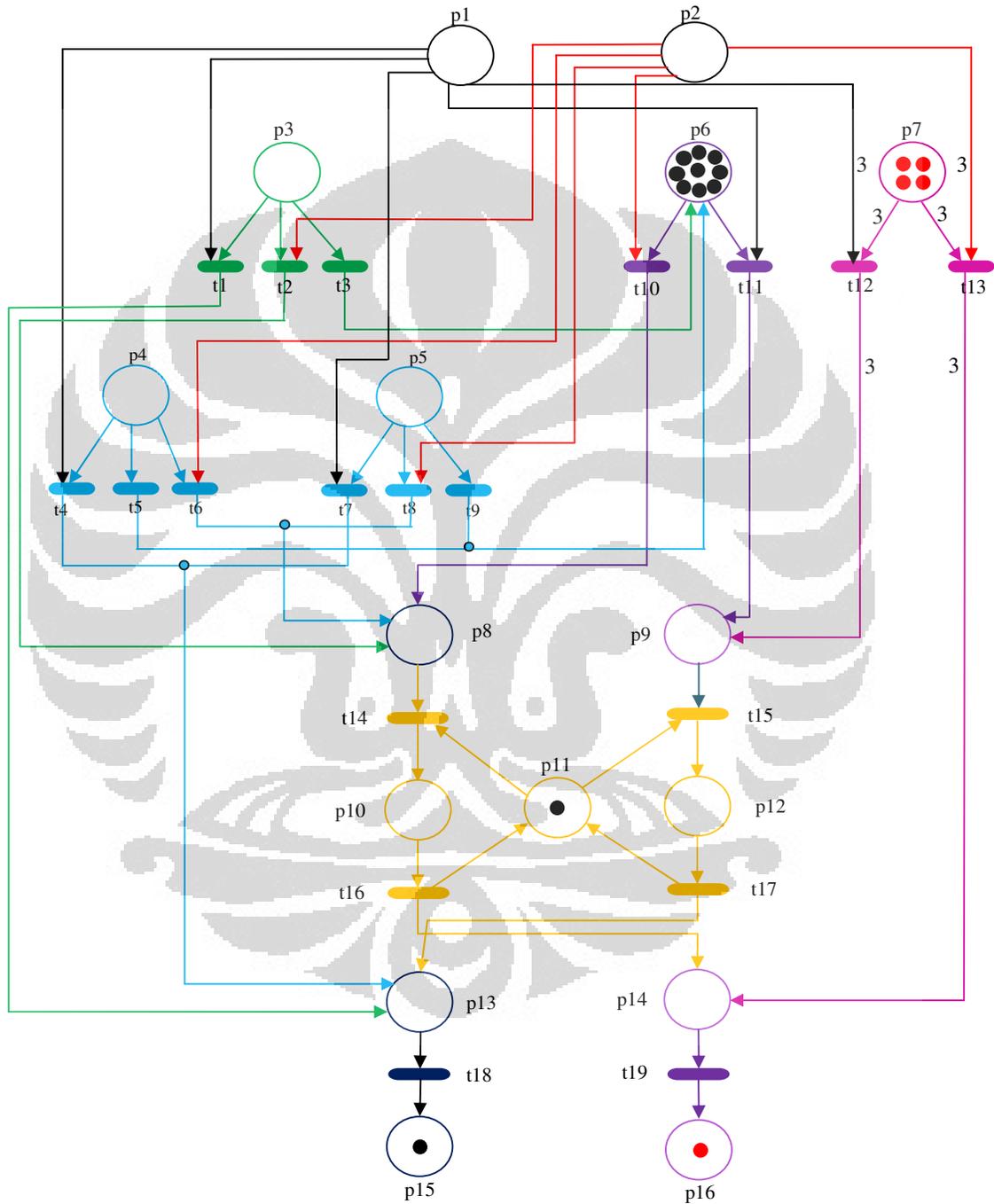
Contoh data yang akan digunakan adalah **simulasi strategi load-following pada jam pertama (siklus pertama)**.

- Kondisi awal (M_0) nampak pada Gambar 4.8 = [1,1,2,1,1,7,4,0,0,0,1,0,0,0,0]



Gambar 4.8 kondisi awal struktur PN model sistem PLTH

Kondisi yang diinginkan (M_i) nampak pada Gambar 4.9 = $[0,0,0,0,0,9,4,0,0,0,1,0,0,0,1,1]$



Gambar 4.9 kondisi yang diinginkan dari struktur PN model sistem PLTH

Setelah disimulasikan, diperoleh hasil seperti terlihat pada tabel 4.8 berikut:

Tabel 4.8 Hasil simulasi load-following pada jam pertama

state	p1	p2	p3	p4	p5	p6	p7	p8	P9	P10	P11	P12	P13	P14	P15	p16	
1	1	1	2	1	1	7	4	0	0	0	1	0	0	0	0	0	→ M ₀
2	0	0	1	0	0	8	4	0	0	0	1	0	0	0	0	0	
3	0	0	0	0	0	8	4	0	0	0	1	0	0	1	0	0	
4	0	0	0	0	0	8	4	0	1	0	1	0	0	0	0	0	
5	0	0	0	0	0	8	4	0	0	1	0	0	0	0	0	0	
6	0	0	0	0	0	8	4	0	0	0	0	0	0	0	0	1	
7	0	0	0	0	0	9	4	0	0	0	0	0	0	0	0	1	
8	0	0	0	0	0	9	4	0	0	0	1	0	1	0	0	1	
9	0	0	0	0	0	9	4	0	0	0	1	0	0	0	1	1	→ M _i

Dari tabel 4.58 di atas nampak bahwa marking M_i dikatakan **reachable** dari M₀ karena urutan firing yang terjadi dapat menghasilkan kondisi seperti yang diinginkan, yaitu M_i.

1.19 Analisis Deadlock/Liveness

Struktur Petri Net sebagai model dari sistem PLTH sebagaimana yang ditunjukkan pada Gambar 3.11 menunjukkan bahwa proses eksekusi dari struktur tersebut selalu akan berakhir pada place beban ac dan/atau beban dc (pBAC dan pBDC). Hal ini dikarenakan tidak adanya tanda panah (arc) yang menghubungkan antara pBAC dan pBDC dengan transition berikutnya atau dengan transition sebelumnya.

Pada subbab 4.6 di atas, kita juga dapat mengamati bahwa kondisi yang diinginkan merupakan akhir dari proses eksekusi. Dengan kata lain, setelah kondisi-kondisi yang diinginkan tersebut tercapai, maka struktur PN tidak dapat lagi dieksekusi.

Berdasarkan pada analisis di atas, maka struktur PN dari sistem PLTH merupakan struktur yang bersifat **deadlock**. Hal tersebut sesuai dengan yang diharapkan terjadi pada sistem PLTH, dimana daya listrik dari sumber-sumber energi listrik yang tersedia harus sampai hingga ke beban.

BAB 5

KESIMPULAN

Berdasarkan hasil simulasi dan analisis pembahasan, dapatlah dibuat kesimpulan sebagai berikut:

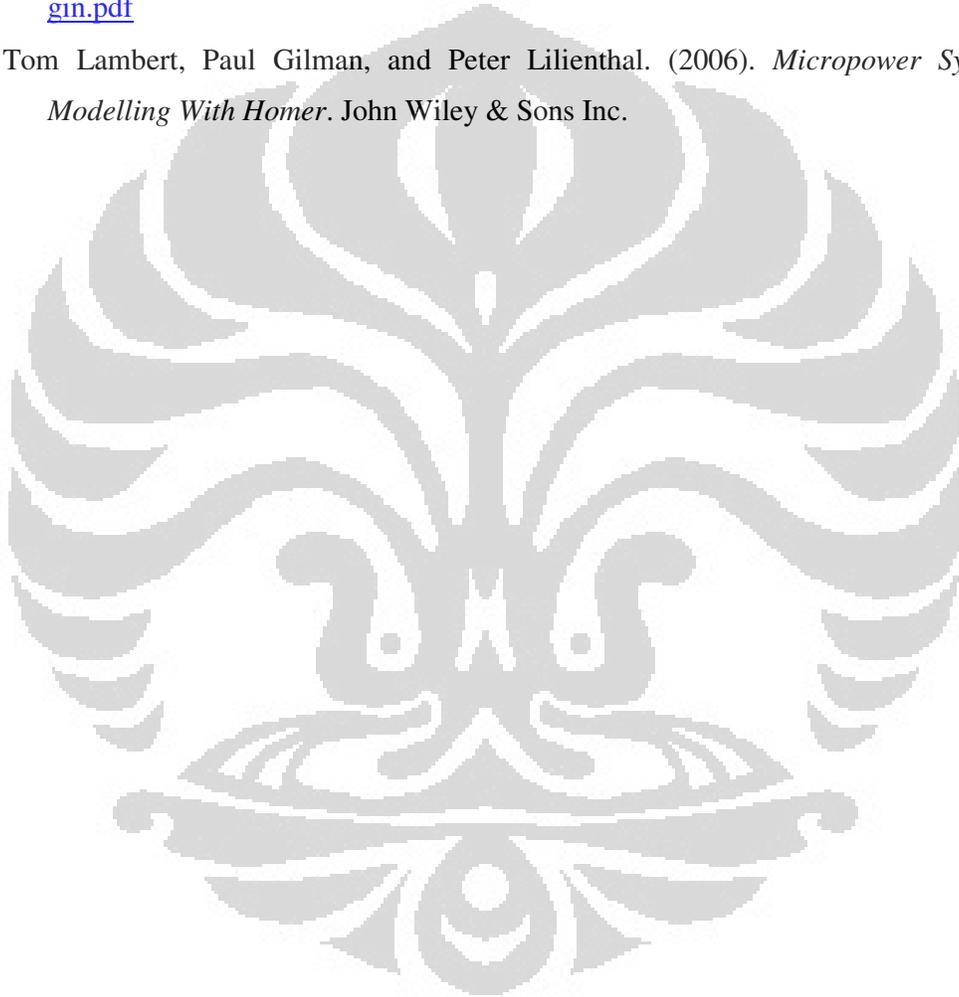
- Sistem Pembangkit Listrik Tenaga Hibrida (PLTH) yang kompleks (dengan empat unit sumber daya listrik, baterai, konverter bidirectional, ac bus, dc bus, beban ac, dan beban dc) dapat dibuat modelnya dengan metode Petri Net dengan mengikuti prosedur pemodelan dengan Petri Net.
- Program GPenSIM yang dijalankan dengan platform MATLAB dapat digunakan untuk membuat model simulasi dari model Petri Net sistem Pembangkit Listrik Tenaga Hibrida (PLTH). Dengan beberapa simulasi yang dilakukan, baik dengan strategi *load-following*, strategi *cycle-charging*, maupun strategi *battery state of charge*, simulasi terhadap model yang dibuat dapat digunakan dengan baik. Berdasarkan perhitungan biaya yang dilakukan dengan kurva kebutuhan beban dan pasokan yang tersedia pada ketiga macam strategi tersebut, biaya terendah diperoleh pada penerapan strategi *battery state of charge*.
- Model simulasi dengan GPenSIM dapat menampilkan pengoperasian sistem Pada simulasi dengan strategi *load-following*, penggunaan daya listrik dari diesel hanya sebanyak 25 token dari total beban 67 token, atau sebesar 37,3% dari total daya yang disuplai ke beban, sedangkan daya listrik dari pv dan wt yang digunakan sebanyak 42 token atau sebesar 62,7% dari total kebutuhan beban. Hal ini menunjukkan bahwa sebagian besar daya listrik yang disuplai ke beban berasal dari sumber-sumber energi baru dan terbarukan (pv dan wt).
- Pada simulasi dengan penerapan strategi *cycle-charging*, besarnya daya listrik dari generator diesel yang terpakai adalah 54 token atau 80,6% dari total daya tersuplai ke beban, sedangkan yang berasal dari pv dan wt hanya sebesar 19,4% dari total daya terpakai. Hal ini menunjukkan tersuplainya beban oleh sebagian besar daya listrik yang bersumber dari generator diesel.

- Pada simulasi yang didasarkan pada penggunaan strategi *battery state of charge*, besarnya daya listrik yang tersimpan di dalam baterai tidak pernah kurang dari 70% dari kapasitas maksimumnya (7 token dari maksimum 10 token). Hal ini sesuai dengan setting baterai dimana suplai daya listrik ke beban dari baterai hanya dilakukan bila terdapat sekurang-kurangnya 8 token di dalam baterai. Jika jumlah token di dalam baterai hanya 7 token atau lebih kecil, maka baterai tidak akan menyuplai daya listrik ke beban.
- Struktur Petri Net dari model sistem Pembangkit Listrik Tenaga Hibrida (PLTH) bersifat **reachable**. Artinya, kondisi akhir yang diinginkan berupa tersalurkannya daya listrik dari sumber-sumber daya listrik ke beban ac dan/atau beban dc dapat dicapai dari kondisi awal tertentu. Mekanisme perpindahan token melalui firing transition-transition dalam proses running model simulasi sistem PLTH berlangsung sedemikian rupa sebagaimana yang telah diatur pada bagian-bagian program GPenSIM, yaitu PDF, TDF, dan MSF.
- Proses perpindahan token dalam simulasi model Petri Net dari sistem PLTH selalu berakhir pada place beban ac (pBAC) dan/atau place beban dc (pBDC). Hal ini menunjukkan bahwa struktur Petri Net dari model sistem PLTH tersebut bersifat **deadlock**.

DAFTAR REFERENSI

1. Davidrajuh, Reggie. (2010). *GPenSIM v4.0, general purpose Petri Net simulator*. University of Stavanger, Norway.
2. Departemen Energi dan Sumber Daya Mineral. 2003. Pedoman dan Pola Tetap Pengembangan Industri Ketenagalistrikan Nasional 2003-2020: Blueprint Implementasi UU No.20 Tahun 2002 tentang Ketenagalistrikan. Jakarta, 21 April 2003.
3. Halim, Abdul. (2010). *Modul kuliah pemodelan kejadian diskrit. Petri Net: Pendahuluan*. Departemen Teknik Elektro FTUI.
4. Murata, Tadao. (1989). *Petri Nets: Properties, analysis, and applications*. Proceedings of the IEEE Vol.77 No.4.
5. Wiandini Fauziah. (2009). *Laporan Kerja Praktek: Sistem Pembangkit, Instalasi, dan Operasi, dan Pemeliharaan Pembangkit Listrik Tenaga Hibrida Surya, Bayu, Diesel PT.LEN Industri*. Bandung.
6. Alpen Steel. (2008). Sistem pembangkit listrik hibrida. January 18, 2011. <http://www.alpensteel.com/article/65-109-energi-fuel-cell-sel-bahan-bakar/2640--sistem-pembangkit-listrik-hibrida.html>
7. Humanity Development Library 2.0. New Zealand Digital Library. <http://www.nzdl.org/gsdmod?a=p&p=home&l=en&w=utf-8>.
8. Ikhwezy Unplugged. (2). *How it works – Configurations*. <http://www.solarec.co.za/solar-electricity-how-it-works.html>
9. Rachman, Taufik. (2010). *Listrik hibrid terbesar di Indonesia dikembangkan di Pandansimo*. January 18, 2011. Republika online, trendtek, sains. <http://www.fkdpm.org/berita/120-listrik-hibrid-terbesar-di-indonesia-dikembangkan-di-pandansimo.html>
10. Setio Dewo, Emanuel. (2006). *Pembangkit listrik tenaga hibrida (surya-diesel) untuk daerah terpencil*. January 18, 2011. <http://dewo.wordpress.com/2006/04/03/pembangkit-listrik-tenaga-hibrida-surya-diesel-untuk-daerah-terpencil>

11. TrenKonstruksi. (2010). *Tren Pembangkit Listrik Tenaga Hibrida (PLTH)*. September 15, 2010. <http://zy-time.blogspot.com/2010/10/tren-pembangkit-listrik-tenaga-hibrida.html>.
12. Universitas Gunadarma. E-learning Fisika Energi. Juni, 2010. http://elearning.gunadarma.ac.id/docmodul/dasar_fisika_energi/bab6_energi_angin.pdf
13. Tom Lambert, Paul Gilman, and Peter Lilienthal. (2006). *Micropower System Modelling With Homer*. John Wiley & Sons Inc.



LAMPIRAN A: Kode Program PDF

A.1 Strategi Load-Following

```
1 | % PDF dari PLTH (pv - wt1 - wt2 - gen diesel)
2 | % File: 'tesis_LF_def.m'
3 |
4 | function [PN_name, set_of_places, set_of_trans, set_of_arcs]...
5 |     =tesis_LF_def(global_info)
6 |
7 | PN_name = 'Model PN dari PLTH dengan strategi Load-Following (LF)';
8 | set_of_places = {'pPDC', 'pPAC', 'pPV', 'pWT1', 'pWT2', 'pBAT', 'pDIS', ...
9 |     'pDC', 'pAC', 'pKON', 'pDCAC', 'pACDC', 'pDCB', 'pACB', 'pBDC', 'pBAC'};
10 | set_of_trans = {'tP4DC', 'tP4AC', 'tP4BT', 'tW1DC', 'tW1AC', 'tW1BT', ...
11 |     'tW2DC', 'tW2AC', 'tW2BT', 'tB4DC', 'tB4AC', 'tD4DC', 'tD4AC', ...
12 |     'tK4AC', 'tK4DC', 'tAC2B', 'tDC2B', 'tBDC', 'tBAC'};
13 | set_of_arcs = {'pPDC', 'tP4DC', 1, 'pPDC', 'tW1DC', 1, 'pPDC', 'tW2DC', 1, ...
14 |     'pPDC', 'tB4DC', 1, 'pPDC', 'tD4DC', 1, ...
15 |     'pPAC', 'tP4AC', 1, 'pPAC', 'tW1AC', 1, 'pPAC', 'tW2AC', 1, ...
16 |     'pPAC', 'tB4AC', 1, 'pPAC', 'tD4AC', 1, ...
17 |     'pPV', 'tP4DC', 1, 'pPV', 'tP4AC', 1, 'pPV', 'tP4BT', 1, ...
18 |     'pWT1', 'tW1DC', 1, 'pWT1', 'tW1AC', 1, 'pWT1', 'tW1BT', 1, ...
19 |     'pWT2', 'tW2DC', 1, 'pWT2', 'tW2AC', 1, 'pWT2', 'tW2BT', 1, ...
20 |     'pBAT', 'tB4DC', 1, 'pBAT', 'tB4AC', 1, ...
21 |     'pDIS', 'tD4DC', 1, 'pDIS', 'tD4AC', 1, ...
22 |     'pKON', 'tK4AC', 1, 'pKON', 'tK4DC', 1, 'pDC', 'tK4AC', 1, 'pAC', 'tK4DC', 1, ...
23 |     'pDCAC', 'tAC2B', 1, 'pACDC', 'tDC2B', 1, 'pDCB', 'tBDC', 1, 'pACB', 'tBAC', 1, ...
24 |     'tP4DC', 'pDCB', 1, 'tP4AC', 'pDC', 1, 'tP4BT', 'pBAT', 1, 'tW1DC', 'pDCB', 1, ...
25 |     'tW2DC', 'pDCB', 1, 'tW1AC', 'pDC', 1, 'tW2AC', 'pDC', 1, ...
26 |     'tW1BT', 'pBAT', 1, 'tW2BT', 'pBAT', 1, ...
27 |     'tB4DC', 'pAC', 1, 'tB4AC', 'pDC', 1, ...
28 |     'tD4DC', 'pAC', 1, 'tD4AC', 'pACB', 1, 'tK4AC', 'pDCAC', 1, ...
29 |     'tK4DC', 'pACDC', 1, 'tAC2B', 'pKON', 1, 'tDC2B', 'pKON', 1, ...
30 |     'tAC2B', 'pACB', 1, 'tDC2B', 'pDCB', 1, 'tBDC', 'pBDC', 1, 'tBAC', 'pBAC', 1,};
```

A.2 Strategi Cycle-Charging

```
1 % PDF dari PLTH (pv - wt1 - wt2 - gen diesel)
2 % File: 'tesis_CC_def.m'
3
4 function [PN_name, set_of_places, set_of_trans, set_of_arcs]...
5     =tesis_CC_def(global_info)
6
7 PN_name = 'Model PN dari PLTH dengan strategi Cycle-Charging (CC)';
8 set_of_places = {'pPDC', 'pPAC', 'pPV', 'pWT1', 'pWT2', 'pBAT', 'pDIS', ...
9     'pDC', 'pAC', 'pKON', 'pDCAC', 'pACDC', 'pDCB', 'pACB', 'pBDC', 'pBAC'};
10 set_of_trans = {'tP4DC', 'tP4AC', 'tP4BT', 'tW1DC', 'tW1AC', 'tW1BT', ...
11     'tW2DC', 'tW2AC', 'tW2BT', 'tB4DC', 'tB4AC', 'tD4DC', 'tD4AC', 'tD4BT', ...
12     'tK4AC', 'tK4DC', 'tAC2B', 'tDC2B', 'tBDC', 'tBAC'};
13 set_of_arcs = {'pPDC', 'tP4DC', 1, 'pPDC', 'tW1DC', 1, 'pPDC', 'tW2DC', 1, ...
14     'pPDC', 'tB4DC', 1, 'pPDC', 'tD4DC', 1, ...
15     'pPAC', 'tP4AC', 1, 'pPAC', 'tW1AC', 1, 'pPAC', 'tW2AC', 1, ...
16     'pPAC', 'tB4AC', 1, 'pPAC', 'tD4AC', 1, ...
17     'pPV', 'tP4DC', 1, 'pPV', 'tP4AC', 1, 'pPV', 'tP4BT', 1, ...
18     'pWT1', 'tW1DC', 1, 'pWT1', 'tW1AC', 1, 'pWT1', 'tW1BT', 1, ...
19     'pWT2', 'tW2DC', 1, 'pWT2', 'tW2AC', 1, 'pWT2', 'tW2BT', 1, ...
20     'pBAT', 'tB4DC', 1, 'pBAT', 'tB4AC', 1, ...
21     'pDIS', 'tD4DC', 1, 'pDIS', 'tD4AC', 1, ...
22     'pKON', 'tK4AC', 1, 'pKON', 'tK4DC', 1, 'pDC', 'tK4AC', 1, 'pAC', 'tK4DC', 1, ...
23     'pDCAC', 'tAC2B', 1, 'pACDC', 'tDC2B', 1, 'pDCB', 'tBDC', 1, 'pACB', 'tBAC', 1, ...
24     'tP4DC', 'pDCB', 1, 'tP4AC', 'pDC', 1, 'tP4BT', 'pBAT', 1, 'tW1DC', 'pDCB', 1, ...
25     'tW2DC', 'pDCB', 1, 'tW1AC', 'pDC', 1, 'tW2AC', 'pDC', 1, ...
26     'tW1BT', 'pBAT', 1, 'tW2BT', 'pBAT', 1, ...
27     'tB4DC', 'pAC', 1, 'tB4AC', 'pDC', 1, ...
28     'tD4DC', 'pAC', 1, 'tD4AC', 'pACB', 1, 'tK4AC', 'pDCAC', 1, ...
29     'tK4DC', 'pACDC', 1, 'tAC2B', 'pKON', 1, 'tDC2B', 'pKON', 1, ...
30     'tAC2B', 'pACB', 1, 'tDC2B', 'pDCB', 1, 'tBDC', 'pBDC', 1, 'tBAC', 'pBAC', 1};
```

A.3 Strategi Battery State of Charge

```
1 % PDF dari PLTH (pv - wt1 - wt2 - gen diesel)
2 % File: 'tesis_SB_def.m'
3
4 function [PN_name, set_of_places, set_of_trans, set_of_arcs]...
5     =tesis_SB_def(global_info)
6
7 PN_name = 'Model PN dari PLTH dengan strategi Setting Battery (SB)';
8 set_of_places = {'pPDC','pPAC','pPV','pWT1','pWT2','pBAT','pDIS',...
9     'pDC','pAC','pKON','pDCAC','pACDC','pDCB','pACB','pBDC','pBAC'};
10 set_of_trans = {'tP4DC','tP4AC','tP4BT','tW1DC','tW1AC','tW1BT',...
11     'tW2DC','tW2AC','tW2BT','tB4DC','tB4AC','tD4DC','tD4AC',...
12     'tK4AC','tK4DC','tAC2B','tDC2B','tBDC','tBAC'};
13 set_of_arcs = {'pPDC','tP4DC',1,'pPDC','tW1DC',1,'pPDC','tW2DC',1,...
14     'pPDC','tB4DC',1,'pPDC','tD4DC',1,...
15     'pPAC','tP4AC',1,'pPAC','tW1AC',1,'pPAC','tW2AC',1,...
16     'pPAC','tB4AC',1,'pPAC','tD4AC',1,...
17     'pPV','tP4DC',1,'pPV','tP4AC',1,'pPV','tP4BT',1,...
18     'pWT1','tW1DC',1,'pWT1','tW1AC',1,'pWT1','tW1BT',1,...
19     'pWT2','tW2DC',1,'pWT2','tW2AC',1,'pWT2','tW2BT',1,...
20     'pBAT','tB4DC',1,'pBAT','tB4AC',1,...
21     'pDIS','tD4DC',1,'pDIS','tD4AC',1,...
22     'pKON','tK4AC',1,'pKON','tK4DC',1,'pDC','tK4AC',1,'pAC','tK4DC',1,...
23     'pDCAC','tAC2B',1,'pACDC','tDC2B',1,'pDCB','tBDC',1,'pACB','tBAC',1,...
24     'tP4DC','pDCB',1,'tP4AC','pDC',1,'tP4BT','pBAT',1,'tW1DC','pDCB',1,...
25     'tW2DC','pDCB',1,'tW1AC','pDC',1,'tW2AC','pDC',1,...
26     'tW1BT','pBAT',1,'tW2BT','pBAT',1,...
27     'tB4DC','pAC',1,'tB4AC','pDC',1,...
28     'tD4DC','pAC',1,'tD4AC','pACB',1,'tK4AC','pDCAC',1,...
29     'tK4DC','pACDC',1,'tAC2B','pKON',1,'tDC2B','pKON',1,...
30     'tAC2B','pACB',1,'tDC2B','pDCB',1,'tBDC','pBDC',1,'tBAC','pBAC',1};
```

LAMPIRAN B: Kode Program TDF

B.1 Strategi Load-Following

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tP4BT_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tP4BT ('tP4BT_pre.m')
4
5 if is_enabled(PN,'tP4DC'),
6 fire=0;
7 elseif is_enabled(PN,'tP4AC'),
8 fire=0;
9 else
10 fire=1;
11 end;
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tW1DC_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tW1DC ('tW1DC_pre.m')
4
5 if is_enabled(PN,'tW1AC'),
6 fire=0;
7 else
8 fire=1;
9 end;
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tW2BT_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tW2BT ('tW2BT_pre.m')
4
5 if is_enabled(PN,'tW2DC'),
6 fire=0;
7 elseif is_enabled(PN,'tW2AC'),
8 fire=0;
9 else
10 fire=1;
11 end;
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tB4DC_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tB4DC ('tB4DC_pre.m')
4
5 if is_enabled(PN,'tB4AC'),
6 fire=0;
7 else
8 fire=1;
9 end;
```

B.2 Strategi Cycle-Charging

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tP4BT_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tP4BT ('tP4BT_pre.m')
4
5 if is_enabled(PN,'tP4DC'),
6     fire=0;
7 elseif is_enabled(PN,'tP4AC'),
8     fire=0;
9 else
10    fire=1;
11 end;
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tW1DC_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tW1DC ('tW1DC_pre.m')
4
5 if is_enabled(PN,'tW1AC'),
6     fire=0;
7 elseif is_enabled(PN,'tD4DC'),
8     fire=0;
9 else
10    fire=1;
11 end;
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2 tW2AC_pre(PN,new_color,override,selected_tokens,global_info)
3 % TDF_PRE for tW2DC ('tW2DC_pre.m')
4
5 if is_enabled(PN,'tD4AC'),
6     fire=0;
7 elseif is_enabled(PN,'tD4AC'),
8     fire=0;
9 else
10    fire=1;
11 end;
```

B.3 Strategi Battery State of Charge

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2     tB4AC_pre(PN,new_color,override,selected_tokens,global_info)
3     % TDF_PRE for tB4AC ('tB4AC_pre.m')
4
5 - b1=get_place(PN,'pBAT');
6
7 - fire=(b1.tokens > 7);
```

```
1 function[fire,PN,new_color,override,selected_tokens,global_info]=...
2     tB4DC_pre(PN,new_color,override,selected_tokens,global_info)
3     % TDF_PRE for tB4DC ('tB4DC_pre.m')
4
5 - b1=get_place(PN,'pBAT');
6
7 - fire=(b1.tokens > 7);
```



LAMPIRAN C: Kode Program MSF

C.1 Strategi Load-Following

```
1  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2  % SIMULASI PEMBANGKIT LISTRIK TENAGA HIBRIDA (PLTH) %
3  % PHOTOVOLTAIC - WIND TURBINES - GEN.DIESEL      %
4  % STRATEGI LOAD-FOLLOWING          1 HARI      %
5  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6  clear; clc;
7  n=24; % menentukan jumlah siklus
8  jml_tok_pbbn_ac=0; % menentukan nilai awal utk mjumlahx token prmt bbn ac
9  jml_tok_pbbn_dc=0; % menentukan nilai awal utk mjumlahx token prmt bbn dc
10 jml_tok_PV_tsedia=0; % menentukan nilai awal utk mjumlahx token PV
11 jml_tok_WT1_tsedia=0; % menentukan nilai awal utk mjumlahx token WT1
12 jml_tok_WT2_tsedia=0; % menentukan nilai awal utk mjumlahx token WT2
13
14 %=> PENENTUAN DATA PERMINTAAN BEBAN
15 % data permintaan/kebutuhan beban ac adalah:
16 p_ac=[1;1;1;2;2;2;2;1;1;3;6;6;6;6;4;4;2;1;1;1;1;2;2;2];
17 % data permintaan/kebutuhan beban dc adalah:
18 p_dc=[1;1;1;1;1;1;1;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0];
19
20 for i=1:n
21     % Konversi dari kW menjadi token
22     if (p_ac(i)<1)
23         tp_ac(i)=0;
24     elseif (p_ac(i)>=1 & p_ac(i)<2)
25         tp_ac(i)=1;
26     elseif (p_ac(i)>=2 & p_ac(i)<3)
27         tp_ac(i)=2;
28     elseif (p_ac(i)>=3 & p_ac(i)<4)
29         tp_ac(i)=3;
30     elseif (p_ac(i)>=4 & p_ac(i)<5)
31         tp_ac(i)=4;
32     elseif (p_ac(i)>=5 & p_ac(i)<6)
33         tp_ac(i)=5;
34     elseif (p_ac(i)>=6 & p_ac(i)<7)
35         tp_ac(i)=6;
36     else
37         tp_ac(i)=7;
38     end
39     jml_tok_pbbn_ac=jml_tok_pbbn_ac+tp_ac(i);
40
41     if (p_dc(i)<1)
42         tp_dc(i)=0;
43     else
44         tp_dc(i)=1;
45     end
46     jml_tok_pbbn_dc=jml_tok_pbbn_dc+tp_dc(i);
47 end
```



```

102
103 %==> MENAMPILKAN DATA INTENSITAS RADIASI SURYA DAN KEC.ANGIN
104 % GRAFIK INTENSITAS RADIASI SURYA
105 subplot(2,2,1);plot(int)
106 xlabel('siklus (jam)')
107 ylabel('Intensitas Radiasi (kWh/m2)')
108 title('Data Intensitas Radiasi Surya selama 1 hari','fontsize',11)
109
110 % GRAFIK KECEPATAN ANGIN
111 subplot(2,2,2);plot(kec1)
112 %plot(kec2)
113 xlabel('siklus (jam)')
114 ylabel('Kecepatan (m/dtk)')
115 title('Data Kecepatan Angin selama 1 hari','fontsize',11)
116
117 %==> PENGGABUNGAN VEKTOR keb beban ac (p_ac) dan beban dc (p_dc)
118 daya_bbn=[p_ac p_dc] % gabungan vektor p_ac,p_dc,pv,dan wt
119 subplot(2,2,3);plot(daya_bbn) % menampilkan grafik daya beban ac dan dc(dal
120 %subplot(2,2,3);bar(daya_bbn,'stack') % menampilkan grafik daya beban ac da
121 xlabel('siklus (jam)')
122 ylabel('daya listrik (kW)')
123 legend('ac','dc')
124 title('Data permintaan beban ac dan dc selama 1 hari','fontsize',11)
125
126 %==> PENGGABUNGAN VEKTOR daya pv, wt1, wt2
127 daya_pv=[pv] % vektor pv
128 daya_wt=[wt1] % vektor wt1/wt2
129
130 %==> MENAMPILKAN DATA TOKEN
131 % DATA TOKEN KEBUTUHAN BEBAN AC DAN DC
132 token_bbn=[tp_ac' tp_dc']
133 % DATA TOKEN PV, WT1, DAN WT2
134 token_sbr=[t_pv' t_wt1' t_wt2']
135
136 token_bat(1)=0; % initial marking utk baterai (pBAT)
137 tok_pasokan(i)=0; % inialisasi token pasokan daya
138 tok_kebutuhan(i)=0; % inialisasi token kebutuhan beban
139 tok_kurang(i)=0; % inialisasi kekurangan token
140 jml_tok_diesel_tpakai=0; % nilai awal utk mjumlahx token-diesel
141 for i=1:n % proses looping sejumlah baris 'data'
142 %%% STATIC DETAILS %%%
143 png = petrinetgraph('tesis_LF_def');
144
145 %%% DYNAMIC DETAILS %%%
146 dyn.initial_markings={'pPAC',tp_ac(i),'pPDC',tp_dc(i),'pPV',t_pv(i),...
147 'pWT1',t_wt1(i),'pWT2',t_wt2(i),...
148 'pBAT',token_bat(i),'pDIS',4,'pKON',1};
149 dyn.firing_times={'tP4DC',10,'tP4AC',10,'tP4BT',10,'tW1DC',10,...
150 'tW1AC',10,'tW1BT',10,'tW2DC',10,'tW2AC',10,'tW2BT',10,...
151 'tB4DC',10,'tB4AC',10,'tD4DC',10,'tD4AC',10,...
152 'tK4AC',10,'tK4DC',10,'tAC2B',10,'tDC2B',10,'tBDC',10,'tBAC',10};
153

```

```

154 % Perhitungan Kecukupan pasokan daya
155 tok_pasokan(i)=t_pv(i)+t_wt1(i)+t_wt2(i)+token_bat(i)+4;
156 tok_kebutuhan(i)=tp_ac(i)+tp_dc(i);
157 tok_kurang(i)=tok_kebutuhan(i)-tok_pasokan(i);
158 if tok_pasokan(i) < tok_kebutuhan(i)
159     disp(['Ketersediaan diesel = ',num2str(4),' kW'])
160     disp(['Ketersediaan pv = ',num2str(t_pv(i)),' kW'])
161     disp(['Ketersediaan wt1 = ',num2str(t_wt1(i)),' kW'])
162     disp(['Ketersediaan wt2 = ',num2str(t_wt2(i)),' kW'])
163     disp(['Ketersediaan di baterai = ',num2str(token_bat(i)),' kW'])
164     disp(['Kebutuhan daya beban = ',num2str(tok_kebutuhan(i)),' kW'])
165     disp('PASOKAN DAYA TIDAK CUKUP')
166     disp(['TERDAPAT KEKURANGAN SEBESAR ',num2str(tok_kurang(i)),' kW'])
167 end
168
169 %%% SIMULATE %%%
170 RES=gpensim(png, dyn);
171 print_statespace(RES);
172 duration_matrix=extractt(RES,{'tP4BT','tW1BT','tW2BT','tD4BT'})
173 occupancy_matrix=occupancy(RES,{'tP4BT','tW1BT','tW2BT','tD4BT'})
174
175 % Penentuan token baterai untuk siklus berikutnya
176 [m1,n1]=size(RES.LOG);
177 token_bat(i+1)=RES.LOG(m1,5);
178
179 if token_bat(i+1) > 10 % membatasi jmlh token baterai (max=40)
180     token_bat(i+1)=10;
181 else
182     token_bat(i+1)=token_bat(i+1);
183 end
184
185 % Perhitungan token GEN.DIESEL yg terpakai
186 token_diesel_tpakai(i)=4-RES.LOG(m1,10)
187 jml_tok_diesel_tpakai=jml_tok_diesel_tpakai+token_diesel_tpakai(i);
188
189 % Melihat daya listrik di beban ac dan dc
190 tok_ac(i)=RES.LOG(m1,4); % di beban ac (dlm token)
191 daya_ac(i)=tok_ac(i); % di beban ac (dlm kW)
192 tok_dc(i)=RES.LOG(m1,6); % di beban dc (dlm token)
193 daya_dc(i)=tok_dc(i); % di beban dc (dlm kW)
194 end
195 beban=[daya_ac' daya_dc']
196 subplot(2,2,4);plot(beban)
197 xlabel('siklus (jam)')
198 ylabel('daya listrik (kW)')
199 legend('ac','dc')
200 title('Daya yang sampai di beban ac dan dc selama 1 hari','fontsize',11)
201
202 token_bat
203 jml_tok_pbbn_ac % jumlah token permintaan beban ac dlm n jam
204 jml_tok_pbbn_dc % jumlah token permintaan beban dc dlm n jam
205 jml_tok_diesel_tpakai % jumlah token diesel yg terpakai dlm n jam

```

C.2 Strategi Cycle-Charging

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 % SIMULASI PEMBANGKIT LISTRIK TENAGA HIBRIDA (PLTH) %
3 % PHOTOVOLTAIC - WIND TURBINES - GEN.DIESEL %
4 % STRATEGI CYCLE-CHARGING 1 HARI %
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 clear; clc;
7 n=24; % menentukan jumlah siklus
8 jml_tok_pbbn_ac=0; % menentukan nilai awal utk mjumlahx token prmt bbn ac
9 jml_tok_pbbn_dc=0; % menentukan nilai awal utk mjumlahx token prmt bbn dc
10 jml_tok_PV_tsedia=0; % menentukan nilai awal utk mjumlahx token PV
11 jml_tok_WT1_tsedia=0; % menentukan nilai awal utk mjumlahx token WT1
12 jml_tok_WT2_tsedia=0; % menentukan nilai awal utk mjumlahx token WT2
13
14 %==> PENENTUAN DATA PERMINTAAN BEBAN
15 % data permintaan/kebutuhan beban ac adalah:
16 p_ac=[1;1;1;2;2;2;1;1;3;6;6;6;6;4;4;2;1;1;1;2;2;2];
17 % data permintaan/kebutuhan beban dc adalah:
18 p_dc=[1;1;1;1;1;1;1;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0];
19
20 for i=1:n
21 % Konversi dari kW menjadi token
22 if (p_ac(i)<1)
23 tp_ac(i)=0;
24 elseif (p_ac(i)>=1 & p_ac(i)<2)
25 tp_ac(i)=1;
26 elseif (p_ac(i)>=2 & p_ac(i)<3)
27 tp_ac(i)=2;
28 elseif (p_ac(i)>=3 & p_ac(i)<4)
29 tp_ac(i)=3;
30 elseif (p_ac(i)>=4 & p_ac(i)<5)
31 tp_ac(i)=4;
32 elseif (p_ac(i)>=5 & p_ac(i)<6)
33 tp_ac(i)=5;
34 elseif (p_ac(i)>=6 & p_ac(i)<7)
35 tp_ac(i)=6;
36 else
37 tp_ac(i)=7;
38 end
39 jml_tok_pbbn_ac=jml_tok_pbbn_ac+tp_ac(i);
40
41 if (p_dc(i)<1)
42 tp_dc(i)=0;
43 else
44 tp_dc(i)=1;
45 end
46 jml_tok_pbbn_dc=jml_tok_pbbn_dc+tp_dc(i);
47 end
```



```

102
103 %==> MENAMPILKAN DATA INTENSITAS RADIASI SURYA DAN KEC.ANGIN
104 % GRAFIK INTENSITAS RADIASI SURYA
105 subplot(2,2,1);plot(int)
106 xlabel('siklus (jam)')
107 ylabel('Intensitas Radiasi (kWh/m2)')
108 title('Data Intensitas Radiasi Surya selama 1 hari','fontsize',11)
109
110 % GRAFIK KECEPATAN ANGIN
111 subplot(2,2,2);plot(kec1)
112 %plot(kec2)
113 xlabel('siklus (jam)')
114 ylabel('Kecepatan (m/dtk)')
115 title('Data Kecepatan Angin selama 1 hari','fontsize',11)
116
117 %==> PENGGABUNGAN VEKTOR keb beban ac (p_ac) dan beban dc (p_dc)
118 daya_bbn=[p_ac p_dc] % gabungan vektor p_ac,p_dc,pv,dan wt
119 subplot(2,2,3);plot(daya_bbn) % menampilkan grafik daya beban ac dan dc(dal
120 %subplot(2,2,3);bar(daya_bbn,'stack') % menampilkan grafik daya beban ac da
121 xlabel('siklus (jam)')
122 ylabel('daya listrik (kW)')
123 legend('ac','dc')
124 title('Data permintaan beban ac dan dc selama 1 hari','fontsize',11)
125
126 %==> PENGGABUNGAN VEKTOR daya pv, wt1, wt2
127 daya_pv=[pv] % vektor pv
128 daya_wt=[wt1] % vektor wt1/wt2
129
130 %==> MENAMPILKAN DATA TOKEN
131 % DATA TOKEN KEBUTUHAN BEBAN AC DAN DC
132 token_bbn=[tp_ac' tp_dc']
133 % DATA TOKEN PV, WT1, DAN WT2
134 token_sbr=[t_pv' t_wt1' t_wt2']
135
136 token_bat(1)=2; % initial marking utk baterai (pBAT)
137 jml_tok_diesel_tpakai=0; % nilai awal utk mjumlahx token diesel
138 for i=1:n % proses looping sejumlah baris 'data'
139 %%% STATIC DETAILS %%%
140 png = petrinetgraph('tesis_CC_def');
141
142 %%% DYNAMIC DETAILS %%%
143 dyn.initial_markings={'pPAC',tp_ac(i),'pPDC',tp_dc(i),'pPV',t_pv(i),...
144 'pWT1',t_wt1(i),'pWT2',t_wt2(i),...
145 'pBAT',token_bat(i),'pDIS',4,'pKON',1};
146 dyn.firing_times={'tP4DC',10,'tP4AC',10,'tP4BT',10,'tW1DC',10,...
147 'tW1AC',10,'tW1BT',10,'tW2DC',10,'tW2AC',10,'tW2BT',10,...
148 'tB4DC',10,'tB4AC',10,'tD4DC',10,'tD4AC',10,...
149 'tK4AC',10,'tK4DC',10,'tAC2B',10,'tDC2B',10,'tBDC',10,'tBAC',10};
150

```

```

151 %%% SIMULATE %%%
152 - RES=gpsim(png, dyn);
153 - print_statespace(RES);
154 - extractp(RES, {'pPAC','pPDC','pPV','pWT1','pWT2','pBAT',...
155 - 'pDIS','pAC','pDC','pDCAC','pKON','pACDC','pACB','pDCB',...
156 - 'pBAC','pBDC'})
157
158 % Penentuan token baterai untuk siklus berikutnya
159 - [m1,n1]=size(RES.LOG);
160 - token_bat(i+1)=RES.LOG(m1,5);
161
162 - if token_bat(i+1) > 10 % membatasi jmlh token baterai (max=40)
163 - token_bat(i+1)=10;
164 - else
165 - token_bat(i+1)=token_bat(i+1);
166 - end
167
168 % Perhitungan token GEN.DIESEL yg terpakai
169 - token_diesel_tpakai(i)=4-RES.LOG(m1,10)
170 - jml_tok_diesel_tpakai=jml_tok_diesel_tpakai+token_diesel_tpakai(i);
171
172 % Melihat daya listrik di beban ac dan dc
173 - tok_ac(i)=RES.LOG(m1,4); % di beban ac (dlm token)
174 - daya_ac(i)=tok_ac(i); % di beban ac (dlm kW)
175 - tok_dc(i)=RES.LOG(m1,6); % di beban dc (dlm token)
176 - daya_dc(i)=tok_dc(i); % di beban dc (dlm kW)
177 - end
178 - beban=[daya_ac' daya_dc']
179 - subplot(2,2,4);plot(beban)
180 - xlabel('siklus (jam)')
181 - ylabel('daya listrik (kW)')
182 - legend('ac','dc')
183 - title('Daya yang sampai di beban ac dan dc selama 1 hari','fontsize',11)
184
185 - token_bat
186 - jml_tok_pbbn_ac % jumlah token permintaan beban ac dlm n jam
187 - jml_tok_pbbn_dc % jumlah token permintaan beban dc dlm n jam
188 - jml_tok_PV_tsedia % jumlah token photovoltaic dlm n jam
189 - jml_tok_WT1_tsedia % jumlah token wind turbinel beban dlm n jam
190 - jml_tok_WT2_tsedia % jumlah token wind turbine2 beban dlm n jam
191 - jml_tok_diesel_tpakai % jumlah token diesel yg terpakai dlm n jam

```

C.3 Strategi Battery State of Charge

```
1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 % SIMULASI PEMBANGKIT LISTRIK TENAGA HIBRIDA (PLTH) %
3 % PHOTOVOLTAIC - WIND TURBINES - GEN.DIESEL %
4 % STRATEGI SETTING BATTERY 1 HARI %
5 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
6 clear; clc;
7 n=24; % menentukan jumlah siklus
8 jml_tok_pbbn_ac=0; % menentukan nilai awal utk mjumlahx token prmt bbn ac
9 jml_tok_pbbn_dc=0; % menentukan nilai awal utk mjumlahx token prmt bbn dc
10 jml_tok_PV_tsedia=0; % menentukan nilai awal utk mjumlahx token PV
11 jml_tok_WT1_tsedia=0; % menentukan nilai awal utk mjumlahx token WT1
12 jml_tok_WT2_tsedia=0; % menentukan nilai awal utk mjumlahx token WT2
13
14 %==> PENENTUAN DATA PERMINTAAN BEBAN
15 % data permintaan/kebutuhan beban ac adalah:
16 p_ac=[1;1;1;2;2;2;1;1;3;6;6;6;6;4;4;2;1;1;1;1;1;2;2;2];
17 % data permintaan/kebutuhan beban dc adalah:
18 p_dc=[1;1;1;1;1;1;1;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0;0];
19
20 for i=1:n
21 % Konversi dari kW menjadi token
22 if (p_ac(i)<1)
23 tp_ac(i)=0;
24 elseif (p_ac(i)>=1 & p_ac(i)<2)
25 tp_ac(i)=1;
26 elseif (p_ac(i)>=2 & p_ac(i)<3)
27 tp_ac(i)=2;
28 elseif (p_ac(i)>=3 & p_ac(i)<4)
29 tp_ac(i)=3;
30 elseif (p_ac(i)>=4 & p_ac(i)<5)
31 tp_ac(i)=4;
32 elseif (p_ac(i)>=5 & p_ac(i)<6)
33 tp_ac(i)=5;
34 elseif (p_ac(i)>=6 & p_ac(i)<7)
35 tp_ac(i)=6;
36 else
37 tp_ac(i)=7;
38 end
39 jml_tok_pbbn_ac=jml_tok_pbbn_ac+tp_ac(i);
40
41 if (p_dc(i)<1)
42 tp_dc(i)=0;
43 else
44 tp_dc(i)=1;
45 end
46 jml_tok_pbbn_dc=jml_tok_pbbn_dc+tp_dc(i);
47 end
```



```

102
103 %==> MENAMPILKAN DATA INTENSITAS RADIASI SURYA DAN KEC.ANGIN
104 % GRAFIK INTENSITAS RADIASI SURYA
105 subplot(2,2,1);plot(int)
106 xlabel('siklus (jam)')
107 ylabel('Intensitas Radiasi (kWh/m2)')
108 title('Data Intensitas Radiasi Surya selama 1 hari','fontsize',11)
109
110 % GRAFIK KECEPATAN ANGIN
111 subplot(2,2,2);plot(kec1)
112 %plot(kec2)
113 xlabel('siklus (jam)')
114 ylabel('Kecepatan (m/dtk)')
115 title('Data Kecepatan Angin selama 1 hari','fontsize',11)
116
117 %==> PENGGABUNGAN VEKTOR keb beban ac (p_ac) dan beban dc (p_dc)
118 daya_bbn=[p_ac p_dc] % gabungan vektor p_ac,p_dc,pv,dan wt
119 subplot(2,2,3);plot(daya_bbn) % menampilkan grafik daya beban ac dan dc (dal
120 %subplot(2,2,3);bar(daya_bbn,'stack') % menampilkan grafik daya beban ac da
121 xlabel('siklus (jam)')
122 ylabel('daya listrik (kW)')
123 legend('ac','dc')
124 title('Data permintaan beban ac dan dc selama 1 hari','fontsize',11)
125
126 %==> PENGGABUNGAN VEKTOR daya pv, wt1, wt2
127 daya_pv=[pv] % vektor pv
128 daya_wt=[wt1] % vektor wt1/wt2
129
130 %==> MENAMPILKAN DATA TOKEN
131 % DATA TOKEN KEBUTUHAN BEBAN AC DAN DC
132 token_bbn=[tp_ac' tp_dc']
133 % DATA TOKEN PV, WT1, DAN WT2
134 token_sbr=[t_pv' t_wt1' t_wt2']
135
136 token_bat(1)=7; % initial marking utk baterai (pBAT)
137 jml_tok_diesel_tpakai=0; % nilai awal utk mjumlahx token diesel
138 for i=1:n % proses looping sejumlah baris 'data'
139 %%%% STATIC DETAILS %%%%
140 png = petrinetgraph('tesis_SB_def');
141
142 %%%% DYNAMIC DETAILS %%%%
143 dyn.initial_markings={'pPAC',tp_ac(i),'pPDC',tp_dc(i),'pPV',t_pv(i),...
144 'pWT1',t_wt1(i),'pWT2',t_wt2(i),...
145 'pBAT',token_bat(i),'pDIS',4,'pKON',1);
146 dyn.firing_times={'tP4DC',10,'tP4AC',10,'tP4BT',10,'tW1DC',10,...
147 'tW1AC',10,'tW1BT',10,'tW2DC',10,'tW2AC',10,'tW2BT',10,...
148 'tB4DC',10,'tB4AC',10,'tD4DC',10,'tD4AC',10,...
149 'tK4AC',10,'tK4DC',10,'tAC2B',10,'tDC2B',10,'tBDC',10,'tBAC',10};
150

```

```

151 %%%% SIMULATE %%%%
152 - RES=gpensim(png, dyn);
153 - print_statespace(RES);
154 - extractp(RES, {'pPAC','pPDC','pPV','pWT1','pWT2','pBAT',...
155 - 'pDIS','pAC','pDC','pDCAC','pKON','pACDC','pACB','pDCB',...
156 - 'pBAC','pBDC'})
157
158 % Penentuan token baterai untuk siklus berikutnya
159 - [m1,n1]=size(RES.LOG);
160 - token_bat(i+1)=RES.LOG(m1,5);
161
162 - if token_bat(i+1) > 10 % membatasi jmlh token baterai (max=40)
163 - token_bat(i+1)=10;
164 - else
165 - token_bat(i+1)=token_bat(i+1);
166 - end
167
168 % Perhitungan token GEN.DIESEL yg terpakai
169 - token_diesel_tpakai(i)=4-RES.LOG(m1,10)
170 - jml_tok_diesel_tpakai=jml_tok_diesel_tpakai+token_diesel_tpakai(i);
171
172 % Melihat daya listrik di beban ac dan dc
173 - tok_ac(i)=RES.LOG(m1,4); % di beban ac (dlm token)
174 - daya_ac(i)=tok_ac(i); % di beban ac (dlm kW)
175 - tok_dc(i)=RES.LOG(m1,6); % di beban dc (dlm token)
176 - daya_dc(i)=tok_dc(i); % di beban dc (dlm kW)
177 - end
178 - beban=[daya_ac' daya_dc']
179 - subplot(2,2,4);plot(beban)
180 - xlabel('siklus (jam)')
181 - ylabel('daya listrik (kW)')
182 - legend('ac','dc')
183 - title('Daya yang sampai di beban ac dan dc selama 1 hari','fontsize',11)
184
185 - token_bat
186 - jml_tok_pbbn_ac % jumlah token permintaan beban ac dlm n jam
187 - jml_tok_pbbn_dc % jumlah token permintaan beban dc dlm n jam
188 - jml_tok_PV_tsedia % jumlah token photovoltaic dlm n jam
189 - jml_tok_WT1_tsedia % jumlah token wind turbinel beban dlm n jam
190 - jml_tok_WT2_tsedia % jumlah token wind turbine2 beban dlm n jam
191 - jml_tok_diesel_tpakai % jumlah token diesel yg terpakai dlm n jam

```