



TUGAS AKHIR - TE 141599

**MULTI OBJEKTIF OPTIMAL POWER FLOW
UNTUK MINIMISASI BIAYA DAN RUGI - RUGI JARINGAN
MENGUNAKAN GRAVITATIONAL SEARCH ALGORITHM**

Rizki Nurfajri Adhim
NRP 2210100169

Dosen Pembimbing
Prof. Ir. Ontoseno Penangsang, M.Sc, Ph.D
Yun Tonce Kusuma Priyanto, ST, MT

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh November
Surabaya 2015



FINAL PROJECT - TE 141599

**MULTI OBJECTIVE OPTIMAL POWER FLOW
TO MINIMIZE GENERATION COST AND LOSSES
USING GRAVITATIONAL SEARCH ALGORITHM**

Rizki Nurfajri Adhim
NRP 2210100169

Advisor
Prof. Ir. Ontoseno Penangsang, M.Sc., Ph.D
Yun Tonce Kusuma Priyanto, ST, MT

JURUSAN TEKNIK ELEKTRO
Fakultas Teknologi Industri
Institut Teknologi Sepuluh November
Surabaya 2015

**MULTI OBJEKTIF OPTIMAL POWER FLOW
UNTUK MINIMISASI BIAYA DAN RUGI – RUGI JARINGAN
MENGUNAKAN GRAVITATIONAL SEARCH ALGORITHM**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Teknik
Pada
Bidang Studi Teknik Sistem Tenaga
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember**

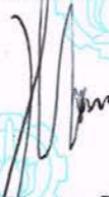
Menyetujui:

Dosen Pembimbing I,

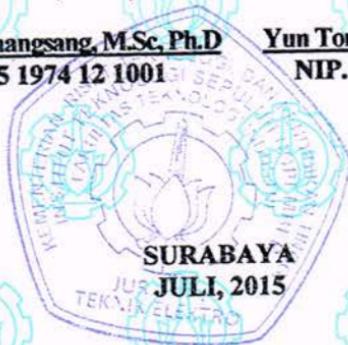


Prof. Ir. Ontoseno Penangsang, M.Sc. Ph.D
NIP. 1949 07 15 1974 12 1001

Dosen Pembimbing II,



Yun Tonce Kusuma Priyanto, ST, MT
NIP. 1984 06 16 2012 12 1001



ABSTRAK

Pengaturan optimal pembangkit telah menjadi hal yang sangat penting dalam studi sistem tenaga modern. Banyak sekali metode dan algoritma yang telah diciptakan untuk memungkinkan pengaturan pembangkitan menjadi optimal. Tujuan dari algoritma dan metode yang dibuat adalah untuk memaksimalkan hasil optimasi baik dari segi biaya, rugi-rugi jaringan maupun mengurangi emisi yang dihasilkan. Dalam pengembangan algoritma dan metode yang telah dibuat diantaranya banyak yang menggunakan algoritma berbasis populasi, seperti *Particle Swarm Optimization* (PSO) dan *Gravitational Search Algorithm* (GSA).

Keduanya sama-sama memanfaatkan sifat partikel yang cenderung membuat kelompok-kelompok berdasarkan solusi terbaiknya. Namun terdapat perbedaan sifat pada bagaimana partikel-partikel saling mendekati. Pada metode *Gravitational Search Algorithm* digunakan teori hukum Newton tentang gravitasi, dimana setiap partikel memiliki gravitasinya masing-masing berdasarkan massa dan jaraknya dengan partikel lain.

Kata kunci : Multi Objective Optimal Power Flow, Economic Dispatch, Losses, Gravitational Search Algorithm

Halaman ini sengaja dikosongkan

ABSTRACT

Optimization becomes an important thing in modern power system. There are several methods and algorithm has created to enable very optimal generation. Many algorithm and methods has created for many purposes, such maximize optimization result with optimizing generation cost (economic dispatch), system losses, and minimize the emission of the generator for environmental purpose. In research of the methods and algorithm has made, there were many of them using population – based algorithm, such as *Particle Swarm Optimization* (PSO) and *Gravitational Search Algorithm* (GSA).

Both has several similarity in its characteristics, made of many particle nature that makes several groups to obtain the best result. But there are several differences in particles nature about how they approaching each other. In *Gravitational Search Algorithm* method, its use Newton's law of gravity, that each particles has its own gravity based on its mass and distance with other particles.

Keywords : *Multi Objective Optimal Power Flow, Economic Dispatch, Losses, Gravitational Search Algorithm*

Halaman ini sengaja dikosongkan

KATA PENGANTAR

Puji syukur kehadiran Tuhan Yang Maha Esa atas rahmat, taufik, dan hidayah-Nya sehingga Tugas Akhir ini dapat terselesaikan. Dengan dibuatnya laporan Tugas Akhir ini, penulis telah menyelesaikan Tugas Akhir yang berjudul **“MULTI OBJECTIVE OPTIMAL POWER FLOW UNTUK MINIMISASI BIAYA DAN RUGI – RUGI JARINGAN DENGAN MENGGUNAKAN GRAVITATIONAL SEARCH ALGORITHM”**. Tugas Akhir ini bersifat penelitian ilmiah dan diselesaikan selama satu semester. Dengan dibuatnya buku Tugas Akhir ini, dilakukan penelitian dengan sumber jurnal dan buku nasional maupun internasional dan menghubungkannya dengan teori yang telah didapatkan penulis semasa kuliah dalam sepuluh semester.

Buku laporan Tugas Akhir ini dapat terselesaikan atas bantuan banyak pihak yang telah membantu baik secara langsung maupun tidak langsung. Oleh karena itu, penulis mengucapkan terima kasih kepada :

1. Allah SWT, Tuhan Yang Maha Esa, yang telah memberikan kehidupan, kesehatan baik jasmani maupun rohani, dan senantiasa menuntun penulis menuju ke jalan yang benar.
2. Ayahku Mundakir, Ibuku Nindyarini Peni Hidayanti dan saudaraku, Muhammad Ismad Hanif dan Muhammad Abu-Dzar Al Ghifari yang telah banyak memberikan dorongan, baik dalam hal spiritual maupun material dalam penyelesaian buku Tugas Akhir ini.
3. Bapak Ontoseno Penangsang selaku dosen pembimbing pertama serta Bapak Yun Tonce Kusuma Priyanto sebagai dosen pembimbing kedua yang telah banyak memberikan masukan dan juga arahan sehingga Tugas Akhir ini dapat terselesaikan dengan baik.
4. Bapak-ibu dosen pengajar Bidang Studi Teknik Sistem Tenaga atas pengajaran, bimbingan, serta perhatian yang diberikan selama ini.
5. Teman – teman seperjuangan : Herbanindya Febri Trusatmaji, Rizki Achmad Tri Rachmanto, Ivas Muzammil, dan Lukman Hendarwin atas waktu, ide – ide, semangat dan dukungan yang membuat penulis selalu termotivasi dan tetap menjaga api semangat untuk melanjutkan kuliah dan Tugas Akhir.

6. Teman – teman diluar kuliah, yang senantiasa menemani diluar jam kuliah : Angger Bagas Prakoso, Ivan Ahmed Giovanni, Bagasworo Rinugroho, Ramy Yahya, Rachmad Priambodo, dan teman – teman dekat penulis lainnya yang terus memotivasi dan menemani penulis sehingga terus bersemangat.
7. Fini Prisilia Dewi yang telah bersedia meluangkan segenap waktu dan selalu memotivasi penulis mulai dari awal kuliah hingga penyelesaian Tugas Akhir.
8. Seluruh rekan – rekan angkatan 2009, 2010, dan 2011 atas kebersamaan dan kerjasamanya selama ini. Khususnya pada teman – teman angkatan 2010.
9. Seluruh rekan – rekan asisten dan anggota Laboratorium Simulasi Sistem Tenaga Listrik (B103) atas kerjasama, waktu, dan kebersamaan dalam proses pengerjaan Tugas Akhir.
10. Seluruh rekan – rekan asisten Laboratorium Teknik Komputer dan Telematika (B201) atas kebersamaan dan ilmu yang diberikan, serta kebersamaan untuk singgah di bidang studi Teknik Komputer dan Telematika.
11. Seluruh keluarga besar Teknik Elektro ITS, para dosen, karyawan, mahasiswa, juga rekan – rekan HIMATEKTRO atas dukungan, masukan, dan kerjasamanya selama masa perkuliahan dan proses pengerjaan Tugas Akhir.
12. Semua pihak yang ikut membantu penulis dalam pelaksanaan dan penyusunan laporan Tugas Akhir yang tidak dapat disebutkan satu per satu.

Besar harapan penulis bahwa buku Tugas Akhir ini dapat memberikan informasi dan manfaat bagi mahasiswa Teknik Elektro ITS pada khususnya dan seluruh pembaca pada umumnya. Jika ada kesalahan, penulis mengharapkan adanya kritik, koreksi, serta saran yang membangun dari pembaca untuk pengembangan kearah yang lebih baik.

Surabaya, Juli 2015

Penulis

DAFTAR ISI

Pernyataan Keaslian Tugas Akhir	vii
Lembar Pengesahan	ix
Abstrak	xi
Abstract	xiii
Kata Pengantar	xv
Daftar Isi	xvii
Daftar Gambar	xxi
Daftar Tabel	xxiii
BAB 1 Pendahuluan.....	1
1. 1. Latar Belakang	1
1. 2. Rumusan Permasalahan.....	2
1. 3. Batasan Permasalahan	2
1. 4. Tujuan Permasalahan	3
1. 5. Sistematika Penulisan.....	3
BAB 2 Sistem Tenaga Listrik dan Studi Aliran Daya.....	5
2. 1. Sistem Tenaga Listrik	5
2. 1. 1. Pembangkit Tenaga Listrik	5
2. 1. 2. Transmisi dan Distribusi	7
2. 1. 3. Beban	9
2. 2. Analisis Aliran Daya	9
2. 3. Permodelan Matematika Aliran Daya	11
2. 3. 1. Daya Kompleks dalam Sistem Tenaga Listrik	13
2. 3. 2. Pencarian Akar dengan Metode Newton Rhapson	14
2. 4. Optimal Power Flow	16
2. 4. 1. <i>Economic Dispatch</i>	17
2. 4. 2. <i>Power Loss Minimization</i>	17
2. 5. Algoritma <i>Particle Swarm Optimization</i>	18
2. 6. <i>Gravitational Search Algorithm</i>	19
BAB 3 Penyelesaian Masalah Optimasi dengan Optimal Power Flow. .	23

3. 1. Perencanaan Penelitian	23
3. 2. Pencarian Pustaka dan Literatur.....	24
3. 3. Pencarian dan Pengolahan Data	24
3. 4. Penentuan Fungsi Objektif.....	24
3. 4. 1. Fungsi <i>Economic Dispatch</i> (Biaya Pembangkitan Ekonomis)	25
3. 4. 2. Fungsi <i>Power Loss Minimization</i> (Rugi – rugi Jaringan)	25
3. 5. Menentukan Batasan – Batasan (<i>Constraint</i>).....	26
3. 5. 1. Batasan Kapasitas Daya Pembangkit	26
3. 5. 2. Keseimbangan Daya	26
3. 6. Pembagian Fungsi Objektif dengan Metode <i>Weighted Sum</i>	27
3. 7. Simulasi <i>Gravitational Search Algorithm</i> (GSA)	28
3. 7. 1. Inisialisasi Ruang Pencarian (<i>Space Bound</i>) dan <i>Agent Mass</i>	30
3. 7. 2. <i>Active – Passive Gravitation</i> , dan <i>Agent Movement</i>	30
3. 8. Simulasi Particle Swarm Optimization (PSO)	32
3. 8. 1. Inisialisasi Individu dan Perhitungan Awal Algoritma	33
3. 8. 2. Pergerakan dan Posisi Partikel.....	33
3. 8. 3. Penentuan Nilai Akhir Partikel	34
BAB 4 Simulasi dan Analisis	37
4. 1. Data IEEE 30-Bus.....	37
4. 2. Simulasi <i>Economic Dispatch</i>	42
4. 2. 1. Simulasi <i>Economic Dispatch</i> Metode <i>Gravitational Search</i> <i>Algorithm</i>	42
4. 2. 2. Simulasi <i>Economic Dispatch</i> Metode <i>Particle Swarm</i> <i>Optimization</i>	44
4. 2. 3. Perbandingan Simulasi <i>Economic Dispatch Gravitational</i> <i>Search Algorithm</i> dan <i>Particle Swarm Optimization</i>	47
4. 3. Simulasi <i>Power Loss Minimization</i>	48
4. 3. 1. Simulasi <i>Power Loss Minimization</i> Metode <i>Gravitational</i> <i>Search Algorithm</i>	48

4. 3. 2. Simulasi <i>Power Loss Minimization</i> Metode <i>Particle Swarm Optimization</i>	49
4. 3. 3. Perbandingan Simulasi <i>Losses Minimization Gravitational Search Algorithm</i> dan <i>Particle Swarm Optimization</i>	51
4. 4. Simulasi <i>Economic Power Loss Minimization</i>	52
4. 4. 1. Simulasi <i>Economic Power Loss Minimization</i> Metode <i>Gravitational Search Algorithm</i>	52
4. 4. 2. Simulasi <i>Economic Power Loss Minimization</i> Metode <i>Particle Swarm Optimization</i>	54
4. 4. 3. Perbandingan Hasil <i>Economic Power Loss Gravitational Search Algorithm</i> dan <i>Particle Swarm Optimization</i>	56
BAB 5 Kesimpulan dan Saran	59
5. 1. Kesimpulan	59
5. 2. Saran.....	60
Daftar Pustaka	61
Lampiran	63
Riwayat Penulis	65

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 4.1.	Data bus IEEE 30-bus	37
Tabel 4.2.	Data saluran IEEE 30-bus	39
Tabel 4.3.	Data generator IEEE 30-bus.....	40
Tabel 4.4.	Setting tap transformator IEEE 30-bus.....	40
Tabel 4.5.	Data Capacitor Bank IEEE 30-bus	41
Tabel 4.6.	Data fungsi biaya pembangkit sistem IEEE 30-bus	41
Tabel 4.7.	Data spesifikasi CPU.....	41
Tabel 4.8.	Parameter Gravitational Search Algorithm	42
Tabel 4.9.	Daya dan biaya pembangkitan hasil economic dispatch dengan metode Gravitational Search Algorithm	43
Tabel 4.10.	Total daya, losses, dan biaya pembangkitan metode Gravitational Search Algorithm	43
Tabel 4.11.	Parameter Algoritma Particle Swarm Optimization	44
Tabel 4.12.	Daya dan biaya pembangkitan hasil simulasi economic dispatch dengan metode Particle Swarm Optimization	45
Tabel 4.13.	Total daya dan biaya pembangkitan simulasi economic dispatch dengan metode Particle Swarm Optimization	46
Tabel 4.14.	Perbandingan nilai fitness dan biaya pembangkitan simulasi economic dispatch metode Gravitational Search Algorithm dan Particle Swarm Optimization.....	47
Tabel 4.15.	Total daya, biaya pembangkitan dan losses hasil simulasi power loss minimization dengan metode Gravitational Search Algorithm	48
Tabel 4.16.	Total daya, biaya pembangkitan dan losses hasil simulasi power loss minimization dengan menggunakan metode Particle Swarm Optimization	50
Tabel 4.17.	Perbandingan nilai fitness dan nilai losses metode Gravitational Search Algorithm dan Particle Swarm Optimization.....	51

Tabel 4.18. Daya aktif dan biaya pembangkitan simulasi Economic Power Loss Minimization menggunakan metode Gravitational Search Algorithm	53
Tabel 4.19. Daya aktif, losses, dan biaya pembangkitan hasil simulasi Economic Power Loss Minimization menggunakan metode Gravitational Search Algorithm	54
Tabel 4.20. Daya aktif dan biaya pembangkitan hasil simulasi Economic Power Loss Minimization menggunakan metode Particle Swarm Optimization	55
Tabel 4.21. Daya aktif, losses, dan biaya pembangkitan hasil simulasi Economic Power Loss Minimization menggunakan metode Particle Swarm Optimization	56
Tabel 4.22. Perbandingan hasil simulasi Economic Power Loss Minimization metode Gravitational Search Algorithm dan Particle Swarm Optimization	56

DAFTAR GAMBAR

Gambar 2.1.	Komponen dasar dalam sistem tenaga listrik	11
Gambar 2.2.	Sistem tenaga listrik sederhana dengan 3 bus	12
Gambar 2.3.	Permisalan pencarian akar persamaan dengan metode Newton Rhapson	15
Gambar 3.1.	Diagram alir pengerjaan Tugas Akhir	23
Gambar 3.2.	Diagram alir algoritma Gravitational Search Algorithm	29
Gambar 3.3.	Algoritma Particle Swarm Optimization dalam diagram alir	32
Gambar 4.1.	Grafik konvergensi economic dispatch menggunakan metode Gravitational Search Algorithm.....	44
Gambar 4.2.	Grafik konvergensi simulasi economic dispatch metode Particle Swarm Optimization	46
Gambar 4.3.	Perbandingan grafik konvergensi nilai fitness simulasi economic dispatch Gravitational Search Algorithm dan Particle Swarm Optimization	47
Gambar 4.4.	Grafik konvergensi nilai fitness simulasi power loss minimization dengan metode Gravitational Search Algorithm	49
Gambar 4.5.	Grafik konvergensi nilai fitness simulasi power loss minimization dengan metode Particle Swarm Optimization.....	50
Gambar 4.6.	Perbandingan grafik konvergensi nilai fitness simulasi power loss minimization dengan metode Gravitational Search Algorithm dan Particle Swarm Optimization ...	51
Gambar 4.7.	Grafik konvergensi simulasi Economic Power Loss Minimization dengan menggunakan metode Gravitational Search Algorithm	53
Gambar 4.8.	Grafik konvergensi simulasi Economic Power Loss Minimization menggunakan metode Particle Swarm Optimization.....	55

Gambar 4.9. Perbandingan grafik konvergensi hasil simulasi Economic Power Loss Minimization menggunakan metode Gravitational Search Algorithm dan Particle Swarm Optimization57

BAB 1

PENDAHULUAN

1. 1. Latar Belakang

Dalam studi sistem tenaga listrik modern optimasi telah menjadi hal yang sangat penting. Hal-hal yang berhubungan dengan optimasi seperti biaya pembangkitan, rugi-rugi daya, besar tegangan bus, dan daya yang dibutuhkan beban menjadi parameter yang sangat dibutuhkan untuk mencari berapa besar daya yang harus dibangkitkan oleh pembangkit. Dalam studi tenaga listrik, parameter-parameter diatas dapat dicari dengan studi analisis aliran daya. Tujuan utama dari analisis aliran daya adalah untuk mencari besar dan sudut tegangan pada setiap bus. Dengan diketahuinya besar dan sudut fasa tegangan bus, maka besarnya daya aktif (P) dan daya reaktif (Q) dapat diketahui. Jika daya aktif dan daya reaktif pada setiap bus dalam suatu plant sudah diketahui maka aliran daya dapat diketahui termasuk rugi-rugi daya pada saluran.

Terdapat beberapa kunci yang menjadi acuan optimasi, yaitu efisien dan ekonomis. Efisien yang dimaksud adalah jumlah energi listrik yang dibangkitkan sebisa mungkin mendekati energi listrik permintaan. Dalam hal ini yang dimaksud adalah menekan rugi-rugi pada jaringan dan biaya pembangkitan energi listrik. Jumlah permintaan energi listrik yang terus bertambah dari waktu ke waktu yang tidak sebanding dengan pertumbuhan kapasitas produksi energi listrik membuat keadaan yang tidak seimbang antara kebutuhan konsumen dan jumlah energi yang tersedia. Oleh karena itu dibutuhkan sebuah metode untuk membuat penyaluran energi listrik yang efisien dan biaya pembangkitan seminimal mungkin.

Jarak bus generator dengan bus beban sangat berpengaruh pada *losses* saat listrik dikirimkan menuju konsumen. Dengan analisis aliran daya dapat diketahui kemungkinan-kemungkinan *losses* pada masing-masing saluran. Efisiensi biaya pembangkitan juga harus dilakukan dengan *economic dispatch*, yaitu mencari daya yang dibangkitkan masing-masing generator untuk menyuplai beban dengan biaya seminimal mungkin. Dengan software MATLAB, dapat dilakukan simulasi untuk mengurangi nilai *losses* dengan aliran daya dan biaya pembangkitan yang ekonomis dengan menggunakan *Gravitational Search Algorithm*.

Gravitational Search Algorithm (GSA)/Algoritma Pencarian Gravitasi merupakan salah satu algoritma berbasis populasi yang

merupakan pengembangan dari beberapa algoritma pendahulunya yang sama-sama berbasis populasi. Contohnya adalah *Particle Swarm Optimization* (PSO), yang nantinya akan dijadikan sebagai pembanding dalam Tugas Akhir ini. *Gravitational Search Algorithm* memanfaatkan teori hukum Newton Gravitasi sebagai konsepnya. Dengan adanya semesta yang dibentuk sebagai batasan, kemudian beberapa partikel (agen) yang masing-masing memiliki massa sendiri. Agen-agen yang dibentuk oleh inisiasi program secara acak ini memiliki posisi masing-masing, yang merupakan posisi yang acak pula. Dengan adanya massa pada agen, maka akan timbul gaya tarik-menarik antar agen, yaitu gravitasi. Dengan massanya sendiri-sendiri, maka setiap agen akan memiliki gravitasi aktif dan pasif, inilah yang nantinya dapat merujuk pada solusi yang dicapai.

Particle Swarm Optimization (PSO) sebenarnya metode yang sudah cukup lama dan terkenal cukup cepat dalam mencapai hasil terbaik. *Particle Swarm Optimization* merupakan algoritma berbasis populasi yang sifatnya berdasarkan kumpulan dari hewan-hewan yang ada di alam, seperti burung dan ikan. Sifat dan perilaku ini kemudian disebut dengan *swarm intelligence*.

Dalam penelitian ini akan menunjukkan perbandingan hasil penyelesaian *Multi Objective Optimal Power Flow* dengan menggunakan metode *Gravitational Search Algorithm* (GSA) dengan metode *Particle Swarm Optimization* (PSO).

1. 2. Rumusan Permasalahan

Permasalahan yang akan dibahas dalam Tugas Akhir ini adalah :

- Bagaimana pengaturan daya pembangkit yang optimal untuk meminimalisasi rugi-rugi daya pada sistem dan mendapatkan biaya operasi pembangkitan.
- Bagaimana perbandingan parameter nilai terbaik dari metode *Gravitational Search Algorithm* dengan *Particle Swarm Optimization*.

1. 3. Batasan Permasalahan

Dalam penyelesaian masalah Tugas Akhir ini terdapat beberapa batasan – batasan dan asumsi sebagai berikut :

- Simulasi dilakukan dengan software MATLAB dengan *plant IEEE 30-Bus*. Algoritma yang digunakan adalah *Gravitational Search Algorithm* dan *Particle Swarm Optimization*.

- Fungsi objektif yang diatur secara optimal meliputi :
 - Fungsi biaya pembangkitan (*Economic Dispatch*)
 - Fungsi rugi – rugi jaringan (*Losses*)
- Penyelesaian *Multi Objective Optimal Power Flow* menggunakan dua program berbeda, yaitu :
 - *Gravitational Search Algorithm* (GSA)
 - *Particle Swarm Optimization* (PSO)

1. 4. Tujuan Permasalahan

Tujuan yang ingin dicapai dalam Tugas Akhir ini adalah :

- Untuk meminimalisir rugi – rugi jaringan (*losses*) dan biaya operasi pembangkitan (*economic dispatch*).
- Mencari nilai parameter terbaik dari metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization* dalam simulasi *Optimal Power Flow*, baik dengan *single objective* maupun *multi objective*.

1. 5. Sistematika Penulisan

Penulisan Tugas Akhir akan dibagi menjadi lima bab dengan beberapa subbab. Berikut sistematika penulisannya :

Bab 1 : Pendahuluan

Dalam bab ini dibahas mengenai latar belakang, batasan masalah, tujuan, dan sistematika penulisan.

Bab 2 : Tinjauan Pustaka

Dalam bab ini dibahas tentang teori penunjang mengenai analisis aliran daya, fungsi *Optimal Power Flow*, *Gravitational Search Algorithm* dan *Particle Swarm Optimization*.

Bab 3 : Metodologi Penelitian

Dalam bab ini dijelaskan urutan atau metodologi penyelesaian Tugas Akhir meliputi konsep *Gravitational Search Algorithm* dan *Particle Swarm Optimization* pada OPF.

Bab 4 : Simulasi dan Analisis

Berisi tentang proses dan hasil simulasi yang telah dilakukan. Dalam bab ini dibandingkan hasil dari penyelesaian permasalahan OPF dengan algoritma yang

dibahas dalam Tugas Akhir, yakni *Gravitational Search Algorithm* dan *Particle Swarm Optimization*.

Bab 5 : Kesimpulan dan Saran

Bab terakhir yang berisi tentang penarikan kesimpulan terhadap hasil simulasi dan perbandingan akhir dari kedua algoritma.

BAB 2

SISTEM TENAGA LISTRIK DAN STUDI ALIRAN DAYA

2. 1. Sistem Tenaga Listrik

Sistem tenaga listrik adalah sebuah sistem yang terdiri dari komponen – komponen penyusun tenaga listrik. Secara umum komponen – komponen dalam sistem tenaga listrik dibagi menjadi tiga bagian, yakni Pembangkit tenaga listrik, jaringan transmisi – distribusi dan beban.

2. 1. 1. Pembangkit Tenaga Listrik

Pembangkit tenaga listrik adalah tempat dimana tenaga listrik dibangkitkan yang terdiri dari beberapa komponen utama, diantaranya adalah generator dan *prime mover*. Generator sendiri merupakan mesin listrik yang mengkonversi energi kinetik menjadi energi listrik. Energi kinetik yang didapatkan oleh generator berasal dari penggerak mula (*prime mover*). Energi kinetik penggerak mula ini bisa didapatkan dengan berbagai macam sumber. Ada yang menggunakan energi potensial dari alam, seperti energi potensial air dan angin. Ada yang menggunakan energi thermal seperti batubara, gas alam, bahan bakar minyak hingga nuklir. Ada juga yang menggunakan energi matahari, yakni *Photovoltaic* (PV).

a. Pembangkit Listrik Tenaga Air (PLTA)

Merupakan pembangkit listrik yang menggunakan energi potensial air sebagai sumber energi penggerak turbin. PLTA biasanya memiliki bendungan untuk menampung air dalam jumlah besar dengan ketinggian tertentu. Dari ketinggian ini kemudian dibuat jalur dengan sudut kemiringan tertentu sehingga didapatkan kecepatan air yang sesuai untuk mendapatkan tenaga yang diinginkan. PLTA dengan bendungan juga memiliki pengatur debit air agar bisa mendapatkan debit air yang sesuai dengan kebutuhan beban. Air yang memasuki jalur turbin juga harus steril dari mikroorganismenya. Oleh karena itu ada penampungan khusus untuk dilakukan sterilisasi air agar air tidak mengandung mikroorganismenya. Hal ini bertujuan untuk menjaga jalur air yang dilewati air tetap steril dan tidak kotor, juga menjaga turbin dari kerusakan yang ditimbulkan oleh adanya mikroorganismenya itu sendiri, seperti adanya lumut dan karat. Contoh PLTA di pulau Jawa adalah PLTA Mrican di Mrican, Jawa Barat dan PLTA Sutami di Malang, Jawa Timur.

b. Pembangkit Listrik Tenaga Angin

Pembangkit dengan tenaga angin memanfaatkan kekuatan hembusan angin untuk memutar turbin yang berbentuk baling – baling. Posisi turbin dan generator harus memiliki ketinggian tertentu agar mendapatkan kecepatan angin yang sesuai. Kecepatan angin pun harus memenuhi nilai tertentu, yakni minimal 3 m/s dan maksimal 20 m/s.

c. Pembangkit Listrik Tenaga Uap

Pembangkit dengan tenaga uap ada beberapa jenis, namun yang ada di Indonesia untuk saat ini menggunakan bahan bakar batubara dan panas bumi. Prinsip kerja kedua pembangkit ini sama, yakni energi panas dari bahan baku yang dibakar atau panas dari perut bumi disalurkan ke dalam *boiler* (ketel uap). Dalam *boiler* ini terdapat pipa – pipa yang berisi air, yang nantinya dipanaskan dalam *boiler*. Semakin lama air dalam pipa tersebut mengalami penguapan dan kenaikan suhu yang signifikan. Akibatnya uap air menjadi bertekanan tinggi dan memutar turbin gas. Putaran turbin yang satu *shaft* dengan generator kemudian menghasilkan energi listrik. Uap air yang memiliki suhu tinggi tadi kemudian didinginkan kembali melalui kondensor. Pendingin kondensor sendiri biasanya memanfaatkan air laut. Oleh karena itu biasanya pembangkit listrik dengan batubara memiliki lokasi di pinggir laut. Selain pendinginan yang menggunakan air laut, pengiriman dan penyimpanan lebih mudah dilakukan dengan transportasi laut (menggunakan kapal tongkang). Pembangkit tenaga batubara contohnya adalah PLTU Paiton dan Suralaya

d. Pembangkit Listrik Tenaga Gas (PLTG/PLTGU)

Pembangkit listrik tenaga gas memiliki prinsip yang sedikit berbeda dengan pembangkit termal batubara dan panas bumi. Pada pembangkit ini, gas yang disalurkan melalui pipa ditampung ke dalam tabung kompresor terlebih dahulu sehingga memiliki tekanan yang sangat tinggi. Setelah itu gas bertekanan tinggi ini dicampurkan dengan intake gas lainnya, yakni udara sekitar yang juga telah dimampatkan terlebih dahulu. Setelah campuran sesuai, kemudian gas campuran ini masuk ke dalam ruang bakar (*combustion chamber*) dan menghasilkan tekanan yang sangat tinggi. Energi kalor dari pembakaran inilah yang kemudian digunakan untuk memutar turbin gas dan memutar generator untuk menghasilkan tenaga listrik. Untuk PLTGU, setelah gas tersebut

digunakan untuk memutar turbin, maka hawa panas tidak langsung dibuang, melainkan dimanfaatkan kembali untuk memanaskan air dalam boiler. Proses pemanasan kembali ini disebut dengan HRSG (*Heat Recovery Generator System*) yang kemudian air menjadi uap bertekanan dan digunakan untuk memutar turbin kembali. Pada beberapa industri di Indonesia gas panas dari PLTG digunakan kembali untuk keperluan industri, seperti memanaskan air dan proses kimiawi. Contoh PLTGU di Indonesia adalah PLTGU Gresik.

e. Pembangkit Listrik Tenaga Matahari (*Photovoltaic / PV*)

Pembangkit ini memanfaatkan Sel Surya (*solar cell*) / *photovoltaic* (PV) untuk menghasilkan energi listrik. Untuk saat ini sel surya adalah alat yang mampu mengkonversi langsung energi panas dari matahari menjadi energi listrik. Dari beberapa jenis pembangkit yang disebutkan sebelumnya, pembangkit ini sangat mendukung konsep “*Renewable energy*” selain PLT Air dan angin. Harga investasi yang dibutuhkan untuk PV juga cukup mahal. Dalam sel surya terdapat lapisan semikonduktor yang memiliki polaritas, yakni positif dan negatif. Semikonduktor ini memiliki hubungan struktur yang disebut dengan *junction* p-n. pada saat terkena sinar matahari, terjadi kontak antara semikonduktor tipe n dan p, sehingga terjadi medan listrik. Saat terjadi kelebihan muatan elektron akibat kontak tersebut, elektron yang berlebih kemudian disalurkan melalui material kontak yang merupakan terminal kutub dari PV.

2. 1. 2. Transmisi dan Distribusi

Untuk menyalurkan tenaga listrik dari produsen (pembangkit) kepada konsumen (beban) maka dibutuhkan saluran transmisi dan distribusi energi listrik. Posisi kawat saluran ada yang di atas tanah (*overhead*) dan dibawah tanah (*underground*). Untuk Indonesia jalur transmisi – distribusi yang digunakan hampir semuanya menggunakan jalur di atas tanah (*overhead*). Pada saluran transmisi tegangan dari generator biasanya dinaikkan. Hal ini bertujuan meminimalisir rugi-rugi daya pada saluran. Pada prinsipnya daya yang hilang di saluran dihitung dengan rumus $P = I^2R$. Dengan dinaikannya tegangan, maka I menjadi lebih kecil (dari rumus $S = \sqrt{3} \cdot V_{LL} \cdot I$) dan diharapkan daya rugi – rugi saluran juga menjadi lebih kecil.

Saluran transmisi dan distribusi ini dibagi – bagi berdasarkan tegangannya. Karena tegangan yang ditransmisikan atau didistribusikan

berbeda, maka spesifikasi kawat atau kabel yang digunakan juga berbeda. Berdasarkan tegangannya saluran transmisi – distribusi di Indonesia dibagi menjadi :

a. Saluran Tegangan Ekstra Tinggi (200 – 500 KV)

Disebut juga saluran udara tegangan ekstra tinggi (SUTET). Saluran transmisi ini memiliki tegangan operasi sebesar 200 KV hingga 750 KV. Untuk saat ini saluran ini merupakan saluran dengan tegangan paling tinggi di Indonesia dengan tegangan 500 KV. Pembangkitan tegangan yang sangat tinggi ini efektif digunakan untuk menyalurkan daya listrik yang jauh dari pembangkit dengan jarak diatas 100 km. Kapasitas dayanya juga sangat tinggi, yakni diatas 500 MW. Saat ini saluran 500 KV terdapat pada pulau jawa, yang membentang dari PLTU Paiton hingga Suralaya. Dari saluran ekstra tinggi ini sepanjang saluran terdapat beban maupun suplai yang telah terinterkoneksi antar pembangkit satu dengan lainnya.

b. Saluran Tegangan Tinggi (70 – 150 KV)

Disebut juga saluran udara tegangan tinggi (SUTT). Saluran transmisi ini memiliki tegangan antara 70 KV hingga 150 KV. Namun untuk saat ini tegangan operasi 70 KV mulai banyak ditinggalkan dan lebih sering digunakan tegangan operasi 150 KV. Konfigurasi jaringan pada umumnya memiliki konfigurasi *single* atau *double circuit*. Dalam *single circuit* terdapat 3 fasa saluran dengan 3 atau 4 kawat (termasuk netral). Untuk 3 kawat saluran netralnya biasanya menggunakan media tanah untuk saluran kembali. Apabila kapasitas daya yang disalurkan cukup besar bisa menggunakan lebih dari satu kawat, bisa dua atau empat kawat yang disebut dengan *bundle conductor*. Saluran ini efektif digunakan dengan jarak dibawah 100 km. Apabila lebih jauh nilai drop tegangan menjadi semakin besar.

c. Saluran Tegangan Menengah (6 – 30 KV)

Disebut juga saluran udara tegangan menengah (SUTM), dengan tegangan operasi antara 6 KV hingga 30 KV. Di Indonesia tegangan 6 KV dan 30 KV mulai banyak ditinggalkan dan lebih condong menggunakan tegangan operasi 20 KV. Saluran tegangan menengah biasa digunakan pada hubungan antara Gardu Induk, penyulang (*feeder*), gardu distribusi, hingga konsumen industri atau pelanggan dengan kapasitas besar. Relay pengaman juga mulai banyak diaplikasikan pada tegangan operasi ini.

Untuk efektifitas penyaluran, pengaruh pentanahan netral dan efektifitas pengaman, maka untuk saluran menengah ini memiliki jarak efektif antara 15 hingga 20 km. Lebih jauh dari itu, dikhawatirkan efektifitas pengaman menurun.

d. Saluran Tegangan Rendah (40 – 1000 V)

Saluran ini disebut juga dengan saluran distribusi, yakni hilir dari saluran transmisi itu sendiri. Dengan kata lain, energi listrik yang dihasilkan oleh generator digunakan untuk memasok pelanggan kebutuhan konsumen rumah tangga. Di Indonesia saluran ini memiliki tegangan operasi 220/380V yang berarti tegangan bernilai 220 Volt *line to neutral* dan 380 Volt *line to line*. Rentang tegangan yang diijinkan adalah +5% dan -10% dari tegangan operasi. Untuk saat ini penghantar yang digunakan sebagai saluran adalah *Low Voltage Twisted Cable* (LVTC)

2. 1. 3. Beban

Beban dari sistem tenaga merupakan hilir dari aliran daya listrik. Bisa dikatakan adanya beban adalah alasan mengapa dibutuhkannya pembangkitan tenaga listrik itu sendiri. Bahkan dalam kehidupan sehari-hari manusia pun hampir semuanya memiliki ketergantungan dengan adanya tenaga listrik. Peralatan rumah tangga, alat – alat perkantoran, hingga proses produksi dalam industri semuanya sangat bergantung pada suplai daya listrik dari pembangkit. Dalam beberapa tahun terakhir pertumbuhan permintaan tenaga listrik sangat besar. Hal ini disebabkan oleh semakin majunya perkembangan teknologi manusia yang menggunakan sumber tenaga berupa daya listrik.

Pada umumnya beban tenaga listrik yang paling besar adalah dari sektor industri, yakni yang menggunakan peralatan industri yang menyerap daya cukup besar, seperti motor induksi, motor sinkron, las, peleburan, hingga proses – proses kimiawi. Oleh karena itu konsumen yang berupa industri biasanya mendapatkan suplai langsung dari *feeder* dengan tegangan operasi antara 20 kV atau 150 kV.

2. 2. Analisis Aliran Daya

Analisis aliran daya merupakan ilmu dasar dalam studi sistem tenaga, dimana analisa harus dilakukan pada suatu sistem tenaga untuk mencari tahu tegangan, arus dan daya yang ada. Hal ini perlu dilakukan

untuk kepentingan simulasi dan penelitian. Dalam analisa aliran daya pada studi sistem tenaga, informasi penting yang dapat diperoleh antara lain daya aktif dan daya reaktif dalam saluran, sudut dan magnitude tegangan pada tiap bus dalam *plant*, juga impedansi saluran. Dengan adanya informasi penting tersebut dapat dilakukan analisa kondisi sistem. Permasalahan yang timbul pada analisis aliran daya adalah pada saat beban fasa satu dengan lainnya tidak seimbang, sehingga perhitungan dan analisa yang dilakukan menjadi lebih sulit dan kompleks. Oleh karena itu untuk mempermudah penyelesaian masalah aliran daya terdapat beberapa asumsi, yakni sistem diasumsikan dalam keadaan stabil, beban seimbang pada setiap fasa dan tidak terjadi gangguan dalam sistem. Pada analisis aliran daya terdapat informasi penting yang dibutuhkan yaitu tegangan dan sudut fasa pada tiap bus.

Sudut fasa dan *magnitude* tegangan pada bus – bus dalam sistem inilah yang nantinya menjadi dasar dalam analisa kondisi sistem sehingga didapatkan kondisi yang paling ideal sesuai dengan fungsi objektif yang diinginkan. Ada tiga macam bus dalam analisa aliran daya, yaitu :

1. Bus Referensi (*Slack bus / Swing bus*)

Slack bus atau *swing bus* merupakan bus yang menyuplai kekurangan daya yang dibangkitkan oleh bus – bus lain. Tegangan pada bus ini dijaga dan dipertahankan pada nilai *magnitude* 1 pu dengan sudut 0° . Yang berubah – ubah pada bus ini adalah daya aktif dan reaktifnya, sesuai dengan *constraint*-nya.

2. Bus Generator (*Voltage controlled bus*)

Bus ini merupakan penyuplai daya aktif dan reaktif yang telah ditetapkan. Daya aktif (P) dan daya reaktif (Q) pada bus ini cenderung statis dan dipertahankan, namun nilai tegangan berubah – ubah sesuai dengan *constraint* yang ditetapkan.

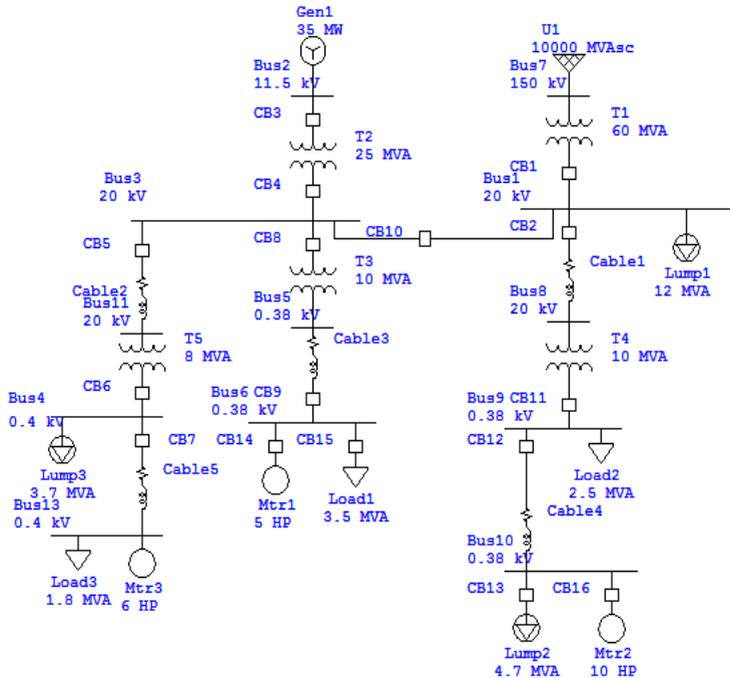
3. Bus beban (*Load bus*)

Bus beban memiliki nilai P dan Q yang tetap, sesuai dengan beban yang aktif dalam sistem. Dalam studi aliran daya biasanya diambil nilai beban puncak untuk menjaga stabilitas sistem.

Adapun untuk studi analisa aliran daya, data – data yang dibutuhkan dari bus – bus dan saluran transmisi, yaitu :

1. *Magnitude* tegangan (dalam bentuk p.u.)
2. Sudut tegangan (θ)
3. Daya pembebanan meliputi daya aktif (P) dan daya reaktif (Q)

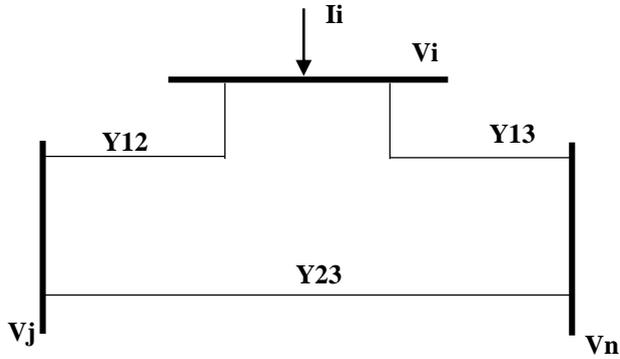
4. Daya pembangkitan meliputi daya aktif (P) dan daya reaktif (Q)
5. *Constraint* pembangkitan, meliputi daya aktif (Pmin dan Pmax) dan daya reaktif (Qmin dan Qmax)



Gambar 2.1. *Komponen dasar dalam sistem tenaga listrik*

2. 3. Permodelan Matematika Aliran Daya

Penyelesaian permasalahan aliran daya dapat dilakukan dengan membuat permodelan matematika dari suatu aliran daya. Hubungan antara bus – bus dalam sistem direpresentasikan dalam bentuk persamaan – persamaan matematis, sehingga perhitungan daya dan nilai – nilai dalam sistem menjadi lebih mudah diselesaikan. Contoh dari sistem tenaga sederhana ditunjukkan pada gambar 2.2.



Gambar 2.2. Sistem tenaga listrik sederhana dengan 3 bus

Dalam sistem sederhana diatas satuan – satuan yang ada dirubah ke dalam bentuk satuan per unit (pu), baik arus, tegangan, impedansi dan admitansi. Untuk mengubah satuan tegangan, arus, dan impedansi dalam bentuk pu, maka dibutuhkan nilai dasar dari masing – masing tegangan, arus, dan impedansi sebagai pembagi, yang disebut sebagai satuan dasar. Untuk arus diperlukan Ibase sebagai pembagi, dan untuk impedansi diperlukan Zbase untuk mengubah satuan I dan Z ke dalam bentuk pu. Untuk mencari nilai I dalam pu maka diperlukan nilai Ibase dari persamaan (2.1).

$$I_{Base} = \frac{KVA_{Base}}{KV_{Base}} \quad (2.1)$$

Kemudian untuk mencari Zbase dapat dicari dengan persamaan (2.2).

$$Z_{Base} = \frac{(KV_{Base})^2}{KVA_{Base}} \quad (2.2)$$

Setelah diketahui nilai dasar dari satuan – satuan tersebut maka nilai I dan Z dapat dirubah ke dalam bentuk pu dengan membagi I dengan Ibase dan Z dengan Zbase. Kemudian hubungan antar bus dinyatakan dalam admitansi Y. Nilai dari Y dapat dicari dari nilai Z dengan persamaan (2.3).

$$Y = \frac{1}{Z} \quad (2.3)$$

Setelah didapatkan nilai admitansi bus (Y) maka admitansi jaringan dalam sistem sederhana pada gambar 2.2 dapat digambarkan dalam matriks admitansi.

$$\begin{bmatrix} I_1 \\ I_2 \\ I_3 \end{bmatrix} = \begin{bmatrix} Y_{11} & Y_{12} & Y_{13} \\ Y_{21} & Y_{22} & Y_{23} \\ Y_{31} & Y_{32} & Y_{33} \end{bmatrix} \begin{bmatrix} V_1 \\ V_2 \\ V_3 \end{bmatrix} \quad (2.4)$$

Kemudian setelah matriks admitansi bus diperoleh maka bisa didapatkan hubungan arus dan tegangan pada persamaan (2.5). Dalam persamaan dimisalkan pencarian dilakukan pada I1.

$$I_1 = (Y_{11} + Y_{12} + Y_{13})V_1 - Y_{11}V_1 - Y_{12}V_2 - Y_{13}V_3 \quad (2.5)$$

Persamaan (2.5) juga berlaku pada pencarian arus pada bus 2 dan bus 3.

2. 3. 1. Daya Kompleks dalam Sistem Tenaga Listrik

Daya dalam sistem tenaga listrik adalah hasil dari perkalian antara arus dan tegangan. Adapun jenis daya yang ada dalam sistem tenaga listrik arus bolak balik (AC) ada 3 yakni daya aktif, daya reaktif dan daya kompleks atau daya total. Dalam sistem tenaga listrik arus bolak balik (AC), perhitungan daya tidak hanya melibatkan nilai magnitudo dari arus dan tegangan, namun juga derajat fasa pada arus maupun tegangan dalam perhitungan. Representasi arus dan tegangan dengan fasa dalam permodelan matematis ditunjukkan pada persamaan (2.6) dan (2.7).

$$V = |V| \angle \theta_v \quad (2.6)$$

$$I = |I| \angle \theta_i \quad (2.7)$$

Kemudian nilai daya kompleks S didapatkan dari perkalian antara V dan I dalam bentuk polar.

$$S = V I^* \quad (2.8)$$

$$S = (|V| \angle \theta_v)(|I| \angle \theta_i)^* \quad (2.9)$$

Pada persamaan (2.8) dan (2.9) terdapat tanda (*) yang berarti conjugate, yakni sudut fasa tersebut menjadi bernilai kebalikan dari sudut fasa yang sebenarnya apabila dimasukkan dalam perhitungan. Pencarian nilai magnitude dari daya kompleks juga dapat dilakukan dengan menggunakan persamaan (2.10).

$$S = P + jQ \quad (2.10)$$

Dimana persamaan tersebut berasal dari rumus daya kompleks dari penjumlahan antara kuadrat daya aktif dan kuadrat daya reaktif pada persamaan (2.11)

$$S = \sqrt{P^2 + jQ^2} \quad (2.11)$$

Daya aktif P dan daya reaktif Q diperoleh dari persamaan (2.12) dan (2.13) yang juga berasal dari perkalian antara arus dan tegangan yang memiliki sudut fasa masing – masing.

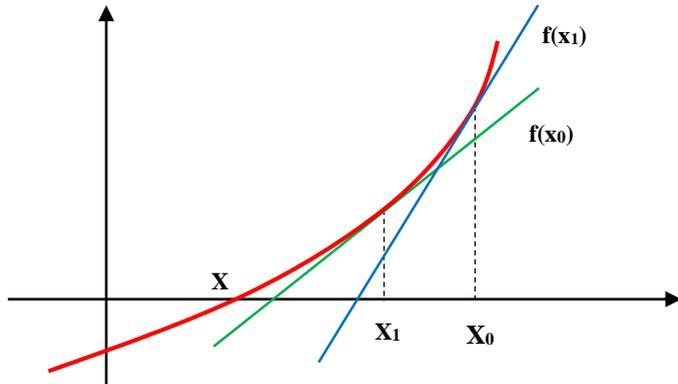
$$P = |V| |I| \cos(\theta) \quad (2.12)$$

$$Q = |V| |I| \sin(\theta) \quad (2.13)$$

2. 3. 2. Pencarian Akar dengan Metode Newton Rhapsion

Metode Newton – Rhapsion adalah metode pendekatan terhadap suatu fungsi $f(x)$ dimana fungsi tersebut dicari akar persamaannya dan memiliki turunan fungsi yang berupa $f'(x)$. Metode ini sering digunakan dalam penyelesaian persamaan non-linier karena termasuk sederhana dan memiliki kecepatan konvergensi yang cukup tinggi. Hal ini dilakukan dengan cara memperkirakan satu titik yang dianggap sebagai titik awal yang dianggap dekat dengan akar sebenarnya. Apabila titik awal pencarian sangat dekat dengan nilai akar yang sebenarnya maka semakin cepat pula konvergensi akar dicapai.

Misalkan terdapat persamaan suatu persamaan $f(x)$ pada gambar 2.3. Fungsi $f'(x)$ akan memotong sumbu x pada saat nilai x tertentu.



Gambar 2.3. Permisalan pencarian akar persamaan dengan metode Newton Rhapsion

Permisalan dari fungsi $f(x)$ direpresentasikan pada persamaan (2.14).

$$f(x) = ax^2 + bx + c \quad (2.14)$$

Maka turunan fungsi $f(x)$ dinyatakan dalam $f'(x)$ sebagai turunan fungsinya pada persamaan (2.16).

$$f'(x) = \frac{\delta f(x)}{\delta x} \quad (2.15)$$

$$f'(x) = 2ax + b \quad (2.16)$$

Sehingga turunan fungsi $f(x)$ menjadi suatu bentuk garis lurus apabila digambarkan dalam grafik. Selanjutnya dengan algoritma metode newton rhapsion, nilai titik awal ditentukan yakni pada x_0 dan nilai tersebut dimasukkan pada fungsi $f(x)$ dan $f'(x)$ sehingga didapatkan nilai $f(x_0)$ dan $f'(x_0)$. Kemudian untuk iterasi selanjutnya, nilai x tidak diperbolehkan ditentukan sendiri, melainkan ditentukan dari persamaan (2.17).

$$x_{n+1} = x_n - \frac{f(x_n)}{f'x_n} \quad (2.17)$$

Untuk menghitung nilai *error* dari hasil iterasi, maka digunakan persamaan (2.18)

$$\varepsilon = \frac{x_n - x_{n-1}}{x_{n-1}} \quad (2.18)$$

Dimana nilai *error* dari titik x pada suatu iterasi dapat diketahui nilainya. Nilai x yang diperoleh dari iterasi dimasukkan kembali dalam persamaan $f(x)$ dan $f'(x)$, begitu seterusnya sehingga parameter penyelesaian tercapai, yakni mendekati kriteria *error* yang minimum atau iterasi maksimum yang diinginkan.

2. 4. Optimal Power Flow

Dari pengertian secara bahasa, *Optimal Power Flow* (OPF) adalah pengaturan terhadap aliran daya sehingga didapatkan nilai yang paling optimal. Dalam studi sistem tenaga OPF merupakan suatu metode untuk melakukan pengaturan yang optimal pada jaringan sistem tenaga listrik dengan melakukan optimasi nilai – nilai fungsi objektif yang dicari, seperti nilai biaya pembangkitan (*economic dispatch*), fungsi rugi – rugi jaringan (*losses*) dengan mempertimbangkan batasan – batasan operasi peralatan (*constraint*).

Dalam sejarahnya OPF memiliki catatan yang cukup panjang. Sebelum banyak penelitian yang sangat pesat dan komputasi yang tinggi seperti sekarang ini, penyelesaian OPF dapat dikatakan membutuhkan waktu yang cukup lama. Seiring berkembangnya ilmu pengetahuan dan teknologi, OPF telah mengalami banyak kemajuan dan dapat dilakukan dengan berbagai algoritma untuk penyelesaiannya.

Fungsi OPF dapat digambarkan dalam bentuk model matematika yang secara umum digambarkan dalam *pseudo code* di bawah.

$$\begin{aligned} & \text{Min } f(x, u) \\ & \text{Subject to : } g(x, u) \\ & h(x, u) \leq 0 \end{aligned} \quad (2.19)$$

$$x = [\theta, V_L, P_{g \text{ slack}}, Q_g]^T \quad (2.20)$$

$$u = [P_g, V_g, t, B_{sh}]^T \quad (2.21)$$

2. 4. 1. *Economic Dispatch*

Economic Dispatch adalah fungsi operasi ekonomis pada pembangkit. Umumnya fungsi ini diterapkan pada pembangkit dengan jenis pembangkit thermal. Dalam proses *economic dispatch* dilakukan pembagian beban terhadap masing – masing pembangkit yang ada dalam sistem sedemikian rupa sehingga didapatkan nilai total biaya pembangkitan yang ada menjadi minimum atau memiliki nilai yang paling ekonomis. Perhitungan dalam fungsi *economic dispatch* membutuhkan input berupa karakteristik pembangkit, dimana karakteristik ini menggambarkan hubungan antara input dan output dari pembangkit. Input yang dimaksud adalah debit bahan bakar pada pembangkit (liter atau ton/jam) dan output pembangkit adalah tenaga listrik yang dihasilkan (MW). Pada setiap pembangkit, input yang dimaksud adalah total energi panas yang dibutuhkan oleh pembangkit (MBtu/jam) atau biaya (\$/jam). Persamaan yang menunjukkan karakteristik input – output pembangkit ditunjukkan pada persamaan (2.22).

$$F(P) = \alpha + \beta P + \gamma P^2 \quad (2.22)$$

Dengan variabel dalam persamaan (2.22) meliputi F yang merupakan input pemakaian bahan bakar/kalor (MBtu/jam), P adalah daya yang dibangkitkan dalam satuan MW dan (a_i , b_i , dan c_i) merupakan konstanta dari pembangkit yang menggambarkan karakteristik input-output pembangkit.

2. 4. 2. *Power Loss Minimization*

Pada *Power Loss Minimization* fungsi objektif yang digunakan yaitu fungsi *losses* (rugi – rugi). Dalam sistem tenaga listrik terdapat daya yang tidak diserap oleh beban, namun daya tersebut hilang dalam jaringan transmisi dalam bentuk energi panas atau berubah menjadi bentuk lain. Hal ini disebabkan oleh adanya impedansi dari jaringan, dimana hal ini akan mengakibatkan adanya penurunan tegangan pada bus yang saling terhubung pada jaringan transmisi. Persamaan (2.23) menunjukkan perhitungan pada fungsi *losses* yang terjadi antar bus.

$$P_{Loss} = \sum_{k=1}^{Nl} g_k [(t_k V_i)^2 + V_j^2 - 2t_k V_i V_j \cos(\theta_i - \theta_j)] \quad (2.23)$$

Pada perhitungan *losses* berlaku keseimbangan aliran daya yang dibangkitkan dengan daya yang diserap oleh beban dan jaringan transmisi yang ditunjukkan pada persamaan (2.24).

$$P_{Load} + P_{Losses} = \sum P_i \quad (2.24)$$

2. 5. Algoritma *Particle Swarm Optimization*

Particle Swarm Optimization merupakan algoritma berbasis populasi yang memiliki performa sangat baik dalam menyelesaikan permasalahan optimasi. Menggunakan sifat algoritma yang berbasis populasi, *Particle Swarm Optimization* melakukan pencarian dengan menggunakan partikel – partikel yang merupakan representasi dari individu. Partikel merupakan sebutan dari individu, sedangkan *swarm* adalah individu – individu yang berkumpul dan membentuk suatu kumpulan. Apabila terdapat *swarm* dengan anggota lebih banyak maka partikel individu di sekitarnya cenderung tertarik dan mendekat pada *swarm* yang memiliki anggota lebih banyak.

Partikel dalam *Particle Swarm Optimization* merepresentasikan penyelesaian atau solusi dari permasalahan. Dalam setiap partikel terdapat nilai kognitif dan nilai sosial. Nilai kognitif adalah kecenderungan partikel untuk bekerja sendiri sebagai partikel mandiri, sedangkan nilai sosial adalah kecenderungan untuk membuat partikel memiliki kecenderungan untuk tertarik dan mengikuti partikel di sekitarnya. Dalam sejarah perkembangan metode AI (Artificial Intellegent), *Particle Swarm Optimization* telah memiliki nama dan sejarah yang sangat besar disebabkan oleh kemampuan dan kapabilitasnya dalam mencapai nilai konvergen yang baik dan relatif cepat. Urutan algoritma dari *Particle Swarm Optimization* adalah sebagai berikut :

- Inisialisasi : Pada *Particle Swarm Optimization*, inisialisasi dilakukan dengan menciptakan *partikel* sejumlah n sesuai dengan parameter yang ditentukan. Setiap partikel juga diberi nilai kognitif dan nilai sosial sesuai parameter. Selain itu inisialisasi juga menciptakan nilai inersia awal setiap partikel, kecepatan awal, juga posisi awalnya dalam dimensi yang ditentukan. Setelah inisialisasi dijalankan, dari parameter yang dimiliki dari setiap individu didapatkan nilai *pbest* dan *gbest* awal, dimana *pbest*

adalah solusi terbaik dari setiap partikel, dan *gbest* adalah solusi terbaik secara keseluruhan.

- Update kecepatan partikel, dimana pada saat inisialisasi sudah terbentuk dan iterasi sudah dilakukan, maka partikel akan mulai bergerak menurut parameter yang telah ditentukan. Kecepatan dari setiap partikel direpresentasikan oleh persamaan (2.25)

$$V_i^{(k+1)} = wV_i^k + C_1rand_1(pb_{best_i} - S_i^k) + C_2rand_2(gbest_i - S_i^k) \quad (2.25)$$

Dengan *V* adalah kecepatan yang dicari. *V* adalah kecepatan saat ini, *C1* adalah nilai sosial partikel dan *C2* adalah nilai kognitif setiap partikel. *Rand* adalah nilai random yang digunakan sebagai pengali dari konstanta. *Pbest* adalah posisi terbaik partikel saat ini dan *gbest* adalah posisi terbaik dari kumpulan partikel atau *swarm*. Setelah kecepatan partikel diketahui maka posisi dari setiap partikel juga diperbaharui dengan persamaan (2.26)

$$S_i^{k+1} = S_i^k + V_i^{k+1} \quad (2.26)$$

- *Update* nilai individual : setelah kecepatan berikut posisi dari setiap *partikel* sudah diperbaharui maka langkah selanjutnya adalah evaluasi terhadap setiap individu berdasarkan posisi terbarunya. Evaluasi ini disimpan dalam variabel *particle best* atau *pbest*.
- Pencarian nilai global : pencarian nilai solusi terbaik berdasarkan keseluruhan nilai terbaik dari masing – masing individu. Semua individu memiliki posisi masing – masing dan nilai solusi diperhitungkan secara keseluruhan.

2. 6. *Gravitational Search Algorithm*

Dalam beberapa tahun terakhir ini banyak sekali dikembangkan metode optimasi. Salah satu cabang yang banyak dikembangkan adalah metode algoritma berbasis populasi. Telah banyak metode yang telah dikembangkan dari algoritma jenis ini, seperti metode *Genetic Algorithm* (GA), *Improved GA*, *Particle Swarm* (PS), *Differential Evolution Algorithm*, *Simulated Annealing*, dan *Evolutionary Programming*.

Metode *Gravitational Search Algorithm* (GSA) merupakan salah satu hasil pengembangan dari algoritma berbasis populasi. *Particle Swarm Optimization* sendiri yang nantinya digunakan sebagai pembanding juga merupakan pengembangan algoritma dengan jenis yang sama. Bedanya adalah *Gravitational Search Algorithm* dikembangkan berdasarkan pola pergerakan partikel dilihat dari massa dan gravitasinya.

Dalam *Gravitational Search Algorithm*, setiap massa memiliki empat hal penting, yaitu :

- Posisi awal
- Massa Inersia (M_i)
- Massa Gravitasi Aktif (M_i^a)
- Massa Gravitasi Pasif (M_i^p)

Pada awal algoritma, posisi suatu sistem dideskripsikan sebagai N (dimensi dari ruang pencarian) massa :

$$X_i = (x_i^1 \dots x_i^d \dots x_i^n) \quad \text{for } i = 1, 2, \dots, N \quad (2.27)$$

Dimana n adalah dimensi *space* dari masalah dan x_i^d adalah posisi dari agen i pada dimensi d_{th} .

Awalnya, agen dari solusi didefinisikan secara acak dan menurut teori gravitasi Newton, gaya gravitasi dari massa j bertindak massa i pada waktu t ditentukan dengan persamaan (2.)

$$F_{ij}^d(t) = G(t) \frac{M_i(t) \times M_j(t)}{R_{ij}(t) + \epsilon} (X_j^d(t) + X_i^d(t)) \quad (2.28)$$

Dimana M_i adalah massa dari objek i , M_j adalah massa dari objek j , $G(t)$ adalah konstanta gravitasi saat waktu t , ϵ adalah konstanta kecil dan $R_{ij}(t)$ adalah jarak antara object i dan j yang diartikan dalam persamaan matematis.

$$R_{ij}(t) = \left\| X_i(t), X_j(t) \right\|_2 \quad (2.29)$$

Total gaya yang dilakukan pada agen i dapat dihitung sebagai berikut :

$$F_i^d(t) = \sum_{j \in kbest, j \neq i}^N rand_j F_{ij}^d(t) \quad (2.30)$$

Dimana $rand_j$ adalah angka acak diantara interval 0 dan 1. $Kbest$ adalah nilai terbaik dari agen K dengan nilai $fitness$ terbaik dan massa paling besar. Untuk mencari percepatan dari agen i_{th} pada waktu t , maka perpindahan agen dilakukan pada dimensi d_{th} . Berdasarkan perpindahan ini, perpindahan menjadi proses pemberian gaya pada agen terikat pada massa agen tersebut. Hal ini berdampak pada percepatan agen yang menerima besarnya gaya. Persamaan yang digunakan ditunjukkan pada persamaan (2.31)

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (2.31)$$

Kemudian pencarian dilakukan kembali pada posisi dan kecepatan selanjutnya. Kecepatan agen yang akan datang didefinisikan sebagai sebuah fungsi dari penjumlahan kecepatan saat ini ditambah dengan percepatan agen. Posisi dan kecepatan agen dapat dihitung secara matematis dengan persamaan (2.32) dan (2.33)

$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (2.32)$$

$$X_i^d(t+1) = X_i^d(t) + v_i^d(t+1) \quad (2.33)$$

Dimana $v_i^d(t)$ dan $x_i^d(t)$ adalah kecepatan dan posisi sebuah agen pada saat waktu t dalam dimensi d . $rand_i$ adalah angka acak pada interval 0-1.

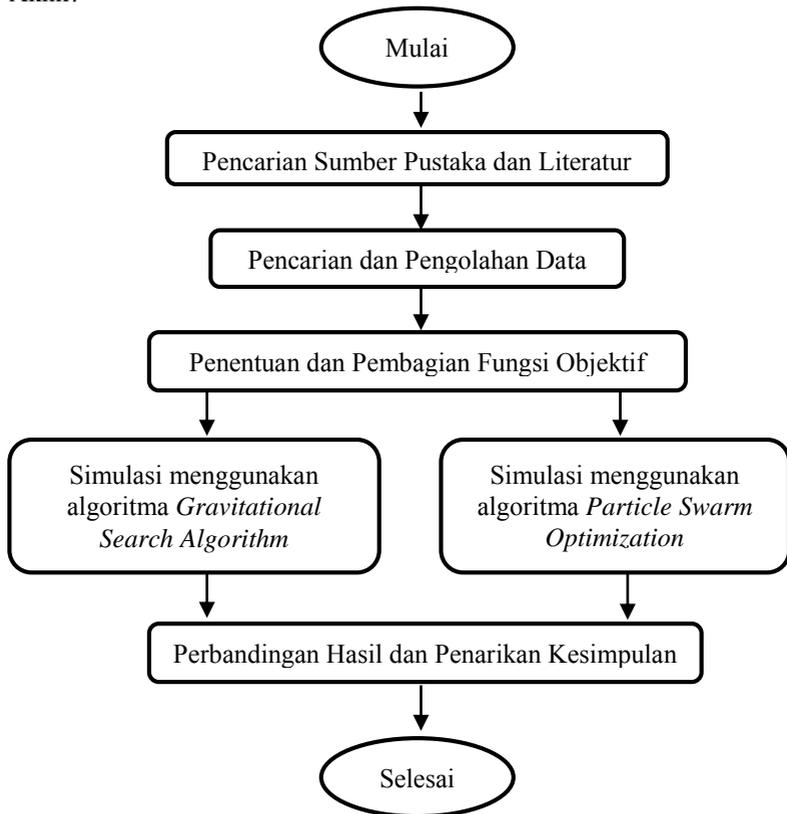
Halaman ini sengaja dikosongkan

BAB 3

PENYELESAIAN MASALAH OPTIMASI DENGAN OPTIMAL POWER FLOW

3. 1. Perencanaan Penelitian

Pada Tugas Akhir ini terdapat tahapan – tahapan yang dilakukan untuk menyelesaikan penelitian OPF menggunakan *Gravitational Search Algorithm*. Gambar 3.1 menunjukkan diagram alir penyelesaian Tugas Akhir.



Gambar 3.1. Diagram alir pengerjaan Tugas Akhir

3. 2. Pencarian Pustaka dan Literatur

Tahap awal Tugas Akhir dimulai dengan pencarian sumber pustaka dan literatur tentang permasalahan inti, yaitu pada *Multi Objective Optimal Power Flow*, *Optimal Power Flow*, *Particle Swarm Optimization*, dan *Gravitational Search Algorithm*. Materi – materi tentang permasalahan didapatkan dari buku dan jurnal nasional maupun internasional yang relevan dengan topik serta judul Tugas Akhir, dimana konten dari literatur akan dipahami untuk mendukung pengerjaan Tugas Akhir. Referensi akan dicantumkan pada daftar pustaka dengan urutan topik sebagai berikut.

1. Literatur tentang *Gravitational Search Algorithm* dicantumkan pada daftar pustaka dengan nomor urut 1 dan 2.
2. Literatur tentang *Optimal Power Flow* dicantumkan pada daftar pustaka dengan nomor urut 3, 4, 5 dan 6.
3. Literatur tentang *Multi Objective Optimal Power Flow* dicantumkan pada daftar pustaka dengan nomor urut 7, 8, 9 dan 10.
4. Literatur tentang *Particle Swarm Optimization* dicantumkan pada daftar pustaka dengan nomor 11, 12, 13, 14, dan 15.

3. 3. Pencarian dan Pengolahan Data

Data yang dibutuhkan dalam penelitian dan simulasi adalah data dari *plant* standar IEEE 30 bus, dimana data dari *plant* akan digunakan sebagai dasar untuk analisa aliran daya. Setelah analisa aliran daya dilakukan, hasil keluaran kemudian digunakan untuk simulasi penentuan biaya pembangkitan optimal dan rugi – rugi pembangkitan yang paling minimum dengan tambahan dari metode yang digunakan.

Data yang dibutuhkan sebagai *input* simulasi :

1. Data bus IEEE 30 bus meliputi jenis bus, tegangan, daya aktif, dan daya reaktif.
2. Data saluran IEEE 30 bus meliputi impedansi saluran antar bus.
3. Data beban
4. Data karakteristik dari setiap pembangkit, meliputi karakteristik input – output daya dan biaya bahan bakar.

3. 4. Penentuan Fungsi Objektif

Fungsi objektif dalam Tugas Akhir ini menggunakan dua fungsi, yaitu fungsi biaya pembangkitan dan fungsi rugi – rugi jaringan, dimana

keduanya memiliki karakteristik dan persamaan masing – masing. Persamaan yang berlaku untuk masing – masing fungsi objektif akan dijelaskan pada sub-subbab berikut.

3. 4. 1. Fungsi *Economic Dispatch* (Biaya Pembangkitan Ekonomis)

Dalam fungsi *economic dispatch*, penekanan fungsi difokuskan pada permasalahan biaya pembangkitan, dimana pengaturan daya output dari generator ditujukan untuk menghasilkan biaya pembangkitan yang paling ekonomis berdasarkan karakteristik dari tiap – tiap pembangkit. Fungsi *economic dispatch* ditunjukkan pada persamaan (3.1) dan (3.2).

$$F = \sum_{i=1}^{NG} F_i(P_i) \quad (3.1)$$

$$F_c(P_g) = \sum_{i=1}^{Ng} a_i + b_i P_{gi} + C_i P_{gi}^2 \quad (3.2)$$

Dimana persamaan (3.1) adalah fungsi biaya total dari semua pembangkit, sedangkan persamaan (3.2) adalah fungsi karakteristik input – output pada setiap pembangkit.

3. 4. 2. Fungsi *Power Loss Minimization* (Rugi – rugi Jaringan)

Dalam fungsi *power loss minimization*, penekanan fungsi objektif ditekankan pada pengurangan rugi – rugi daya pada jaringan antar bus. Selain fungsi pengurangan rugi – rugi daya, terdapat persamaan yang menjadi acuan agar nilai daya pembangkitan sama dengan nilai daya yang diserap oleh beban dan jaringan transmisi. Persamaan (3.3) menunjukkan perhitungan rugi – rugi jaringan, sedangkan persamaan (3.4) menunjukkan keseimbangan daya yang dibangkitkan dengan daya yang diserap oleh beban dan jaring transmisi.

$$P_{Loss} = \sum_{k=1}^{Nl} g_k [(t_k V_i)^2 + V_j^2 - 2t_k V_i V_j \cos(\theta_i - \theta_j)] \quad (3.3)$$

$$P_{Load} + P_{Losses} - \sum P_i = 0 \quad (3.4)$$

Sedangkan persamaan yang menunjukkan jumlah total daya beban yang ada dalam sistem ditunjukkan pada persamaan (3.5)

$$P_L = \sum_{k=1}^{Nl} P_l \quad (3.5)$$

3. 5. Menentukan Batasan – Batasan (*Constraint*)

3. 5. 1. Batasan Kapasitas Daya Pembangkit

Pembangkitan pada setiap generator memiliki beberapa tujuan, diantaranya adalah menjaga peralatan tetap bekerja pada rating dan memperpanjang umur pemakaian dari peralatan. Pada generator terdapat batasan daya minimum dan daya maksimum yang ditunjukkan pada persamaan (3.6).

$$P_{Gi}^{min} \leq P_{Gi} \leq P_{Gi}^{max} \quad (3.6)$$

Selain batasan daya aktif P terdapat juga batasan terhadap daya reaktif Q pada persamaan (3.7).

$$Q_{Gi}^{min} \leq Q_{Gi} \leq Q_{Gi}^{max} \quad (3.7)$$

Sebagai kompensasi terhadap daya reaktif dalam jaringan maka terdapat kapasitor. Batasan untuk kompensasi daya reaktif pada *shunt capacitor* ditunjukkan pada persamaan (3.8).

$$Q_{shunt\ i}^{min} \leq Q_{shunt\ i} \leq Q_{shunt\ i}^{max} \quad (3.8)$$

Dengan pengertian dari masing – masing variabel pada persamaan (3.6), (3.7), dan (3.8) sebagai berikut.

- P_{Gi}^{min} : Pembangkitan daya aktif minimum pada generator ke-i (MW)
- P_{Gi}^{max} : Pembangkitan daya aktif maksimum pada generator ke-i (MW)
- Q_{Gi}^{min} : Pembangkitan daya reaktif minimum pada generator ke-i (MVAR)
- Q_{Gi}^{max} : Pembangkitan daya reaktif maksimum pada generator ke-i (MVAR)
- $Q_{shunt\ i}^{min}$: Batas minimum rating *shunt capacitor* ke-i (MVAR)
- $Q_{shunt\ i}^{max}$: Batas maksimum rating *shunt capacitor* ke-i (MVAR)

3. 5. 2. Keseimbangan Daya

Jumlah nilai daya yang dibangkitkan oleh generator harus sama dengan jumlah daya yang diserap oleh beban. Sehingga apabila daya

serap beban dijumlahkan dengan daya jaringan, maka nilainya harus sama dengan nilai yang dibangkitkan oleh generator. Secara teori, apabila daya yang dibangkitkan berlebihan ataupun kurang dari daya serap beban maka yang terjadi adalah akan terjadi perubahan sudut rotor pada generator. Akibatnya akan terjadi gangguan dan pengaman akan melepas generator yang bermasalah. Jika peristiwa ini terus berlanjut dan kestabilan transien pada setiap generator terganggu, maka generator akan melepaskan diri satu per satu dan terjadi blackout pada sistem. Persamaan untuk keseimbangan daya pembangkitan dengan daya serap beban dan rugi – rugi daya jaringan ditunjukkan pada persamaan (3.9)

$$P_L + P_{loss} = \sum_{i=1}^N P_{Gi} \quad (3.9)$$

Dengan P_L adalah daya total beban yang ada pada sistem (MW), P_{Loss} adalah daya yang diserap oleh jaringan atau *losses* (MW), dan P_{Gi} daya yang dibangkitkan generator (MW).

3. 6. Pembagian Fungsi Objektif dengan Metode *Weighted Sum*

Dalam Tugas Akhir ini terdapat dua fungsi objektif yang dijalankan secara individu dan simultan. Oleh karena itu diperlukan pembagian fungsi objektif apabila terdapat lebih dari satu fungsi objektif. Tujuannya adalah agar fungsi – fungsi objektif yang ada bisa disatukan menjadi satu kesatuan fungsi. Caranya adalah dengan mengalikan fungsi objektif yang ada dengan koefisien pembobot (*weight*). Pembagian nilai koefisien pembobot tergantung pada titik berat pembagian masing – masing fungsi yang diinginkan. Persamaan yang berlaku pada fungsi – fungsi objektif yang dikombinasikan terdapat persamaan (3.10)

$$F = w_1 f_1 + w_2 f_2 \quad (3.10)$$

Dalam persamaan (3.10) fungsi f_1 dan f_2 adalah fungsi objektif yang diinginkan, sedangkan F adalah fungsi objektif gabungan, dimana fungsi objektif pertama dan fungsi objektif kedua dilakukan secara simultan.

$$w_1 + w_2 = 100\% \quad (3.11)$$

Faktor pengali w_1 dan w_2 adalah koefisien pembobot *weight*, dimana nilai dari w_1 dan w_2 merupakan representasi dari berapa persen

fungsi objektif yang diimplementasikan. Jumlah total antara w_1 dan w_2 adalah 100 persen. Adapun jika terdapat hal lain yang mempengaruhi fungsi objektif, maka ada faktor pengali lagi yang disebut dengan *penalty factor*. Dimana apabila terdapat *penalty factor* pada fungsi yang dimaksud, maka persamaan kombinasi dari fungsi objektif menjadi seperti persamaan (3.12).

$$\alpha = \frac{f_{refmax}}{f_1max}$$

$$\beta = \frac{f_{refmax}}{f_2max}$$

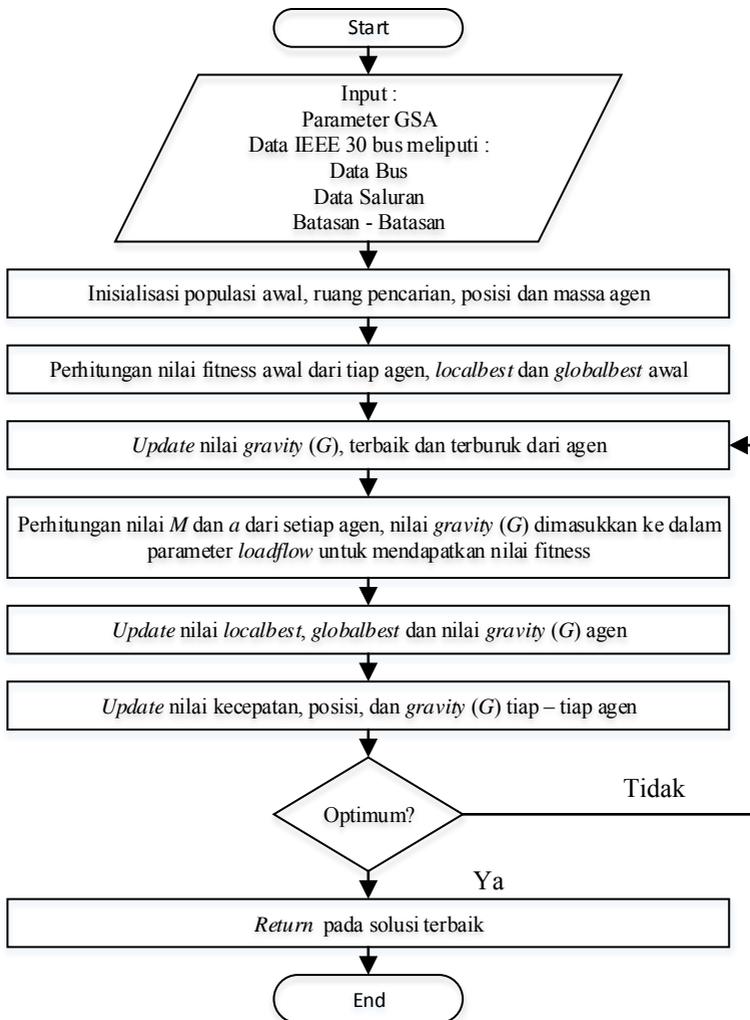
$$F = w_1\alpha f_1 + w_2\beta f_2 \quad (3.12)$$

Dengan α adalah *penalty factor* untuk fungsi objektif pertama, dan β merupakan *penalty factor* untuk fungsi objektif yang kedua.

3. 7. Simulasi *Gravitational Search Algorithm* (GSA)

Simulasi Optimal Power Flow yang dilakukan dalam Tugas Akhir ini menggunakan *Gravitational Search Algorithm* sebagai metode utama. Dimana metode utama juga digunakan untuk menyelesaikan permasalahan dengan dua fungsi objektif, yakni *economic dispatch* dan power loss minimization secara satu persatu maupun secara simultan.

Diagram alir penyelesaian OPF dengan *Gravitational Search Algorithm* dideskripsikan pada gambar 3.2.



Gambar 3.2. Diagram alir algoritma Gravitational Search Algorithm

3. 7. 1. Inisialisasi Ruang Pencarian (*Space Bound*) dan *Agent Mass*

Dalam beberapa tahun terakhir ini banyak sekali dikembangkan metode optimasi. Salah satu cabang yang banyak dikembangkan adalah metode algoritma berbasis populasi. Telah banyak metode yang telah dikembangkan dari algoritma jenis ini, seperti metode *Genetic Algorithm* (GA), *Improved GA*, *Particle Swarm* (PS), *Differential Evolution Algorithm*, *Simulated Annealing*, dan *Evolutionary Programming*. Metode *Gravitational Search Algorithm* (GSA) merupakan salah satu hasil pengembangan dari algoritma berbasis populasi. *Particle Swarm Optimization* sendiri yang nantinya digunakan sebagai pembanding juga merupakan pengembangan algoritma dengan jenis yang sama. Bedanya adalah *Gravitational Search Algorithm* dikembangkan berdasarkan pola pergerakan partikel dilihat dari massa dan gravitasinya. Pada awal algoritma, posisi suatu sistem dideskripsikan sebagai dimensi N (dimensi dari ruang pencarian) massa :

$$X_i = (x_i^1 \dots x_i^d \dots x_i^n) \quad \text{for } i = 1, 2, \dots, N \quad (3.13)$$

Dimana n adalah dimensi *space* dari masalah dan x_i^d adalah posisi dari agen i pada dimensi d_{th} .

3. 7. 2. *Active – Passive Gravitation, dan Agent Movement*

Dalam *Gravitational Search Algorithm*, setiap *agent* memiliki empat hal penting yang bisa didapatkan dari massa dan inisialisasi *agent*, yaitu Posisi awal, Massa Inersia (Mii), Massa Gravitasi Aktif (Mai), Massa Gravitasi Pasif (Mpi). Awalnya, agen dari solusi didefinisikan secara acak dan menurut teori gravitasi Newton, gaya gravitasi dari massa j bertindak pada massa i pada waktu iterasi t ditentukan sebagai berikut :

$$F_{ij}^d(t) = G(t) \frac{M_i(t) \times M_j(t)}{R_{ij}(t) + \epsilon} (X_j^d(t) + X_i^d(t)) \quad (3.14)$$

Dimana M_i adalah massa dari object i , M_j adalah massa dari objek j , $G(t)$ adalah konstanta gravitasi saat waktu t , ϵ adalah konstanta kecil dan $R_{ij}(t)$ adalah jarak di antara object i dan j yang diartikan sebagai berikut :

$$R_{ij}(t) = \left\| X_i(t), X_j(t) \right\|_2 \quad (3.15)$$

Total gaya yang dilakukan pada agen i_{th} dapat dihitung sebagai berikut :

$$F_i^d(t) = \sum_{j \in kbest, j \neq i}^N rand_j F_{ij}^d(t) \quad (3.16)$$

Dimana $rand_j$ adalah angka acak diantara interval 0 dan 1. $Kbest$ adalah nilai terbaik dari agen K dengan nilai *fitness* terbaik dan massa paling besar.

Untuk mencari percepatan dari agen i_{th} pada waktu t , hukum perpindahan dimensi d_{th} digunakan langsung untuk perhitungan. Berdasarkan hukum ini, hal tersebut proporsional untuk memberikan gaya pada agen dan tidak proporsional untuk massa agen tersebut. Persamaan yang digunakan adalah sebagai berikut :

$$a_i^d(t) = \frac{F_i^d(t)}{M_{ii}(t)} \quad (3.17)$$

Kemudian pencarian dilakukan pada posisi dan kecepatan selanjutnya. Kecepatan agen yang akan datang didefinisikan sebagai sebuah fungsi dari penjumlahan kecepatan saat ini ditambah dengan percepatan agen. Posisi dan kecepatan agen dapat dihitung dengan persamaan berikut :

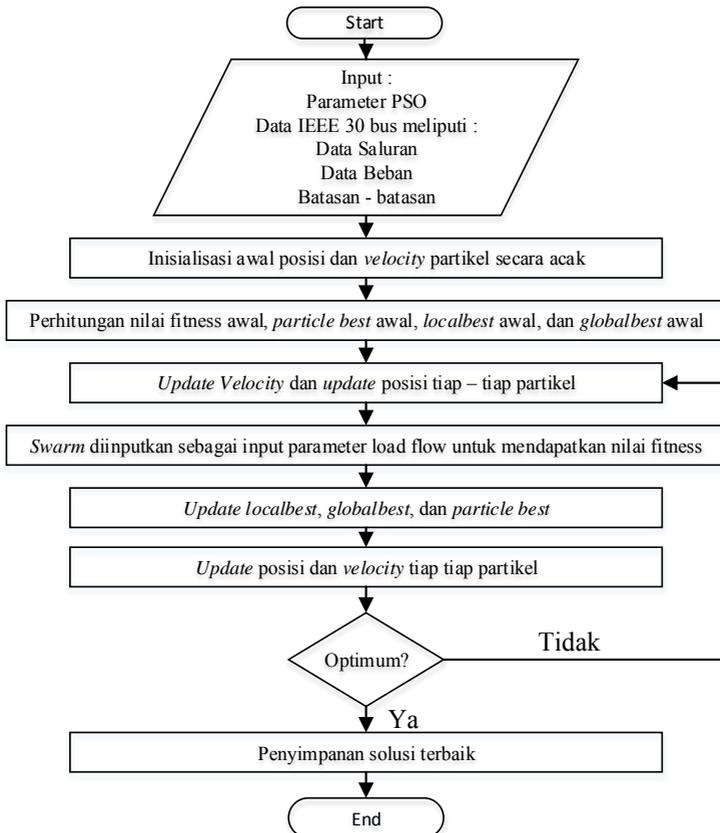
$$v_i^d(t+1) = rand_i \times v_i^d(t) + a_i^d(t) \quad (3.18)$$

$$X_i^d(t+1) = X_i^d(t) + v_i^d(t+1) \quad (3.19)$$

Dimana $v_i^d(t)$ dan $x_i^d(t)$ adalah kecepatan dan posisi sebuah agen pada saat waktu t di dimensi d . $rand_i$ adalah angka random pada interval 0-1. Posisi final agent pada akhir iterasi inilah yang merupakan solusi terbaik pada iterasi. Begitu seterusnya hingga kriteria yang diharapkan tercapai, baik dari iterasi maksimum maupun acuan nilai *error*.

3. 8. Simulasi Particle Swarm Optimization (PSO)

Dalam Tugas akhir ini *Particle Swarm Optimization* digunakan sebagai pembanding dalam menyelesaikan permasalahan OPF yang ada. Metode *Particle Swarm Optimization* memiliki catatan yang sangat baik dalam sejarah penyelesaian optimasi berbasis AI (Artificial Intellegent) karena memiliki sifat iterasi yang cenderung stabil dan nilai konvergensi yang sangat baik. Algoritma *Particle Swarm Optimization* ditunjukkan dalam diagram pada gambar 3.3.



Gambar 3.3. Algoritma Particle Swarm Optimization dalam diagram alir

3. 8. 1. Inisialisasi Individu dan Perhitungan Awal Algoritma

Pada tahap awal algoritma, inisialisasi dilakukan dengan membuat suatu dimensi k (ruang pencarian solusi). Kemudian dalam ruang pencarian k dilakukan pembuatan individu – individu secara acak, yang direpresentasikan sebagai *partikel*. Posisi awal partikel dibuat secara acak dalam dimensi k dengan jumlah partikel yang sesuai dengan parameter yang telah ditentukan. Setiap partikel yang diciptakan memiliki nilai sosial dan kognitif masing – masing sesuai parameter yang telah ditentukan. Setiap partikel juga memiliki nilai inersia masing – masing yang juga ditentukan secara acak, dimana nilai inersia adalah nilai kecenderungan untuk mengubah posisi individu.

Kemudian dari inisialisasi yang telah dilakukan, dilakukan perhitungan solusi dari setiap individu dengan menentukan nilai individu terbaik yang direpresentasikan dalam *pbest* dan nilai terbaik solusi keseluruhan yang direpresentasikan dalam *gbest*, dimana solusi awal akan direpresentasikan nilai *fitness* awal dari nilai *localbest* dan *globalbest*. Sedangkan posisi masing – masing partikel dalam iterasi pembentuk nilai *fitness* ini disimpan nilainya dalam partikel best. Nilai – nilai inilah yang dijadikan sebagai pembandingan dan atau dasar perhitungan pada iterasi selanjutnya.

3. 8. 2. Pergerakan dan Posisi Partikel

Individu – individu yang telah diciptakan dalam dimensi k kemudian mengalami perpindahan akibat dari parameter yang telah dijelaskan sebelumnya, yakni adanya nilai sosial dan nilai kognitif dari masing – masing individu. Apabila ada beberapa partikel lain yang membentuk *swarm* atau kumpulan dari beberapa partikel maka akan ada kecenderungan dari partikel individu untuk mendekat pada *swarm*. Perpindahan posisi dipengaruhi oleh kecepatan dari partikel yang mengalami perpindahan, dimana rumus dari kecepatan partikel ditentukan dengan persamaan (3.20) :

$$V_i^{(k+1)} = wV_i^k + C_1rand_1(pbest_i - S_i^k) + C_2rand_2(gbest_i - S_i^k) \quad (3.20)$$

Kemudian setelah perpindahan terjadi, individu memiliki posisi baru berdasarkan posisi awal dijumlahkan dengan kecepatannya masing – masing pada iterasi. Persamaan yang digunakan untuk memperbaharui

posisi dari partikel untuk mendapatkan posisi baru direpresentasikan pada persamaan (3.21).

$$S_i^{(k+1)} = S_i^{(k+1)} + V_i^{(k+1)} \quad (3.21)$$

Dengan $S_i^{(k+1)}$ adalah nilai posisi partikel dan $V_i^{(k+1)}$ adalah kecepatan partikel yang dihasilkan pada iterasi. Dengan adanya perpindahan posisi, maka akan terjadi perbedaan nilai fitness dari iterasi sebelumnya dan akan dilakukan perbandingan pada langkah berikutnya.

3. 8. 3. Penentuan Nilai Akhir Partikel

Setelah proses perpindahan posisi partikel dilakukan maka selanjutnya adalah melakukan perhitungan kembali nilai solusi dari setiap individu dan *swarm* yang terbentuk akibat adanya perpindahan pada posisi dan kecepatan masing – masing partikel. Hasil yang didapatkan akan dimasukkan dan disimpan dalam variabel *pbest* untuk setiap individu dan *gbest* untuk solusi secara keseluruhan. Nilai dari *pbest* dan *gbest* pada iterasi terbaru kemudian dibandingkan nilainya dengan nilai *pbest* dan *gbest* pada iterasi sebelumnya, lebih baik, lebih buruk, atau sama. Apabila nilai lebih baik maka solusi pada iterasi sebelumnya akan terjadi pergeseran *ranking* pada beberapa nilai solusi yang telah disimpan. Nilai solusi yang paling buruk dari *ranking* dihapus dan nilai solusi pada iterasi saat ini akan disimpan dan dimasukkan pada daftar nilai terbaik dalam *ranking*.

Sedangkan apabila lebih buruk nilai akan dibuang dan dilakukan iterasi baru dengan nilai *pbest* dan *gbest* yang terbaik pada iterasi sebelumnya. Dengan menggunakan persamaan (3.22) nilai *pbest* dirubah pada setiap iterasinya dengan syarat nilainya lebih baik dari nilai konvergensi pada iterasi sebelumnya. Namun apabila nilai *pbest* lebih buruk atau sama dengan iterasi sebelumnya maka digunakan nilai pada iterasi sebelumnya dengan persamaan (3.23) untuk mempertahankan nilai *fitness* yang dihasilkan pada iterasi sebelumnya.

$$Pbest_{ij}^{(k+1)} = X_{ij}^{(k+1)} \text{ jika } TC_i^{(k+1)} < TC_i^k \quad (3.22)$$

$$Pbest_{ij}^{(k+1)} = Pbest_i^k \text{ jika } TC_i^{(k+1)} \geq TC_i^k \quad (3.23)$$

Dengan $X_{ij}^{(k+1)}$ adalah posisi partikel pada iterasi selanjutnya, $TC_i^{(k+1)}$ merepresentasikan fungsi objektif yang dievaluasi pada posisi partikel dan $Pbest_{ij}^{(k+1)}$ adalah posisi terbaik individu pada iterasi selanjutnya. Kemudian iterasi akan dihentikan sesuai dengan kriteria yang diharapkan. Dalam Tugas Akhir ini kriteria yang diharapkan untuk menghentikan iterasi adalah tercapainya iterasi maksimum pada parameter *Particle Swarm Optimization*.

Halaman ini sengaja dikosongkan

BAB 4 SIMULASI DAN ANALISIS

Dalam Tugas Akhir ini dilakukan simulasi *Optimal Power Flow* dengan dua macam permasalahan yaitu fungsi *economic dispatch* dan fungsi *power loss minimization* baik secara terpisah maupun secara simultan. Simulasi yang dilakukan adalah dengan mengatur daya aktif (P) dan daya reaktif (Q) dari pembangkit. Metode penyelesaian yang digunakan adalah *Gravitational Search Algorithm* (GSA) sebagai metode utama, dengan metode pembanding *Particle Swarm Optimization* (PSO).

Simulasi yang dilakukan dalam Tugas Akhir :

1. Simulasi OPF menggunakan satu fungsi objektif
 - a. Simulasi *Economic Dispatch*
 - b. Simulasi *Power Loss Minimization*
2. Simulasi OPF dengan dua fungsi objektif
 Simulasi *Economic Power Loss Minimization*

4. 1. Data IEEE 30-Bus

Dalam Tugas Akhir ini digunakan sistem standar IEEE 30-bus dengan 6 pembangkit. Tabel 4.1 menyajikan data bus pada sistem standar IEEE 30-bus.

Tabel 4.1. Data bus IEEE 30-bus

Bus no	Bus code	Voltage Magn.	Angle Deg	Load		Generation				Static Mvar
				MW	MVAR	MW	MVAR	Qmin	Qmax	
1	1	1.05	0	0	0	260.2	-16.1	0	0	0
2	2	1.043	-5.48	21.7	12.7	40	50	-40	50	0
3	0	1.021	-7.96	2.4	1.2	0	0	0	0	0
4	0	1.012	-9.62	7.6	1.6	0	0	0	0	0
5	2	1.01	-14.37	94.2	19	0	37	-40	40	0
6	0	1.01	-11.34	0	0	0	0	0	0	0
7	0	1.002	-13.12	22.8	10.9	0	0	0	0	0
8	2	1.01	-12.1	30	30	0	37.3	-10	40	0
9	0	1.051	-14.38	0	0	0	0	0	0	0

(Lanjutan **Tabel 4.1.** Data bus IEEE 30-bus)

Bus	Bus	Voltage	Angle	Load		Generation				Static
no	code	Magn.	Deg	MW	MVAR	MW	MVAR	Qmin	Qmax	Mvar
10	0	1	0	5.8	2	0	0	-6	24	0
11	2	1.062	-14.39	0	0	0	16.2	-10	24	-8.5
12	0	1.057	-15.24	11.2	7.5	0	0	0	0	0
13	2	1.05	-15.24	0	0	0	10.6	-10	24	-10
14	0	1.042	-16.13	6.2	1.6	0	0	0	0	0
15	0	1.038	-16.22	8.2	2.5	0	0	0	0	0
16	0	1.045	-15.83	3.5	1.8	0	0	0	0	0
17	0	1.04	-16.14	9	5.8	0	0	0	0	0
18	0	1.028	-16.82	3.2	0.9	0	0	0	0	0
19	0	1.026	-17	9.5	3.4	0	0	0	0	0
20	0	1.03	-16.8	2.2	0.7	0	0	0	0	0
21	0	1.033	-16.42	17.5	11.2	0	0	0	0	0
22	0	1.033	-16.41	0	0	0	0	0	0	0
23	0	1.027	-16.61	3.2	1.6	0	0	0	0	0
24	0	1.021	-16.78	8.7	6.7	0	0	0	0	0.043
25	0	1.017	-16.35	0	0	0	0	0	0	0
26	0	1	-16.77	3.5	2.3	0	0	0	0	0
27	0	1.023	-15.82	0	0	0	0	0	0	0
28	0	1.007	-11.97	0	0	0	0	0	0	0
29	0	1.003	-17.06	2.4	0.9	0	0	0	0	0
30	0	0.992	-17.94	2.6	1.9	0	0	0	0	0

Pada tabel 4.2 disajikan data impedansi saluran pada IEEE 30-bus.

Tabel 4.2. Data saluran IEEE 30-bus

Bus	Bus	R	X	$\frac{1}{2} b$	Nilai 1 tap setting
nl	nr	pu	pu	pu	
1	2	0.0192	0.0575	0.0264	1
1	3	0.0452	0.1852	0.0204	1
2	4	0.057	0.1737	0.0184	1
3	4	0.0132	0.0379	0.0042	1
2	5	0.0472	0.1983	0.0209	1
2	6	0.0581	0.1763	0.0187	1
4	6	0.0119	0.0414	0.0045	1
5	7	0.046	0.116	0.0102	1
6	7	0.0267	0.082	0.0085	1
6	8	0.012	0.042	0.0045	1
6	9	0	0.208	0.97	1
6	10	0	0.556	0.96	1
9	11	0	0.208	0	1
9	10	0	0.11	0	1
4	12	0	0.256	0.93	1
12	13	0	0.14	0	1
12	14	0.1231	0.2559	0	1
12	15	0.0662	0.1304	0	1
12	16	0.0945	0.1987	0	1
14	15	0.221	0.1997	0	1
16	17	0.0824	0.1923	0	1
15	18	0.1073	0.2185	0	1
18	19	0.0693	0.1292	0	1
19	20	0.034	0.068	0	1
10	20	0.0936	0.209	0	1
10	17	0.0324	0.0845	0	1
10	21	0.0348	0.0749	0	1
10	22	0.0727	0.1499	0	1
21	22	0.0116	0.0236	0	1
15	23	0.1	0.202	0	1
22	24	0.115	0.179	0	1
23	24	0.132	0.27	0	1

(Lanjutan **Tabel 4.2.** Data saluran IEEE 30-bus)

Bus	Bus	R	X	$\frac{1}{2} b$	Nilai 1 tap setting
nl	nr	pu	pu	pu	
1	2	0.0192	0.0575	0.0264	1
25	27	0.1093	0.2087	0	1
28	27	0	0.396	0.96	1
27	29	0.2198	0.4153	0	1
27	30	0.3202	0.6027	0	1
29	30	0.2399	0.4533	0	1
8	28	0.0636	0.2	0.0214	1
6	28	0.0169	0.0599	0.065	1

Generator pada *plant* IEEE 30-bus terdiri dari 6 pembangkit yang masing – masing memiliki pembangkitan minimum dan maksimum.

Tabel 4.3. Data generator IEEE 30-bus

No	Pembangkit	Daya Aktif (MW)		Daya Reaktif (MVAR)	
		Minimum	Maksimum	Minimum	Maksimum
1	Pembangkit 1	50	200	-20	200
2	Pembangkit 2	20	80	-20	100
3	Pembangkit 3	15	50	-15	80
4	Pembangkit 4	10	35	-10	60
5	Pembangkit 5	10	30	-10	50
6	Pembangkit 6	12	40	-10	60

Batasan pada *tap setting* trafo ada pada tabel 4.4.

Tabel 4.4. Setting tap transformator IEEE 30-bus

No	Tap Trafo	Tap Setting (pu)	
		Minimum	Maksimum
1	Tap Trafo 1	0.95	1.1
2	Tap Trafo 2	0.95	1.1
3	Tap Trafo 3	0.95	1.1
4	Tap Trafo 4	0.95	1.1

Batasan kemampuan *Capasitor Bank* ada pada tabel 4.5 :

Tabel 4.5. Data Capacitor Bank IEEE 30-bus

No	Capasitor Bank	Minimum (MW)	Maksimum (MW)
1	Capasitor 1 (Bus 10)	2	5
2	Capasitor 2 (Bus 12)	2	5
3	Capasitor 3 (Bus 15)	2	5
4	Capasitor 4 (Bus 17)	2	5
5	Capasitor 5 (Bus 20)	2	5
6	Capasitor 6 (Bus 21)	2	5
7	Capasitor 7 (Bus 23)	2	5
8	Capasitor 8 (Bus 24)	2	5
9	Capasitor 9 (Bus 29)	2	5

Pada setiap pembangkit dalam sistem IEEE 30 bus memiliki karakteristik masing – masing yang ditunjukkan oleh tabel 4.6 :

Tabel 4.6. Data fungsi biaya pembangkit sistem IEEE 30-bus

No	Pembangkit	Fungsi Biaya (\$ / Jam)
1	Pembangkit 1 (Bus 1)	$C_1 = 0.00375P_1^2 + 2P_1 + 0$
2	Pembangkit 2 (Bus 2)	$C_2 = 0.00175P_2^2 + 1.75P_2 + 0$
3	Pembangkit 3 (Bus 5)	$C_3 = 0.0625P_3^2 + 1P_3 + 0$
4	Pembangkit 4 (Bus 8)	$C_4 = 0.00834P_4^2 + 3.25P_4 + 0$
5	Pembangkit 5 (Bus 11)	$C_5 = 0.025P_5^2 + 3P_5 + 0$
6	Pembangkit 6 (Bus 13)	$C_6 = 0.025P_6^2 + 3P_6 + 0$

Dalam melakukan simulasi, komputer yang digunakan memiliki spesifikasi sebagai berikut

Tabel 4.7. Data spesifikasi CPU

No	Parameter	Spesifikasi CPU
1	Processor	Inter Core™ i5-3470
2	Speed Processing	3.6 GHz
3	Installed Memory (RAM)	4GB
4	System Type	Win 8 64-bit OS

4. 2. Simulasi *Economic Dispatch*

Simulasi *Economic Dispatch* adalah pengujian metode OPF untuk meminimalisasi biaya pembangkitan. Metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization* akan digunakan untuk menyelesaikan masalah *Economic Dispatch* dan akan dibandingkan hasil antara keduanya.

Pada simulasi *economic dispatch* digunakan satu fungsi objektif saja (*single objective*), yaitu *economic dispatch* saja. Dalam simulasi ini nilai yang ditekan adalah nilai *economic dispatch* saja, tanpa mempedulikan nilai *losses*.

4. 2. 1. Simulasi *Economic Dispatch* Metode *Gravitational Search Algorithm*

Parameter yang digunakan untuk simulasi OPF dengan metode *Gravitational Search Algorithm* pada sistem IEEE 30 bus ditunjukkan pada tabel 4.8.

Tabel 4.8. Parameter *Gravitational Search Algorithm*

Parameter	Nilai
Jumlah <i>Agent</i>	20
Jumlah Iterasi	150
G0	98
Alfa	25
Mmin	0
Mmax	2

Dalam tabel 4.8 terdapat beberapa parameter diantaranya jumlah *agent* merupakan jumlah penyimpan solusi OPF. Jumlah iterasi merupakan jumlah perhitungan yang dilakukan dalam satu kali pengaplikasian program. Alfa merupakan konstanta untuk menghitung nilai *a* dan G0 adalah konstanta untuk perhitungan *G*. Kedua variabel tersebut digunakan sebagai nilai evaluasi dari posisi *agent*. Nilai dari parameter dalam tabel 4.8 digunakan secara terus menerus pada seluruh simulasi fungsi dengan metode *Gravitational Search Algorithm*.

Pada simulasi *Economic Dispatch* hanya terdapat satu fungsi objektif yang diaktifkan, yakni minimalisasi biaya pembangkitan. Dengan metode *Gravitational Search Algorithm* didapatkan daya dan biaya pembangkitan yang ditunjukkan pada pada tabel 4.9.

Tabel 4.9. Daya dan biaya pembangkitan hasil *economic dispatch* dengan metode *Gravitational Search Algorithm*

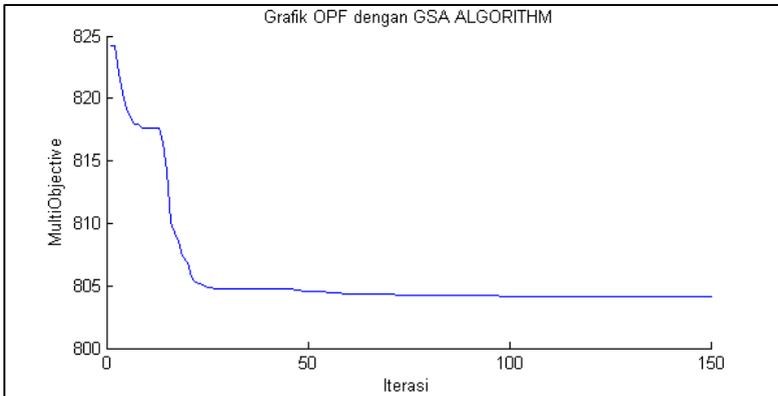
No	Parameter	Nilai (MW)	No	Parameter	Nilai (\$/Jam)
1	Daya Pembangkitan 1	167.63	7	Biaya Pembangkitan 1	440.660
2	Daya Pembangkitan 2	52.641	8	Biaya Pembangkitan 2	140.615
3	Daya Pembangkitan 3	25.804	9	Biaya Pembangkitan 3	67.420
4	Daya Pembangkitan 4	22.086	10	Biaya Pembangkitan 4	75.846
5	Daya Pembangkitan 5	10.231	11	Biaya Pembangkitan 5	33.310
6	Daya Pembangkitan 6	14.279	12	Biaya Pembangkitan 6	47.934

Dan total daya pembangkitan, *losses* dan total biaya pembangkitan ditunjukkan pada tabel 4.10.

Tabel 4.10. Total daya, *losses*, dan biaya pembangkitan metode *Gravitational Search Algorithm*

No	Pembangkit	Daya Pembangkitan (MW)	<i>Losses</i> (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	167.638	9.0062	440.660
2	Daya Pembangkitan 2	52.641		140.615
3	Daya Pembangkitan 3	25.804		67.420
4	Daya Pembangkitan 4	22.086		75.846
5	Daya Pembangkitan 5	10.231		33.310
6	Daya Pembangkitan 6	14.279		47.934
	Total	292.679	9.0062	805.7860

Grafik konvergensi *economic dispatch* dengan metode *Gravitational Search Algorithm* ditunjukkan pada gambar 4.1.



Gambar 4.1. Grafik konvergensi economic dispatch menggunakan metode Gravitational Search Algorithm

Total daya aktif, losses dan total biaya pembangkitan hasil dari simulasi *Economic Dispatch* pada sistem IEEE 30 bus menggunakan metode *Gravitational Search Algorithm* ditunjukkan pada tabel 4.10. Pada simulasi OPF ini digunakan 150 iterasi dan *Gravitational Search Algorithm* mencapai titik konvergensinya pada iterasi ke-126 dengan nilai *fitness* mencapai 804.1233. Total daya aktif yang dibangkitkan mencapai 292.679 MW, total losses 9.0062 MW dan biaya pembangkitan pada 805.7860 \$/jam.

4. 2. 2. Simulasi Economic Dispatch Metode Particle Swarm Optimization

Particle Swarm Optimization memiliki parameter yang berbeda dari *Gravitational Search Algorithm*. Nilai dari parameter yang digunakan dalam simulasi *Particle Swarm Optimization* ditunjukkan pada tabel 4.11.

Tabel 4.11. Parameter Algoritma Particle Swarm Optimization

Parameter	Nilai
Jumlah Iterasi	150
Jumlah Swarm	20
c1 (nilai sosial)	1.3
c2 (nilai kognitif)	2.0

Pada tabel 4.11 terdapat beberapa parameter *Particle Swarm Optimization*, diantaranya jumlah *swarm* yang merupakan representasi dari kumpulan individu penyimpanan solusi. Pada simulasi parameter ini diatur dengan nilai 20. Parameter *c1* dan *c2* merupakan representasi dari nilai sosial dan nilai kognitif. Nilai sosial adalah kecenderungan partikel untuk berkumpul dengan *swarm* sedangkan nilai kognitif adalah kecenderungan partikel untuk berdiri sendiri sebagai solusi (*stand alone*).

Hasil simulasi *economic dispatch* dengan metode *Particle Swarm Optimization* ditunjukkan pada tabel 4.12.

Tabel 4.12. Daya dan biaya pembangkitan hasil simulasi *economic dispatch* dengan metode *Particle Swarm Optimization*

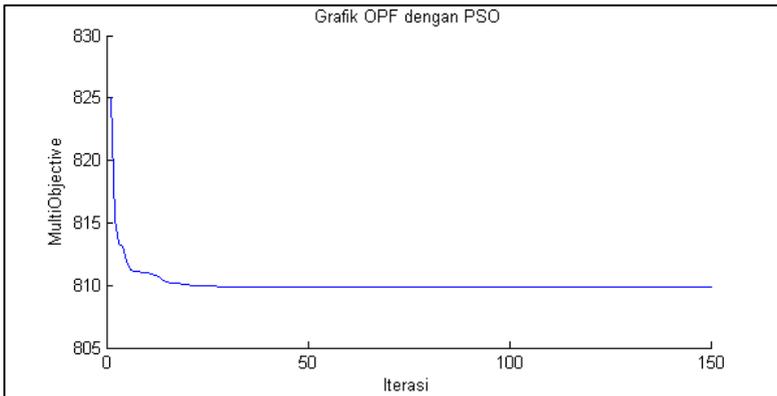
No	Parameter	Nilai (MW)	No	Parameter	Nilai (\$/Jam)
1	Daya Pembangkitan 1	167.46	7	Biaya Pembangkitan 1	440.083
2	Daya Pembangkitan 2	44.254	8	Biaya Pembangkitan 2	111.716
3	Daya Pembangkitan 3	23.270	9	Biaya Pembangkitan 3	57.114
4	Daya Pembangkitan 4	22.065	10	Biaya Pembangkitan 4	75.772
5	Daya Pembangkitan 5	10.966	11	Biaya Pembangkitan 5	35.904
6	Daya Pembangkitan 6	24.842	12	Biaya Pembangkitan 6	88.431

Daya aktif total, *losses* total dan biaya total dari simulasi *economic dispatch* menggunakan *Particle Swarm Optimization* ditunjukkan pada tabel 4.13.

Tabel 4.13. Total daya dan biaya pembangkitan simulasi economic dispatch dengan metode Particle Swarm Optimization

No	Pembangkit	Daya Pembangkitan (MW)	Losses (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	167.461	9.3660	440.083
2	Daya Pembangkitan 2	44.254		111.716
3	Daya Pembangkitan 3	23.270		57.114
4	Daya Pembangkitan 4	22.065		75.772
5	Daya Pembangkitan 5	10.966		35.904
6	Daya Pembangkitan 6	24.842		88.431
	Total	292.858	9.3660	809.0195

Grafik konvergensi metode *Particle Swarm Optimization* untuk penyelesaian simulasi *economic dispatch* ditunjukkan pada gambar 4.2.



Gambar 4.2. Grafik konvergensi simulasi economic dispatch metode Particle Swarm Optimization

Total daya aktif, losses dan total biaya pembangkitan hasil dari simulasi *Economic Dispatch* pada sistem IEEE 30 bus menggunakan metode *Particle Swarm Optimization* ditunjukkan pada tabel 4.13. Pada simulasi OPF ini digunakan 150 iterasi dan *Particle Swarm Optimization* mencapai titik konvergensinya pada iterasi ke-29 dengan nilai *fitness* mencapai 809.8931. Total daya aktif yang dibangkitkan mencapai

292.858 MW, dengan total *losses* 9.3660 MW dan biaya pembangkitan pada 809.0195 \$/jam.

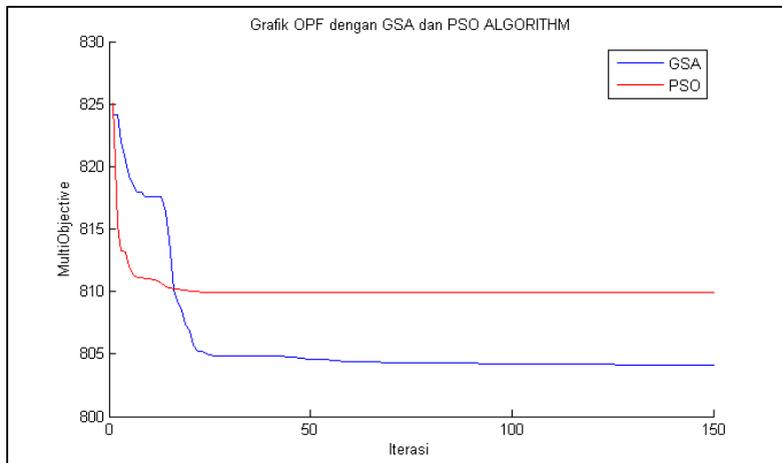
4. 2. 3. Perbandingan Simulasi *Economic Dispatch Gravitational Search Algorithm dan Particle Swarm Optimization*

Perbandingan dari hasil simulasi *economic dispatch* dengan menggunakan metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization* ditunjukkan pada tabel 4.14, meliputi nilai *fitness* dan biaya pembangkitan dari kedua metode.

Tabel 4.14. Perbandingan nilai *fitness* dan biaya pembangkitan simulasi *economic dispatch* metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Metode	Nilai <i>Fitness</i> Terbaik	Biaya Pembangkitan (\$/Jam)
GSA	804.1233	805.7860
PSO	809.8931	809.0195

Dengan perbandingan grafik konvergensi antara *Gravitational Search Algorithm* dan *Particle Swarm Optimization* pada gambar 4.3.



Gambar 4.3. Perbandingan grafik konvergensi nilai *fitness* simulasi *economic dispatch Gravitational Search Algorithm dan Particle Swarm Optimization*

Grafik konvergensi *Gravitational Search Algorithm* dan *Particle Swarm Optimization* ditunjukkan pada gambar 4.3. *Particle Swarm*

Optimization konvergen pada iterasi ke-29 dengan nilai *fitness* mencapai 809.8931. Sedangkan *Gravitational Search Algorithm* mencapai nilai konvergensi pada iterasi ke-126 dengan nilai *fitness* mencapai 804.1233 dan lebih kecil dari nilai *fitness* dari *Particle Swarm Optimization*. Nilai biaya pembangkitan yang dihasilkan oleh *Particle Swarm Optimization* adalah 809.0195 \$/jam dan biaya pembangkitan yang dihasilkan *Gravitational Search Algorithm* adalah sebesar 805.7860 \$/jam.

Dari pengujian *economic dispatch* dapat disimpulkan bahwa nilai konvergensi *Gravitational Search Algorithm* lebih kecil dan nilai biaya pembangkitan yang lebih ekonomis dengan nilai *fitness* 804.1233 dan biaya pembangkitan total sebesar 805.7860 \$/jam. Namun hasil tersebut dicapai oleh *Gravitational Search Algorithm* dengan selisih mencapai 97 iterasi lebih lambat dibandingkan dengan *Particle Swarm Optimization*.

4. 3. Simulasi Power Loss Minimization

Pada simulasi *power loss minimization* digunakan satu fungsi objektif saja yaitu penekanan pada rugi – rugi jaringan tanpa mempedulikan nilai *economic dispatch*. Berikut adalah hasil simulasi *power loss minimization* dengan menggunakan metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*.

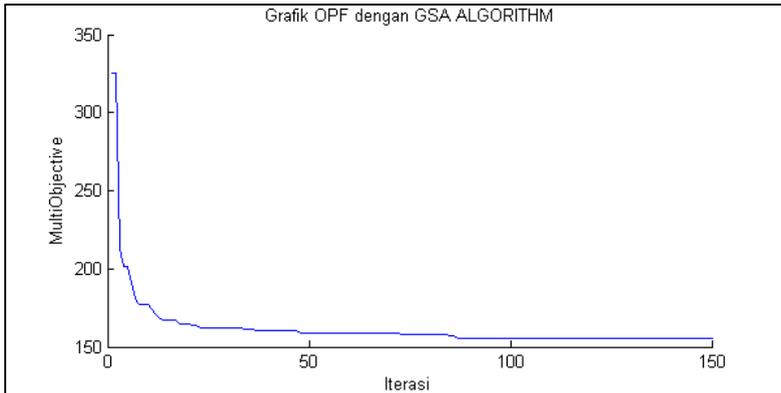
4. 3. 1. Simulasi Power Loss Minimization Metode Gravitational Search Algorithm

Dengan metode *Gravitational Search Algorithm* didapatkan total daya, rugi – rugi jaringan (*losses*) dan biaya pembangkitan yang ditunjukkan pada pada tabel 4.15.

Tabel 4.15. Total daya, biaya pembangkitan dan losses hasil simulasi power loss minimization dengan metode Gravitational Search Algorithm

No	Pembangkit	Daya Pembangkitan (MW)	Losses (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	52.278	4.3004	114.806
2	Daya Pembangkitan 2	80.000		252.000
3	Daya Pembangkitan 3	49.911		205.606
4	Daya Pembangkitan 4	34.980		123.891
5	Daya Pembangkitan 5	29.966		112.345
6	Daya Pembangkitan 6	39.974		159.871
	Total	287.109	4.3004	968.5193

Gambar konvergensi OPF *power loss minimization* dengan metode *Gravitational Search Algorithm* ditunjukkan oleh gambar 4.4.



Gambar 4.4. Grafik konvergensi nilai fitness simulasi power loss minimization dengan metode Gravitational Search Algorithm

Dari simulasi *power loss minimization* didapatkan grafik nilai *fitness* seperti gambar 4.4. Nilai *fitness* yang didapatkan dari hasil simulasi adalah sebesar 155.3305. Sedangkan nilai *losses* yang dihasilkan adalah sebesar 4.3004 MW. Titik konvergensi tersebut diraih oleh *Gravitational Search Algorithm* pada iterasi ke-87 dari 150 iterasi.

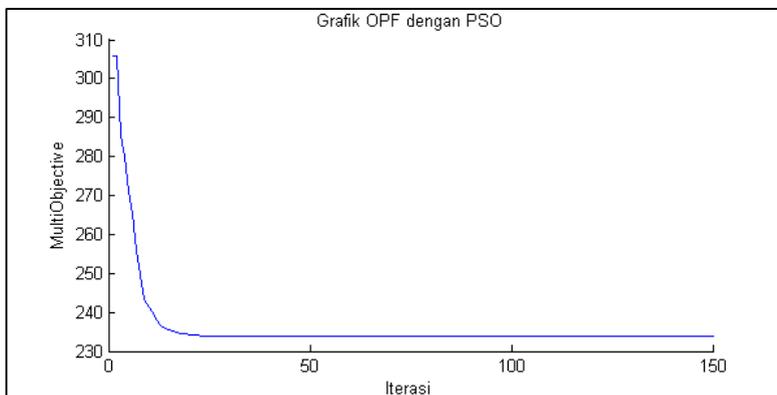
4.3.2. Simulasi Power Loss Minimization Metode Particle Swarm Optimization

Dengan metode *Particle Swarm Optimization* pada simulasi *power loss minimization* didapatkan total daya, rugi – rugi jaringan (*losses*) dan biaya pembangkitan yang ditunjukkan pada pada tabel 4.16.

Tabel 4.16. Total daya, biaya pembangkitan dan losses hasil simulasi power loss minimization dengan menggunakan metode Particle Swarm Optimization

No	Pembangkit	Daya Pembangkitan (MW)	Losses (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	94.078	4.7958	221.345
2	Daya Pembangkitan 2	55.378		150.579
3	Daya Pembangkitan 3	53.422		231.794
4	Daya Pembangkitan 4	24.298		83.892
5	Daya Pembangkitan 5	30.731		115.805
6	Daya Pembangkitan 6	30.043		112.695
	Total	287.950	4.7958	916.1102

Gambar konvergensi OPF simulasi power loss minimization dengan metode Particle Swarm Optimization ditunjukkan oleh gambar 4.5.



Gambar 4.5. Grafik konvergensi nilai fitness simulasi power loss minimization dengan metode Particle Swarm Optimization

Hasil dari simulasi power loss minimization dengan menggunakan Particle Swarm Optimization menghasilkan nilai fitness sebesar 233.7550 dengan total losses mencapai 4.7958 MW. Nilai konvergensi tersebut dapat diraih oleh metode Particle Swarm Optimization pada iterasi ke-24 dari 150 iterasi.

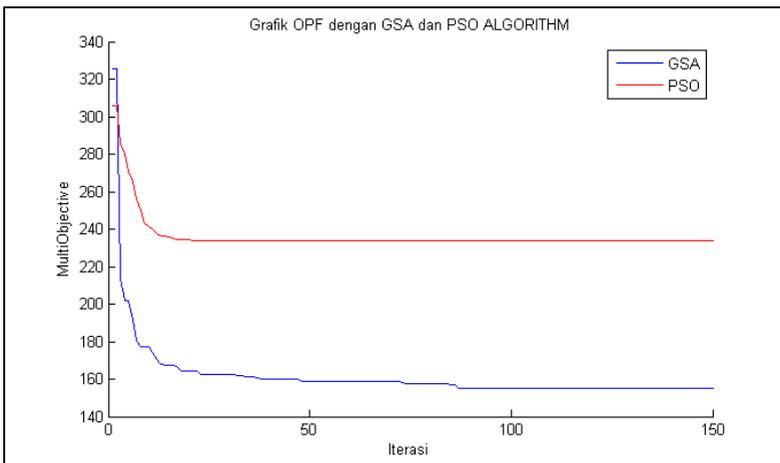
4.3.3. Perbandingan Simulasi *Losses Minimization Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Perbandingan hasil OPF *power loss minimization* antara *Gravitational Search Algorithm* dan *Particle Swarm Optimization* ditunjukkan pada tabel 4.17.

Tabel 4.17. Perbandingan nilai *fitness* dan nilai *losses* metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Metode	Nilai <i>Fitness</i> Terbaik	Nilai <i>Losses</i> (MW)
GSA	155.3305	4.3004
PSO	233.7550	4.7958

Dengan perbandingan nilai *fitness* keduanya adalah 233.7550 untuk *Particle Swarm Optimization* dan 155.3305 untuk *Gravitational Search Algorithm*. Sedangkan nilai *losses* yang diperoleh adalah sebesar 4.3004 MW untuk *Gravitational Search Algorithm*, lebih kecil dari *Particle Swarm Optimization* yang menghasilkan nilai *losses* sebesar 4.7958 MW. Gambar 4.6 memperlihatkan nilai *fitness* simulasi *power loss minimization* untuk *Gravitational Search Algorithm* dan *Particle Swarm Optimization*.



Gambar 4.6. Perbandingan grafik konvergensi nilai *fitness* simulasi *power loss minimization* dengan metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Terlihat dari grafik tersebut perbandingan grafik konvergensi *Gravitational Search Algorithm* dan *Particle Swarm Optimization*. Pada grafik *Particle Swarm Optimization* titik konvergensi didapatkan pada iterasi ke-24 dari 150 iterasi sedangkan pada grafik *Gravitational Search Algorithm* titik konvergensi didapatkan pada titik ke-87 dari 150 iterasi. Dari simulasi *power loss minimization* menunjukkan *Gravitational Search Algorithm* memiliki nilai konvergensi yang lebih rendah dari *Particle Swarm Optimization*, namun titik konvergensi dicapai pada iterasi yang lebih lambat, yakni 63 iterasi lebih lambat dibandingkan dengan *Particle Swarm Optimization*.

4. 4. Simulasi *Economic Power Loss Minimization*

Dalam simulasi *Economic Power Loss Minimization* digunakan dua fungsi objektif dalam pengujian metode OPF yakni meminimalisasi biaya pembangkitan dan rugi – rugi jaringan. Kedua fungsi objektif dijalankan secara bersamaan atau simultan dalam setiap iterasi. Metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization* akan digunakan untuk menyelesaikan masalah *Economic Power Loss Minimization*, dengan metode *Gravitational Search Algorithm* sebagai metode utama dan *Particle Swarm Optimization* sebagai metode pembanding. Hasil pengujian kedua metode ini kemudian dibandingkan berdasarkan nilai gabungan, yakni nilai *fitness* yang merupakan representasi dari biaya pembangkitan yang lebih ekonomis dan nilai rugi – rugi jaringan yang lebih kecil.

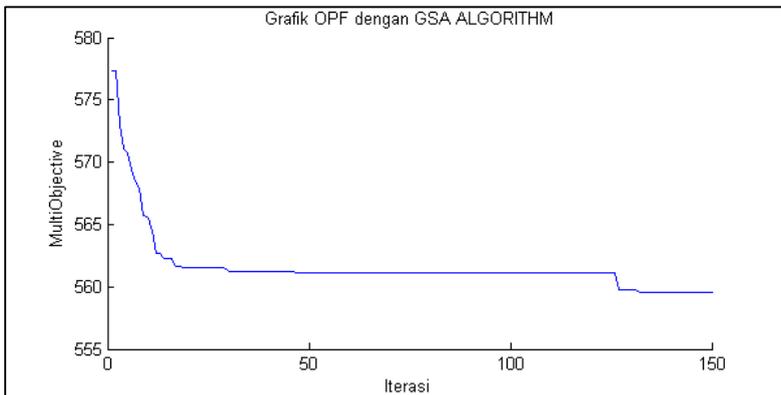
4. 4. 1. Simulasi *Economic Power Loss Minimization* Metode *Gravitational Search Algorithm*

Pada simulasi *Economic Power Loss Minimization* terdapat dua fungsi objektif yang diaktifkan, yakni minimalisasi biaya pembangkitan dan rugi – rugi jaringan. Dengan metode *Gravitational Search Algorithm* didapatkan daya dan biaya pembangkitan yang ditunjukkan pada pada tabel 4.18.

Tabel 4.18. Daya aktif dan biaya pembangkitan simulasi *Economic Power Loss Minimization* menggunakan metode *Gravitational Search Algorithm*

No	Parameter	Nilai (MW)	No	Parameter	Nilai (\$/Jam)
1	Daya Pembangkitan 1	108.60	7	Biaya Pembangkitan 1	261.433
2	Daya Pembangkitan 2	68.619	8	Biaya Pembangkitan 2	202.481
3	Daya Pembangkitan 3	36.794	9	Biaya Pembangkitan 3	121.407
4	Daya Pembangkitan 4	26.256	10	Biaya Pembangkitan 4	91.081
5	Daya Pembangkitan 5	17.451	11	Biaya Pembangkitan 5	59.965
6	Daya Pembangkitan 6	31.641	12	Biaya Pembangkitan 6	119.950

Grafik konvergensi *Gravitational Search Algorithm* untuk penyelesaian simulasi *Economic Power Loss Minimization* ditunjukkan pada gambar 4.7.



Gambar 4.7. Grafik konvergensi simulasi *Economic Power Loss Minimization* dengan menggunakan metode *Gravitational Search Algorithm*

Total daya aktif, *losses* dan total biaya pembangkitan hasil dari simulasi *Economic Power Loss Minimization* pada sistem IEEE 30 bus menggunakan metode *Gravitational Search Algorithm* ditunjukkan pada tabel 4.19. Pada simulasi ini digunakan 150 iterasi dan *Gravitational Search Algorithm* mencapai titik konvergensinya pada iterasi ke-132 dengan nilai *fitness* mencapai 559.5379. Total daya aktif yang dibangkitkan mencapai 289.363 MW, dengan total *losses* 4.5189 MW dan biaya pembangkitan pada 856.3164 \$/jam.

Tabel 4.19. Daya aktif, *losses*, dan biaya pembangkitan hasil simulasi *Economic Power Loss Minimization* menggunakan metode *Gravitational Search Algorithm*

No	Pembangkit	Daya Pembangkitan (MW)	Losses (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	108.602	4.5189	261.433
2	Daya Pembangkitan 2	68.619		202.481
3	Daya Pembangkitan 3	36.794		121.407
4	Daya Pembangkitan 4	26.256		91.081
5	Daya Pembangkitan 5	17.451		59.965
6	Daya Pembangkitan 6	31.641		119.950
	Total	289.363	4.5189	856.3164

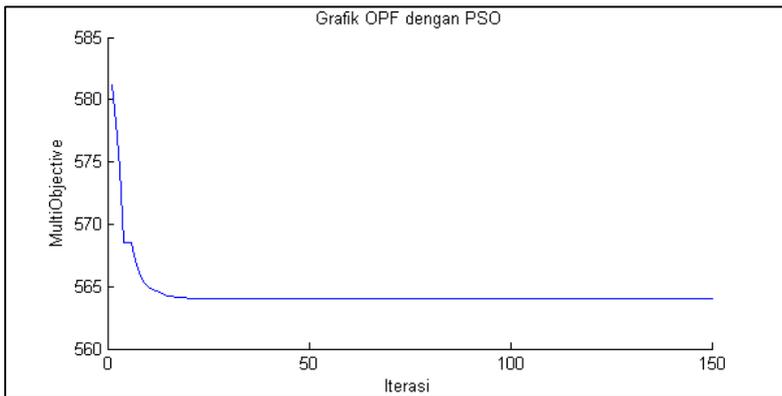
4. 4. 2. Simulasi *Economic Power Loss Minimization* Metode *Particle Swarm Optimization*

Pada simulasi *Economic Power Loss Minimization* terdapat dua fungsi objektif yang diaktifkan, yakni minimalisasi biaya pembangkitan dan rugi – rugi jaringan. Dengan metode *Particle Swarm Optimization* didapatkan daya dan biaya pembangkitan yang ditunjukkan pada pada tabel 4.20.

Tabel 4.20. Daya aktif dan biaya pembangkitan hasil simulasi *Economic Power Loss Minimization* menggunakan metode *Particle Swarm Optimization*

No	Parameter	Nilai (MW)	No	Parameter	Nilai (\$/Jam)
1	Daya Pembangkitan 1	100.961	7	Biaya Pembangkitan 1	240.146
2	Daya Pembangkitan 2	61.329	8	Biaya Pembangkitan 2	173.148
3	Daya Pembangkitan 3	45.958	9	Biaya Pembangkitan 3	177.970
4	Daya Pembangkitan 4	25.554	10	Biaya Pembangkitan 4	88.498
5	Daya Pembangkitan 5	27.939	11	Biaya Pembangkitan 5	103.332
6	Daya Pembangkitan 6	26.761	12	Biaya Pembangkitan 6	98.185

Grafik konvergensi *Particle Swarm Optimization* untuk penyelesaian simulasi *Economic Power Loss Minimization* ditunjukkan pada gambar 4.8.



Gambar 4.8. Grafik konvergensi simulasi *Economic Power Loss Minimization* menggunakan metode *Particle Swarm Optimization*

Total daya aktif, *losses* dan total biaya pembangkitan hasil dari simulasi *Economic Power Loss Minimization* pada sistem IEEE 30 bus menggunakan metode *Particle Swarm Optimization* ditunjukkan pada tabel 4.21. Pada simulasi OPF ini digunakan 150 iterasi dan *Particle Swarm Optimization* mencapai titik konvergensinya pada iterasi ke-23 dengan nilai *fitness* mencapai 563.9798. Total daya aktif yang dibangkitkan mencapai 288.502 MW, total *losses* 5.0680 MW dan biaya pembangkitan pada 881.2797 \$/jam.

Tabel 4.21. Daya aktif, *losses*, dan biaya pembangkitan hasil simulasi *Economic Power Loss Minimization* menggunakan metode *Particle Swarm Optimization*

No	Pembangkit	Daya Pembangkitan (MW)	Losses (MW)	Biaya (\$/Jam)
1	Daya Pembangkitan 1	94.962	5.0680	240.146
2	Daya Pembangkitan 2	70.737		173.148
3	Daya Pembangkitan 3	41.345		177.970
4	Daya Pembangkitan 4	21.572		88.498
5	Daya Pembangkitan 5	28.514		103.332
6	Daya Pembangkitan 6	31.492		98.185
	Total	288.502	5.0680	881.2797

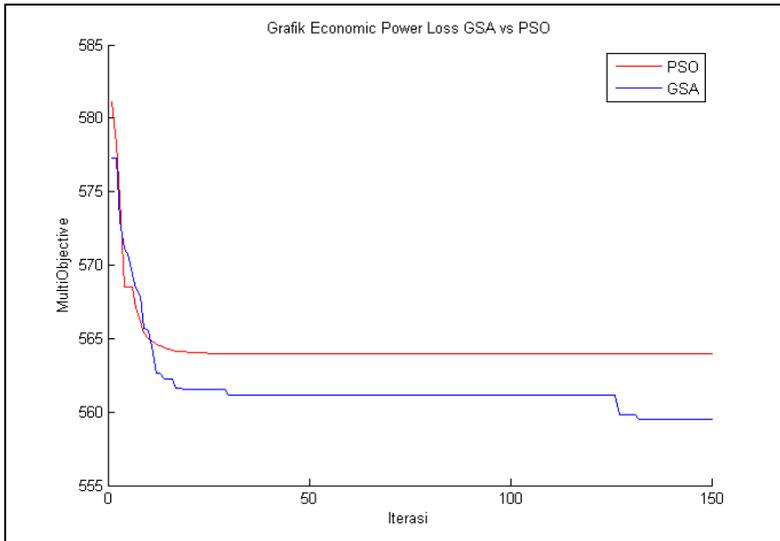
4. 4. 3. Perbandingan Hasil *Economic Power Loss Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Perbandingan dari hasil simulasi *Economic Power Loss Minimization* dengan menggunakan *Gravitational Search Algorithm* dan *Particle Swarm Optimization* ditunjukkan pada tabel 4.22 yang berisi nilai *fitness* terbaik, nilai rugi – rugi jaringan (*losses*), dan nilai biaya pembangkitan yang dihasilkan.

Tabel 4.22. Perbandingan hasil simulasi *Economic Power Loss Minimization* metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Metode	Nilai Konvergensi Terbaik	Power Loss (MW)	Biaya Total Pembangkitan (\$/Jam)
GSA	559.5379	4.5189	856.3164
PSO	563.9798	5.0680	881.2797

Dengan perbandingan grafik konvergensi *Gravitational Search Algorithm* dan *Particle Swarm Optimization* pada gambar 4.9.



Gambar 4.9. Perbandingan grafik konvergensi hasil simulasi *Economic Power Loss Minimization* menggunakan metode *Gravitational Search Algorithm* dan *Particle Swarm Optimization*

Grafik konvergensi *Gravitational Search Algorithm* dan *Particle Swarm Optimization* ditunjukkan pada gambar 4.9. *Particle Swarm Optimization* konvergen pada iterasi ke-23 dengan nilai *fitness* mencapai 563.9798. Sedangkan *Gravitational Search Algorithm* mencapai nilai konvergensi pada iterasi ke-132 dengan nilai *fitness* 559.5379 dan lebih kecil dari nilai *fitness* dari *Particle Swarm Optimization*.

Dari pengujian kedua algoritma ini dapat disimpulkan bahwa nilai konvergensi *Gravitational Search Algorithm* lebih kecil dengan nilai *fitness* 559.5379 dengan biaya pembangkitan yang lebih ekonomis sebesar 856.3164 \$/jam sekaligus nilai rugi – rugi jaringan yang lebih kecil dengan 4.5189 MW. Namun hasil tersebut didapatkan oleh *Gravitational Search Algorithm* dengan 109 iterasi lebih lambat dibandingkan dengan *Particle Swarm Optimization*.

Halaman ini sengaja dikosongkan

BAB 5

KESIMPULAN DAN SARAN

5. 1. Kesimpulan

1. Dalam simulasi OPF *single objective* : *economic dispatch*, nilai konvergensi *Gravitational Search Algorithm* memberikan nilai *fitness* yang lebih baik bila dibandingkan dengan *Particle Swarm Optimization*. Nilai *fitness* yang didapatkan oleh *Gravitational Search Algorithm* adalah 804.1233 pada iterasi ke-126 sedangkan nilai *fitness Particle Swarm Optimization* 809.8931 pada iterasi ke-29 dari 150 iterasi.
2. Pada simulasi OPF dengan *single objective* : *Power Loss Minimization*, nilai konvergensi *Gravitational Search Algorithm* juga lebih kecil dari nilai *fitness* yang didapatkan oleh *Particle Swarm Optimization*. Nilai *fitness* terbaik dari *Gravitational Search Algorithm* adalah 155.3305 pada iterasi ke-87 sedangkan dari *Particle Swarm Optimization* adalah 233.7550 pada iterasi ke-24.
3. Dalam simulasi OPF *multi objective* : *Economic Power Loss Minimization*, nilai konvergensi *Gravitational Search Algorithm* kembali memberikan nilai yang lebih baik dibandingkan dengan *Particle Swarm Optimization*. Nilai *fitness Gravitational Search Algorithm* pada simulasi mencapai 559.5379 pada iterasi ke-132 sedangkan nilai *fitness Particle Swarm Optimization* sebesar 563.9798 pada iterasi ke-23.
4. Dalam simulasi OPF yang diujikan pada Tugas Akhir ini nilai konvergensi *Particle Swarm Optimization* didapat pada iterasi yang lebih cepat dibandingkan dengan *Gravitational Search Algorithm* pada keseluruhan simulasi, namun hasil nilai konvergensi yang dihasilkan oleh *Gravitational Search Algorithm* masih lebih unggul dari *Particle Swarm Optimization*.
5. Berdasarkan hasil simulasi baik dengan *single objective* maupun *multi objective*, *Gravitational Search Algorithm* merupakan metode yang lebih baik dari segi nilai konvergensinya (nilai *fitness*) dibandingkan dengan *Particle Swarm Optimization*.

5. 2. Saran

1. Metode yang diujikan dalam penelitian ini diharapkan dapat diterapkan pada sistem riil, dengan syarat sistem tersebut memenuhi kriteria minimum untuk *Optimal Power Flow*, seperti data impedansi dan admitansi yang valid, interkoneksi antar pembangkit, dan data – data jaringan lainnya, termasuk data penunjang lainnya seperti keadaan dan batasan kompleks pada jaringan yang tidak stabil.
2. Untuk kepentingan penelitian *Multiobjective* OPF dengan metode *Gravitational Search Algorithm*, diperlukan adanya pengembangan lebih lanjut dengan batasan – batasan kompleks dalam permasalahan sistem tenaga nyata yang harus direpresentasikan dalam bentuk program.
3. Dengan adanya perubahan data penunjang pada point 1 dan 2, maka nilai – nilai dalam parameter dalam metode harus dirubah kembali agar didapatkan hasil yang lebih maksimal.

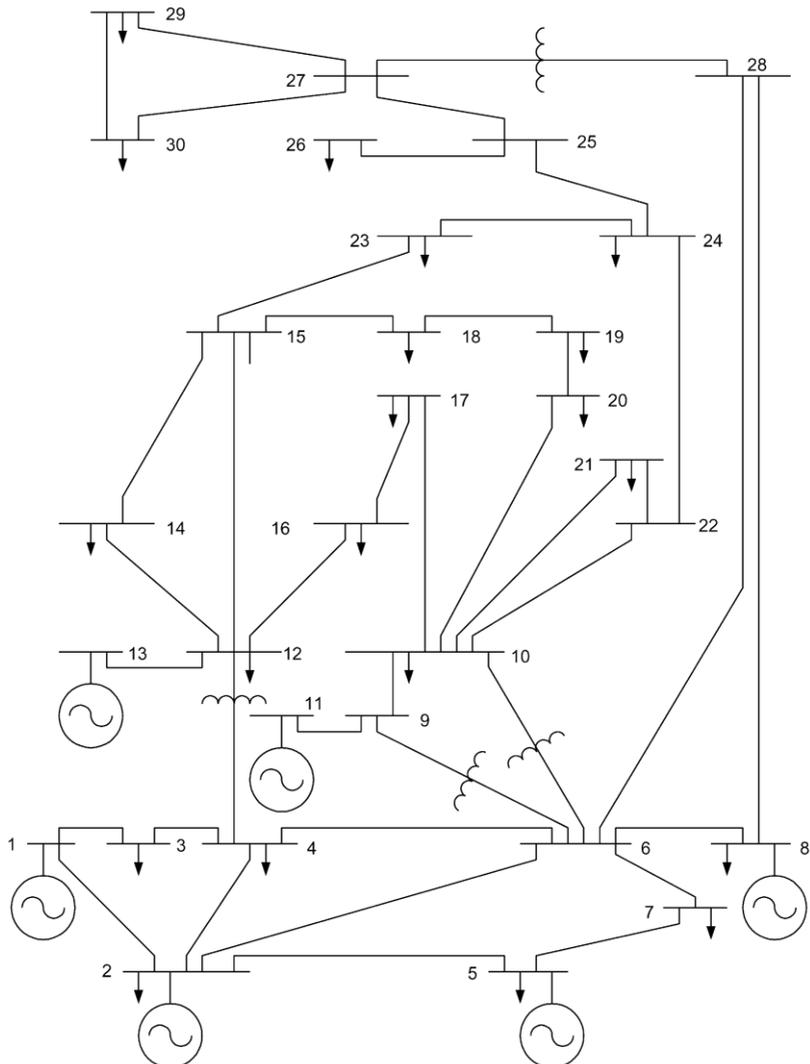
DAFTAR PUSTAKA

- [1] Durman, Serhat ; Uğur Güvenç; Yusuf Sönmez; Nuran Yörükeren, “Optimal Power Flow using Gravitational Search Algorithm”, ELSEVIER, 2012.
- [2] Esmat Rashedi, Hossein Nezamabadi-pour, Saeid Sryazdi, “GSA : A Gravitational Search Algorithm”, 2009.
- [3] Hadi Saadat, Power System Analysis, second edition. McGraw-Hill, 2004.
- [4] Allen J. Wood, Bruce F. Wollenberg, Power Generation, Operation and Control, John Wiley & Sons, 1984.
- [5] Jizhong Zhu, PhD, Optimization of Power System Operation, 2009.
- [6] Mary B. Cain, Richard P. O’Neill, Anya Castillo, History of Optimal Power Flow and Formulations, 2012.
- [7] Xin-She Yang, Engineering Optimization, An Introduction with Metaheuristic Applications, John Wiley & Sons, 2010.
- [8] J. Hazra, A. K. Sinha, A Multi-Objective Optimal Power Flow Using Particle Swarm Optimization, 2010.
- [9] M. A. Abido, “Multiobjective Particle Swarm for Environmental/Economic Dispatch Problem” *The 8th International Power Engineering Conference*, 2007, pp. 1385-1390.
- [10] P. Venkatesh, K. Y. Lee, Multi-Objective Evolutionary Programming for Economic Emission Dispatch problem, *Power and Energy Society General Meeting-Conversion and Delivery of Electrical Energy in the 21st Century*, 2008, pp. 1-8.
- [11] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In Proceedings of the 1995 IEEE International Conference on Neural Networks. IEEE Service Center, Piscataway, 1995Xin-She Yang, Engineering Optimization, An Introduction with Metaheuristic Applications, John Wiley & Sons, 2010.
- [12] M. A. Abido, “Optimal Power Flow Using Particle Swarm Optimization” *International Journal of Electrical Power and Energy Systems*, Vol. 24, No. 7, October 2002, pp. 563-571.
- [13] Mahor A., Prasad V., Rangnekar S., *Economic Dispatch Using Particle Swarm Optimization: a Review*, Renewable and Sustainable Energy Reviews, 2009, 13, p. 2134-2141.
- [14] Jong-Bac Park, A Particle Swarm Optimization for Economic Dispatch With Nonsmooth Cost Functions, 2005

- [15] J. B. Park, K. S. Lee, J. R. Shin, and K. Y. Lee, A *particle* swarm optimization for economic dispatch with nonsmooth cost functions, *IEEE Trans. On Power Systems*, Vol. 20, No. 1, pp. 34-42, Feb. 2005.

LAMPIRAN

- *Plant sistem IEEE 30 bus (sumber : M. A. Abido / Electrical Power and Energy Systems 24 (2002) 563-571)*



Halaman ini sengaja dikosongkan

RIWAYAT PENULIS



Rizki Nurfajri Adhim dilahirkan di Gresik, 26 Februari 1992. Merupakan putra pertama dari tiga bersaudara pasangan Bapak Mundakir dan Ibu Nindyarini Peni Hidayanti. Penulis menempuh jenjang pendidikan dari SD Muhammadiyah GKB Gresik, dan melanjutkan ke SMP Muhammadiyah 12 Gresik. Kemudian penulis melanjutkan ke SMA Negeri 1 Gresik hingga lulus tahun 2010. Pada tahun yang sama penulis melanjutkan studinya ke Institut Teknologi Sepuluh Nopember Surabaya Fakultas Teknologi Industri Jurusan Teknik Elektro. Pada tahun 2012 penulis mengambil bidang studi Teknik Komputer dan Telematika, namun pada semester genap 2014, penulis berpindah haluan dan mengambil bidang studi Teknik Sistem Tenaga.

Penulis dapat dihubungi melalui email : rizki.nfajri@gmail.com

Halaman ini sengaja dikosongkan