

# OPTIMAL GENERATOR SCHEDULING BASED ON PARTICLE SWARM OPTIMIZATION

Maickel Tuegeh<sup>1</sup>, Adi Soeprijanto<sup>2</sup>, Mauridhi Hery P<sup>3</sup>

<sup>1</sup>Jurusan Teknik Elektro ITS, Surabaya 60111, Jurusan Teknik Elektro UNSRAT Manado.

<sup>2</sup>Jurusan Teknik Elektro ITS, Surabaya 60111

<sup>3</sup>Jurusan Teknik Elektro ITS, Surabaya 60111

email: [m\\_tuegeh@yahoo.co.id](mailto:m_tuegeh@yahoo.co.id), [m\\_tuegeh@elect-eng.its.ac.id](mailto:m_tuegeh@elect-eng.its.ac.id), [adisup@ee.its.ac.id](mailto:adisup@ee.its.ac.id), [hery@ee.its.ac.id](mailto:hery@ee.its.ac.id)

## Abstrak

Generator scheduling merupakan bagian penting dalam pengoperasian sistem tenaga listrik. Tidak ada penjadwalan yang baik dapat menyebabkan biaya yang sangat besar dalam proses pengoperasian terutama pada sisi pembangkitan, juga dapat menyebabkan tidak ada koordinasi dalam menyalurkan tenaga listrik. Optimasi penjadwalan dari generator diperoleh dengan kecerdasan buatan diantaranya menggunakan teknik Particle Swarm Optimization (PSO). Hasil simulasi menunjukkan bahwa metode yang digunakan memberikan performa yang sangat baik dan hasil ini juga dibandingkan dengan metoda iterasi lamda untuk melihat keakuratan dari hasil yang didapat. Metode ini diaplikasikan pada dua contoh sistem dan diselesaikan dengan menggunakan Matlab.

**Kata kunci :** *Optimal Generator Scheduling, PSO.*

## 1. PENDAHULUAN

Semakin berkembangnya jumlah penduduk menyebabkan penyebaran penduduk yang semakin luas. Hal ini juga berdampak pada kebutuhan akan energi listrik yang semakin hari makin meningkat dan juga tersebar luas. Kebutuhan akan listrik yang sangat besar tidak dapat dipenuhi oleh satu atau dua unit pembangkit tapi harus dipenuhi oleh beberapa unit pembangkit pada beberapa daerah dan tersebar luas. Pengoptimalan *generator scheduling* dalam sistem tenaga listrik sangat diperlukan, karena proses pembangkitan dan penyaluran dalam sistem tenaga listrik memerlukan biaya yang sangat besar. Koordinasi antar pembangkitan diperlukan dalam upaya melakukan *optimal generator scheduling* untuk memperoleh biaya yang minimum. Teknik solusi dalam masalah optimisasi dapat menggunakan kecerdasan buatan. Salah satu teknik solusi masalah optimisasi dalam kecerdasan buatan menggunakan teknik *Particle Swarm Optimization (PSO)*. Kelebihan utama algoritma PSO adalah mempunyai konsep sederhana, mudah diimplementasikan, dan efisien dalam perhitungan jika dibandingkan dengan algoritma matematika dan teknik optimisasi heuristik lainnya. *Optimal generator scheduling* berbasis *Particle Swarm Optimization (PSO)* merupakan penelitian yang dilakukan untuk mendapatkan *optimal generator scheduling* suatu sistem dengan memperoleh biaya yang minimum. Hasil simulasi menunjukkan bahwa metode yang digunakan memberikan performa yang sangat baik dan hasil ini juga dibandingkan dengan metoda iterasi lamda untuk melihat keakuratan dari hasil yang didapat. Metode ini diaplikasikan pada dua contoh sistem dan diselesaikan dengan menggunakan Matlab.

## 2. TINJAUAN PUSTAKA

### 2.1. Biaya Pembangkitan

Masalah *economic dispatch* adalah pembagian pembebanan pada setiap unit pembangkit sehingga diperoleh kombinasi unit pembangkit yang dapat memenuhi kebutuhan beban dengan biaya yang optimum atau dengan kata lain untuk mencari nilai optimum dari output daya dari kombinasi unit pembangkit yang bertujuan untuk meminimalkan total biaya pembangkitan. Biaya operasi dari suatu sistem tenaga listrik merupakan biaya terbesar dalam pengoperasian suatu perusahaan listrik. Biaya yang dikeluarkan oleh suatu perusahaan listrik untuk menghasilkan energi listrik dalam suatu sistem tenaga listrik ditentukan oleh biaya investasi dan biaya operasi atau biaya produksi. Besar biaya investasi tidak bergantung pada besar daya keluaran pembangkit tetapi bergantung pada besar kapasitas daya terpasang pembangkit. Biaya investasi meliputi biaya pembangunan pusat pembangkit, jaringan transmisi dan distribusi serta peralatan sistem lainnya, sedangkan biaya operasi atau biaya produksi merupakan semua biaya yang dikeluarkan dalam pengoperasian suatu pembangkit. Untuk sistem yang sudah ada (telah beroperasi) biaya investasi telah tertentu besarnya. Meminimumkan biaya operasi pembangkitan adalah merupakan optimisasi, sehingga optimisasi pembangkitan dapat didefinisikan sebagai suatu proses pembangkitan yang bertujuan untuk mengoptimalkan daya dan meminimumkan biaya pembangkitan.

**2.1.1 Karakteristik Input-Output Pembangkit**

Karakteristik input output pembangkit termal adalah karakteristik yang menggambarkan hubungan antara input bahan bakar (liter/jam) dan output yang dihasilkan oleh pembangkit (MW).

Pada umumnya karakteristik input output pembangkit termal didekati dengan fungsi polinomial orde dua yaitu :

$$H_i = \alpha_i + \beta_i P_i + \gamma_i P_i^2 \tag{1}$$

dengan :

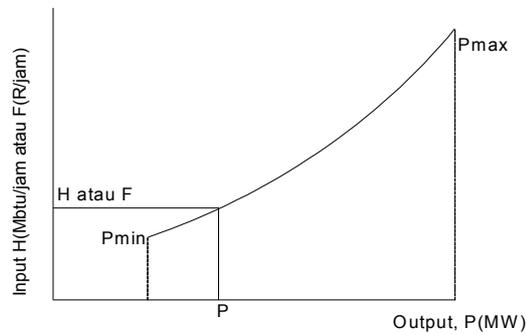
$H_i$  = Input bahan bakar pembangkit termal ke- $i$  (liter/jam)

$P_i$  = Output pembangkit termal ke- $i$  (MW)

$\alpha_i, \beta_i, \gamma_i$  = Konstanta input-output pembangkit termal ke- $i$ . Penentuan parameter  $\alpha_i, \beta_i$ , dan  $\gamma_i$  membutuhkan data yang berhubungan dengan input bahan bakar  $H_i$  dan output pembangkit  $P_i$ .

Gambar 1 adalah karakteristik input-output unit pembangkit termal dapat dinyatakan sebagai berikut :

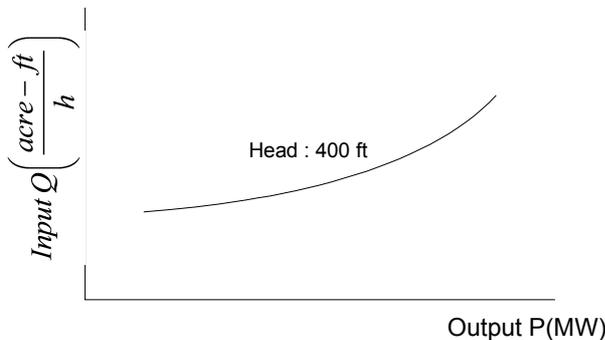
1. Input dari pembangkit dinyatakan dalam  $H = \text{Mbtu/jam}$  (energi panas yang dibutuhkan), atau  $F = \text{R/jam}$  (total biaya bahan bakar)
2. Output dari pembangkit dinyatakan dalam  $P = \text{MW}$  (daya).



**Gambar 1** Karakteristik Input-Output unit pembangkit termal (ideal)

Data karakteristik input-output biasanya diperoleh dari hasil perhitungan desain atau dari hasil pengukuran. Jika data yang digunakan adalah data dari hasil pengukuran maka akan diperoleh kurva yang tidak kontinyu (*smooth*). Unit pembangkit termal mempunyai batas kritis operasi minimum dan maksimum, batas beban minimum umumnya disebabkan oleh kestabilan pembakaran dan masalah desain generator, sebagai contoh beberapa unit pembangkit termal tidak dapat beroperasi di bawah 30 % dari kapasitas desain.

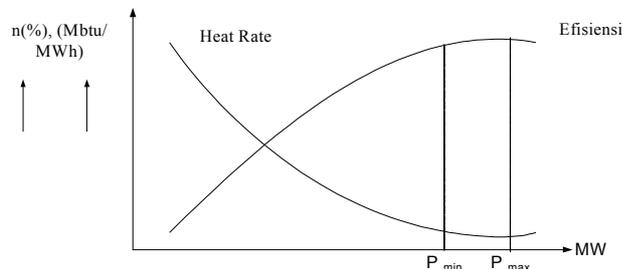
Unit pembangkit hidro mempunyai karakteristik input-output yang mirip dengan unit pembangkit termal. Karakteristik input-output pembangkit hidro menggambarkan hubungan antara input penggerak mula, berupa volume air yang dialirkan diantara sudu-sudu turbin persatuan waktu dan outputnya adalah daya listrik dari generator dalam MW. Gambar 2 menunjukkan karakteristik input-output untuk pembangkit tenaga air dengan head tetap.



**Gambar 2** Karakteristik input-output unit pembangkit hidro

### 2.1.2 Batas Pembebanan Ekonomis Pembangkit Termal.

Umumnya mesin pembangkit tenaga listrik mempunyai batas pembebanan yang dibatasi oleh kapasitas ekonomis minimum dan kapasitas ekonomis maksimum dan juga keterbatasan kerja komponen-komponen mesin, sehingga dalam pembebanan pembangkit termal harus diperhatikan karakteristik efisiensi dan karakteristik kenaikan panas tiap unit pembangkit termal seperti pada Gambar 3



Gambar 3 Batas Pembebanan Suatu Unit Pembangkit

## 2.2 PARTICLE SWARM OPTIMIZATION (PSO)

### 2.2.1 Dasar PSO

Algoritma *Particle Swarm Optimization* (PSO) diperkenalkan oleh Kennedy dan Eberhart pada tahun 1995, proses algoritmanya diinspirasi oleh perilaku sosial dari binatang, seperti sekumpulan burung dalam suatu swarm. *Particle Swarm Optimization* (PSO) adalah salah satu dari teknik komputasi *evolutioner*, yang mana populasi pada PSO didasarkan pada penelusuran algoritma dan diawali dengan suatu populasi yang random yang disebut dengan *particle*. Masalah *optimal generator scheduling* sudah sering diteliti, contoh penelitian yang telah dilakukan yaitu *Generation maintenance scheduling in restructured power sistem* oleh Antonio J. Conejo, dan kawan-kawan tahun 2004, dan *Security-constrained optimal generation scheduling in large-scale power sistem* oleh Jorge Martínez-Crespo, dan kawan-kawan tahun 2006. Berbeda dengan teknik komputasi *evolutioner lainnya*, setiap *particle* di dalam PSO juga berhubungan dengan suatu *velocity*. *Particle-particle* tersebut bergerak melalui penelusuran ruang dengan *velocity* yang dinamis yang disesuaikan menurut perilaku historisnya. Oleh karena itu, *particle-particle* mempunyai kecenderungan untuk bergerak ke area penelusuran yang lebih baik setelah melewati proses penelusuran. *Particle Swarm Optimization* (PSO) mempunyai kesamaan dengan *genetic algorithm* yang mana dimulai dengan suatu populasi yang random dalam bentuk matriks. Namun PSO tidak memiliki operator evolusi yaitu *crossover* dan *mutasi* seperti yang ada pada *genetic algorithm*. Baris pada matriks disebut *particle* atau dalam *genetic algorithm* sebagai kromosom yang terdiri dari nilai suatu *variable*. Setiap *particle* berpindah dari posisinya semula ke posisi yang lebih baik dengan suatu *velocity*. Pada algoritma PSO vektor *velocity* diupdate untuk masing-masing *particle* kemudian menjumlahkan vektor *velocity* tersebut ke posisi *particle*. *Update velocity* dipengaruhi oleh kedua solusi yaitu *global best* yang berhubungan dengan biaya yang paling rendah yang pernah diperoleh dari suatu *particle* dan solusi *local best* yang berhubungan dengan biaya yang paling rendah pada populasi awal. Jika solusi *local best* mempunyai suatu biaya yang kurang dari biaya solusi global yang ada, maka solusi *local best* menggantikan solusi *global best*.

Prosedur standar untuk menerapkan algoritma PSO adalah sebagai berikut :

1. Inisialisasi populasi dari *particle-particle* dengan posisi dan *velocity* secara random dalam suatu ruang dimensi penelusuran.
2. Evaluasi fungsi *fitness* optimisasi yang diinginkan di dalam variabel  $d$  pada setiap *particle*.
3. Membandingkan evaluasi *fitness* *particle* dengan  $P_{best}$ -nya. Jika nilai yang ada lebih baik dibandingkan dengan nilai  $P_{best}$ -nya, maka  $P_{best}$  diset sama dengan nilai tersebut dan  $P_i$  sama dengan lokasi *particle* yang ada  $X_i$  dalam ruang dimensional  $d$ .
4. Identifikasi *particle* dalam lingkungan dengan hasil terbaik sejauh ini.
5. *Update velocity* dan posisi *particle*.
6. Kembali ke step 2 sampai kriteria terpenuhi, biasanya berhenti pada nilai *fitness* yang cukup baik atau sampai pada jumlah maksimum iterasi.

### 2.2.2 Koefisien Akselerasi

Pada umumnya nilai-nilai untuk koefisien akselerasi  $c_1$  dan  $c_2 = 2.0$ . Namun demikian, nilai koefisien akselerasi tersebut dapat ditentukan sendiri yang digunakan di dalam penelitian yang berbeda, biasanya nilai  $c_1$  dan  $c_2$  adalah sama dan berada pada rentang antara 0 sampai 4.

### 2.2.3 Inertia Weight

Perubahan *velocity* pada algoritma PSO terdiri atas tiga bagian yaitu *social part*, *cognitive part* dan *momentum part*. Ketiga bagian tersebut menentukan keseimbangan antara kemampuan penelusuran *global* dan *local*, oleh karena itu dapat memberikan performansi yang baik pada PSO.

Parameter *inertia weight* digabungkan dengan *social part* di dalam algoritma PSO standar. Persamaan dinamis dari PSO dengan *inertia weight* ( $w$ ) dimodifikasi menjadi :

$$V_{id} = wV_{id} + c_1rand_1(P_{id} - X_{id}) + c_2rand_2(P_{gd} - X_{id}) \quad (2)$$

dan

$$X_{id} = X_{id} + V_{id} \quad (3)$$

Persamaan (2) sama dengan persamaan (3) kecuali ada penambahan parameter baru yaitu *inertia weight* ( $w$ ). Inertia weight diperkenalkan untuk keseimbangan antara kemampuan penelusuran *global* dan *local*. Secara umum parameter *inertia weight* ( $w$ ) diperoleh dengan menggunakan persamaan berikut :

$$w = w_{\max} - \frac{w_{\max} - w_{\min}}{Iter_{\max}} Iter \quad (4)$$

Dengan :

$w_{\max} - w_{\min}$  = Weight awal dan akhir

$Iter_{\max}$  = jumlah iterasi maksimum

$Iter$  = jumlah iterasi yang ada

Konsep *inertia weight* ( $w$ ) dikembangkan oleh Shi dan Eberhart pada tahun 1998 menjadi suatu parameter kontrol yang baik dari lingkup penelusuran yang dimotivasi keinginan untuk mengurangi  $V_{\max}$ .

### 2.2.4 Constriction Factor

Parameter lain yang dikenal didalam algoritma PSO adalah *constriction factor*, parameter ini diperkenalkan oleh Clerc dengan tujuan dapat menjamin suatu penelusuran dalam algoritma PSO untuk konvergen lebih cepat.

*Coefisient constriction* yang paling sederhana biasa disebut oleh Clerc sebagai type 1 yang memerlukan aplikasi dari koefisien untuk kedua bentuk persamaan *velocity*. Pada umumnya peneliti menerapkan metode *constriction* dengan menetapkan nilai  $\phi = 4.1$  maka nilai-nilai untuk  $c_1$  dan  $c_2$  diset sama dengan 2.05 sehingga nilai  $C = 0.729$ . Dengan metode *constriction* ini mengakibatkan konvergensi dapat terjadi dari waktu ke waktu dimana lintasan osilasi amplitudo semakin berkurang .

### 2.2.6 Modified Improved Particle Swarm Optimization (MIPSO)

Persamaan (2)-(4) adalah persamaan dasar algoritma PSO yang dimodifikasi dengan menggunakan *Inertia Weight Approach* (IWA). *Inertia weight* diperkenalkan untuk menyeimbangkan kemampuan antara penelusuran *global* dan lokal . Clerc memperkenalkan parameter lain yang disebut dengan *Constriction Factor Approach* (CFA) yang digunakan untuk memodifikasi algoritma IPSO yang ada yang disebut dengan *Modified Improved Particle Swarm Optimization* (MIPSO). Parameter ini dapat memberikan performansi yang lebih baik pada algoritma MIPSO. Pada umumnya peneliti menerapkan *constriction factor* pada algoritma PSO dengan mengeset nilai  $c_1$  dan  $c_2 = 2.05$  sehingga diperoleh nilai  $C = 0.729$ . Secara aljabar nilai ini equivalen dengan menggunakan *inertia weight* ketika  $w=0.729$  dan  $c_1 = c_2 = 1.49445$ , semakin besar nilai  $\phi$  maka  $C$  menjadi semakin kecil dan efek peredaman menjadi lebih baik. Karakteristik konvergensi dan perilaku pada sistem dapat dikontrol oleh  $\phi$  .

Berbeda dengan metode *evolutionary* komputasi (EC) yang lain, CFA pada algoritma MIPSO dipastikan konvergen pada penelusuran yang didasarkan pada teori matematika. Algoritma MIPSO dengan *Constriction Factor Approach* (CFA) dapat menghasilkan solusi yang baik dibandingkan dengan algoritma IPSO yang menggunakan *Inertia Weight Approach* (IWA), meskipun CFA hanya memperhitungkan perilaku dinamik dari particle atau agent dan pengaruh dari interaksi antara particle-particle, dimana persamaan-persamaan itu telah dikembangkan dengan posisi terbaik dengan  $P_{best}$  dan  $G_{best}$ , meskipun dapat berubah selama penelusuran di dalam persamaan dasar PSO

## 3. METODE PENELITIAN

### 3.1 Pengumpulan Data

Data yang digunakan dalam penelitian ini terdiri dari dua contoh, pertama diambil dalam *Exp. 3.7 page 82 in "Power Generation, Operation and Control" by Allen J Wood* dan yang kedua diperoleh dari PT. PLN (Persero) AP2B Sulawesi Selatan. Adapun data-data tersebut adalah data kapasitas pembangkit termal sistem Sulsel, data input-output pembangkit termal, data pembebanan maksimum dan minimum pembangkit.

**Tabel 1** Data contoh pertama

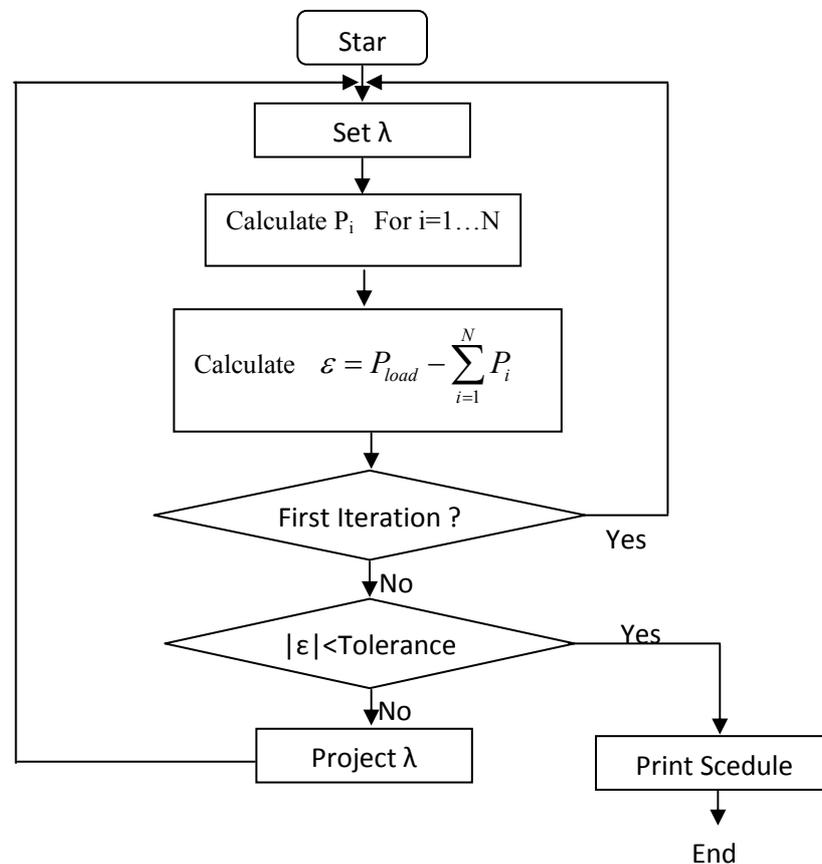
Unit	Unit data	Min (MW)	Max (MW)	Fuel Cost (\$/MBtu)
1	$225+8.4P_1+0.0025P_1^2$	45	350	0.8
2	$729+6.3P_2+0.0081P_2^2$	45	350	1.02
3	$400+7.5P_3+0.0025P_3^2$	47.5	350	0.9

**Tabel 2** Data contoh kedua

Unit	Unit data	Min (MW)	Max (MW)	FuelCost (R/liter)
1	$4547.8075 + 62.8734 P_1 + 3.8639 P_1^2$	8	30	7431.8
2	$4450.825 + 53.1832 P_2 + 4.181 P_2^2$	8	30	7431.8
3	$345.92 + 82.923 P_3 + 10.555 P_3^2$	6	9	5441.7
4	$780.7825 + 39.872 P_4 + 0.5906 P_4^2$	6	8	5441.7
5	$322.5 + 156.65 P_5 + 8.625 P_5^2$	6	8	5441.7

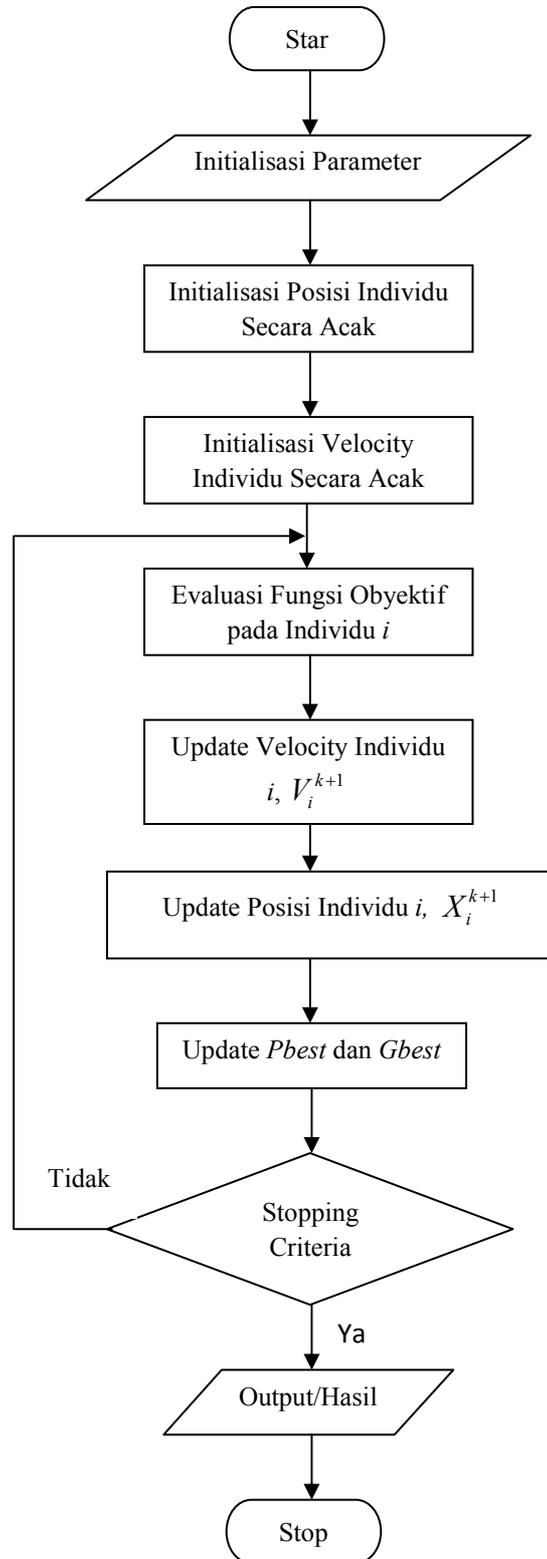
### 3.2 Perancangan Sistem

#### 3.2.1 Flow chart dengan metode iterasi lambda.



**Gambar 4** Flow chart dengan metode iterasi lambda pada economic dispatch.

### 3.2.2 Flowchart Algoritma MIPSO yang diusulkan



**Gambar 5** Flowchart Algoritma MIPSO yang diusulkan

#### 4. HASIL DAN PEMBAHASAN

Simulasi menggunakan perangkat lunak MATLAB dipeoleh hasil :

**Tabel 3** (a) Hasil dari contoh pertama dengan PSO, (b) hasil dari contoh pertama dengan Iterasi Lambda

No	Generator	PSO
	Load (MW)	450
1	Unit 1	206.1473
2	Unit 2	67.2187
3	Unit 3	176.6341
	Total Load (MW)	450
	Total cost (Rp. / Hour)	4485.6

(a)

No	Generator	Iterasi lambda
	Load (MW)	450
1	Unit 1	206.0
2	Unit 2	67.6
3	Unit 3	176.4
	Total Load (MW)	450
	Total cost (Rp. / Hour)	4485.64

(b)

**Tabel 4** (a) Hasil dari contoh kedua dengan PSO, (b) hasil dari contoh kedua dengan Iterasi Lambda

No	Generation	PSO
	Load (MW)	82
1	Unit 1	29.2045
2	Unit 2	28.1483
3	Unit 3	9
4	Unit 4	8
5	Unit 5	7.6472
	Total load (MW)	82
	Total cost (Rp. / Hour)	<b>174.857.063,47</b>

(a) (a)

No	Generation	Iterasi Lambda
	Load (MW)	82
1	Unit 1	29.2
2	Unit 2	28.1
3	Unit 3	9
4	Unit 4	8
5	Unit 5	7.6
	Total load (MW)	82
	Total cost (Rp. / Hour)	<b>174.857.931,11</b>

(b)

#### 5. KESIMPULAN

1. Dalam tulisan ini masalah Optimal Generator Scheduling dapat diselesaikan dengan menggunakan MIPSO
2. Perhitungan telah diaplikasikan dalam dua sistem dan dua metode. Hasil simulasi yang dilakukan pada contoh pertama dan kedua menggunakan metode *Modified Improved Particle Swarm Optimization* (MIPSO) memberikan performansi yang lebih baik dibandingkan dengan metode iterasi lambda. Total biaya bahan bakar yang diperoleh pada contoh pertama metode MIPSO sebesar **Rp. 4485.6** per jam sedangkan dengan metode iterasi lambda sebesar **Rp. 4485.64** per jam, untuk contoh kedua dengan

metode MIPSO sebesar **Rp. 174.857.063,47** per jam sedangkan dengan metode iterasi lambda sebesar **Rp. 174.857.931,11** per jam.

## 7. DAFTAR PUSTAKA

- Allen J, Wollenberg Bruce F, (1996), "*Power Generation, Operational, and Control*", Second Edition, Jhon Wiley & Sons, Inc
- Kundur Praba (1993) "*Power Sistem Stability And Control*" McGraw-Hill
- Shahidehpour Mohammad, Wang Yaoyu, (2003) "*Communication And Control In Electric Power Sistem*" Jhon Wiley & Sons, Inc
- Saadat Hadi (2004) "*Power Sistem Analysis*" International Edition, McGraw-Hill
- James Kennedy and Russell C. Eberhart, with Yuhui Shi (2001) "*Swarm Intelligence*" Morgan Kaufmann Publishers
- Clerc Maurice (2006) "*Particle Swarm Optimization*" ISTE, Ltd
- Pablo E. Oñate Yumbra, Juan M. Ramirez and Carlos A. Coello Coello (Feb 2008) "*Optimal Power Flow Subject to Security Constraints Solved With a Particle Swarm Optimizer*" IEEE Transactions on Power Systems, vol. 23, no. 1
- Andi Syarifuddin, 2008, "*Economic Dispatch For Thermal Generator In Sulawesi Selatan System Using Modified Improved Particle Swarm Optimization*" ITS Surabaya