

Optimisasi *Economic Dispatch* pada Sistem Kelistrikan Jawa Bali 500 kV menggunakan *Differential Evolutionary Algorithm*

Anesya Violita, Ardyono Priyadi, dan Imam Robandi

Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111

E-mail: priyadi@ee.its.ac.id

Abstrak— Optimisasi besarnya pembangkitan untuk dapat memenuhi kebutuhan beban dengan biaya seminimal mungkin merupakan salah satu masalah tersendiri dalam suatu operasi sistem tenaga listrik. Permasalahan ini sendiri lebih dikenal dengan istilah *Economic Dispatch* (ED). Optimisasi ED ini sendiri sudah banyak dilakukan dengan berbagai macam metode *Artificial Intelligence* (AI). Untuk tugas akhir ini, metode AI yang dicoba untuk diaplikasikan pada optimisasi ED ini yakni *Differential Evolutionary* (DE) *Algorithm*. DE *Algorithm* ini akan dicoba diaplikasikan pada sistem kelistrikan Jawa-Bali 500 kV kemudian hasilnya dibandingkan dengan metode lainnya yakni *Lagrange* dan PSO. Hasilnya metode DE *Algorithm* terbukti mampu menemukan solusi optimal dari permasalahan ED dengan penghematan biaya sebesar Rp. 104,76 juta/jam atau 1,545 % dibandingkan dengan metode PSO, dan penghematan biaya pembangkitan sebesar Rp. 1.167,72 juta/jam atau 14,892 % dibandingkan metode *Lagrange*. Sebagai referensi, metode DE *Algorithm* ini juga akan disimulasikan pada sistem tenaga listrik IEEE-30 bus yang hasilnya juga akan dibandingkan dengan hasil yang didapatkan apabila menggunakan metode *Lagrange* dan PSO. Hasil yang didapat juga mampu membuktikan bahwa metode *Differential Evolutionary* (DE) *Algorithm* juga mampu menemukan solusi lebih optimal dengan penghematan biaya sebesar 0,06 \$/jam atau sekitar 0,008 % dibandingkan metode PSO, dan penghematan biaya sebesar 23,47 \$/jam atau 2,92 % dibandingkan dengan metode *Lagrange*.

Kata Kunci— *Differential Evolutionary* (DE) *Algorithm*, *Economic Dispatch*, *Lagrange*, PSO, Sistem Kelistrikan Jawa Bali 500 kV, Sistem Tenaga Listrik IEEE-30 Bus

I. PENDAHULUAN

SISTEM interkoneksi merupakan sistem pembangkitan tenaga listrik modern yang sangat populer saat ini. Pada sistem ini, generator yang dioperasikan letaknya berjauhan satu sama lainnya namun tetap terhubung oleh suatu jaring transmisi. Sistem ini juga tergantung pada kontrol untuk memanfaatkan sumber daya yang ada secara optimal. Pengoptimalan sumber daya disini yakni pengoptimalan penggunaan bahan bakar mengingat komponen biaya operasi terbesar adalah biaya bahan bakar. Jadi efisiensi pemakaian bahan bakar sangat mempengaruhi besarnya penghematan biaya operasi. Kampanye hemat energi oleh karena krisis energi global juga mendorong terus berkembangnya cara-cara optimisasi bahan bakar ini. Permasalahan ini sendiri lebih dikenal dengan istilah *Economic Dispatch* (ED). Dalam ED

ditentukan pembagian beban yang optimal diantara unit-unit pembangkit yang beroperasi setiap saat terjadi perubahan beban sehingga diperoleh total biaya operasi yang minimum dengan tetap memperhatikan batas-batas teknis dan operasional yaitu pembangkitan minimum dan maksimum setiap unit generator dan permintaan beban serta rugi-rugi transmisi [1].

Banyak metode yang sudah dikembangkan dalam optimisasi ED ini namun pada Tugas Akhir ini metode yang ditawarkan adalah *Differential Evolutionary* (DE) *Algorithm*. Parameter kontrol yang sedikit dan konvergensi yang tinggi dan dapat diandalkan [2] merupakan dasar pemilihan DE *Algorithm* sebagai AI yang akan diaplikasikan untuk menyelesaikan permasalahan ED ini.

II. DASAR TEORI

A. *Economic Dispatch*

Economic Dispatch adalah salah satu permasalahan sistem tenaga dimana analisis aliran daya optimal dilakukan untuk meminimalkan biaya pembangkitan. Jadi dalam *Economic Dispatch* pembebanan pada unit-unit pembangkit yang ada dalam sistem dibagi secara optimal ekonomi pada harga beban sistem tertentu sehingga biaya operasi dapat ditekan seminimal mungkin namun tetap dapat memenuhi permintaan beban dan memperhatikan batasan-batasan teknis dan operasional dari masing-masing pembangkit.

Biaya bahan bakar adalah biaya pembangkitan yang utama. Fungsi biaya bahan bakar yang sederhana pada masing-masing unit pembangkit dapat direpresentasikan dalam bentuk fungsi kuadrat yang diberikan dibawah ini :

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \quad (1)$$

i = masing-masing unit pembangkit,
 a_i, b_i dan c_i = koefisien biaya bahan bakar dari unit i ,
 P_i = daya yang ditentukan untuk unit i

Jadi total biaya pembangkitan untuk semua unit pembangkit dapat direpresentasikan sebagai berikut:

$$F_t = \sum_{i=1}^n F_i(P_i) \tag{2}$$

F_t = total biaya bahan bakar,
 F_i = biaya bahan bakar unit i ,
 P_i = pembangkitan dari unit i ,
 n = jumlah unit pembangkit

Optimisasi ED bertujuan untuk meminimalkan total biaya bahan bakar (F_t) ini namun dengan memperhatikan dua batas yakni :

1. Jumlah semua daya yang dibangkitkan harus samadengan total permintaan beban ditambah total rugi-rugi transmisi, sesuai persamaan :

$$\sum_{i=1}^N P_i = P_d + P_l \tag{3}$$

P_i = pembangkitan dari unit i ,
 P_d = total permintaan beban,
 P_l = rugi-rugi transmisi,
 N = jumlah unit pembangkit

2. Batas operasional untuk unit i diberikan oleh pertidaksamaan:

$$P_{i_{min}} \leq P_i \leq P_{i_{max}} \tag{4}$$

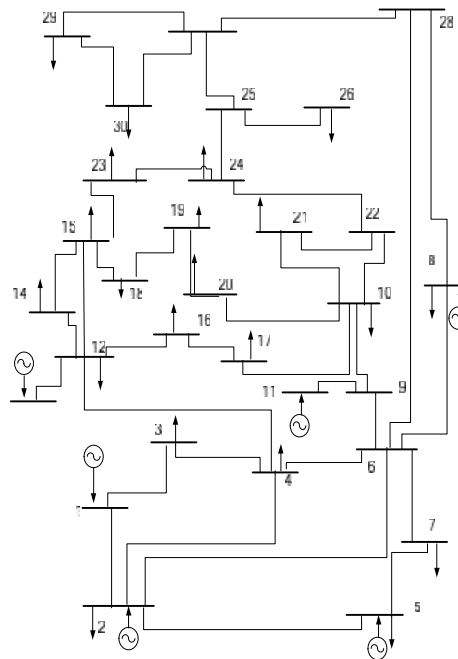
P_i = pembangkitan dari unit i
 $P_{i_{min}}$ = pembangkitan minimum dari unit i
 $P_{i_{max}}$ = pembangkitan maksimum dari unit i

B. Permodelan Sistem

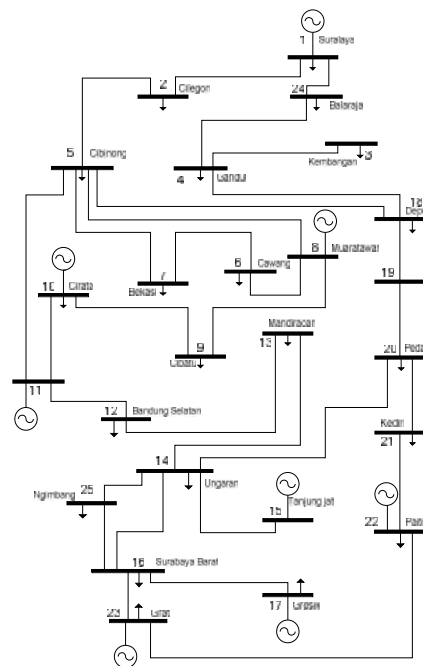
Terdapat dua sistem tenaga listrik yang digunakan pada Tugas Akhir ini yakni sistem tenaga listrik Jawa-Bali 500 kV dan sistem tenaga listrik IEEE-30 bus sebagai pembanding.

Sistem tenaga listrik IEEE-30 bus terdiri dari 6 buah generator dengan rugi-rugi transmisi diasumsikan konstan sebesar 10,07 MW. Gambar 1 menunjukkan *single line diagram* dari sistem.

Untuk sistem tenaga listrik Jawa-Bali 500 kV, terdapat 25 buah bus dengan 1 buah *slack bus*, 7 buah generator bus, dan 17 buah load bus. 1 buah *slack bus* yakni bus Suralaya. Bus generator terdiri dari bus Muara Tawar, Cirata, Saguling, Tanjung Jati, Gresik, Paiton dan Grati. Sedangkan untuk bus beban sendiri yakni bus Cilegon, Kembangan, Gandul, Cibinong, Cawang, Bekasi, Cibatu, Bandung Selatan, Mandiracan, Ungaran, Surabaya Barat, Depok, Tasikmalaya, Pedan, Kediri, Ngimbang, dan Balaraja. Gambar *single line diagram* dari sistem ditunjukkan pada Gambar 2. Simulasi pada sistem ini dilakukan dengan menggunakan data beban dan pembangkitan PT PLN (Persero) P3B Jawa Bali pada tanggal 19 April 2011. Total beban sistem pada saat pukul 18.30 WIB adalah sebesar 10912.52 MW.



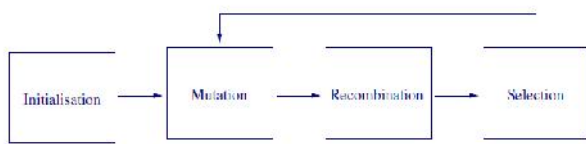
Gambar. 1. *Single Line Diagram* sistem tenaga listrik IEEE 30 Bus



Gambar. 2. *Single Line Diagram* sistem tenaga listrik Jawa Bali 500kV

C. Differential Evolution (DE) Algorithm

Differential Evolution Algorithm adalah salah satu tipe teknik optimasi modern yang berbasis mekanisme pencarian populasi seperti GA, optimasi bee colony, dan PSO [2]. Algoritma yang diperkenalkan oleh Storn dan Price pada tahun 1995 ini mempunyai kelebihan dibandingkan teknik optimasi lainnya. Beberapa diantaranya adalah strukturnya yang sederhana, parameter kontrolnya yang sedikit dan konvergensi keandalannya yang tinggi.



Gambar. 3. Tahap-tahap dalam DE Algorithm

Operasi DE Algorithm untuk optimisasi *Economic Dispatch* pada *plant* dapat dijelaskan secara rinci dibawah ini:

1. Pembacaan Data

Langkah pertama yang dilakukan adalah pengumpulan dan pembacaan data sistem yang akan digunakan dalam pengujian yakni data bus, *line data*, fungsi biaya dan batasan pembangkitan daya.

2. Inisialisasi [3]

Pada tugas akhir ini digunakan jumlah maksimum iterasi sebanyak 100. Jadi pada setiap iterasi, dibangkitkan sebuah populasi vektor awal yang berisi 30 buah vektor (sesuai jumlah populasi yang dipilih). Karena terdapat 8 buah pembangkit, maka dimensi yang digunakan berjumlah 8 buah. Artinya tiap vektor awal nantinya akan memiliki 8 buah kombinasi nilai pembangkitan yang akan diuji tingkat *fitness*-nya. Namun nilai pembangkitan yang dirandom ini harus tetap memperhatikan batas maksimum/minimum pembangkitan generator tersebut dan memenuhi permintaan beban. Oleh sebab itulah dilakukan analisis aliran daya dengan menggunakan *Newton-Rhapson* untuk mendapatkan nilai rugi-rugi transmisi sehingga bisa didapatkan batas nilai pembangkitan yang masih memenuhi nilai permintaan beban

3. Perhitungan Nilai *Fitness* Awal

Objective function dari program DE ini adalah bagaimana mendapatkan harga pembangkitan semimumimum-minimumnya. Oleh sebab itu perhitungan nilai *fitness* dari vektor awal dilakukan dengan memasukkan nilai daya yang telah dirandom pada tahap inisialisasi ke dalam fungsi harga pembangkitan sebagai berikut.

$$F_i(P_i) = a_i + b_i P_i + c_i P_i^2 \tag{5}$$

Dengan

F_i = Besar biaya pembangkitan pada pembangkit ke-i (Rp)

P_i = Daya output dari pembangkit ke-i (MW)

4. Mutasi [3]

Pada tahap ini akan dibentuk populasi yang berisi *mutant vector* ($v_{i,g}$). Pembentukan ini dilakukan dengan mengkombinasi perbedaan vektor (*differential vector*) dari vektor satu ($x_{r1,g}$) dan vektor dua ($x_{r2,g}$) yang dipilih secara acak pada vektor ketiga (x_{r0}). Pengkombinasian tiga perbedaan vektor ini merupakan strategi mutasi. DE

memiliki beberapa macam strategi mutasi. Untuk Tugas Akhir ini dipilih strategi mutasi 5 dengan persamaan sebagai berikut :

$$v_{i,g} = x_{r0} + F \times (x_{r1,g} - x_{r2,g}) \tag{6}$$

F adalah nilai *real* dengan *range* [0,1]. Nilai F inilah yang akan mengontrol pengaruh dari variasi *differential* dari ($x_{r1,g} - x_{r2,g}$). Pada Tugas Akhir ini dipilih nilai F sebesar 0,7.

5. Rekombinasi [3]

Di tahap rekombinasi akan dibentuk sebuah populasi berisi *trial vector* yang merupakan hasil rekombinasi antara populasi vektor awal dengan populasi vektor mutan.

Operasi dari tahap rekombinasi ini sendiri dengan menggunakan nilai probabilitas *crossover* atau C_r dengan *range* [0,1] yang telah ditentukan oleh user. Hal ini berguna untuk mengontrol pembagian nilai parameter yang digandakan dari *mutant*. Pada tugas akhir ini dipilih nilai probabilitas *crossover* sebesar 0,8. Terdapat $rand_j(0,1)$ yang merupakan nilai random yang menentukan apakah vektor tersebut di-*crossover* atau tidak. Jika nilai $rand_j(0,1)$ suatu vektor tersebut kurang dari nilai probabilitas *crossover* yakni 0,8 maka nilai vektor yang akan digandakan pada *trial vector* adalah *mutant vector*. Jika yang terjadi adalah sebaliknya, maka nilai vektor yang akan digandakan pada *trial vector* adalah vektor awal.

6. Perhitungan Nilai *Fitness* Akhir

Dilakukan perhitungan nilai *fitness* dari vektor *trial* yang sudah terbentuk tadi dengan memasukkannya ke dalam fungsi harga pembangkitan yang sudah dijabarkan pada persamaan 5 sebelumnya.

7. Seleksi [3]

Pada tahap ini dilakukan seleksi untuk menentukan vektor yang akan menjadi anggota populasi untuk iterasi selanjutnya. Jika *trial vector*, $u_{i,g}$, memiliki nilai *objective function* atau *fitness* sama atau lebih kecil daripada *fitness* vektor awal, maka ia mengganti *target vector* pada pembangkitan atau iterasi selanjutnya. Sebaliknya, Jika *trial vector*, $u_{i,g}$, memiliki nilai *objective function* atau *fitness* yang lebih besar daripada *fitness* vektor awalnya, maka vektor awal tadi tetap menjadi anggota pada pembangkitan atau iterasi selanjutnya. Selanjutnya setelah populasi baru diperoleh, proses mutasi, rekombinasi, dan seleksi diulangi sampai nilai optimum diperoleh atau kriteria yang diinginkan tercapai atau mencapai iterasi maksimum yakni 100 kali.

III. HASIL DAN DISKUSI

A. Simulasi ED pada Sistem Tenaga Listrik IEEE-30 Bus

Dari 10 kali *running* program yang dilakukan dengan setiap *running*-an terdiri dari 100 kali iterasi didapatkan hasil yang paling baik dari metode *Lagrange*, PSO maupun *DE Algorithm* seperti ditunjukkan pada Tabel 1.

Dari Tabel 1 dapat dilihat bahwa dengan pembebanan yang sama yakni sebesar 283,4 MW, metode *DE Algorithm* menghasilkan biaya pembangkitan sebesar 779,77 \$/jam dengan daya terbangkitkan sebesar 286,97 MW. Sedikit lebih hemat 0,06 \$ /jam nya atau sekitar 0,008 % bila dibandingkan dengan metode PSO yang menghasilkan biaya pembangkitan sebesar 779,83 \$/jam dengan total pembangkitan 286,43 MW dan rugi-rugi daya sebesar 3,03 MW.

Metode *Lagrange* menghasilkan biaya pembangkitan paling besar yakni 803,24 \$/jam dengan daya terbangkitkan sebesar 293,47 MW dan rugi-rugi sebesar 10,07 MW. Dengan ini jika dibandingkan dengan metode *Lagrange*, *DE Algorithm* mampu menghemat biaya pembangkitan lebih besar sekitar sebesar 23,47 \$/ jam nya atau lebih hemat sekitar 2,92%.

Tabel 1. Perbandingan hasil simulasi ED pada sistem tenaga listrik 30 bus menggunakan metode *Lagrange*, PSO dan *DE Algorithm*

No	Pembangkit	<i>Lagrange</i>	PSO	DE
		Daya Aktif (MW)	Daya Aktif (MW)	Daya Aktif (MW)
1	P1	178,16	183,18	186,480
2	P2	48,171	44,66	47,835
3	P3	21,21	18,92	19,316
4	P4	21,99	11,68	11,328
5	P5	11,93	11,00	10,010
6	P6	12,00	16,97	12,000
Total Daya		293,47	286,43	286,97
Rugi-rugi Daya (MW)		10,07	3,03	3,57
Total Biaya		803,24	779,83	779,77

Tabel 2 Perbandingan hasil simulasi ED pada sistem kelistrikan Jawa Bali 500 kV menggunakan metode *Lagrange*, PSO dan *DE Algorithm*

No	Pembangkit	<i>Lagrange</i>	PSO	DE
		Daya Aktif (MW)	Daya Aktif (MW)	Daya Aktif (MW)
1	Suralaya	3.287	3.176,78	2.652,805
2	Muaratawar	2.115	2.095,81	2.114,000
3	Cirata	1.000	1.000,00	1.000,000
4	Saguling	698	698,00	698,000
5	Tanjung Jati	1.321	1.216,50	1.320,920
6	Gresik	1.050	322,11	238,000
7	Paiton	3.240	2.360,26	2.868,154
8	Grati	150	163,50	150,001
Total Daya		12.861	11.032,96	11.042,88
Rugi-rugi daya (MW)		1.948,48	120,44	130,36
Total biaya*		1.948,48	6.778,55	6.673,79

B. Simulasi ED pada Sistem Kelistrikan Jawa Bali 500 kV

Dilakukan 10 kali *running* program dengan masing-masing 100 kali iterasi setiap *running*-nya. Perbandingan hasil simulasi ditunjukkan pada Tabel 2. Dapat diketahui bahwa dengan pembebanan yang sama, yaitu sebesar 10.912.52 MW,

DE Algorithm mampu menyelesaikan permasalahan ED dengan biaya pembangkitan sebesar Rp. 6.673,79 juta/jam dan daya terbangkitkan sebesar 11.042,88 MW. Metode PSO menghasilkan biaya pembangkitan sebesar Rp. 6.778,55 juta/jam, dengan daya terbangkitkan sebesar 11.032,96 MW. Sedangkan metode *Lagrange* menghasilkan biaya pembangkitan sebesar Rp. 7.841,51 juta/jam, dengan daya terbangkitkan sebesar 12.861 MW.

Dari data hasil simulasi dapat dianalisis bahwa optimisasi ED dengan *DE Algorithm* menghasilkan solusi yang lebih optimal dibandingkan dengan metode PSO dan *Lagrange*. *DE Algorithm* mampu menghemat biaya pembangkitan sebesar Rp. 104,76 juta/jam atau 1,545% dibandingkan dengan metode PSO. Sedangkan jika dibandingkan dengan metode *Lagrange*, *DE Algorithm* mampu menghemat biaya pembangkitan sebesar 14,892% atau Rp. 1.167,72 juta/jam.

Dapat dilihat juga bahwa 2 PLTA yakni PLTA Cirata dan PLTA Saguling *disetting* membangkitkan daya maksimal karena keduanya merupakan pembangkit listrik paling murah.

IV. KESIMPULAN/RINGKASAN

Pada sistem tenaga listrik 30 bus, metode *Differential Evolutionary (DE) Algorithm* mampu menemukan solusi optimal dari permasalahan ED dengan penghematan biaya sebesar 0,06 \$/jam atau sekitar 0,008 % dibandingkan metode PSO, dan penghematan biaya sebesar 23,47 \$/jam atau 2,92 % dibandingkan dengan metode *Lagrange*.

Hasil yang tak jauh berbeda didapatkan juga dari simulasi ED pada sistem kelistrikan Jawa Bali 500 kV dimana metode *DE Algorithm* mampu menemukan solusi optimal dari permasalahan ED dengan penghematan biaya sebesar Rp. 104,76 juta/jam atau 1,545 % dibandingkan dengan metode PSO, dan penghematan biaya pembangkitan sebesar Rp. 1.167,72 juta/jam atau 14,892 % dibandingkan metode *Lagrange*.

Dari kedua hasil diatas maka dapat diambil kesimpulan bahwa metode *DE Algorithm* menghasilkan nilai yang lebih optimal dibandingkan dengan metode PSO dan *Lagrange*.

DAFTAR PUSTAKA

- [1] Wood, A.J. dan Wollenberg, B.F., (1996), "Power Generation Operation and Control, 2nd edition, John Wiley & Sons. Inc., New York
- [2] S. Khamsawang and S. Jiriwibhakorn. "Solving the Economic Dispatch Problem by Using Differential Evolutionary" presented at International Journal of Electrical Power and Energy Systems Engineering 2:2,2009.
- [3] R. Storn, K. Price, "Differential Evolution : A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces", Journal of Global Optimization 11:341±359, 1997G.