

Koordinasi Optimal *Capacitive Energy Storage* (CES) dan Kontroler PID Menggunakan *Differential Evolution Algorithm* (DEA) pada Sistem Tenaga Listrik

Akbar Swandaru, Imam Robandi, Ardyono Priyadi

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

Jl. Arief Rahman Hakim, Surabaya 60111

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

Abstrak— Peningkatan suplai daya listrik diperlukan untuk memenuhi kebutuhan daya listrik. Generator cenderung beroperasi dalam beban penuh. Hal ini berpengaruh pada keamanan generator dalam operasi sistem tenaga listrik. Salah satu masalah adalah osilasi frekuensi. Bila perubahan beban terjadi, kontroler diperlukan untuk meredam osilasi frekuensi ini. Pada tugas akhir ini diusulkan sebuah koordinasi antara Kontroler *Capacitive Energy Storage* (CES) dan Kontroler PID. CES disini berfungsi untuk membantu kinerja Governor agar meredam osilasi frekuensi dengan cepat. Kontroler CES ini digunakan bersama dengan PID controller yang dioptimalkan dengan *Differential Evolution Algorithm* (DEA).

Kata Kunci—CES, Kontroler PID, SMIB, *Differential Evolution Algorithm* (DEA)

I. PENDAHULUAN

PERUBAHAN beban yang tidak dapat diprediksi dapat menyebabkan kestabilan sistem terganggu. Kestabilan sistem tenaga listrik umumnya terganggu oleh gangguan yang bersifat transien dan bersifat dinamik. Gangguan transien disebabkan oleh gangguan besar seperti putus jaringan dan gangguan dinamik disebabkan oleh perubahan kecil seperti perubahan beban. Adapun perubahan beban secara tiba-tiba dapat mengganggu stabilitas sistem. Karena perubahan beban mengakibatkan perubahan nilai daya yang dihasilkan pada pembangkitan, yang mana menyebabkan pergeseran nilai frekuensi pada sistem.

Sehingga pada pemecahan masalah ini, mustahil hanya mengandalkan governor sebagai kontrol kecepatan, namun harus dibantu oleh kontroler tambahan. Fungsi dari kontrol tambahan ini, beserta nilai parameter-parameter kontrol tambahan yang telah ditetapkan, adalah untuk mengambil nilai pergeseran frekuensi tersebut menjadi nilai referensi jika terjadi gangguan. Penempatan nilai parameter-parameter kontrol tambahan tersebut menjadi hal yang sangat penting disini, mengingat lama sistem mengalami gangguan untuk kembali

menjadi *steady* dipengaruhi oleh nilai parameter-parameter yang akan ditempatkan kelak. Sehingga diperlukan analisis kestabilan untuk memecahkan permasalahan ini. Dimana analisis kestabilan dinamik dapat digunakan pendekatan linier sedangkan pendekatan model nonlinear digunakan untuk analisis kestabilan transien [1].

Capacitive Energy Storage (CES) merupakan salah satu kontroler yang dapat berfungsi untuk memecahkan masalah ini. *Capacitive Energy Storage* (CES) merupakan peralatan penyimpan energi yang dapat meredam osilasi pada sistem tenaga listrik [2]. Pada penerapan praktisnya, CES tidak memerlukan biaya pengoperasian dan juga tidak memberikan dampak pada lingkungan. Pengoperasian dari CES juga cukup sederhana dan lebih murah jika dibandingkan dengan peralatan penyimpan energi yang lain seperti SMES. SMES memerlukan sistem pendinginan helium cair yang kontinyu selama operasi. Selama pengoperasian, peralatan *magnetic storage system* memerlukan arus yang kontinyu, namun tidak dengan CES.

Pada penelitian ini, DEA digunakan untuk mencari koordinasi optimal PID dan CES untuk meningkatkan performansi sistem pada saat terjadi gangguan. Koordinasi PID dan CES yang telah dioptimisasi diukur dengan menggunakan *Integral Time Absolute Error* (ITAE). Kemudian menganalisis pengaruh penalaan PID dan CES dengan DEA terhadap kestabilan dinamik pada sistem. Hasil simulasi menunjukkan perbandingan respon *plant* terhadap sistem tanpa kontrol, menggunakan CES dan menggunakan CES dan kontroler PID yang dioptimasi dengan menggunakan DEA.

II. PEMODELAN SISTEM

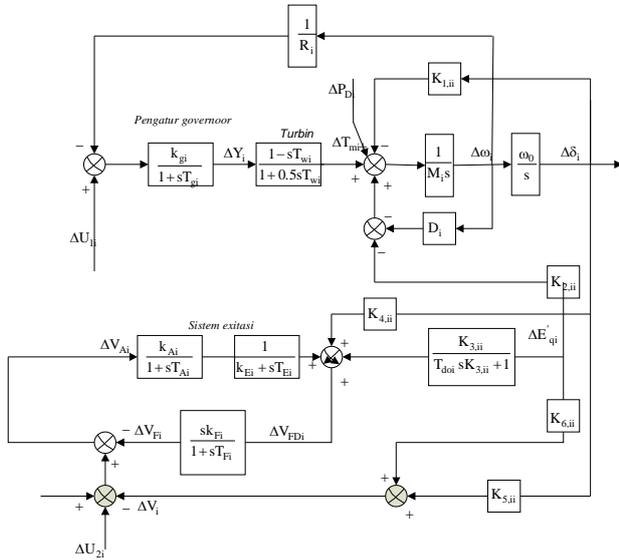
Model sistem tenaga listrik yang digunakan pada riset ini Single Machine Infinite Bus (SMIB) yang dikontrol oleh CES dan kontroler PID. Pemodelan SMIB, CES dan penempatan PID akan dijelaskan dalam sub bab berikut.

A. SMIB

Sistem eksitasi dari sistem tenaga listrik harus mendukung kontrol tegangan efektif dan mempertinggi stabilitas [3], serta harus mampu merespon secara tepat perubahan stabilitas gangguan transien dan stabilitas sinyal kecil.

Sistem eksitasi merupakan bagian yang amat penting.

Exciter berfungsi menghasilkan EMF dari generator tidak hanya mengatur tegangan output tetapi juga mengatur faktor daya dan besaran arus yang sesuai [3]. Pada mesin listrik misal generator, pengaturan arus eksitasi di bantu oleh Automatic Voltage Regulator (AVR).



Gambar 1. SMIB

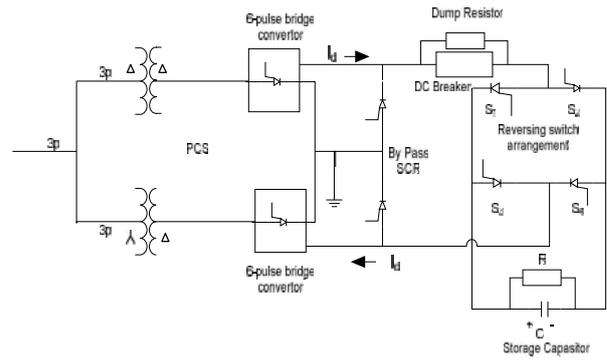
B. Capacitor Energy Storage

CES adalah sebuah devais untuk melakukan penyimpanan dan pelepasan daya dalam jumlah yang besar secara simultan. CES disini berfungsi untuk menyimpan energi dalam bentuk medan listrik pada kapasitor. Sebuah CES terdiri dari storage capacitor dan Power Conversion System (PCS) dengan control dan fungsi pengaman yang terintegrasi.

Dari Gambar 2 dapat dilihat bahwa kapasitor penyimpan dihubungkan pada jalajala AC melalui power conversion system (PCS) yang terdiri dari inverter/rectifier. Kapasitor penyimpan ini terdiri dari beberapa kapasitor diskrit yang dihubungkan secara paralel, dengan kapasitansi lumped 'C'. Rugirugi bocor dan dielektrik kapasitor bank pada CES dimodelkan dengan resistansi R yang dihubungkan secara paralel pada kapasitor [2].

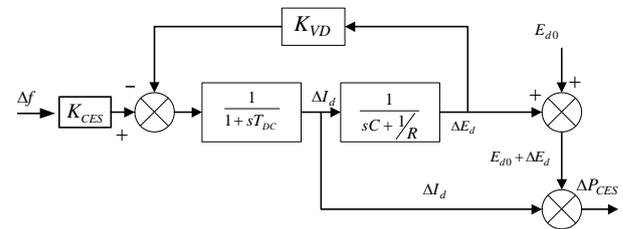
Perubahan arah arus saat charge dan discharge diatasi dengan penyusunan switch secara terbalik menggunakan gate-turn-off thyristors GTO. Pada mode charging, switch S₁ dan S₄ di set on dan switch S₂ dan S₃ di set off. Pada mode discharging, switch S₂ dan S₃ diset on dan S₁ dan S₄ di set off [2].

Proses operasi keluar masuk arus, pada CES dikontrol dengan menerapkan tegangan yang tepat pada kapasitor. Penerapan tegangan tersebut dikontrol dengan mengontrol sudut penyalan pada jembatan converter [2,8].



Gambar 2. Capacitive Energy Storage

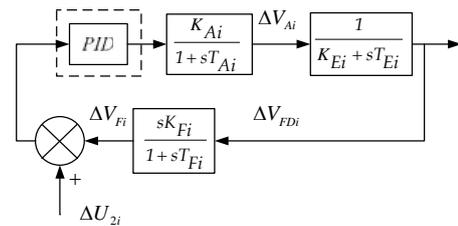
Model linier dari CES ditunjukkan pada Gambar.3.



Gambar 3. Diagram blok CES

C. Penambahan PID Controller pada Sistem Eksitasi

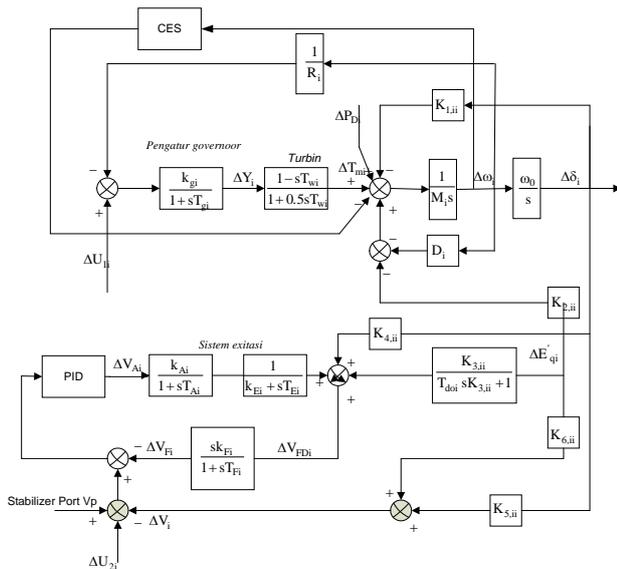
PID ditambahkan pada sistem eksitasi, dan penentuan nilai K_p, K_i, dan K_d menggunakan metode optimisasi DEA. Hal ini dilakukan guna membantu kinerja system eksitasi agar respon system yang dihasilkan lebih halus dan lebih baik [3]. Sehingga diperlukan controller tambahan untuk menjamin kehalusan respon sistem yang didesain. Blok diagram yang dikompensasi oleh PID controller ditunjukkan pada Gambar 4 berikut:



Gambar 4. Sistem eksitasi yang dikompensasi oleh PID controller

D. Penerapan PID Controller & CES pada Sistem

Pemodelan sistem multimesin ditunjukkan pada Gambar 5. Gambar 5 menunjukkan sistem pembangkit Suralaya yang dipasang kontroler PID dan CES.



Gambar 5. Test Sistem yang dipasang kontroler PID dan CES

III. DIFFERENTIAL EVOLUTION ALGORITHM

DEA merupakan metode pencarian berbasis populasi yang menggunakan siklus perulangan dari rekombinasi dan seleksi untuk menggiring populasi menuju nilai optimum global [13]. Berikut ini merupakan tahapan DEA untuk mencari nilai optimum [14]:

Inisialisasi

Evaluasi

Ulangi

Mutasi

Rekombinasi

Evaluasi

Seleksi

Sampai kriteria yang diinginkan terpenuhi

A. Struktur Populasi [14]

Differential Evolution Algorithm (DEA) mencari suatu nilai optimal menggunakan sepasang populasi vector. Masing-masing populasi terdiri dari N_p vector dengan D-dimensional parameter. Populasi awal P_x terdiri dari vector $x_{i,g}$ yang ditentukan sebagai *initial point*. Berikut ini merupakan penjelasan populasi awal DE secara matematis:

$$P_{x,g} = (x_{i,g}), \quad i=1, \dots, N_p, \quad g=1, \dots, g_{max} \quad (1)$$

$$x_{i,g} = (x_{j,i,g}), \quad j = 1, \dots, D.$$

Dengan $x_{i,g}$ merupakan vector ke-i pada pembangkitan ke g. Sedangkan $x_{j,i,g}$ adalah nilai vector ke-i pada parameter ke-j, pada pembangkitan ke g. Nilai i adalah bilangan bulat dari 0 sampai N_p , g adalah bilangan bulat dari 0 sampai g_{max} , dan j adalah bilangan bulat dari 0 sampai D-1. Populasi berikutnya, $P_{v,g}$, merupakan populasi yang mengandung N_p vector-vector DE yang bermutasi secara random $v_{i,g}$. Berikut ini merupakan

penjelasan populasi $P_{v,g}$ secara matematis:

$$P_{v,g} = (v_{i,g}), \quad i=1, \dots, N_p, \quad g=1, \dots, g_{max} \quad (2)$$

$$v_{i,g} = (v_{j,i,g}), \quad j = 1, \dots, D - 1.$$

Kemudian, masing-masing vector pada populasi awal direkombinasi dengan *mutant vector* untuk menghasilkan populasi *trial*, $P_{u,g}$, dengan N_p buah *trial vector*, $u_{i,g}$. Berikut ini merupakan penjelasan populasi $P_{u,g}$ secara matematis:

$$P_{u,g} = (u_{i,g}), \quad i=1, \dots, N_p, \quad g=1, \dots, g_{max} \quad (3)$$

$$u_{i,g} = (u_{j,i,g}), \quad j = 1, \dots, D.$$

Pada proses rekombinasi, populasi mutant digantikan oleh populasi *trial* sehingga diperoleh sepasang populasi yang akan diproses dalam DE yaitu, populasi awal (*current population*) dan populasi *trial*.

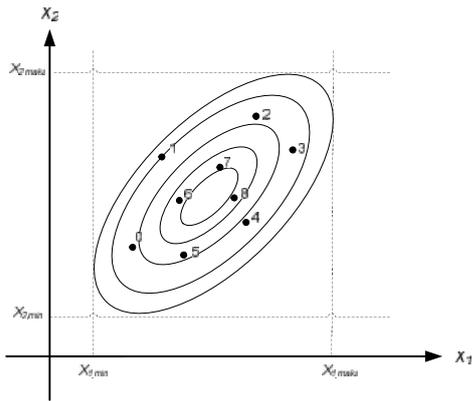
B. Inisialisasi [14]

Inisialisasi nilai awal DEA dilakukan secara random pada suatu daerah tertentu yang memiliki suatu batasan. Batasan tersebut terdiri dari batas atas, $b_{j,U}$, dan batas bawah, $b_{j,L}$, yang ditentukan bagi setiap parameter. Nilai masing-masing vector awal dibangkitkan secara acak di antara batasan-batasan tersebut. Berikut ini merupakan persamaan yang digunakan untuk inisialisasi populasi.

$$x_{j,i,g} = rand_j(0,1) \times (b_{j,U} - b_{j,L}) + b_{j,L} \quad (4)$$

Dengan $x_{j,i,g}$ adalah nilai vector ke-i pada parameter ke-j dan pembangkitan ke g, $b_{j,U}$ adalah batas atas parameter ke-j, $b_{j,L}$ adalah batas bawah parameter ke-j. Angka pembangkitan random, $rand_j(0,1)$, adalah angka yang dibangkitkan secara acak antara *range* [0.1].

Pada Gambar 6 diilustrasikan proses inisialisasi DEA menggunakan dua buah parameter yaitu x_1 dan x_2 . Ruang pencarian DEA dibatasi oleh $(x_{1,min}, x_{1,maks})$ untuk x_1 dan $(x_{2,min}, x_{2,maks})$ untuk x_2 . Pada proses inisialisasi, vektor-vektor pada DE disebar secara acak atau *random* di dalam suatu ruang pencarian. Masing-masing vektor ditempatkan pada suatu titik dengan nilai x_1 dan x_2 yang berbeda-beda bagi setiap vektor. Kemudian, *fitness* masing-masing vektor didapatkan dengan cara memasukkan nilai x_1 dan x_2 pada fungsi $f(x_1, x_2)$. *Fitness* masing-masing vektor direpresentasikan oleh beberapa garis kontur. Garis kontur terluar merepresentasikan daerah dengan nilai *fitness* yang paling buruk sedangkan kontur terdalam merepresentasikan daerah dengan nilai *fitness* yang paling baik.



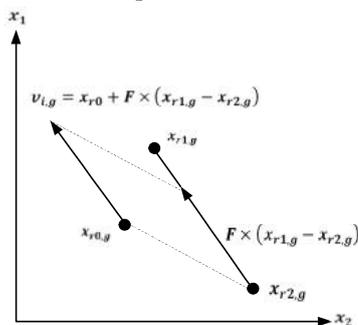
Gambar 6. Inisialisasi populasi DE

C. Mutasi [14]

Setelah inisialisasi, DEA memutasi dan merekombinasi populasi untuk menghasilkan N_p trial vector. Mutasi pada DE, yang disebut dengan differential mutation, ini dilakukan dengan mengkombinasi perbedaan vector (differential vector) dari vector satu dan vector dua yang dipilih secara acak pada vector ketiga untuk menghasilkan mutant vector. Pengkombinasian tiga perbedaan vector ini ditunjukkan oleh persamaan 2.72 :

$$v_{i,g} = x_{r0} + F \times (x_{r1,g} - x_{r2,g}) \quad (5)$$

Dengan $v_{i,g}$ adalah mutant vector, x_{r1} , x_{r2} dan x_{r0} adalah vector yang dipilih secara acak. F adalah bilangan nyata dengan range $[0,1]$ yang mengontrol pengaruh variasi perbedaan (differential) dari $(x_{r1,g} - x_{r2,g})$. Gambar 7 menunjukkan ilustrasi mutasi suatu vector pada contoh 2 dimensional.



Gambar 7. Mutasi differential: bobot differential, $F \times (x_{r1,g} - x_{r2,g})$, ditambahkan pada vector $x_{r0,g}$ untuk menghasilkan mutant vector $v_{i,g}$

D. Crossover [14]

Crossover digunakan DEA untuk melengkapi strategi pencarian mutasi differential (differential mutation). Tujuan crossover pada DE adalah membentuk trial vector dari nilai parameter yang digandakan dari dua vector yang berbeda yaitu, vector awal dengan mutant vector.

$$u_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} & \text{if } (rand_j(0,1) \leq Cr \text{ or } j = j_{rand}) \\ x_{j,i,g} & \text{yang lain} \end{cases} \quad (6)$$

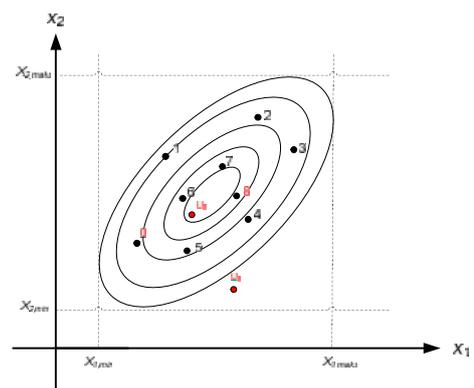
Probabilitas crossover, Cr dengan range $[0,1]$, adalah nilai yang ditentukan oleh pengguna untuk mengontrol pembagian nilai parameter yang digandakan dari mutant. $rand_j(0,1)$ adalah nilai random yang menentukan apakah vector tersebut di-crossover atau tidak. Jika nilai $rand_j(0,1)$ suatu vector adalah kurang dari Cr , maka nilai vector yang akan digandakan pada trial vector adalah mutant vector. Jika yang terjadi adalah sebaliknya, maka nilai vector yang akan digandakan pada trial vector adalah vector awal. Selain itu, trial parameter dengan indeks yang dipilih secara acak, j_{rand} , diperoleh dari mutant untuk memastikan bahwa trial vector tidak menduplikat $x_{i,g}$. Karena kebutuhan tambahan ini, Cr hanya mendekati probabilitas yang tepat, p_{Cr} , yang trial parameter-nya didapatkan dari mutant.

E. Seleksi [14]

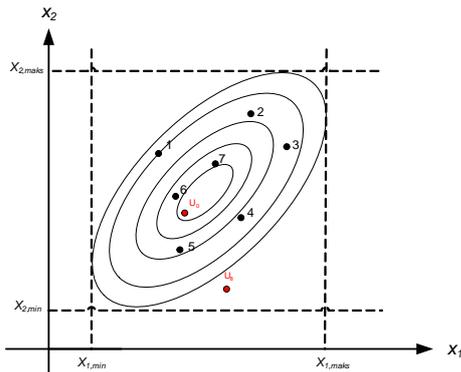
Seleksi dilakukan untuk menentukan vector yang akan menjadi anggota populasi untuk iterasi selanjutnya. Jika trial vector, $u_{i,g}$, memiliki nilai objective function sama atau lebih kecil daripada target vector $x_{i,g}$, maka ia mengganti target vector pada pembangkitan atau iterasi selanjutnya. Sebaliknya, Jika trial vector, $u_{i,g}$, memiliki nilai objective function yang lebih besar daripada target vector $x_{i,g}$, maka target vector tetap menjadi anggota pada pembangkitan atau iterasi selanjutnya. Hal ini dinyatakan dalam persamaan 2.74:

$$x_{i,g} = \begin{cases} u_{i,g} & \text{iff } (u_{i,g} \leq f(x_{i,g})) \\ x_{j,i,g} & \text{yang lain} \end{cases} \quad (7)$$

Proses seleksi pada DEA diilustrasikan pada Gambar 8. dan Gambar 9. Pada Gambar 9 ditunjukkan vector hasil inisialisasi dan dua trial vector U_0 dan U_8 sebagai contoh proses seleksi. U_0 merupakan trial vector yang dikompetisikan dengan vector ke 0 dan U_8 merupakan trial vector yang dikompetisikan dengan vector ke 8.



Gambar 8. Proses seleksi pada DEA (a)



Gambar 9. Proses seleksi pada DEA (b)

Pada Gambar 8, dapat diamati bahwa nilai *fitness trial vector* U_0 lebih baik daripada U_8 sehingga U_8 dihapus dan U_0 ditetapkan untuk mengikuti iterasi berikutnya. Sedangkan pada *trial vector* U_8 dan U_8 , *fitness vector* U_8 lebih baik daripada U_8 sehingga *trial vector* U_8 tidak digunakan dan U_8 berhak memasuki iterasi berikutnya. Hal ini ditunjukkan pada Gambar 8.

Setelah populasi baru diperoleh, proses mutasi, rekombinasi, dan seleksi diulangi sampai nilai optimum diperoleh atau kriteria yang diinginkan tercapai atau mencapai pembangkitan maksimum g_{maks} .

IV. HASIL SIMULASI DAN ANALISIS

Gambar10 menunjukkan flowchart penalaan parameter kontroler PID dan CES dengan menggunakan DEA. Untuk mencari parameter kontroler PID dan CES yang optimal maka digunakan DE. Fungsi objektif yang digunakan untuk menguji kestabilan sistem adalah dengan *Integral Time Absolute Error* (ITAE).

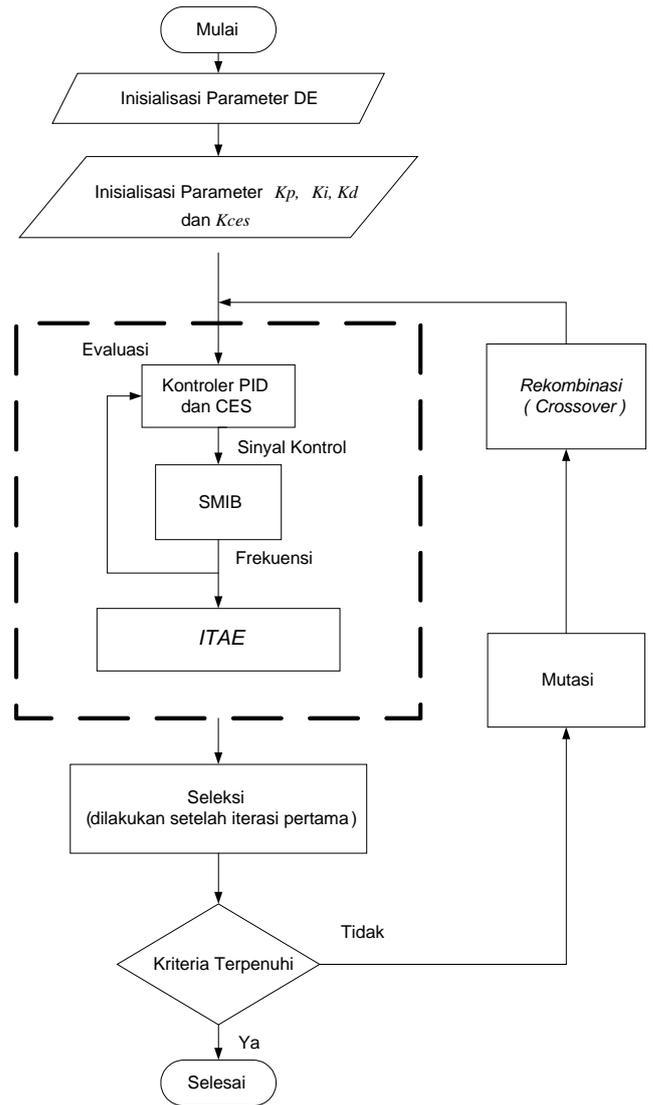
Disini, *Integral Time Absolute Error* (ITAE) dilakukan dengan persamaan.

$$ITAE = \int_0^{\infty} t|\omega(t)|dt \tag{8}$$

Sedangkan data parameter DEA yang digunakan pada Riset ini ditunjukkan pada Tabel 1.

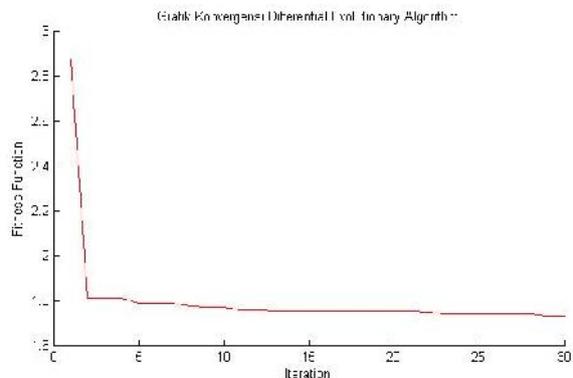
Tabel 1.
Parameter DEA

Strategy	1
Dimensi	4
Crossover	0,8
Jumlah Populasi	30
Pembobot (F)	0,7
Iterasi maksimum	50



Gambar 10. Flowchart penalaan PID & CES menggunakan DEA

Running optimisasi parameter CES dan parameter kontroler PID menggunakan DEA dilakukan sebanyak 30 iterasi. Nilai minimum *fitness function* sistem pada setiap iterasi di-plot pada grafik konvergensi yang ditunjukkan pada Gambar 11.

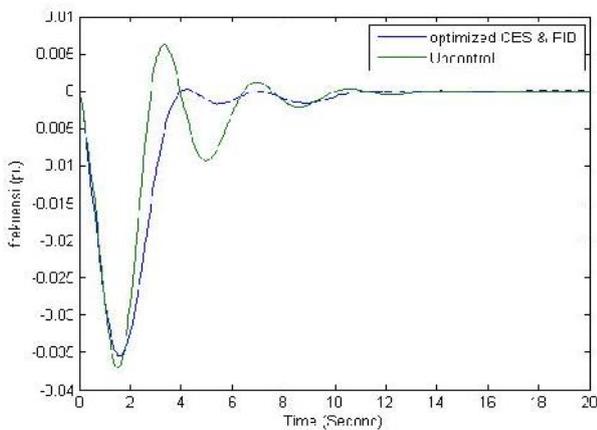


Gambar 11. Grafik konvergensi DEA

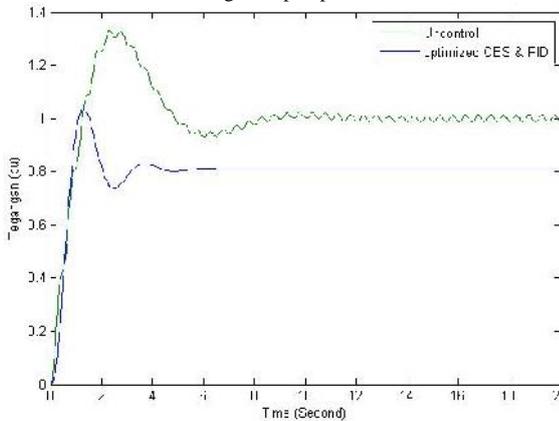
Tabel 2.
Hasil koordinasi parameter PID & CES menggunakan DEA

K_{CES}	K_p	K_i	K_d
0.0861	3.2328	3.5563	3.3947

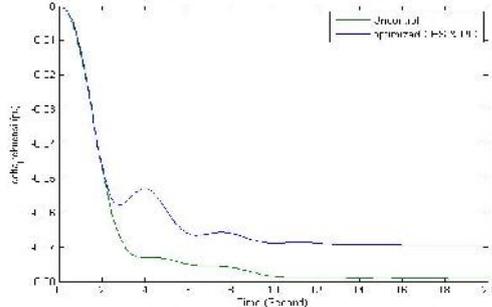
Simulasi sistem menggunakan CES & Kontroler PID yang parameternya dioptimisasi menggunakan DEA dilakukan selama 20 detik. Untuk menguji kestabilan dinamis gangguan perubahan beban diberikan pada sistem sebesar 0.05 pu. Dalam riset ini dilakukan analisa perubahan frekuensi, tegangan, dan delta frekuensi. Hasil simulasi diperoleh hasil respon dalam satuan per unit (pu) yang ditunjukkan pada Gambar 12, 13, &14.



Gambar 12. Perbandingan respon perubahan frekuensi sistem



Gambar 13. Perbandingan respon perubahan tegangan sistem



Gambar 14. Perbandingan respon perubahan delta frekuensi sistem

V. KESIMPULAN DAN SARAN

A. Kesimpulan

Hasil simulasi menunjukkan bahwa penerapan DEA untuk menala parameter CES& PID pada SMIB dapat memperbaiki respon perubahan frekuensi dan respon perubahan sudut rotor. Penggunaan koordinasi CES & PID yang optimal dapat mempercepat settling time respon perubahan frekuensi SMIB ketika terjadi gangguan perubahan beban.

B. Saran

Penerapan koordinasi CES & PID sebagai peredam osilasi sebaiknya juga diuji untuk kondisi gangguan transien. Penalaan koordinasi CES& PID dapat diuji dengan metode optimisasi lain untuk mengamati perbandingan dengan metode DEA.

REFERENSI

- [1] Imam Robandi, "Desain Sistem Tenaga Modern", Penerbit ANDI, Yogyakarta, (2006).
- [2] S.C. Tripathy, I.P. Mishra, "Dynamic Performance Of Wind-Diesel Power System With Capacitive Energy Storage", Energy Convers. Mgm Vol. 37, No. 12, (1996).
- [3] H.Saadat, "Power System Analysis," McGraw-Hill International Edition, (1999).
- [4] Stevenson, William D, "Power System Analysis", McGraw-Hill, Inc, (1994).
- [5] Charles A. Gross, "Power System Analysis", Second Edition, Auburn University, (1986).
- [6] Paul M. Anderson, A.A. Fouad, "Power System Control and stability," IEEE Press Power system engineering series, (1993).
- [7] P.Kundur, "Power System Stability and Control," McGraw-Hill, Inc, (1993).
- [8] S.C. Tripathy, "Improved Load Frequency Control With Capacitive Energy Storage", Energy Cottut-rs. hfgmt Vol. 38, No. 6. pp. S-562. (1997).
- [9] Rajesh Joseph Abraham, D. Das and Amit Patra, "Automatic Generation Control of an Interconnected Power System with Capacitive Energy Storage", International Journal of Electrical Power and Energy System Engineering 3:1 (2010).
- [10] Rajesh Joseph Abraham, D. Das & Amit Patra, "Effect of Capacitive Energy Storage on Automatic Generation Control", The 7th International Power Engineering Conference (IPEC), (2005).
- [11] Muh Budi R Widodo, Muhammad Abdillah, Imam Robandi "Aplikasi Fuzzy PIPD pada Single Machine Infinite Bus (SMIB)", Seminar on intelegent technology and it's Aplication (SITIA). (2009), 90.
- [12] Mochamad Avid Fassamsi, "Optimal Tuning PID Superconducting Magnetic Energy Storage (SMES) Menggunakan Imperialist Competitive Algorithm (ICA) Untuk Meredam Osilasi Daya Pada Sistem Kelistrikan Jawa Bali 500 kV", Tugas Akhir, Teknik Elektro, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia (2010).
- [13] R. Storn, K. Price, J. Lampinen, "Differential Evolution A Practical Approach to Global Optimization" Springer-Verlag, Berlin, Heidelberg, (2005).
- [14] Joel H. Van Sickle, Kwang Y. Lee, Jin S. Heo, "Differential Evolution and its Applications to Power Plant Control", IEEE, (2007).