

Perbandingan Metode *Particle Swarm Optimization* dan *Firefly Algorithm* untuk Optimasi *Virtual Inertia Control* Berbasis *Capacitor Energy Storage*

Comparative Analysis of PSO and Firefly Algorithm for Virtual Inertia Controller Optimization Using Capacitor Energy Storage

Alief Nur Aisyi Maulidhia¹, Mirza Ardiana², Mat Syai'in³, Yudi Andika⁴, Riko Satrya Fajar Jaelani Putra⁵

¹ Program Studi D4 Teknik Kelistrikan Kapal, Jurusan Teknik Kelistrikan Kapal, Politeknik Perkapalan Negeri Surabaya

² Program Studi D4 Teknik Manajemen Bisnis, Jurusan Teknik Bangunan Kapal, Politeknik Perkapalan Negeri Surabaya

^{3,5} Program Studi D4 Teknik Otomasi, Jurusan Teknik Kelistrikan Kapal Politeknik Perkapalan Negeri Surabaya

⁴ Program Studi D4 Teknik Pengelasan, Jurusan Teknik Bangunan Kapal, Politeknik Perkapalan Negeri Surabaya

^{1,2,3,4,5} Jl. Teknik Kimia, Kampus ITS, Keputih Sukolilo, 60111, Indonesia
email: ¹aliefnur@ppns.ac.id, ²mirzaardiana@ppns.ac.id, ³matt.syaiin@ppns.ac.id,
⁴yudi.andika@ppns.ac.id, ⁵riko.satrya@ppns.ac.id

Informasi Artikel

Dikirim, 8 Oktober 2025
Diterima, 7 Desember 2025
Diterbitkan, 17 Desember 2025

Kata Kunci :

Penyimpanan Energi Kapasitor, *Particle Swarm Optimization*, *FireFly Optimization*, *Virtual Inertia Controller*

Keyword :

Capacitor Energy Storage, *Particle Swarm Optimization*, *FireFly Optimization*, *Virtual Inertia Control*.

ABSTRAK

Peningkatan kebutuhan energi mendorong integrasi pembangkit energi terbarukan berbasis power electronics ke dalam sistem tenaga listrik. Berbeda dengan pembangkit konvensional berbasis mesin sinkron yang secara alami menyediakan inersia mekanis, sebagian besar pembangkit terbarukan tidak memiliki kontribusi inersia langsung terhadap sistem. Kondisi ini menyebabkan penurunan total inersia sistem tenaga listrik, yang berdampak pada melemahnya respons frekuensi serta meningkatnya risiko osilasi frekuensi akibat gangguan perubahan beban. Penelitian ini mengusulkan penerapan *Virtual Inertia Controller* (VIC) berbasis *Capacitor Energy Storage* (CES) sebagai solusi untuk meniru karakteristik inersia mesin sinkron guna meningkatkan stabilitas frekuensi sistem. Parameter VIC dioptimasi menggunakan *Particle Swarm Optimization* (PSO) dan *Firefly Algorithm* (FA). Analisis dilakukan untuk mengevaluasi pengaruh VIC terhadap respons osilasi frekuensi akibat gangguan beban. Hasil simulasi menunjukkan bahwa penerapan VIC teroptimasi mampu meningkatkan redaman sistem, mempercepat waktu tunak (*settling time*), serta menurunkan nilai *Integral of Squared Error* (ISE) secara signifikan. Perbandingan kedua metode optimasi menunjukkan bahwa FA memberikan performa sedikit lebih unggul dibandingkan PSO, terutama dalam menurunkan nilai ISE dan mempercepat respons dinamis sistem. Dengan demikian, VIC berbasis CES yang dioptimasi menggunakan FA terbukti lebih efektif dalam meningkatkan stabilitas frekuensi sistem tenaga listrik pasca gangguan.

ABSTRACT

The increasing demand for electrical energy has accelerated the integration of renewable energy sources interfaced through power electronic converters into power systems. Unlike conventional synchronous generators that inherently provide mechanical inertia, most renewable energy-based generators contribute little to no physical inertia to the grid. This condition leads to a reduction in total system inertia, which weakens frequency response and increases the risk of frequency oscillations following load disturbances. This study proposes the implementation of *Virtual Inertia Controller* (VIC) based on *Capacitor Energy Storage* (CES) to emulate the inertia characteristics of synchronous generators and enhance frequency stability. The VIC parameters are optimized using *Particle Swarm Optimization* (PSO) and the *Firefly Algorithm* (FA). The analysis focuses on evaluating the effect of VIC on

frequency oscillation responses caused by load variations. Simulation results demonstrate that the optimized VIC significantly improves system damping, reduces *settling time*, and decreases the Integral of Squared Error (ISE). A comparative study further indicates that FA exhibits slightly superior performance compared to PSO, particularly in minimizing ISE and accelerating system dynamic response. Therefore, the proposed CES-based VIC optimized using FA is proven to be more effective in improving post-disturbance frequency stability in power systems.

1. PENDAHULUAN

Perkembangan pesat teknologi energi terbarukan, seperti tenaga surya dan angin, telah membawa tantangan baru dalam stabilitas sistem tenaga listrik. Sumber energi ini bersifat fluktuatif dan memiliki inersia rendah, yang dapat menyebabkan ketidakstabilan frekuensi sistem tenaga listrik. Salah satu solusi yang diterapkan untuk mengatasi masalah ini adalah penggunaan *Virtual Inertia Controller (VIC)* berbasis *Capacitor Energy Storage (CES)* [1]. *Virtual Inertia Controller (VIC)* berfungsi sebagai penyedia inersia buatan yang dapat meningkatkan kestabilan sistem dengan merespons perubahan frekuensi secara cepat. Namun, efektivitas *Virtual Inertia Controller* sangat bergantung pada parameter kontrol yang digunakan, sehingga diperlukan optimasi pengaturan yang tepat untuk mencapai kinerja terbaik [2][3].

Seiring dengan meningkatnya penetrasi energi terbarukan dalam sistem tenaga listrik, tantangan dalam menjaga kestabilan frekuensi semakin besar. Sistem tenaga listrik konvensional mengandalkan inersia alami dari generator sinkron untuk meredam perubahan frekuensi. Namun, dengan berkurangnya jumlah generator konvensional dalam jaringan, dibutuhkan strategi baru untuk mempertahankan stabilitas frekuensi. *Virtual Inertia Controller* berbasis *Capacitor Energy Storage* dapat memberikan solusi dengan menyimpan dan melepaskan energi dalam waktu singkat sesuai dengan kebutuhan sistem.

Penelitian-penelitian terkini menunjukkan bahwa pengembangan *Virtual Inertia Controller* telah bergerak dari konsep dasar menuju implementasi yang lebih adaptif dan efisien. Penelitian [3] mengembangkan *Virtual Inertia Controller* berbasis *machine learning* untuk sistem *Superconducting Magnetic Energy Storage (SMES)* yang mampu mempercepat respon dinamik dan menurunkan *overshoot* dibandingkan kontrol konvensional. Kemudian berdasarkan penelitian [4] mengusulkan strategi pengendalian *state of charge (SoC)* agar kapasitas penyimpanan energi tetap optimal selama operasi *Virtual Inertia Controller*, terutama pada sistem mikrogrid terisolasi. Selain itu, [5] memperkenalkan pendekatan fractional-order *Virtual Inertia Control* dengan teknik penalaan parameter menggunakan *stability boundary locus* yang terbukti meningkatkan ketahanan sistem terhadap variasi parameter. Kajian lain [6] menyoroti arah riset masa depan yang mencakup integrasi kontrol adaptif, pemulihan energi penyimpanan, serta penggunaan metode optimasi cerdas seperti *Particle Swarm Optimization (PSO)* dan *Neural Network* untuk meningkatkan efisiensi dan stabilitas *Virtual Inertia Control*.

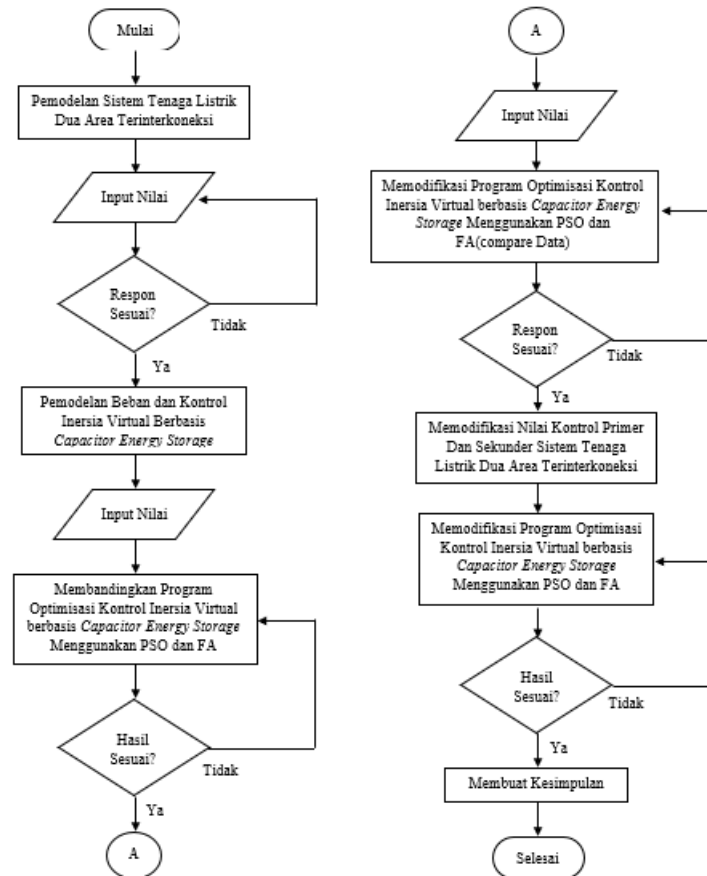
Dari berbagai penelitian tersebut, terlihat bahwa efektivitas VIC sangat dipengaruhi oleh penentuan parameter kontrol yang tepat. Optimasi parameter menjadi aspek kunci untuk mendapatkan keseimbangan antara stabilitas dinamik dan efisiensi energi. Pendekatan berbasis algoritma metaheuristik seperti *Particle Swarm Optimization (PSO)* [7] dan *Firefly Algorithm (FA)* [8] telah banyak digunakan untuk menyelesaikan permasalahan optimasi pada sistem tenaga listrik, termasuk dalam perancangan kontrol inersia virtual. PSO dikenal memiliki kemampuan konvergensi yang cepat, sementara FA unggul dalam eksplorasi ruang solusi yang lebih luas.

Berdasarkan hal tersebut, penelitian ini bertujuan untuk membandingkan kinerja metode PSO dan FA dalam optimasi parameter VIC berbasis *Capacitor Energy Storage (CES)* guna meningkatkan stabilitas frekuensi sistem tenaga listrik. Penelitian ini diharapkan dapat memberikan kontribusi baru dalam pengembangan strategi kontrol inersia virtual yang lebih adaptif, efisien, dan aplikatif pada sistem tenaga modern dengan penetrasi tinggi energi terbarukan. Oleh karena itu, penelitian ini bertujuan untuk membandingkan kinerja kedua algoritma dalam optimasi parameter VIC guna meningkatkan stabilitas frekuensi sistem tenaga listrik.

2. METODE PENELITIAN

2.1. Perancangan Sistem

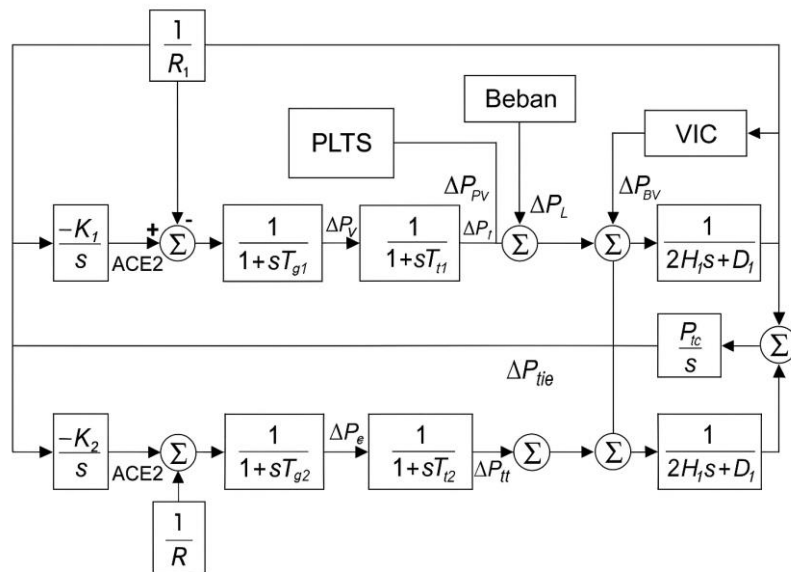
Dalam mengembangkan sistem Optimasi *Setting Virtual Inertia Control (VIC)* Berbasis *Capacitor Energy Storage (CES)* Menggunakan *Particle Swarm Optimization (PSO)* dan *Firefly Algorithm (FA)* terdapat beberapa tahapan yang harus dilakukan. Berikut adalah flowchart dari alur sistem yang akan dilaksanakan pada penelitian kali ini, yaitu sebagai berikut:



Gambar1. Metode Penelitian

2.2. Sistem Tenaga Listrik Dua Area Terinterkoneksi

Sistem tenaga listrik ini terdiri dari dua area, di mana masing-masing area memiliki empat unit pembangkit termal, dengan kapasitas 100 MVA per unit. Sistem ini menggunakan 100 MVA sebagai basis perhitungannya. Sebelum terjadi perubahan beban, sistem berada dalam kondisi keseimbangan, dengan beban aktual di Area 1 dan Area 2 masing-masing sebesar 250 MW. Di Area 1, terdapat pembangkit listrik tenaga surya (PLTS) berkapasitas 100 MW yang terintegrasi ke dalam sistem. Selain itu, sistem juga menggunakan CES (*Compressed Energy Storage*) di Area 1 dengan kapasitas 100 MJ/55 MW, yang berfungsi sebagai pengendali frekuensi untuk menjaga stabilitas sistem.



Gambar 2. Diagram Sistem Tenaga Listrik 2 Area Terinterkoneksi

Tabel 1. Data Saluran Transmisi Sistem Kelistrikan dua Area Terinterkoneksi

No	Dari Bus Nomor	Ke Nomor	Resistansi Seri (Rs) (Ω)	Reaktansi Seri (Xs) (Ω)	Susceptansi Shunt (B) (μS)
1	101	102	2.645	26.45	39.675
2	101	102	2.645	26.45	39.675
3	112	111	2.645	26.45	39.675
4	102	3	1.058	10.58	15.87
5	102	3	1.058	10.58	15.87
6	3	13	11.638	116.38	174.57
7	3	13	11.638	116.38	174.57
8	3	13	11.638	116.38	174.57
9	13	112	1.058	10.58	15.87
10	13	112	1.058	10.58	15.87
11	112	111	2.645	26.45	39.675
12	101	1	0.529	6.348	0.00
13	111	1	0.529	6.348	0.00
14	102	2	0.529	6.348	0.00
15	112	12	0.529	6.348	0.00

Tabel 2. Data Tegangan dan Daya Sistem Kelistrikan Dua Area Terinterkoneksi

No	Bus No	Tegangan Bus (mag) (pu)	Angle (deg)	Daya Nyata Generator (MW)	Daya Reaktif Generator (Mvar)
1	1	1.03	-0.18921	180	61.44
2	101	1.02378	-1.23204	0	0
3	102	1.00563	-3.65352	0	0
4	11	1.01	-10.2051	-36.5	-38.89
5	111	1.01412	-9.98764	0	0
6	112	1.02418	-9.53540	0	0
7	12	1.03	-8.49295	180	57.59
8	13	1.02098	-10.31073	0	0
9	2	1.01	-2.57070	180	42.83
10	3	0.99335	-5.66005	0	0

Tabel 3. Data Tegangan dan Beban Sistem Kelistrikan Dua Area Terinterkoneksi

No	Bus No	Tegangan Bus (mag) (pu)	Angle (deg)	Real Power Load (MW)	Reactive Power Load (Mvar)
1	1	1.03	-0.18921	0	0
2	101	1.02378	-1.23204	0	0
3	102	1.00563	-3.65352	0	0
4	11	1.01	-10.2051	0	0
5	111	1.01412	-9.98764	0	0
6	112	1.02418	-9.53540	0	0
7	12	1.03	-8.49295	0	0
8	13	1.02098	-10.31073	250	187.5
9	2	1.01	-2.57070	0	0
10	3	0.99335	-5.66005	250	187.5

Tabel 4. Data Dinamik Generator

No	Variabel	Gen. 1	Gen. 2	Gen. 3	Gen. 4
1	Rating (MVA)	250	250	250	250
2	Xl (pu)	0.022	0.022	0.022	0.022
3	Xd (pu)	1.8	1.8	1.8	1.8
4	X'd (pu)	0.3	0.3	0.3	0.3
5	X''d (pu)	0.25	0.25	0.25	0.25
6	T'do (sec)	8.0	8.0	8.0	8.0
7	T''do (sec)	0.03	0.03	0.03	0.03
8	Xq (pu)	1.8	1.8	1.8	1.8
9	X'q (pu)	1.7	1.7	1.7	1.7
10	X''q (pu)	0.25	0.25	0.25	0.25
11	T'qo (sec)	0.4	0.4	0.4	0.4
12	T''qo (sec)	0.05	0.05	0.05	0.05
13	H (sec)	6.5	6.5	6.5	6.2
14	D (pu)	0.6	0.6	0.6	0.6

Tabel 5. Data Parameter Sistem Kelistrikan 2 Area Terinterkoneksi

No.	Parameter	Sebelum Integrasi PLTS		Setelah Integrasi PLTS	
		Area 1	Area 2	Area 1	Area 2
1	Ki	0,3	0,3	0,3	0,3
2	R	0,05	0,0625	0,05	0,0625
3	β	20,6	16,9	20,6	16,9
4	D	0,6	0,9	0,48	0,72
5	H	5	4	4	3,2
6	Base Power	100	100	100	100
7	τ_g	0,2	0,3	0,2	0,3
8	τ_t	0,5	0,6	0,5	0,6
9	P_L	0,2	-	0,2	-

2.3. Optimasi Menggunakan Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) merupakan salah satu algoritma optimasi berbasis swarm intelligence yang terinspirasi dari perilaku sosial kawanan burung dan ikan. Setiap partikel dalam PSO merepresentasikan solusi kandidat yang bergerak dalam ruang pencarian dengan kecepatan dan posisi yang diperbarui berdasarkan pengalaman individu (personal best) serta informasi global (global best) [7], [9], [10].

Langkah pertama dalam optimasi ini adalah menentukan fungsi objektif, yaitu fungsi untuk meminimalkan kesalahan (error). Setelah itu, dilakukan pengaturan beberapa parameter PSO dalam menentukan populasi awal partikel diinisialisasi secara acak, di mana setiap partikel merepresentasikan inersia virtual (J_v) dan redaman virtual (D_v). Setelah mendapatkan hasil terbaik dari optimasi, nilai J_v dan D_v diekspor ke Simulink untuk diuji lebih lanjut. Setiap partikel kemudian dievaluasi berdasarkan nilai fitness-nya, untuk menentukan seberapa baik solusi yang dihasilkan dalam optimasi ini. Berikut adalah Langkah-langkah dari realisasi pengaturan parameter VIC dengan PSO, Algoritma dimulai dengan melakukan inisialisasi variabel awal dan memberikan posisi acak pada setiap partikel dalam ruang pencarian. Setelah itu, waktu iterasi pertama ditetapkan ($t = 1$), lalu masing-masing partikel dievaluasi nilai fungsi objektifnya. Nilai tersebut dibandingkan dengan nilai terbaik individu (personal best). Jika nilai baru lebih baik, maka personal best partikel tersebut diperbarui. Selain itu, nilai terbaik global (global best), yaitu nilai terbaik dari seluruh partikel, juga ikut

diperbarui apabila ditemukan solusi yang lebih optimal. Selanjutnya, setiap partikel memperbarui kecepatannya berdasarkan tiga komponen: kecepatan sebelumnya, jarak terhadap posisi personal best, dan jarak terhadap posisi global best. Setelah kecepatan baru dihitung, posisi partikel diperbarui. Iterasi kemudian dilanjutkan dengan meningkatkan nilai t dan kembali mengevaluasi fungsi objektif semua partikel. Pada setiap iterasi, dilakukan kembali pengecekan apakah personal best maupun global best menemukan solusi lebih baik dari sebelumnya, lalu dilakukan pembaruan jika diperlukan. Proses ini terus berulang hingga mencapai jumlah iterasi maksimum yang telah ditentukan. Ketika kondisi berhenti terpenuhi, algoritma mengakhiri proses dan menghasilkan solusi terbaik yang ditemukan oleh partikel selama pencarian berlangsung.

Particle Swarm Optimization (PSO) adalah salah satu algoritma optimasi berbasis populasi yang terinspirasi oleh perilaku sosial hewan, seperti kawanan burung atau ikan. Algoritma ini dikembangkan oleh Kennedy dan Eberhart pada tahun 1995. PSO digunakan untuk mencari solusi optimal (minimum atau maksimum) dari suatu fungsi objektif dengan cara menggerakkan partikel-partikel dalam ruang pencarian [11].

Particle Swarm Optimization (PSO) terinspirasi dari perilaku sosial kawanan, di mana setiap partikel dalam populasi mewakili solusi potensial dalam ruang pencarian. Posisi partikel menggambarkan nilai variabel dari fungsi objektif, sedangkan kecepatan menentukan arah dan jarak pergerakan partikel. Selama iterasi, partikel menyimpan posisi terbaik yang pernah dicapainya (personal best atau $pbest$) dan juga mengacu pada posisi terbaik seluruh partikel dalam populasi (global best atau $gbest$). PSO menggunakan dua rumus utama untuk memperbarui posisi dan kecepatan partikel. Rumus kecepatan sesuai dengan persamaan (1) adalah:

$$v_{i(t+1)} = w \cdot v_{i(t)} + c_1 \cdot r_1 \cdot (pbest_i - x_{i(t)}) + c_2 \cdot r_2 \cdot (gbest - x_{i(t)}) \quad (1)$$

di mana $v_i(t)$ adalah kecepatan partikel i pada iterasi t , w adalah faktor inersia, c_1 dan c_2 adalah koefisien percepatan (umumnya bernilai 2), serta r_1 dan r_2 adalah bilangan acak antara 0 dan 1. Rumus posisi diformulasikan pada persamaan (2) adalah:

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

Yang memperbarui posisi partikel berdasarkan kecepatan terbarunya. Dengan pendekatan ini, PSO secara iteratif mendekati solusi optimal melalui interaksi dan pengalaman kolektif.

2.4. Optimasi Menggunakan *Firefly Algorithm* (FA)

Firefly Algorithm (FA) merupakan salah satu algoritma metaheuristik berbasis perilaku alami yang dikembangkan oleh Xin-She Yang. FA terinspirasi dari perilaku biologis kunang-kunang yang berkomunikasi menggunakan cahaya untuk menarik perhatian satu sama lain. Dalam konteks komputasi, setiap *firefly* mewakili solusi potensial yang saling "tertarik" berdasarkan tingkat kecerahan (*brightness*), yang ditentukan oleh nilai fungsi objektif [12], [13], [14]

Langkah pertama dalam FA adalah mendefinisikan fungsi objektif, yaitu fungsi yang akan meminimalkan error dalam sistem. Setiap kunang-kunang akan merepresentasikan nilai parameter *Virtual Inertia* (J_v) dan *Virtual Damping* (D_v). Nilai awal dari populasi ini dipilih secara acak dalam rentang tertentu. Intensitas cahaya setiap kunang-kunang dihitung berdasarkan fungsi objektif, yang mengukur seberapa baik kombinasi nilai J_v dan D_v dalam mengurangi kesalahan sistem. Kunang-kunang dengan intensitas lebih rendah (solusi kurang optimal) akan bergerak menuju kunang-kunang yang lebih terang (solusi lebih baik). Pergerakan ini dipandu oleh persamaan daya tarik, yang bergantung pada jarak antara dua kunang-kunang dan faktor peredupan cahaya. Jika tidak ada kunang-kunang yang lebih terang, gerakan dilakukan secara acak untuk eksplorasi solusi baru. Setelah setiap iterasi, posisi kunang-kunang diperbarui dan kembali dievaluasi menggunakan fungsi objektif. Proses ini berulang hingga ditemukan kombinasi J_v dan D_v yang optimal. Setelah mendapatkan kombinasi terbaik dari J_v dan D_v , nilai ini diekspor ke Simulink untuk diuji dalam simulasi sistem tenaga listrik.

Firefly Algorithm (FA) merupakan salah satu algoritma metaheuristik berbasis perilaku alami yang meniru mekanisme pencahayaan kunang-kunang dalam menarik pasangan atau mangsa. Prinsip utamanya adalah bahwa setiap individu dalam populasi bergerak menuju individu lain dengan tingkat kecerahan lebih tinggi, yang secara matematis merepresentasikan solusi dengan nilai fungsi objektif lebih baik. Kunang-kunang yang lebih terang menarik kunang-kunang yang lebih redup, sementara intensitas cahaya menurun seiring jarak, sehingga ketertarikan lebih kuat terhadap kunang-kunang yang dekat dan lebih terang. Jika tidak ada kunang-kunang yang lebih terang, maka gerakannya bersifat acak. Reduksi cahaya memungkinkan algoritma menjaga keseimbangan antara eksplorasi solusi baru dan eksploitasi solusi terbaik. Proses pencarian solusi dimulai dengan inisialisasi parameter seperti jumlah kunang-kunang (N), koefisien penyerapan cahaya (γ), skala langkah (α), dan jumlah iterasi, serta menentukan batas atas dan bawah variabel. Setelah populasi awal dibuat secara acak dan intensitas

cahaya dievaluasi menggunakan fungsi objektif, setiap kunang-kunang akan mencari dan bergerak menuju yang lebih terang berdasarkan rumus:

$$x_i = x_i + \beta \cdot (x_j - x_i) + \alpha \cdot (\text{rand} - 0.5) \quad (3)$$

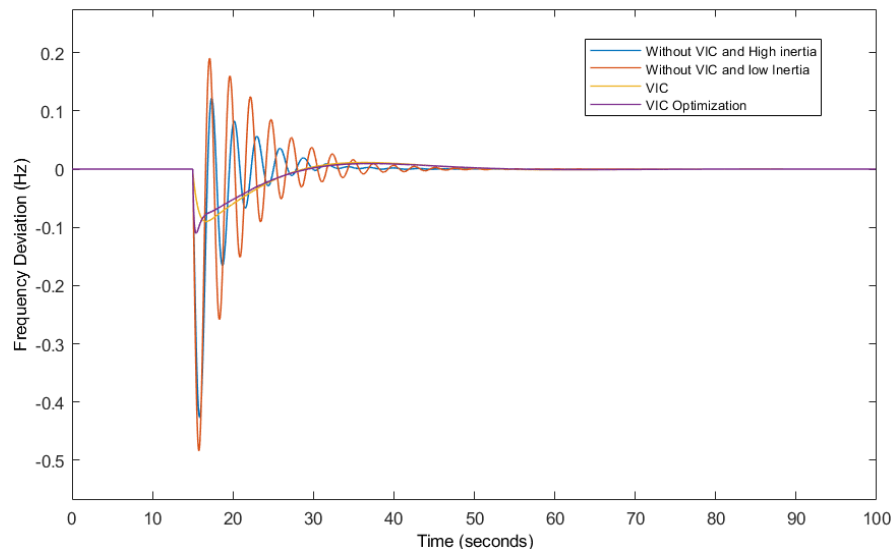
dengan β sebagai daya tarik yang dipengaruhi jarak, dan rand sebagai bilangan acak. Setelah pergerakan, intensitas cahaya diperbarui berdasarkan hasil evaluasi fungsi objektif, dan kunang-kunang diurutkan dari yang paling terang. Proses ini diulang hingga kriteria berhenti terpenuhi, seperti jumlah iterasi maksimal atau tercapainya konvergensi solusi.

3. HASIL DAN PEMBAHASAN

Dalam simulasi ini, gangguan sebesar 0.1 pu dan 0.2 pu diterapkan pada sistem di Area 1. Simulasi dilakukan untuk menganalisis respons sistem terhadap perubahan frekuensi pada dua rentang frekuensi berbeda, serta bagaimana daya antar-area berkontribusi selama perubahan beban. Hasilnya kemudian dibandingkan antara beberapa skenario: Sistem dengan VIC berbasis CES yang dioptimalkan menggunakan metode *Particle Swarm Optimization* (PSO) dan *FireFly* (FA), Sistem dengan VIC berbasis CES tanpa optimasi, Sistem dengan inersia tinggi tanpa VIC, Sistem dengan inersia rendah tanpa VIC. Tujuannya adalah untuk melihat bagaimana VIC yang telah dioptimalkan dapat meningkatkan stabilitas frekuensi dibandingkan dengan sistem yang tidak menggunakan VIC atau sistem dengan VIC tetapi tanpa optimasi.

3.1 Penambahan Beban 0.1pu dengan Optimasi PSO

Sistem kelistrikan interkoneksi dua area diberikan gangguan berupa penambahan beban sebesar 0.1 p.u atau sebesar 10 MW di area satu pada detik ke-15. Simulasi dilakukan selama 100 detik.



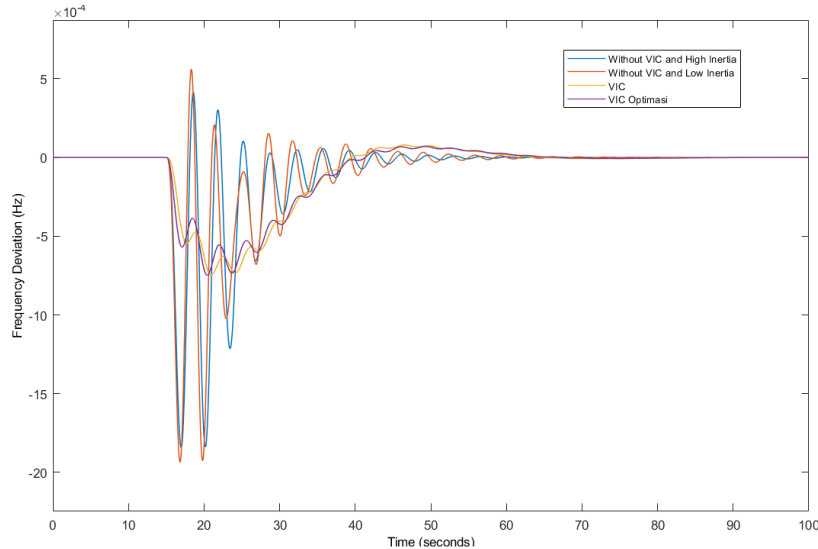
Gambar 3. Respon Frekuensi Area 1 (Beban=0.1pu) dengan PSO

Tabel 6. Metode Control Frekuensi Area 1 Terhadap perubahan Beban=0.1pu dengan PSO

Kriteria Kendali	Tanpa VIC dan High Inertia	Tanpa VIC dan Low Inertia	VIC	VIC Optimasi
Overshoot (Hz)	$1,21 \times 10^{-1}$	$1,90 \times 10^{-1}$	$1,12 \times 10^{-2}$	$9,47 \times 10^{-3}$
Nadir (Hz)	$-4,26 \times 10^{-1}$	$-4,83 \times 10^{-1}$	$-9,00 \times 10^{-2}$	$-1,10 \times 10^{-1}$
Settling time (detik)	11,51	17,98	10,31	9,45
ISE	$8,05 \times 10^{-5}$	$1,20 \times 10^{-3}$	$1,50 \times 10^{-5}$	$1,43 \times 10^{-5}$

Berdasarkan gambar dan tabel, integrasi PV (low inertia) ke dalam sistem memperburuk deviasi frekuensi dibandingkan sistem tanpa VIC dengan inersia tinggi, terlihat dari overshoot dan nadir yang lebih besar. Penerapan VIC berbasis CES membantu meredam osilasi frekuensi dengan overshoot yang lebih kecil ($1,12 \times 10^{-2}$ Hz), nadir yang lebih tinggi ($-9,00 \times 10^{-2}$ Hz), dan waktu pemulihan lebih cepat (10,31 detik).

Optimasi VIC berbasis CES dengan algoritma PSO dan FA semakin meningkatkan kinerja sistem dengan *overshoot* yang lebih rendah ($9,47 \times 10^{-3}$ Hz), waktu pemulihan yang lebih cepat (9,45 detik), dan error kuadrat terintegrasi (ISE) yang lebih kecil. Meskipun nadir sedikit lebih rendah ($-1,10 \times 10^{-1}$ Hz), optimasi ini secara keseluruhan meningkatkan stabilitas sistem. Dengan demikian, penerapan VIC berbasis CES, terutama setelah optimasi, efektif dalam meningkatkan stabilitas frekuensi pada sistem tenaga listrik berinersia rendah akibat integrasi PV.



Gambar 4. Respon Frekuensi Area 2 (Beban=0.1pu) dengan PSO

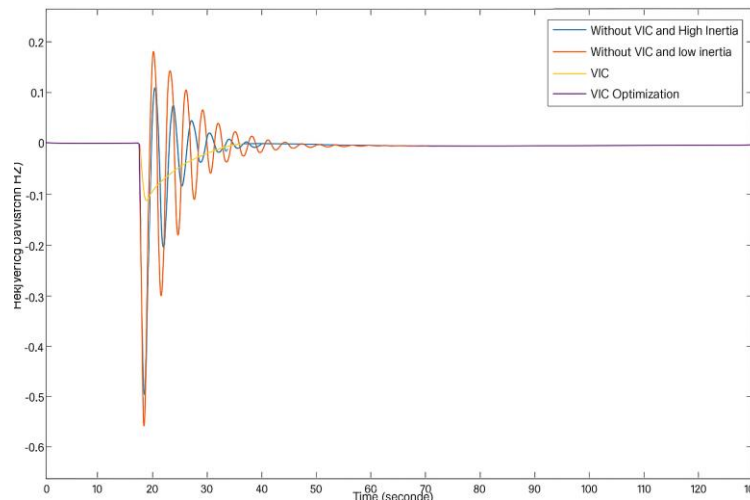
Tabel 7. Metode Control Frekuensi Area 2 terhadap Perubahan Beban=0.1pu dengan PSO

Kriteria Kendali	Tanpa VIC dan High Inertia	Tanpa VIC dan Low Inertia	VIC	VIC Optimasi
Overshoot (Hz)	$4,14 \times 10^{-4}$	$4,53 \times 10^{-4}$	$7,68 \times 10^{-5}$	$6,61 \times 10^{-5}$
Nadir (Hz)	$-1,84 \times 10^{-3}$	$-1,88 \times 10^{-3}$	$-7,41 \times 10^{-4}$	$-7,36 \times 10^{-4}$
Settling time (detik)	26,45	28,44	38,89	36,09
ISE	$1,00 \times 10^{-5}$	$9,04 \times 10^{-6}$	$5,32 \times 10^{-1}$	$5,06 \times 10^{-6}$

Berdasarkan gambar dan tabel di atas, dapat disimpulkan bahwa integrasi PV (*low inertia*) memperburuk stabilitas frekuensi, ditunjukkan dengan nilai *overshoot* ($4,53 \times 10^{-4}$ Hz) dan nadir ($-1,88 \times 10^{-3}$ Hz) yang lebih besar dibandingkan sistem dengan inersia tinggi. Penerapan VIC berbasis CES mampu meredam deviasi frekuensi secara signifikan, terlihat dari penurunan *overshoot* menjadi $7,68 \times 10^{-5}$ Hz dan nadir menjadi $-7,41 \times 10^{-4}$ Hz, meskipun waktu pemulihan meningkat menjadi 38,89 detik. Setelah optimasi VIC berbasis CES, *overshoot* lebih lanjut berkurang menjadi $6,61 \times 10^{-5}$ Hz dan nadir sedikit membaik menjadi $-7,36 \times 10^{-4}$ Hz, dengan waktu pemulihan yang lebih cepat (36,09 detik). Nilai ISE juga menunjukkan perbaikan setelah optimasi, mengindikasikan peningkatan efisiensi sistem dalam meredam osilasi frekuensi. Dengan demikian, penerapan dan optimasi VIC berbasis CES terbukti efektif dalam meningkatkan stabilitas frekuensi pada sistem tenaga listrik dengan inersia rendah akibat integrasi PV.

3.2 Penambahan Beban 0.1pu dengan Optimasi FA

Sistem kelistrikan interkoneksi dua area diberikan gangguan berupa penambahan beban sebesar 0.1 p.u atau sebesar 10 MW di area satu pada detik ke-15. Simulasi dilakukan selama 100 detik.

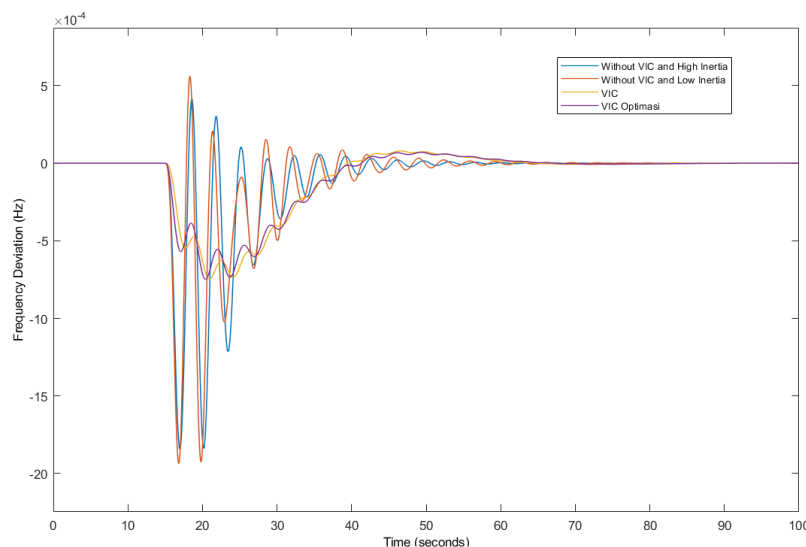


Gambar 5. Respon Frekuensi Area 1 (Beban=0.1pu) dengan FA

Tabel 8. Metode Control Frekuensi Area 1 terhadap Perubahan Beban=0.1pu dengan FA

Kriteria Kendali	Tanpa VIC dan High Inertia	Tanpa VIC dan Low Inertia	VIC	VIC Optimasi
Overshoot (Hz)	$1,11 \times 10^{-1}$	$1,80 \times 10^{-1}$	$1,10 \times 10^{-2}$	$8,49 \times 10^{-3}$
Nadir (Hz)	$-4,20 \times 10^{-1}$	$-4,50 \times 10^{-1}$	$-8,00 \times 10^{-2}$	$-1,05 \times 10^{-1}$
Settling time (detik)	9,50	15,89	8,30	7,45
ISE	$7,05 \times 10^{-5}$	$1,10 \times 10^{-4}$	$1,40 \times 10^{-5}$	$1,33 \times 10^{-5}$

Berdasarkan gambar dan tabel di atas, dapat disimpulkan bahwa penambahan PV (*low inertia*) tanpa kontrol VIC menyebabkan variasi frekuensi yang lebih besar dibandingkan dengan sistem dengan high inertia. Instalasi VIC berbasis CES membantu meredam osilasi frekuensi dengan mengurangi *overshoot*, meningkatkan nadir, dan mempercepat *settling time*. Setelah dilakukan optimasi VIC, hasilnya menunjukkan *overshoot* yang lebih kecil dan *settling time* yang lebih cepat dibandingkan VIC tanpa optimasi. Namun, nadirnya sedikit lebih rendah dari sebelum optimasi. Nilai ISE (*Integral of Squared Error*) yang lebih kecil pada VIC optimasi juga menunjukkan kinerja yang lebih baik dalam stabilisasi frekuensi sistem dibandingkan metode lainnya.



Gambar 6. Respon Frekuensi Area 2 (Beban=0.1pu) dengan FA

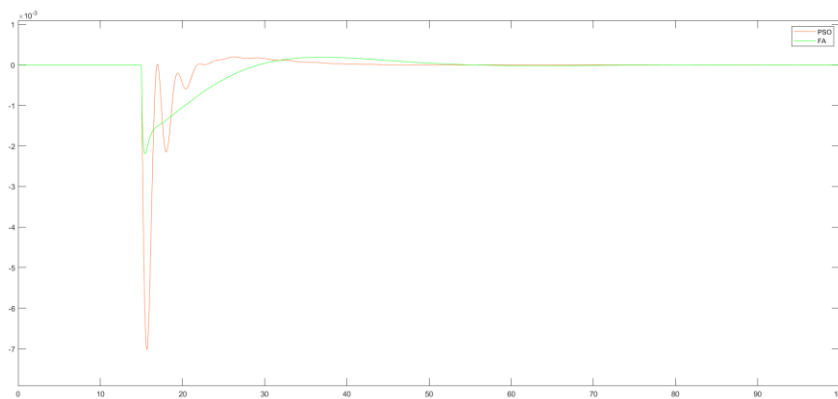
Tabel 9. Metode Control Respon Frekuensi Area 2 terhadap Perubahan Beban=0.1pu dengan FA

Kriteria Kendali	Tanpa VIC dan High Inertia	Tanpa VIC dan Low Inertia	VIC	VIC Optimasi
Overshoot (Hz)	$3,14 \times 10^{-4}$	$4,60 \times 10^{-4}$	$8,91 \times 10^{-5}$	$6,03 \times 10^{-5}$
Nadir (Hz)	$-1,74 \times 10^{-3}$	$-1,83 \times 10^{-3}$	$-6,46 \times 10^{-4}$	$-6,47 \times 10^{-4}$
Settling time (detik)	13,45	15,89	10,40	8,20
ISE	$7,05 \times 10^{-5}$	$1,20 \times 10^{-4}$	$1,50 \times 10^{-5}$	$1,43 \times 10^{-5}$

Berdasarkan gambar dan tabel di atas, dapat disimpulkan bahwa metode *Virtual Inertia Control* (VIC) dan VIC dengan optimasi memberikan kinerja yang lebih baik dibandingkan sistem tanpa VIC, baik dengan inersia tinggi maupun rendah. Hal ini ditunjukkan oleh nilai *overshoot* dan nadir yang lebih kecil, waktu *settling* yang lebih cepat, serta nilai *Integral Squared Error* (ISE) yang lebih rendah. VIC dengan optimasi menunjukkan hasil terbaik dengan ISE terendah ($1,43 \times 10^{-5}$), *overshoot* dan nadir terkecil, serta waktu *settling* tercepat (8.20 detik), yang menandakan stabilitas sistem yang lebih baik dan osilasi yang lebih cepat mereda. Oleh karena itu, penggunaan VIC, terutama dengan optimasi, dapat meningkatkan stabilitas frekuensi sistem tenaga listrik secara signifikan.

3.3 Perbandingan PSO dan FA

Berikut Analisa perbandingan performa dua algoritma optimasi, yaitu PSO (*Particle Swarm Optimization*) dan FA (*Firefly Algorithm*), dalam upaya menurunkan nilai fungsi objektif (dalam skala sekitar 10^{-3}) selama 100 iterasi.



Gambar 7. Perbandingan PSO dan FA

Dari Gambar 7 dapat dilihat sumbu horizontal mewakili jumlah iterasi, sedangkan sumbu vertikal menunjukkan nilai fungsi objektif lebih kecil nilainya berarti solusi yang lebih baik. Dari grafik terlihat bahwa FA menunjukkan penurunan tajam dan fluktuatif di awal (sekitar iterasi ke-15 hingga 30), yang mengindikasikan kemampuan menemukan solusi optimal dengan cepat, namun kurang stabil. Sebaliknya, PSO menunjukkan penurunan yang lebih bertahap dan stabil sepanjang iterasi, meskipun tidak secepat FA pada fase awal. Keduanya kemudian konvergen ke nilai minimum yang hampir sama setelah sekitar iterasi ke-50. Dengan demikian, FA unggul dari segi kecepatan konvergensi awal, sedangkan PSO lebih unggul dalam hal stabilitas dan konsistensi penurunan nilai. Namun, untuk menilai lebih mendalam, diperlukan informasi tambahan seperti jenis fungsi objektif, parameter yang dipakai, serta konteks aplikasinya.

4. KESIMPULAN

Berdasarkan hasil penelitian yang membandingkan optimasi menggunakan *Particle Swarm Optimization* (PSO) dan *Firefly Algorithm* (FA) dalam implementasi *Virtual Inertia Control* (VIC), dapat

disimpulkan bahwa metode optimasi memberikan peningkatan signifikan terhadap kestabilan frekuensi sistem tenaga listrik. Dari hasil simulasi, VIC dengan optimasi menunjukkan respons yang lebih cepat dengan waktu settling yang lebih pendek dibandingkan metode tanpa optimasi. Selain itu, nilai *Integral Squared Error* (ISE) yang lebih rendah pada VIC dengan optimasi menunjukkan bahwa kesalahan akumulatif dalam sistem lebih kecil, yang mengindikasikan performa kontrol yang lebih baik. Dibandingkan dengan kondisi tanpa VIC atau VIC tanpa optimasi, baik metode *Particle Swarm Optimization* (PSO) maupun *Firefly Algorithm* (FA) memberikan peningkatan dalam peredaman osilasi serta pengurangan *overshoot* dan nadir, yang berarti sistem lebih cepat mencapai kestabilan setelah gangguan terjadi.

Adapun perbandingan antara *Particle Swarm Optimization* (PSO) dan *Firefly Algorithm* (FA) menunjukkan bahwa *Firefly Algorithm* (FA) memiliki performa yang sedikit lebih baik dibandingkan *Particle Swarm Optimization* (PSO) dalam hal mengurangi nilai ISE dan mempercepat parameter *settling time*. Dari tabel hasil simulasi, nilai ISE untuk optimasi *Firefly Algorithm* (FA) lebih kecil dibandingkan dengan *Particle Swarm Optimization* (PSO), yang menunjukkan efektivitas *Firefly Algorithm* (FA) dalam mengoptimalkan parameter VIC untuk peredaman osilasi yang lebih baik. Selain itu, waktu settling pada *Firefly Algorithm* (FA) lebih cepat dibandingkan dengan *Particle Swarm Optimization* (PSO), yang berarti sistem dengan *Firefly Algorithm* (FA) lebih cepat kembali ke kondisi stabil setelah mengalami gangguan. Dengan demikian, meskipun kedua metode optimasi memberikan peningkatan signifikan dibandingkan metode tanpa optimasi, *Firefly Algorithm* (FA) lebih unggul dibandingkan *Particle Swarm Optimization* (PSO) dalam meningkatkan stabilitas frekuensi sistem tenaga listrik.

Selain itu, dari sisi kompleksitas komputasi, kedua algoritma sebenarnya memiliki orde kompleksitas yang serupa, yaitu $O(N \cdot D \cdot T)$, dengan N sebagai jumlah agen, D jumlah parameter yang dioptimasi, dan T jumlah iterasi. Namun, *Firefly Algorithm* (FA) cenderung memiliki kompleksitas komputasi yang sedikit lebih tinggi daripada PSO karena setiap *firefly* harus membandingkan tingkat kecerahannya dengan seluruh *firefly* lain sehingga menghasilkan interaksi berpasangan (*pairwise*), yang mendekati $O(N^2)$. Sebaliknya, PSO memiliki mekanisme pembaruan posisi dan kecepatan yang lebih sederhana tanpa interaksi penuh antar partikel. Meskipun demikian, hasil optimasi menunjukkan bahwa peningkatan kualitas solusi yang diberikan FA terutama nilai ISE yang lebih rendah dan waktu settling yang lebih cepat menjadikannya lebih efisien secara keseluruhan dalam konteks optimasi VIC untuk perbaikan stabilitas frekuensi sistem tenaga listrik.

Penelitian selanjutnya dapat difokuskan pada pengujian algoritma optimasi lain yang lebih modern untuk dibandingkan dengan PSO dan FA, seperti *Grey Wolf Optimizer* (GWO), *Whale Optimization Algorithm* (WOA), *Ant Lion Optimizer* (ALO), maupun *Genetic Algorithm* (GA). Evaluasi terhadap algoritma-algoritma tersebut diharapkan dapat memberikan gambaran yang lebih luas mengenai metode optimasi yang paling efektif dan efisien dalam meningkatkan performa *Virtual Inertia Control*, terutama dalam aspek kecepatan konvergensi, kestabilan hasil, serta kemampuan menemukan solusi global pada sistem tenaga listrik dengan dinamika yang kompleks.

DAFTAR PUSTAKA

- [1] B. Kroposki *et al.*, "Achieving a 100% Renewable Grid: Operating Electric Power Systems with Extremely High Levels of Variable Renewable Energy," *IEEE Power and Energy Magazine*, vol. 15, no. 2, pp. 61–73, Mar. 2017, doi: 10.1109/MPE.2016.2637122.
- [2] U. Tamrakar, D. Shrestha, M. Maharjan, B. P. Bhattarai, T. M. Hansen, and R. Tonkoski, "Virtual inertia: Current trends and future directions," Jun. 26, 2017, *MDPI AG*. doi: 10.3390/app7070654.
- [3] H. Setiadi, M. Abdillah, Y. Afif, and R. Delfianti, "Adaptive virtual inertia controller based on machine learning for superconducting magnetic energy storage for dynamic response enhanced," *International Journal of Electrical and Computer Engineering*, vol. 13, no. 4, pp. 3651–3659, Aug. 2023, doi: 10.11591/ijece.v13i4.pp3651-3659.
- [4] F. H. Jufri, J. Jung, B. Sudiarto, and I. Garniwa, "Development of Virtual Inertia Control with State-of-Charge Recovery Strategy Using Coordinated Secondary Frequency Control for Optimized Battery Capacity in Isolated Low Inertia Grid," *Energies (Basel)*, vol. 16, no. 14, Jul. 2023, doi: 10.3390/en16145463.
- [5] Y. Zeng, Q. Yang, Y. Lin, Y. Chen, X. Chen, and J. Wen, "Fractional-Order Virtual Inertia Control and Parameter Tuning for Energy-Storage System in Low-Inertia Power Grid," *Protection and Control of Modern Power Systems*, vol. 9, pp. 70–83, Oct. 2024, doi: 10.23919/PCMP.2023.000111.
- [6] M. A. Shobug, N. A. Chowdhury, M. A. Hossain, M. J. Sanjari, J. Lu, and F. Yang, "Virtual Inertia Control for Power Electronics-Integrated Power Systems: Challenges and Prospects," Jun. 01, 2024, *Multidisciplinary Digital Publishing Institute (MDPI)*. doi: 10.3390/en17112737.
- [7] H. Huang, J. Qiu, and K. Riedl, "On the Global Convergence of Particle Swarm Optimization Methods," *Appl Math Optim*, vol. 88, no. 2, Oct. 2023, doi: 10.1007/s00245-023-09983-3.
- [8] I. D. Fajuke and A. K. Raji, "Firefly Algorithm-Based Optimization of the Additional Energy Yield of Bifacial PV Modules," *Energies (Basel)*, vol. 15, no. 7, Apr. 2022, doi: 10.3390/en15072651.

- [9] L. K. Okwu Modestus O. and Tartibu, "Particle Swarm Optimisation," in *Metaheuristic Optimization: Nature-Inspired Algorithms Swarm and Computational Intelligence, Theory and Applications*, Cham: Springer International Publishing, 2021, pp. 5–13. doi: 10.1007/978-3-030-61111-8_2.
- [10] S. Nasional, T. Elektro, S. Informasi, and T. Informatika, "Particle Swarm Optimazition (Pso)," pp. 49–56, 2021.
- [11] H. Akewe, E. P. Fasina, and B. A. Sawyerr, "Convergence and Stability Analysis of Particle Swarm Optimization Using The Fixed Point Method," *International Journal of Mathematical Modelling & Computations*, vol. 15, no. 03, pp. 89–99, 2025, doi: 10.71932/ijm.2025.1081384.
- [12] H. Setiadi and K. O. Jones, "Power system design using firefly algorithm for dynamic stability enhancement," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 1, no. 3, pp. 446–455, Mar. 2016, doi: 10.11591/ijeecs.v1.i3.pp446-455.
- [13] J. Lloret *et al.*, "An Optimized Load Balancing Using Firefly Algorithm in Flying Ad-Hoc Network," 2022, doi: 10.3390/electronics.
- [14] S. Padhan, R. K. Sahu, and S. Panda, "Application of Firefly Algorithm for Load Frequency Control of Multi-area Interconnected Power System," *Electric Power Components and Systems*, vol. 42, no. 13, pp. 1419–1430, 2014, doi: 10.1080/15325008.2014.933372.

