CHAPTER 5

Metaheuristic Controller Optimization for HESS

5.1 Metaheuristic Optimization Techniques

The selection of optimization algorithms is guided by the need to effectively handle the problem of multi-objective and non-linear characteristics. Advanced heuristic and metaheuristic algorithms are chosen for their proven ability to explore large solution spaces, manage dynamic constraints, and achieve optimal trade-offs between conflicting objectives, such as minimizing energy losses and maximizing efficiency. Figure 5.1 represents the Metaheuristic Optimization Techniques.

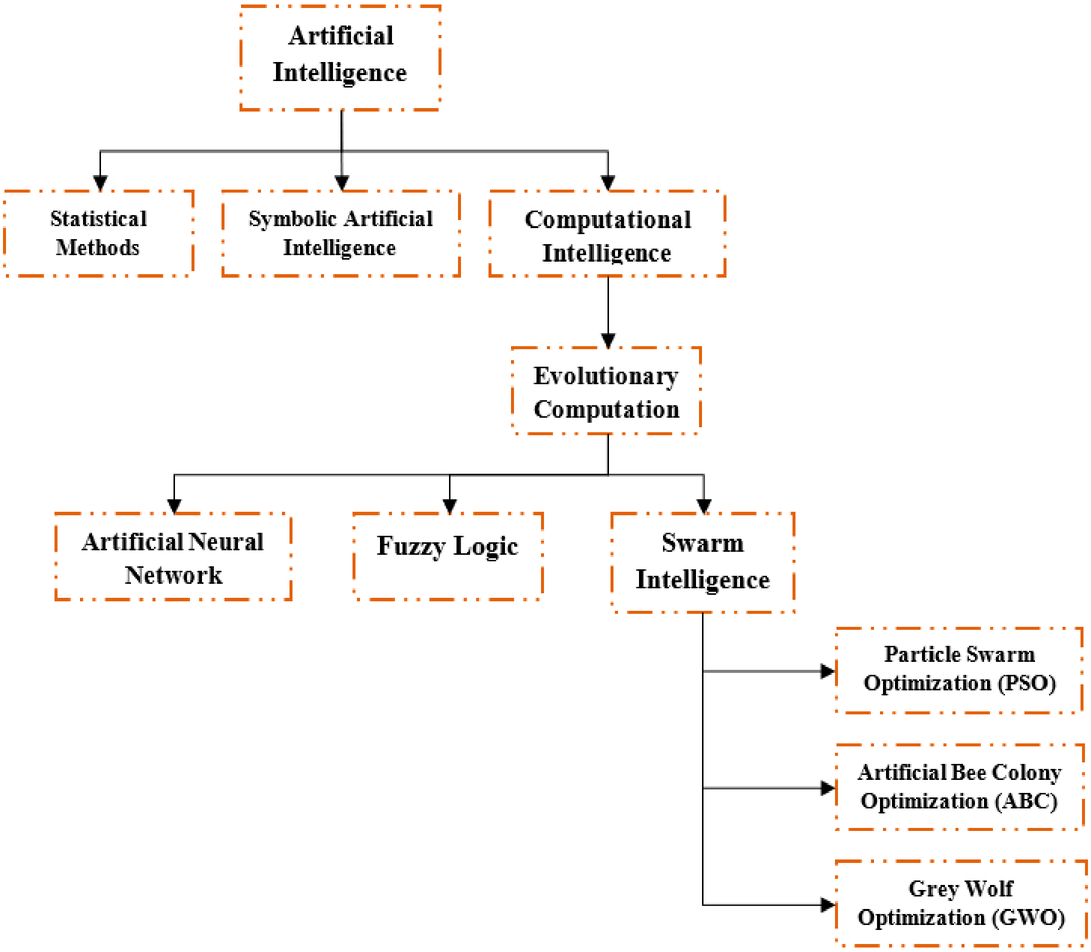

Figure 5. 1 Classification of Metaheuristic Techniques

Several optimization algorithms, such as Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), and Grey Wolf Algorithm (GWO), can be employed to achieve reliable and efficient system performance.

5.2 Problem Formulation of the Optimization Problem

To simultaneously minimize $\boldsymbol{E_{loss}}$ and maximize $\boldsymbol{\eta_{bat}}$, the problem can be expressed as a multi-objective optimization problem:

$$\boldsymbol{f}(\boldsymbol{x}) = \min \sum_{t=1}^{T} (\boldsymbol{E_{loss}})$$

Subjected to $\max \sum_{t=1}^{T} (\boldsymbol{\eta_{bat}})$

$$\mathbf{X} = [P_{bat}, SOC_{bat}, P_{super}, SOC_{super}]$$

Objective Function

single objective function by using a weighted sum approach:

$$\text{Objective Function} = W_1 \cdot \boldsymbol{E_{loss}} - W_2 \cdot \boldsymbol{\eta_{bat}} \tag{4.1}$$

Where:

w1 and w2 are weights representing the relative importance of minimizing losses and maximizing efficiency.

Objective 1: Minimize Energy Losses ($\boldsymbol{E_{loss}}$)

The energy losses in the battery are calculated as:

$$E_{loss} = \sum_{t=1}^{T} \left(\frac{P_{bat}^2(t) \cdot R_{int} \cdot \Delta t}{V_{bat,max}} \right)$$

Where:

$P_{bat}(t)$: Power output/input of the battery at time t.

R_{int}: Internal resistance of the battery.

Δt: Time step duration.

$V_{bat,max}$: Maximum voltage of the battery.

T: Total number of time steps.

This equation sums up the power losses at each time step due to the battery's internal resistance. The losses are proportional to the square of the power Pbat(t), meaning higher power levels result in significantly greater losses.

Objective 2: Maximize Battery Efficiency ($\boldsymbol{\eta_{bat}}$)

Battery efficiency is defined as:

$$\eta_{bat} = \frac{E_{out}}{E_{out} + E_{loss}}$$

$$E_{out} = \sum_{t=1}^{T} (P_{bat}(t) \cdot \Delta t)$$

Where:

E_{out} : Total energy output of the battery over time.

E_{loss} : Total energy losses in the battery over time.

Efficiency represents the fraction of the total energy input that is useful energy output. It is a measure of how effectively the battery delivers usable energy, accounting for losses.

Design Constraints

The following constraints are essential for ensuring the safe, efficient, and reliable operation of the energy storage system. These constraints define the permissible operational ranges for the battery and supercapacitor, encompassing both power and state of charge (SOC).

1. Battery Power (P_{bat})

The battery power represents the energy delivered or absorbed by the battery during operation. To prevent overheating, overloading, or premature degradation, must remain within the operational boundaries dictated by the system's design:

Battery Power ($\boldsymbol{P_{bat}}$): $P_{bat,min} \leq P_{bat} \leq P_{bat,max}$

Where:

Pbat, min: Minimum allowable battery power.

Pbat, max: Maximum allowable battery power.

These limits are established based on the battery's physical characteristics and safety considerations, ensuring both performance and durability.

2. Battery State of Charge (SOC_{bat})

The battery's state of charge is a critical parameter that reflects its current energy level relative to its capacity. Maintaining within the prescribed range ensures that the battery is neither overcharged nor excessively discharged, which could otherwise lead to efficiency losses or irreversible damage:

$$\textbf{Battery State of Charge } (\boldsymbol{SOC_{bat}}): \qquad SOC_{bat,min} \leq SOC_{bat} \leq SOC_{bat,max}$$

Where:

SOC_{bat}, min: Minimum allowable state of charge for the battery.

SOC_{bat}, max: Maximum allowable state of charge for the battery.

These thresholds are typically determined by the battery manufacturer and are influenced by the chemistry and intended application of the battery.

3. Supercapacitor Power (P_{SC})

The power exchanged by the supercapacitor must be managed within its design specifications to prevent thermal stress and ensure optimal performance. The allowable range for is expressed as:

$$\textbf{Supercapacitor Power } (\boldsymbol{P_{SC}}): \qquad P_{SC,min} \leq P_{SC} \leq P_{SC,max}$$

Where:

- P_{SC}, min: Minimum allowable supercapacitor power.
- P_{SC}, max: Maximum allowable supercapacitor power.

These limits account for the supercapacitor's rapid charge and discharge capabilities, which are critical for handling high-power transients in hybrid energy systems.

4. **Supercapacitor State of Charge (SOCsuper)**

The state of charge of the supercapacitor defines the amount of energy stored at a given time. Operating outside the specified range could reduce the efficiency or lifespan of the supercapacitor. The permissible range is given by:

Supercapacitor State of Charge (SOC_{bat}): $SOC_{super,min} \leq SOC_{super} \leq SOC_{super,max}$

Where:

$SOC_{super,\ min}$: Minimum allowable state of charge for the supercapacitor.

$SOC_{super,\ max}$: Maximum allowable state of charge for the supercapacitor.

This constraint ensures that the supercapacitor operates within its energy storage capacity while mitigating the risks of overcharging or deep discharge.

Decision Parameters

- Battery Parameters

1 **Capacity (C_{bat}):** The energy storage capacity of the battery, essential for balancing energy supply and demand within the system [1].

$$E_{bat}=C_{bat} * V_{bat,\ nom} \tag{4.2}$$

Where,

C_{bat}: Battery capacity (Ah)

$V_{bat,\ nom}$: Nominal voltage of the battery (V)

2 **Initial State of Charge ($SOC_{bat,\ initial}$):** Represents the initial charge level of the battery, which influences its operational readiness and lifecycle performance.

$$SOC_{bat}(t)=SOC_{bat}(t-1)+\frac{E_{bat}*P_{bat}(t)}{\Delta t}$$

Subject to

$$SOC_{bat,\ min} \leq SOC_{bat}(t) \leq SOC_{bat,\ max}$$

3 **Voltage Limits ($V_{bat,\ max}$, $V_{bat,\ min}$):** The maximum and minimum allowable voltages to ensure safe operation within the electrochemical stability range.

$$V_{bat,\,min} \leq V_{bat}(t) \leq V_{bat,\,max}$$

4 **Internal Resistance (R_{int}):** A critical parameter affecting energy efficiency by dictating resistive losses during charging and discharging cycles.

$$P_{loss}(t)=I_{bat}^2(t) * R_{int} \tag{4.3}$$

Where $I_{bat}(t)$ is the current flow in the battery.

- Supercapacitor Parameters

1 **Capacitance (C_{SC}):** Determines the energy storage capacity and response time of the supercapacitor for handling transient power demands.

$$E_{SC} = \frac{1}{2}\, C_{sc} * V_{sc}^2 \tag{4.4}$$

Where,

C_{sc} : Capacitance (F)

V_{SC}: Voltage of the supercapacitor (V)

2 **Maximum Voltage ($V_{SC,\,max}$):** The upper voltage limit to ensure safe operation and prevent overvoltage conditions.

$$V_{SC,\,min} \leq V_{SC}(t) \leq V_{SC,\,max}$$

3 **Initial State of Charge ($SOC_{SC,initial}$):** The starting charge level of the supercapacitor, optimized to ensure effective energy buffering and system responsiveness [2-5].

$$SOC_{SC}(t)=SOC_{SC}(t-1) + \frac{E_{SC}*P_{SC}(t)}{\Delta t}$$

Subject to

$$SOC_{SC,\,min} \leq SOC_{SC}(t) \leq SOC_{SC,\,max}$$

- Environmental and Operational Parameters

1 **Temperature Range (T_{low}, T_{high}):** The operational temperature limits that ensure the system components perform efficiently under varying ambient conditions.

$$T_{low} \leq T(t) \leq T_{high}$$

2 **Photovoltaic (PV) Generation Profiles:** Realistic PV output profiles that incorporate stochastic variations in solar irradiance, ensuring robust energy management under fluctuating renewable energy inputs.

$$P_{PV}(t) = \mathrm{F}\ (I_{irr}(t)T_{PV}(t)) \tag{4.5}$$

Where,

$I_{irr}(t)$: Solar irradiance at time.

$T_{PV}(t)$: PV cell temperature at time.

3 **Dynamic Load Profiles:** Load demand patterns reflecting real-world conditions with temporal variations, are critical for ensuring the system adapts to changing power requirements.

$$P_{load}(t) = P_{base} + P_{var}(t) \tag{4.6}$$

Where,

P_{base}: Base load demand

$P_{var}(t)$: Time-varying load demand

5.3 Particle Swarm Optimization (PSO)

- A population-based stochastic optimization technique inspired by the social behavior of birds flocking or fish schooling.
- Each particle in particle swarm optimization has an associated *position, velocity*, and *fitness value*. Each particle keeps track of the best fitness value and position.
- Also, the record of global best fitness value and position is maintained.

5.3.1 Implementation of PSO Algorithm

Implementing the Particle Swarm Optimization (PSO) algorithm for energy management in a DC microgrid leverages the optimization technique to balance the power distribution between a hybrid energy storage system (HESS) and the load. The HESS includes a battery and a supercapacitor to minimize energy losses and improve overall efficiency while adhering to system constraints. Table 5.1 represents PSO Parameters and Figure 5.2 shows the Flowchart of Particle Swarm Optimization.

5.3.2 Parameters of the PSO Algorithm

Table 5. 1 PSO Parameters

S.R. NO.	**Variable Name**	**Value**
1.	No. of Particles	100
2.	Maximum iteration	50
4	Inertia weight (ω)	0.5
5	Cognitive coefficient (c_1)	1.5
6	Social coefficient (c_2)	1.5

5.3.3 Flowchart and Procedure of PSO

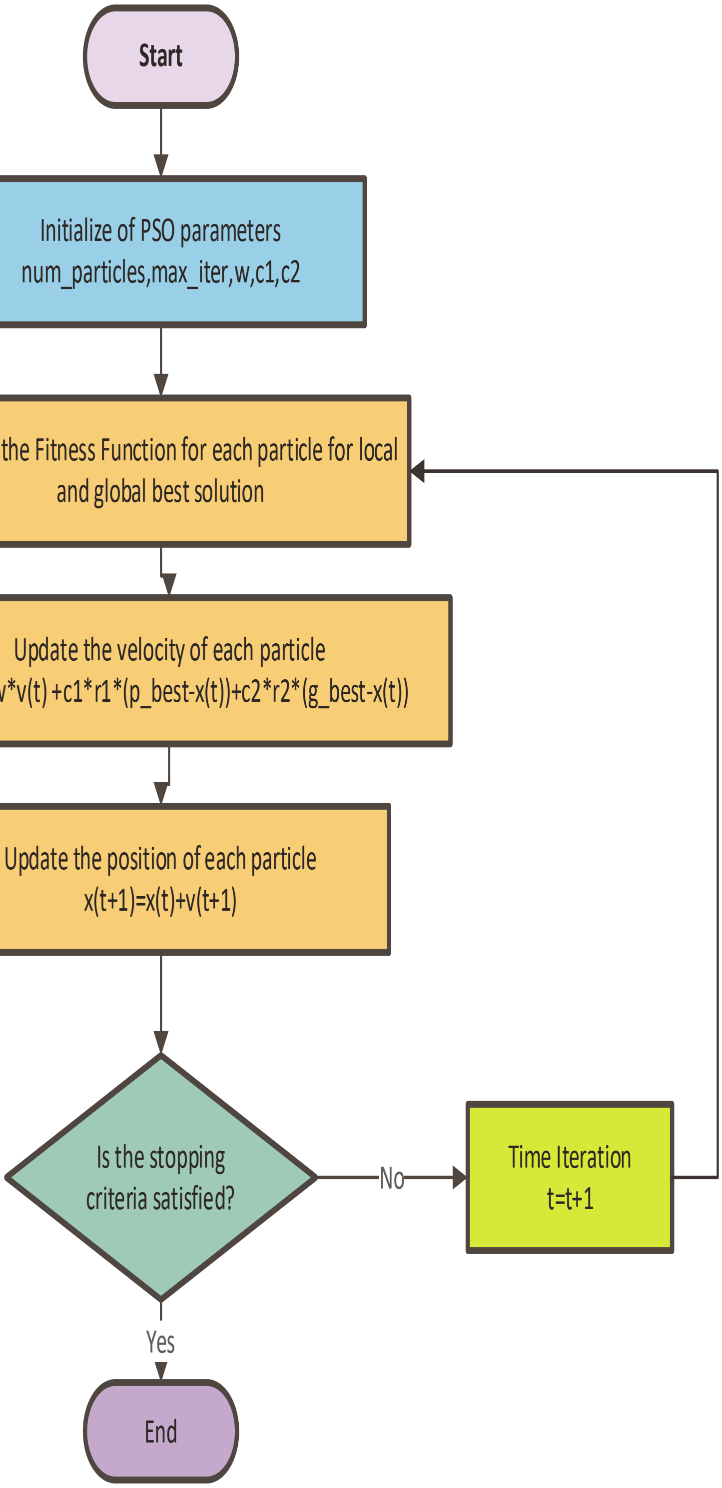

Figure 5. 2 Flowchart of Particle Swarm Optimization

Below is an overview of the implementation steps of PSO in DC microgrid:

Step 1: Define System Parameters

These System parameters include the characteristics of the battery, supercapacitor, and other components in the DC microgrid, such as:

- **Battery Parameters**:

  Capacity, SOC_{min}, SOC_{max}, R_{int}, etc.

- **Supercapacitor Parameters**:

  C_{SC}, SOC_{SC}_min, SOC_{SC}_max.

- **Environmental Parameters**:

  PV generation profile, load profile, etc.

Step 2: Initialize Particles and Velocity

Particles represent potential solutions to the optimization problem. Each particle contains a set of parameters that can be optimized. Particles represent the **power split ratio** between the battery and supercapacitor, or **the SOC levels** of the energy storage systems.

1. **Initialize Particle Positions**:
   The particle positions $\mathbf{x_i}$ = [P_b, SOCb, P_{sc}, SOCsc] are randomly initialized within predefined bounds:

   $P_{bat} \in$ [P_{bat}, min, Pbat, max], $SOC_{bat} \in$ [SOC_{bat}, min, SOCbat, max]

   Where, P_{bat} is the battery power, and SOC_{bat} is the battery's state of charge.
2. **Initialize Velocities**:
   The initial velocities $\mathbf{v_i}$ of the particles are set to zero or small random values.

Step 3: Evaluate Fitness (Objective Function)

The objective function is a mathematical representation of the system's performance. In the context of HESS and a DC microgrid, the fitness can be evaluated by considering factors such as:

- Minimizing Energy Loss
- Maximizing Battery Efficiency
- Balancing SOC Levels

Step 4: Update Personal Best (pbest)

Each particle has a personal best solution, which is the best position (set of parameter values) it has found so far. For each particle i, we check if the current fitness score is better than the previous best:

$$f(xi) < f\,(pbest,\, i) \Rightarrow pbest,\, i{=}xi.$$

Step 5: Update Global Best (gbest)

The global best position **gbest** is the best solution for any particle in the entire swarm. After evaluating the fitness of all particles, we update the global best as follows:

$$f(xi) < f(gbest) \Rightarrow gbest{=}xi.$$

The global best solution represents the optimal configuration for the microgrid, minimizing energy loss, balancing SOC levels, and ensuring efficient energy sharing between battery and supercapacitor.

Step 6: Update Velocity and Position

Once the personal best and global best positions are determined, the particles' velocities and positions are updated according to the standard PSO update rules:

1. **Velocity Update**:
   The velocity of each particle is updated using:

$$V_i^{(t+1)} = \omega \cdot V_i^{(t)} + c_1 \cdot r_1 \cdot \left(P_i^{best} - X_i^{(t)}\right) + c_1 \cdot r_1 \cdot \left(G^{best} - X_i^{(t)}\right) \tag{4.7}$$

where:

ω is the inertia weight,

c_1 and c_2 are cognitive and social coefficients,

r_1 and r_2 are random numbers between 0 and 1.

2. **Position Update**: The position of each particle xi is updated using:

$$X_i^{(t+1)} = X_i^{(t)} + V_i^{(t+1)} \tag{4.8}$$

3. **Boundary Constraints**:

   To ensure the particles stay within feasible bounds (Pb should be between Pb, min and Pb, max).

   Now, check and adjust the position if necessary:

   Pb=min (max (Pb,min),Pb,max).

Step 7: Termination Condition

The algorithm continues iterating until a stopping criterion is met. The global best solution **gbest** represents the optimal parameter configuration that minimizes energy losses, balances SOC levels, and optimizes energy sharing.

Particle Swarm Optimization (PSO) techniques were evaluated in terms of stability improvements, and enhanced battery life to demonstrate the effectiveness of the proposed approach.

5.3.4 Real-Time 3D Visualization PSO Results

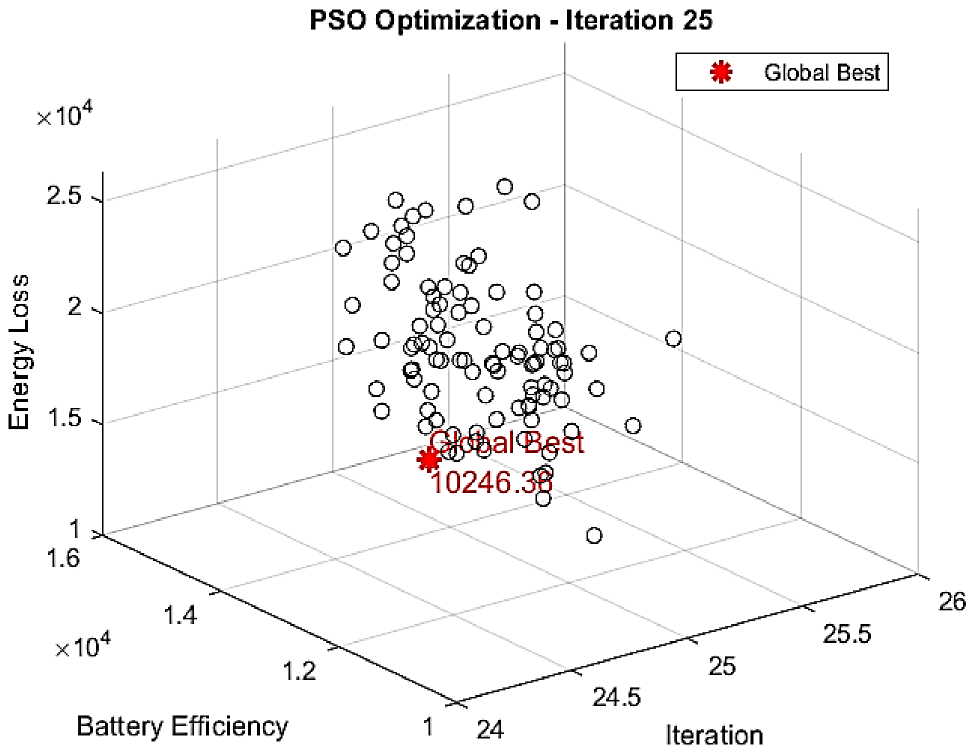

Figure 5. 3 PSO Optimization at Iteration 25

Iteration 25

Global Best Position: [11296.5289213849 0.242076816176983 4716.33000822496 0.688134227613088]

Global Best Score: 10246.36 W

Battery Efficiency = 70%

Figures 5.3 and 5.4 represent the real-time visualization of the Particle Swarm Optimization (PSO) algorithm applied to optimize a DC microgrid system. The plot showcases the interaction between battery efficiency, energy loss, and iteration progress, providing a dynamic view of the optimization process. The scattered black circles represent the positions of particles within the swarm, with each particle corresponding to a candidate solution evaluated by the algorithm. These particles explore the search space, converging toward regions of higher performance as the iterations progress.

Figure 5.3, corresponding to Iteration 25, the PSO algorithm achieves a global best score of 10,246.36 W, indicating the lowest energy loss recorded at this stage of the optimization process. The corresponding global best position is identified as [11296.53, 0.2421, 4716.33, 0.6881], representing key control parameters, such as voltage levels, droop coefficients, and current limits. The battery efficiency achieved at this iteration is 70%, meaning the system operates with a relatively moderate level of energy loss compared to the initial state. The scatter points are distributed across a broad range of energy loss values, indicating that the particles are still exploring various potential solutions. However, the convergence of particles towards the global best solution suggests that the optimization process is progressing effectively.

In Figure 5.4 corresponding to Iteration 38, further improvements are observed as the algorithm converges toward a more optimal solution. At this stage, the global best score improves to 9,851.56 W, indicating a significant reduction in energy loss compared to Iteration 25. The corresponding global best position is identified as [6641.54, 0.5814, 12735.59, 0.7184], showing refined control parameters that result in better system performance. Notably, the battery efficiency at this iteration increases to 76%, reflecting enhanced energy management within the DC microgrid. The distribution of scatter points in the plot shows a tighter clustering around the global best solution, indicating that the particles have converged towards a more stable and efficient operating point.

Therefore, Iteration 38 highlights the effectiveness of the PSO algorithm in reducing energy loss while improving battery efficiency. The battery efficiency of 76% improvement that the optimization process effectively tunes the control parameters to achieve better performance as iterations progress. The reduction in energy loss of 9,851.56 W also confirms the algorithm's ability to identify optimal solutions over time.

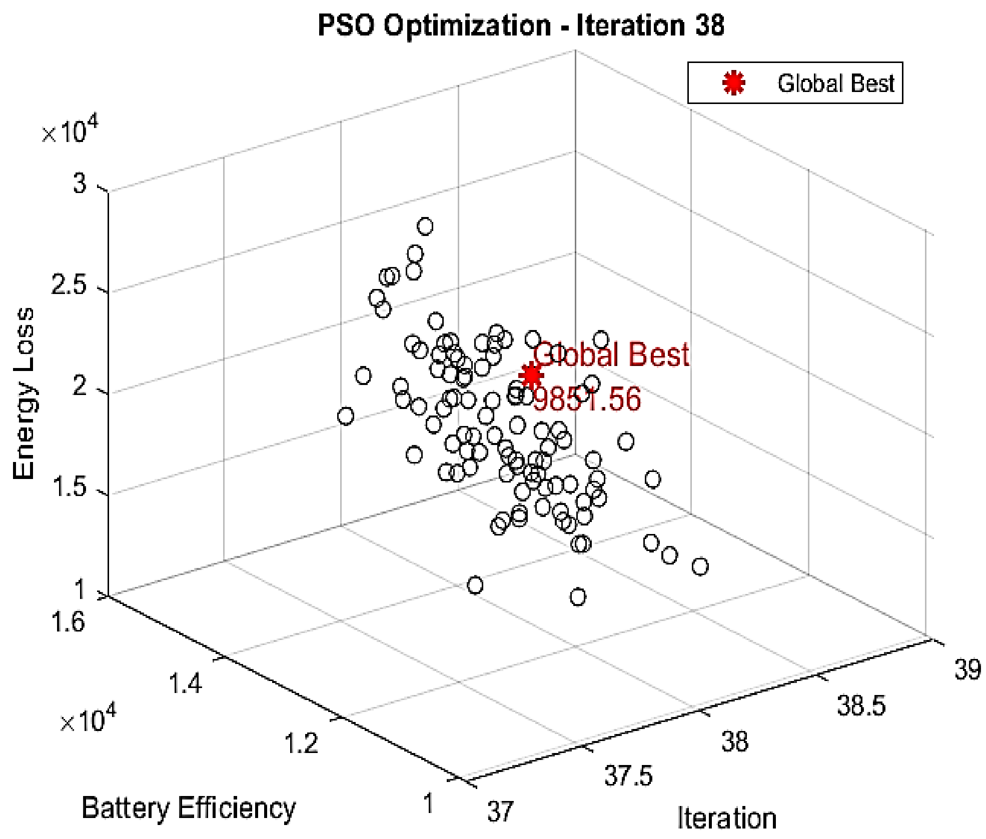

Figure 5. 4 PSO Optimization at Iteration 38

Iteration 38

Global Best Position: [6641.53811641003 0.581405748899299 12735.5943685987 0.718382351983945]

Global Best Score: 9851.56 W

Best battery efficiency achieved: 76 %

5.3.5 Total energy loss trajectory

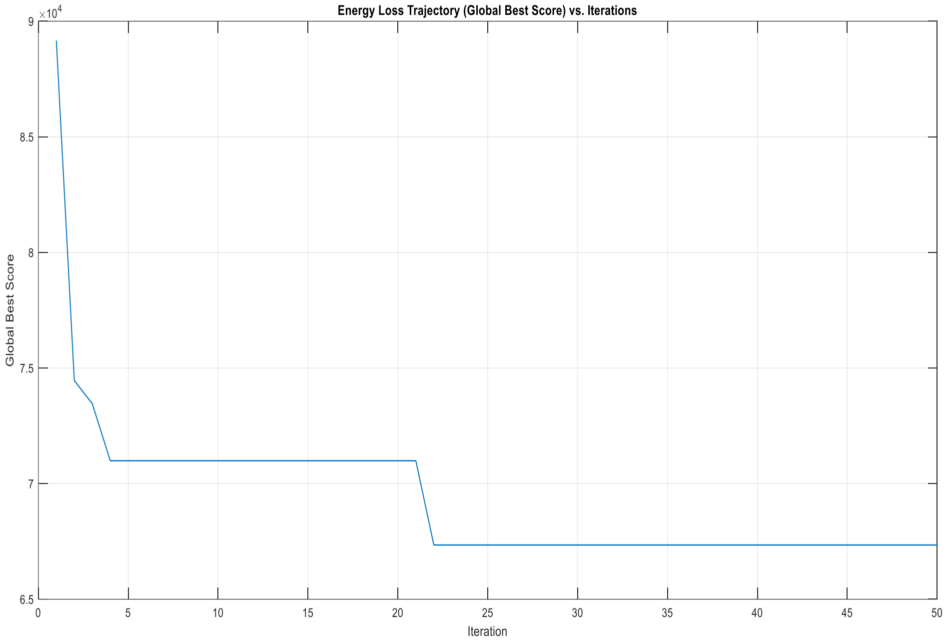

Figure 5. 5 Energy Loss Trajectory for PSO

5.3.6 Best battery Efficiency for PSO

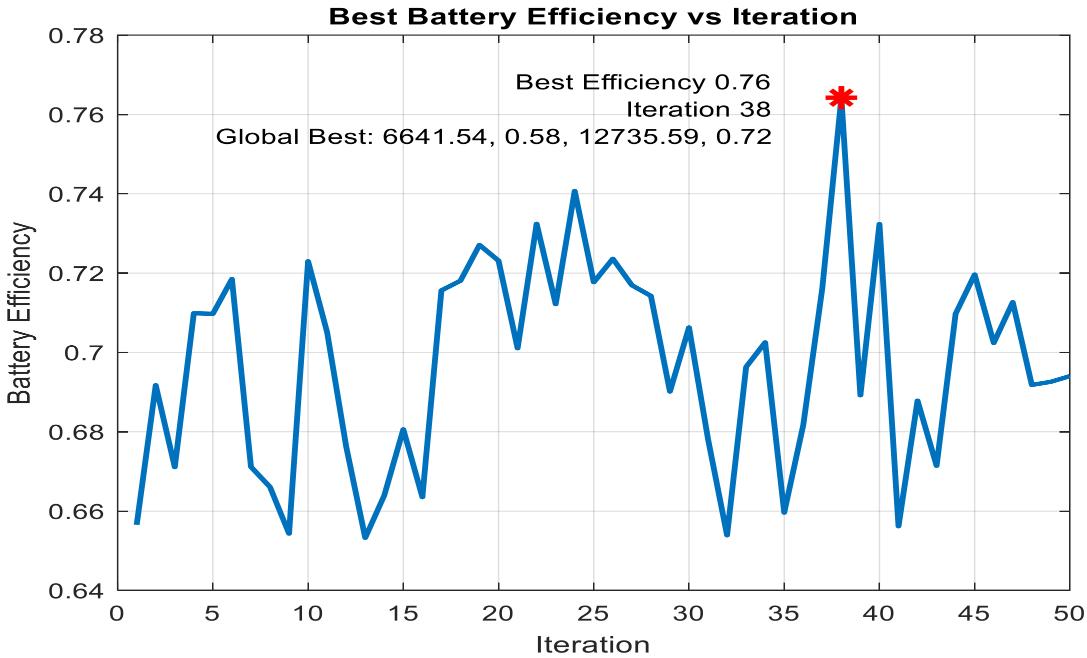

Figure 5. 6 Best Battery Performance for PSO

The total energy loss trajectory in Figure 5.5 illustrates the convergence behavior of the PSO algorithm over 50 iterations. The total loss decreases significantly during the initial iterations, indicating rapid convergence of the algorithm. The trajectory stabilizes at iteration 38 with a minimum energy loss of 9851.56 W. At iteration 38 the stabilization point is marked, indicating where the algorithm converges to a steady solution efficiently. Also, figure 5.6 represents the best battery performance at iteration 38 is 76%.

Table 5. 2 Global Score and Position for PSO

1.	Global Best Score	9851.56 W
2.	Global Best Position	[6641.53 0.5814 12735.59 0.7183]

Table 5.2 represents the Particle Swarm Optimization (PSO) algorithm that achieves a global best score of 9851.56 W, representing the lowest energy loss recorded during the optimization. The corresponding global best position is identified as [6641.53, 0.5814, 12735.59, 0.7183], which signifies the set of control parameters, including battery efficiency, soc value, load distribution, and voltage regulation settings that yield the most efficient performance of the system.

Table 5. 3 Output Parameters of PSO Algorithm

Iteration	Parameters	Values
38	E_{loss}	9851.56 W
	η_{bat}	76%
	t_s	0.23 sec
	$\%M_p$	6.2%

Furthermore, the system's settling time (t_s) is recorded at 0.23 seconds, showing a fast dynamic response to changes in load and generation conditions. Additionally, the percentage overshoot ($\%M_p$) is limited to 6.2%, which demonstrates improved stability and reduced voltage fluctuations during transient conditions are shown in Table 5.3. Table 5.4 presents the battery efficiency achieved at different iterations of the optimization process. The Battery's highest efficiency of 76% was recorded at iteration 38.

Table 5. 4 Iteration & Battery Efficiency for PSO

Iteration	Battery Efficiency (%)	Iteration	Battery Efficiency (%)
1	0.66	26	0.71
2	0.62	27	0.70
3	0.69	28	0.67
4	0.69	29	0.71
5	0.64	30	0.68
6	0.65	31	0.71
7	0.72	32	0.66
8	0.73	33	0.71
9	0.71	34	0.67
10	0.71	35	0.68
11	0.71	36	0.65
12	0.72	37	0.73
13	0.69	**38**	**0.76**
14	0.67	39	0.68
15	0.66	40	0.69
16	0.74	41	0.69
17	0.71	42	0.71
18	0.72	43	0.72
19	0.69	44	0.70
20	0.70	45	0.62
21	0.70	46	0.62
22	0.72	47	0.68
23	0.71	48	0.73
24	0.73	49	0.68
25	0.72	50	0.68

5.4 Artificial Bee Colony Optimization (ABC)

The Artificial Bee Colony (ABC) algorithm is a nature-inspired optimization technique based on the foraging behavior of honeybees. It effectively explores and exploits the solution space to find optimal or near-optimal solutions for complex optimization problems. Table 5.12 shows the parameters of the ABC algorithm.

5.4.1 Implementation of ABC Algorithm

ABC algorithm, influenced by honey bee behaviour. The algorithm of a honey bee colony can find the best quality food sources in nature with ease. Bee colonies are classified into three types according to their foraging ability: Employed bees, Onlooker bees, and Scout bees. Table 5.5 represents ABC Parameters and Figure 5.7 shows the Flowchart of ABC.

5.4.2 Parameters of ABC Algorithms

Table 5. 5 ABC Parameters

Parameters	Value
No. of Iteration	50
No. of Bees	100

5.4.3 Flowchart and Procedure of ABC Algorithm

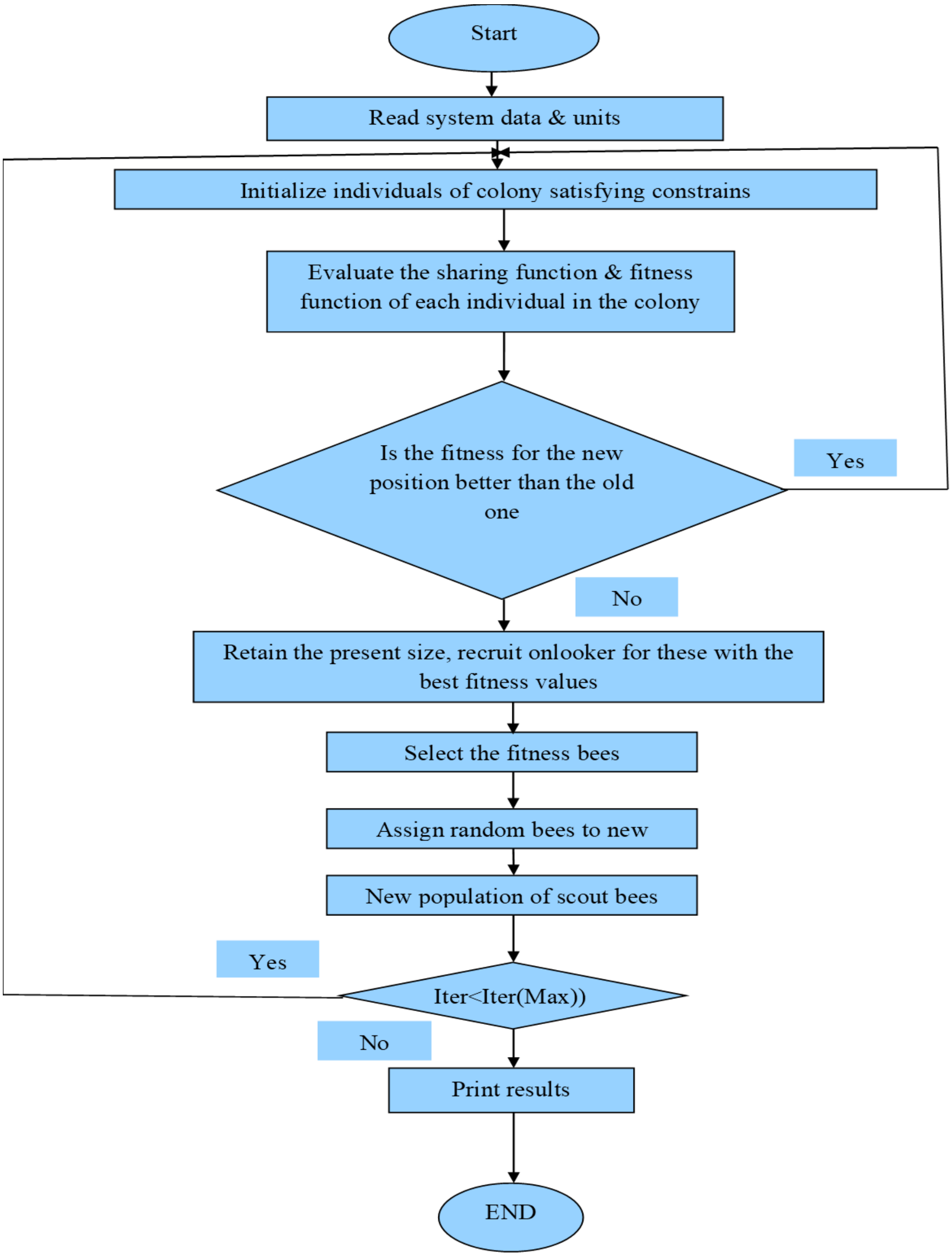

Figure 5. 7 Flowchart of ABC

Below is an overview of the implementation steps of ABC in DC microgrid:

Step 1: Initialization

ABC Parameters: Number of bees, iterations, and the limit for scout bees.

Step 2: Randomly initialize positions for bees

(battery power, battery SOC, supercapacitor power, Supercapacitor SOC).

$$P_b^i = rand(n_{bees}, 1) \times (P_b^{max} - P_b^{min}) + P_b^{min}$$

$$SOC_b^i = rand(n_{bees}, 1) \times (SOC_b^{max} - SOC_b^{min}) + SOC_b^{min}$$

Step 3: Evaluate the Objective Function for all Bees

Update personal best positions and scores

$$f(X^i) < f(Pbest_i) then\ Pbest_i = X^i$$

Update global best position and score

$$\mathrm{f(X^i) < f(Gbest_i)\ then\ Gbest_i = X^i}$$

Step 4: Employed Bee Phase

Select a random bee:

$$j \neq i$$

Generate a new position:

$$X_{new} = X^i + \emptyset \times (X^i - X^j); \tag{4.9}$$

Where, $\emptyset$ is a random number in [-1,1].

Apply boundary constraints:

$$P_b^{min} \leq P_b^{new} \leq P_b^{max}$$

Update if they find a better solution.

$$f(x_{new}) < f(x^i)\ then\ x^i = x_{new}$$

Step 5: Onlooker Bee Phase:

Calculate fitness values and probabilities based on fitness:

$$fitness^i = \frac{1}{1 + f(x^i)}$$

Select new positions based on probabilities:

$$Probability^i \ = \ \frac{fitness^i}{\sum_{k=1}^{n_{bess}} fitness^i} \tag{4.10}$$

Update positions if they find a better solution

Step 6: Scout Bee Phase:

Bees reset their positions if they haven't found improvements after a set number of trials.

$$counter_i \ > \ limit\ then\ reinitialize\ x^i$$

Update and Store Metrics:

- Calculate energy loss and battery efficiency for the current iteration
- Store best efficiencies, energy losses, SOC profiles, and power profiles

5.4.4 Real-Time 3D Visualization ABC Results

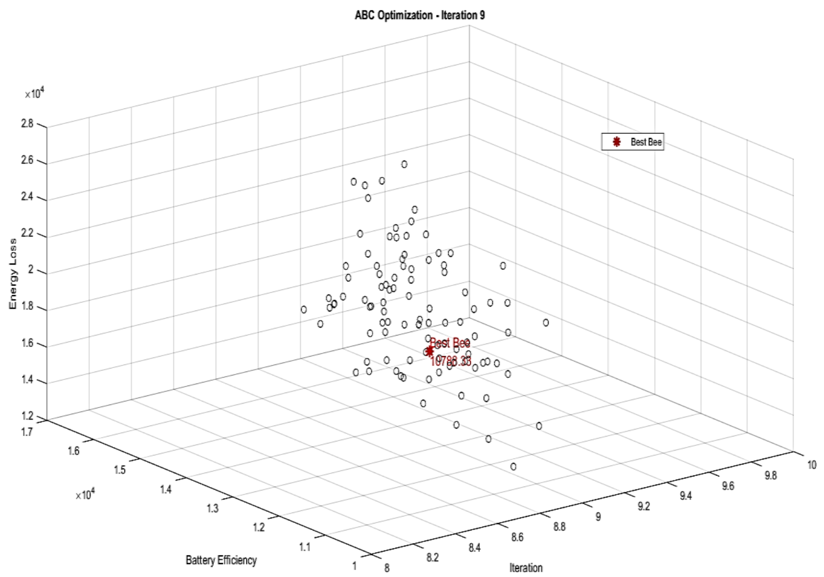

Figure 5. 8 ABC Optimization at Iteration 9

Iteration 9

Best Bee Position: [6323.07581953068 0.309108216981712 10234.117072585 0.604646137690806] Best Bee Score: 10788.33 W

Battery Efficiency = 62%

Figures 5.8 illustrate the optimization process of the Artificial Bee Colony (ABC) algorithm applied to a DC microgrid, focusing on minimizing energy loss and maximizing battery efficiency over successive iterations. In the first figure, representing iteration 9, the algorithm is in its exploratory phase, with the swarm of artificial bees dispersed across the search space. The "Best Bee" achieves an energy loss of 10,788.33 W and a battery efficiency of 62%, with its position defined as [6323.08,0.3091,10234.12,0.6046][6323.08, 0.3091, 10234.12, 0.6046][6323.08,0.3091,10234.12,0.6046].

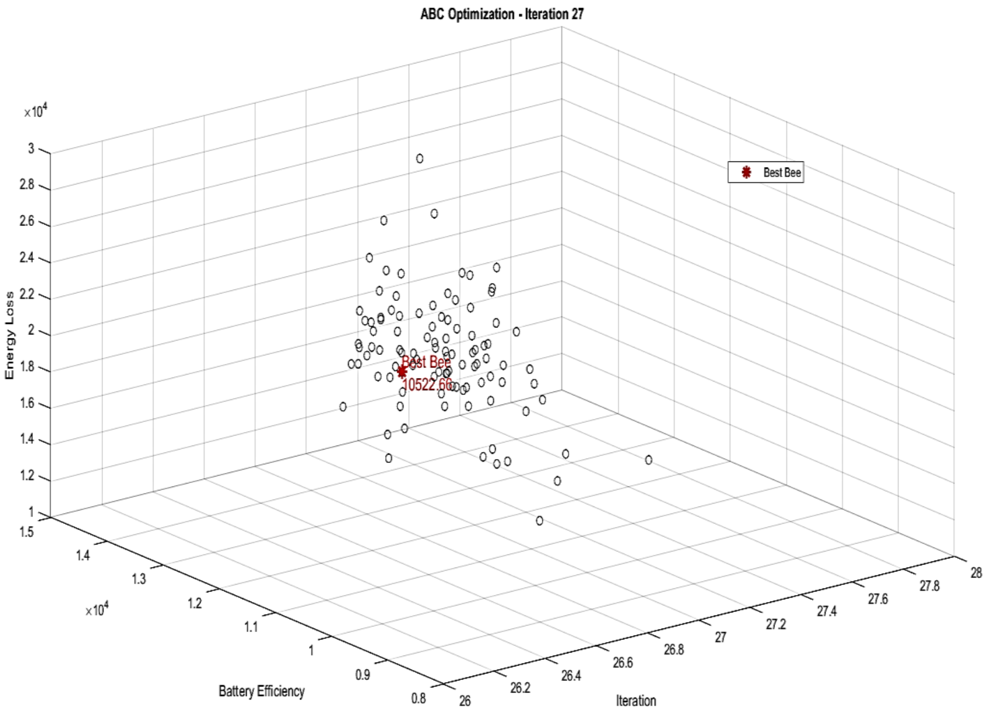

Figure 5. 9 ABC Optimization at Iteration 27

Iteration 27

Best Bee Position: [3302.5302903043 0.508320438200315 4193.99983697547 0.857627353761112] Best Bee Score: 10522.66

Best battery efficiency achieved: 69%

This scattered distribution indicates the algorithm's focus on exploring diverse regions to locate promising solutions. By contrast, the second figure 5.9, corresponding to iteration 27, reflects the algorithm's progression into the exploitation phase. The swarm has converged toward the optimal region, as seen in the tighter clustering of solutions. At this stage, the "Best Bee" achieves an improved energy loss of 10,522.66 W and a battery efficiency of 69%, with its position given as [3302.53,0.5083,4194.00,0.8576][3302.53, 0.5083, 4194.00, 0.8576][3302.53,0.5083,4194.00,0.8576].

5.4.5 Total Energy Loss Trajectory for ABC

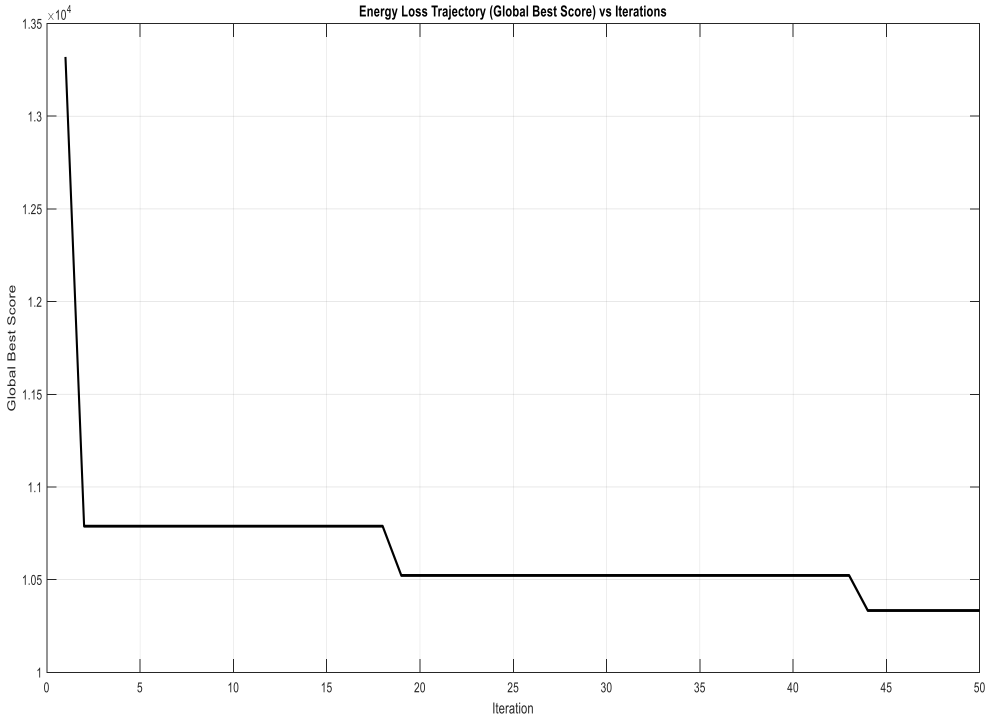

Figure 5. 10 Energy Loss Trajectory for ABC

5.4.6 Best Battery Efficiency for ABC

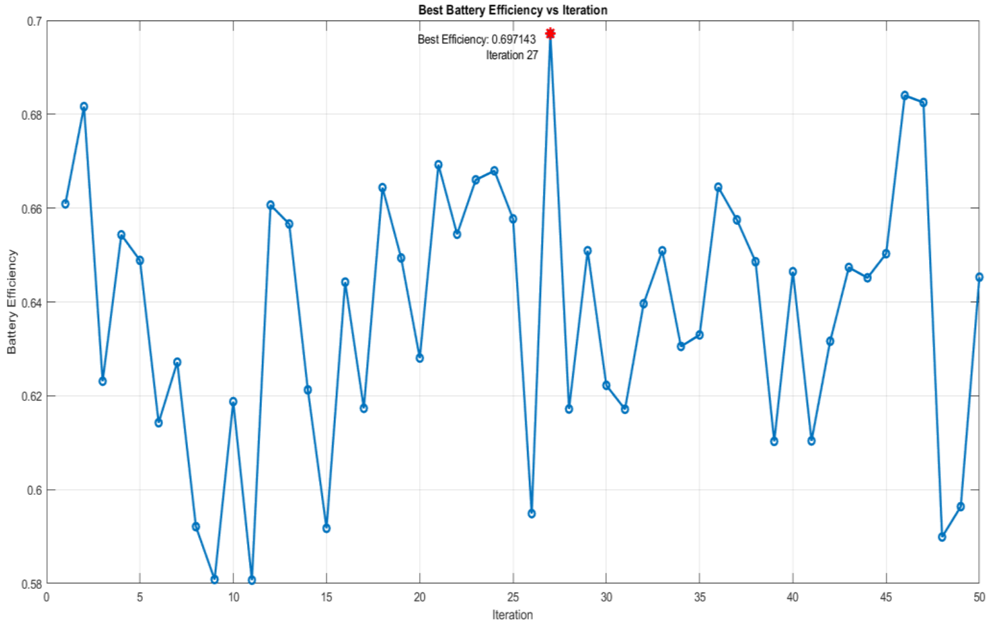

Figure 5. 11 Best Battery Performance for ABC

The total energy loss trajectory in Figures 5.10 and 5.11 illustrates the ABC algorithm's convergence behavior over 50 iterations in optimizing the DC microgrid's performance. Figure 5.10 shows that energy loss decreases significantly during the initial iterations, demonstrating the algorithm's rapid convergence. By iteration 27, the optimization stabilizes, achieving a minimum energy loss of 10,522.66 W, which marks the point of steady convergence to an optimal solution. This stabilization indicates the algorithm's efficiency in refining solutions as the search progresses.

Similarly, Figure 5.11 highlights the best battery performance achieved at iteration 27, reaching an efficiency of 69%. This result underscores the ABC algorithm's capability to enhance the DC microgrid's operational parameters, balancing energy efficiency and minimizing energy losses.

Table 5. 6 Global Score and Position for ABC algorithm

1.	Global Best Score	10522.66 W
2.	Global Best Position	[3302.53 0.50832 4193.99 0.85762]

Table 5. 7 Output Parameters of ABC Algorithm

Iteration	Parameters	Values
27	E_{loss}	10522.66 W
	η_{bat}	69%
	t_s	0.4 sec
	$\%M_p$	5.95%

Table 5.6 presents the performance of the Artificial Bee Colony (ABC) algorithm, which achieves a global best score of 10,522.66 W, representing the lowest energy loss recorded during the optimization process. The corresponding global best position is identified as [3302.53,0.5083,4194.00,0.8576] signifying the optimized control parameters for the DC microgrid. These parameters include battery efficiency, state of charge (SOC) levels, load distribution, and voltage regulation settings. Collectively, these configurations enable the DC microgrid to operate with maximum energy efficiency, minimizing losses while ensuring optimal system performance.Table 5.7 presents the output parameters of the Artificial Bee Colony (ABC) algorithm. The battery efficiency is optimized to 69%, reflecting improved energy storage performance. The settling time is recorded at 0.4 seconds, indicating a rapid response in the system's dynamic performance. Additionally, the percentage overshoot is measured at 5.95%, showing a well-controlled transient behavior. These results underscore the ABC algorithm's ability to optimize critical parameters, ensuring energy efficiency and stability within the DC microgrid. Table 5.8 summarizes the battery efficiency values achieved by the ABC algorithm over 50 iterations. The highest efficiency of 69% is achieved at iteration 27, reflecting the algorithm's ability to refine and improve system parameters.

Table 5. 8 Iteration & Battery Efficiency for ABC

Iteration	Battery Efficiency	Iteration	Battery Efficiency
1	0.66	26	0.59
2	0.68	**27**	**0.69**
3	0.62	28	0.61
4	0.65	29	0.65
5	0.64	30	0.62
6	0.61	31	0.61
7	0.62	32	0.63
8	0.59	33	0.65
9	0.58	34	0.63
10	0.61	35	0.63
11	0.58	36	0.66
12	0.66	37	0.65
13	0.65	38	0.64
14	0.62	39	0.61
15	0.59	40	0.64
16	0.64	41	0.61
17	0.61	42	0.63
18	0.66	43	0.64
19	0.64	44	0.64
20	0.62	45	0.65
21	0.66	46	0.68
22	0.65	47	0.68
23	0.66	48	0.58
24	0.66	49	0.59
25	0.65	50	0.64

5.5 Grey Wolf Optimization (GWO)

5.5.1 Implementation of GWO Algorithm

Grey Wolf Optimizer (GWO) is a population-based metaheuristic algorithm inspired by the social structure and hunting tactics of grey wolves. The algorithm involves four types of grey wolves: alpha, beta, delta, and omega, which represent different ranks in the hierarchy.

- Hierarchy of Grey Wolves

  Grey wolves are divided into four categories based on their social hierarchy:
  Alpha (α): The leader of the pack.
  Beta (β): The second in command, assists the alpha.
  Delta (δ): The third level, takes commands from α\alphaα and β\betaβ.
  Omega (ω): The lowest ranking, follows the commands from all other levels.

5.5.2 Parameters of GWO Algorithm

Table 5. 9 GWO Parameters

Parameters	Value
No. of Iteration	50
No. of Wolves	100

The optimization process mimics the hunting process of grey wolves, where alpha, beta, and delta wolves guide the search while omega wolves follow them. Table 5.9 represents the GWO parameters and Figure 5.12 shows the flowchart of the GWO algorithm.

5.5.3 Flowchart and Procedure of GWO

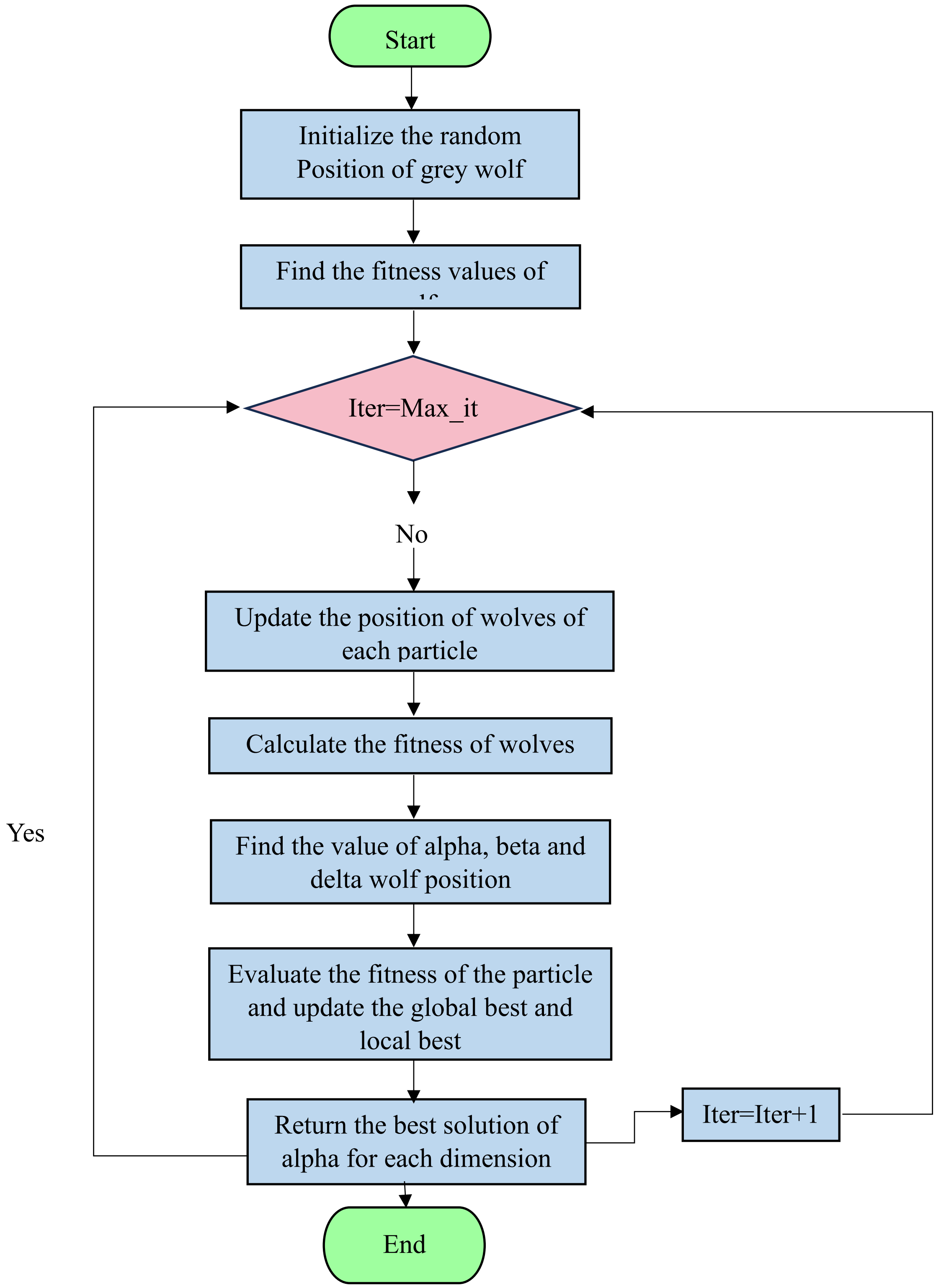

Figure 5. 12 Flowchart of GWO

Below is an overview of the implementation steps of GWO in DC microgrid:

Step 1: Initialization

- Define the parameters of the GWO:

  (Number of wolves, Maximum number of iterations)

- Set the search space boundaries for each variable:

  Battery power (P_b): Minimum (P_b_min) and Maximum (P_b_max).

  Battery SOC (SOC_b): Minimum (SOC_b_min) and Maximum (SOC_b_max).

  Supercapacitor power (P_sc): Minimum and Maximum values.

  Supercapacitor SOC (SOC_sc): Minimum and Maximum values.

- Initialize the wolves' positions randomly within the defined search space.

Step 2: Evaluate Fitness

Calculate the fitness (objective function value) for each wolf to measure its solution quality.

Step 3: Identify Alpha, Beta, and Delta Wolves

Alpha Wolf: The best solution with the highest fitness.
Beta Wolf: The second-best solution.
Delta Wolf: The third-best solution.
The rest are considered "Omega Wolves."

Step 4: Update Positions

Adjust each wolf's position based on the Alpha, Beta, and Delta wolves:

$$\vec{X}(t+1) = \vec{X}_P(t) - \vec{A} \cdot \vec{D}\mathrm{C}$$

Distance Calculation:

$$\vec{D} = \left|\vec{C} \cdot \vec{X}(t) - \vec{X}(t)\right|$$

$\vec{X}(t+1)$ is the position vector of a grey wolf.

$\vec{X}_P(t)$ is the position vector of the prey.

$\vec{A}$ and $\vec{C}$ are coefficient vectors.

Simulate hunting behaviour by moving closer to these best wolves while considering exploration and exploitation.

Step 5: Boundary Handling

Ensure wolves remain within the defined bounds of the problem.

Xi,j=min(max(Xi,j,Xjmin),Xjmax)

Step 6: Convergence

Repeat the evaluation and position update for the specified number of iterations or until the solutions converge.

Step 7: Output the Best Solution

The position of the Alpha Wolf at the end of the optimization represents the optimal solution.

5.5.4 Real-Time 3D Visualization in GWO Results

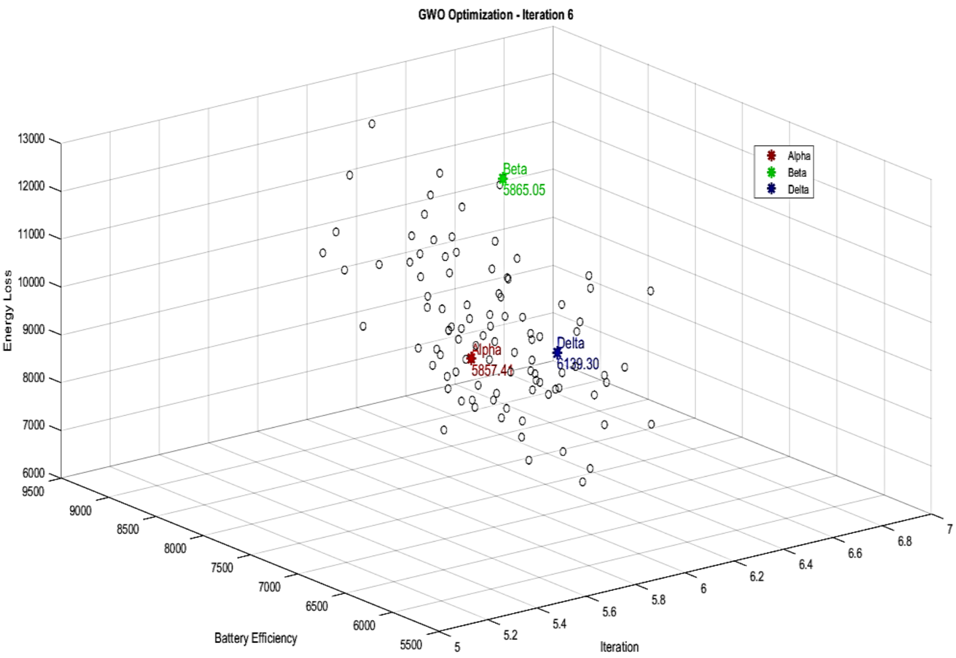

Figure 5. 13 GWO Optimization at Iteration 6

Iteration: 6

Alpha: 5857.46W

Beta: 5865.05W

Delta: 6129.30W

Best battery efficiency achieved: 74%

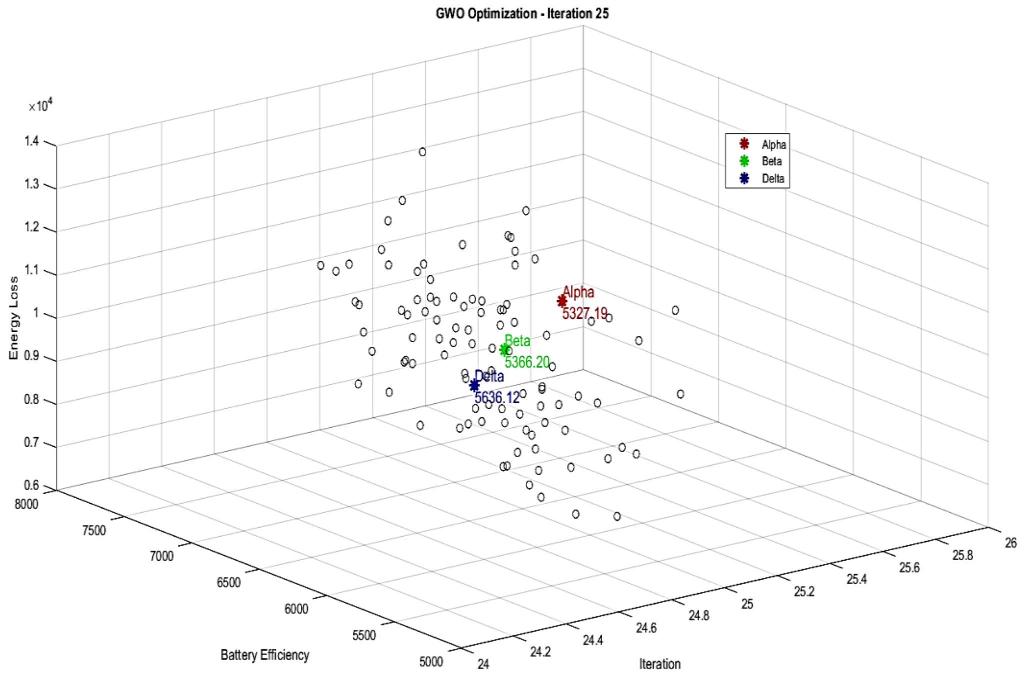

Figure 5. 14 GWO Optimization at Iteration 25

Iteration: 25

Alpha: 5327.19W

Beta: 5366.20W

Delta: 5636.12W

Best battery efficiency achieved: 81%

Figure 5.13 represents the optimization process at Iteration 6, where the algorithm is in the exploratory phase. The scatter plot illustrates the relationship between Battery Efficiency, Iteration, and Energy Loss. At this stage, search space indicates a focus on exploring diverse regions. The three best solutions are Alpha, Beta, and Delta, with Alpha achieving the lowest Energy Loss of approximately 5857.46W, followed by Beta with 6865.05W, and Delta with 8139.30W. The broad distribution of points suggests that the process primarily identifies and evaluates potential regions of interest without significant convergence.

Figure 5.14 represents the optimization process at Iteration 25, showing a clear transition to the exploitation phase. The plot shows the same variables as in Figure 5.27 but with a higher degree of convergence. The Alpha solution, representing the best outcome, achieves a significantly reduced Energy Loss of approximately 5327.19W, while Beta and Delta follow closely with Energy Loss values of 5366.20W and 5636.12W, respectively. The clustering of solutions demonstrates that the optimization process has effectively narrowed down the search to high-quality solutions, with reduced Energy Loss and improved performance.

5.5.5 Total Energy Loss Trajectory

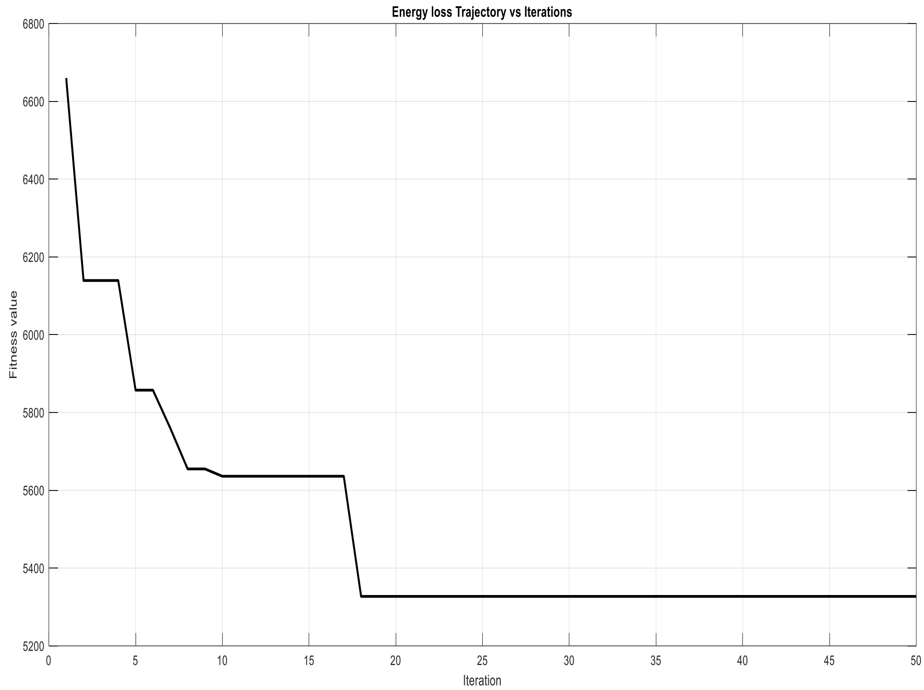

Figure 5. 15 Energy Loss Trajectory for GWO

The trajectory depicted in Figure 5.15 represents the convergence behavior of the optimization process over 50 iterations. Initially, a sharp decrease in fitness value is observed, indicating efficient exploration of the search space and rapid identification of better solutions. Between Iterations 10 and 20, the rate of improvement diminishes, signifying a transition to exploitation as the algorithm refines promising regions. Beyond Iteration 20, the fitness value stabilizes, reflecting convergence to an optimal or near-optimal solution. This trajectory highlights the algorithm's ability to balance exploration and exploitation effectively, ensuring progressive optimization. Also, figure 5.16 represents the best battery performance achieved at iteration 25, reaching an efficiency of 81%. This result underscores the GWO algorithm's capability to enhance the DC microgrid's operational parameters, balancing energy efficiency and minimizing energy losses.

5.5.6 Best Battery Efficiency for ABC

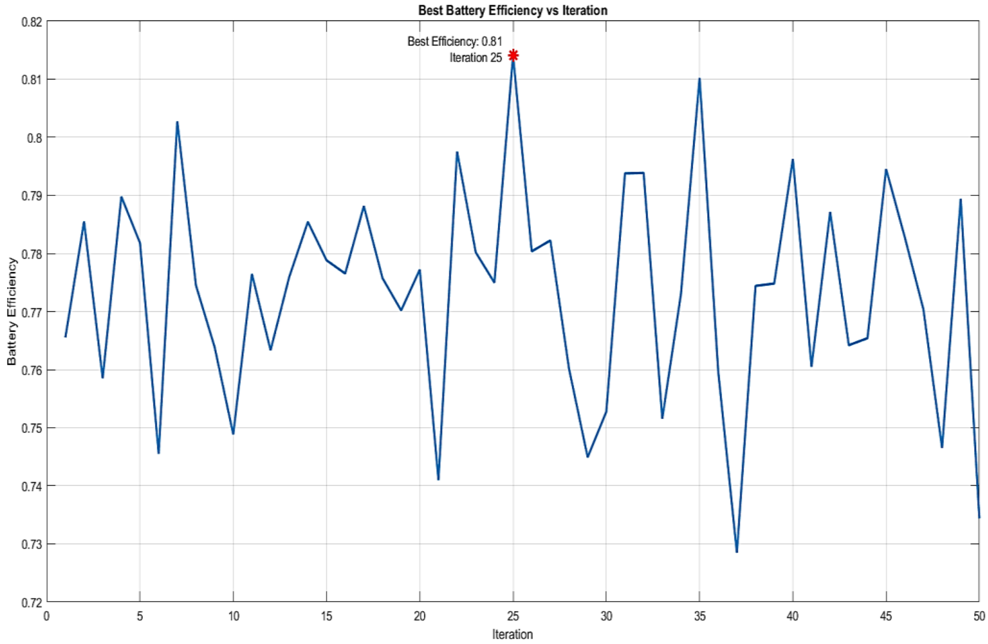

Figure 5. 16 Best battery Performance for GWO

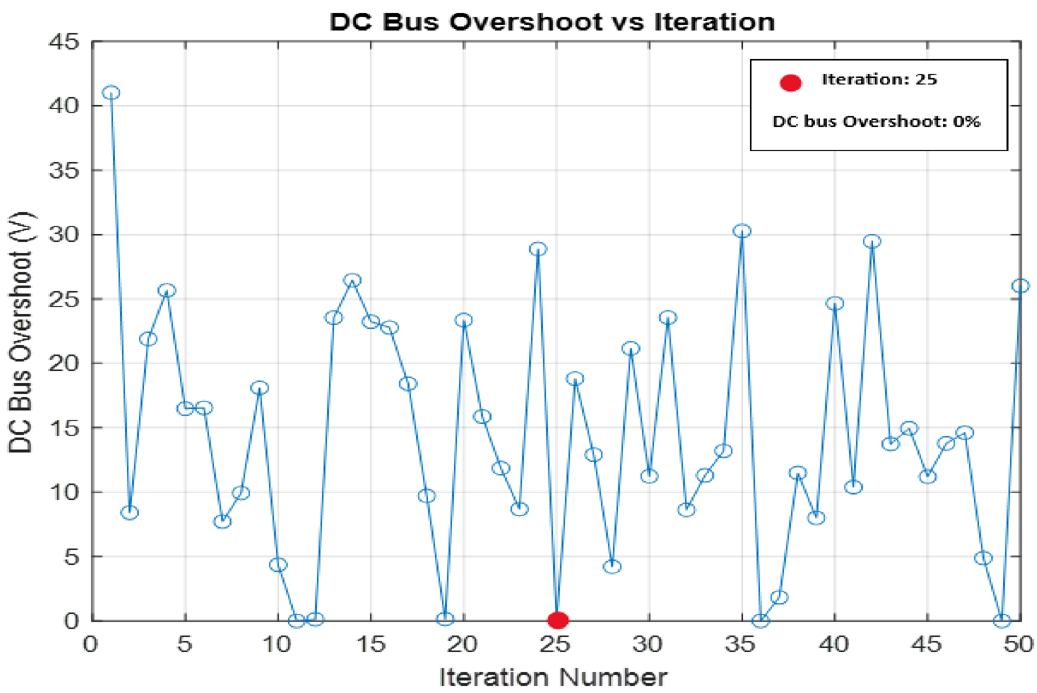

Figure 5. 17 DC bus Overshoot performance for GWO

Figure 5.17 shows the DC Bus Overshoot (in volts) plotted against the Iteration Number. Initially, the overshoot fluctuates significantly, with high values indicating instability in the system. Over time, the magnitude of these fluctuations reduces, and at Iteration 25, the DC Bus Overshoot achieves a value of 0%, indicating complete elimination of overshoot. This demonstrates the system's stabilization and convergence toward optimal performance after successive iterations.

Table 5. 10 Global Score and Position for GWO

1.	Global Best Score	5327.19 W
2.	Global Best Position	[8183.38 0.3611 5518.53 0.6417]

The global performance of the optimization is summarized with the best score of 5327.19 W, achieved at the global best position [8183.38, 0.3611, 5518.53, 0.6417], as shown in Table 5.10. Table 5.11 outlines the output parameters at Iteration 25, where the best Energy Loss values for Alpha, Beta, and Delta are 5327.19 W, 5366.20 W, and 5636.12 W, respectively. The battery efficiency is 81%, the settling time is 0.1 second, and the percentage overshoot is 0%, indicating stable and efficient system performance. Table 5.12 summarizes the battery efficiency values achieved by the GWO algorithm over 50 iterations. The highest efficiency of 81% is achieved at iteration 25.

Table 5. 11 Output Parameters of GWO Algorithm

Iteration	Parameters	Values
25	E_{loss}	x_α = 5327.19 W x_β = 5366.20 W x_δ=5636.12 W
	η_{bat}	81%
	t_s	0.1 sec
	$\%M_p$	0%

Table 5. 12 Iteration & Battery Efficiency for GWO

Iteration	Battery Efficiency	Iteration	Battery Efficiency
1	0.76	26	0.78
2	0.78	27	0.78
3	0.75	28	0.76
4	0.78	29	0.74
5	0.78	30	0.75
6	0.74	31	0.79
7	0.80	32	0.79
8	0.77	33	0.75
9	0.76	34	0.77
10	0.74	35	0.81
11	0.77	36	0.75
12	0.76	37	0.72
13	0.77	38	0.77
14	0.78	39	0.77
15	0.77	40	0.79
16	0.77	41	0.76
17	0.78	42	0.78
18	0.77	43	0.76
19	0.77	44	0.76
20	0.77	45	0.79
21	0.74	46	0.78
22	0.79	47	0.77
23	0.78	48	0.74
24	0.77	49	0.78
25	**0.81**	50	0.73

5.6 Comparative Analysis with Metaheuristic Optimization Algorithms

Table 5. 13 Comparative analysis for Individual algorithm

Algorithm	**Iteration**	**Best Score (W)**	**Best Efficiency (%)**	t_s	$\%M_p$ **(4 sec)**
PSO	38	9851.56W	76%	0.23 sec	6.2%
ABC	27	10522.66W	69%	0.4 sec	5.95%
GWO	25	5327.19W	81%	0.1 sec	0%

Table 5.13 compares three optimization algorithms: PSO, ABC, and GWO based on their performance metrics, including DC bus overshoot. Among them, GWO stands out as the best algorithm due to its superior efficiency of 81%, minimal resource usage, and excellent stability

with no overshoot (0%). It required the fewest iterations (25), had a less settling time of 0.1 seconds, it highly effective for precise and stable optimization.

PSO showed a balanced performance with 38 iterations, a best score of 9851.56W, and an efficiency of 76%. Its fast-settling time of 0.23 seconds and its overshoot of 6.2% suggest aggressive responses and potential convergence on local minima, limiting its suitability for systems requiring high stability. ABC achieved the highest best score of 10522.66W in 27 iterations, with an efficiency of 69%. It demonstrated strong exploration and exploitation, balancing speed and transient control with an overshoot of 5.95%. Despite a slightly higher settling time of 0.4 seconds.

In summary, GWO's high efficiency, stability, and less overshoot make it the optimal choice for robust DC bus optimization. ABC offers a balanced trade-off with moderate overshoot and good performance. At the same time, PSO, though fast, may be less reliable due to its higher overshoot and potential to converge on local minima.

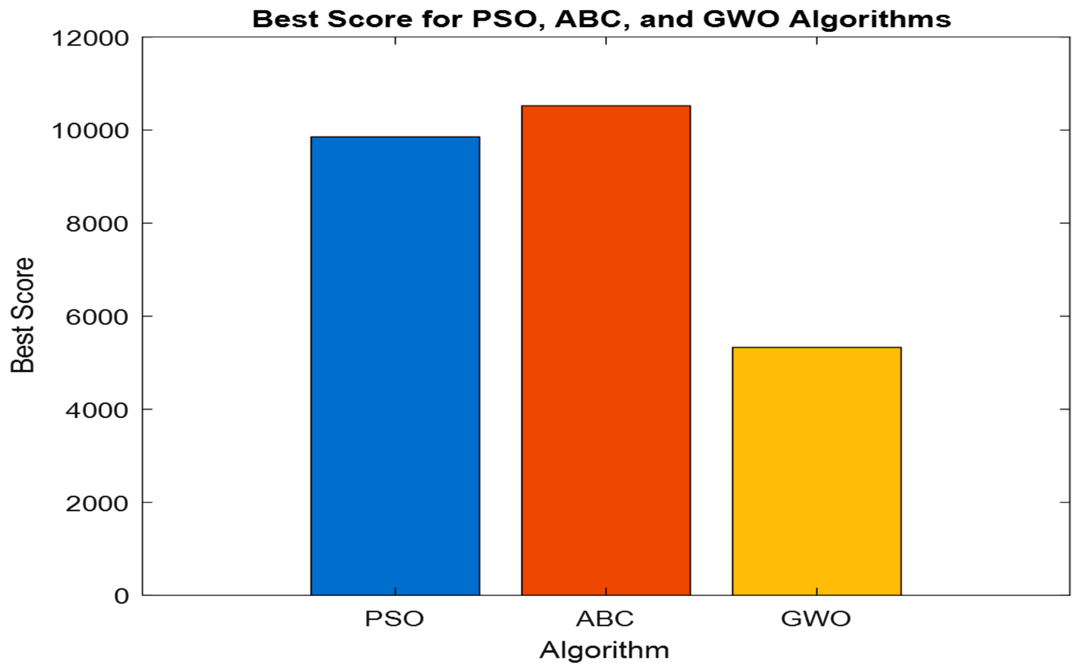

Figure 5. 18 Best Score for PSO, ABC, and GWO Algorithms

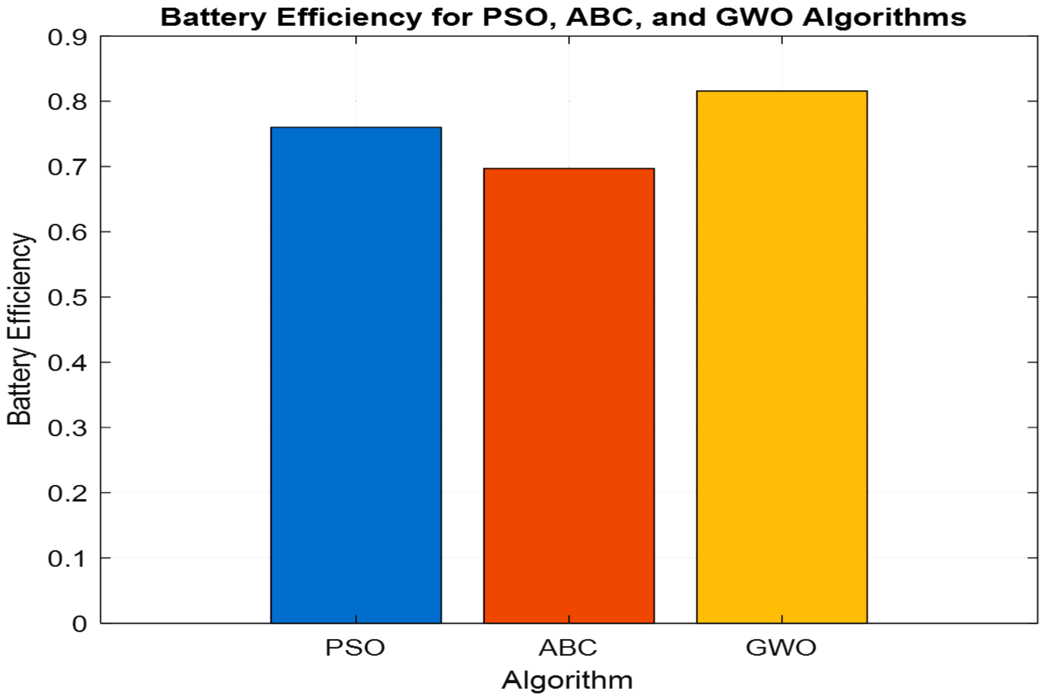

Figure 5. 19 Battery Efficiency for PSO, ABC, and GWO Algorithms

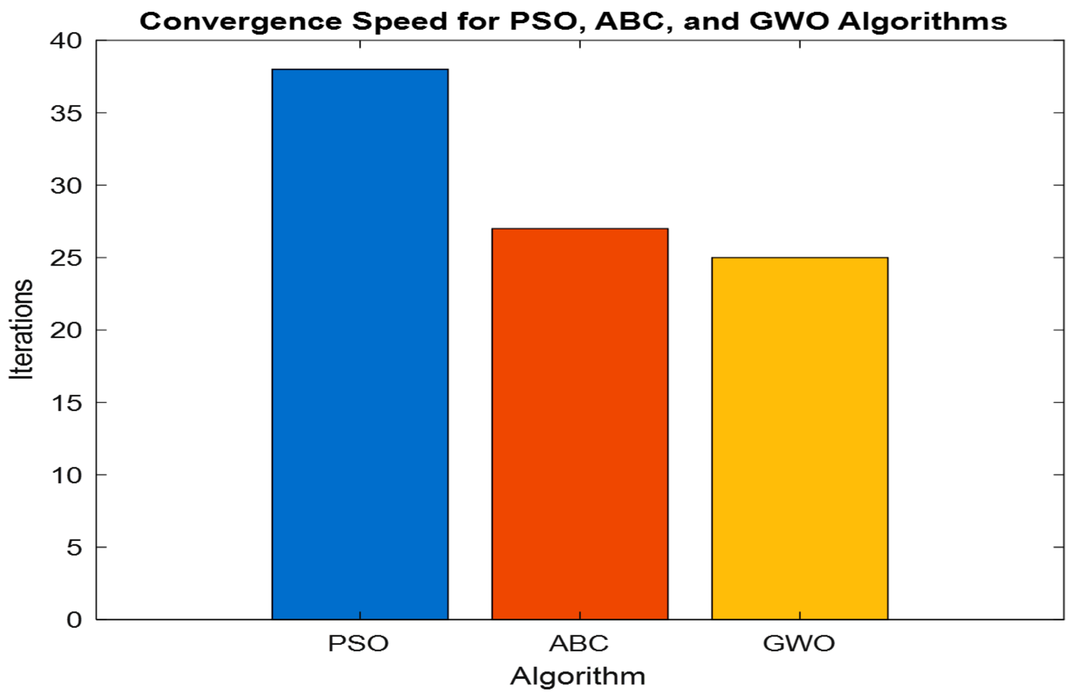

Figure 5. 20 Convergence Speed for PSO, ABC, and GWO Algorithms

Figures 5.18,5.19 and 5.20 evaluate the performance of PSO, ABC, and GWO algorithms based on power loss (best score), efficiency, and convergence, to minimize voltage fluctuations in the

DC bus voltage. GWO satisfied this objective by achieving the highest efficiency (81%), fastest convergence (25 iterations), and no overshoot. Its performance highlights exceptional stability and resource optimization, making it the most suitable algorithm for minimizing DC bus voltage fluctuations. In comparison, ABC, despite having the highest power loss (10522.66W), showed moderate efficiency (69%) and balanced convergence (27 iterations) but was less effective in meeting the objective. While relatively fast with 38 iterations and moderate efficiency (76%), PSO exhibited higher power loss (9851.56W) and significant overshoot, resulting in greater voltage fluctuations. Overall, GWO satisfied the objective of minimizing DC bus voltage fluctuations most effectively while ensuring stability and efficiency. Figures 5.21 and 5.22 show the comparison of Peak overshoot and settling time.

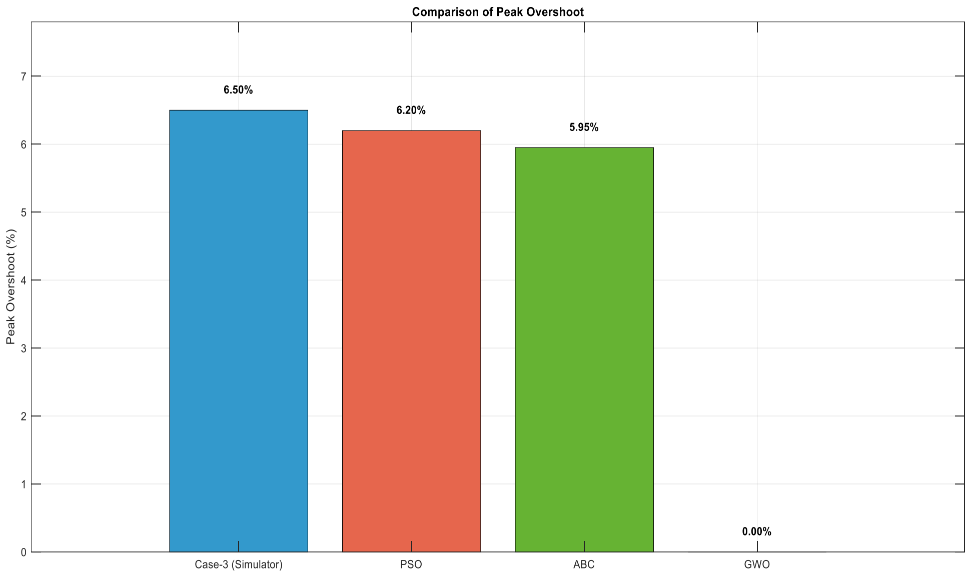

Figure 5. 21 Comparison of Peak Overshoot

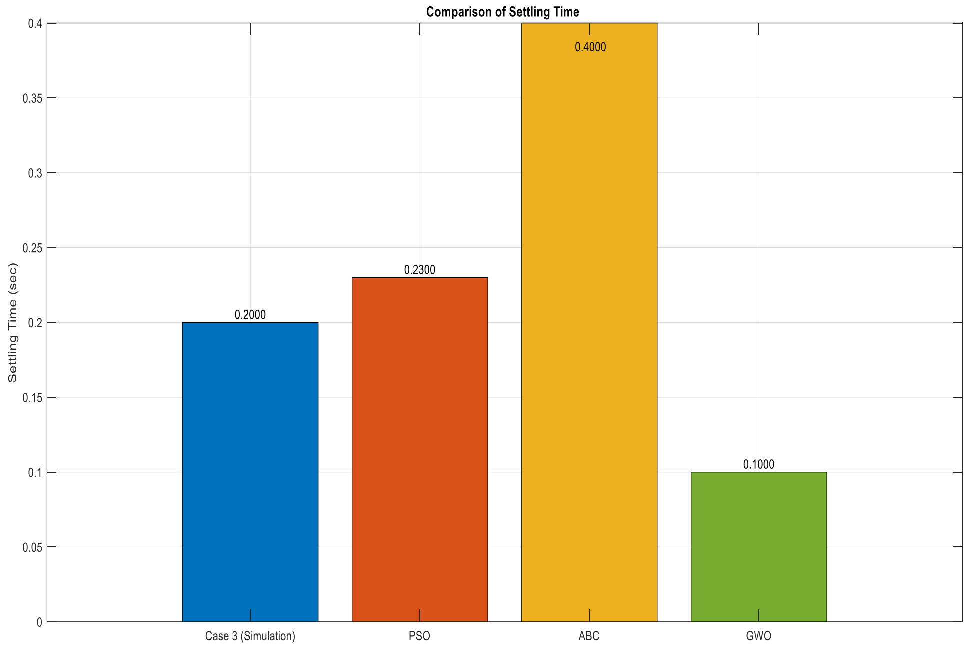

Figure 5. 22 Comparison of Settling time

Table 5. 14 Analysis of Simulation and Optimization Results

Simulation Results			Optimization Results (Case 3)		
	t_s	$\%M_p$	**Algorithms**	t_s	$\%M_p$
Case:1	0.5 sec	9.24%	**PSO**	0.23 sec	6.2%
Case:2	0.4 sec	18.46%	**ABC**	0.4 sec	5.95%
Case:3	0.2 sec	6.5%	**GWO**	0.1 sec	0%

Table 5.14 compares simulation and optimization results for three cases. The simulation results show baseline performance, while optimization techniques (PSO, ABC, and GWO) demonstrate improvements in specific metrics. Each optimization algorithm focuses on reducing computational time and enhancing performance measures, highlighting their effectiveness compared to the simulation outcomes.

Summary

The PSO, ABC, and GWO optimization algorithms effectively enhance the performance of the DC microgrid by minimizing energy losses, improving battery efficiency, and ensuring system stability. PSO achieves a balance between energy loss reduction and rapid, stable system response with low settling time and controlled overshoot. ABC initially explores the solution space before refining solutions for higher efficiency and lower energy losses, optimizing microgrid performance. GWO demonstrates strong performance by combining broad exploration with focused refinement, leading to high efficiency, stability, and rapid convergence without overshoot. Overall, these iterative optimization approaches ensure efficient power management and optimal system parameters.