Comparative Analysis of Particle Swarm Optimization and Salp Swarm Algorithm in improving the Voltage Profile of IEEE 9 Bus System

Lhendup Thinley Dorji*[1], Phuntsho Namgay[1], Karma Tenzin[1], Sreekanth Nethagani[1], Chenga Dorji[1], Chencho Dorji[1]

Electrical and Electronics Engineering, Jigme Namgyel Engineering College, Royal University of Bhutan
E-Mail:05210012.jnec@rub.edu.bt*

Received: 14 April 2025; Revised :9 June 2025; Accepted:10 July 2025: Published: 17 August 2025

Abstract
Voltage stability is a critical aspect of modern power system operation, directly influencing system reliability, operational efficiency, and protection against grid collapse. Voltage instability often arises due to heavy reactive power demand, insufficient local reactive compensation, and weak transmission infrastructure, especially under high loading or fault conditions. To address these challenges, advanced optimization techniques are often employed for reactive power planning and voltage profile enhancement. Hence, a comparative analysis of two nature-inspired metaheuristic algorithms—Particle Swarm Optimization (PSO) and Salp Swarm Algorithm (SSA)—in mitigating voltage instability of the IEEE 9-Bus test system is done. The aim is to compute the optimal placement and sizing of reactive power compensation using shunt capacitors. The objective function is formulated to minimize overall voltage deviation from the nominal value (1 p.u.), and ensure computational efficiency of the algorithms. Both algorithms were evaluated on three main performance metrics: voltage loss minimization, convergence characteristics, and recommended capacitor sizes (MVAr ratings). PSO algorithm showed faster convergence towards near-optimal solutions, making it suitable for time-constrained applications. On the other hand, SSA showed superior performance in recommending more economical MVAr injection sizes. But SSA achieving comparable voltage profile improvements, resulted with slightly higher iteration counts. Original contribution of this work lies in the direct comparison of PSO and SSA for increasing voltage stability under the same system conditions, offering insights into their relative strengths. The significance of this study is its potential to guide power system engineers in selecting appropriate optimization methods based on computational and economic trade-offs. Future research can extend this approach to meshed and large-scale systems, integrating uncertainty in load demand and renewable generation.
Key Words: *Particle Swarm Optimization, Salp Swarm Algorithm, Voltage Stability, IEEE 9 Bus System*

1. INTRODUCTION

Voltage instability is a growing concern in power systems because of networks becoming more complex from renewable energy integration, associated increased implementation of power electronics and increased demand. Maintaining a firm voltage profile ensures power system stability and efficiency with reliable power supply which is essential in modern society. Voltage instability often arises due to excessive reactive power demand, insufficient local compensation, and sudden disturbances such as faults or load changes on long transmission lines. If not addressed promptly, voltage instability can lead to cascading failures, system blackouts, and significant economic losses (Mokred & Wang, 2024).

Power systems are inherently nonlinear due to the interdependence of voltages, currents, and power flows across a large number of buses. Traditional optimization techniques—such as gradient-based or linear programming methods—often fail to converge or become trapped in local optima when modelling such nonlinear, non-convex problems (Ayalew, Hussen, & Pasam, 2019). These limitations have sparked interest in metaheuristic algorithms, which are better suited for navigating complex solution spaces without requiring gradient information.

Among these, Particle Swarm Optimization (PSO) and Salp Swarm Algorithm (SSA) have shown promise in various power system applications. PSO is a population-based stochastic optimization technique inspired by the social behavior of bird flocking, known for its fast convergence. SSA, a more recent algorithm inspired by salp chain dynamics in oceanic swarming, is appreciated for its balance between exploration and exploitation in high-dimensional search spaces. These techniques are particularly effective in tuning control variables such as reactive power injections (MVAr compensation), generator voltage setpoints, and tap-changing transformers to maintain voltage stability (Valencia-Rivera et al., 2024).

To benchmark these algorithms, the IEEE 9-bus test system is selected for its simplicity, yet representative behavior of real-world transmission networks. It provides a clear framework to study voltage stability issues under different loading and compensation scenarios, making it a widely accepted standard in voltage control studies.

Despite a large body of work on voltage stability and reactive power optimization, few comparative studies have systematically evaluated PSO and SSA under the same system and objective function. The knowledge gap lies in understanding their relative performance in terms of voltage profile enhancement, computational efficiency, and optimal reactive compensation sizing.

This study aims to fill this gap by formulating a voltage deviation minimization problem and solving it using both PSO and SSA under identical test conditions. The central hypothesis is that while both algorithms can enhance voltage stability, SSA may yield more economical compensation sizes, whereas PSO may offer faster convergence. The results of this study are expected to provide insights into selecting appropriate metaheuristic techniques for voltage stability enhancement in practical power system applications.

2. METHODOLOGY

2.1 Overview of Approach

This study uses both MATLAB and DIgSILENT PowerFactory to simulate and optimize the IEEE 9-bus power system. MATLAB was used to implement and run the optimization algorithms—PSO and SSA—while DIgSILENT provided the platform for high-fidelity power flow simulations, network modelling, and voltage profile visualizations.

2.2 Test System: IEEE 9-Bus

The IEEE 9-bus test system consists of:

- 3 generator buses (Slack: Bus 1; PV: Buses 2 and 3)
- 6 load buses (Buses 4–9)
- Standard system base: 100 MVA, 60 Hz
- Line and transformer impedances as per IEEE standard test case data.
- Load types: Predominantly inductive, contributing to lagging power factor and voltage drops.

A summary of bus data, line parameters, and load/generation distribution is modelled in Fig 3.

2.3 Load Flow Analysis

Power flow analysis is performed using Newton-Raphson method in DIgSILENT. Initial bus voltages, real/reactive power injections, and line flows are extracted and exported to MATLAB for optimization.

2.4 Objective Function

The objective is to minimize the total voltage deviation from 1 p.u. across all load buses:
Minimize

$$F = \sum_{i=1}^{n} |V_i - 1.0\,|$$

Subject to:

- $0.95 \leq V_i \leq 1.05$ p.u. (voltage limits)
- $0 \leq Q_{cap,i} \leq Q_{max}$ (capacitor rating constraints)

Where:

- V_i : Voltage magnitude at bus *i*
- $Q_{cap,i}$: MVAr compensation at bus *i*
- Q_{max}: Maximum allowable MVAr (set to 10 MVAr in this study)

2.5 MVAr Sizing Logic

Capacitor sizes are treated as continuous optimization variables. The reactive power injection improves local voltage based on the simplified relation:

$$\Delta V \; \alpha \; \frac{Q_{cap}}{V}$$

The algorithm iteratively adjusts MVAr values to bring voltages within optimal range using the defined objective function.

2.6 Simulation Setup

- **DIgSILENT**: Used to model the IEEE 9-bus system and perform base case load flow.
- **MATLAB**: Used for running optimization loops, applying capacitor injections, after obtaining loadflow results in DIgSILENT
- **Integration Flow**:
    1. Initial load flow in DIgSILENT
    2. Export line data and bus data to MATLAB
    3. Run PSO/SSA optimization

4. Reconfigure optimized MVAr values to DIgSILENT for final voltage analysis

3. OPTIMIZATION TECHNIQUES

3.1 Particle Swarm Optimization

PSO is a swarm-based optimization technique mimicking the communal actions of birds and fish. Every particle in the swarm represents a personal solution and the best one is updated as the global solution of the overall swarm. The system is initialized by random particle positions and velocities, where updates are administered by cognitive and social coefficients. The inertia weight parameter is attuned to balance exploration and exploitation. The global optimal setting is calculated using velocity and position updates of each particle across the iterations. Though PSO is acknowledged for its quicker convergence, occasionally local optima occur from premature convergence (Houssein, E. H., et. al, 2021).

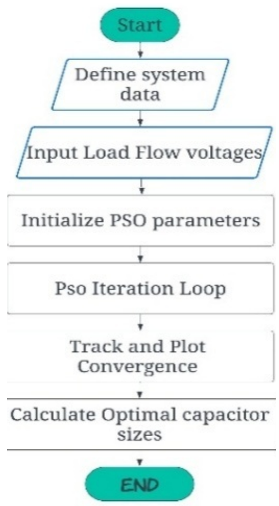

Fig.1: *Flowchart for Particle Swarm Optmization*

3.2 Salp Swarm Algorithm

SSA imitates the behavior of individual salps in the swarm by classifying as leaders and followers. Leaders explore the search space and consequently followers update positions. SSA delivers a sensible exploration-to-exploitation balance, decreasing chances of local optima entrapment. Contrasting PSO, SSA dynamically fine-tunes its search behavior from swarm interactions besides environmental conditions, providing more resilience to stagnation in intricate solution spaces (Abualigah, L., et. al, 2021).

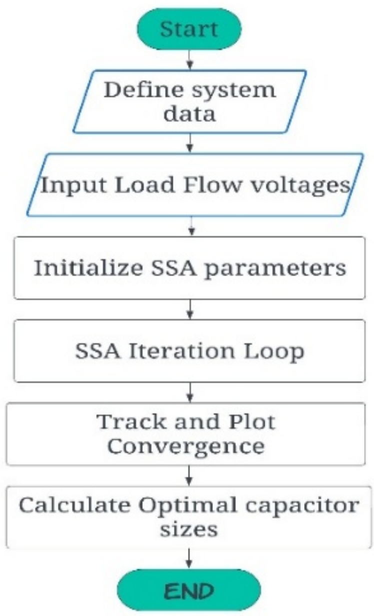

Fig.2: *Flowchart for Salp Swarm Optmization*

4. FORMULAE

Particle Swarm Optimization

$$v_i(t+1) = \omega . v_i(t) + c_1 . r_1\big(pBest_i - x_i(t)\big) + c_2 . r_2\big(gBest_i - x_i(t)\big) \qquad (1a)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \qquad (1b)$$

Particle Swarm Optimization (PSO)

- **Swarm Size**: 30 particles
- **Max Iterations**: 100
- **Inertia Weight**: w=0.7
- **Acceleration Coefficients**: c1=c2=1.5

Steps:

1. Initialize particle positions (MVAr values) and velocities.
2. Evaluate fitness using voltage deviation.
3. Update particle velocities and positions using above formulae.
4. Apply power flow to get new voltages, update fitness.
5. Repeat until convergence.

- **Salp Swarm Algortihm**

$$Position_{Leader}(i) = Current\ Position_{Leader}(i) + rand() \times (ub - Ib) \times (Target_{position} - Position_{Leader}(i)) \qquad (2a)$$

Where:

rand()= 0 −1

ub−Ib =1.05−0.95,

Salp Swarm Algorithm (SSA)

- Swarm Size: 30 salps
- Max Iterations: 100
- Coefficient c1: Linearly decreases from 2 to 0

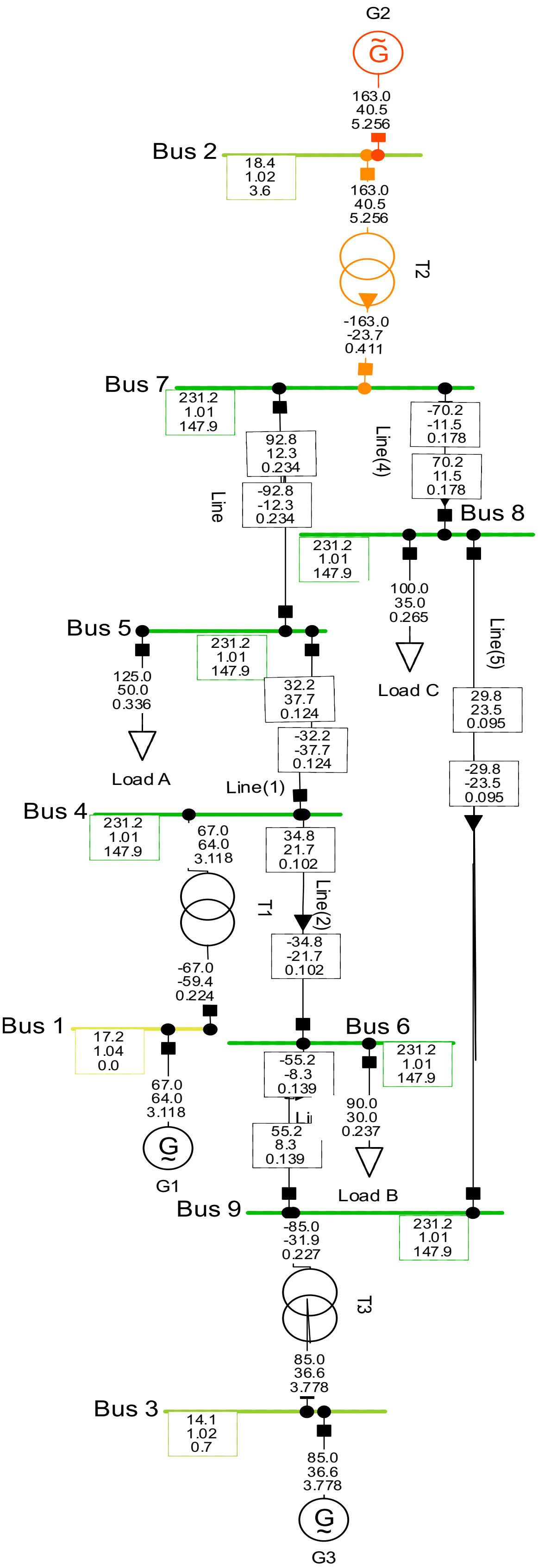

Fig. 3: Load flow analysis without MVAr devices

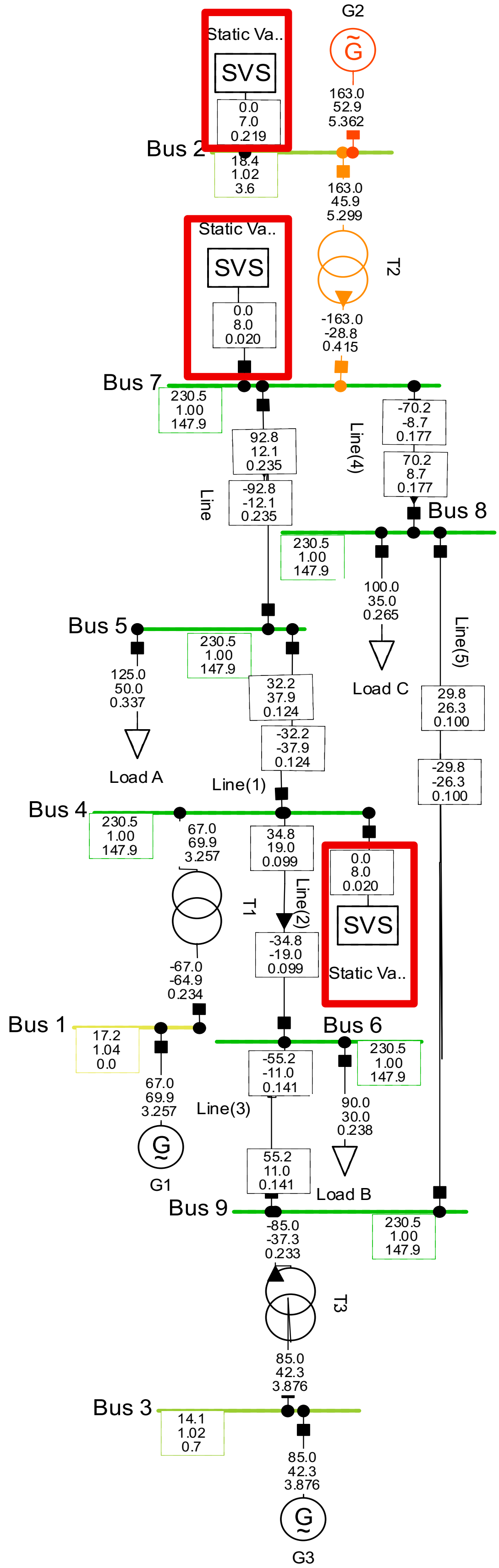

Fig.4: Load flow analysis with 3 PSO recommended MVAr devices

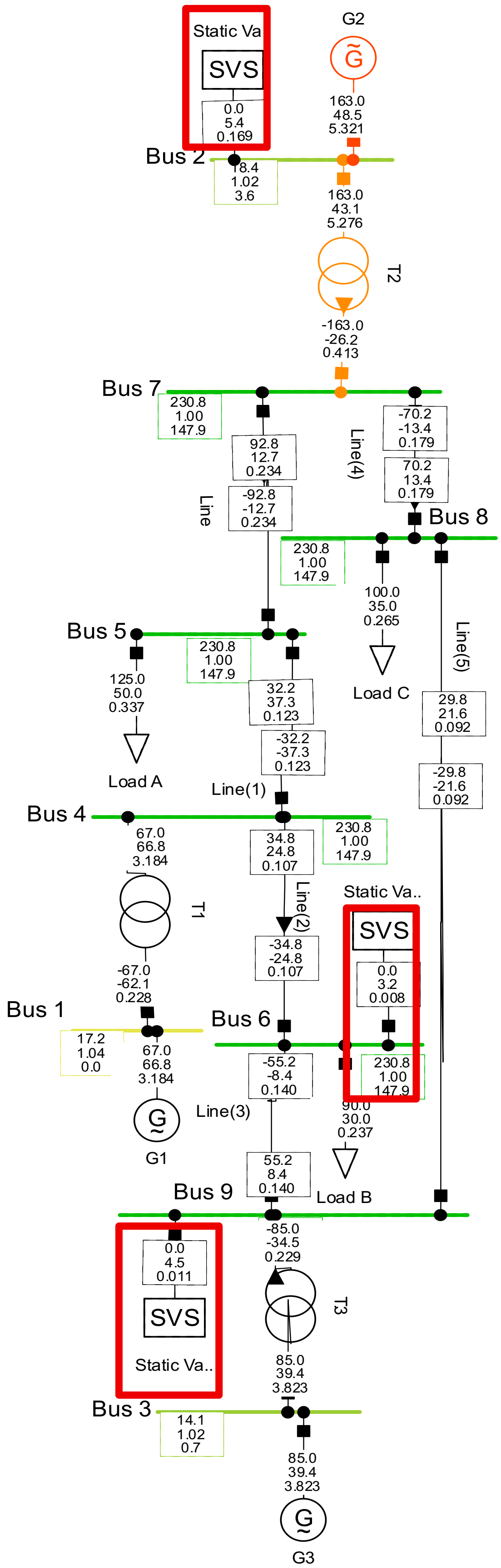

Fig.5: *Load flow analysis with 3 SSA recommended MVAr devices*

Steps:

1. Initialize salp positions (MVAr values).
2. Leader salp updates position relative to food source (best fitness):
3. Follower salps update using above formulae
4. Perform power flow with updated MVAr positions.
5. Repeat for all iterations.

5. LOADFLOW ANALYSIS

The load flow analysis of the IEEE 9 bus system is obtained in Fig. 3. Then, the PSO algorithm recommended 7 MVAr, 8 MVAr and 8 MVAr at buses 2,4 and 7 are configured in Fig. 4. Afterwards, the SSA algorithm recommended MVAr requirements of 5.4, 3.2 and 4.5 at bus numbers 2,6 and 9 respectively is configured in Fig. 5.

6. RESULTS

This section analyses the performance of PSO and SSA algorithms for optimal capacitor placement in IEEE 9-bus transmission system, focusing on voltage stability, convergence behaviour, and computational efficiency. The findings are based on the simulation results presented in Tables 1 and 2, and are supported by voltage profiles and convergence plots shown in Fig. 6,7 and 8 respectively.

Table 1 *Load flow analysis results and MVAr recommendations*

Bus	Vpu	MVAr (PSO)	MVAr (SSA)	Vpu (PSO)	Vpu (SSA)
1	1.04	0	0	1.04	1.04
2	1.02	7	5.4	1.02	1.02
3	1.02	0	0	1.02	1.02
4	1.03	8	0	1.02	1.02
5	1	0	0	0.99	0.99
6	1.01	0	3.2	1.01	1.01
7	1.03	8	0	1.02	1.02
8	1.02	0	0	1.01	1.01
9	1.03	0	4.5	1.03	1.03

6.1 Voltage Profile Improvement and Stability

Table 1 and Fig. 6 illustrates the voltage magnitudes at all buses before and after compensation using PSO and SSA. It is evident that both algorithms effectively enhance the voltage profile, bringing the voltages closer to the nominal value of 1.0 pu. Notably, SSA and PSO demonstrates superior performance in minimizing voltage deviations, particularly at buses 4, 5, 7, and 8, which were initially identified as weak buses (lowest pre-compensation voltages).

Table 2 (voltage profile comparison) and figure 6 confirms this observation, showing that SSA and PSO results in smoother and more stable voltage recovery across the system. This highlights the strength of the algorithms in finding optimal MVAR injection configurations that target voltage instability zones more effectively. It is interesting to note that PSO and SSA both have arrived to an equal total voltage deviation of 0.18 pu as indicated in Table 2.

6.2 Capacitor Placement Justification

Table 1 also presents the optimal locations and sizes (in MVAR) of capacitors determined by PSO and SSA. SSA selects capacitor placements at buses 2, 6, and 9 with values 5.4, 3.2 and 4.5 MVAR respectively. PSO also recommends similar buses but of higher MVAR compensation values with placements at buses 2,4 and 7 with MVAR values of 7,8 and 8 respectively. These locations are consistent with buses exhibiting deviating pre-compensation voltages, indicating that both algorithms correctly identify the irregular voltage areas for reactive power support.

The SSA's sum of injected MVAR values is lower than sum of PSO injected MVAR values. It highlights SSA's more economical voltage profile results. However, PSO's selection pattern reflects a more centralized strategy that reduces overall line losses and voltage drops.

7. COMPARATIVE ANALYSIS

7. 1 Comparison with Literature

A comparison study on SSA and PSO for economic load dispatch optimization problems found that SSA often provides better solution quality and robustness, especially in nonlinear and constrained problems typical in electrical engineering, whereas PSO offers faster convergence but can be prone to premature convergence.

While the statement still holds true in this study, in contrast to [Sinha, 2021], our study introduces a voltage constraint-based optimization using MVAR sizing to maintain optimal performance. SSA still maintains better performance under tighter bounds, suggesting robustness in constrained environments.

Table 2 Voltage deviation after MVAr injections

Bus	PSO deviation	SSA deviation
1	0.04	0.04
2	0.02	0.02
3	0.02	0.02
4	0.02	0.02
5	0.01	0.01
6	0.01	0.01
7	0.02	0.02
8	0.01	0.01
9	0.03	0.03
SUM=>	SUM: 0.18	SUM:0.18

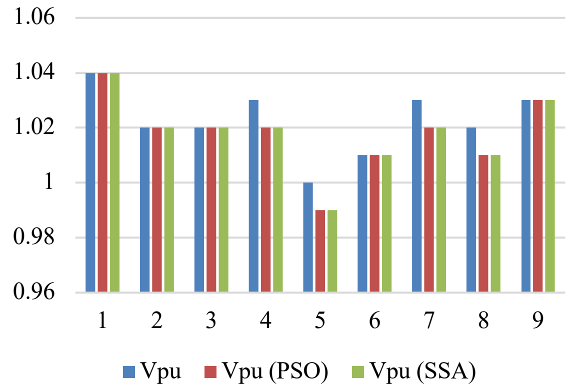

Fig. 6: *Voltage profile before and after compensation*

7.2 Convergence and Computational Time

Fig. 7 (convergence plot) reveals that SSA takes marginally more iterations to converge compared to PSO. However, despite more iterations, SSA demonstrates lower overall computational time (as detailed in Figure 8), attributed to its simpler mathematical operations and fewer parameter dependencies. This makes SSA more computationally efficient and easier to tune for large-scale systems.

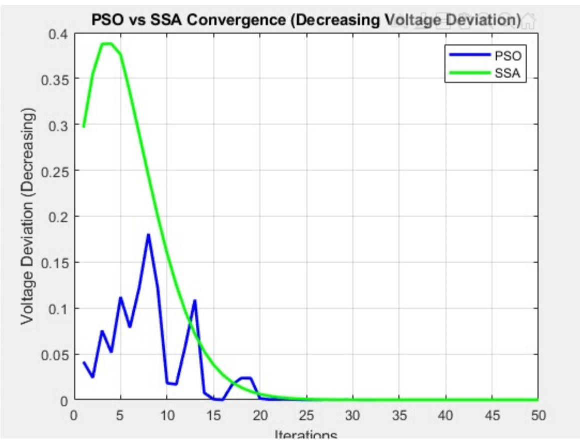

Fig. 7: *Decreasing voltage deviation over the iterations*

7.3 Interpretation

Overall, SSA not only improves voltage profiles more consistently but also demonstrates computational advantages under constrained conditions. Its effective capacitor placement strategy, supported by convergence and voltage plots, confirms its suitability for real-time voltage stability enhancement in transmission networks. Future studies could explore hybrid approaches (e.g., PSO-SSA) to further enhance performance and adaptability.

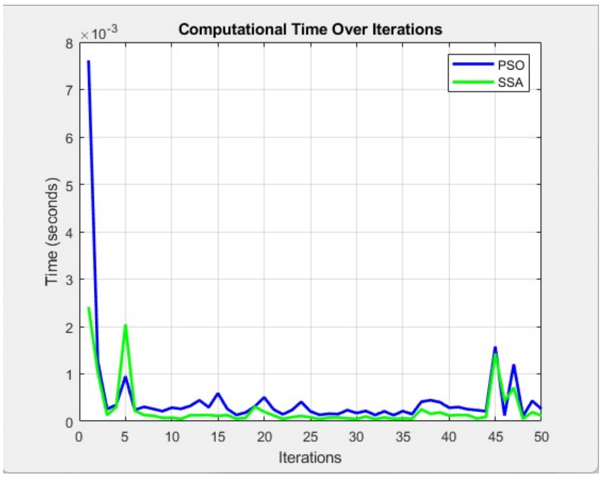

Fig. 8: *Computational time over the iterations*

8. CONCLUSIONS

The simulation results on the IEEE 9-bus system reveal that the Salp Swarm Algorithm (SSA) offers superior performance over Particle Swarm Optimization (PSO) in terms of MVAr sizing. Voltage profile uniformity, and computational efficiency. While PSO converges faster in the initial iterations, it tends to get trapped in local optima, leading to less effective reactive power compensation. SSA, despite requiring slightly more iterations, compensates with better global exploration, producing more optimal and stable voltage profiles. It also demonstrates lower computational time per iteration, indicating higher efficiency. These findings are supported by convergence plots and summarized in the tables provided. Importantly, SSA achieves this performance with fewer configuration requirements, making it a practical and scalable solution for power system applications.

REFERENCES

Mokred, S., & Wang, Y. (2024, September). Voltage stability assessment and contingency ranking in power systems based on modern stability assessment index. *Results in Engineering, 23*.https://www.sciencedirect.com/science/article/pii/S259012302400803X

Ayalew, F., Hussen, S., & Pasam, G. K. (2019, February). Optimization techniques in power system: Review. *International Journal of Engineering Applied Sciences and Technology, 3*(10), 8–16. https://www.researchgate.net/publication/339472409_OPTIMIZATION_TECHNIQUES_IN_POWER_SYSTEM_REVIEW

Valencia-Rivera, G. H., Benavides-Robles, M. T., Morales, A. V., Amaya, I., Cruz-Duarte, J. M., Ortiz-Bayliss, J. C., & Avina-Cervantes, J. G. (2024, January). A systematic review of metaheuristic algorithms in electric power systems optimization. *Applied Soft Computing, 150*, 111047. https://www.sciencedirect.com/science/article/abs/pii/S1568494623010657

Houssein, E. H., Gad, A. G., Hussain, K., & Suganthan, P. N. (2021, June). Major advances in particle swarm optimization: Theory, analysis, and application. *Swarm and Evolutionary Computation, 63*, 100868. https://doi.org/10.1016/j.swevo.2021.100868

Abualigah, L., Shehab, M., Alshinwan, M., & Alabool, H.(2019, November). Salp swarm algorithm: A comprehensive survey. *Neural Computing and Applications*, 1–25. https://doi.org/10.1007/s00521-019-04629-4

Devarapalli, R., Sinha, N. K., Knypiński, Ł., Rao, B. V.,

Lakshmi, N. J. N., & García Márquez, F. P. (2021). Allocation of real power generation based on computing over all generation cost: An approach of Salp Swarm Algorithm. *Archives of Electrical Engineering,* 70(2), 337–349. https://doi.org/10.24425/aee.2021.136988