OPEN ACCESS
International Journal of Innovations in Science & Technology

Optimizing Economic Load Dispatch Using a Hybrid PSO-SA Algorithm: A Novel Approach

Abdul Rafay[1], Uzair Khan[1], Muhammad Mujtaba[1], Abbas Javeed[2]*, Muhammad Tayyab Khalid[1], Zubair Ali[1]

[1]Department of Electrical Engineering (RCET), University of Engineering and Technology, Lahore, 39161, Pakistan

[2]Department of Electrical Engineering (KSK), University of Engineering and Technology, Lahore, 39161, Pakistan

***Correspondence:** Abbas Javeed; abbasjaveed493@gmail.com

Citation | Rafay, A., Khan. U Mujtaba. M., Javeed. A., Khalid. M. T., Ali. A, "Optimizing Economic Load Dispatch Using a Hybrid PSO-SA Algorithm: A Novel Approach", IJIST, Vol. 07 Special Issue. pp 385-396, May 2025

Received | April 21, 2025 **Revised |** May 15, 2025 **Accepted |** May 17, 2025 **Published |** May 19, 2025.

Economic Load Dispatch (ELD) is a crucial power system optimization task. It aims to minimize the total cost of electricity generation by strategically allocating power output among available generating units to meet the system's demand while respecting operational limits. This paper investigates how soft computing methods can improve the effectiveness of Electronic Logging Device (ELD) solutions. Specifically, Particle Swarm Optimization (PSO) and Simulated Annealing (SA) algorithms are employed to minimize generation costs for a power system comprising three generating units. The optimization process considers loss coefficients, generation limits, and a predefined cost function. Initially, PSO is used to determine near-optimal solutions, which are further refined using SA to avoid local minima. A hybrid PSO-SA method integrates the global exploration of Particle Swarm Optimization (PSO) with the local refinement of Simulated Annealing (SA) to enhance convergence and solution quality. 1 This approach was implemented in MATLAB and validated through a case study. Simulation results demonstrate that the hybrid method consistently yields high-quality solutions with reduced computational effort, proving its robustness and reliability for solving ELD problems. Combining metaheuristic algorithms shows promise for real-world power system optimization.

Keywords: Particle Swarm Optimization (PSO); Simulating Annealing (SA); Economic Load Dispatch (ELD); Hybrid Optimization; Power System Optimization.

Introduction

Optimization can play a key role in solving important scientific and industrial problems because many challenges can be converted into standard optimization problems, which can be solved using various techniques. Researchers have developed several methods, including analytical, numerical, rule-based, and advanced search strategies. While analytical methods are cost-effective, they are limited to simpler problems. To address this, numerical, rule-based, and advanced search strategies were developed for solving more complex problems. Unlike analytical methods, which require mathematical proofs, advanced search algorithms are based on a developer's intuition, approach, and experience. The effectiveness of these algorithms is assessed through statistical results from benchmark tests [1]. Particle Swarm Optimization (PSO) is a modern heuristic technique inspired by a simplified social system. It was designed to solve continuous nonlinear optimization problems and has become a reliable method for doing so. PSO provides high-quality solutions quickly, requiring less computation time compared to other stochastic methods [2]. Traditional methods Newton's method [3], Gradient method [4], Lambda Iteration [5], Simulated Annealing [6], Dynamic Programming [6], and Tabu Search Algorithm (TSA) [7], along with more recent methods like Firefly Algorithm [8], Ant Colony Optimization [9], and Neural Networks [10], have been widely used for minimizing generation costs in Economic Load Dispatch (ELD).

PSO stands out because of its fast convergence and low computational cost. It is also often used as a foundation for developing hybrid algorithms. This paper explores the application of PSO, Simulated Annealing (SA), and the hybrid PSO-SA algorithms to solve the ELD problem in a small power system with three generating units [11].

Among all available optimization methods, PSO stands out for its parallel search techniques, making it typically faster than the SA method. However, like Genetic Algorithms (GA) [12], PSO has a key drawback, premature convergence [13]. This occurs when both the best solution of a particle and the best solution of the group become stuck in local minima, limiting the ability to explore globally. In contrast, the most notable feature of SA is its probabilistic leaping property, which helps overcome local optima and has a high probability of reaching the global optimum. Although SA has slower convergence [12], it can avoid getting stuck in local minima.

This paper introduces a new SA-PSO technique that combines the strengths of both methods. The hybrid approach improves solution quality and accelerates convergence, thus reducing computational costs. PSO speeds up convergence in the initial stage, while SA enhances the quality of the solutions obtained. This combined PSO-SA algorithm also improves reliability, especially for complex and large-scale problems, by minimizing the risk of local minima.

Objective:

The objectives of this research are as follows:

- Use Particle Swarm Optimization (PSO) AND Simulated Annealing (SA) algorithms to minimize generation costs in a power system with three generating units.
- Develop a hybrid PSO-SA method that combines the global search power of PSO with the local refinement capability of SA.
- Achieve better convergence and improved solution quality.
- Validate the effectiveness of the proposed approach by implementing it in the MATLAB environment.
- Show that the hybrid method consistently provides high-quality solutions while reducing computational effort.

Literature Review & Research Gap:

To address the limitations of individual methods and improve performance across a wider range of tasks, researchers have developed various hybrid metaheuristic algorithms. These hybrid approaches combine the strengths of different algorithms to enhance their effectiveness in solving diverse optimization problems. For instance, Liu et al. introduced a PSO-DE algorithm [14], which combines Swarm Optimization (PSO) and Differential Evolution (DE) for constrained optimization.

Separately, S. S. Jadon et al. developed a hybrid artificial bee colony algorithm [15] that incorporates Simulated Annealing to improve search efficiency and reduce computational costs [13]. Rizk-Allah et al. designed a hybrid algorithm combining Ant Colony Optimization (ACO) and Firefly Algorithm (FA) to solve unconstrained optimization problems [16].

Wang et al. proposed several novel methods, including a krill herd algorithm that integrates genetic operators and a harmony search-krill herd hybrid to enhance global optimization. Also introduced a biogeography-based krill herd and chaotic krill herd algorithm to optimize performance and improve global convergence. Myszkowski et al. used a hybrid ACO to address project scheduling problems by combining ACO with heuristic rules.

Samuel and Rajan developed hybrid PSO-based algorithms for generation maintenance scheduling. Additionally, Wang et al. enhanced the krill herd algorithm [15] using opposition-based learning, Cauchy mutation, and position clamping. Jung et al. applied a hybrid Simulated Annealing algorithm to optimize dynamic ride-sharing, while Wang et al. integrated firefly-inspired techniques into the krill herd algorithm [15] to improve local search and population diversity.

These hybrid techniques have proven effective in solving a variety of benchmark problems and demonstrate significant potential for addressing complex optimization challenges with efficiency.

Research Flow:

The authors present a hybrid method combining Particle Swarm Optimization (PSO) and Simulated Annealing (SA) to solve the Economic Load Dispatch (ELD) problem. PSO is first used to find near-optimal solutions, which are then improved by SA to avoid local minima. This approach takes advantage of PSO's global search capability and SA's local refinement strength. Implemented in MATLAB and validated on a three-unit power system, the simulation results demonstrate that the hybrid PSO-SA method effectively provides high-quality solutions with reduced computational effort.

Novelty Statement:

The novelty of this research lies in the introduction of a hybrid PSO-SA algorithm to address the Economic Load Dispatch (ELD) problem. By combining PSO's global search capability with SA's local search strength, this hybrid algorithm addresses the issue of PSO getting stuck in local minima. The annealing process in SA helps the algorithm escape these local minima traps. As a result, the hybrid algorithm delivers superior solutions compared to using PSO or SA alone, demonstrating better convergence rates and more optimal solutions for power system optimization problems like economic dispatch.

Material and Methods:

Formulation of the ELD Problem:

The Economic Load Dispatch (ELD) problem seeks to minimize the fuel costs of power generation by optimally distributing real power among generating units, while respecting operational constraints and ensuring demand is met.

$$\sum_{i=1}^{n} (a_i P_i^2 + b_i P_i + c_i (1)$$

Where:

- Ci represents the total cost function.
- Ai, bi, and Ci are the cost coefficients for the i-th generator.
- n represents the total number of generating units.
- Pi is the power output of the i-th generator.

The objective function is subject to the following equality and inequality constraints.

Generator Output Limits:

Each generator unit has upper and lower limits on its output to ensure safe operation. These limits are dictated by the thermal capacity of the generating units as well as practical operational constraints, such as the stability of the flame in a boiler. This can be expressed as:

$$Pi,min \;\leq\; Pi \leq\; Pi,max \quad for\, i \,=1,2,\ldots,n \qquad (2)$$

Where:

- Pi, min represents the minimum power limit of the i-th generator
- Pi, max represents the maximum power limit of the i-th generator

Power Balance Constraint:

To satisfy the total system load, the sum of the power generated by all units must equal the load demand plus any transmission losses. This can be expressed as:

$$\sum_{i=1}^{n} P_i - P_L = P_D \qquad (3)$$

Where:

- PD is the total load demand
- Pi is the power output from the i-th generator
- PL represents the total transmission losses in the network

Transmission Losses: The transmission losses in the system are typically calculated using the B-coefficient method. The losses can be expressed as:

$$P_L = P^T BP + P^T B_o + B_1 \quad (4)$$

Where:

- P is the vector of generator power outputs
- B is the loss coefficient matrix,
- Bo is the loss coefficient vector
- B1 is a constant representing fixed transmission losses

Summary of Conventional PSO, SA, and Their Hybrid Approaches:

Overview of PSO, SA, and Hybrid PSO-SA:

1. Particle Swarm Optimization (PSO):

- **Concept:** Inspired by the social behavior of birds flocking or fish schooling, PSO optimizes a problem by iterating over a population of candidate solutions (particles). Each particle adjusts its position in the solution space based on its own best-known position and the best-known position of its neighbors.
- **Goal:** Minimize or maximize a given objective function by searching the solution space with the help of the swarm.
- **Steps:**

1. Initialize particles with random positions and velocities.
2. Evaluate the fitness of each particle.
3. Update each particle's velocity and position based on its best-known position and the best-known position in the swarm.
4. Repeat until convergence.

2. **Simulated Annealing (SA):**

- **Concept:** SA is inspired by the annealing process in metallurgy, where materials are heated and then gradually cooled to find the minimum energy state. It mimics this process to escape local minima and explore the solution space more thoroughly.
- **Goal:** Find a global minimum (or maximum) of a function by allowing a controlled random step.
- **Steps:**

1. Start with high "temperature" (initial exploration).
2. Gradually decrease the temperature, making smaller adjustments to the solution.
3. If a worse solution is found, accept it with a certain probability that decreases as the temperature lowers.
4. Repeat until convergence.

3. **Hybrid PSO-SA:**

- **Concept:** The hybrid PSO-SA combines the global search capability of PSO and the local refinement power of SA. PSO is used initially to explore the solution space and find the near-optimal solutions, while SA is employed to refine the solutions and avoid getting trapped in local minima.
- **Goal:** Achieve faster convergence and higher-quality solutions by combining the strengths of both PSO and SA.
- **Steps:**

1. Apply PSO to find near-optimal solutions.
2. Use SA to refine these solutions by avoiding local minima and improving the solution quality.
3. Iterate the process until the best solution is found.

This hybrid approach leverages the exploration of PSO and the exploitation ability of SA to provide high-quality solutions with reduced computational effort. It is particularly useful in complex optimization problems like Economic Load Dispatch (ELD).

PSO Technique:

Particle Swarm Optimization (PSO) was introduced by James Kennedy and Russell Eberhart in 1995. The algorithm is inspired by social behavior observed in animals, like flocks of birds or schools of fish, as they search for food. These "particles" move through the solution space, adjusting their positions based on personal experience and the experience of neighboring particles.

This method mimics the collective behavior of animals, where each animal adjusts its position based on its own best position and the best position found by others, effectively converging toward the optimal solution.

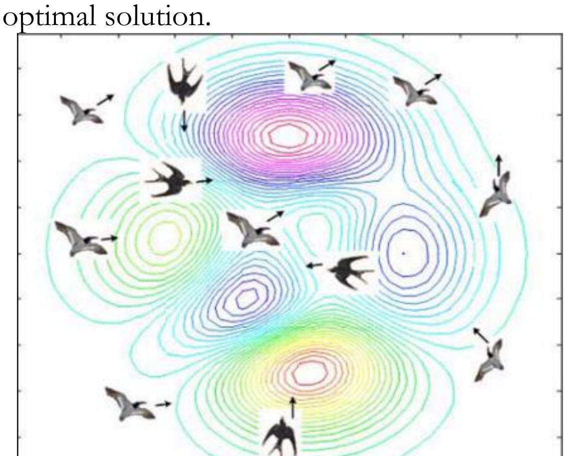

Figure 1. Food searching by a swarm of birds.

Particle Swarm Optimization (PSO) is an optimization technique that simulates the behavior of a flock of birds searching for food. Each solution, called a particle, represents a candidate in the search space and has the following attributes:

- **Position:** The current state or solution in the search space.
- **Velocity:** The rate of change in the position, determining how the particle moves through the search space.
- **Personal Best (pbest):** The best solution the particle has ever encountered.

Particles are organized into a "swarm," and they interact with each other, sharing their knowledge about the best solutions they have found. Each particle is guided by:

1. Personal Best (pbest): The best position that a particle has found so far.
2. Global best (gbest): The best position found by any particle in the swarm.

The particles update their position and velocities based on these factors. The equations for updating the position and velocity are:

- **PSO Process:**

1. **Initialization:** Initialize a swarm of particles with random positions and velocities.
2. **Evaluation:** Evaluate each particle's fitness (objective function values).
3. **Update:** Update the particle's velocity and position using the formulas above.
4. **Convergence Check:** If the stopping criteria are met, the algorithm stops. Otherwise, repeat the evaluation and update process.

Through this iterative process, the swarm convergence toward the optimal solutions, balancing exploration (search globally) and exploitation (refining the current best solutions). PSO is widely used in optimization problems because of its simplicity, efficiency, and ability to explore large, complex search spaces.

$$xi(t) = (xi1(t), xi2(t), \dots, xin(t)) \qquad (5)$$

$$vi(t) = \left(vi1(t), vi2(t), \dots, vin(t)\right) \qquad (6)$$

The "global best" (gbest) signifies the optimal position discovered by any member of the entire swarm up to that point, while each particle follows its own best-known position, referred to as the "local best" (pbest). At any given time, t, the values of pbest and gbest are expressed as follows:

$$pbesti(t) = \left(pbesti1(t), pbesti2(t), \dots, pbestin(t)\right) \qquad (7)$$

$$gbesti(t) = \left(gbesti1(t), gbesti2(t), \dots, gbestin(t)\right) \qquad (8)$$

To explore the search space and improve fitness, particles iteratively update their positions and velocities using the following formulas

$$\mathit{v}i(t+1) = w \cdot vi(t) + c1r1(pbesti(t) - xi(t)) + c2r2(gbesti(t) - xi(t)) \qquad (9)$$

$$xi(t+1) = xi(t) + vi(t+1) \qquad (10)$$

Where:

- w is the inertia weight that controls exploration.
- c1 and c2 are acceleration coefficients.
- r1 and r2 are random values between 0 and 1.
- xi represents the current position of particle III.
- I is the velocity of particle III.

These equations help avoid stagnation, enabling continuous updates, allowing the particle to explore the search space, and converge towards optimal solutions. The PSO flowchart is shown in Figure 2.

SA Technique:

In contrast, current methods using Simulated Annealing (SA) algorithms treat each generator's actual power output as a decision variable to solve the Economic Load Dispatch (ELD) problem. This approach makes the problem large and complex, which slows down the algorithms and reduces efficiency when there are many generating units. To simplify the problem, it is recommended to use a penalty factor (λ), derived from the traditional λ -λ-iteration method, as a single decision variable, regardless of the number of generating units. The actual power outputs of the generating units are then calculated as a function of λ. For each value of λ found during the SA iterations, the real power outputs are compared. The system's minimum and maximum power requirements determine the appropriate range for λ. These bounds are calculated by replacing the power limits of all generating units in equation (5) and assessing the lower and upper incremental cost values.

$$ICmax = 2ai\ Pmax + bi \qquad (11)$$

$$ICmin = 2ai\ Pmin + bi \qquad (12)$$

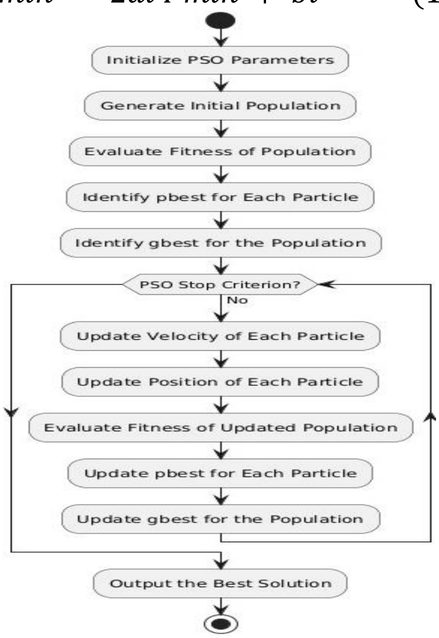

Figure 2. PSO Algorithm flowchart.

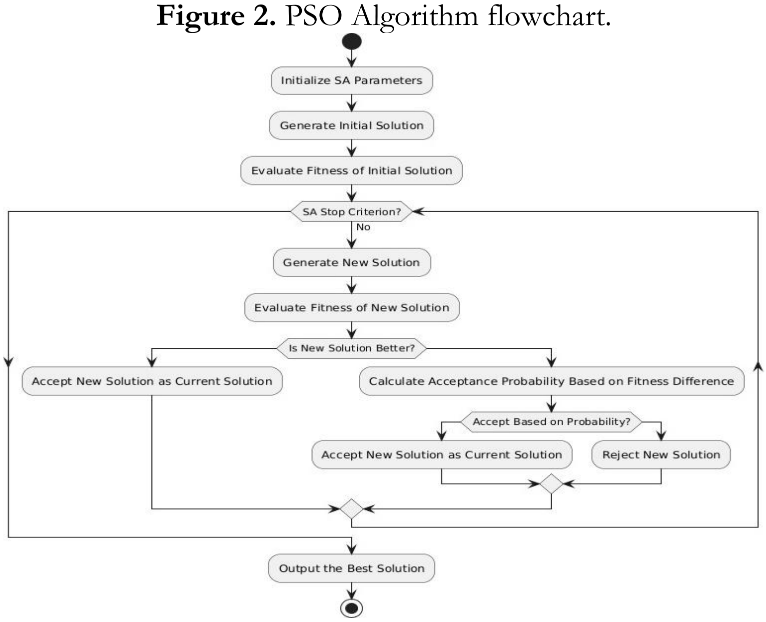

Figure 3. Algorithm flowchart of SA.

Next, the minimum and maximum values of λ are determined based on the lowest and highest incremental cost values derived from equation (11), as shown below:

$$\lambda max = max\ (IC1, IC2, \dots, ICN) \qquad (13)$$

$$\lambda min = min\ (IC1, IC2, \dots, ICN) \qquad (14)$$

The SA algorithm minimizes a cost function to find the optimal solution. In the proposed method, the total fuel costs of the generating units are considered as the objective function. To address the power balance constraint, a penalty term is also included. This penalty increases the cost function for infeasible solutions. The final cost function combines the fuel cost and the power balance constraint, as shown below:

$$\sum_{i=1}^{N} F_i(P_i) + \eta\,(P_D + P_{loss} - \sum_{i=1}^{N} P_i) \qquad (15)$$

Unlike other SA-based methods that require the generation levels of all units to be treated as variables, this approach uses only one decision variable, η. This simplification reduces the number of variables, thereby lowering computational complexity and enhancing the algorithm's convergence speed. The SA flowchart is shown in Figure 3.

The Hybrid PSO-SA Method:

The hybrid algorithm for solving the ELD problem combines the strengths of PSO and SA. PSO focuses on exploration, while SA contributes to local optimization and refining the convergence. PSO effectively handles the global search, but local optima must be avoided, and local minima must be escaped during the refinement process using SA. In this process, the SA technique iterates to refine the best solution found. SA introduces randomness by perturbing the solution and gradually cooling down the temperature to explore smaller neighborhoods within the solution space.

1. **Minimized Fuel Cost**: The hybrid algorithm minimizes the fuel cost based on the given power demand. Each generator's output is adjusted to avoid infeasibility and prevent an increase in the overall cost.
2. **Power Balance**: The total power output meets the system's demand, including transmission losses, while ensuring each unit's generation limits are respected. This guarantees that realistic and feasible solutions meet the system's demand, including transmission losses, while ensuring each unit's generation limits are respected. This guarantees realistic and feasible power allocations.
3. **Enhanced Convergence**: By combining PSO and SA, the hybrid approach increases the likelihood of finding the global optimum solution to the ELD problem. After PSO converges, SA refines the solution, yielding a more precise and economical power dispatch.

The flow diagram of the hybrid model is shown in Figure 4.

Simulation Results and Discussion:

A power system consisting of three generators and five buses, with a total load demand of 150 MW, was evaluated using Particle Swarm Optimization (PSO), Simulated Annealing (SA), and a hybrid PSO-SA approach in MATLAB to address the Economic Load Dispatch (ELD) challenge. The system's parameters, based on Example 7.7 from reference [5], are summarized in Table 1.

Table 1. Cost Function and Generator Constraints

Unit	P_{min} (MW)	P_{max} (MW)	Alpha	Beta	gamma
1	10	85	200	7	0.008
2	10	80	180	6	0.009
3	10	70	140	6	0.007

Where the loss matrix BB is defined as:

$$B_{ij} = \begin{bmatrix} 0.0218 & 0.0093 & 0.0028 \\ 0.0093 & 0.0228 & 0.0017 \\ 0.0028 & 0.0017 & 0.0179 \end{bmatrix}$$

$$B_{0i} = [\,0.0003 \quad 0.0031 \quad 0.0015\,]$$

$$B_{00} = 00030523$$

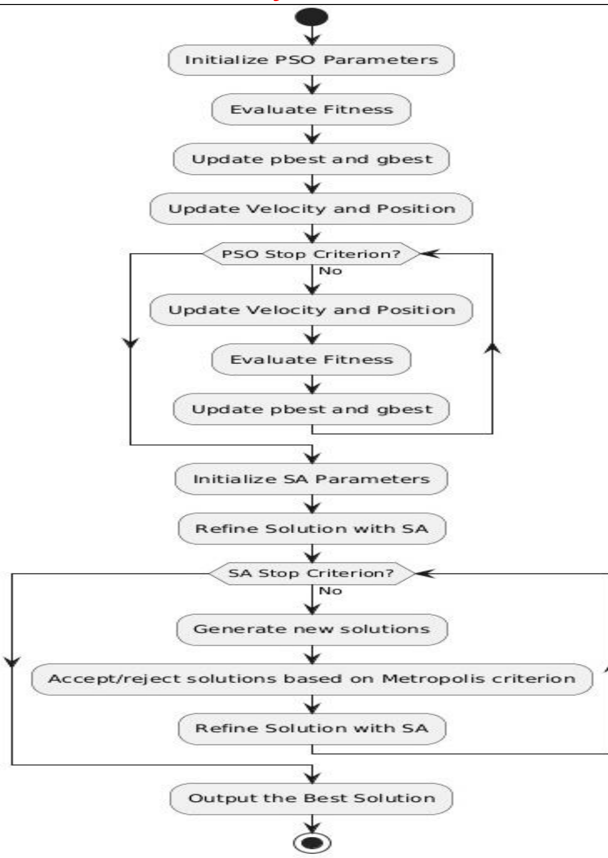

Figure 4. Flowchart Hybrid PSO-SA Technique.

Based on the information provided in Table I, the lowest power output that each of the three generators can produce is 10 megawatts. The maximum generation limits are 85 MW for Generator 1, 80 MW for Generator 2, and 70 MW for Generator 3. These upper and lower bounds, along with the cost function input data, define the operating range of the generators. Ideally, electricity generation should match the system's demand, but practical constraints, such as transmission line losses, make this challenging. To account for these losses, a matrix B is used to optimize the problem by incorporating load distribution. The coefficients of this matrix, known as loss coefficients, are derived from the bus matrix of the observed bus.

The Economic Load Dispatch (ELD) problem for this test system was also analyzed in Example 7.7 [5], with the minimum fuel cost recorded as $1599.98. To efficiently optimize the ELD problem and demonstrate the superior optimization capabilities of the hybrid PSO-SA approach, the same test system was evaluated using PSO and SA. Each algorithm was set to a maximum of 1000 iterations. The test system was simulated in MATLAB for PSO, SA, and

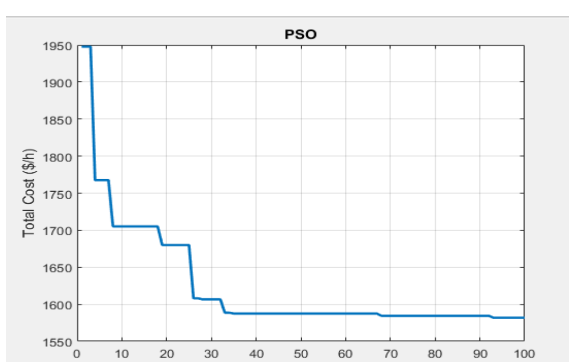

Figure 5. Total cost of generation vs no of iterations (PSO).

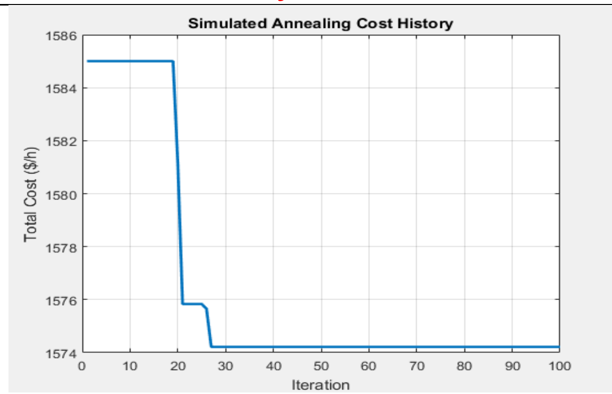

Figure 6. Total cost of generation vs no of iterations (SA).

Hybrid PSO-SA, with the best results obtained from 30 runs per algorithm. The total production cost calculated using PSO was $1580/h, while SA achieved a lower cost of $1573/h. However, the hybrid PSO-SA method yielded the lowest total fuel cost of $1567/h, proving to be the most cost-effective approach.

The outcomes of the three optimization techniques are summarized using bar charts, as shown in Figure 8, to facilitate better evaluation.

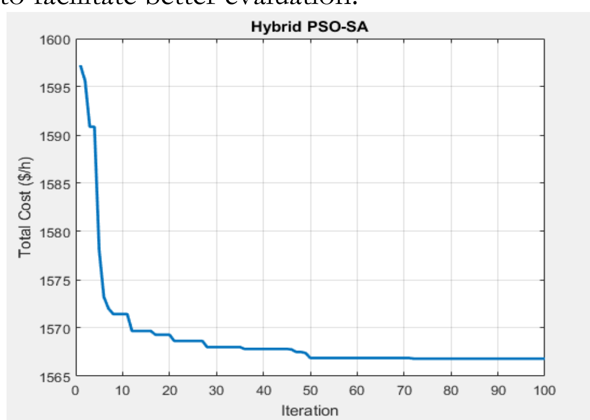

Figure 7. Total cost of generation vs no of iterations (Hybrid PSO-SA).

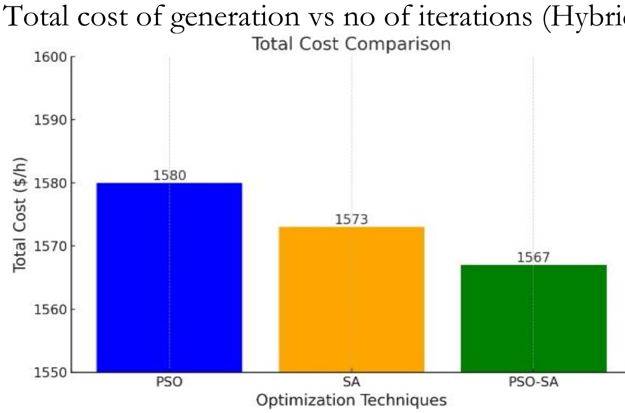

Figure 8. Comparison of fuel cost.

Discussion:

The comparative analysis of the simulation results reveals the efficiency and reliability of the proposed hybrid PSO-SA algorithm over standalone optimization techniques such as Particle Swarm Optimization (PSO) and Simulated Annealing (SA). While PSO demonstrates fast convergence and effective global exploration, it often suffers from premature convergence, leading to sub-optimal solutions. On the other hand, SA provides robustness against local optima but at the cost of slower convergence rates. The hybridization of these two metaheuristic approaches leverages the global search capability of PSO and the local refinement strength of SA, achieving a synergistic effect that enhances overall optimization performance.

For the given Economic Load Dispatch (ELD) problem involving three generators and a load demand of 150 MW, the hybrid PSO-SA algorithm achieved the lowest total fuel cost of $1567/h, compared to $1573/h for SA and $1580/h for PSO. This outcome clearly illustrates the hybrid model's ability to converge on more cost-effective solutions by overcoming local minima that would typically hinder PSO alone. Furthermore, the hybrid approach reduced the number of iterations required to reach optimal or near-optimal solutions, thereby minimizing computational effort.

The improved performance of the hybrid algorithm can be attributed to the intelligent sequencing of operations. PSO first locates a promising region in the search space, followed by SA's fine-tuning mechanism, which enhances solution precision. This combination not only speeds up convergence but also increases the likelihood of achieving global optimum solutions. In real-world scenarios where power system optimization is subject to various nonlinear constraints and uncertainties, the robustness of such hybrid methods is highly advantageous.

Additionally, the implementation of the loss coefficient matrix in the optimization model ensures realistic power system modeling by incorporating transmission losses, thus improving the practical applicability of the results. The power balance constraint and generator limits were strictly maintained in all simulations, reinforcing the feasibility of the proposed method for real-world ELD problems.

Overall, the hybrid PSO-SA algorithm stands out as a powerful and efficient tool for optimizing complex power system problems. It not only improves the quality of the solutions but also reduces the computational burden, making it a promising approach for modern power system operators who are constantly seeking cost-effective and reliable solutions.

Conclusion:

In this research paper, the hybrid PSO-SA algorithm is introduced to effectively and efficiently solve complex power system optimization problems. It combines the global search strength of PSO with the local search capabilities of SA, offering a more accurate and efficient optimization process. This combination helps avoid the local minima trap in PSO, with SA's annealing process further assisting in escaping these traps. The study shows that the hybrid algorithm outperforms conventional PSO and SA, yielding superior solutions with better convergence rates and more optimal solutions for power system optimization problems like economic dispatch. The results are promising and make a significant contribution to optimization research. This work can be expanded in the future to tackle more complex and non-linear Economic Load Dispatch (ELD) problems, broadening its applicability to a wider range of scenarios.

Acknowledgement: We would acknowledge the University of Engineering and Technology, Lahore, Punjab, Pakistan, for providing the environment to conduct the experimental work.

Author's Contribution: Each author has made an equal contribution to this study**.**

Conflict of interest: There is no conflict of interest among the authors.

References:

[1] V. T. Sunil Gupta, Kamal Saluja, Ankur Goyal, Amit Vajpayee, "Comparing the performance of machine learning algorithms using estimated accuracy," *Meas. Sensors*, vol. 24, p. 100432, 2022, doi: https://doi.org/10.1016/j.measen.2022.100432.

[2] R. Rebolledo, "Stochastic Processes," *Int. Encycl. Stat. Sci.*, pp. 1534–1539, 2011, doi: 10.1007/978-3-642-04898-2_529.

[3] B.T. Polyak, "Newton's method and its use in optimization," *Eur. J. Oper. Res.*, vol. 181, no. 3, pp. 1086–1096, 2007, doi: https://doi.org/10.1016/j.ejor.2005.06.076.

[4] Jasbir S. Arora, "Chapter 10 - Numerical Methods for Unconstrained Optimum

Design," *Introd. to Optim. Des. (Third Ed.*, pp. 411–441, 2012, doi: https://doi.org/10.1016/B978-0-12-381375-6.00010-3.

[5] Z. N. Jan, "Economic Load Dispatch using Lambda Iteration, Particle Swarm Optimization & Genetic Algorithm," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 9, no. 8, 2021, [Online]. Available: https://www.ijraset.com/fileserve.php?FID=37527

[6] M. and K. F. Mohammadi, "A dynamic programming–enhanced simulated annealing algorithm for solving bi-objective cell formation problem with duplicate machines," *Decis. Sci. Lett.*, pp. 261–276, 2015, [Online]. Available: https://www.growingscience.com/dsl/Vol4/dsl_2014_39.pdf

[7] V. K. Prajapati, M. Jain, and L. Chouhan, "Tabu Search Algorithm (TSA): A Comprehensive Survey," *Proc. 3rd Int. Conf. Emerg. Technol. Comput. Eng. Mach. Learn. Internet Things, ICETCE 2020*, pp. 222–229, Feb. 2020, doi: 10.1109/ICETCE48199.2020.9091743.

[8] Arpan Kumar Kar, "Bio inspired computing – A review of algorithms and scope of applications," *Expert Syst. Appl.*, vol. 59, pp. 20–32, 2016, doi: https://doi.org/10.1016/j.eswa.2016.04.018.

[9] M. Dorigo and T. Stützle, "Ant Colony Optimization: Overview and Recent Advances," *Int. Ser. Oper. Res. Manag. Sci.*, vol. 272, pp. 311–351, 2019, doi: 10.1007/978-3-319-91086-4_10.

[10] Larry Hardesty, "Explained: Neural networks," *MIT News Off.*, 2017, [Online]. Available: https://news.mit.edu/2017/explained-neural-networks-deep-learning-0414

[11] M. Ahmad, W. Ali, H. Farooq, M. Jamil, M. Ali, and A. Ur Rehman, "Solving the Problem of Economic Load Dispatch for a Small Scale Power System Using a Novel Hybrid PSO-GSA Algorithm," *RAEE 2018 - Int. Symp. Recent Adv. Electr. Eng.*, Jul. 2018, doi: 10.1109/RAEE.2018.8706897.

[12] I. G. T. & A. T. Vasileios Charilogis, "An Improved Parallel Particle Swarm Optimization," *SN Comput. Sci.*, vol. 4, no. 766, 2023, doi: https://doi.org/10.1007/s42979-023-02227-9.

[13] G. Xu and G. Yu, "On convergence analysis of particle swarm optimization algorithm," *J. Comput. Appl. Math.*, vol. 333, pp. 65–73, 2018, doi: https://doi.org/10.1016/j.cam.2017.10.026.

[14] H. L. Zhao Liu, Zhiwei Qin, Ping Zhu, "An adaptive switchover hybrid particle swarm optimization algorithm with local search strategy for constrained optimization problems," *Eng. Appl. Artif. Intell.*, vol. 95, p. 103771, 2020, doi: https://doi.org/10.1016/j.engappai.2020.103771.

[15] J. C. B. Shimpi Singh Jadon, Ritu Tiwari, Harish Sharma, "Hybrid Artificial Bee Colony algorithm with Differential Evolution," *Appl. Soft Comput.*, vol. 58, pp. 11–24, 2017, doi: https://doi.org/10.1016/j.asoc.2017.04.018.

[16] M. S. Huimin Zhao, Weitong Gao, Wu Deng, "Study on an Adaptive Co-Evolutionary ACO Algorithm for Complex Optimization Problems," *Symmetry (Basel).*, vol. 10, no. 4, p. 104, 2018, doi: https://doi.org/10.3390/sym10040104.