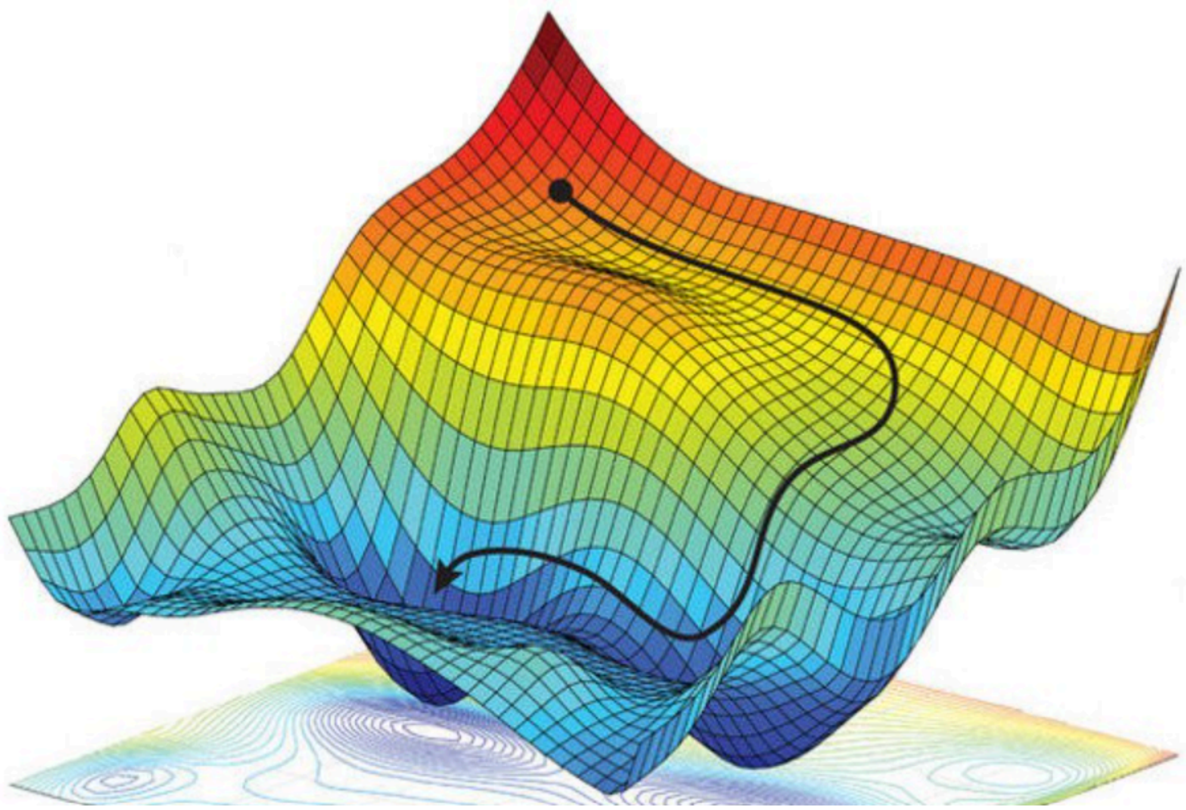
Task 3 Group Report: Optimization Techniques

Eneko ORHATEGARAY, Paul BROCVIELLE, Aiert CECCON, Achille LARREGLE

Date: 2025-12-12



1. Introduction

The objective of this study is to investigate and compare the performance of two distinct stochastic metaheuristic optimization algorithms: Simulated Annealing (SA) and Particle Swarm Optimization (PSO).

In engineering design, optimization problems are often non-linear, multi-modal (having multiple local optima), and subject to constraints. Deterministic methods (like Gradient Descent) often fail in such landscapes or require expensive gradient calculations. Stochastic methods, which rely on probabilistic rules to explore the search space, offer a robust alternative.

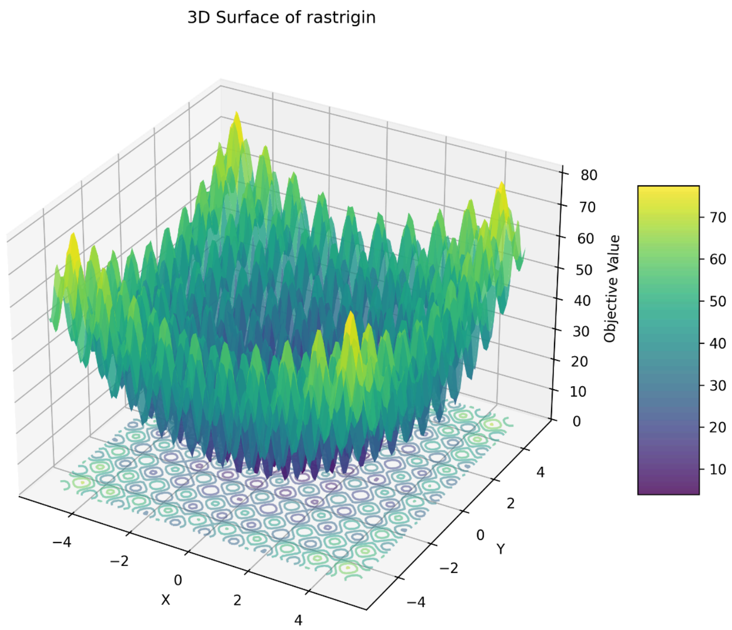
This report evaluates these two algorithms on a test bench of three mathematical functions to assess their:

1. Global Search Capability: Ability to escape local minima.
2. Precision: Ability to refine the solution to high accuracy.
3. Constraint Handling: Ability to respect feasible boundaries using penalty functions.

2. Mathematical Problem Definitions

The following three problems were selected to test different aspects of the optimizers.

2.1 Rastrigin Function (Unconstrained)



This function is highly multi-modal, with a regular lattice of local minima. It serves as a stress test for an algorithm's ability to maintain population diversity (PSO) or accepting uphill moves (SA) to avoid premature convergence.

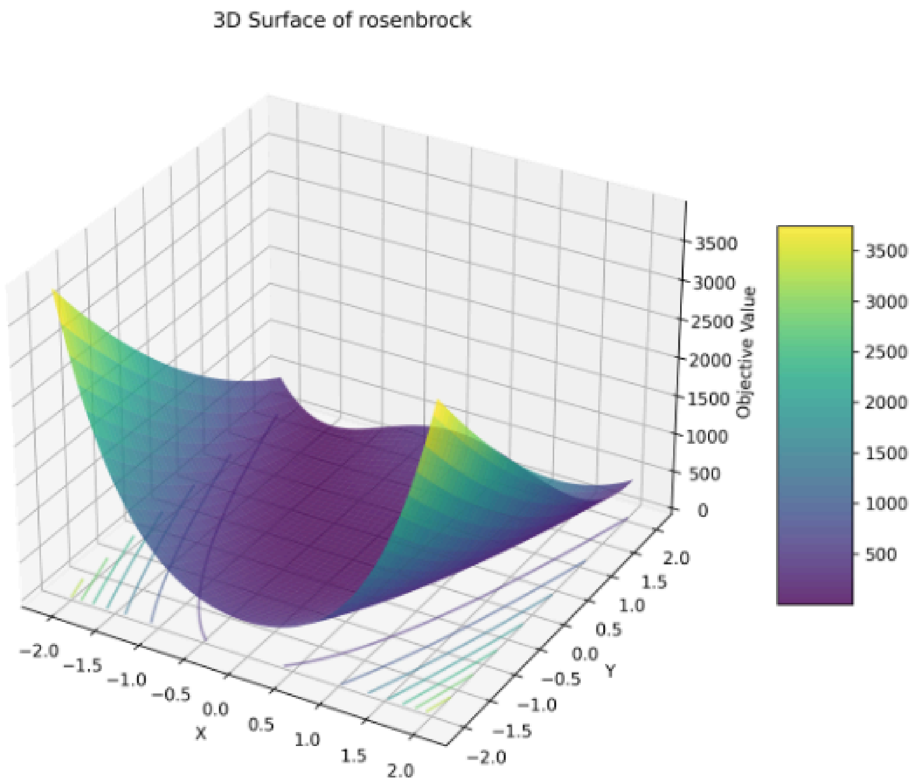
$x_i \in [-$

$$f(x) = 10d + \sum_{i=1}^{d} [x_i^2 - 10\cos(2\pi x_i)]$$

Bounds:

Global Minimum: $f(0) = 0$

2.2 Rosenbrock Function (Unconstrained)



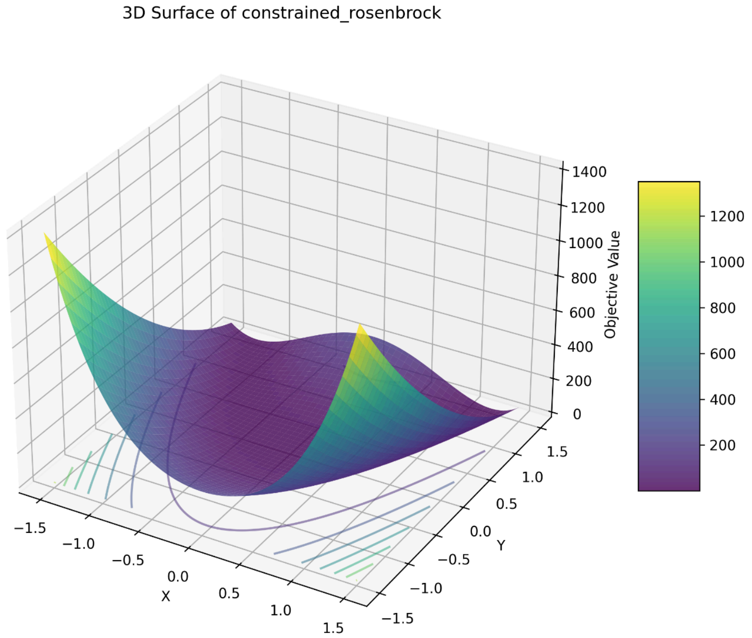
The global minimum lies in a narrow, parabolic valley. Finding the valley is trivial, but converging to the exact minimum is difficult. This tests the algorithm's local exploitation capability.

$$f(x) = \sum_{i=1}^{d-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$$

Bounds: $x_i \in [-2.048, 2.048]$

Global Minimum: $f(1) = 0$

2.3 Constrained Rosenbrock Function



The unconstrained global minimum at (1,1) is at a distance of sqrt(2)≈1.414 from the origin, which is outside the feasible region (radius 1). The true constrained minimum lies on the boundary of the feasible region. This tests the algorithms' ability to balance objective reduction with feasibility.

Constraint: $g(x) = x_1^2 + x_2^2 - 1 \leq 0.$

3. Methodology & Algorithms

3.1 Algorithms Implemented

Simulated Annealing (SA): A trajectory-based method inspired by thermodynamics. We utilized a geometric cooling schedule ($T_{k+1}=\alpha T_k$) with $\alpha=0.95$. This allows the algorithm to accept worse solutions with high probability early in the search (exploration) and settle into a minimum later (exploitation).

Particle Swarm Optimization (PSO): A population-based method inspired by bird flocking. Implemented with standard inertia weight ($w=0.5$) and cognitive/social coefficients ($c_1=c_2=1.5$). This leverages collective intelligence to explore the search space.

3.2 Constraint Handling

Constraints were handled using a Static Penalty Function. The constrained optimization problem is transformed into an unconstrained one by adding a penalty term to the objective function: $P(x) = f(x) + R * (\max(0, g(x)))^2$
Where R is the penalty factor. This "soft" constraint approach guides the stochastic optimizer back towards the feasible region if it strays.

3.3 Experimental Design and Fairness

To ensure a rigorous and fair academic comparison, the following controls were strictly enforced:

Evaluation Budget: Both algorithms were restricted to a maximum of 10,000 function evaluations per run. This is the primary "cost" metric.

- SA: 10,000 sequential steps.
- PSO: 30 particles x 333 generations.

Stochasticity Control: Each experiment was repeated 30 times independently.

- Reporting: Results are reported as the Mean Best Value and Standard Deviation across these 30 runs to account for random variance.

4. Results and Analysis

4.1 Unconstrained Optimization Performance

Problem	Algorithm	Mean Best Value	Std Dev	Mean Time (s)
Rastrigin	SA	1.7834e+01	1.0451e+01	0.1211
	PSO	2.3216e-01	4.2082e-01	0.1539
Rosenbrock	SA	1.1972e-05	8.6648e-06	0.1376
	PSO	0.0000e+00	0.0000e+00	0.1766

Performance

Rastrigin: PSO significantly outperformed SA. The mean value of ~0.23 indicates that PSO frequently found the global minimum (0) or a very close local one. SA (mean ~17.8) consistently failed to escape local basins of attraction. This demonstrates PSO's superior global search capability given the fixed budget.

Rosenbrock: PSO achieved perfect convergence (0.00) with zero variance, demonstrating exceptional exploitation capabilities in the parabolic valley. SA performed well (1e-5) but could not achieve the same level of machine precision within the evaluation limit.

4.2 Constrained Optimization

Problem	Algorithm	Mean Best Value	Std Dev
Constrained Rosenbrock	SA	4.6214e-02	3.6140e-04
	PSO	4.5671e-02	5.9421e-18

Analysis

The distinct advantage of PSO is visible in the Standard Deviation ($5.9\times10-18$). This indicates that in every single one of the 30 runs, PSO converged to the exact same numerical value, suggesting it reliably identified the true constrained optimum on the boundary. SA found a solution close to this value (~0.046) but with significantly higher variance, struggling to stabilize exactly on the boundary.

4.3 Penalty Sensitivity Analysis (PSO)

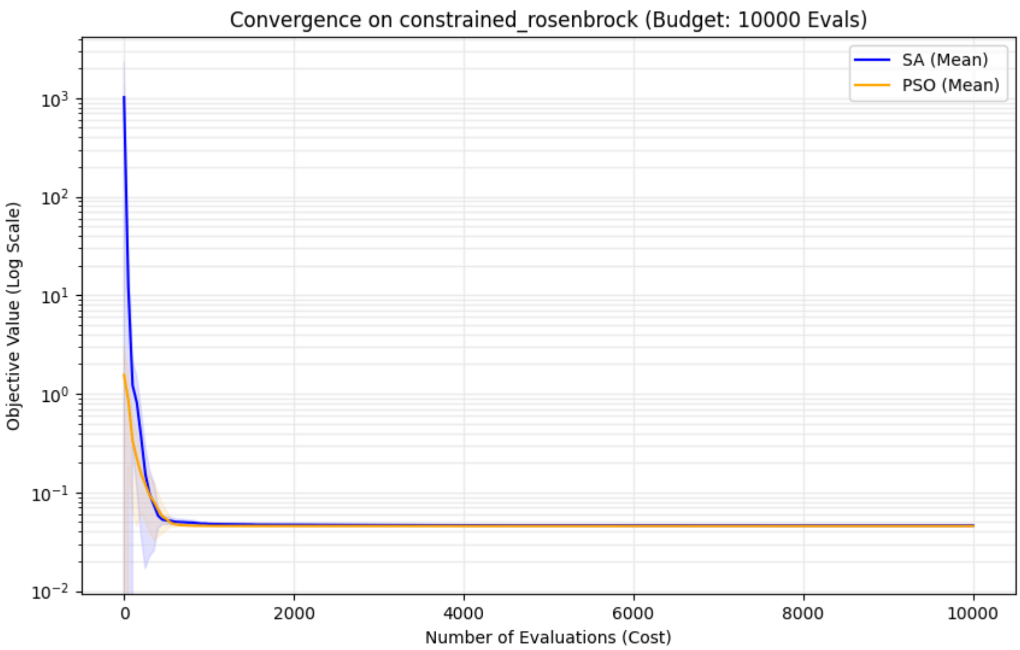
We performed a sensitivity analysis on the Constrained Rosenbrock problem by varying the Penalty Factor R (using 5,000 evaluations).

Penalty Factor (R)	Mean Best Value	Mean Constraint Violation	Interpretation
1	4.2387e-02	5.4226e-02	Under-penalized: Algorithm accepts high violation to lower objective. Infeasible.
10	4.5310e-02	5.9991e-03	Better, but significant violation remains.
100	4.5638e-02	6.0671e-04	Approaching feasibility.
1000	4.5671e-02	6.0741e-05	Optimal Balance: Good objective, negligible violation.
10000	4.5674e-02	6.0747e-06	Strict: Smallest violation, slightly higher objective score (harder to search).

There is a clear trade-off. A low R allows the algorithm to "cheat" by wandering into the infeasible region to find lower objective values. As R increases, the algorithm is forced strictly onto the boundary. R=1000 to R=10000 provides the most engineering-compliant solutions.

4.4 Convergence Analysis

Convergence plots confirm that PSO typically reduces the error by several orders of magnitude within the first 1,000 evaluations, whereas SA shows a slower, linear improvement on the log-scale. This improved convergence speed suggests that the "social" aspect of PSO accelerates the discovery of promising regions.



5. Limitations and Assumptions

This study is limited by the scope of the benchmark set. Only three benchmark problems were considered, and all were low-dimensional (two decision variables). While this allows the objective landscapes to be visualised and discussed clearly, it also means that conclusions should be interpreted as specific to these test cases. Behaviour on higher-dimensional engineering problems may differ due to larger search spaces and different landscape and feasibility structures.

Algorithm settings were kept fixed during the experiments rather than being extensively tuned for each benchmark. This limits what can be claimed about best-possible performance for either method. The comparison is therefore conditional on the specific parameter configurations used, and should be read as an evaluation under consistent settings rather than an optimisation of each algorithm’s hyperparameters.

Regarding performance measurement, elapsed computational time is dependent on the execution environment, so it is treated as a secondary indicator. The number of evaluations was used as the primary cost metric because it is more directly comparable across runs and aligns with the assumption that objective evaluations are the dominant computational expense.

Finally, constraint related reporting is not fully symmetric across algorithms. The penalty-factor sensitivity analysis was performed for PSO, but not repeated for SA in the same way. This limits the strength of any comparison regarding how each algorithm responds to changes in the penalty magnitude.

6. Conclusions

This comparative study evaluated the performance of Simulated Annealing (SA) and Particle Swarm Optimization (PSO) across unconstrained and constrained non-linear landscapes. Based on the experimental data obtained under a fixed budget of 10,000 evaluations, the following conclusions are drawn:

6.1 Algorithmic Efficiency and Convergence

PSO demonstrated superior efficiency, consistently converging to the optimum within the first 1,000 evaluations. In contrast, SA exhibited a slower, linear convergence rate on the log-scale. This confirms that for this specific set of low-dimensional problems, the population-based "social" mechanism of PSO accelerates the discovery of promising regions more effectively than the single-point trajectory approach of SA.

6.2 Robustness in Multi-Modal Landscapes

On the highly multi-modal Rastrigin function, PSO proved significantly more robust (Mean Best Value ~0.23) compared to SA (Mean ~17.8). While SA struggled to escape local basins of attraction within the evaluation limit, PSO's collective intelligence allowed it to avoid stagnation and locate the global minimum's vicinity reliably.

6.3 Constraint Handling and Sensitivity

The study confirms that a Static Penalty Function is an effective method for handling constraints in PSO, provided the penalty factor is sufficiently high. Sensitivity analysis on the Constrained Rosenbrock problem revealed a critical threshold: low penalty factors ($R < 100$) resulted in infeasible solutions ("cheating"), while factors of $R \geq 1000$ yielded an optimal balance of feasibility and objective reduction.

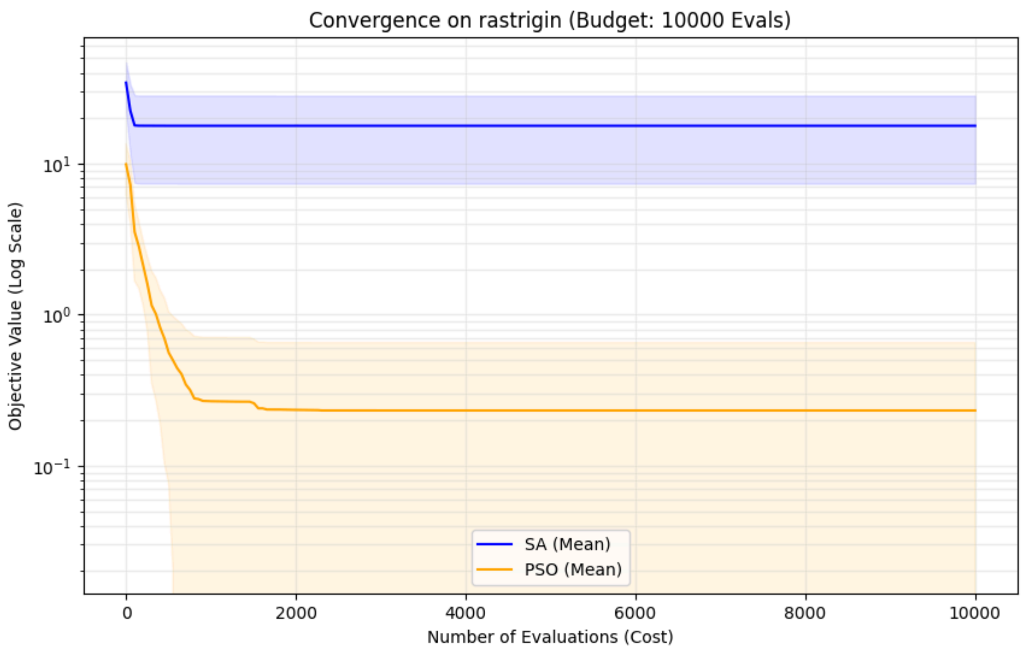
6.4 Recommendation for Future Tasks

Given its superior precision (Zero variance on Rosenbrock), faster convergence, and robust constraint handling, Particle Swarm Optimization is recommended as the primary algorithm for the upcoming engineering application in Task 6. However, computationally cheaper local search methods could be considered to hybridize with PSO to fine-tune solutions if higher dimensionality increases the computational cost.

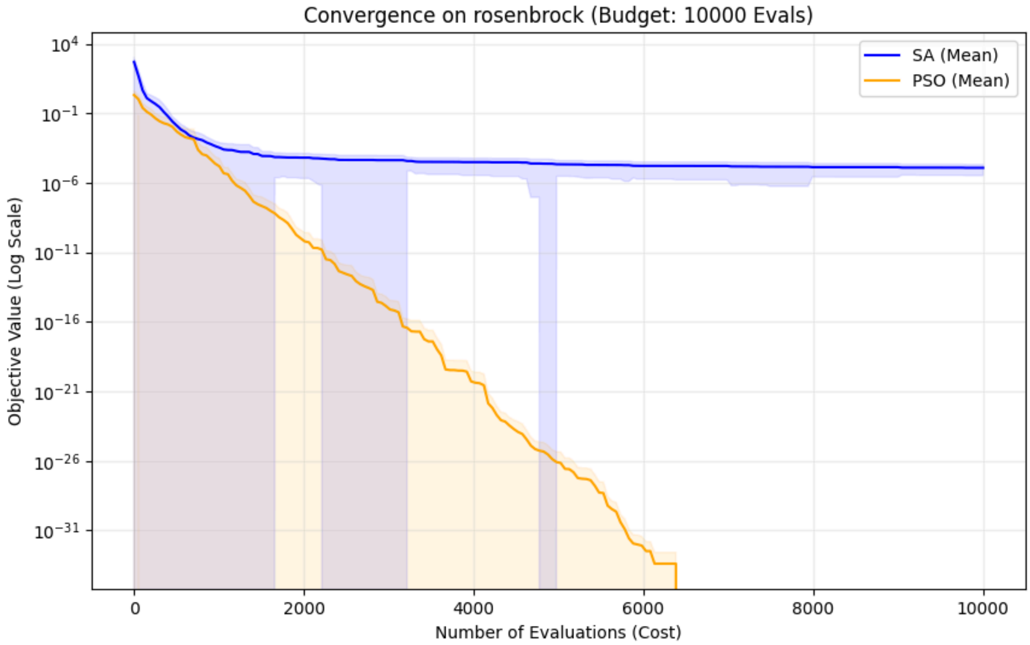
7. Appendices

7.1 Convergence Plots

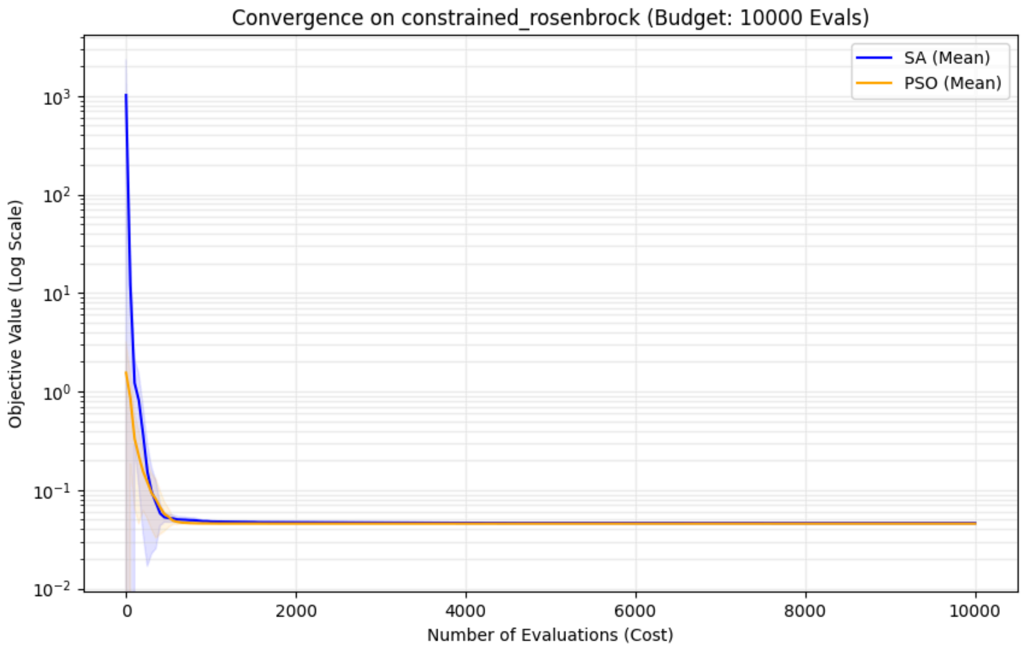
Rastrigin Convergence



Rosenbrock Convergence



Constrained Rosenbrock Convergence



7.2 Penalty Sensitivity Analysis Plot