Evaluating Guide Selection Strategies for Enhanced Particle Swarm Optimisation Performance

R.P Bosma 25857606
Dept. of Computer Science
Stellenbosch University
Stellenbosch, South Africa
25857606@sun.ac.za

***Abstract*—The particle swarm optimization (PSO) algorithm is a stochastic search technique based on the social dynamics of a flock of birds. This report investigates alternative guide selection strategies for PSO to balance exploration and exploitation. Two deterministic approaches and three inherently probabilistic strategies are compared within the standard inertia weight and the global best PSO framework. Deterministic strategies, such as the elitist approach and simulated annealing, converged faster but risked stagnation. Probabilistic methods, including roulette wheel, rank-based, and tournament selection, maintained diversity and performed better in complex landscapes but converged more slowly.**

***Index Terms*—Particle Swarm Optimisation, Guide Selection, Diversity, Convergence**

I. Introduction

Particle Swarm Optimisation (PSO) has emerged as an effective algorithm for balancing exploration and exploitation. PSO was originally developed by Kennedy and Eberhart in 1995 [1] as a method for simulating social behavior, particularly the movement of organisms in a bird flock or fish school. A central component of PSO is the guide selection process, which determines how particles update their personal best (pbest) and global best (gbest) positions.

Evaluating alternative guide selection strategies is essential because inappropriate choices may lead to premature convergence or inefficient exploration, significantly impacting optimisation performance. Understanding the relative strengths and limitations of deterministic and probabilistic strategies assists in making informed decisions tailored to specific optimisation problems.

This report evaluated two deterministic strategies within a standard inertia weight, global best PSO framework. These strategies included the conventional elitist approach and simulated annealing. Additionally, three probabilistic methods were examined. The evaluation considered problems categorised as separable, non-separable, and both separable and non-separable landscapes. Comparisons were made based on average global best values, convergence behaviour, and effects on swarm diversity. Empirical results indicated that deterministic, exploitative strategies converged rapidly but reduced swarm diversity, becoming frequently trapped in local optima. Probabilistic methods—particularly per-particle approaches, maintained higher diversity, beneficial in complex or high-dimensional problems.

The report is organised as follows. Section II provides background on PSO and clearly defines the exploration-exploitation trade-off. It also describes two main guide selection approaches: Approach A (global guide selection) and Approach B (per-particle guide). Section III details the methodology and experimental setup. Section IV presents experimental results, including comparisons of global best values, convergence behaviour, and diversity metrics.

II. Background

This section provides an overview of the fundamental principles of PSO. The discussion is organised as follows. First, the basic components of PSO are described, including the initialisation of a swarm in an n-dimensional search space, and the update mechanisms for particle positions and velocities. In particular, the roles of the personal best (pbest) and global best (gbest) solutions are explained. Second, the phenomenon of particle roaming and the associated repair strategies are discussed, detailing both position and velocity repair methods that prevent particles from permanently leaving the feasible search space.

A. Particle Swarm Optimization

In PSO, a swarm of candidate solutions, or particles, is initialised in an n-dimensional search space. Each particle has a position $x_i(t)$ and a velocity $v_i(t)$ at iteration t. The movement of each particle is primarily guided by two key attractors: the pbest and the gbest. Here, "best" refers to the solution yielding the lowest objective function value, assuming a minimisation problem. The pbest position, denoted $y_i(t)$, represents the best solution found by an individual particle and acts as a local attractor, encouraging the particle to explore its most promising region. In contrast, the gbest position, denoted $\hat{\mathbf{y}}(t)$, represents the best solution found by the entire swarm and serves as a global attractor, guiding particles towards the most promising region identified so far.

The velocity update for each dimension j is defined as follows [2]:

$$v_{ij}(t+1) = v_{ij}(t) + c_1\, r_{1j}(t)\Big[y_{ij}(t) - x_{ij}(t)\Big] + c_2\, r_{2j}(t)\Big[\hat{y}_j(t) - x_{ij}(t)\Big], \tag{1}$$

where c_1 and c_2 are positive acceleration coefficients, and $r_{1j}(t)$ and $r_{2j}(t) \sim U(0,1)^x$ are random numbers sampled for each dimension.

The position of each particle is then updated per dimension using

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{2}$$

The personal best position $\mathbf{y}_i(t)$ is updated based on the following criterion

$$\mathbf{y}_i(t+1) = \begin{cases} \mathbf{y}_i(t), & \text{if } f\left(x_i(t+1)\right) \geq f\left(y_i(t)\right), \\ \mathbf{x}_i(t+1), & \text{if } f\left(x_i(t+1)\right) < f\left(y_i(t)\right). \end{cases} \tag{3}$$

The global best position is computed based on the approach chosen, approach A is algorithm [4] and approach B is algorithm [5].

$$\begin{gathered} \hat{\mathbf{y}}(t) \in \{\mathbf{y}_0(t), \mathbf{y}_1(t), \ldots, \mathbf{y}_{n_s}(t)\} \quad \text{s.t.} \\ f\big(\hat{\mathbf{y}}(t)\big) = \min\{f(\mathbf{y}_0(t)), f(\mathbf{y}_1(t)), \ldots, f(\mathbf{y}_{n_s}(t))\} \end{gathered} \tag{4}$$

$$\hat{\mathbf{y}}(t) = \arg\min\{f(\mathbf{x}_0(t)), \ldots, f(\mathbf{x}_{n_s}(t))\} \tag{5}$$

Here, n_s denotes the number of particles in the swarm. The basic gbest PSO algorithm is summarised in Algorithm 1.

Algorithm 1 gbest PSO Algorithm

Create and initialise an n_x-dimensional swarm S.
repeat
  for each particle $i = 1, \ldots, S_{n_s}$ **do**
    if $f(S.x_i) < f(S.y_i)$ **then**
      Update $S.y_i \leftarrow S.x_i$;
    end if
    if $f(S.y_i) < f(S.\hat{y})$ **then**
      Update $S.\hat{y} \leftarrow S.y_i$;
    end if
  end for
  for each particle $i = 1, \ldots, S_{n_s}$ **do**
    Update the velocity;
    Update the position;
  end for
until a stopping condition is true

For further information on the theoretical foundations and parameter settings of PSO, readers are referred to the works of Kennedy and Eberhart [1], Eberhart and Shi [3], and Engelbrecht [2].

B. Particle Roaming Behaviour and Repair Strategies

Roaming behaviour, as defined by Engelbrecht [2], refers to particles exiting the search space during initial iterations if not constrained. These particles are eventually returned to the feasible search space, incurring additional computational effort. Engelbrecht [4] elaborates that roaming can result in infeasible solutions due to boundary violations.

Two main categories of roaming repair strategies exist. Position repair strategies modify particle positions to adhere to boundary constraints, while velocity repair strategies adjust velocities to prevent repeated boundary violations.

III. Methodology

This section describes the implementation and evaluation procedures used for the five guide selection strategies within the PSO framework. The selected strategies were the standard elitist approach, simulated annealing, roulette wheel selection, tournament selection, and rank-based selection. Each strategy was applied to five benchmark functions, explicitly chosen to represent varying complexities in optimisation. The evaluation measured global best values and swarm diversity. Outcomes were predicted based on theoretical insights into convergence speed, global exploration, and exploration-exploitation trade-offs.

1) Strategy 1: Standard Elitist Approach: The standard elitist approach updated the pbest only when a new solution produced a lower objective function value. The gbest was the best pbest across all particles. Rapid convergence is predicted due to its greedy mechanism. Premature convergence will likely occur if the swarm identifies suboptimal solutions early. The pseudocode in Algorithm 2 summarises the standard elitist approach implemented.

Algorithm 2 Standard Elitist Approach

Input: current_value, candidate_value
if candidate_value $<$ current_value **then**
  return True (Accept candidate)
else
  return False
end if

2) Strategy 2: Simulated Annealing for pbest Update: Simulated annealing modified the elitist approach by probabilistically accepting inferior solutions based on a Boltzmann probability. Engelbrecht [2] recommended using an exponential cooling schedule with factors ranging between 0.95 and 0.99. With a cooling schedule selected as 0.99, it is expcted to produce low global best averages but with the risk of continual acceptance of inferior solutions. The pseudocode in Algorithm 3 summarises the simulated annealing approach implemented.

Algorithm 3 Simulated Annealing for *pbest* Update

Input: current_value, candidate_value, temperature
if candidate_value $<$ current_value **then**
  Accept candidate
else
  probability $\leftarrow \exp\left(-\frac{candidate\_value - current\_value}{temperature}\right)$
  Generate random number $r \in U(0,1)$
  if $r <$ probability **then**
    Accept candidate
  else
    Reject candidate
  end if
end if

3) Strategy 3: Roulette Wheel Selection for gbest: Roulette wheel selection assigned probabilities to each pbest proportional to its fitness. A balanced approach is expected to give better solutions a higher probability of selection while still retaining influence from less optimal solutions, thereby sustaining exploration capability. The pseudocode in Algorithm 4 summarises the roulette wheel selection approach implemented.

Algorithm 4 Roulette Wheel Selection for $gbest$

```
Calculate selection probabilities proportional to inverted fitness
Generate random number r ∈ U(0, 1)
cum_prob ← 0
for each candidate do
    cum_prob ← cum_prob + candidate's probability
    if r < cum_prob then
        Select candidate
    end if
end for
Return last candidate (if none previously selected)
```

4) Strategy 4: Tournament Selection for gbest: Tournament selection randomly drew a subset of pbest solutions and selected the solution with the lowest fitness as the gbest. Adjusting tournament size controlled selection pressure, with the current study setting this size at three, following guidelines from Lavinas et al. [8], the selected tournament size is expected to offer a suitable compromise between randomness and exploitation. The pseudocode in Algorithm 5 summarises the tournament selection approach implemented.

Algorithm 5 Tournament Selection

```
Input: candidates, fitnesses, tournament_size
Randomly select tournament_size indices (without replacement)
best ← candidate with the lowest fitness in the selected subset
return best
```

5) Strategy 5: Rank-Based Selection for gbest: Rank-based selection sorted all pbest solutions according to their fitness, assigning selection probabilities based on relative rankings. This selection strategy is expected to promote diversity and moderately fit solutions to retain a non-zero selection probability and promote broader solution representation in complex or multimodal landscapes. The pseudocode in Algorithm 6 summarises the rank-based selection approach implemented.

Algorithm 6 Rank-Based Selection for $gbest$

```
Rank candidates from best to worst based on fitness
Assign probabilities proportional to rank (highest rank = highest probability)
Generate random number r ∈ U(0, 1)
cum_prob ← 0
for each candidate in ranked order do
    cum_prob ← cum_prob + candidate's probability
    if r < cum_prob then
        return candidate
    end if
end for
return last candidate
```

6) Two Approaches for Strategies 3 to 5: Two guiding approaches were implemented for the probabilistic methods (roulette wheel, tournament, and rank-based selection): Approach A selected one global guide per iteration, forcing all particles towards the same gbest. Approach A is expected to accelerate convergence due to its use of a singular global guide, but will likely reduce diversity. Approach B assigned individual guides to each particle and is expected to foster exploration at the measurable cost of slower convergence speeds.

IV. Empirical Procedure

This section discussed the empirical procedure used to apply the implemented algorithms to the goals and hypotheses of the study. Performance measures were explained, and benchmark problems were described. All control parameters were provided, along with motivations for each choice. The number of independent runs was explicitly stated. This ensured that readers could replicate these experiments and achieve results similar to those reported here.

Five benchmark functions were selected to represent a range of optimisation challenges, including unimodal (U), multimodal (M), separable (S), and non-separable (NS) problems. Table I lists the selected functions and their properties. The Spherical and Ackley functions were tested in 30 dimensions to increase the complexity of the search space, while Booth and Rosenbrock were limited to 2 dimensions. Booth is naturally two-dimensional, and Rosenbrock remains unimodal for fewer than four dimensions. The Michalewicz function was evaluated in 10 dimensions. These dimensional choices were based on experimental standards compiled by Hussain et al. [7].

TABLE I
BENCHMARK FUNCTIONS

Function	Equation	Modality	Separability	Dimensions
$f1$: Spherical	Eq. (5)	U	S	30
$f2$: Booth	Eq. (6)	M	S	2
$f3$: Rosenbrock	Eq. (7)	U	NS	2
$f4$: Ackley	Eq. (8)	M	NS	30
$f5$: Michalewicz	Eq. (9)	M	S	10

Particle Swarm Optimisation (PSO) requires suitable control parameters to ensure sufficient exploration and convergence. Based on recommendations by Eberhart and Shi [3], the inertia weight was set to $w = 0.7$, and the acceleration coefficients were both set to 1.4. The maximum iteration count followed guidelines proposed by Si-Ma et al. [5], indicating that more iterations were necessary for higher dimensions. The iteration counts were set as follows:

- For small dimensions (dim < 10): $50 \times$ dim iterations
- For medium dimensions ($10 \leq$ dim < 30): $100 \times$ dim iterations
- For high dimensions (dim ≥ 30): $\max(5000, 100 \times \text{dim})$ iterations

Each benchmark function test comprised of 30 independent runs. During each run, the best fitness value was recorded. The median of these best fitness values represented typical performance, reducing the influence of outliers. The standard deviation of the best fitness values indicated consistency across runs. Swarm diversity was computed by averaging the mean Euclidean distance among all particles at each iteration. Approach B was used as the standard approach to compare strategies.

TABLE II
AVERAGE GLOBAL BEST AND STANDARD DEVIATION FOR EACH FUNCTION AND STRATEGY

Function	Strategy	Avg Global Best	Std Dev
Spherical (f1)	elitist	1.13148e-06	0.00334
	simulated_annealing	4.79107e-06	2999.99639
	roulette	4410.63478	1143.37612
	tournament	526.04420	375.07221
	rank	10118.02091	1033.96726
Booth (f2)	elitist	9.79802e-13	8.42117e-12
	simulated_annealing	3.80114e-13	3.49926e-12
	roulette	6.40888e-06	8.60341e-06
	tournament	2.86085e-08	5.39792e-08
	rank	0.00029826	0.00082243
Rosenbrock (f3)	elitist	1.40426e-04	2.07205
	simulated_annealing	1.82452e-05	0.41645
	roulette	0.27407	0.34912
	tournament	0.07582	1.47268
	rank	0.38591	0.72637
Ackley (f4)	elitist	4.52506	1.45511
	simulated_annealing	5.51729	3.03584
	roulette	11.67245	0.84538
	tournament	6.39604	0.95451
	rank	19.63110	1.11242
Michalewicz (f5)	elitist	-7.78860	0.66056
	simulated_annealing	-8.03088	0.79280
	roulette	-5.39346	0.37871
	tournament	-5.58769	0.44510
	rank	-5.17789	0.35553

V. RESEARCH RESULTS

This section discussed the experimental findings for the five guide selection strategies, with a focus on convergence behaviour and swarm diversity. The performance measures used were average global best fitness values and standard deviations.

A. Overall Average Global Best Results

Table II summarised the average global best values and standard deviations obtained for each benchmark function and strategy.

For function f_1, the elitist strategy achieved the lowest average global best value, indicating rapid convergence with minimal variance. Simulated annealing showed a similarly low average but an extremely high standard deviation. This high variance may result from the slow cooling schedule (0.99), which permits prolonged acceptance of inferior solutions and allowed it to get stuck in poorer regions. Probabilistic selection strategies demonstrated poorer performance; notably, the rank-based selection resulted in the highest average value and substantial variability.

The evaluation of function f_2 indicated that elitist and simulated annealing strategies produced excellent average global best values with low variability. In contrast, roulette wheel and rank-based methods yielded higher average values and standard deviations, indicating inferior convergence. Tournament selection demonstrated intermediate performance, achieving results between these two groups.

The analysis of f_3 showed simulated annealing and elitist strategies achieved superior average global best values, reflecting strong convergence performance and consistency. Roulette wheel and rank-based methods yielded noticeably higher averages and standard deviations, indicating inferior convergence performance and greater inconsistency. Tournament selection showed moderate performance between these extremes.

The evaluation of f_4 showed that the elitist method achieved the lowest average global best value, demonstrating efficient convergence; however, its variability remained notable. Simulated annealing and tournament selections closely followed but with greater variability. Roulette wheel and rank-based strategies delivered poorer performance, with higher averages and substantial variability.

For f_5, simulated annealing achieved the lowest average global best value, indicating effective convergence with moderate variability. The elitist strategy closely followed but exhibited slightly larger variability. Roulette wheel and rank-based selection strategies significantly underperformed, as indicated by their higher average values and lower consistency.

B. Swarm Diversity Trends

Swarm diversity trends, illustrated explicitly in Appendix B, highlighted distinct differences among strategies. The elitist and simulated annealing strategies typically yielded strong final fitness values on lower-dimensional functions such as Booth and Rosenbrock, yet rapidly reduced diversity, as observed in Figures 2 and 4. Conversely, rank-based and roulette wheel selections consistently preserved higher diversity throughout optimisation runs, evident from the plots for Michalewicz (Figure 1) and Spherical (Figure 5). Higher diversity enabled these methods to explore the search space more extensively, reducing the likelihood of premature convergence to local optima, especially valuable in multimodal and high-dimensional landscapes. This detailed comparison underlines the trade-off between convergence speed and exploration capacity across different strategy implementations.

C. Comparison of Approach A (Global Guide) vs. Approach B (Per-Particle Guide) for Probabilistic Methods

Tables III and IV summarise the average global best values and standard deviations for two guiding approaches. Approach A, a memory-based method employing a single global guide selected from historical pbest, inherently emphasised exploitation and typically showed faster convergence, as reflected by lower average global best values for simpler problems such as Booth and Rosenbrock functions. Approach B frequently updated the global best within each iteration, assigning individual iteration-based guides to particles, thus promoting diversity and greater exploration at the cost of slower convergence. Although Approach B maintained higher swarm diversity beneficial in challenging optimisation landscapes, it generally demonstrated poorer performance compared to Approach A.

TABLE III
APPROACH A (GLOBAL GUIDE): AVERAGE GLOBAL BEST AND STANDARD DEVIATION

Function	Strategy	Avg Global Best	Std Dev
Ackley (f4)	roulette	11.78072	1.17068
	tournament	6.64641	1.03027
	rank	19.22540	1.83688
Booth (f2)	roulette	4.63875e-06	1.13994e-05
	tournament	8.55882e-09	5.48311e-08
	rank	0.00013785	0.00043587
Rosenbrock (f3)	roulette	0.13059	0.34912
	tournament	0.07582	1.47268
	rank	0.44440	1.40700
Michalewicz (f5)	roulette	-5.73741	0.47690
	tournament	-5.67049	0.43290
	rank	-5.09150	0.38571
Spherical (f1)	roulette	3827.12966	979.31451
	tournament	616.70441	333.92219
	rank	10669.78128	1611.86513

TABLE IV
APPROACH B (PER-PARTICLE GUIDE): AVERAGE GLOBAL BEST AND STANDARD DEVIATION

Function	Strategy	Avg Global Best	Std Dev
Ackley (f4)	roulette	11.67245	0.84538
	tournament	6.39604	0.95451
	rank	19.63110	1.11242
Booth (f2)	roulette	6.40888e-06	8.60341e-06
	tournament	2.86085e-08	5.39792e-08
	rank	0.00029826	0.00082243
Rosenbrock (f3)	roulette	0.27407	0.34912
	tournament	0.07582	0.50358
	rank	0.38591	1.40700
Michalewicz (f5)	roulette	-5.39346	0.37871
	tournament	-5.58769	0.44510
	rank	-5.17789	0.35553
Spherical (f1)	roulette	4410.63478	1143.37612
	tournament	526.04420	375.07221
	rank	10118.02091	1033.96726

VI. CONCLUSION

This section revisited the primary objectives, summarised the methodology, and highlighted the key findings of the study. The main aim was to investigate alternative guide selection strategies in Particle Swarm Optimisation (PSO) and to analyse their effects on convergence behaviour and swarm diversity. Five distinct strategies were implemented: the standard elitist method, simulated annealing, roulette wheel selection, tournament selection, and rank-based selection. These strategies were evaluated using five benchmark functions, each chosen to represent combinations of unimodal, multimodal, separable, and non-separable characteristics.

Results generally indicated that deterministic approaches, such as elitist selection, offered faster convergence but reduced swarm diversity. In contrast, probabilistic methods, including roulette wheel and rank-based selections, maintained greater diversity but converged more slowly.

Overall, this study demonstrated that different guide selection methods balanced exploration and exploitation in varying degrees. It reinforced the importance of selecting appropriate strategies based on specific problem characteristics such as dimensionality, modality, and separability. Future research could explore adaptive mechanisms that dynamically switch between guide selection strategies or adjust parameters such as cooling factors, tournament sizes, or rank-based probabilities. Expanding the variety and dimensionality of benchmark functions could further validate these findings. This study highlighted the critical role of guide selection mechanisms in effectively maintaining convergence quality and swarm diversity in Particle Swarm Optimisation.

REFERENCES

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, pp. 1942–1948. doi: 10.1109/ICNN.1995.488968.

[2] A. P. Engelbrecht, *Fundamentals of Computational Swarm Intelligence*. Hoboken, NJ, USA: Wiley, 2007.

[3] R. C. Eberhart and Y. Shi, "Comparing inertia weights and constriction factors in particle swarm optimization," in *Proceedings of the 2000 Congress on Evolutionary Computation (CEC00)*, vol. 1, 2000, pp. 84–88. doi: 10.1109/CEC.2000.870279.

[4] A. P. Engelbrecht, "Roaming behavior of unconstrained particles," in *Proceedings - 1st BRICS Countries Congress on Computational Intelligence, BRICS-CCI 2013*, 2013, pp. 104–111. doi: 10.1109/BRICS-CCI-CBIC.2013.28.

[5] S. Si-Ma, H. M. Liu, H. X. Zhan, Z. F. Liu, G. Guo, C. Yu, and P. C. Hu, "Efficient maximum iterations for swarm intelligence algorithms: a comparative study," *Artificial Intelligence Review*, vol. 58, no. 3, p. 87, Jan. 2025. doi: 10.1007/s10462-024-11104-7.

[6] P. Suganthan, N. Hansen, J. Liang, K. Deb, Y.-P. Chen, A. Auger, and S. Tiwari, "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," *Natural Computing*, vol. 341–357, 2005.

[7] K. Hussain, M. N. Mohd Salleh, S. Cheng, and R. Naseem, "Common benchmark functions for metaheuristic evaluation: A review," *International Journal on Informatics Visualization*, vol. 1, no. 4-2, 2017. ISSN: 2549-9610, eISSN: 2549-9904.

[8] Y. Lavinas, C. Aranha, T. Sakurai, and M. Ladeira, "Experimental analysis of the tournament size on genetic algorithms," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2018, pp. 3647–3653. doi: 10.1109/SMC.2018.00617.

Appendix A: Benchmark Functions

A. *Spherical*

$$f(x) = \sum_{i=1}^{n} x_i^2 \tag{6}$$

with $x_i \in [-100, 100]$, for which $f(x^*) = 0$, $x^* = (0, \ldots, 0)$.

B. *Booth*

$$f(x) = (x_1 + 2x_2 - 7)^2 + (2x_1 + x_2 - 5)^2 \tag{7}$$

with $x_i \in [-10, 10]$, for which $f(x^*) = 0$ for $x^* = (1, 3)$.

C. *Rosenbrock*

$$f(x) = \sum_{i=1}^{n-1} \left[100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2\right] \tag{8}$$

with $x_i \in [-30, 30]$, for which $f(x^*) = 0$, and $x^* = (1, \ldots, 1)$.

D. *Ackley*

$$f(x) = -20e^{-0.2\sqrt{\frac{1}{n}\sum_{j=1}^{n} x_j^2}} - e^{\frac{1}{n}\sum_{j=1}^{n}\cos(2\pi x_j)} + 20 + e \tag{9}$$

with $x_i \in [-32, 32]$, for which $f(x^*) = 0.0$, and $x^* = (0, \ldots, 0)$.

E. *Michalewicz*

$$f(x) = -\sum_{i=1}^{n} \sin(x_i)\left(\sin\left(i x_i^2 \pi\right)\right)^{2m} \tag{10}$$

where $m = 10$ and with $x_i \in [0, \pi]$, for which $f(x^*) = -0.966n$, and $x^* = (0, \ldots, 0)$.

APPENDIX B: DIVERSITY PLOTS

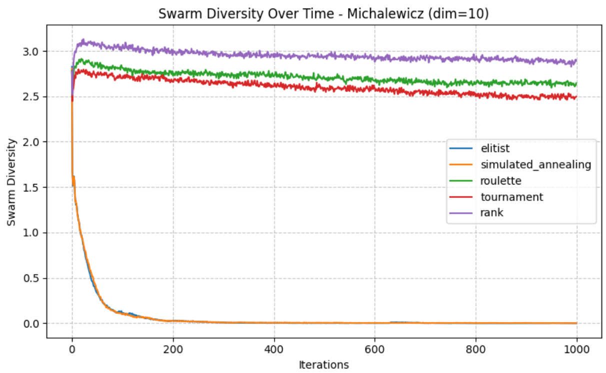

Fig. 1. Approach B — Michalewicz

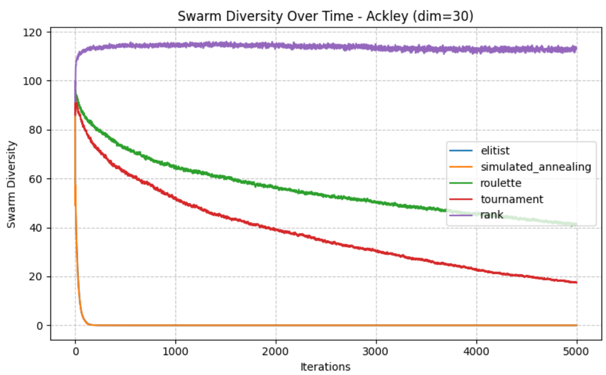

Fig. 2. Approach B — Ackley

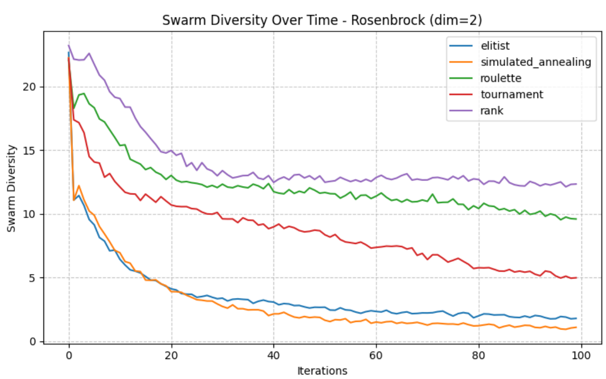

Fig. 3. Approach B — Rosenbrock

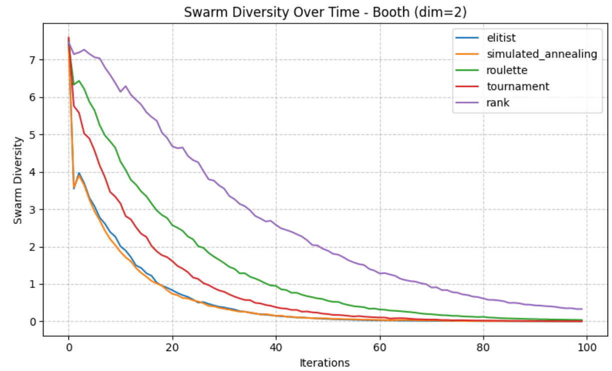

Fig. 4. Approach B — Booth

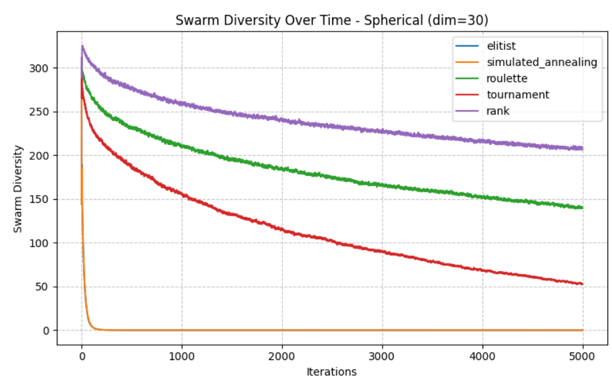

Fig. 5. Approach B — Spherical

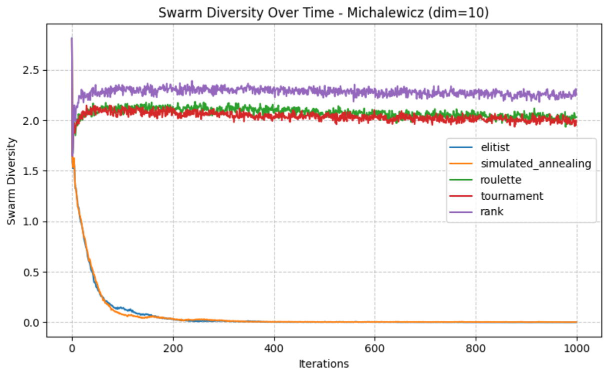

Fig. 6. Approach A — Michalewicz

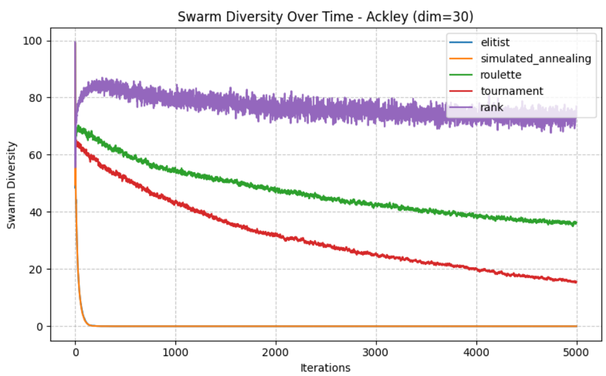

Fig. 7. Approach A — Ackley

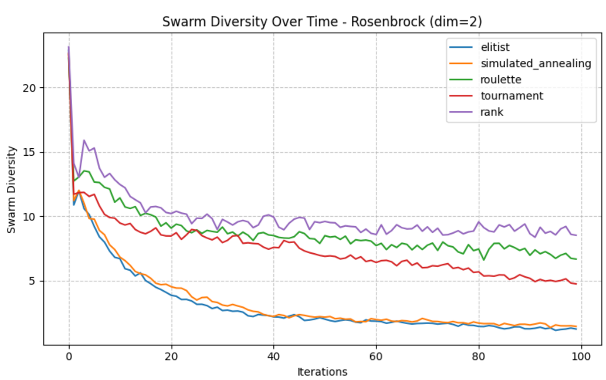

Fig. 8. Approach A — Rosenbrock

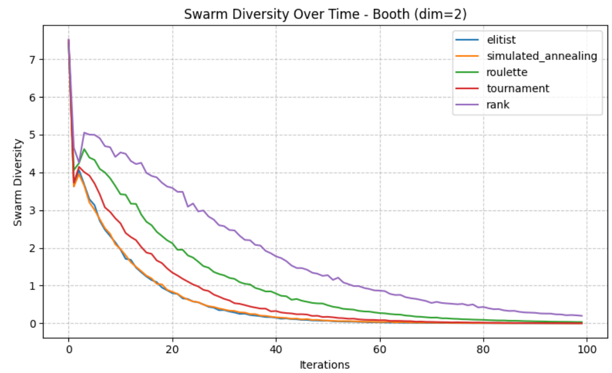

Fig. 9. Approach A — Booth

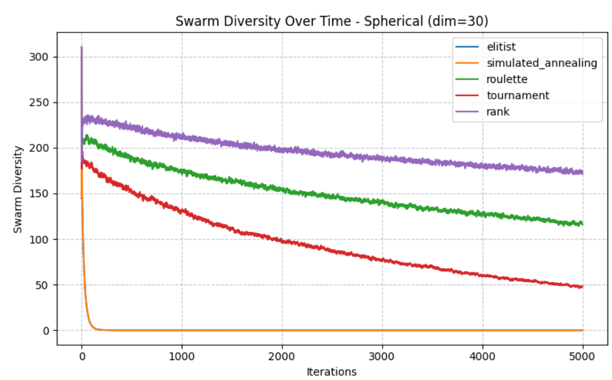

Fig. 10. Approach A — Spherical