Available online at website: https://jurnal.iaii.or.id/index.php/RESTI

JURNAL RESTI
(Rekayasa Sistem dan Teknologi Informasi)
Vol. 10 No. 3 (2026) 794 - 803 e-ISSN: 2580-0760

Convergence and Empirical Performance of Tanh-Based Adaptive Particle Swarm Optimization

Joko Riyono[1], Aina Latifa Riyana Putri[2*], Sofia Debi Puspa[3], Supriyadi[4], Christina Eni Pujiastuti[5], Fayza Nayla Riyana Putri[6]
[1,3,4,5]Department of Mechanical Engineering, Faculty of Industrial Technology, Trisakti University, Jakarta, Indonesia
[2]Data Science, Telkom University, Purwokerto, Indonesia
[6]Department of Artificial Intelligence, Faculty of Computer Science and Digital Technology, Muhammadiyah University of Semarang, Indonesia.

[*]*ainaqp@telkomuniversity.ac.id*

Abstract

Particle Swarm Optimization (PSO) is a widely used population-based optimization method but faces challenges in premature convergence, leading to suboptimal solutions. To address this issue, this study proposes a Tanh-Based Acceleration Coefficient PSO (TB-PSO), where the acceleration coefficients are modified using the hyperbolic tangent (tanh) function. The smooth and continuous behavior of tanh enables gradual coefficient updates, limits excessive particle velocities, and maintains swarm diversity, thereby improving convergence stability and balancing exploration and exploitation. The convergence theorem analysis confirms that TB-PSO meets stability criteria before being evaluated on unimodal and multimodal benchmark functions in 10 and 30 dimensions. Its performance is compared against several PSO variants, including TVAC-PSO, SCAC-PSO, NDAC-PSO, and SAC-PSO. In the 10-dimensional experiments, TB-PSO achieves the best overall final ranking based on the average and standard deviation of best solution, ranking first for functions f_3 and f_5, second for f_2 with only a marginal difference from the best-performing method, and remaining competitive for f_1 and f_4. These results indicate superior solution quality and stable convergence. For the 30-dimensional benchmark functions, TB-PSO ranks first for f_2, second for f_5, and third for f_1, f_3, and f_4 based on the same evaluation criteria. Although its ranking decreases compared to the 10-dimensional case, TB-PSO remains competitive, reflecting the increased complexity of high-dimensional optimization problems. Overall, the results demonstrate that the tanh-based acceleration coefficient modification effectively enhances PSO performance, particularly in lower-dimensional search spaces, while maintaining robustness in higher-dimensional scenarios.

Keywords: acceleration coefficient; convergence; particle swarm optimization; tanh function

How to Cite: J. Riyono, A. Latifa Riyana Putri, S. D. Puspa, S. Supriyadi, C. E. Pujiastuti, and F. N. Riyana Putri, "Convergence and Empirical Performance of Tanh-Based Adaptive Particle Swarm Optimization", *J. RESTI (Rekayasa Sist. Teknol. Inf.)* , vol. 10, no. 3, pp. 794 - 803, Jun. 2026.
*Permalink/*DOI: https://doi.org/10.29207/resti.v10i3.7247

Received: October 6, 2025
Accepted: June 1, 2026
Available Online: June 24, 2026

This is an open-access article under the CC BY 4.0 License
Published by Ikatan Ahli Informatika Indonesia

1. Introduction

Particle Swarm Optimization (PSO) is a population-based optimization technique that mimics the cooperative movement patterns observed in groups of birds and schools of fish. Within this framework, a swarm of particles explores the search space to locate optimal solutions. Each particle, acting as a candidate solution, maintains a position and a velocity vector. These attributes are iteratively modified by incorporating both the individual's historical best performance (pbest) and the global optimum achieved by the entire population (gbest).

Particle Swarm Optimization (PSO) has been extensively utilized in solving optimization problems because of its capability to effectively search for optimal solutions through a population of interacting particles [1]. This capability enables PSO to perform computationally efficient searches and maintain strong performance when addressing high-dimensional optimization problems, compared with algorithms such as Bayesian Optimization [2], Grid Search [3], and Random Search [4]. However, one of the main challenges of PSO is convergence, as particles in the swarm may become trapped in local optima or undergo inefficient exploration [5] highlighted that PSO has a tendency to generate divergent trajectories, indicating insufficient convergence. Additionally, PSO particles often converge prematurely to a stable state, leading to suboptimal solutions trapped in local optima [6], [7].

Consequently, numerous studies have been conducted to enhance the performance of PSO, particularly by modifying the acceleration coefficients (C_1 and C_2), which are essential for maintaining an appropriate balance between exploration and exploitation during the search process [8]. Proper adjustment of these coefficients can improve convergence speed and solution quality.

Earlier studies introduced a modification of the acceleration coefficients by assigning unequal values ($C_1 = 0.5$ and $C_2 = 2.5$), resulting in improved convergence behavior for certain optimization tasks [9]. In addition, dynamically adjusting the acceleration coefficients at each iteration was reported to enhance the algorithm's capability to obtain the global optimum solution [10]. To address this, several approaches have been introduced using both linear and nonlinear functions to adaptively adjust acceleration coefficients throughout the iterations, including Sigmoid-Based Acceleration Coefficients (SBAC) [8], Nonlinear Dynamics Acceleration Coefficients (NDAC), Time-Varying Acceleration Coefficients (TVAC) [11], and Sine-Cosine Acceleration Coefficients (SCAC) [12]. These modifications allow for adaptive changes in acceleration coefficients during the optimization process, improving PSO's convergence and solution quality. Among these, the Sigmoid-Based Acceleration Coefficients have shown particularly strong performance. In previous studies, this method consistently outperformed others across several benchmark functions and was ranked first overall in final performance evaluations [8]. Despite these promising results, the nature of the sigmoid function still introduces relatively abrupt transitions in acceleration values during iterations, which may affect the stability of the swarm's movement in more complex search landscapes. Recognizing this limitation, this study explores whether a smoother functional form could further enhance the adaptive behavior of PSO.

To this end, this research proposes the use of the hyperbolic tangent (tanh) function as the basis for modifying acceleration coefficients. The smooth and continuous characteristics of the tanh function allow acceleration coefficients to change more gradually across iterations, thereby helping preserve swarm diversity and reducing the risk of premature convergence to local optima [13], [14]. Additionally, the inherent properties of tanh, which map input values to an output range of -1 to 1, help prevent excessive particle velocity updates, thereby maintaining stability throughout the optimization process [11].

This study introduces a modification to the PSO hyperparameters by defining the acceleration coefficients (C_1 and C_2) using a hyperbolic tangent (tanh)-based scheme. The effectiveness of the proposed hyperparameter modification in achieving convergence of the modified PSO algorithm will be evaluated using the convergence criteria theorem from [5]. To evaluate the effectiveness of the proposed PSO modification, comparisons will be made with several existing approaches, including SBAC-PSO, NDAC-PSO, TVAC-PSO, SCAC-PSO. Performance evaluation will be conducted using benchmark unimodal and multimodal functions to assess the method's effectiveness under different optimization scenarios and to provide a more thorough numerical analysis. The assessment considers important performance indicators, including the mean and standard deviation of the obtained best solutions from several independent runs. Unimodal functions are used to measure how quickly the method reaches the optimal solution in a relatively simple landscape, while multimodal functions are used to assess its capability to avoid local optima within more complex search environments.

Furthermore, this study will also investigate the influence of parameters within the tanh function on the solution search dynamics of PSO to understand the extent to which the tanh function can adapt to the characteristics of different optimization problems. Through this approach, deeper insights into the acceleration mechanism in PSO can be obtained, along with recommendations for optimal hyperparameter tuning across various optimization problem types. Additionally, this study can serve as a reference for researchers and developers of swarm intelligence-based optimization algorithms in addressing challenges related to convergence and solution exploration more effectively.

2. Methods

Based on the limitations discussed in the previous section, This study introduces a modified Particle Swarm Optimization (PSO) algorithm by incorporating a hyperbolic tangent (tanh)-based approach for adjusting the acceleration coefficients. The objective of this modification is to enhance the convergence performance of PSO by achieving a better balance between exploration and exploitation throughout the optimization process. Before evaluating the performance of this proposed modification, it is important to first examine the mathematical convergence criteria for PSO and how these criteria are associated with parameter configurations, including the acceleration coefficients (C_1 and C_2) and inertia weight (ω).

The methodology section describes the analytical approaches utilized in this study in a sequential manner, as presented in Figure 1. This includes the analysis of PSO convergence criteria, formulation of the proposed tanh-based modification, experimental design using benchmark functions, and performance comparison with other acceleration coefficient strategies.

2.1 Analysis of Convergence Criteria for PSO Parameters

Particle Swarm Optimization (PSO) was initially introduced by [15] as an algorithm for optimizing its continuous nonlinear functions.

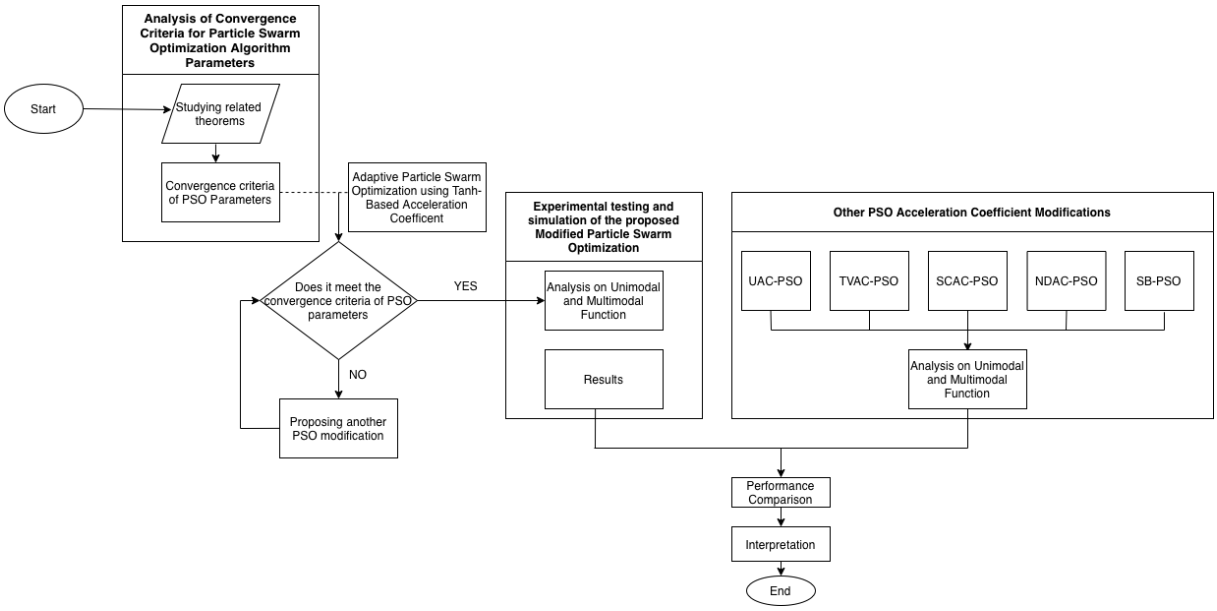


Figure 1. Flowchart of the research

In PSO, each particle independently searches for its best position (Pbest) while incorporating the optimal position identified by the entire swarm (Gbest). This allows the algorithm to converge toward an optimal solution. Each particle shares information about its best position with other particles and adjusts its position and velocity accordingly based on the received information. This indicates that the position and velocity of each particle are iteratively adjusted throughout the search process. The iteration stops when a stopping criterion is met, or when the convergence value is achieved. The position and velocity of each particle are updated according to Equations 1 and 2.

$$x_i^{(t+1)} = v_i^{(t+1)} + x_i^{(t)} \quad (1)$$

$$v_i^{(t+1)} = \omega\, v_i^{(t)} + c_1 r_1 \left(p_{(i,lb)}^{(t)} - x_i^{(t)}\right) + c_2 r_2 \left(p_{gb}^{(t)} - x_i^{(t)}\right) \quad (2)$$

$x_i^{(t+1)}$ is the position of particle i at iteration t+1; $v_i^{(t+1)}$ represents the velocity of particle i at iteration t+1; $x_i^{(t)}$ represents the position of particle i at iteration t; $v_i^{(t)}$ represents the velocity of particle i at iteration t; c_1 and c_2 represent the acceleration coefficients, which are assigned a value of 2 in the standard PSO algorithm; ω is the inertia weight; $p_{(i,lb)}^{(t)}$ is the personal best position; r_1 and r_2 are random values [0,1]; $p_{gb}^{(t)}$ is the gbest.

The original PSO algorithm has several drawbacks, as highlighted by studies such as [6], [7]. These studies demonstrate that PSO particles may experience premature convergence toward a stable state, causing the search process to become trapped in local optimal solutions. Furthermore, PSO convergence has been found to be influenced by the proper balance between exploration and exploitation capabilities, allowing particles to effectively approach the global optimal solution [16].

The following theorem presents an analysis of the convergence behavior of PSO toward the global optimal solution.

Theorem 1. A particle in the Particle Swarm Optimization (PSO) algorithm has been proven to converge toward a stable point given by $\frac{c_1 p_{(i,lb)}^{(t)} + c_2 p_{gb}^{(t)}}{c_1 + c_2}$, provided that $\max\{|\lambda_1|, |\lambda_2|\} < 1$ where λ_1 and λ_2 represent the eigenvalues that characterize the dynamic behavior of a basic PSO system with an inertia weight(ω) [5] .

Based on this theorem, if ω, c_1, c_2 are chosen in such a way: $\frac{c_1+c_2}{2} - 1 < \omega$ and $0 < c_1 + c_2$ that the condition $max\{||\lambda_1||, ||\lambda_2||\} < 1$ is satisfied, then the system guarantees convergence to the stable point. It can be seen that the parameter settings used in the original PSO, namely $c_1 = c_2 = 2$, and $w = 1$ do not fulfill the convergence conditions required for PSO parameters. This occurs because $c_1 + c_2 = 2 + 2 = 4 > 0$ and $\frac{c_1+c_2}{2} - 1 = \frac{2+2}{2} - 1 = 1 = w$. These conditions indicate that the original PSO formulation may generate a divergent particle trajectory [17]. Consequently, this issue creates limitations in applying the original PSO to practical optimization problems. The divergence of particle trajectories indicates that the standard PSO may not consistently achieve satisfactory convergence performance. To overcome this limitation, this study investigates the modification of acceleration coefficients to establish a more effective balance between exploration and exploitation capabilities. At this stage, we will examine whether the proposed tanh-based acceleration coefficient modification for PSO, as shown below, satisfies the PSO convergence criteria stated in Theorem 2.1. Specifically, we aim to verify whether $max\{||\lambda_1||, ||\lambda_2||\} < 1$ ensuring that the

particle search process toward personal best and global best is guaranteed to achieve convergence.

2.2 Performance Assessment and Simulation Analysis of the Proposed TANH-PSO

This study also conducts an empirical analysis through a set of experimental evaluations to assess the performance of the proposed acceleration coefficient modification. The experiments are conducted by implementing the modified PSO algorithm on a set of standard benchmark optimization functions. Testing is performed using two main categories of objective functions: unimodal functions and multimodal functions. The selected functions are intended to evaluate the convergence performance and effectiveness of the proposed method in obtaining optimal solutions under various optimization conditions.

Unimodal functions are optimization functions that contain only one global optimum, making them appropriate for analyzing the convergence speed of an algorithm. In this regard, an algorithm with higher efficiency is expected to achieve the optimal solution within fewer iterations. The Sphere Function using Equation 3.

$$\min\ f_1(x) = \sum_{i=1}^{n} x_i^2 \quad (3)$$

The global optimal solution is positioned at $x^* = 0$ with $f(x^*) = 0$, and the search space is bounded by $-10 \leq x_i \leq 10$.The Schwefel Function using Equations 4.

$$\min\ f_2(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i| \quad (4)$$

where The global optimal solution is positioned at $x^* = 0$ with $f(x^*) = 0$, and the search space is bounded by $-100 \leq x_i \leq 100$.The Rosenbrock Function using Equation 5.

$$\min f_3(x) = \sum_{i=1}^{n} [100\,(x_i^2 - x_{i+1})^2 + (x_i - 1)^2] \quad (5)$$

The global optimal solution is positioned at $x^* = (1,1,\ldots,1)$with $f(x^*) = 0$, and the search domain is constrained within the range defined by $-30 \leq x_i \leq 30$. These benchmark functions are commonly used to evaluate the exploitation ability, convergence characteristics, and solution accuracy of metaheuristic algorithms in unimodal optimization environments.

Multimodal functions have multiple local optima, making them more challenging for optimization algorithms. The primary purpose of this test is to assess the capability of PSO in avoiding local optima and identifying the global optimal solution. The multimodal functions used in this study are the Griewank Function and Ackley Function.The Griewank Function using Equations 6.

$$min\ f_4(x) = \frac{1}{4000} \sum_{i=1}^{n} (x_i)^2 - \prod_{i=1}^{d} \cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (6)$$

The global optimal solution is positioned at $x^* = 0$with $f(x^*) = 0$, and the search space is bounded by $-600 \leq x_i \leq 600$.The Ackley Function using Equations 7.

$$min\ f_6(x) = -20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)\right) + 20 + e \quad (7)$$

The global optimum occurs at $x^* = 0$ with $f(x^*) = 0$, and the search space is bounded by $-600 \leq x_i \leq 600$. These multimodal benchmark functions are characterized by multiple local optima, making them appropriate for assessing the exploration ability and robustness of optimization algorithms in preventing premature convergence and reaching the global optimum.

2.3. Performance Comparison with Other PSO Acceleration Coefficient Modifications

At this stage, a performance comparison is carried out between the proposed modification of PSO, namely the Tanh-Based Acceleration Coefficient, and several existing PSO variants. The Unbalanced Acceleration Coefficient PSO (UACPSO) was proposed by [6] by modifying the cognitive and social acceleration parameters (C_1 and C_2) to enhance the convergence speed of PSO. In their implementation, the acceleration coefficients were assigned fixed and unbalanced values, with $C_1 = 0.5$and $C_2 = 2.0$. TVAC-PSO, proposed in [9], employs a dynamic adjustment strategy for the cognitive and social coefficients during the optimization process. In this approach, the cognitive coefficient (C_1) gradually decreases, while the social coefficient (C_2) increases as the iteration progresses. The corresponding equations are shown in Equations 8 and 9.

$$C_1 = \left(C_{1f} - C_{1i}\right)\frac{iter}{MAXITR} + C_{1i} \quad (8)$$

$$C_2 = \left(C_{2f} - C_{2i}\right)\frac{iter}{MAXITR} + C_{2i} \quad (9)$$

C_{1f}, C_{1i}, C_{2f}, and $C_{2i}$are predefined constants, $iter$ indicates the current iteration, and $MAXITR$ refers to the predefined maximum number of iterations. Inspired by TVAC-PSO and the work reported in [9], SCAC-PSO utilizes trigonometric functions to control the cognitive and social components throughout the optimization process. The acceleration coefficients are calculated using Equations 10 and 11.

$$C_1 = \partial\ x \sin\left(\left(1 - \frac{iter}{MAXITR}\right) x\ \frac{\pi}{2}\right) + \delta \quad (10)$$

$$C_2 = \partial\ x \cos\left(\left(1 - \frac{iter}{MAXITR}\right) x\ \frac{\pi}{2}\right) + \delta \quad (11)$$

$\alpha$and $\delta$are constants with values $\alpha = 2$and $\delta = 0.5$. NDAC-PSO incorporates nonlinear adjustment strategies for the acceleration coefficients to achieve a more effective balance between exploration and exploitation. The cognitive and social coefficients are updated according to Equations 12 and 13.

$$C_1 = -\left(C_{1f} - C_{1i}\right)\left(\frac{iter}{MAXITR}\right)^2 + C_{1f} \quad (12)$$

$$C_2 = C_{1i}\ x\ \left(1 - \frac{iter}{MAXITR}\right)^2 + C_{1f}\ x\ \frac{iter}{MAXITR} \quad (13)$$

C_{1f} and C_{1i} denote positive constants with values of 2.5 and 0.5, respectively, $iter$ represents the current iteration, and $MAXITR$ indicates the maximum iteration limit. SBAC-PSO utilizes a sigmoid function to generate nonlinear variations of the acceleration coefficients throughout the optimization process. The coefficients are determined using Equations 14 and 15.

$$C_1 = \frac{1}{1+e^{(-\lambda.\frac{iter}{MAXITR})}} + 2(C_{1f} - C_{1i})(\frac{iter}{MAXITR} - 1)^2 \quad (14)$$

$$C_2 = \frac{1}{1+e^{(-\lambda.\frac{iter}{MAXITR})}} + (C_{1f} - C_{1i})(\frac{iter}{MAXITR})^2 \quad (15)$$

λ represents a control parameter that adjusts the sigmoid-based acceleration coefficient, with a value of 0.0001. The parameters C_{1f} and C_{1i} are positive constants assigned values of 2.5 and 0.5, respectively. The terms $iter$ and $MAXITR$ represent the current iteration and the maximum number of iterations. These PSO variants are utilized as comparison methods to assess the performance of the proposed Tanh-Based Acceleration Coefficient based on optimization robustness, solution quality, and convergence speed. The experiments were conducted by applying each method to the unimodal and multimodal benchmark functions described previously. The results generated by each PSO variant were subsequently analyzed to compare their convergence behavior and the stability of the obtained solutions. Through this comparative evaluation, the study aims to identify the capability of the proposed acceleration coefficient modification in enhancing PSO performance across various optimization problems.

3. Results and Discussions

Although the proposed tanh-based approach is designed to enhance the balance between exploration and exploitation, adjustments to PSO parameters must also maintain the convergence characteristics of the algorithm. Therefore, the next subsection analyzes the convergence conditions of the proposed approach to ensure that the resulting particle dynamics remain stable.

3.1 Tanh-Based Acceleration Coefficient

The performance of PSO is strongly influenced by its ability to maintain an appropriate balance between exploration and exploitation. Therefore, the right ratio between these two aspects must be carefully established. In general, PSO is expected to initially emphasize exploration to investigate a broad search space and progressively increase exploitation to improve the accuracy of the obtained optimal solution.

Based on this principle, various time-varying strategies have been developed to adjust PSO parameters, including the acceleration coefficients C_1 and C_2. As previously discussed, C_1 influences local exploration tendencies, while C_2 determines the level of global exploitation. Therefore, adaptive adjustment of these coefficients plays an important role in enhancing the balance between exploration and exploitation in PSO.

In this study, the proposed modification of the acceleration coefficients in PSO based on the tanh function is defined as Equations 16 and 17.

$$c_1(iter) = 0{,}5.\left(\tanh\left(2.(c_{1f} - c_{1i}).\left(\frac{iter\ max - iter}{iter\ max} - 0{,}5\right)\right) + 1\right).(c_{1f} - c_{1i}) + c_{1i} \quad (16)$$

$$c_2(iter) = 0{,}5.\left(\tanh\left(2.(c_{2f} - c_{2i}).\left(\frac{iter}{iter\ max} - 0{,}5\right)\right) + 1\right).(c_{2f} - c_{2i}) + c_{2i} \quad (17)$$

c_{1f}, c_{2f}, c_{1i}, and c_{2i} are positive constant coefficients, with (c_{1f} and $c_{2f} = 2{,}5$; c_{1i} and $c_{2i} = 0{,}5$). The term "iter" denotes the current iteration during the search process [18] , while 'iter max' refers to the maximum iteration count.

Figure 2 visualizes the plot of equations (16) and (17). The value of c_1 gradually decreases per iteration, ranging from 2.5 to 0.5, represented by the blue line. Meanwhile, the value c_2 increases per iteration, ranging from 0.5 to 2.5, shown by the yellow line.

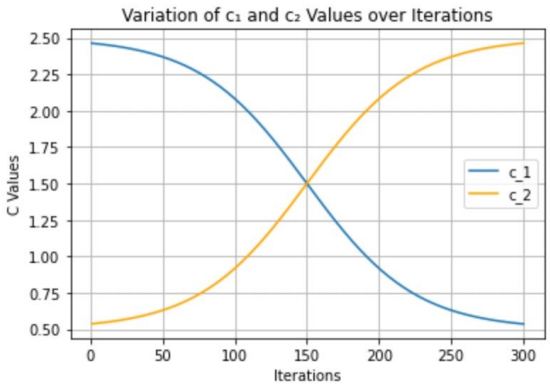


Figure 2. Visualization of c_1 and c_2 Values per Iteration

The proposed approach facilitates a gradual transition between the exploration and exploitation phases. In the initial phase of the search process, c_1 is assigned a relatively high value to encourage particles to perform broader exploration of the search space. Conversely, c_2 begins with a lower value, preventing premature exploitation. As the iterations progress, c_1 gradually decreases while c_2 increases [18]. This dynamic adjustment is designed to create a nonlinear and more natural transition from exploration to exploitation compared to linear-based approaches.

3.2 Convergence Analysis of Tanh-Based Acceleration Coefficients in Particle Swarm Optimization

As discussed earlier, the conditions required for each particle in PSO to converge to a stable point have been analyzed. Based on the proposed modification, the values obtained are $c_1 = 0{,}5$; $c_2 = 2{,}5$; w = 0.7298 [19], considering that c_1 and c_2 represent the upper bounds of φ_1 and φ_2. These values were verified using Theorem 1. The results demonstrate that the selected parameter values fulfill the convergence requirements of PSO, as they satisfy the specified

conditions $c_1 + c_2 = 0{,}5 + 2{,}5 = 3{,}5 > 0$ and $\frac{c_1+c_2}{2} - 1 = \frac{0{,}5+2{,}5}{2} - 1 = 1{,}5 - 1 = 0{,}5 < 0{,}7298$.

Further explanation can be obtained by calculating the value of $\max\{\|\lambda_1\|, \|\lambda_2\|\}$ using the equations for λ_1 and λ_2. As previously discussed, considering the stochastic components with $\varphi_1 = c_1 r_1$ and $\varphi_2 = c_2 r_2$, where $r_1, r_2 \sim U(0,1)$, it is evident that $0 < \varphi_1 + \varphi_2 < 3$ when $c_1 = 0{,}5$; $c_2 = 2{,}5$. Next, by substituting $\varphi = \varphi_1 + \varphi_2$ into the equations for λ_1 and λ_2, two sets of calculations are obtained: one for real values of γ and one for complex values of γ. Consider the case where γ is $\mathbb{R}$, when $\varphi \in [0;\ 0{,}021233349266117]$as shown in Equation 18.

$$max\ \{\|\lambda_1\|, \|\lambda_2\|\} \approx \frac{1{,}7298-\varphi \pm \sqrt{\varphi^2 - 3{,}4596\varphi + 0{,}073}}{4} \quad (18)$$

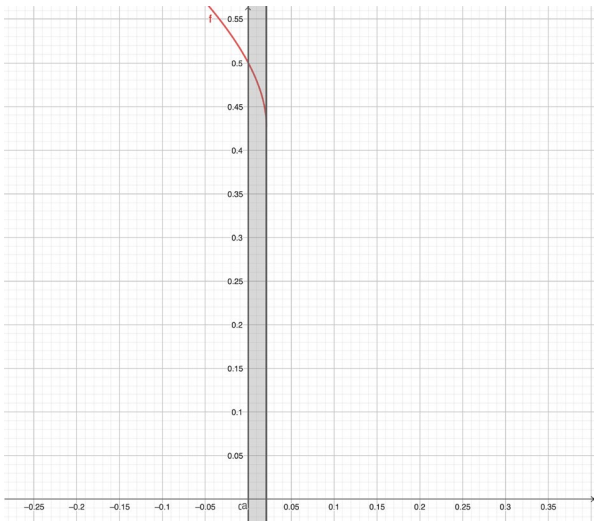


Figure 3a. Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for $\gamma \in \mathbb{R}$ (Positive Case)

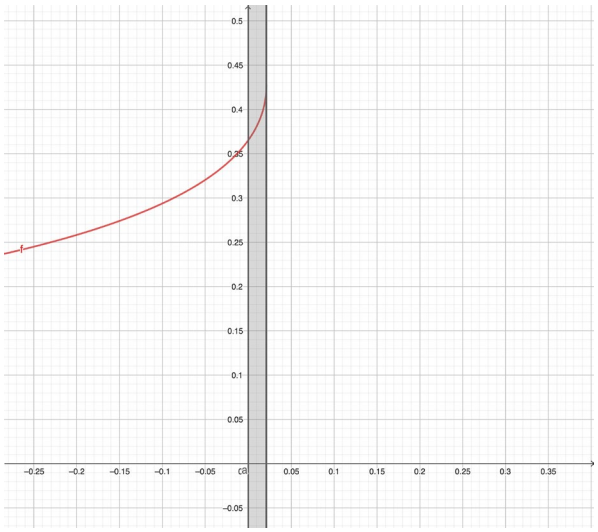


Figure 3b. Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for Real $\gamma \in \mathbb{R}$ (Negative Case)

Figures 3a and 3b illustrate the solution set for $\|\lambda_1\|$ and $\|\lambda_1\|$ when $\gamma \in \mathbb{R}$. The red line in Figure 3a represents the function: $\max\{\|\lambda_1\|, \|\lambda_2\|\} \approx \frac{1{,}7298-\varphi+\sqrt{\varphi^2-3{,}4596\varphi+0{,}073}}{4}$, Within the range $\varphi \in [0;\ 0{,}021233349266117]$, the value of $\max\{\|\lambda_1\|, \|\lambda_2\|\}$ remains below 0.5, indicating that the convergence requirements of PSO are satisfied, which requires $\max\{\|\lambda_1\|, \|\lambda_2\|b\} < 1$. Similarly, the red line in Figure 3b represents the function: $\max\{\|\lambda_1\|, \|\lambda_2\|\} \approx \frac{1{,}7298-\varphi-\sqrt{\varphi^2-3{,}4596\varphi+0{,}073}}{4}$, Within the range $\varphi \in [0;\ 0{,}021233349266117]$, the value of $\max\{\|\lambda_1\|, \|\lambda_2\|\}$ also remains below 0.5, further confirming that the proposed modification satisfies the convergence criteria for PSO. Now, Consider the case where γ is $\mathbb{C}$, when $\varphi \in (0{,}021233349266117; 3]$as shown in Equation 19.

$$\|\lambda_1\| = \|\lambda_2\| = \sqrt{\frac{(1{,}7298-\varphi)^2}{4} + \frac{-\varphi^2+3{,}4596\varphi-0{,}073}{4}} \approx 0{,}8542845 \quad (19)$$

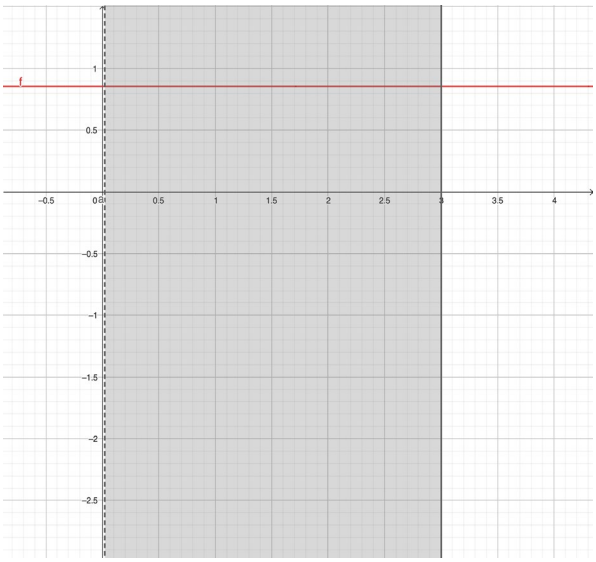


Figure 4. Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for $\gamma \in \mathbb{C}$

Figure 4 illustrates the solution set for $\|\lambda_1\|$ and $\|\lambda_2\|$ when $\gamma \in \mathbb{C}$. The red line shown in Figure 4 represents the function have been described in Equation (19).

$$\max\{\|\lambda_1\|, \|\lambda_2\|\} \approx \sqrt{\frac{(1{,}7298-\varphi)^2}{4} + \frac{-\varphi^2+3{,}4596\varphi-0{,}073}{4}} \quad (19)$$

Within the range $\varphi \in (0{,}021233349266117; 3]$, the value of $\max\{\|\lambda_1\|, \|\lambda_2\|\}$ is approximately 0.8542845, which satisfies the convergence criterion of PSO, as it remains below the condition $\max\{\|\lambda_1\|, \|\lambda_2\|\} < 1$. Thus, it can be observed that the proposed modification ensures the generation of a convergent trajectory.

3.3 Analysis and Discussion of Experimental Results

To assess the effectiveness of the proposed TBPSO, a series of experiments were performed using a collection of widely recognized benchmark functions in Table 1, covering six global optimization problems. These six functions have been described in the equations (3)-(7). Figure 5 presents the two-dimensional visualization of the five benchmark test functions used. Table 1 provides the mathematical formulas, number of dimensions, search space boundaries, global optimum values, and the characteristics of each function. This

section focuses on analyzing the impact of different acceleration coefficients in the PSO algorithm. The implementation of other methods, such as UACPSO, TVAC-PSO (eq. 8 and 9), SCAC-PSO (eq. 10 and 11), NDAC-PSO (eq.12 and 13), and SB-PSO (eq. 14 dan 15), follows a similar procedure to TBPSO, with the main difference lying in the use of different acceleration coefficients.

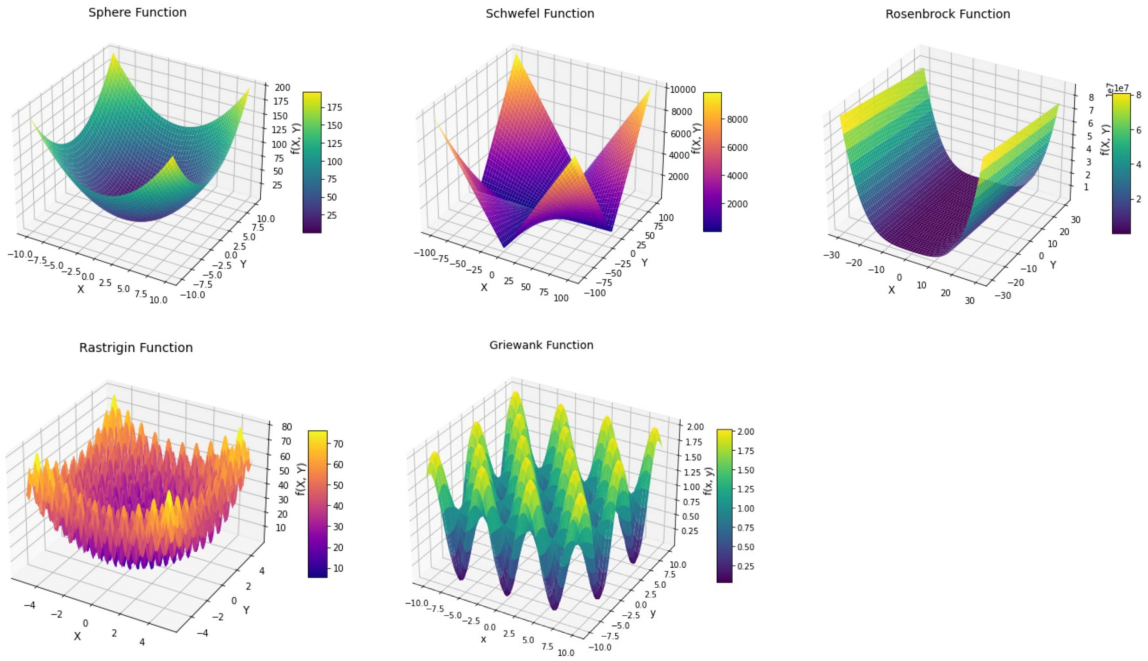


Figure 5. The two-dimensional visualization of the Five Benchmark Test Functions

Table 1. Properties of the Test Function

Test Functions	Dimensions	Search Range	Global Optimum	Properties
Sphere (f1)	10/30	[-10,10]^D	0	Unimodal
Schwefel (f2)	10/30	[-100,100]^D	0	Unimodal
Rosenbrock (f3)	10/30	[-30,30]^D	0	Unimodal
Griewank (f4)	10/30	[-600,600]^D	0	Multimodal
Ackley (f5)	10/30	[-32,32]^D	0	Multimodal

The implementation of other methods, such as UACPSO, TVAC-PSO, SCAC-PSO, NDAC-PSO, and SB-PSO, follows a similar procedure to TBPSO, with the main difference lying in the use of different acceleration coefficients.

The experimental configuration used in this study is established as follows: the swarm population size is set to N = 40 [20], and each benchmark function is independently evaluated through 30 runs, where each run performs 1000 iterations. All PSO algorithms are terminated after reaching the predefined maximum iteration limit to ensure a consistent comparison among different methods and to manage computational resources, as unlimited execution may lead to excessive runtime without guaranteeing significant improvement in solution quality, as commonly practiced in heuristic optimization studies [21]. The performance of TB-PSO is evaluated using several commonly adopted optimization metrics, including the best solution, average solution, and standard deviation. The best solution represents the lowest (or highest, depending on the problem) objective function value achieved across all runs. The average solution provides insight into the algorithm's general performance over multiple runs, which is important because metaheuristic algorithms like PSO are stochastic [22] in nature and may yield different results in each execution. By observing the average, we can assess whether the algorithm consistently finds good solutions or only performs well occasionally. The standard deviation, on the other hand, indicates the stability or robustness of the algorithm [23]. A low standard deviation means the results are tightly clustered around the average, suggesting the algorithm performs reliably across runs. A high standard deviation implies high variability, which may indicate sensitivity to initial conditions or instability in convergence. These metrics assess the effectiveness of TB-PSO in solving unimodal and multimodal benchmark functions compared to other PSO methods, particularly in terms of result stability.

To further evaluate the advantages of the proposed PSO modification in this study, TB-PSO is compared with five other PSO variants (UAPSO, TVAC-PSO, SCAC-PSO, NDAC-PSO, SB-PSO). It should be noted that the

dimensionality of all benchmark functions is configured at 10 and 30 dimensions. The performance evaluation is based on the average best solution (Avg.) and the standard deviation of the best solution (Std.), where the superior results are emphasized in bold.

From the results obtained for function dimensions of 10 on Table 2, it can be observed that although TB-PSO ranks fourth in terms of Avg. for f_1, it secures the top rank twice for f_3 and f_5. For f_2, it ranks second, with a minimal difference compared to TVAC-PSO, which holds the first position. Similarly, for f_4, TB-PSO ranks third, with a slight difference compared to NDAC-PSO in first place and SB-PSO in second. Overall, TB-PSO obtains the highest ranking among all evaluated methods, demonstrating its superior performance based on the average best solution and standard deviation metrics.

Table 2. Results of TB-PSO with different acceleration coefficients under D=10

Function	Item	UACPSO	TVAC-PSO	SCAC-PSO	NDAC-PSO	SAC-PSO	TB-PSO
f1	Avg.	3.538e-39	9.9284e-45	1.5785e-20	4.6945e-65	5.8679e-73	3.3224e-43
	Std.	8.428e-39	3.9978e-44	5.4567e-20	9.2316e-65	1.4079e-72	8.6482e-43
	Rank	5	3	6	2	1	4
f2	Avg.	103.3333	4.0695e-21	7.3839e-09	4.8592e-08	2.1391	4.8640e-20
	Std.	87.4960	1.5557e-20	1.6545e-08	2.6168e-07	0.0001	8.7043e-20
	Rank	6	1	3	4	5	2
f3	Avg.	15644.57	2.3721	4.5008	7.2407	6.9532	1.6594
	Std.	33273.91	3.0931	2.5277	18.6422	11.1224	1.4737
	Rank	6	2	3	5	4	1
f4	Avg.	0.1311	0.0565	0.0695	0.0416	0.0482	0.0542
	Std.	0.0675	0.0307	0.0355	0.0258	0.0313	0.0273
	Rank	6	4	5	1	2	3
f5	Avg.	0.0385	4.1152	5.4052	4.2337	5.6547	4.70e-15
	Std.	0.2074	6.3773	4.0718	8.8620	1.7724	1.42e-15
	Rank	2	3	5	4	6	1
Avg rank		5	2.6	4.4	3.2	3.6	2.2
Final rank		6	2	5	3	4	1

Table 3. Results of TB-PSO with different acceleration coefficients under D=30

Function	Item	UACPSO	TVAC-PSO	SCAC-PSO	NDAC-PSO	SAC-PSO	TB-PSO
f_1	Avg.	60.00	0.0007	3.0866e-05	1.3887	0.5386	0.0165
	Std.	75.7188	0.0019	4.5051e-05	0.8974	0.5100	0.0349
	Rank	6	2	1	5	4	3
f_2	Avg.	620.2326	10.0128	34.0329	121.5802	112.7581	0.0748
	Std.	285.9189	39.5781	59.1743	57.5216	80.2891	0.1778
	Rank	6	2	3	5	4	1
f_3	Avg.	8024586.7	55.1368	132.4436	8637.22	2438.94	132.834
	Std.	23993091.4	45.8024	120.7221	10601.94	11.1224	143.74
	Rank	6	1	2	5	4	3
f_4	Avg.	69.3386	0.1325	0.0209	2.1649	1.4999	0.3633
	Std.	72.5798	0.2176	0.0204	0.8729	0.6209	0.3399
	Rank	6	2	1	5	4	3
f_5	Avg.	14.5509	2.2814	0.0729	5.1091	3.3869	1.8031
	Std.	6.0579	0.6517	0.0592	1.0914	0.8370	0.6055
	Rank	6	3	1	5	4	2
Avg rank		6	2	1.6	5	4	2.4
Final rank		6	2	1	5	4	3

For the 30-dimensional functions on Table 3, TB-PSO ranks first once for f_2. It secures the second position for f_5, just behind SCAC-PSO. Meanwhile, for f_1, f_3, and f_4, TB-PSO ranks third. Although TB-PSO's ranking is not as high in the 30-dimensional case as in the 10-dimensional one, it still demonstrates competitive results compared to other PSO variants. This decline in performance does not imply that TB-PSO is ineffective; rather, it highlights the increased complexity of high-dimensional optimization, which requires broader exploration strategies. Such performance degradation is a well-documented challenge in PSO-based optimization methods, especially in high-dimensional search spaces [24], [25] where premature convergence or loss of diversity often occurs. In comparison, previous studies such as [8] have also reported that their proposed PSO variants showed performance drops when scaling from 10 to 30 dimensions, emphasizing that maintaining a balance between exploration and exploitation becomes more difficult as dimensionality increases.

The key factor influencing TB-PSO's performance is the tanh-based acceleration mechanism, which provides a controlled transition from exploration to exploitation. In lower dimensions, this strategy has proven highly effective in achieving rapid convergence without compromising solution quality. However, in higher dimensions, additional modifications may be required to enhance exploration and prevent convergence to suboptimal solutions. Minor refinements, such as more

dynamic parameter adaptation or additional search mechanisms, could further strengthen TB-PSO's capability in handling optimization across different dimensional scenarios. Overall, these findings demonstrate that the proposed TB-PSO modification provides substantial improvements in optimization performance, particularly in lower-dimensional search spaces. With further refinements, the method has the potential to become a more flexible and effective optimization solution, even for high-dimensional problems.

For future research, it is recommended to apply TB-PSO to real-world optimization problems, such as scheduling, energy management, or weather prediction with complex variables. Testing the algorithm in practical situations will help assess how well the tanh-based approach works in dynamic environments and may reveal ways to improve the method so it becomes more adaptive and effective for real-world applications.

4. Conclusions

This study introduces a Tanh-Based Particle Swarm Optimization (TB-PSO) approach that incorporates a hyperbolic tangent (tanh)-based mechanism for adjusting acceleration coefficients to enhance the balance between exploration and exploitation. The continuous and smooth characteristics of the tanh function enable a gradual adjustment of acceleration values throughout iterations, helping to preserve swarm diversity and avoid premature convergence. Moreover, since the tanh function maps inputs to a bounded range of −1 to 1, it prevents excessive velocity updates and contributes to stable convergence behavior. Convergence analysis confirms that TB-PSO satisfies stability criteria, making it suitable for optimization tasks. Experimental results demonstrate that TB-PSO performs particularly well in 10-dimensional benchmark problems, achieving the 1st rank out of 5 PSO variants based on solution quality and stability. In 30-dimensional problems, TB-PSO remains competitive, ranking 3rd out of 5, indicating its adaptability to increased problem complexity, although broader exploration mechanisms are required for higher-dimensional search spaces.In terms of computational complexity, TB-PSO does not introduce significant additional overhead compared to others PSO variants. Furthermore, applying TB-PSO to real-world optimization problems and benchmark datasets would provide deeper insights into its practical effectiveness and scalability.

References

[1] J. Zhang *et al.*, "Capacity estimation for series-connected battery pack based on partial charging voltage curve segments," *Journal of Energy Storage*, vol. 95, p. 112576, Aug. 2024, doi: https://doi.org/10.1016/j.est.2024.112576.

[2] Y.-L. Chen, J. Cheng, C. Lin, X. Wu, Y. Ou, and Y. Xu, "Classification-based learning by particle swarm optimization for wall-following robot navigation," *Neurocomputing*, vol. 113, pp. 27–35, Aug. 2013, doi: https://doi.org/10.1016/j.neucom.2012.12.037.

[3] K. Deligkaris, "Particle Swarm Optimization and Random Search for Convolutional Neural Architecture Search," *IEEE Access*, vol. 12, pp. 91229–91241, 2024, doi: https://doi.org/10.1109/access.2024.3420870.

[4] E. Siivola, A. Paleyes, J. González, and A. Vehtari, "Good practices for Bayesian optimization of high dimensional structured spaces," *Applied AI Letters*, vol. 2, no. 2, May 2021, doi: https://doi.org/10.1002/ail2.24.

[5] F. VANDENBERGH and A. ENGELBRECHT, "A study of particle swarm optimization particle trajectories," *Information Sciences*, vol. 176, no. 8, pp. 937–971, Apr. 2006, doi: https://doi.org/10.1016/j.ins.2005.02.003.

[6] L. Wu, D. Zhao, X. Zhao, and Y. Qin, "Nonlinear Adaptive Back-Stepping Optimization Control of the Hydraulic Active Suspension Actuator," *Processes*, vol. 11, no. 7, p. 2020, Jul. 2023, doi: https://doi.org/10.3390/pr11072020.

[7] X. Li and A. P. Engelbrecht, "Particle swarm optimization," Jul. 2007, doi: https://doi.org/10.1145/1274000.1274118.

[8] D. Tian, X. Zhao, and Z. Shi, "Chaotic particle swarm optimization with sigmoid-based acceleration coefficients for numerical function optimization," *Swarm and evolutionary computation*, vol. 51, pp. 100573–100573, Dec. 2019, doi: https://doi.org/10.1016/j.swevo.2019.100573.

[9] C.-Y. Tsai and I-Wei. Kao, "Particle swarm optimization with selective particle regeneration for data clustering," *Expert Systems with Applications*, vol. 38, no. 6, pp. 6565–6576, Jun. 2011, doi: https://doi.org/10.1016/j.eswa.2010.11.082.

[10] Ponnuthurai Nagaratnam Suganthan, "Particle swarm optimiser with neighbourhood operator," Jan. 2003, doi: https://doi.org/10.1109/cec.1999.785514.

[11] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-Organizing Hierarchical Particle Swarm Optimizer With Time-Varying Acceleration Coefficients," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 240–255, Jun. 2004, doi: https://doi.org/10.1109/tevc.2004.826071.

[12] K. Chen, F. Zhou, L. Yin, S. Wang, Y. Wang, and F. Wan, "A hybrid particle swarm optimizer with sine cosine acceleration coefficients," vol. 422, pp. 218–241, Jan. 2018, doi: https://doi.org/10.1016/j.ins.2017.09.015.

[13] G. Xu, "An adaptive parameter tuning of particle swarm optimization algorithm," *Applied Mathematics and Computation*, vol. 219, no. 9, pp. 4560–4569, Jan. 2013, doi: https://doi.org/10.1016/j.amc.2012.10.067.

[14] Y. Tang, Z. Wang, and J. Fang, "Feedback learning particle swarm optimization," *Applied Soft Computing*, vol. 11, no. 8, pp. 4713–4725, Dec. 2011, doi: https://doi.org/10.1016/j.asoc.2011.07.012.

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

[16] N. Lynn and P. N. Suganthan, "Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation," *Swarm and Evolutionary Computation*, vol. 24, pp. 11–24, Oct. 2015, doi: https://doi.org/10.1016/j.swevo.2015.05.002.

[17] E. Siivola, A. Paleyes, J. González, and A. Vehtari, "Good practices for Bayesian optimization of high dimensional structured spaces," *Applied AI Letters*, vol. 2, no. 2, May 2021, doi: https://doi.org/10.1002/ail2.24.

[18] M. Lin, Z. Wang, F. Wang, and D. Chen, "Improved Simplified Particle Swarm Optimization Based on Piecewise Nonlinear Acceleration Coefficients and Mean Differential Mutation Strategy," *IEEE Access*, pp. 1–1, 2020, doi: https://doi.org/10.1109/access.2020.2994984.

[19] H. Noriega, R. Lazzari, and L. Piegari, "Particle Swarm Optimization for the Sizing of a Battery and Hydrogen Storage System for Residential Buildings," *2024 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM)*, pp. 1142–1147, Jun. 2024, doi: https://doi.org/10.1109/speedam61530.2024.10609206.

[20] C. Gong, N. Zhou, S. Xia, and S. Huang, "Quantum particle swarm optimization algorithm based on diversity migration strategy," *Future Generation Computer Systems*, vol. 157, pp.

445–458, Apr. 2024, doi: https://doi.org/10.1016/j.future.2024.04.008.
[21] Absalom El-Shamir Ezugwu, "Metaheuristic Optimization for Sustainable Unrelated Parallel Machine Scheduling: A concise overview with a proof-of-concept study," *IEEE access*, pp. 1–1, Jan. 2023, doi: https://doi.org/10.1109/access.2023.3347047.
[22] M. Jain, V. Saihjpal, N. Singh, and S. B. Singh, "An Overview of Variants and Advancements of PSO Algorithm," *Applied Sciences*, vol. 12, no. 17, p. 8392, Aug. 2022, doi: https://doi.org/10.3390/app12178392.
[23] B. Gao, W. Shen, H. Guan, L. Zheng, and W. Zhang, "Research on Multistrategy Improved Evolutionary Sparrow Search Algorithm and its Application," *IEEE Access*, vol. 10, pp. 62520–62534, 2022, doi: https://doi.org/10.1109/access.2022.3182241.
[24] M. Al-Shalabi, M. Shehab, M. T. Alshammari, M. Alazmi, R. O. Alrawashdeh, and M. A. Mahdi, "Enhanced lotus effect optimization algorithm for efficient problem-solving in high-dimensional complex landscapes," *Case Studies in Thermal Engineering*, vol. 73, p. 106437, Sep. 2025, doi: https://doi.org/10.1016/j.csite.2025.106437.
[25] S. Helwig and R. Wanka, "Particle Swarm Optimization in High-Dimensional Bounded Search Spaces," Apr. 2007, doi: https://doi.org/10.1109/sis.2007.368046.