Convergence Analysis and Numerical Functions Implementation in Adaptive Particle Swarm Optimization using Tanh-Based Acceleration Coefficients

by Fayza Nayla Riyana Putri

Submission date: 16-Mar-2025 08:22PM (UTC+0700)
Submission ID: 2615922470
File name: jurnal_IJAI.docx (2.59M)
Word count: 4861
Character count: 29736

IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 99, No. 1, Month 2099, pp. 1~1x
ISSN: 2252-8938, DOI: 10.11591/ijai.v99.i1.pp1-1x

Convergence Analysis and Numerical Functions Implementation in Adaptive Particle Swarm Optimization using Tanh-Based Acceleration Coefficients

Aina Latifa Riyana Putri[1], Joko Riyono[2], Christina Eni Pujiastuti[3], Supriyadi[4]
[1]Data Science, Telkom University, Purwokerto, Indonesia
[2,3,4] Department of Mechanical Engineering, Faculty of Industrial Technology, Universitas Trisakti, Jakarta, Indonesia

Article Info

Article history:

Received month dd, yyyy
Revised month dd, yyyy
Accepted month dd, yyyy

Keywords:

Particle Swarm Optimization
Tanh Function
Acceleration Coefficient
Convergence
Single Objective Function

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, we propose Tanh-Based Acceleration Coefficient PSO (TB-PSO), which utilizes the hyperbolic tangent function to ensure smoother and more stable acceleration adjustments, maintaining a balance between 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. Experimental results demonstrate that TB-PSO outperforms other methods in 10-dimensional problems, achieving the best final ranking. Although its performance slightly declines in 30-dimensional cases, it remains competitive with strong convergence stability. These findings highlight TB-PSO's effectiveness in improving PSO performance, particularly in lower-dimensional search spaces, while showing potential for further enhancement in higher-dimensional optimization problems.

Corresponding Author:

Kennedy Okokpujie
Department of Electrical and Information Engineering, College of Engineering, Covenant University
Km. 10 Idiroko Road, Canaan Land, Ota, Ogun State, Nigeria
Email: kennedy.okokpujie@covenantuniversity.edu.nga

1. INTRODUCTION

Particle Swarm Optimization (PSO) is a population-based optimization method inspired by the social behavior of bird flocks and fish schools. PSO operates by utilizing a population of particles that navigate the search space to find an optimal solution. Each particle represents a candidate solution and is characterized by its position and velocity, which are updated iteratively based on its personal best position (pbest) and the global best position (gbest) of the entire population.

Particle Swarm Optimization (PSO) has been widely applied to various optimization problems due to its ability to intelligently explore the search space using a population-based approach. This characteristic allows PSO to achieve more efficient computations and perform well in handling high-dimensional problems compared to other algorithms such as Grid Search [1], Random Search [2], and Bayesian Optimization [3]. 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. [4] 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 [5]-[8]. Consequently, numerous studies have been conducted to enhance the performance of PSO, particularly by modifying the acceleration coefficients (C_1 and C_2), which play a crucial role in balancing the exploration and exploitation of the search space [7]. Proper adjustment of these coefficients can improve convergence speed and solution quality.

[9] Proposed a modification to the acceleration coefficients with unbalanced values (C_1 = 0.5 and C_2 = 2.5), demonstrating improved convergence for specific optimization problems. [10] found that acceleration coefficients should change dynamically in each iteration to achieve a global optimal solution. To address this, several approaches have been introduced using both linear and nonlinear functions to adaptively adjust acceleration coefficients throughout the iterations, including Time-Varying Acceleration Coefficients [11], Sine-Cosine Acceleration Coefficients [6], Nonlinear Dynamics Acceleration Coefficients [6], and Sigmoid-Based Acceleration Coefficients [7]. These modifications allow for adaptive changes in acceleration coefficients during the optimization process, improving PSO's convergence and solution quality.

However, research on the use of hyperbolic functions, such as tanh, in PSO modifications remains relatively unexplored. In the tanh-based PSO modification, the acceleration coefficients are adjusted using the characteristics of the hyperbolic tangent (tanh) function. The smooth and continuous nature of the tanh function enables a more gradual transition between acceleration values in each iteration, which is expected to maintain swarm diversity and prevent premature convergence to local optima [12];[13]. 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 proposes a modification of PSO hyperparameters, where the acceleration coefficients (C_1 and C_2) are based on the hyperbolic tangent (tanh) function. The proposed hyperparameter modification will be analyzed to verify whether it ensures the convergence of the modified PSO algorithm by applying the convergence criteria theorem from [4]. To demonstrate the effectiveness of the proposed PSO modification, comparisons will be made with several existing approaches, including Time-Varying Acceleration Coefficients, Sine-Cosine Acceleration Coefficients, Nonlinear Dynamics Acceleration Coefficients, and Sigmoid-Based Acceleration Coefficients. Performance evaluation will be conducted using benchmark unimodal and multimodal functions to assess the method's effectiveness under different optimization scenarios. Unimodal functions are used to measure how quickly the method reaches the optimal solution in a relatively simple landscape, while multimodal functions evaluate its ability to escape local optima in more complex search spaces.

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. METHOD

The methodology section will sequentially present the analytical methods employed in this study on Figure 1.

1. Analysis of Convergence Criteria for Particle Swarm Optimization Algorithm Parameters
   Particle Swarm Optimization (PSO) was first proposed by [14] as an algorithm for optimizing continuous nonlinear functions. In PSO, each particle independently searches for its best position (personal best or Pbest) while also considering the best position found by the entire swarm (global best or Gbest). This allows the algorithm to converge toward an optimal solution. Several key terms commonly used in PSO include:
   a. Pbest (Personal Best)
      $p_{(i,lb)}^{(t)}$ represents the personal best position of particle i at generation t. Assuming a minimization problem, Pbest in PSO is updated using the following equation:

$$p_{(i,lb)}^{(t+1)} = \begin{cases} x_i^{(t+1)}, \text{if } f\left(x_i^{(t+1)}\right) < f\left(p_{(i,lb)}^{(t)}\right) \\ p_{(i,lb)}^{(t)}, \quad \text{other} \end{cases} \quad (1)$$

   b. Gbest (Global Best)
      $p_{gb}^{(t)}$ represents the global best position at generation ttt, which is determined as follows:

$$p_{gb}^{(t)} \in \left\{p_{(i,best)}^{(t+1)}, \dots, p_{(N,best)}^{(t+1)}\right\} | f\left(p_{gb}^{(t)}\right) = \min p_{gb}^{(t)} \in \left\{p_{(i,best)}^{(t+1)}, \dots, p_{(N,best)}^{(t+1)}\right\} \quad (2)$$

where N is the number of particles in the population.

Each particle shares information about its best position with other particles and adjusts its position and velocity accordingly based on the received information. This means that both the position and velocity of each particle are continuously updated in each iteration during 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 using the following equations:

a. Particle Position Update

$$x_i^{(t+1)} = v_i^{(t+1)} + x_i^{(t)} \quad (3)$$

Where $x_i^{(t+1)}$ is the position of particle i at generation t+1; $v_i^{(t+1)}$ is the velocity of particle i at generation t+1; $x_i^{(t)}$ is the position of particle i at generation t.

b. Particle Velocity Update

$$v_i^{(t+1)} = \omega\, v_i^{(t)} + c_1 r_1 \left(p_{(i,lb)}^{(t)} - x_i^{(t)}\right) + c_2 r_2 \left(p_{gb}^{(t)} - x_i^{(t)}\right) \quad (4)$$

Where $v_i^{(t+1)}$ is the velocity of particle iii at generation t+1; $v_i^{(t)}$ is the velocity of particle I at generation t; c_1 and c_2 are acceleration coefficients, which are set to 2 in the original PSO algorithm; ω is the inertia weight; $p_{(i,lb)}^{(t)}$ is the personal best position; $x_i^{(t)}$ is the previous position of particle i; r_1 and r_2 are random values in the range [0,1]; $p_{gb}^{(t)}$ is the global best position.

The original Particle Swarm Optimization (PSO) algorithm has several drawbacks, as highlighted by studies such as [5]-[8]. These studies indicate that PSO particles tend to converge prematurely to a stable state, leading to the solutions being trapped in local optima. Additionally, [15] explain that the convergence behavior of population-based algorithms like PSO can be achieved when there is a proper balance between exploration and exploitation of the search space, thereby guiding particles toward the global optimal solution.

The convergence of PSO toward the global optimal solution is discussed in the following theorem.

Theorem 1. [4] A particle in the Particle Swarm Optimization algorithm converges to a stable point given by $\frac{c_1 p_{(i,lb)}^{(t)} + c_2 p_{gb}^{(t)}}{c_1 + c_2}$, if $max\{||\lambda_1||, ||\lambda_2||\} < 1$ where λ_1 and λ_2 are the eigenvalues representing the dynamics of a simple Particle Swarm Optimization system with inertia (ω).

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 observed that in the original Particle Swarm Optimization (PSO), the chosen values of $c_1 = c_2 = 2$, and $w = 1$ do not satisfy the convergence criteria for PSO parameters. This is because $c_1 + c_2 = 2 + 2 = 4 > 0$ and $\frac{c_1+c_2}{2} - 1 = \frac{2+2}{2} - 1 = 1 = w$. This seemingly implies that the original PSO equation produces a divergent trajectory [4]. Consequently, this raises concerns regarding the application of the original PSO in real-world problems. The trajectory divergence, suggests that the original PSO does not provide adequate convergence results. To address this issue, this study focuses on modifying the acceleration coefficients to achieve a better balance between exploration and exploitation. At this stage, we will examine whether the proposed tanh-based acceleration coefficient modification for the Particle Swarm Optimization (PSO) algorithm, 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. Experimental Testing and Simulation of the Proposed Modified Particle Swarm Optimization (PSO) Algorithm

This study also conducts an empirical analysis through a series of experimental tests to evaluate the effectiveness of the proposed acceleration coefficient modification. The experiments are carried out by applying the modified PSO algorithm to various standard benchmark functions in optimization. Testing is performed using two main categories of objective functions: unimodal functions and multimodal functions. The selection of these functions aims to assess the convergence capability and effectiveness of the proposed method in finding optimal solutions across different optimization scenarios.

- Unimodal Functions

Unimodal functions are objective functions with a single global optimum, making them particularly suitable for evaluating the convergence speed of an algorithm. In this context, a more efficient algorithm is expected to reach the optimal solution more quickly. The unimodal functions used in this study include:

1. Sphere Function

$$\min f_1(x) = \sum_{i=1}^{n} x_i^2 \quad (5)$$

where the global optimum is $x^* = 0$ and $f(x^*) = 0$ for $-10 \le x_i \le 10$.

2. Schwefel Function

$$\min f_2(x) = \sum_{i=1}^{n} |x_i| + \prod_{i=1}^{n} |x_i| \quad (6)$$

where the global optimum is $x^* = 0$ and $f(x^*) = 0$ for $-100 \le x_i \le 100$.

3. Rosenbrock Function

$$\min f_3(x) = \sum_{i=1}^{n}\left[100\,(x_i^2 - x_{i+1})^2 + (x_i - 1)^2\right] \quad (7)$$

where the global optimum is $x^* = (1,1,\ldots,1)$ and $f(x^*) = 0$ for $-30 \le x_i \le 30$.

- Multimodal Functions

Multimodal functions have multiple local optima, making them more challenging for optimization algorithms. The main objective of this test is to evaluate PSO's ability to escape local optima and locate the global optimum. The multimodal functions used in this study include:

4. Griewank Function

$$\min f_4(x) = \frac{1}{4000}\sum_{i=1}^{n}(x_i)^2 - \prod_{i=1}^{d}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1 \quad (8)$$

where the global optimum is $x^* = 0$ and $f(x^*) = 0$ for $-600 \le x_i \le 600$.

5. Ackley Function

$$\min f_6(x) = -20exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - exp\left(\frac{1}{n}\sum_{i=1}^{n} cos(2\pi x_i)\right) + 20 + e \quad (9)$$

where the global optimum is $x^* = 0$ and $f(x^*) = 0$ for $-32 \le x_i \le 32$.

3. Performance Comparison with Other PSO Acceleration Coefficient Modifications

At this stage, a performance comparison analysis is conducted between the proposed modification of the Particle Swarm Optimization (PSO) algorithm, namely the Tanh-based Acceleration Coefficient, and several other PSO modification methods, such as:

1. Unbalanced Acceleration Coefficient PSO (UACPSO)

   The study conducted by [8] developed a modification of the cognitive and social parameters (C_1 and C_2) to accelerate the convergence of the Particle Swarm Optimization algorithm. In their experiments, the Unbalanced Acceleration Coefficient was set as $C_1 = 0.5$ and $C_1 = 2.0$.

2. Time Varying Acceleration Coefficients PSO (TVAC-PSO)

   This study also proposes a modification of the cognitive and social parameter values (C_1 and C_2) based on the research by [9], which employs a linear decrement method for both acceleration coefficients over time to produce better solutions. In this modification, the cognitive component (C_1) is gradually decreased, while the social component (C_2) is increased over time, as described by the following equations:

$$C_1 = (C_{1f} - C_{1i})\frac{iter}{MAXITR} + C_{1i} \quad (10)$$

$$C_2 = (C_{2f} - C_{2i})\frac{iter}{MAXITR} + C_{2i} \quad (11)$$

where C_{1f}, C_{1i}, C_{2f}, C_{2f} are constants, $iter$ represents the current iteration number, and $MAXITR$ is the predefined maximum number of iterations.

3. Sine Cosine Acceleration Coefficients PSO (SCAC-PSO)
   Inspired by TVAC-PSO and the study by [9], this research also proposes the use of time-varying acceleration coefficients, where individuals in the population are expected to explore the entire search space during the early stages of the optimization process, while in the final stages, the convergence ability towards the global optimum is enhanced. This study introduces the Sine-Cosine Acceleration Coefficient (SCAC) as a new parameter adjustment strategy for the cognitive and social components:

$$C_1 = \partial \; x \sin\left(\left(1 - \frac{iter}{MAXITR}\right) x \; \frac{\pi}{2}\right) + \delta \quad (12)$$

$$C_2 = \partial \; x \cos\left(\left(1 - \frac{iter}{MAXITR}\right) x \; \frac{\pi}{2}\right) + \delta \quad (13)$$

   where ∂ and δ are constants $\partial = 2; \delta = 0.5$).

4. Nonlinear Dynamics Acceleration Coefficients PSO (NDAC-PSO)
   In this study, the acceleration coefficient of PSO is modified using the Nonlinear Dynamic Acceleration Coefficient (NDAC) method as a parameter update mechanism to adjust the cognitive component (C_1) and the social component (C_2). The equations representing NDAC are as follows:

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

$$C_2 = C_{1i} \; x \; \left(1 - \frac{iter}{MAXITR}\right)^2 + C_{1f} \; x \; \frac{iter}{MAXITR} \quad (15)$$

   where C_{1f} and C_{1i} are positive constants ($C_{1f} = 2.5$ and $C_{1i} = 0.5$), $iter$ represents the current iteration, and $MAXITR$ is the maximum number of iterations.

5. Sigmoid-based Acceleration Coefficient PSO (SAC-PSO)
   This study introduces the Sigmoid-Based Acceleration Coefficient (SBAC) with the following equations:

$$C_1 = \frac{1}{1+e^{(-\lambda \cdot \frac{iter}{MAXITR})}} + 2\left(C_{1f} - C_{1i}\right)\left(\frac{iter}{MAXITR} - 1\right)^2 \quad (16)$$

$$C_2 = \frac{1}{1+e^{(-\lambda \cdot \frac{iter}{MAXITR})}} + \left(C_{1f} - C_{1i}\right)\left(\frac{iter}{MAXITR}\right)^2 \quad (17)$$

   where λ is a control parameter used to regulate the sigmoid-based acceleration coefficient ($\lambda = 0.0001$), C_{1f} and C_{1i} are two positive constants ($C_{1f} = 2.5$ and $C_{1i} = 0.5$). The terms iter and MAXITR denote the current iteration and the maximum number of iterations, respectively.
   The experiments were conducted by applying each method to the previously discussed unimodal and multimodal benchmark functions. The results obtained from each PSO modification were then compared to evaluate the stability of the final solution. By performing this analysis, this study aims to identify the effectiveness of the proposed acceleration coefficient modification in enhancing PSO performance across various optimization problems.

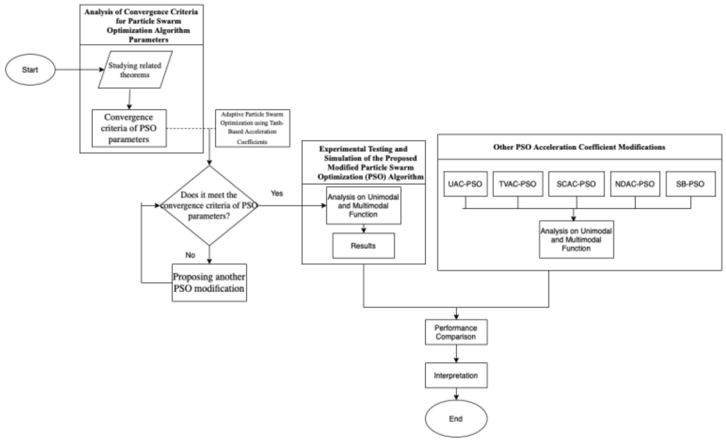

Figure 1 . Shows the flowchart of the research

3. RESULTS AND DISCUSSION

3.1. Tanh-Based Acceleration Coefficient

The performance of PSO heavily depends on maintaining a proper balance between exploration and exploitation. Therefore, the right ratio between these two aspects must be carefully established. In general, PSO should begin with strong exploration to widely search the solution space and gradually shift toward exploitation to refine the optimal solution with greater precision.

Based on this principle, various time-varying strategies have been developed to adjust PSO parameters, including the acceleration coefficients C_1 and C_2 (Tian). As previously discussed, C_1 influences local exploration tendencies, while C_2 determines the level of global exploitation. Consequently, dynamically adjusting these coefficients is crucial for improving the balance between exploration and exploitation in PSO.

In this study, the proposed modification of the acceleration coefficients in the Particle Swarm Optimization (PSO) algorithm based on the tanh function is defined as follows:

$$c_1(iter) = 0{,}5.\left(\tanh\left(2.(c_{1f} - c_{1i}).\left(\frac{iter\ max - iter}{iter\ max} - 0{,}5\right)\right) + 1\right).(c_{1f} - c_{1i}) + c_{1i} \quad (18)$$

$$c_2(iter) = 0{,}5.\left(\tanh\left(2.(c_{2f} - c_{2i}).\left(\frac{iter}{iter\ max} - 0{,}5\right)\right) + 1\right).(c_{2f} - c_{2i}) + c_{2i} \quad (19)$$

where 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" represents the current iteration, while 'iter max' denotes the maximum number of allowed iterations.

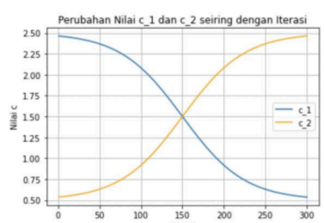

Figure 2. Visualization of c_1 and c_2 Values per Iteration

Figure 2 visualizes the plot of equations (18) and (19). 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.

The proposed strategy enables a smoother transition from exploration to exploitation. During the early stages of the search, c_1 starts at a high value, allowing particles to explore the search space more broadly. Conversely, c_2 begins with a lower value, preventing premature exploitation. As the iterations progress, c_1 gradually decreases while c_2 increases. 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 previously explained, the conditions ensuring that each particle in the Particle Swarm Optimization algorithm converges to a stable point were analyzed. Based on the proposed modification, the values obtained are $c_1 = 0{,}5; c_2 = 2{,}5; \mathrm{w} = 0.7298$, 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 indicate that the selected values satisfy the convergence criteria of the Particle Swarm Optimization algorithm, as they meet the 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 $\varphi \in [0;\ 0{,}021233349266117]$:

$$max\{\|\lambda_1\|, \|\lambda_2\|\} \approx \frac{1{,}7298-\varphi \pm \sqrt{\varphi^2 - 3{,}4596\varphi + 0{,}073}}{4} \quad (20)$$

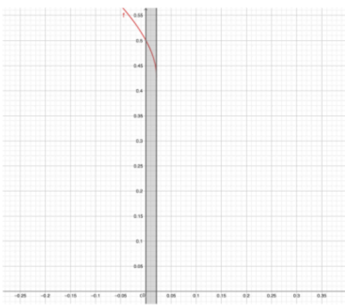

Figure 3.A Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for $\gamma \in \mathbb{R}$ (Positive Case)

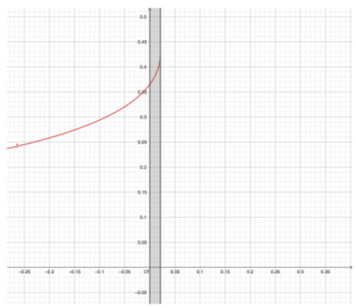

Figure 3.B Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for Real $\gamma \in \mathbb{R}$ (Negative Case)

Figures 3.A and 3.B illustrate the solution set for $\|\lambda_1\|$ and $\|\lambda_1\|$ when $\gamma \in \mathbb{R}$. The red line in Figure 3.A 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, satisfying the convergence criteria of the Particle Swarm Optimization algorithm, which requires $\max\{\|\lambda_1\|, \|\lambda_2\|\} < 1$. Similarly, the red line in Figure 3.B 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 the Particle Swarm Optimization algorithm.

Now, Consider the case where γ is $\mathbb{C}$, when $\varphi \in (0{,}021233349266117; 3]$:

$$\|\lambda_1\| = \|\lambda_2\| = \sqrt{\frac{(1{,}7298-\varphi)^2}{4} + \frac{-\varphi^2+3{,}4596\varphi-0{,}073}{4}} \approx 0{,}8542845 \quad (21)$$

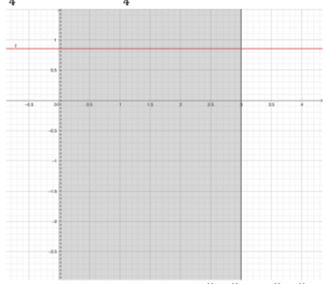

Figure 4. Graphical Visualization of $\|\lambda_1\|$ and $\|\lambda_1\|$ for $\gamma \in \mathbb{C}$

Figure 4 illustrates the solution set for $\|\lambda_1\|$ and $\|\lambda_2\|$ when $\gamma \in \mathbb{C}$. The red line in Figure 4 represents the function: $\max\{\|\lambda_1\|, \|\lambda_2\|\} \approx \sqrt{\frac{(1{,}7298-\varphi)^2}{4} + \frac{-\varphi^2+3{,}4596\varphi-0{,}073}{4}}$, 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 the Particle Swarm Optimization algorithm, 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.

The complete procedure of the proposed Modified Particle Swarm Optimization (PSO) algorithm with Tanh-based Acceleration Coefficients can be summarized as follows, with its flowchart presented in Figure 5.

Algorithm 2 : Pseudocode of the proposed TB-PSO Algorithm

Input :

w_0: inertia weight (0.7298) ; $c_{1i}, c_{2i}, c_{1f}, c_{2f}$; N: swarm size; D: swarm dimension; iter_max: maximum iterations

Process:

```
Initialize the swarm particles with random positions and velocities.
Evaluate the fitness of the each particle.
Identify the personal best (pbest) dan global best (gbest) solutions.
While iter ≤ iter_max do
      Calculate c1,c2 by Eqs. (20-21).
      for n = 1 to N do
          Update velocity and position of particles by Eqs.(24)-(25)
          Ensure boundaries are respected for xi
          Evaluate the fitness of the new particle position
          If fitness of xi is better than pbesti then
             Update pbesti with xi
          end if
          If fitness of xi is better than gbest then
             Update gbest with xi
          end if
      end for
      Update iteration counter
end while
```

Output:

Gbest particle as the final optimal solution.

Figure 5. Pseudocode of the Proposed TB-PSO Algorithm

3.3. Experimental Results and Analysis

To evaluate the performance of the proposed TBPSO, a series of experiments were conducted on a set of well-known benchmark functions, covering six global optimization problems. These six functions have been described in the equations (5)-(9). Figure below presents the two-dimensional visualization of the five benchmark test functions used.

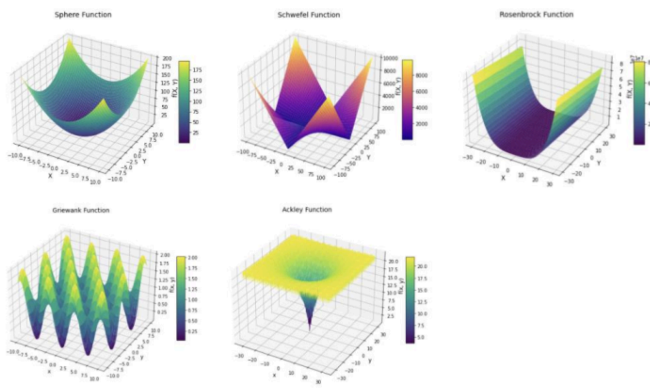

Figure 6. The Two-Dimensional Visualization of the Five Benchmark Test Functions

All these functions are designed to be minimized and are labeled as f_1 to f_6, as summarized in Table 1. The table 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.

Table 1. Properties of the Test Function

Test Functions	Dimensions	Search Range	Global Optimum	Properties
Sphere (f_1)	10/30	$[-10,10]^D$	0	Unimodal
Schwefel (f_2)	10/30	$[-100,100]^D$	0	Unimodal
Rosenbrock (f_3)	10/30	$[-30,30]^D$	0	Unimodal
Griewank (f_4)	10/30	$[-600,600]^D$	0	Multimodal
Ackley (f_5)	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 setup in this study is defined as follows: the swarm size is set to N = 40, and each benchmark function is executed independently 30 times, with each execution consisting of 1000 iterations. All PSO algorithms are terminated upon reaching the predefined maximum number of iterations. The performance evaluation of TB-PSO is conducted using commonly used optimization metrics, namely the best solution, average solution, and standard deviation. 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 in the table, including UAPSO, TVAC-PSO, SCAC-PSO, NDAC-PSO, SB-PSO, and TB-PSO. It is important to note that the dimensionality of all benchmark functions is set to 10 and 30. The average best solution (Avg.) and the standard deviation of the best solution (Std.) are used to measure performance, with the best results in the comparison highlighted in bold.

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
f_1	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
f_2	Avg.	103.3333	**4.0695e-21**	7.3839e-09	4.8592e-08	2.1391	4.8640e-20
	Std.	87.4960	**1.5557e-20**	1.6454e-08	2.6168e-07	0.0001	8.7043e-20
	Rank	6	1	3	4	5	2
f_3	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
f_4	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
f_5	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

From the results obtained for function dimensions of 10, it can be observed that although TB-PSO ranks fourth in terms of average best solution (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 achieves the best final ranking among all methods, indicating superior performance in terms of average best solution and standard deviation.

For the 30-dimensional functions, 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 common challenge in PSO-based optimization methods, where higher dimensions make it more difficult for algorithms to consistently find the global optimum.

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 results indicate that the proposed TB-PSO modification offers significant advantages, 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.

4. CONCLUSION

This study proposes a modification of PSO using a tanh-based acceleration coefficient to enhance the balance between exploration and exploitation. Convergence analysis confirms that the Tanh-Based PSO (TB-PSO) meets stability criteria and is suitable for testing on benchmark functions. Experimental results show that TB-PSO excels in 10-dimensional problems, ranking 1st out of 5 compared to other PSO variants. In 30-dimensional problems, its performance remains competitive, ranking 3rd out of 5, demonstrating adaptability to increased complexity. The key advantage of TB-PSO lies in its ability to prevent premature convergence and improve solution stability. For higher-dimensional scenarios, a broader exploration strategy is needed to further enhance its performance. Overall, the tanh-based modification proves to be an effective approach to improving PSO, making it a promising solution for various optimization problems.

REFERENCES

[1] Chen, Y. L., Cheng, J., Lin, C., Wu, X., Ou, Y., & Xu, Y. (2013). Classification-based learning by particle swarm optimization for wall-following robot navigation. Neurocomputing, 113, 27-35.

[2] Mananoma, T., & Soetopo, W. (2004). TEKNIK SIMULASI UNTUK OPTIMASI RANDOM SEARCH DENGAN MENERAPKAN RELAKSASI PADA PERBAIKAN FUNGSI TUJUAN. TEKNIK SIMULASI UNTUK OPTIMASI RANDOM SEARCH DENGAN MENERAPKAN RELAKSASI PADA PERBAIKAN FUNGSI TUJUAN, 11(3), 1-79.

[3] Siivola, E., Paleyes, A., González, J., & Vehtari, A. (2021). Good practices for Bayesian optimization of high dimensional structured spaces. Applied AI Letters, 2(2), e24.

[4] Van den Bergh, F., & Engelbrecht, A. P. (2006). A study of particle swarm optimization particle trajectories. Information sciences, 176(8), 937-971.

[5] Li, X., & Engelbrecht, A. P. (2007, July). Particle swarm optimization: an introduction and its recent developments. InProceedings of the 9th annual conference companion on Genetic and evolutionary computation (pp. 3391-3414).

[6] Chen, K., Zhou, F., Yin, L., Wang, S., Wang, Y., & Wan, F. (2018). A hybrid particle swarm optimizer with sine cosine acceleration coefficients.Information Sciences, 422, 218-241.

[7] Tian, D., Zhao, X., & Shi, Z. (2019). Chaotic particle swarm optimization with sigmoid- based acceleration coefficients for numerical function optimization. Swarm and Evolutionary Computation, 51, 100573.

[8] Wu, L., Zhao, D., Zhao, X., & Qin, Y. (2023). Nonlinear Adaptive Back-Stepping Optimization Control of the Hydraulic Active Suspension Actuator. Processes, 11(7), 2020.

[9] Tsai, C. Y., & Kao, I. W. (2011). Particle swarm optimization with selective particle regeneration for data clustering. Expert Systems with Applications, 38(6), 6565- 6576.

[10] Suganthan, P. N. (1999, July). Particle swarm optimiser with neighbourhood operator. Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406) (Vol. 3, pp. 1958-1962). IEEE.

[11] Ratnaweera, A., Halgamuge, S. K., & Watson, H. C. (2004). Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients.IEEE Transactions on evolutionary computation, 8(3), 240-255.

[12] Xu, G. (2013). An adaptive parameter tuning of particle swarm optimization algorithm. *Applied Mathematics and Computation*, *219*(9), 4560-4569.

[13] Tang, Y., Wang, Z., & Fang, J. A. (2011). Feedback learning particle swarm optimization. Applied Soft Computing, 11(8), 4713-4725.

[14] Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. In *Proceedings of ICNN'95-international conference on neural networks* (Vol. 4, pp. 1942-1948). ieee.

[15] N. Lynn, P. Suganthan, Heterogeneous comprehensive learning particle swarm optimization with enhanced exploration and exploitation, Swarm and Evolutionary Computation 24 (2015) 11–24.

BIOGRAPHIES OF AUTHORS

Aina Latifa Riyana Putri received the M.Mat. degree in mathematics from Diponegoro University, Semarang, Indonesia in 2023. She also received his S.Si (Mathematics) from Universitas Negeri Semarang, Semarang Indonesia, in 2021. She is currently a lecture at Data Science in Telkom University since 2024. Her research includes meta-heuristics, global optimization, machine learning, and data mining. She can be contacted at email: ainaqp@telkomuniversity.ac.id.

Joko Riyono earned his M.Si. in mathematics from Gajah Mada University, Yogyakarta, Indonesia in 1998. He also earned his Drs. (Mathematics) from Diponegoro University, Semarang, Indonesia, in 1991. He is currently a lecturer in mathematics and statistics in the Mechanical Engineering Department at Trisakti University since 1992. His research interests include meta-heuristics, global optimizer, machine learning, and data mining. He can be contacted via email: jokoriyono@trisakti.ac.id

Christina Eni Pujiastuti earned her M.Si. in mathematics from Universitas Gajah Mada, Yogyakarta, Indonesia in 1996. She also earned her Dra (Mathematics) from Universitas Gajah Mada, Yogyakarta, Indonesia, in 1988. She is currently a lecturer in mathematics and statistics in the Department of Mechanical Engineering at Trisakti University since 1989. Her research interests include meta-heuristics, global optimizer, machine learning, and data mining. She can be contacted via email: christina.eni@trisakti.ac.id

Supriyadi earned his Dr. degree in Mechanical Engineering from the University of Indonesia, Jakarta, Indonesia in 2017. He also earned his M.Sc. (Mathematics) from the Bandung Institute of Technology, Bandung, Indonesia, in 1997. His Drs. (Mathematics) degree from Gajah Mada University, Yogyakarta, Indonesia, in 1990. He is currently a lecturer in statistics and thermodynamics in the Department of Mechanical Engineering at Trisakti University since 1992. His research interests include Hydrogen Storage, Renewable Energy, machine learning, and data mining. He can be contacted via email: supri@trisakti.ac.id

Convergence Analysis and Numerical Functions Implementation in Adaptive Particle Swarm Optimization using Tanh-Based Acceleration Coefficients

ORIGINALITY REPORT

18%	18%	14%	4%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	kursorjournal.org Internet Source	7%
2	coek.info Internet Source	2%
3	Dongping Tian, Xiaofei Zhao, Zhongzhi Shi. "Chaotic particle swarm optimization with sigmoid-based acceleration coefficients for numerical function optimization", Swarm and Evolutionary Computation, 2019 Publication	2%
4	Submitted to Arab Academy for Science, Technology & Maritime Transport CAIRO Student Paper	1%
5	www.techscience.com Internet Source	1%
6	irep.iium.edu.my Internet Source	1%
7	ijai.iaescore.com Internet Source	1%
8	www.cs.uoi.gr Internet Source	1%
9	Meijin Lin, Zhenyu Wang, Fei Wang, Danfeng Chen. "Improved Simplified Particle Swarm Optimization Based on Piecewise Nonlinear Acceleration Coefficients and Mean	1%

Differential Mutation Strategy", IEEE Access, 2020
Publication

10	Ke Chen, Fengyu Zhou, Yugang Wang, Lei Yin. "An ameliorated particle swarm optimizer for solving numerical optimization problems", Applied Soft Computing, 2018 Publication	1 %
11	Junwei Zhang, Weige Zhang, Yanru Zhang, Caiping Zhang, Bo Zhao, Xinze Zhao, Shichang Ma. "Capacity estimation for series-connected battery pack based on partial charging voltage curve segments", Journal of Energy Storage, 2024 Publication	1 %

Exclude quotes On
Exclude bibliography On
Exclude matches < 1%

Convergence Analysis and Numerical Functions Implementation in Adaptive Particle Swarm Optimization using Tanh-Based Acceleration Coefficients

GRADEMARK REPORT

FINAL GRADE

/100

GENERAL COMMENTS

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12