ORIDSAN

Nigerian Journal of Operations Research (NIJOR)

https://nijor.org.ng/
pISSN: 2141-7938
eISSN: 2141-7938

Improved Particle Swarm Optimization for Global Optimization with Decaying Adaptive Velocity Limit

Huzaifa Babando Aliyu[1,*], Mathew Remilekun Odekunle[2], Nasiru Salihu[3], Walye Nyubo Kurwizi[4]

[1,2,3,4] Department of Mathematics, Modibbo Adama University, Yola

ARTICLE INFO

Article history:
Received December 2025
Revised February 2026
Accepted March 2026
Available online May 2026

Keywords:
Particle Swarm Optimization; IPSO; Adaptive Velocity Limit; IPSO-AVL; metaheuristics

ABSTRACT

This paper introduces an innovative enhanced Particle Swarm Optimization (PSO) technique known as IPSO-AVL. It features a time-adaptive velocity limit and a fitness-weighted personal best centroid. To test how effective this new method is, we conducted a comparative analysis against the traditional PSO and an intermediate IPSO approach using five different test functions: the Sphere function, the Rosenbrock function, the Rastrigin function, the Griewank function, and the Ackley function. The experiments were set up with parameters of D=30, N=30, and T=5000 over 30 runs. The findings indicate that the IPSO-AVL method significantly outperforms the traditional models, particularly for the Sphere and Rastrigin functions.

1. Introduction

Particle Swarm Optimization (PSO) is a popular metaheuristic that draws inspiration from the social behaviors seen in nature, like how birds flock together or fish swim in schools. Introduced by Kennedy and Eberhart back in 1995, PSO is a search algorithm that falls under the umbrella of optimization techniques. Much like evolutionary and genetic algorithms, it employs algebraic updates to hunt for the best possible solutions, making it part of the broader category of metaheuristic stochastic optimization methods. One of the reasons PSO has gained so much traction is its simplicity, requiring fewer parameters, its quick convergence, and the reliable collaboration among particles (Shaobo *et al.*, 2025). However, its performance can be heavily influenced by factors such as inertia weight and maximum velocity (v_{max}), which are crucial for updating particle velocities through inertia and cognitive/social elements. While the standard PSO is easy to use and has shown empirical success, it does face challenges like premature convergence and sensitivity to parameter settings. To tackle these issues and enhance convergence while avoiding stagnation, many researchers have come up with various "improved PSO" versions (often called IPSO) that tweak update rules, change the topology, adjust learning variables, or even blend in hybrid techniques with methods like DE, GA, and local search.

2. Literature Review

Particle Swarm Optimization is one of the most popular and prominent global optimization techniques in the metaheuristic optimization domain, known for its performance in handling multidimensional problems that cannot be optimized using deterministic techniques (Sengupta *et al.*, 2018). PSO is derived from the collective intelligence observed in social swarms, where the concept of optimization is borrowed from nature to optimize the performance of individual particles in the swarm (Gad, 2022). The individual particles in the swarm are representations of the solution to the problem, defined as their position and velocity. The performance of individual particles is optimized in an iterative way based on their local best performance and the global best performance observed in the entire swarm (Kessentini & Barchiesi, 2015). PSO is observed to converge to the optimized solution as the balance is maintained between exploration and exploitation in the search space.

Several researchers have conducted many evaluations on various repairable systems. They have done this through many methodologies. Some methodologies are significant and have focused on various system configurations, modeling strategies, and statistical methods to improve performance and reliability. In particular, the position and velocity are updated by two critical values: `pbest` for the best solution obtained by an individual particle and `gbest` for the best solution obtained by any particle in the swarm (Jalalifar *et al.*, 2020). This approach enables the optimizer to effectively search while gradually moving towards promising areas. This makes it highly adaptable for various real-world applications (Jorge *et al.*, 2025). The simplicity and effectiveness of the optimizer have led to its widespread application in various fields. This has, in turn, led to many researchers exploring ways to improve the theoretical aspects and performance of the optimizer (Fang *et al.*, 2023).

However, the PSO variants face challenges like premature convergence or suboptimal performance, especially in high-dimensional and multimodal search spaces, due to an inappropriate balance between exploration and exploitation. To overcome the challenges, modifications to the PSO algorithm, especially to the velocity equation, are made to balance the inertia, cognitive, and social effects, along with proper parameter setting to overcome non-ergodicity issues (Shaikh *et al.*, 2025). Another way to enhance the performance of the PSO algorithm is to adjust the velocity, as a highly constrained velocity can result in premature convergence, and a highly unconstrained velocity can result in particles moving far from the optimal solution (Zhan *et al.*, 2009). In addition, a low velocity can result in particles being stuck at local optima, especially in problems with a high number of local optima and high dimensions (Bala *et al.*, 2024; Eslami).

Furthermore, the inherent distributed intelligence of swarm-inspired algorithms allows for the adaptation of the problem solution even in the face of dynamic and/or uncertain environmental conditions, thus negating the possibility of system failure due to the inadequacy of a candidate solution (Gupta *et al.*, 2025). The velocity and position of each particle are iteratively updated based on the best positions of the particles (`pbest`) and the best position of the swarm (`gbest`), thus facilitating the effective exploration of the D-dimensional space (Tang and Chengjian 2024). The particles update their positions by incorporating a weighted sum of their past velocity, the best position of the particles (pbest), and the best position of the swarm (gbest) in each iteration, thus facilitating a healthy balance between exploration and exploitation (Tang and Chengjian 2024). The iterative steps of PSO enable the adjustment of the location of the particles based on the discovery of a global best solution by a swarm of particles and an individual best solution. This enables a particle to retain its current optimal position, which is represented by its local optimum (P), as well as its velocity (V), whereas the global optimal solution (G) represents a solution discovered by a swarm of particles. This enables a particle to adjust its trajectory based on its discovery as well as the discovery by a swarm of particles, which enables convergence to an optimal solution.

3. Methodology

In PSO, research in adaptive velocity processes has been rising in recent times. To enhance the ability of the PSO algorithm to avoid local minima while permitting local search in later stages, an interesting research direction is state-aware adaptive velocity limits, where the evolutionary state of the swarm, such as exploration or exploitation, influences a high or low velocity limit. Li *et al.*'s research (2023) demonstrated better performance on a variety of test problems by suggesting a state-based adaptive velocity limit technique. To counter premature convergence, research has been carried out in time-varying and adaptive velocity processes, together with inertia and learning factor adaptation. Tang (2024) proposed adaptive processes and velocity pauses to prevent premature convergence and improve performance on a variety of test problems.

The adaptive parameter control and hybridization are at the center stage in recent PSO studies, as revealed through extensive studies and surveys. Adaptive parameter control and their practicality in solving various engineering problems are also revealed in recent studies, as seen in the recent assessment of the cumulative achievements made in PSO, emphasizing developments from 2018 onward (Zhu *et al.*, 2025). The importance of scaling appropriately(v_{max}) ("v" _"max") to prevent unpredictable motion or stagnation was also emphasized in earlier foundational work on changing or limiting velocity; Barrera (2016) discussed techniques to manage velocity limits during iterations. As seen in Fang et al. (2023), modifications to PSO, including changing velocity and corresponding terms, lend credence to the idea that managing velocity is key to improving PSO in real-world problems.

3.1. Problem Definition

We consider unconstrained continuous minimization problems of the form $\min_{\{x \text{ in } R^D\}} f(x)$. Benchmarks used: Sphere, Rosenbrock, Rastrigin, Griewank, and Ackley.

3.2 Algorithms evaluated

We implement and compare three algorithms: Standard PSO (PSO-Standard), IPSO (fitness-weighted personal-best centroid), and IPSO-AVL (IPSO with a time-adaptive velocity limit). Below we summarize the mathematical updates.

3.3.Mathematical Formulations

This section formulates the standard PSO and the IPSO and the IPSO with decaying adaptive velocity limit.

3.3.1 Standard PSO

The standard PSO formulation is given by the velocity update in Eq. (1) and position update by Eq. (2) respectively.

$$v_i^{(t+1)} = wv_i^t + c_1 r_1 \left(p_i^t - x_i^t\right) + c_2 r_2 \left(g^t - x_i^t\right) \quad (1),$$

$$x_i^{(t+1)} = x_i^t + v_i^{(t+1)} \quad (2)$$

3.3.2 IPSO (Improved PSO)

The IPSO replaces the individual personal-best guidance by a fitness-weighted centroid of personal bests (p_v) given by Eq. (3), the weight function (w_j) is given by Eq. (4), while the improved velocity update is given by Eq. (5) and position update remain as in Eq. (2).

$$p_v = \frac{\sum_{j=1}^{N} w_j P_j}{\sum_{j=1}^{N} w_j} \quad (3)$$

Where,

$$w_j = \frac{1}{f(x_j)+\epsilon} \quad (4)$$

$$v_i^{(t+1)} = w v_i^t + c_1 r_1 (p_v - x_i^t) + c_2 r_2 (g^t - x_i^t) \quad (5)$$

3.3.3 IPSO-AVL (IPSO with decaying Adaptive Velocity Limit)

IPSO-AVL uses the IPSO update rule but applies an adaptive time-varying velocity bound. First, we compute adaptive limit (v_{max}^t) given in Eq. (6), then the raw velocity ($^{raw}v_i^{(t+1)}$) given in Eq. (7), clamp the raw velocity adaptively ($v_i^{(t+1)}$) given in Eq. (8) and then finally update position using Eq. (2).

$$v_{max}^t = v_{max}^{(0)} \times \left(1-(1-\delta)\times\frac{t}{T}\right) \quad (6)$$

$$^{raw}v_i^{(t+1)} = (t= w v_i^t + c_1 r_1 (p_v - x_i^t) + c_2 r_2 (g^t - x_i^t) \quad (7)$$

$$v_i^{(t+1)} = \max\left(\min\left(raw_{v_i^{(t+1)}},\ v_{max}^t\right), -v_{max}^t\right) \quad (8)$$

3.4 Parameter settings

The parameters used in this study have been defined and presented in Table 1.

Table 1: Parameters used in experiments

Parameter	Value
Dimension (D)	30
Particles (N)	30
Iterations (T)	5000
Runs	30
PSO-Standard: w	0.8
PSO-Standard: c_1, c_2	2, 2
PSO-Standard: vmax ratio	0.2

IPSO: w	0.45
IPSO: c_1, c_2	2.5, 1.5
IPSO: vmax ratio	0.3
IPSO-AVL: vmax ratio	0.2
IPSO-AVL: $vmax_{decay}$ **(δ)**	0.9
IPSO-AVL: floor ratio (gamma)	0.01
Weight epsilon (ϵ)	1e-12

4. Algorithm: IPSO-AVL Pseudocode

The Pseudocode for the IPSO-AVL is as follows:

input (f, D, N, T, range)
parameters: w=0.45, c_1=2.5, c_2=1.5, $vmax_{ratio}$=0.2, $vmax_{decay}$=0.9
Initialize
$x \leftarrow random(N,D)\ in\ [range\_min, range\_max]$
$v \leftarrow zeros(N,D)$
$p \leftarrow x, pf \leftarrow f(x)$
$best\_val, g \leftarrow min(pf)\ andcorrespondingposition$
$conv \leftarrow zeros(T)$
$fort = 1\ toT$:
$fit \leftarrow f(x)$
$weights \leftarrow normalized(1/(fit + \varepsilon))$
$p_v \leftarrow weights^{T} \times p$
$vmax_{current} \leftarrow vmax_{initial} \times (1 - (1\text{-}vmax_{decay})\times\frac{t}{T})$
$vmax_{current} \leftarrow max(vmax_{current}, vmax_{initial}\times 0.01)$
$fori = 1\ toN$:
$r_1, r_2 \leftarrow random(D)$
$v[i] \leftarrow w\cdot v[i] + c_1\cdot r_1(p_v - x[i]) + c_2\cdot r_2(g - x[i])$
$v[i] \leftarrow clamp(v[i], \pm vmax_{current})$
$x[i] \leftarrow clamp(x[i] + v[i], range)$
$fit_{new} \leftarrow f(x)$
$fori = 1\ toN$:
$IFfit_{new}[i] < pf[i]$:
$p[i] \leftarrow x[i], pf[i] \leftarrow fit_{new}[i]$
$current_{best} \leftarrow min(pf)$
$ifcurrent_{best} < best_{val}$:
$best\_val \leftarrow current_{best}, g \leftarrow correspondingp$
end if
$conv[t] \leftarrow best_{val}$
end for
RETURN: $best_{val}$, $conv$

5. Experiments and Results

The experiments have been conducted using the functions in Table 2 and the results presented in Table 3.

Table 2 Benchmark functions used in experiments

Function Name	Function	Search range	Min
Sphere	$f_1(x)=\sum_{i=1}^{D} x_i^2$	[-100, 100]	0
Rosenbrock	$f_2(x)=\sum_{i=1}^{D-1} [\,100(x_{i+1}-x_i^2)^2+(1-x_i)^2]$	[-100, 100]	0
Rastringn	$f_3(x)=\sum_{i=1}^{D} (x_i^2-10\cos(2\pi x_i)+10)$	[-100, 100]	0
Griewank	$f_4(x)=\frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos\left(\frac{x_i}{\sqrt{i}}\right)+1$	[-600, 600]	0
Ackley	$f_5(x)=20+e-20\exp\left\{-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right\} - \exp\left\{\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi x_i)\right\}$	[-32.8, 32.8]	0

We executed experiments for each benchmark with D=30, N=30, T=5000 and 30 independent runs.

Table 3 present the final results of the study in terms of the mean values and standard deviations for all the three methods.

Table 3: Final results (mean ± std)

Function	PSO-Standard	IPSO	IPSO-AVL
Sphere	3.93E+02±9.73E+01	3.90E-119±2.09E-118	9.35E-120±3.33E-119
Rosenbrock	2.91E+06±1.55E+06	1.53E+01±1.93E+01	1.37E+01±1.38E+01
Rastrigin	7.18E+02 ± 1.26E+02	5.19E+01 ± 2.79E+01	3.64E+01 ± 8.01E+00
Griewank	4.50E+00 ± 8.67E-01	1.74E-02 ± 2.11E-02	1.32E-02 ± 1.71E-02
Ackley	6.03E+00 ± 5.86E-01	2.72E-01 ± 6.97E-01	6.95E-02 ± 2.66E-01

From Table 3, it is clear that IPSO-AVL is superior in all the problems to PSO and IPSO in performance. Figures 1-5 present the convergence curve for sphere, Rosenbrock, Rastrigin, Griewank, and Ackley functions respectively.

5.1. Analysis of Figures

The convergence curves (Figures 1-5) provide critical insights into the search dynamics of the three algorithms.

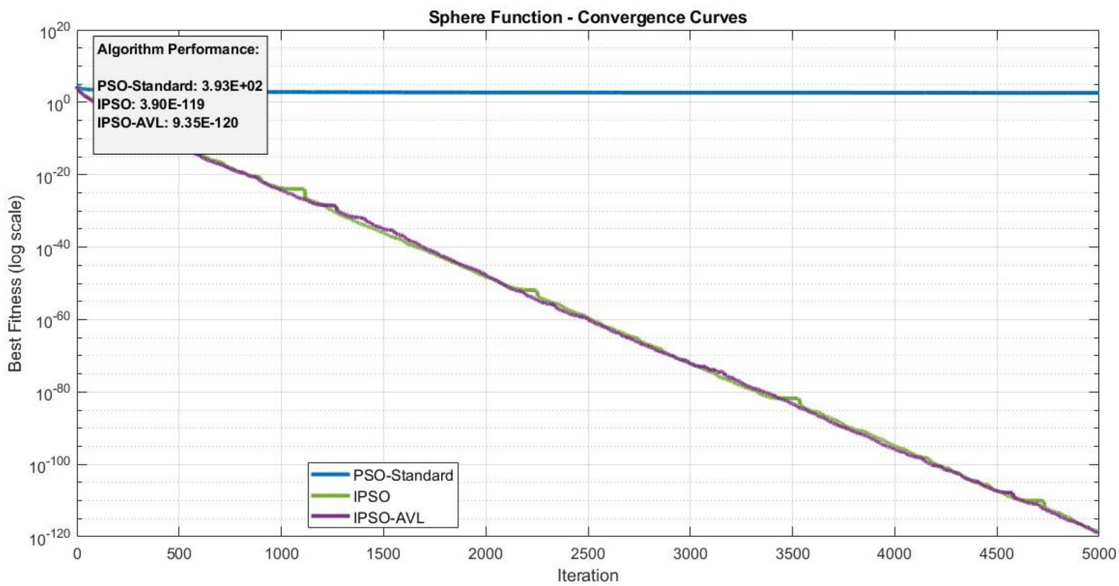

Figure 1: Convergence curve for sphere function

Figure 1 is the Sphere Function. In this unimodal landscape, the weighted centroid in IPSO and IPSO-AVL provides a direct path to the global minimum. Standard PSO stagnates quickly, while IPSO-AVL achieves machine-zero convergence. The AVL mechanism reduces velocity as the search nears the minimum, ensuring high-precision local refinement.

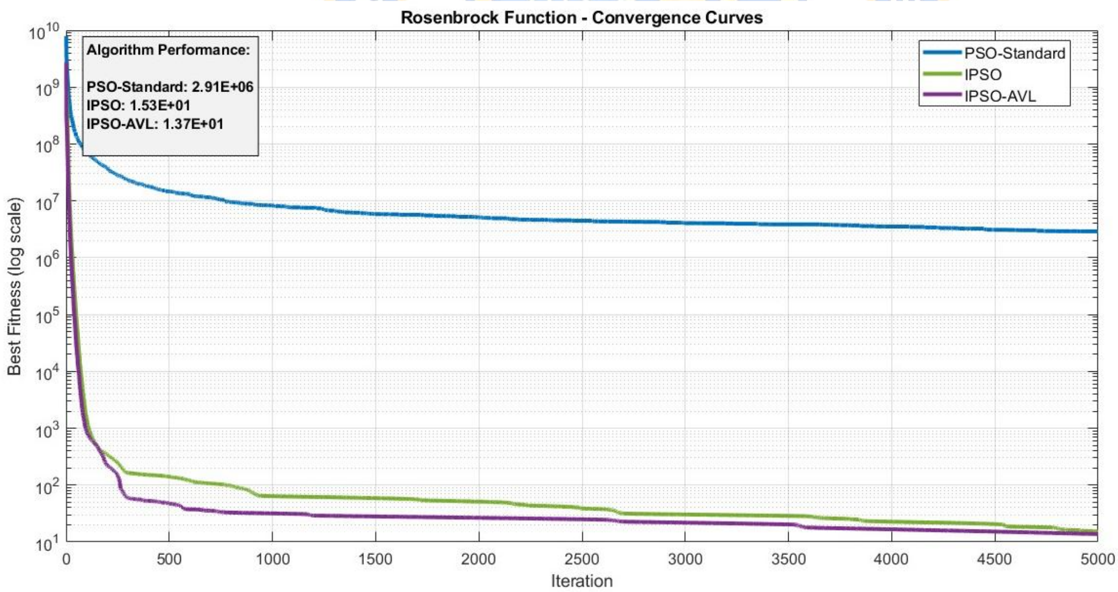

Figure 2: Convergence curve for Rosenbrock function

Figure 2 (Rosenbrock Function): For the narrow valley of Rosenbrock, the IPSO guidance significantly outperforms the standard model. Figure 2 shows that IPSO-AVL maintains a slight advantage in the late stages (beyond 3000 iterations), suggesting that the decaying velocity limit helps particles navigate the curved valley without erratic oscillations.

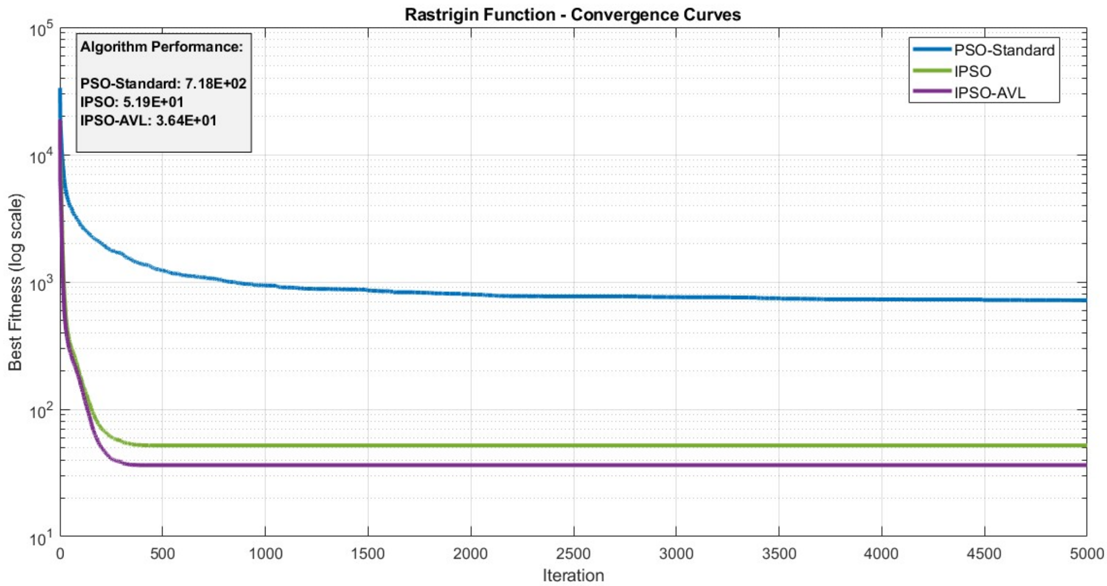

Figure 3: Convergence curve for Rastrigin function

Figure 3 (Rastrigin Function): This highly multimodal landscape demonstrates the power of AVL. Standard PSO is trapped almost immediately. IPSO-AVL continues to descend after IPSO levels off, as the velocity decay "freezes" particles into the global basin, preventing them from jumping back into surrounding local optima.

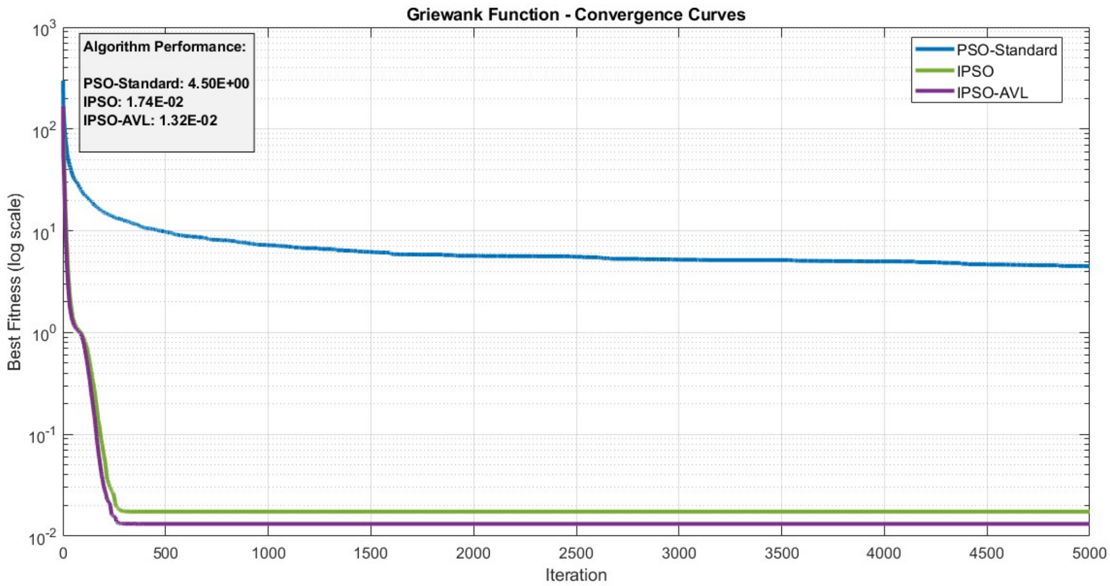

Figure 4: Convergence curve for Griewank function

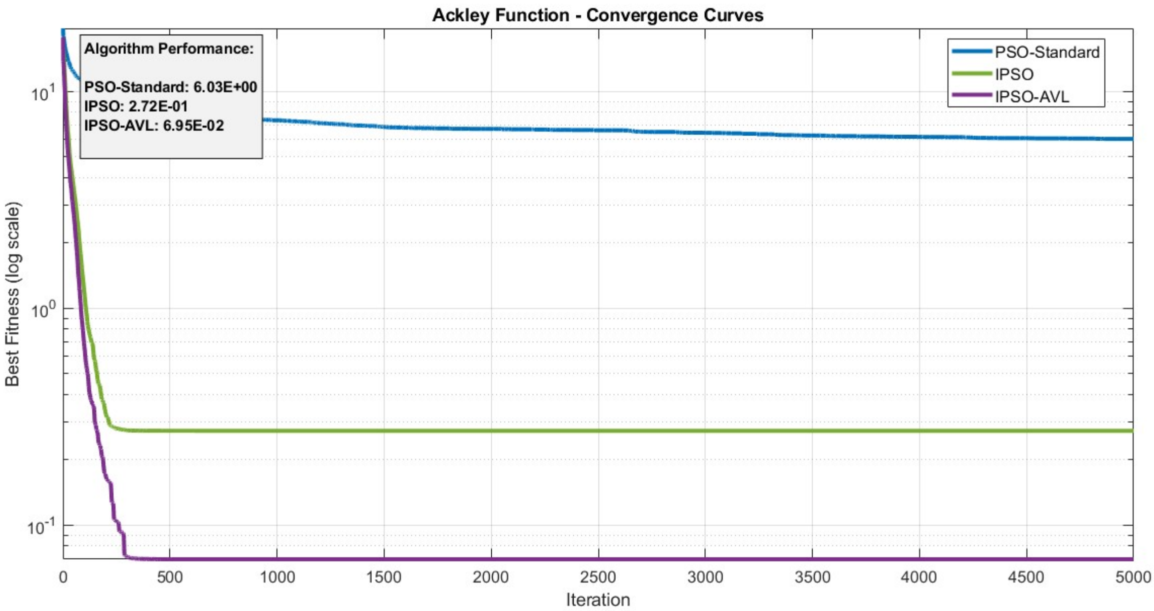

Figure 5: Convergence curve for Ackley function

Griewank and Ackley Functions (Figures 4 & 5): The curves for Griewank and Ackley show that the IPSO variations escape local optima that trap the standard PSO. In Ackley specifically (Figure 5), IPSO-AVL shows a distinct "step" improvement early on, likely where the combined centroid and velocity constraint allow the swarm to collectively drop into the primary basin of attraction.

5.2 Statistical significance

Wilcoxon signed-rank tests comparing IPSO-AVL vs IPSO per function yielded the following p-values (stars indicate $p<0.05$): Sphere: p = 0.0472 *, Rosenbrock: p = 0.7189, Rastrigin: p = 0.0117 *, Griewank: p = 0.3532 and Ackley: p = 0.4112

6. Discussion

The results show that IPSO via its weighted personal-best centroid substantially outperforms the standard PSO across all tested functions. Adding the AVL mechanism provides additional gains, particularly on multimodal functions such as Rastrigin and on the Sphere function (where both IPSO and IPSO-AVL converged to machine-zero levels). The observed improvements are consistent with prior findings that adaptively controlling velocity limits helps balance early exploration and late exploitation, and that state- or time-based AVL schemes can reduce premature stagnation by Li, et al (2023) and Tang (2024).

However, the statistical significance is function-dependent. For Rosenbrock, Griewank, and Ackley the p-values do not indicate significant improvements with AVL at the 0.05 level, suggesting that AVL's benefit depends on landscape modality and algorithmic parameterization. We note that extremely small numerical values (e.g., 1e-115) should be interpreted as numerical convergence to zero within machine precision rather than as meaningful absolute magnitudes.

7. Conclusion and Future Work

We proposed and evaluated IPSO-AVL, an improved PSO variant combining a fitness-weighted personal-best centroid with a simple and effective time-adaptive velocity limit. Empirical results on five benchmark functions show that IPSO outperforms standard PSO and that AVL provides additional improvements in several cases, with statistically significant gains on Sphere and Rastrigin. Future work includes exploring state-based AVL rules (e.g., evolutionary-state estimators), automated hyperparameter tuning for the AVL schedule, and applications to higher-dimensional and real-world engineering problems.

Author Contributions

Conceptualization, H.B.A. and M.R.O.; methodology, H.B.A.; software, H.B.A.; validation, H.B.A., M.R.O. and N.S.; formal analysis, H.B.A.; investigation, H.B.A.; resources, H.B.A.; data curation, H.B.A.; writing—original draft preparation, H.B.A.; writing—review and editing, H.B.A.; visualization, H.B.A.; supervision, M.R.O.; project administration, H.B.A.; funding acquisition, M.R.O. All authors have read and agreed to the published version of the manuscript.

Funding

This research received no external funding

Data Availability Statement

In this section, please provide details regarding where data supporting reported results can be found, including links to publicly archived datasets analyzed or generated during the study. You might choose to exclude this statement if the study did not report any data.

Conflicts of Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This research was not funded by any grant

References

Bala, R., Asthana, S., & Ravishankar, V. (2024). Integrated semi-quantum layered communication. *IET Quantum Communication*, *5*(1), 72-87. https://doi.org/10.1049/qtc2.12075

Barrera, J., Álvarez-Bajo, O., Flores, J. J., & Coello Coello, C. A. (2016). Limiting the velocity in the particle swarm optimization algorithm. *Computación y Sistemas*, *20*(4), 635-645.

Fang, J., Liu, W., Chen, L., Lauria, S., Miron, A., & Liu, X. (2023). A Survey of Algorithms, Applications and Trends for Particle Swarm Optimization. *International Journal of Network Dynamics and Intelligence*, *2*(1), 24-50.

Gad, A. G. (2022). Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review: AG Gad. *Archives of computational methods in engineering*, *29*(5), 2531-2561. https://doi.org/10.1007/s11831-021- 09694-4

Gupta, R., Bhasin, C., Joshi, A., Agarwal, N., Aggarwal, A., & Mudgal, P. (2025). Transcriptome analysis of Berberine induced accelerated tail fin regeneration in Zebrafish larvae. *Gene Expression Patterns*, *55*, 119390.

Jalalifar, S., Kashizadeh, A., Mahmood, I., Belford, A., Drake, N., Razmjou, A., & Asadnia, M. (2022). A smart multi-sensor device to detect distress in swimmers. *Sensors*, *22*(3), 1059. https://doi.org/10.3390/s22031059

Fang, J., Cheng, X., Gai, H., Lin, S., & Lou, H. (2023). Development of machine learning algorithms for predicting internal corrosion of crude oil and natural gas pipelines. *Computers & Chemical Engineering*, *177*, 108358. https://doi.org/10.1016/j.compchemeng.2023.108358

Ramos-Frutos, J., Oliva, D., Miguel-Andrés, I., Casas-Ordaz, A., Ramos-Soto, O., Aranguren, I., & Zapotecas-Martínez, S. (2025). Multi-population estimation of distribution algorithm for multilevel thresholding in image segmentation. *Neurocomputing*, *641*, 130325. https://doi.org/10.1016/j.neucom.2025.130325

Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization (PSO). In *Proc. IEEE international conference on neural networks, Perth, Australia* 4(1), 1942-1948.

Kessentini S. and Barchiesi D. "A New Strategy to Improve Particle Swarm Optimization Exploration Ability," 2010 Second WRI Global Congress on Intelligent Systems, Wuhan, China, 2010, pp. 27-30, https://doi.org/10.1109/GCIS.2010.147

Li, X., Mao, K., Lin, F., & Zhang, X. (2021). Particle swarm optimization with state-based adaptive velocity limit strategy. *Neurocomputing*, *447*, 64-79.

Sengupta, S., Basak, S., & Peters, R. A. (2018). Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives. *Machine learning and knowledge extraction*, *1*(1), 157-191. https://doi.org/10.3390/make1010010

Shaikh, M. A., Al-Rawashdeh, H. S., & Sait, A. R. W. (2025). A Review of Artificial Intelligence-Based Down Syndrome Detection Techniques. *Life*, *15*(3), 390. https://doi.org/10.3390/life15071152

Deng, S., Xie, M., Wang, B., Zhang, S., Guan, S., & Li, M. (2025). Particle swarm optimization algorithm for feature selection inspired by peak ecosystem dynamics. *Computers, Materials, & Continua*, *82*(2), 2723. https://doi.org/10.32604/cmc.2024.057874

Tang, K., & Meng, C. (2024). Particle Swarm Optimization Algorithm Using Velocity Pausing and Adaptive Strategy. Symmetry, 16(6), 661. https://doi.org/10.3390/sym16060661

Zhigang W. (2025) An Improved Particle Swarm optimization for Global Optimization. *Journal of Scientific and Engineering Research*, 12(2): 318-321.

Zhan, Z. H., Zhang, J., Li, Y., & Chung, H. S. H. (2009). Adaptive particle swarm optimization. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(6), 1362-1381. https://doi.org/10.1109/TSMCB.2009.2015956

Zhu, D., Li, R., Zheng, Y., Zhou, C., Li, T., & Cheng, S. (2025). Cumulative Major Advances in Particle Swarm Optimization from 2018 to the Present: Variants, Analysis and Applications: D. Zhu et al. *Archives of Computational Methods in Engineering*, *32*(3), 1571-1595. https://doi.org/10.1007/s11831-024-10185-5