Research on Engineering Structures & Materials
www.jresm.org
Research Group

Research Article

An improved PSO algorithm based on human approach with levy flight distribution applied to mechanical problems

Asma Gouzi *,[a], Siham Ouhimmou [b]

Laboratory of Mechanics, Faculty of Sciences Ain Chock, Hassan II University, Casablanca, Morocco

Article Info

Article History:

Received 15 May 2025

Accepted 11 Oct 2025

Keywords:

TVETPSO;
PSO;
Human strategy learning;
Levy flight;
Mechanical engineering design

Abstract

This study presents a novel enhancement to Technical & Vocational Education & Training based on the Particle Swarm Optimization algorithm (TVETPSO). In this approach, we integrate the Levy Flight strategy in the self-improvement phase rooted in human strategy learning, aimed at improving the optimization capabilities of the algorithm. TVETPSO approach, while effective, often limited exploration of the solution space. By leveraging the Levy flight strategy, characterized by its unique movement patterns of occasional large leaps and frequent smaller steps, our enhancement enables particles to escape local optima and better exploit their experiences. This dynamic exploration facilitates more effective learning and adaptation, leading to improved convergence rates and efficient solutions. In the context of mechanical engineering, this enhancement holds significant promise for optimizing design processes, structural analysis, and resource allocation, ultimately contributing to more efficient and innovative engineering solutions. By fostering a balance between exploration and exploitation, our approach not only advances optimization methodologies but also opens new avenues for applications in complex mechanical systems.

1. Introduction

Optimization problems in the real world have become increasingly complex, requiring more efficient solving methods [1]. The Particle Swarm Optimization algorithm has captured a surge of interest among researchers, it starts with a population of particles exploring the search space, these particles can process and analyze information while retaining memory of the best position encountered, and the algorithm combines knowledge and experiences of the swarm to converge towards the optimal solution [2]. This cooperative approach enables the particles to capitalize on the wisdom of the swarm. The PSO algorithm has proven to be effective and discovering solutions to various optimization problems in the real- world [3], it has demonstrated successful application in various engineering fields involved in mechanical engineering [4], biomedical image registration [5], communication networks [6], and flexible job-shop scheduling [7]. Although PSO is known for its simplicity and fast convergence rate, it's not without its limitations. Researchers have identified two notable shortcomings: premature convergence too early in the search process and Poor global search ability, which may struggle to explore optimal solutions and fall into local optimal in a larger search space. However, the researchers have responded by addressing these challenges. Various enhancements of PSO have been proposed and developed over the years, these modifications aim to overcome the deficiencies of the original algorithm and improve its overall performance, and it can be classified into several distinct strategies:

The Parameter adaptation on PSO has become the most appealing in research works, as for the time-varying acceleration coefficients introduced by Ratnaweera and Halgamuge aim to adjust the

*Corresponding author: asmaa.00gouzi@gmail.com
[a]orcid.org/0009-0008-3567-9988; [b]orcid.org/0000-0002-0703-7127
DOI: http://dx.doi.org/10.17515/resm2025-899me0515rs

acceleration coefficients over time to reduce the cognitive component and increase the social component, particles are encouraged to explore the search space focusing on the population's bets position [8], time-varying inertia weights proposed by Yang & al which have been developed by dynamically adjusting the influence of the velocity and position of particles updates throughout the optimization process [9], adaptive inertial weight proposed by Shi and Eberhart [10].

Population Topology Structure on PSO is a strategy to enhance the PSO Algorithms, each particle evaluates the information based on all other particles. However, the population size is considered challenging to determine a suitable size due to increased computation costs and slower convergence with a larger population while a smaller population may struggle to a local optimal, in this case, the researchers have explored the concept of dividing the population into multiple subgroups by investigating different topology structures. Yu & al proposed a surrogate-assisted hierarchical particle swarm optimizer [11], Zhang and al proposed an enhancing comprehensive learning particle swarm optimization with local optima topology [12], Zhang & al proposed a terminal crossover and steering-based particle swarm optimization algorithm with disturbance [13].

Novel learning strategies, Comprehensive learning particle swarm optimizer for global optimization of multimodal functions developed by Liang & al, CLPSO has demonstrated improved convergence speed and high-quality solution, it aims to improve exploration and exploitation capabilities from different areas of the search space. This approach has found applications in various domains, including engineering, data mining, and optimization of machine learning models [14]. Zhan and al presented an orthogonal learning strategy for PSO [15]. A social learning PSO presented by Zhang et al [16].

Hybridization-based PSO algorithm refers to a class of optimization strategies that combines Particle Swarm Optimization (PSO) with other computational algorithms such as local search algorithms, evolutionary algorithms, or machine learning methods to enhance the performance and capabilities of the basic PSO Algorithm. Hybrid PSO Algorithms can handle complex optimization problems more effectively. Angeline proposed an evolutionary selection operator [17], a hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems presented by Aydilek [18]. Jindal and Bedi proposed an improved hybrid ant particle algorithm for reducing travel time in vanets [19].

Recent studies have emphasized the role of optimization in advancing both structural and design engineering. For instance, Muthusamy and Patil (2023) demonstrated that applying optimization techniques to nano-reinforced concrete composites significantly enhances their mechanical properties, underlining the benefits of material-level optimization in structural applications [20]. In the field of design methodology, Guo, Allen, and Mistree (2024) examined optimization versus satisficing strategies in engineering design, providing valuable insights into managing multi-objective and uncertain design problems [21].

This work aims to contribute a novel enhancement of the PSO hybrid with the learning strategy approach and the levy flight distribution treated for design optimization problems.

2. TVETPSO Algorithm

The TVETPSO has been recently developed as a new hybrid algorithm that incorporates the swarm features with human learning allowing the population to learn & develop their skills using the human approach for good exploration and exploitation by simulating multiple mathematical modeling techniques, mitigating their weaknesses. This hybridization-based optimization process is capable of navigating complex, multimodal landscapes where traditional algorithms might struggle which is particularly robust in engineering design in a complex problem that requires precision in the mechanical field to enhance the quality solution against the local optima in the search space [22].

2.1. PSO Algorithm Overview

PSO is a population-based optimization algorithm inspired by social behaviour in birds and fish. It operates using a swarm of particles, each representing a potential solution to an optimization problem. Each particle adjusts its position in the solution space based on its own experience and the experience of neighbour particles [2]. These Particles are initialized with random positions and velocities within the search space. Each particle's velocity is updated based on its own best-known position and the best-known position of its neighbors:

$$V_i^d = wV_i^d + c_1 rand1_i^d * \left(Pbest_i^d - X_i^d\right) + c_2 rand2_i^d * \left(Gbest_i^d - X_i^d\right) \tag{1}$$

Where $Pbest_i^d$ is the best position of ith particle of the dth dimension, $Gbest_i^d$ is the global best position of ith particle of the dth dimension, c_1 and c_2 are the acceleration coefficients. W is the inertia weight and $rand1_i^d$ and $rand2_i^d$ are two uniform random numbers defined in dth dimension in the range [0, 1]. The new position of the particle is updated as:

$$X_i^d = X_i^d + V_i^d \tag{2}$$

In this hybrid approach, the standard PSO method is enhanced by introducing three different strategies for updating the positions of particles, which can be respectively chosen based on certain criteria.

2.2. TVET Algorithm Overview

TVET algorithm is a relatively recent metaheuristic optimization algorithm inspired by the Human Learning process in educational environments that mimics how candidates learn from instructors and each other, making it particularly effective for solving complex optimization problems.

The TVET algorithm consists of three main phases: the Theory Education Phase, the Practical Education Phase, and the Improving Individual Skills Phase. These phases simulate the process of education and training in a classroom environment [23].

2.2.1 Theory Education Phase

Theoretical Knowledge Transfer: The instructor imparts knowledge to the candidates (other solutions) to improve their fitness. This is established by adjusting the candidates' positions towards the instructors' position using a predefined step size. The update for a candidate learner can be mathematically expressed as:

$$Xnew_i^d = X_i^d + r.(I_d - SX_i^d) \tag{3}$$

Where $Xnew_i^d$ is the updated position of ith particle of the dth dimension, r is a uniform random in the range [0, 1], I_d is the instructor of the member’s candidate of the dth dimension and S is a random number from the set [1, 2].

2.2.2 Practical Education Phase

Workshop Training: In this phase, candidates interact with each other to enhance their knowledge. Each candidate attempts to improve their position based on the performance of others. A typical update for a candidate in this phase is represented as:

$$Xnew_i^d = I_d + r.\frac{t}{T}\left(I_d - X_i^d\right) \tag{4}$$

Where t denotes the actual generation and T is the generation’s number.

2.2.3 Improving Individual Skills Phase

Self-Improvement: The candidate makes a small change by performing their own experience & learning. A typical update for a learner in this phase is represented as:

$$Xnew_i^d = X_i^d + (1 - 2.r).\ \frac{(ub_d - lb_d)}{t} \tag{5}$$

Where lb_d and ub_d represent the lower and upper boundaries.

2.2.4 Computation Complexity of TVETPSO

The initialization step for an optimization problem has a complexity of O(Pn) where P is the number of population members and N is the number of decision variables. Each iteration, which includes updates during both the PSO and TVET phases, incurs a complexity of O(4PnT) with T representing the maximum number of iterations. Thus, the overall complexity of the TVETPSO algorithm is O(Pn(1+4T)).

Table 1. Pseudo-code of TVETPSO Algorithm

TVETPSO Algorithm
Define Optimization Problem: Objective function, Constraint functions and variables
Input parameter settings: Population, Dimension, Iterations, and PSO Coefficients…
Initiate population, velocity & function evaluation.
Update Gbest & Pbest.
For i=1:T
For j=1:N
Using equation (1) to update velocity.
Using equation (2) to update Position.
Update Pbest & Gbest based on improved fitness.
Set the Gbest as an Instructor of the ith population.
Phase 1: Calculate the new position by equation (3)
Update the new position for improved fitness.
Phase 2: Calculate new position by equation (4)
Update the new position for improved fitness.
Phase 3: Calculate new position by equation (5)
Update the new position for improved fitness.
End
Storage of the Instructor.
End
Output

3. Levy Flight

Levy flights are intriguing stochastic processes that have been observed in various natural systems, including animal foraging patterns and the movement of particles in complex environments. These processes are characterized by their erratic trajectories, which can sometimes lead to significant distances being covered in a short time. The concept of the Levy distribution, developed by the French mathematician Paul Lévy in the 1930s, plays a crucial role in understanding these movements. Recent applications of Levy flight principles have emerged in optimization techniques, such as algorithms designed for solving complex problems in logistics to machine learning [24].

In optimization contexts, Levy flight has become a valuable tool. Algorithms that incorporate this method, such as those inspired by natural phenomena like cuckoo breeding behaviours, benefit from enhanced exploration capabilities. By enabling occasional large leaps in the search space, Levy flight helps avoid local optima, thereby improving the likelihood of finding a global solution. This is particularly useful in complex landscapes where traditional search methods might struggle. Due to the intricate characteristics of the Levy distribution, researchers often rely on the Mantegna algorithm for efficient simulation. The size of each step taken during a Levy flight can be calculated using the formula:

$$Flight_{size} = \frac{\mu}{|\nu|^{1/\gamma}} \tag{6}$$

In this equation, γ=1.5 is a defining parameter of the Levy distribution, while μ and ν are random variables derived from a normal distribution, represented as:

$$\begin{cases} \mu \sim N(0, \tau_\mu^2) \\ \nu \sim N(0, \tau_\nu^2) \end{cases} \tag{7}$$

Here, τ_u and τ_v are defined by:

$$\begin{cases} \tau_u = \{\frac{\Gamma(1+\gamma) sin\,(\pi\gamma/2)}{\Gamma(1+\gamma/2)\gamma 2^{(\gamma-1)/2}}\}^{1/\gamma} \\ \tau_v = 1 \end{cases} \tag{8}$$

Consequently, the effective step size for particles can be expressed as:

$$PS_{size} = \rho \times Flight_{size} \tag{9}$$

4. Enhanced TVETPSO Algorithm with Levy Flight Distribution: LF_TVETPSO

In this study, we present a novel enhancement to the TVETPSO algorithm by integrating Levy flight into the self-improvement phase, aiming to improve the algorithm's ability to exploit particle experiences and facilitate more effective learning processes. Traditional TVETPSO relies on standard movement strategies, which can sometimes limit exploration and lead to local optima. By incorporating Levy flight, characterized by occasional large jumps and frequent smaller steps, our approach mimics natural foraging behaviors, allowing for a dynamic exploration of the solution space. This enhancement offers several advantages: it improves exploration by enabling particles to make substantial leaps to escape local optima, facilitates better utilization of past experiences by encouraging particles to refine their search strategies based on promising regions, and promotes dynamic learning where particles adapt their movement patterns based on accumulated knowledge. Additionally, this integration balances exploration and exploitation, creating a more robust optimization process. Overall, our enhancement to TVETPSO represents a significant advancement in particle swarm optimization methodologies, leading to improved performance and opening new research avenues in optimization techniques applicable across various fields, from engineering to machine learning.

$$Xnew_i^d = X_i^d + (1 - 2.r).\frac{(ub_d - lb_d)}{t} + PS_{size} \tag{10}$$

Table 2. Pseudo-code of: TVETPSO algorithm

LF_TVETPSO Algorithm
Define Optimization Problem: Objective function, Constraint functions and variables
Input parameter settings: Population, Dimension, Iterations, and PSO Coefficients…
Initiate population, velocity & function evaluation.
Update Gbest & Pbest.
For i=1:T
For j=1:N
Using equation (1) to update velocity.
Using equation (2) to update Position.
Update Pbest & Gbest based on improved fitness.
Set the Gbest as an Instructor of the ith population.
Phase 1: Calculate the new position by equation (3).
Update the new position for improved fitness.
Phase 2: Calculate new position by equation (4).
Update the new position for improved fitness.
$Flight_{size} = \frac{\mu}{\lvert\nu\rvert^{1/\gamma}}$
$PS_{\text{size}} = \rho \times Flight_{size}$
Phase 3: Calculate new position by equation:

$$\mathrm{Xnew}_i^d = X_i^d + (1 - 2.r).\ \frac{(ub_d - lb_d)}{t} + PS_{size}$$

Update the new position for improved fitness.

End

Storage the Instructor.

End

Output

5. Numerical Analysis and Comparison

5.1 Benchmark Functions

In this section, we conduct comparative experiments to evaluate our proposed LF_TVETPSO, alongside several other configurations, including the TVETPSO algorithm, the basic TEVT algorithm, and PSO. These experiments are performed on unimodal and multimodal benchmark functions in a 30-dimensional space. For the experiments, the population size is set to 50 individuals, and the algorithms are allowed to run for 2000 iterations. The algorithms are implemented in MATLAB and executed on a system with an Intel Core i7 processor running at 2.90 GHz. The system has 4 physical cores, 8 logical processors, and is equipped with 16 GB of RAM.

5.2 Results and Discussion

The performance of the proposed optimization algorithm LF_TVETPSO along with other algorithms; TVETPSO, TVET, and PSO was evaluated on 12 benchmark functions, each representing different optimization problems. The algorithms were compared based on statistical measures, including the best, mean, worst, and standard deviation of the objective values. A detailed analysis of the results reveals significant differences in the performance of these algorithms across various benchmark functions. Overall, the LF_TVETPSO algorithm demonstrated superior performance across most functions, consistently providing the best results for both the best and mean objective values. This indicates that the LF_TVETPSO algorithm is highly effective in finding optimal solutions within the search space. The TVETPSO algorithm emerged as a strong contender, often providing consistent results with low standard deviation, making it a more reliable option than TVET and PSO. For example, in F2 and F4, TVETPSO performed well with minimal variation in the results, demonstrating its robustness. However, it did not always outperform LF_TVETPSO in terms of the best objective value, as LF_TVETPSO typically achieved optimal solutions. In contrast, the TVET algorithm showed poor performance across most functions, particularly in terms of stability. It exhibited high variability, as evidenced by its very large standard deviations, which indicates that it struggled to find good solutions consistently. While the TVET algorithm did perform better on some functions, such as F5, where it reached a significantly lower best value, its high variance (e.g., Std for F5) renders it less reliable for practical applications. On the other hand, PSO exhibited a mixed performance. While it showed stability on several functions, it did not achieve the same level of accuracy as LF_TVETPSO in most cases. For instance, in F1 and F2, PSO performance was slightly worse than TVETPSO, and it struggled to match the optimality achieved by LF_TVETPSO.

In terms of worst values, TVET performed poorly, with some instances of extremely high worst values (e.g., F1), highlighting its instability. The analysis also highlighted that LF_TVETPSO consistently provided the lowest worst values which is particularly suited for problems where global optimality is crucial, especially for continuous objective functions (e.g., F1, F2, and F3). TVETPSO, with its smaller standard deviation, showed more consistent results across all functions. However, it was still outperformed by LF_TVETPSO in terms of optimality, offering consistently good performance without the large variances observed in TVET and PSO. The analysis of the convergence behavior highlights that LF_TVETPSO consistently demonstrated the best convergence behavior, approaching the optimal solution across all functions. This rapid convergence, combined with its ability to maintain stability in its objective values, makes LF_TVETPSO the most suitable choice for problems where both optimality and fast convergence are crucial. In summary, LF_TVETPSO is the best-performing algorithm across most of the benchmark functions, achieving superior optimality. TVETPSO is a strong competitor, especially when consistency is prioritized, while TVET and PSO show weaknesses in stability and convergence.

Table 3. Benchmark functions

Type	Name	Benchmark Functions	Search Range	f_{min}
	Sphere (F1)	$f_1(x)=\sum_{i=1}^{D} x_i^2,$	$[-100,100]^D$	0
	Schwefel 2.22 (F2)	$f_4(x)=\sum_{i=1}^{D}\lvert x_i\rvert+\prod_{i=1}^{D}\lvert x_i\rvert,$	$[-10,10]^D$	0
	Schwefel 1.2 (F3)	$f_2(x)=\sum_{i=1}^{D}\left(\sum_{j=1}^{i} x_j\right)^2$	$[-100,100]^D$	
	Schwefel 2.21 (F4)	$f_3(x)=\max\left(\lvert x_i\rvert\right),i=1,2,\ldots,D$	$[-100,100]^D$	0
Unimodal	Rosenbrock (F5)	$f_5(x)=\sum_{i=1}^{D} 100\times(x_{i+1}-x_i^2)^2+(1-x_i)^2$	$[-10,10]^D$	0
	Step (F6)	$f_6(x)=\sum_{i=1}^{D} 100\times(x_{i+1}-x_i^2)^2+(1-x_i)^2$	$[-10,10]^D$	0
	Quartic (F7)	$f_7(x)=\sum_{i=1}^{D} ix_i^4+\text{random}\,[0,1]$	$[-1.28,1.28]^D$	0
	Rastrigin (F8)	$f_8(x)=\sum_{i=1}^{D}[x_i^2-10\cos(2\pi x_i)+10]$	$[-5.12,5.12]^D$	0
	Ackley (F9)	$f_9(x)=-20\exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right)-\exp\left(\sum_{i=1}^{D}\cos 2\pi x_i\right)+20+e,$	$[-32,32]^D$	0
	Griewank (F10)	$f_{10}(x)=\frac{1}{4000}\sum_{i=1}^{D} x_i^2-\prod_{i=1}^{D}\frac{x_i}{\sqrt{i}}+1,$	$[-600,600]^D$	0
	Penalized 1 (F11)	$f_{11}(x)=\frac{\pi}{D}\left\{10\sin^2(\pi y_i)+\sum_{i=1}^{D-1}(y_i-1)^2\cdot[1+10\sin^2(\right.$ $+\sum_{i=1}^{D} u(x_i,10,100,4),\text{where } y_i=1+0.25(x_i+1),$ $u(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, x_i>a \\ 0,\ -a\le x_i\le a \\ k(-x_i-a)^m, x_i<-a \end{cases}$	$[-50,50]^D$	0
Multimodal	Penalized 2 (F12)	$f_{12}(x)=0.1[\sin^2\pi 3y_i+\sum_{i=1}^{D}(y_i-1)^2\cdot[1+\sin^2(3\pi y_i$ $+(x_n-1)^2[1+\sin^2(2\pi y_D)]]+\sum_{i=1}^{D} u(x_i,10,100,4)$ where $y_i=1+0.25(x_i+1),(x_i,a,k,m)=\begin{cases} k(x_i-a)^m, x_i>a \\ 0,\ -a\le x_i\le a \\ k(-x_i-a)^m, x_i<-a \end{cases}$	$[-50,50]^D$	0

Table 4. Statistical Results on Benchmark Functions for 30-D

Benchmark Functions	Statistics	LF_TVETPSO	TVETPSO	PSO	TVET
F1	Best	0	0.001337248	0.001629439	0.00238084
	Mean	0	0.00162635	2000.00686	0.002666359
	Worst	0	0.001875768	10000.00032	0.002953222
	Std	0	0.00019579	4472.132297	0.000270805
F2	Best	1.3946E-205	0.015149279	20.00415664	0.020526395
	Mean	4.3539E-203	0.016147028	38.00205737	0.021866472
	Worst	1.3464E-202	0.016946148	50.00043276	0.023409264
	Std	0	0.000836006	13.03648617	0.001186268
F3	Best	0	0.000190805	6256.753985	0.005162815
	Mean	0	0.000338658	8246.741513	0.006342233
	Worst	0	0.000539546	10936.18582	0.007721469
	Std	0	0.00012778	2408.281279	0.000980763
F4	Best	3.6578E-206	0.015166071	3.244514278	0.018335134
	Mean	6.7901E-203	0.016582498	4.100603733	0.020411718
	Worst	1.7391E-202	0.01816344	4.622138865	0.021474328
	Std	0	0.001125214	0.569124304	0.001281094
F5	Best	9.499501437	27.3016577	36.94097885	27.77304531
	Mean	11.0526833	27.4030115	364.655996	27.84780643
	Worst	14.03659589	27.59000568	1020.253114	27.921559
	Std	1.766692996	0.109391857	394.2630169	0.058047578
F6	Best	2.37275E-31	0.000148122	0.002409307	0.044862035
	Mean	9.17877E-22	0.006875711	0.021803873	0.061123303
	Worst	4.58938E-21	0.012313232	0.066649544	0.114352154
	Std	2.05244E-21	0.006154542	0.026339094	0.029810594
F7	Best	3.17275E-05	2.60452E-05	0.011445317	2.78137E-06
	Mean	6.32971E-05	8.94719E-05	1.089195571	6.02673E-06
	Worst	0.000101789	0.000160996	2.701960776	8.39017E-06
	Std	2.59144E-05	5.54262E-05	1.46812863	2.60906E-06
F8	Best	0	0.000670282	114.1657152	0.001220756
	Mean	0	7.364846508	163.8488413	23.16191342
	Worst	0	18.9080737	209.3922034	59.54711588
	Std	0	10.08961327	41.91674238	31.73530702
F9	Best	4.44089E-15	0.009500604	0.026829572	0.011896258
	Mean	11.94236415	0.01009854	5.867713731	0.012817086
	Worst	19.93811644	0.010429866	14.72110139	0.013964903
	Std	10.90187094	0.000370673	7.964028929	0.000821668
F10	Best	0	0.002826306	0.034432158	0.003386444
	Mean	0	0.007835158	18.30369424	0.009818895
	Worst	0	0.016502169	90.94503065	0.015055563
	Std	0	0.006640445	40.60794429	0.005599352
F11	Best	1.7803E-32	1.74137E-11	0.000341553	0.004328196
	Mean	0.020733804	7.9426E-10	0.02276477	0.00529677
	Worst	0.10366902	2.92922E-09	0.090874565	0.007358588
	Std	0.046362195	1.21889E-09	0.03864991	0.001191502
F12	Best	7.94278E-30	4.56818E-11	0.006277228	0.11670952
	Mean	0.158088011	7.46563E-08	0.025187377	0.158921356
	Worst	0.549076372	3.59808E-07	0.042455793	0.208057787
	Std	0.2257989	1.59442E-07	0.015043878	0.033016445

6. Engineering Design Problems

The engineering design problems enable the evaluation of the algorithm's performance and develop efficient solutions for a wide range of complexities. In this section, we have chosen three constraint design applications; Cantilever design problem, Tree bar truss design problem, and Gear Train design problem to validate the proposed algorithm and assess its effectiveness in handling constraint [25]. These problems are often standard in the engineering optimization field with appropriate measures such as the convergence rate and the efficiency of the solutions across varying problem complexities. We conducted experiments using:

- Number of populations: 500
- Number of iterations: 2000
- c_1 and c_2 are between 2 and 2.99
- w: between 0.4 and 0.99

6.1 Cantilever Beam Design Problem

The cantilever beam design problem is a constrained optimization problem, as shown in Fig 2. This problem is defined by five hollow blocks which are rigid by one end while the other end is left free which is applied by a vertical force. This design optimization aims to minimize the weight of the cantilever. The mathematical expression is given by the equations below:

Minimize

$$f(x) = 0.0624(x_1 + x_2 + x_3 + x_4 + x_5) \tag{11}$$

Subject to:

$$g(x) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \leq 0 \tag{12}$$

The design variables have bounded as follows: $0.01 \leq x_i \leq 100$, i=1, 2,3,4,5

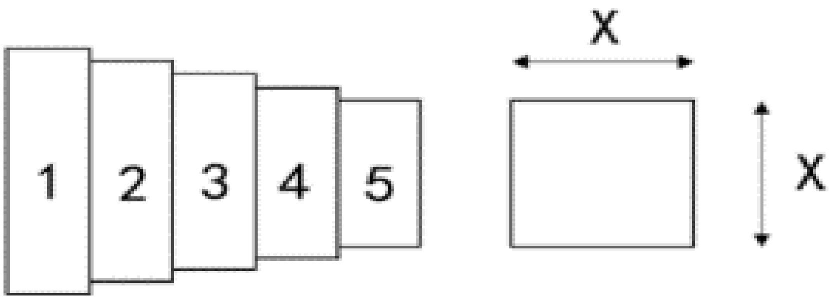

Fig. 1. Cantilever beam design

6.1.1 Results and Discussion for Cantilever Beam Study

The results show good optimum solutions of the design variables for the LF-TVETPSO, TVETPSO, and PSO basic algorithms, as presented in Table 5. The outcomes demonstrate that the LF-TVETPSO algorithm effectively addresses the cantilever beam problem by minimizing its weight.

Table 5. Design Variables of Cantilever Beam Design Problem

Design Variables	LF-TVETPSO	TVETPSO	PSO Basic
x1	6.0231	6.0230	6.0085
x2	5.3202	5.3056	5.3078
x3	4.4794	4.4880	4.4982
x4	3.5038	3.5057	3.5026
x5	2.1473	2.1514	2.1566
f	1.34	1.34	1.34

6.2 Tree Bar Truss Design Problem

This optimization problem is an optimal design to minimize the volume of the three-bar truss structure subject to stress constraints shown in Fig 2. The mathematical formulation is described below. Consider Design Variables: X= $[x_1, x_2]$ = [A1, A2]

Minimize

$$f(x) = (2\sqrt{2}\, x_1 + x_2)l \tag{13}$$

Subject to:

$$g_1(x) = \frac{\sqrt{2}\, x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0 \tag{14}$$

$$g_2(x) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2}P - \sigma \leq 0 \tag{15}$$

$$g_3(x) = \frac{1}{\sqrt{2}x_2 + x_1}P - \sigma \leq 0 \tag{16}$$

The design variables are bounded as follows: $0.05 \leq x_1, x_2 \leq 2$

Where: P= 2KN/cm2, σ = 2KN/cm2, l= 100cm

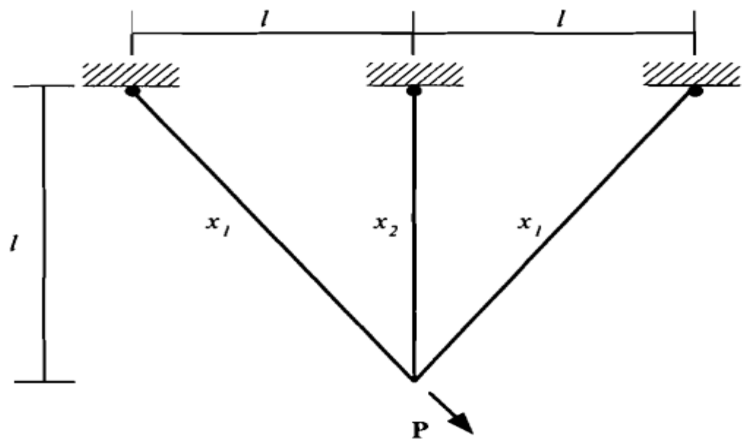

Fig. 2. Tree Bar Truss design problem

6.2.1 Results and Discussion for Tree Bar Truss Study

We conducted a numerical analysis between the LF-TVETPSO algorithm, TVETPSO and PSO Basic. The results are presented in Table 6 indicating that LF-TVETPSO achieved optimal solutions demonstrating the effectiveness of LF-TVETPSO in addressing this constraint problem.

Table 6. Design Variables of Tree bar Truss Design Problem

Design Variables	LF-TVETPSO	TVETPSO	PSO Basic
x1	0.7888	0.7888	0.7887
x2	0.4080	0.4080	0.4083
f	263.8958	263.8959	263.8958

6.3 Gear Train Design Problem

This structural issue is commonly identified as an unconstrained optimization problem. As demonstrated in Fig. 3, the core challenge revolves around determining the appropriate gear ratio within the minimized map. The design variables A, B, C, and D are crucial to solving this optimization problem [20]. The mathematical model representing this problem is given by equation [16]:

Consider $X = [x_1, x_2, x_3, x_4] = [n_A, n_B, n_C, n_D]$

Minimize

$$f(x) = \left(\frac{1}{6.931} + \frac{x_2 x_3}{x_1 x_4}\right)^2 \tag{16}$$

Variable range $12 \le x_1, x_2, x_3, x_4 \le 60$

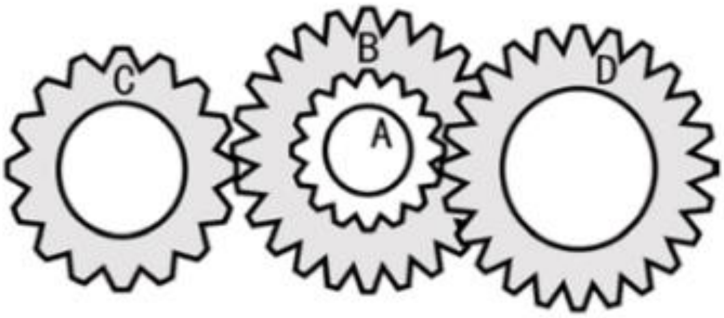

Fig. 3. Gear Train structure design

6.3.1 Results and Discussion for Gear Train Study

The optimization result was successfully obtained by using LF-TVETPSO, delivering excellent outcomes as presented in Table VI. The results highlight the efficiency of LF-TVETPSO in addressing the Gear Train design optimization.

Table 9. Design Variables of Gear Train Design Problem

Design Variables	LF-TVETPSO	TVETPSO	PSO
x1	12.0167	12.1067	32.3002
x2	37.3837	18.1457	12.0000
x3	53.6627	58.5048	50.8193
x4	58.0220	26.0259	52.8632
f	0	1.0446e-24	8.0026e-19

7. Conclusion

Based on the results obtained in this study, we propose an enhanced optimization approach that integrates the Levy flight strategy into the TVETPSO algorithm. The LF_TVETPSO algorithm itself is inspired by human learning behavior, allowing particles to improve their positions through a guided learning process. By incorporating the Levy flight mechanism, the algorithm achieves a more effective balance between exploration and exploitation, which is critical for avoiding local optima and enhancing the overall convergence speed. This integration enables the algorithm to explore the search space more thoroughly while still refining promising solutions efficiently, leading to improved optimization outcomes.

The performance of the proposed approach was evaluated using a set of well-established benchmark functions, which serve as standard tests for optimization algorithms. The results indicate that the LF_TVETPSO algorithm, augmented with Levy flight, consistently delivers accurate and robust solutions across diverse problem types. In addition to synthetic benchmarks, the algorithm has also been applied to practical design optimization problems within the field of mechanical engineering; Cantilever design problem, Tree bar truss design problem, and Gear Train design problem demonstrating its ability to handle complex, real-world application effectively. These results highlight not only the reliability and efficiency of the proposed approach but also its versatility in addressing a wide range of optimization challenges.

Looking forward, further refinements of the LF_TVETPSO algorithm are planned to enhance its performance even further. Potential improvements include adaptive parameter tuning, hybridization with other optimization strategies, and the incorporation of problem-specific metaheuristic optimization to tailor the algorithm to specialized applications in Mechanical Structures. By continuing to refine and expand the capabilities of the proposed method, we aim to establish it as a powerful and practical tool for solving complex engineering optimization problems. Overall, the integration of human-inspired learning mechanisms with Levy flight strategies represents a promising direction for advancing optimization techniques and addressing increasingly challenging design and engineering problems.

Acknowledgement

The author authors express gratitude to the associate editor and reviewers for their contributions to improving the quality of the article throughout this research.

References

[1] El Moumen S, Ouhimmou S. A new multiobjective optimization algorithm using NBI-SASP approaches for mechanical structural problems. Int J Simul Multidiscip Des Optim. 2022;13(4). https://doi.org/10.1051/smdo/2021037
[2] Kennedy J, Eberhart RC. Particle swarm optimization. In: Proc IEEE Int Conf Neural Networks. Perth, Australia; 1995. p. 1942-8. https://doi.org/10.1109/ICNN.1995.488968
[3] Eberhart RC, Shi YH. Particle swarm optimization: Developments, applications, and resources. In: Proc IEEE Congr Evol Comput. Seoul, Korea; 2001. p. 81-6. https://doi.org/10.1109/CEC.2001.934374
[4] Hasanoglu MS, Dolen M. Feasibility-enhanced particle swarm optimization for constrained mechanical design problems. Proc Inst Mech Eng Part C J Mech Eng Sci. 2018;232:381-400. https://doi.org/10.1177/0954406216681593
[5] Wachowiak MP, Smolikova R, Zheng Y, Zurada JM, Elmaghraby AS. An approach to multimodal biomedical image registration utilizing particle swarm optimization. IEEE Trans Evol Comput. 2004;8(3):289-301. https://doi.org/10.1109/TEVC.2004.826068
[6] Soo KK, Siu YM, Chan WS, Yang L, Chen RS. Particle-swarm-optimization-based multiuser detector for CDMA communications. IEEE Trans Veh Technol. 2007;56:3006-13. https://doi.org/10.1109/TVT.2007.900383
[7] Nouiri M, Bekrar A, Jemai A, Niar S, Ammari AC. An effective and distributed particle swarm optimization algorithm for flexible job-shop scheduling problem. J Intell Manuf. 2018;29:603-15. https://doi.org/10.1007/s10845-015-1039-3
[8] Ratnaweera A, Halgamuge S, Watson HC. Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients. IEEE Trans Evol Comput. 2004;8(3):240-55. https://doi.org/10.1109/TEVC.2004.826071
[9] Yang C, Gao W, Liu N, Song C. Low-discrepancy sequence initialized particle swarm optimization algorithm with high-order nonlinear time-varying inertia weight. Appl Soft Comput. 2015;29:386-94. https://doi.org/10.1016/j.asoc.2015.01.004
[10] Shi YH, Eberhart R. A modified particle swarm optimizer. In: Proc IEEE Conf Evol Comput. Piscataway: IEEE Press; 1998. p. 69-73.
[11] Yu H, Tan Y, Zeng J, Sun C, Jin Y. Surrogate-assisted hierarchical particle swarm optimization. Inf Sci. 2018;454:59-72. https://doi.org/10.1016/j.ins.2018.04.062
[12] Zhang K, Huang Q, Zhang Y. Enhancing comprehensive learning particle swarm optimization with local optima topology. Inf Sci. 2019;471:1-18. https://doi.org/10.1016/j.ins.2018.08.049
[13] Zhang X, Liu H, Zhang T, Wang Q, Wang Y, Tu L. Terminal crossover and steering-based particle swarm optimization algorithm with disturbance. Appl Soft Comput. 2019;85:105841. https://doi.org/10.1016/j.asoc.2019.105841
[14] Liang JJ, Qin AK, Suganthan PN, Baskar S. Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. IEEE Trans Evol Comput. 2006;10:281-95. https://doi.org/10.1109/TEVC.2005.857610
[15] Zhan ZH, Zhang J, Li Y, Shi YH. Orthogonal learning particle swarm optimization. IEEE Trans Evol Comput. 2011;15:832-47. https://doi.org/10.1109/TEVC.2010.2052054
[16] Zhang X, Wang X, Kang Q, Cheng J. Differential mutation and novel social learning particle swarm optimization algorithm. Inf Sci. 2019;480:109-29. https://doi.org/10.1016/j.ins.2018.12.030
[17] Angeline P. Using selection to improve particle swarm optimization. In: Proc IEEE Int Conf Evol Comput. IEEE World Congress on Computational Intelligence; 1998. p. 84-9.

[18] Aydilek B. A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems. Appl Soft Comput. 2018;66:232-49. https://doi.org/10.1016/j.asoc.2018.02.025
[19] Jindal P, Bedi P. An improved hybrid ant particle optimization (IHAPO) algorithm for reducing travel time in VANETs. Appl Soft Comput. 2018;64:526-35. https://doi.org/10.1016/j.asoc.2017.12.038
[20] Muthusamy N, Patil M. Optimization of high-performance concrete composites by using nano materials. Res Eng Struct Mater. 2023;10(1):579-96. http://dx.doi.org/10.17515/resm2022.602ma1213
[21] Guo L, Allen JK, Mistree F. Optimize or satisfice in engineering design? Res Eng Des. 2024;35:239-67. https://doi.org/10.1007/s00163-023-00431-5
[22] Gouzi A, Ouhimmou S, Daqaq F, Sehaqui R. A new hybrid metaheuristic algorithm based human approach for mechanical engineering problems. In: Proc 10th Int Conf Optimization and Applications (ICOA). Almería, Spain; 2024 Oct 17-18. IEEE. https://doi.org/10.1109/ICOA62581.2024.10753836
[23] Hubálovská M, Major S. A new human-based metaheuristic algorithm for solving optimization problems based on technical and vocational education and training. Biomimetics. 2023;8(6):508. https://doi.org/10.3390/biomimetics8060508
[24] Zhou X, Zhou S, Han Y, Zhu S. Lévy flight-based inverse adaptive comprehensive learning particle swarm optimization. Math Biosci Eng. 2022;19(3):2482-502. https://doi.org/10.3934/mbe.2022246
[25] Han X, Yue L, Dong Y, Xu Q, Xie G, Xu X. Efficient hybrid algorithm based on moth search and fireworks algorithm for solving numerical and constrained engineering optimization problems. J Supercomput. 2009.