International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

AN INTELLIGENT PARTICLE SWARM OPTIMIZATION(PSO) ALGORITHM TO OPTIMIZE EXPLOITATION (LOCAL SEARCH) AND EXPLORATION (GLOBAL SEARCH) TO ELECTIVE THE BEST LEADER.

***Dr*. Burla Srinivas[1], Dr. K Nagaiah[2], Dr. Veluru. Chinnaiah[3], Y. Varalaxmi[4], Dr. Y. Sowmya Reddy[5], Kallem Vijayalalitha[6], Dr. Voore Subrahmanyam[7]**

[1]Associate Professor, Dept of CSE(AI &ML), CMR Institute of Technology, Hyderabad, Telangana,
E-mail: drburlasrinivas@gmail.com

[2]Assistant Professor, ECE Department, Faculty of Engineering & Technology, ICFAI University, Raipur Chhattisgarh, E-mail: nagaiah.k@iuraipur.edu.in , ORCD ID: 0009-0003-9436-5886

[3]Associate Professor, Department of Computer Science and Engineering, Vijaya Engineering College, Khammam, mail-id: vtchinna2k12@gmail.com

[4]Assistant Professor, Department of CSE, CMR Technical Campus, Hyderabad, Telangana, E-mail: varalaxmi.cse@cmrtc.ac.in

[5]Associate Professor, Department of CSE(AI&ML), CVR College of Engineering, Ibrahimpatnam, R.R District. Telangana, E-mail: yalla.soumya.reddy1980@gmail.com, ORCID ID: 0000-0002-8931-8263

[6]Associate Professor, Department of IT, Guru Nanak Institute of Technology, Hyderabad, Telangana, E-mail: kallemvijayalalitha515@gmail.com

[7]Associate Professor, Department of IT, Guru Nanak Institute of Technology, Hyderabad, Telangana, E-mail: subrahmanyam.voore@gmail.com, ORCID: 0000-0002-9860-6355

Abstract

The searching of local search (exploitation) and global search (exploration) to find optimal results to improve optimality for a solution achieve by Swarm Intelligence algorithms are one of the category of Nature Inspired Computing algorithms. These SI are the best model for solving computational problems to find better optimal solution with appropriate results for a given problem. These Swarm Intelligence algorithms are including features are exploration and exploitation. The efficiency of these features influences the swarm intelligence metaheuristics algorithms. In existing research there is no principles and methods for better balancing of these two major features. In our proposed paper proposing balancing between exploration and exploitation of SI. To improve the performance for simulation tested by benchmark functions were performed by various dimensions 100,500 and 1000. n. In this paper SI algorithms are high capability of searching strategy of exploitation (local search) and exploration (global search) for finding optimum results to reach optimality. In this paper balance between exploitation and exploration using SI algorithms. We are considering a PSO (particle swarm optimization) algorithm is a SI algorithm to elective of best leaders among the group using by fitness function to balance exploitation and exploration to find best search strategy for an elective of best leader among the group.

Received: August 12 2025

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Keywords: Swarm Intelligence Algorithms, NIC Algorithms, Particle Swarm Optimization, Exploitation, Exploration.

INTRODUCTION

An efficient searching methods of local well as and global to find optimality to acquire optimum results. Nature inspired computing algorithms are the inspiration from nature for solving complex problems and produce nearest results at reasonable amount of time[1,2]. The Swarm Intelligence Algorithms are one of the type of NIC algorithms. These Swarm Intelligence algorithms are inbuilt features of self discipline, self adaptation, self learning, co-ordination and collective behaviour, are suitable to solve complex computational problems. The nature inspired computing algorithms have feature of random probability i.e., algorithms pick random solutions for better optimum results[3,4,5,6].

Particle swarm optimization(PSO) is popular swarm intelligence algorithm proposed by Dr. Eberhart and Dr. Kennedy in 1995. PSO influence by social behavior of birds and animals[7,8,9,10]. In PSOP is a particle, S is swarm is a group of birds, O is optimization. PSO is a algorithm to solve optimization problems by initialization random population of solutions. Now a days PSO is a model of solving real-time computational problems. One of the best examples of PSO to optimize traffic control problem in the City to avoid traffic accidents, void traffic congestion. Particles effectively fly personnel positions in mean while co-ordinate to neighbor birds flying positions and not hit other particles. SI algorithms provide the best possible to find optimum results.

Particle Swarm Optimization (PSO) is a simple optimization technique inspired by the way birds flock or fish swim in groups to find food. In PSO, many possible solutions, called particles, move around in a search space to find the best solution to a problem. Each particle remembers the best position it has found so far (called personal best or P_{best}) and also knows the best position found by the entire group (called global best or G_{best}). At every step, particles update their position by considering their own best experience, the group's best experience, and a bit of randomness so that they don't all move in the same way. Over time, this process makes the particles move closer to the best solution. PSO is popular because it is easy to understand, does not need complex mathematical derivatives, and works well for many real-world problems where we want to find the most effective or optimal solution.

In PSO, several parameters control how the particles move and search for the best solution. The most important ones are: Number of particles (Swarm size): This is how many particles (possible solutions) are in the swarm. A larger swarm explores more, but takes more time. Number of iterations(Max generations): This is how many times the particles are allowed to move and update before the algorithm stops. Inertia weight (w): This controls how much a particle keeps moving in its current direction. A higher value means the particle explores more widely, while a smaller value means it focuses more on the best solutions found so far. Cognitive coefficient(c1): This tells how strongly a particle follows its own best experience (Pbest). Social coefficient(c2): This tells how strongly a particle follows the swarm's best solution(Gbest).Velocity: The speed and direction of a particle's movement. Velocity is updated at each step based on inertia, cognitive, and social factors. Position: The current solution a particle represents. It changes after velocity updates.

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Kennedy and Eberhart first introduced Particle Swarm Optimization (PSO) as a continuous real-valued algorithm. In PSO, a group of particles moves within a D-dimensional search space to find an optimal solution. Each particle *i* has a velocity vector Vi=[vi1,vi2,…,viD] and a position vector Xi=[xi1,xi2,…,xiD] where D represents the number of dimensions. The PSO process begins by randomly assigning values to both Vi_and Xi. Then, in each iteration, the best position that has been found by particle I Pbesti=[$Pbest_{i1}$,$Pbest_{i2}$,…,$Pbest_{iD}$] and the best position that has been found by the whole swarm Gbest=[$Gbest_1$,$Gbest_2$,…,$Gbest_D$] guide particle i to update its velocity and position by (1) and (2):

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1 \left(Pbest_{id}(t) - x_{id}(t)\right) + c_2 r_2 \left(Gbest_d(t) - x_{id}(t)\right), \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad (2)$$

In above equation,PSO,c1c_1c1 and c2c_2c2 are called the cognitive and social acceleration coefficients. The values r1 and r2 are random numbers that are chosen within the range [0,1]. The pseudo-code of the Standard PSO (PSO) that is used to solve a minimization problem is given in Algorithm 1.

SO is efficient in controlling the balance between exploration and exploitation. Particles in the exploration phase explore the space extensively while the exploitation phase focuses on promising regions. The more balance between exploration and exploitation, the better the PSO performance. n Particle Swarm Optimization (PSO), the swarm needs to balance two important tasks: **exploration** and **exploitation.** Exploration means searching widely in the solution space to find new and better possibilities, while exploitation means focusing around the best solutions already discovered to improve them further. PSO achieves this balance through its velocity update rule. The **cognitive component (c1)** makes each particle follow its own best experience, encouraging individual exploration. The **social component (c2)** makes particles follow the global best solution of the swarm, which promotes exploitation. Random factors r1r_1r1 and r2r_2r2 add variation so particles do not move in the same way, keeping the search diverse. By combining these influences, particles are neither spread out too far (only exploring) nor stuck too quickly (only exploiting). Over time, this balance helps the swarm gradually move toward the optimal solution, allowing the swarm to elect the **best leader (global best particle)** that guides others to the final solution.

Algorithm 1: Particle Swarm Optimization(PSO)

Input

#swarm, #$particle_i$, con1, con2,#dimension,#maxgen

Output

Best solution(leader)

Procedure

Start

Swarm initialization

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

```
   For each particle evaluate fitness
while(iteration≤ #maxgen)
      For every s in swarm
         for every swarm member s
          For updation use equation3 for velocity of member in swarm
                   For updation use equation2 for position of member in swarm
         next
         select local best as leader_s
Next
Among all leaders select the best leader
For updation use equation6 for inertia weight
For updation use equation7 for leader_s velocity
For updation use equation8 for leader_s position
Selection of Best_leader for Global_best
Loop
Return to Best_leader
Stop
```

EXISTING SYSTEM

In[11] proposed exploration particle swarm optimization about a technique of master and slave method to solve local search(exploitation) and global search(exploration). The leader swarm is acting as master and other swarms acting slaves. The slave known as swarm(bird)given a updated positions along the most preferable fitness position values with the master(leader) swarms. New particles update their positions as per master swarm to reach best position. The master swarm(leader) by updating position of their own position and co-ordinate with slave swarms positions to update their new positions. In[12] authors proposed reduce energy conservation model for power system for socio-economic development by applying modified particle swarm optimization. The balance in between for the exploitation and exploration is new method of search by multi-swarm. In the paper used PSO algorithms that represent swarms is a group words the authors in place of 'swarm' mention 'clan' is group of human having leader among them. The PSO optimum results to find local leader among the group of human by P_{best} i.e. personnel best. In the meeting the of selection procedure of group leader, all the leaders of various groups to represent them selves in a selection procedure for best global leader by G_{best}.

In [13] authors proposed the **Partial Leader Optimizer (PLO)** is introduced as a new metaheuristic method based on swarm intelligence. In this approach, several autonomous agents are used, and each one represents a potential solution. The overall intelligence of the swarm is expressed through the best solution, referred to as the leader. PLO has a special way of searching

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

for acceptable solutions in every iteration. In each step, every agent is directed toward a specific target. There are two possible ways to determine this target: the first is by moving the virtual best solution away from the agent, and the second is by randomly selecting a point within the search space. This selection of the target is done randomly but controlled by a threshold that is defined before the iteration begins. After the target is chosen, multiple candidate points are generated between the current position of the agent and the target, with equal spacing between each candidate. The agent then moves to the most suitable candidate and updates the best solution. To evaluate its performance, simulations were carried out where PLO was compared with other algorithms such as the Marine Predator Algorithm (MPA), Particle Swarm Optimization (PSO), Average Subtraction-Based Optimizer (ASBO), Slime Mold Algorithm (SMA), and Pelican Optimization Algorithm (POA). The simulation results show that PLO is competitive when compared with these optimization techniques.

In [14] authors proposed the improvement proposed in this work modifies the velocity update equation of PSO by introducing a **local best murmuration particle,** which is determined through the k-means clustering approach. In this enhanced version, particle movement toward the global best is guided not only by the personal best and the global best but also by the local best murmuration particle. Thus, particles update their positions based on three factors: their own best experience, the swarm's global best, and the identified local best particle.

In [15] authors proposed a hierarchical hybrid particle swarm optimization (HH-PSO) variant that introduces a hierarchical hybrid search framework, where different search strategies are assigned to particles during different evolutionary phases. The proposed framework manages the balance between exploration and exploitation through two main contributions: a hybrid search design that merges global topology with local refinement, and an adaptive fitness-controlled balance factor that dynamically modifies the velocity coefficients of particles. Experiments conducted on the CEC2019 and CEC2022 benchmark functions show that HH-PSO improves solution accuracy compared with five leading PSO variants. The proposed algorithm achieves superior performance rankings in 80% of the comparisons, demonstrating its effectiveness in balancing search intensification and diversification.

Based on the review of the existing literature, I thoroughly studied and analyzed the research contributions of several eminent authors in the field of swarm intelligence and optimization algorithms. These studies provided valuable insights into the behavior and performance of Particle Swarm Optimization (PSO), particularly in balancing exploration and exploitation. Motivated by these findings, I, as the corresponding author, along with my team of co-authors, undertook a research study aimed at improving the PSO algorithm. Our focus was on enhancing its ability to explore the search space effectively while exploiting the best solutions, with the goal of selecting the optimal leadership within a given search space. This work builds upon the concepts and methods proposed in earlier research but introduces new strategies to guide particles more efficiently toward the best solution.

Received: August 12 2025

Table1: Comprehensive analysis of existing research

Authors	Year	Advantages	Methods/Algorithms used	Limitations
Kumar,K. Praveen, et al.[11]	2022	▪ Master and slave options ▪ Local search(exploitation) and global search(exploration).	▪ Particle swarm optimization(PSO)	▪ Not mention extend velocity.
Niu, Ben, et al.[12]	2007	▪ Reduce energy ▪ Conservation model for power system ▪ Exploitation and exploration ▪ Swam and clan	▪ Particle swarm optimization(PSO)	▪ There is no clarity reduce energy for power
Kusuma, Purba Daru[13]	2023	▪ Searching for target solution	▪ Partial Leader Optimizer (PLO)	▪ Target search solution no clarity
Twumasi, Elvis, et al.[14]	2024	▪ Particle update the positions by factors.. ▪ Own best experience, ▪ Swarm's global best, ▪ Identified local best particle.	▪ K-means clustering approach ▪ Particle swarm optimization (PSO)	▪ No clarity application murmuration Exploitation and exploration particle.
Shen, Xiucheng, et al. [15]	2025	▪ Balance between exploration and exploitation ▪ CEC2019 and CEC2022 ▪ benchmark functions	▪ Hierarchical hybrid particle swarm optimization (HH-PSO)	▪ To test efficiency list by only two bench mark function

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

PRELIMINARIES

Particle Swarm Optimization

The Participle Swarm Optimization initialized with a randomly selected selection solutions. The potential solutions of the PSO called as particles. These particles is to move in D dimension solution at a velocity will be adjusted dynamically as per particle's position and neighbor positions. Particle Swarm Optimization initially valued randomly for generate population of particles known as swarm. The random velocity being set for every particle that initiate search space to reach optima by number of iterations. Every particle possess memory for remember for the best position gain by this particle in previous called as personnel best(P_{best}). Every particle possessing P_{best}that attain the these particle with best fitness value is known as global best(G_{best}).

If the search space is D dimensional in that ith particle of population will be considered by values representation of D-dimensional vector as $(x_i^1, x_i^2, \ldots, x_i^D)^T$. Particle velocity will be represented by other D-dimensional vector $(V_i^1, V_i^2, \ldots, V_i^D)^T$.The existed already visited position of ith particle is noted by P_i.The best particle in swarm (group of particles) considered by P_g. The updated value of particle position is valued these equations. Equ(1) used to calculating updated velocity for every particle as per its previous velocity. Equ(2) to update of every position of a particular particle's in a search space. In addition to linearly decrease inertia weight method(LPSO), is a random inertia weight factor for dynamic systems tracking [13]. The Equ(1)n the Inertia_weight(ω)vary randomly.

$$V_{id}^{k+1} = \omega V_{id}^{k} + c_1 r_1 [p_{id}^{k}(t) - x_{id}(t)] + c_2 r_2 [p_{g}^{k}(t) - x_{id}^{k}(t)] \quad (1)$$

$$x_{id}^{k+1}(t+1) = x_{id}^{k}(t) + v_{id}^{k+1}(t+1) \quad (2)$$

In above equations, k denoted as iteration number, d *denoted as* 1,2,3 up to D; i is denoted as 1,2,3 upto$N$; N is denoted as size of swam, w is denoted as inertia_weight is to control the particle's movement by providing weight as per previous velocity, c1 & c2 denoted as acceleration coefficients these are positive constants. r_1 ,r_2 denoted as random numbers between [0,1] uniformly distributed.

In Particle Swarm Optimization(PSO), the particle represents in D-dimensional area as $particle_i(particle_{i\ i1}; particle_{i2};particle_{i3}...;particle_{iD})$; where D represents dimension where $particle_{id}$ ε $[minimum_{dimension};maximum_{dimension}]$, dimension ε [1,dimension], $minimum_{dimension}$, $maximum_{dimension}$ is minimum and maximum with in values of d^{th} dimensions. Velocity of particle i mention $velocity_i=(velocity_{i1}; velocity_{i2,}\ velocity_{i2}\ ; velocity_{iD})$ of high $velocity_{max}$. Particles $everutime_t$ made updations according to relation follows.

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

$$velocity_i(time_t + 1) = velocity_i(time_t) + rvalue_1 con_1 \ (P_{pbp} - article_i(time_t)) - rvalue_2 con_2(Pgbp - particle_i(time_t)) \quad (3)$$

$$particle_i(time_t + 1) = particle_i(time_t) + velocity_i(time_t) \quad (4)$$

As per Equ(4) rvalue1 & rvalue2 both generated randomly for range of 0 to 1. The con_1& con_2 are denoted as velocity constants improve for movement of particle as per iteration value. I^{th} particle the $P_{pbp}$is last best_position. I^{th} particle as the P_{gbp} the best_position is taken to consider.

$$velocity_i(time_t + 1) = inertia_w \ x \ velocity_i(time_t) + rvalue_1 con_1 \ (P_{pbp} - particle_i(time_t)) + rvalue_2 con_2(Pgbp - particle_i(time_t)) \quad (5)$$

In the above equation -3 inertia $_{is}$ added [12]

A perfect balancing between the global and local explorations through inertia, for consideration of minimum iterations to achieve optimal solution.

$$inertia = inertia_{maximum} = \frac{inertia_{maximum-} \ inertia_{minimum}}{inertia_{maximum}} \ x \ iterations \quad (6)$$

Eq(6),$inertia_{max}$is noted as weight(initial), $inertia_{min}$is noted as weights(updated), $iterations_{max}$ noted as maximum iterations. And for every iterations noted as no. of iterations at present.

$$inertia = 0.5 - \frac{rad()}{2} \quad (7)$$

Equ(7), rand() function noted as equally assign random numbers in between 0,1 values. It is should be maintenance a particular 1.494 value by acceleration co-efficients. Equ(7) referred for random weight method (RPSO)

Received: August 12 2025

Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

.

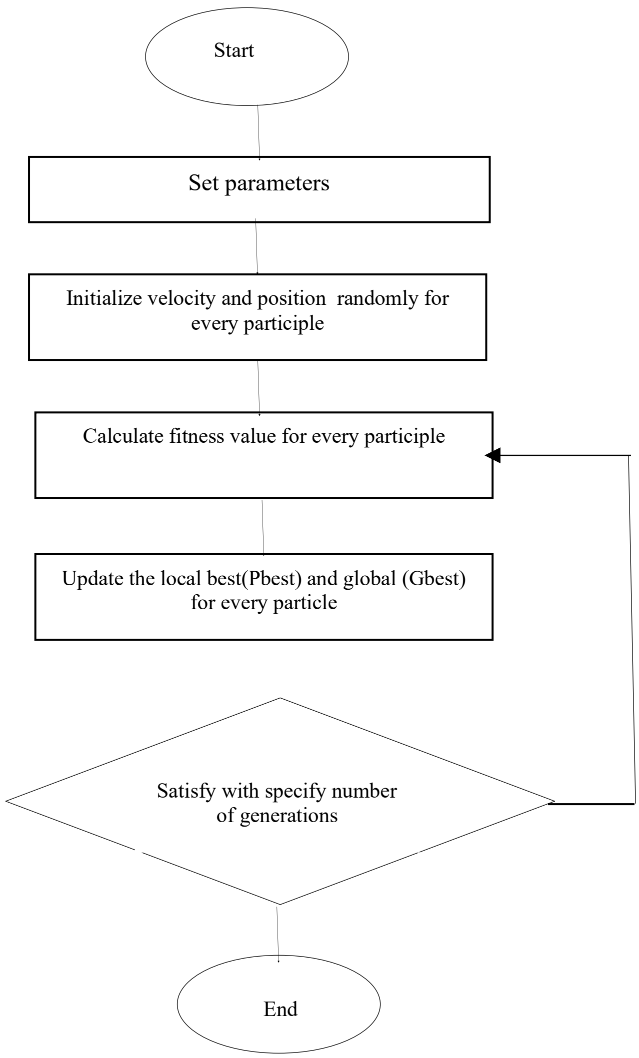

Figure1. Flow chart for Particle Swarm Optimization(PSO)

PROPOSED WORK

3.1 Improved Multiple Particle Swarm Optimization(IMPSO)

As per effects swarm intelligence for searching capability of exploitation and exploration searching methods to find optimality balancing of local and global searching. PSO is a Swarm

Received: August 12 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Intelligence Algorithm is solving real-time computational problems by taking the model of nature inspired computing algorithms. In this proposed paper the leader is elected as a leader in group of people. PSO is an optimization algorithm to elect best leader among the group of people by using an efficient fitness function[15].

In the proposed algorithm objective is implemented for co-operation an effective communication for group of people is known as 'clan' is a group of people. The group leader is searching for solutions. In PSO the group of birds(swarm) taken a model in this proposed paper 'swarm 'is known as 'clan' is a group people and each clan posses solutions that the provided by group members. The best member of a particular group i.e. clan is known as leader of a particular group. A newly elected leader of a groups(clan) meet in a room will selected overall best leader who will update the position of other leaders. This treated as exploitation(local search) of PSO. The proposed algorithm the exploration(global best) is being control by inertia function. The 'clan' is a group of people to manage as well as control over movement of other members in a clan. As per updated positions, each leader meet in a room i.e. location to update to find new global leader. Every round already elected best leader share his position information to other leaders of groups[16,17,18].

Algorithm: The proposed Improved Multiple Particle Swarm Optimization(IMPSO) exploration and exploitation

Step1: Initialize parameters
- Create a swarm of particles with random positions and velocities.
- Set each particle's personal best (pBest) as its starting position.
- Select the best particle among all as the global best (gBest, the leader).

Step2: Update velocity for each particle: New velocity = (inertia × old velocity) of momentum for exploration.
- (c1 × r1 × (pBest – current position)) update and makes particle follow its own best, encouraging **individual exploration**
- (c2 × r2 × (gBest – current position)) update and makes particle follow the leader, promoting **exploitation**.

Step3: Update position to move each particle to its new position using the updated velocity.
Step4: Evaluate fitness to Check how good the new position is using the fitness function.
Step5: Update best position if the new position is better than pBest, update it. If the new position is better than gBest, update the leader.
Step6: Repeat and continue steps 2–5 until the stopping condition is met (like max iterations or reaching required accuracy).
Step7: Result for the gBest (leader) represents the best solution found by the swarm.

As per particle swarm optimization(PSO) algorithm, to find optimal solution. It is including the balancing of positions for global best as well as local best. The local best is chosen a best leader in a particular group 'clans'. Where as global best is chosen a best leader among all groups of 'clan'. As per PSO, OMRM i.e optimization of meeting room method is include the information about 'clans'(group of members) is equal to 'swarm' (group of birds). The proposed algorithm providing P_{best} and G_{best} positions. In the first step, select the local best leader(in member of group i.e. Clan) . In the second step, select the global best leader(from all clan group) by OMRM (optimization meeting room method). The who being selected overall best leader should share own position information to other leaders in all groups[19,20,21].

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Figure 2. Swarm optimization with groups of members(clan) with individual member(leader)

$$iweight^{nl} + \frac{iWeight^{gl} - iWeight^{nl}}{iteration_{maximum}} \, x \, rand() \tag{8}$$

$$Velocity_i^{nl}(\mathrm{t}+1) = iweight^{nl} \; x \; Velocity_i^{nl}(t) + rvaluecon(Particle_{global}^{Leader} - Particle_{local}^{Leader}(\mathrm{t})) \tag{9}$$

$$Position_i^{nl}(\mathrm{t}+1) = Position_i^{nl}(t) + Velocity_i^{nl}(t)$$

In Equ(8,9) nl indicates normal leaders, gl indicates the overall best_leader, $Position_i^{nl}$ stated position of the normal_leaders, $Velocity_i^{nl}$ stated velocity of normal_leaders, $iWeight^{gl}$, $iWeight^{nl}$ denotes inertia weight about all-round best_leader and normal_leader.

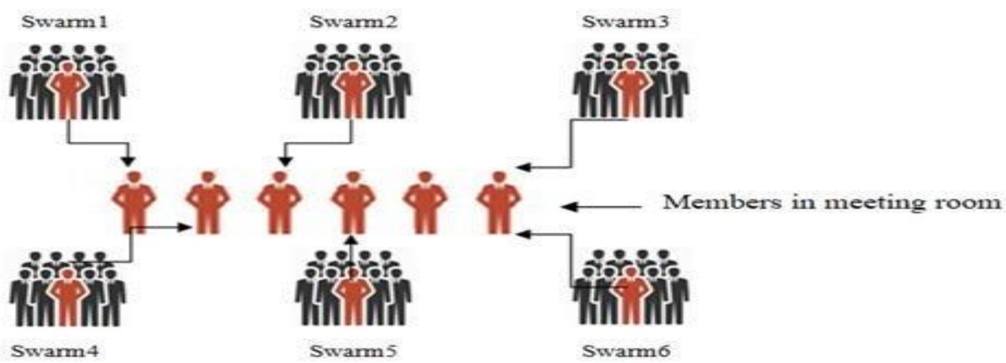

Figure 3. Meeting room approach

A newly leader of a groups(clan) being selected as per every generations as per position updating of members in a group to participate in a particular meeting. The control exploration(global best) by inertia function in proposed[21,22,23].

Received: August 12 2025

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

RESULTS ANALYSIS AND DISCUSSION

Improved Multiple Particle Swarm Optimization(IMPSO) performance is evaluated using benchmark functions compared with MPSO, Master-Slave SPO[11], SPSO[14]. Parameters are used for the similar taken all algorithms. The iterations are consider upto 500, 100 dimensions upto 50. Each experiment taken runs is 30.

Table 1 -Testing with Function

Function_name	Function
Sphereunimodal	$f_1 = \sum_{i=0}^{D} X^2$
Griewankunimodal	$f_2 = \sum_{i=1}^{D} -X^2_{1/4000} - \prod_{I=1}^{d} \cos xj/\sqrt{i} + 1$

Table 2 - Parameter Values

Algorithm	Parameter	Value
SPSO	W(inertia_weight)	0.9 – 0.4
	Number of swarms	1
	Swarm size(population)	50
	c1, c2 (velocity constants)	1.5
MCPSO	W(inertia_weight)	0.9 – 0.4
	Number of swarms	1
	Swarm size(population)	50
	c1, c2 (velocity constants)	1.5
MSPO	W^{ln}	0.8-0.5
	W^{lg}	0.9-0.7
	c1, c2 (velocity constants)	1.5
	Number of clans	5
	Clan size	10
IMPSO	$Inertia^{nl}$	0.8-0.9
	$Inertia^{gl}$	0.9-0.9
	Constant1, Constant2	1.5
	Number of clans	6
	Clan size	10
EIMPSO	$Inertia^{nl}$	0.8-0.9
	$Inertia^{gl}$	0.9-0.9
	Constant1, Constant2	1.5
	Number of clans	6
	Clan size Values	20

Received: August 12 2025

International Journal of Applied Mathematics
Volume 38 No. 5, 2025
ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Algorithm	Results				
MCPSO	0.60278926	1.50234832	2.2067892	2.4056982	2.4859157
SPSO	0.27852127	0.12387125	0.0798215	1.0108791	1.2785781
MPSO	0.10267261	0.01905213	0.0150378	0.0086342	0.0026472
IMPSO	0.039298127	0.03298127	0.0502563	0.0609525	0.0006555
EIMPSO	0.003421288	0.00952327	0.0035962	0.0008931	0.0003212

Table 3 - Benchmark Test Functions Results

In Table-3 including the columns name as Fn, Swarm size, Algorithm taken, Best performance results to considering for proposed algorithm. The Table-3 indicate results of benchmark functions. The benchmark functions are Sphereunimodal, Griewankunimodal are taken to consider . When compare other algorithms IMPSO is showing better performance results[24]. IMPSO include swarm size is 6, clan contains size is 10 members and members in a group. It shows that IMPSO is having the best parameter values when compare with SPSO, MCPSOS, MPSO[25].

The above Table 3 shows the Benchmark Test Functions for estimate the Results for Fitness Value Convergence_curve using sphere function, Convergence curve using Griewank function for algorithms MCPOS, MPSO, SPSO, IMPSO and EIMPSO best results. Benchmark Test Functions for estimate the Results for Fitness Value Convergence_curve using sphere function for (iterations 500)

Table 4 - Convergence_curve using sphere Functions Results (500 iterations)

Fn	Swarm	Algorithm	Best
F1	500	MCPSO	2.4859157
		MPSO	0.0026472
		SPSO	1.2785781
		IMPSO	0.0006555
		EIMPSO	0.0001352
F2	500	MCPSO	0.9045472
		MPSO	0.0041816
		SPSO	48.995751
		EMPSO	0.0000500
		EIMPSO	0.0000375

Received: August 12 2025

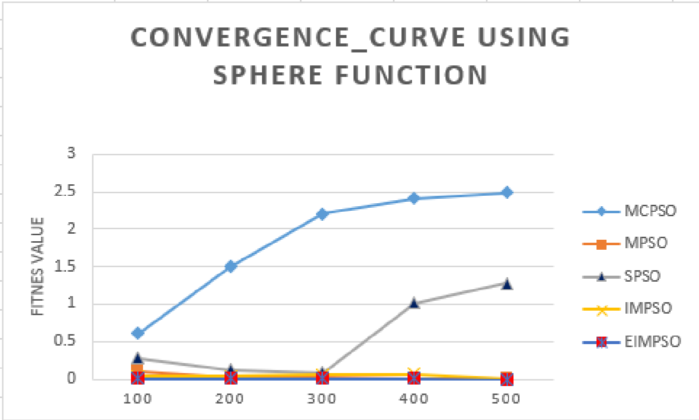

Figure 4(a). Convergence_curve using sphere function

The Figure 4(a) shows simulation results apply Convergence_curve using Sphere function. The X Axis shows Number of Iterations (upto 500), Y Axis shows Fitness Value. The existing algorithms like MCPSO, SPSO, MPSO, IMPSO and proposed algorithm EIMPSO are performing about fitness value. The proposed Enhanced Iimproved Multi-particle Swarm Optimization(EIMPSO) algorithm shows better performance for fitness value when comparing with existing algorithms MCPSOS, MPSO, SPSO and IMPSO[15]. In the Figure 4(a) which one is the minimum value will be consider the best fitness value. Of all algorithms MCPSO, SPSO, MPSO, IMPSO and EIMPSO the proposed algorithm EIMPSO is lowest value showing best fitness value.

Benchmark Test Functions for estimate the Results for Fitness Value Convergence_curve using Griewank function (iterations 500)

Received: August 12 2025

Table 5 - Convergence_curve using Griewank Functions Results (500 iterations)

Algorithm	Results				
MCPSO	0.10179521	0.12030217	0.20079212	0.58033271	0.9045472
SPSO	9.0052678	10.8985232	21.798521	37.567212	48.995751
MPSO	15.125671	12.952121	10.1046671	0.0951212	0.0041816
IMPSO	21.011121	15.044611	11.089511	6.0951752	0.0029712
EIMPSO	18.118911	16.799121	6.912782	0.0975212	0.0010798

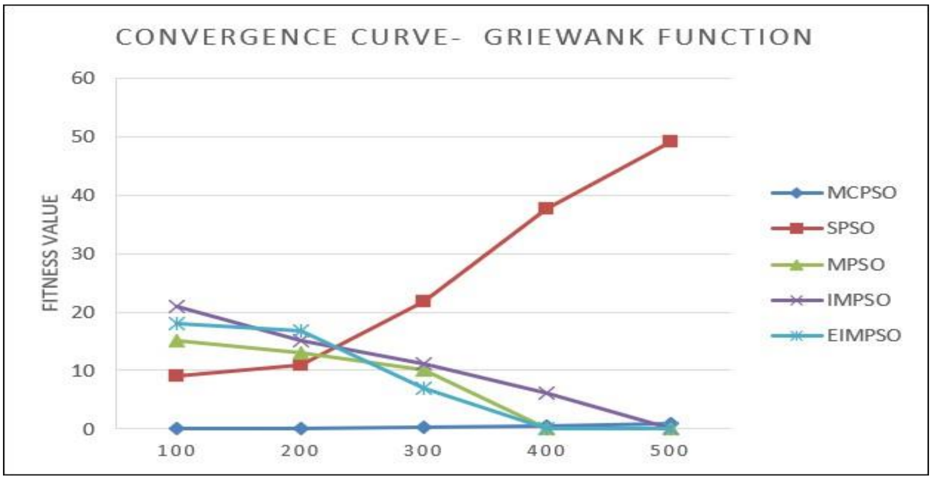

Figure 4(b). Convergence curve- Griewank function

In Figure 4(b) shows the Convergence curve using Griewank function. In this figure X axis shows the number of iterations (upto 500) and Y axis shows the fitness value. The existing algorithms like MCPSOS, SPSO, MPSO, IMPSO and proposed algorithm EIMPSO are performing about fitness value. The proposed Enhanced Iimproved Multi-particle Swarm Optimization(EIMPSO) algorithm shows better performance for fitness value when comparing with existing algorithms MCPSOS, MPSO, SPSO and IMPSO[15,16]. Of all algorithms MCPSO, SPSO, MPSO, IMPSO and EIMPSO the proposed algorithm EIMPSO is lowest value showing best fitness value. Benchmark Test Functions for estimate the Results for Fitness Value Convergence_curve using sphere function for (iterations 1000).

Received: August 12 2025

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

Table 6 - Convergence_curve using sphere Functions Results (1000 iterations)

Algorithm	Results				
MCPSO	1.0971224	1.0647911	2.5179211	3.7891211	4.7923729
SPSO	1.0001112	1.0941245	2.6479111	2.7971121	3.0899761
MPSO	1.0000171	1.0000231	1.0004112	1.0064791	1.0749752
IMPSO	1.0007151	1.0000612	1.0025671	1.0069781	1.0097821
EIMPSO	1.0006127	1.0000011	1.0011712	1.0000812	1.0000791

Table7 - Convergence_curve using Griewank Functions Results (1000 iterations)

Algorithm	Results				
MCPSO	1.9782211	2.6471121	3.4516447	4.6478111	5.2574914
SPSO	17.611447	18.640111	20.4464477	20.000448	21.695847
MPSO	0.0914477	0.0097971	0.7981111	0.8187112	0.9177297
IMPSO	0.0945112	0.1104411	0.18221144	0.2144477	0.5189652
EIMPSO	0.0011471	0.0811201	0.09100221	0.1026448	0.1265789

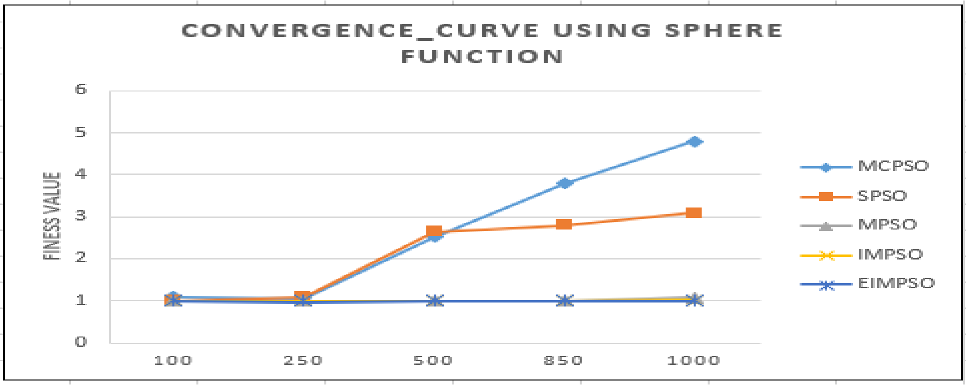

Figure 5(a). Convergence_curve using sphere function
X Axis shows Number of Iterations (upto 1000), Y Axis shows Fitness Value

The Figure 5(a) shows simulation results apply Convergence_curve using Sphere function. The X Axis shows Number of Iterations (upto 1000), Y Axis shows Fitness Value. The existing algorithms like MCPSO, SPSO, MPSO, IMPSO and proposed algorithm EIMPSO are performing about fitness value. The proposed Enhanced Iimproved Multi-particle Swarm Optimization(EIMPSO) algorithm shows better performance for fitness value when comparing with existing algorithms MCPSOS, MPSO, SPSO and IMPSO[17]. Of all algorithms MCPSO, SPSO, MPSO, IMPSO and EIMPSO the proposed algorithm EIMPSO is lowest value showing best fitness value.

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

1. Benchmark Test Functions for estimate the Results for Fitness Value Convergence_curve using Griewank function (iterations 500)

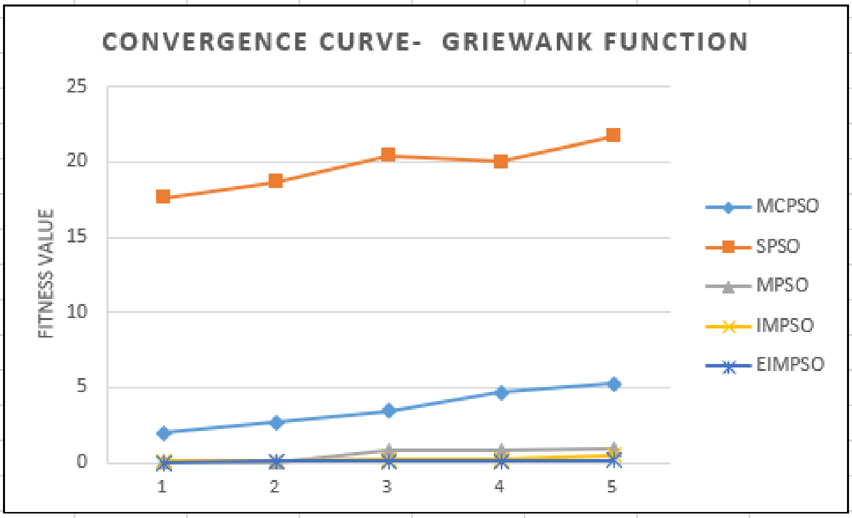

Figure 5(b). Convergence curve- Griewank function

The Figure 5(b) shows simulation results apply Convergence_curve using - Griewank function. The X Axis shows Number of Iterations (upto 1000), Y Axis shows Fitness Value. The existing algorithms like MCPSO, SPSO, MPSO, IMPSO and proposed algorithm EIMPSO are performing about fitness value. The proposed Enhanced Iimproved Multi-particle Swarm Optimization(EIMPSO) algorithm shows better performance for fitness value when comparing with existing algorithms MCPSOS, MPSO, SPSO and IMPSO[18,19]. Of all algorithms MCPSO, SPSO, MPSO, IMPSO and EIMPSO the proposed algorithm EIMPSO is lowest value showing best fitness value.

CONCLUSION

The paper proposed an improved Particle Swarm Optimization is a the social inspired algorithm simulate the group of people and their interactions between the leaders. The proposed algorithm is control the balancing between the exploitation and exploration with solving the large dimensions (100..500...1000) includes various algorithms with testing experiments. In Simulation part introduce with well known bench mark functions using various algorithms. The main objective of proposed paper is for solving real-time problems to find the best optimum results avail for exploitation and exploration search applying by SI algorithm. SI algorithms are related to Nature Inspired Computing algorithms. The main tasks are the human in social group behavior as well as getting solutions by changing of positions and to find local best as well as global best leaders of group members has been solved by proposed swarm intelligence algorithm. EIMPSO solves the exploitation and exploration to balancing of each positions. As per considering of bench mark functions, the results of proposed algorithm EIMPSO shows better simulation results than existing swarm optimization algorithms like SPSO, MCPSOS, MPSO, IMPSO. The future work planning to apply the exploration and exploitation methods with hybrid SI algorithms to get more performance.

Received: August 12 2025

International Journal of Applied Mathematics

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

REFERENCES

[1] Binitha, S., & Sathya, S. S. (2012). A survey of bio inspired optimization algorithms. *International journal of soft computing and engineering*, *2*(2), 137-151.

[2] Fang, J., Liu, W., Chen, L., Lauria, S., Miron, A., & Liu, X. (2023). A survey of algorithms, applications and trends for particle swarm optimization.

[3] Yang, F., Wang, P., Zhang, Y., Zheng, L., & Lu, J. (2017, October). Survey of swarm intelligence optimization algorithms. In *2017 IEEE international conference on unmanned systems (ICUS)* (pp. 544-549). IEEE.

[4] Slowik, A., & Kwasnicka, H. (2017). Nature inspired methods and their industry applications—Swarm intelligence algorithms. *IEEE Transactions on Industrial Informatics*, *14*(3), 1004-1015.

[5] Diao, R., & Shen, Q. (2015). Nature inspired feature selection meta-heuristics. *Artificial Intelligence Review*, *44*, 311-340.

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

[7] Wu, G., Qiu, D., Yu, Y., Pedrycz, W., Ma, M., & Li, H. (2014). Superior solution guided particle swarm optimization combined with local search techniques. *Expert Systems with Applications*, *41*(16), 7536-7548.

[8] Mousa, A. A., El-Shorbagy, M. A., & Abd-El-Wahed, W. F. (2012). Local search based hybrid particle swarm optimization algorithm for multiobjective optimization. *Swarm and Evolutionary Computation*, *3*, 1-14.

[9] 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.

[10] Gad, A. G. (2022). Particle swarm optimization algorithm and its applications: a systematic review. *Archives of computational methods in engineering*, *29*(5), 2531-2561.

[11] Kumar, K. Praveen, et al. "Balancing Exploration and Exploitation in Nature Inspired Computing Algorithm." *International Conference on Intelligent Cyber Physical Systems and Internet of Things*. Cham: Springer International Publishing, 2022.

[12] Niu, Ben, et al. "MCPSO: A multi-swarm cooperative particle swarm optimizer." *Applied Mathematics and computation* 185.2 (2007): 1050-1062.

[13] Kusuma, Purba Daru, and Faisal Candrasyah Hasibuan. "Partial Leader Optimizer." *International Journal on Advanced Science, Engineering & Information Technology* 13.4 (2023).

Received: August 12 2025

Volume 38 No. 5, 2025

ISSN: 1311-1728 (printed version); ISSN: 1314-8060 (on-line version)

[14] Twumasi, Elvis, et al. "A novel improvement of particle swarm optimization using an improved velocity update function based on local best murmuration particle." *Journal of Electrical Systems and Information Technology* 11.1 (2024): 42.

[15] Shen, Xiucheng, et al. "A Hierarchical Hybrid PSO for Single-Objective Numerical Optimization." *2025 6th International Conference on Computer Engineering and Application (ICCEA)*. IEEE, 2025.

[16] Shami, T. M., El-Saleh, A. A., Alswaitti, M., Al-Tashi, Q., Summakieh, M. A., & Mirjalili, S. (2022). Particle swarm optimization: A comprehensive survey. *Ieee Access*, *10*, 10031-10061.

[17] Houssein, E. H., Gad, A. G., Hussain, K., & Suganthan, P. N. (2021). Major advances in particle swarm optimization: theory, analysis, and application. *Swarm and Evolutionary Computation*, *63*, 100868.

[18] Zhang, X., & Lin, Q. (2022). Three-learning strategy particle swarm algorithm for global optimization problems. *Information Sciences*, *593*, 289-313.

[19] Zaman, H. R. R., & Gharehchopogh, F. S. (2022). An improved particle swarm optimization with backtracking search optimization algorithm for solving continuous optimization problems. *Engineering with Computers*, *38*(Suppl 4), 2797-2831.

[20] Abualigah, L., Sheikhan, A., Ikotun, A. M., Zitar, R. A., Alsoud, A. R., Al-Shourbaji, I., ... & Jia, H. (2024). Particle swarm optimization algorithm: review and applications. *Metaheuristic Optimization Algorithms*, 1-14.

[21] Xu, Y., Zhang, M., Yang, M., & Wang, D. (2024). Hybrid quantum particle swarm optimization and variable neighborhood search for flexible job-shop scheduling problem. *Journal of Manufacturing Systems*, *73*, 334-348.

[22] Sato, Y., Yamashita, Y., & Guo, J. (2023, July). PSO with Local Search Using Personal Best Solution for Environments with Small Number of Particles. In *International Conference on Swarm Intelligence* (pp. 123-135). Cham: Springer Nature Switzerland.

[23] Wo, Chengkai. "A Novel Particle Swarm Optimization Based on Multi-Swarm Crossover Hybrid Strategy." *2024 2nd International Conference on Algorithm, Image Processing and Machine Vision (AIPMV)*. IEEE, 2024.

[24] Salih, Sinan Q., et al. "Novel multi-swarm approach for balancing exploration and exploitation in particle swarm optimization." *Recent Trends in Data Science and Soft Computing: Proceedings of the 3rd International Conference of Reliable Information and Communication Technology (IRICT 2018)*. Springer International Publishing, 2019.

[25] Kumar, K. Praveen, and N. Suresh Kumar. "ADVANCED ATTACK MITIGATION IN IOT GATEWAY PROTOCOLS." *Computers & Security* (2025): 104539.

Received: August 12 2025