ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

Particle Swarm Optimization (PSO) and Benchmark Functions: An Extensive Analysis

[1] Sakpere, Wilson, [2] Yisa, Fatai Idowu, [3] Salami Adewale and [4] Olaniyi, Ganiyu Akanji

[1] [2] Department of Computer Science, Lead City University, Ibadan, Oyo State, Nigeria
[3] First bank, Plc., Nigeria
[4] Department of Physics, Crescent University, Abeokuta, Ogun State, Nigeria
Corresponding Author Email: [1] sakpere.wilson@lcu.edu.ng , [2] al.fattah2003@gmail.com, [3] adewale.salami@gmail.com, [4] olaakanji14@gmail.com

Abstract— **Particle Swarm Optimization (PSO) is a popular example of a swarm intelligence technique. PSO has been applied to a variety of fields, including noise, dynamic settings, multi-objective, limited, mini-max, bioinformatics, cloud computing, and medical informatics to mention but a few. Current studies on the PSO algorithm are examined in this paper. Current high-impact studies that have studied and/or modified PSO algorithms have been the focus of the review. The main advantages of the PSO are its ease of use and small number of fine-tuning parameters. The early convergence and lack of a search space balance between exploration and exploitation searches; however, are the main drawbacks of PSO. In this paper, Mathematical operations known as benchmark functions are employed to assess the performance of the algorithm. These functions are complex and possess a range of characteristics. Benchmark functions are used instead of real-world objective functions because they perform reliably better in algorithm testing. Few benchmark functions, spanning from 1 to 50, are listed in most recent literature. Its most important characteristic is the fundamental classification of benchmark functions into unimodal and multimodal functions, each with unique features. In this study, population sizes of 20, 50, 70, 100, and 120 were used, along with two different categories of benchmark functions. A visual depiction of the findings was given.**

***Index Terms*— swarm intelligence, benchmark functions, and particle swarm optimization.**

I. INTRODUCTION

Particle swarm optimization (PSO) based on swarm interactions is a theory-based optimization method that does not require supervision or prior knowledge (Gheitanchi, Ali, and Stipidis 2010). Kennedy and Eberhart (1995) introduced a stochastic population-based optimization technique called PSO. PSO has been around since 1995. The original PSO has been modified and improved. Many evaluations have been made over the past 30 years. To provide up-to-date information, it is necessary to review statistics from the last 10 years. PSO is based on studies of synchronized schools of fish, buzzing bees, and flocks of birds. It forms groups or groups of people called particles. To determine the best global solution, PSO changes the movement of each node in each iteration based on the global consensus (correlation) and individual (known) positions of all products in the entire population. PSO is a widely used method in society as an optimization technique for solving multidimensional and multi-objective problems. According to Priya and Kamlu (2023), groups of animals cooperate with each other to increase speed, and in many cases, they can escape from animals or achieve realistic goals.

According to Jordekhi and Jasni (2013), PSO has several properties that contribute to its efficiency in solving optimization problems:

a. The user needs to configure fewer parameters than other heuristics.
b. It is also very easy to code.
c. Fast convergence is one of its advantages.
d. The computational load is lower than most other heuristics.
e. It has high accuracy.
f. The early solved problems have less impact on the computational behavior of PSO than other heuristics. The behavior is not significantly affected by the increase in dimension.
g. Discrete/integer variables, constraints, multi-objective problems, and multi-modality are efficiently solved.

According to the creators of the PSO algorithm (Mirjalili & Dong, 2020), the intelligence of a flock of birds can be reduced to the following four rules:

a. Each member of the flock has the ability to remember the best solution they have found so far.
b. They all tend to find the best solution they have found so far.
c. They can all see the best solution found by the entire flock at a given time.
d. And they all gravitate toward the best solution found by the flock.

A. Swarm Intelligence

Optimization algorithms are mostly inspired by nature and are usually based on swarm intelligence. Swarm intelligence is a field of artificial intelligence (AI) that deals with collective behavior in distributed and self-organizing systems (Mohammed et. al., 2019). Swarm Intelligence is a subfield of artificial intelligence that studies the emergent properties

IFERP Explore Your Research Journey...

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

and collective behavior of complex, self-organizing, distributed systems with social structures. These systems consist of simple interacting agents organized into small communities (swarms). Although each agent has a very small space to operate and no central control, the collective behavior of the entire swarm exhibits characteristics of intelligence, such as decision-making ability.

a. Organization; Integration is performed based solely on local information, without reference to a global model.
   b. Multiple interactions, positive and negative feedback;
b. It is a task performed simultaneously by specialized personnel in the division of labor.

Wang, Tang, and Liu (2017) studied the behavior of social animals using artificial life theory to build a swarm artificial life system that uses computers to perform cooperative behavior, and proposed the following five basic principles, namely:

1) Proximity: Basic temporal and spatial calculations should be performed by the swarm.
2) Swarm intelligence: The swarm should be able to detect and respond to environmental changes.
3) Multiple answers: The swarm should not limit resource acquisition to a specific area.
4) Stability: The swarm should not change its behavioral pattern to respond to environmental changes.
5) Adaptability: When change is needed, we must change our behavior patterns.

Organization of the paper: The following is how the paper is structured: related works are covered in Section 2, PSO fundamentals are covered in Section 3, and experiments are covered in Section 4.

II. RELATED WORKS

Swarm intelligence-based algorithms, physics-based and chemistry-based algorithms, and evolutionary algorithms (EA) are three major subfields of metaheuristic algorithms. Evolutionary algorithms (EA), inspired by natural selection and natural genetics, are always well known among professional researchers. Some examples of evolutionary algorithms (EA) include differential evolution algorithms, virulence optimization algorithms, genetic algorithms, biogeography-based algorithms, and differential search algorithms. For example, cultural algorithms. Physics-based and chemistry-based algorithms are algorithms that mimic specific physical and/or chemical rules. In other words, they are inspired by these areas. Some of the famous algorithms in physics and chemistry include Big Bang-Big Crunch, spiral optimization, electromagnetism optimization, black hole, charge system search, and galaxy-based search algorithms. One important area of computational intelligence is swarm intelligence (SI). It is based on the study of how swarms behave collectively in the wild and interact locally without any form of supervision. Swarm intelligence is another name for swarm intelligence. Ant colonies, fish swarms, bacterial growth, animal herding behavior, bird flocks, and other phenomena are examples of SI. Numerous swarm intelligence algorithms have been the subject of much research to date to solve various optimization problems. The technical classification of selected content, including hybridization, enhancement, and PSO variants, and the practical application of algorithms are analyzed in the article by Gad (2020) on existing studies on methodologies and applications published between 2017 and 2019. . This study focuses on the latest developments in PSO from 2020 to the present. The analysis of 30 benchmark functions by Pleuris and Solorzano (2022) is the most recent set of benchmark functions.

Table I: Some of the popular bio-inspired meta heuristic algorithm inspired by swarm intelligence (Biswas, Kalayci, & Mirjalili, 2023).

Year	Algorithm proposed	Inspiration / method
2022	Artificial hummingbird optimisation algorithm	Zhao et al. developed the artificial hummingbird algorithm (AHA) to solve optimization problems, and experimental results showed that it outperformed other metaheuristics.
2022	Chimp Optimization Algorithm	Jia et al. presented an enhanced chimp optimization algorithm (EChOA) and evaluated its performance on 12 conventional benchmark functions and 15 CEC2017 benchmark functions.
2021	Rat swarm optimization algorithm	Swarm-based rat swarm optimization is presented by Dhiman et al., who also evaluate its performance using CEC-15 special session benchmark functions, unimodal, and multimodal functions.
2021	African Vulture's Optimization Algorithm	Abdollahzadeh and colleagues present the African Vulture's Optimization Algorithm (AVOA), a new metaheuristic. They proved it to be the best method using 30 out of 36 benchmark functions.
2021	Dragonfly optimization algorithm	Bhardwaj and Kim presented the dragonfly node identification algorithm (DNIA) and evaluated its robustness and efficiency using statistical

IFERP
ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 12, Issue 01, January 2025

Year	Algorithm proposed	Inspiration / method
		analysis, convergence rate analysis, Wilcoxon test, Friedman rank test, and analysis of variance on both traditional and modern IEEE CEC 2014 benchmark functions.
2021	Horse herd optimization algorithm	The Horse herd Optimization Algorithm (HOA) is a new meta-heuristic algorithm that MiarNaeimi et al. created to address high dimensional optimization strategies.Using statistical findings, they illustrated the advantages of their suggested algorithm.
2021	Flamingo Search optimisation Algorithm	It is based on flamingo migration and foraging behaviors.
2021	Horse herd Optimization Algorithm	Six key characteristics—grazing, hierarchy, sociability, imitation, defensive mechanism, and roaming—are used to implement what horses perform at different ages.
2020	Gaining-sharing knowledge-based optimisation algorithm	Using CEC 2017 benchmark functions and trials on a variety of challenges, Mohamed et al. demonstrated the superiority of their suggested gaining-sharing knowledge-based approach.
2020	Coronavirus optimization algorithm	A unique bio-inspired metaheuristic based on coronavirus behavior was presented by Martinez-Alvarez et al. They described the main benefits of the coronavirus optimization algorithm over other comparable tactics.
2020	Chimp Optimization Algorithm	It draws inspiration from chimps' sexual behaviors and individual intelligence when they discover a group to be in
2020	Black Widow Optimization Algorithm	It is primarily based on Black widow spiders' intriguing sibling cannibalism behavior.
2020	Sparrow Search optimisation Algorithm	The clever methods that sparrows employ to find food based on their circumstances serve as the foundation for this algorithm.
2020	Rat Swarm Optimisation algorithm	It draws inspiration from rats' propensity for chasing and attacking.
2019	The Sailfish Optimisation algorithm	Based on a population of hunting sailfish, this algorithm employs the sailfish population for searching and the sardine population to diversify the search space.
2018	Meerkat Clan optimisation Algorithm	It is modeled after Meerkats, which are known for their extraordinary intelligence, strategic planning abilities, and impressive directional cunning when foraging in the desert.
2018	Grasshopper Optimization Algorithm	It is inspired by the feeding and swarming behaviors of grasshoppers.
2017	Salp Swarm optimisation Algorithm	It draws inspiration from salps' swarming behavior in the ocean.
2017	Camel Herds optimisation Algorithm	This algorithm is modeled after camels, which have a leader for each herd and depend on the humidity of their surroundings to find food and water.
2017	Duck Pack optimisation Algorithm	It is based on how ducks forage based on their orientation toward food and imprinting behavior.
2016	Dragonfly optimisation Algorithm	It is predicated on how dragon flies behave both statically and dynamically.
2016	Sperm Whale optimisation Algorithm	It is inspired by the way of life of sperm whales.
2016	Dolphin Swarm optimisation algorithm	It is based on the biological traits and lifestyle of dolphins, including their ability to echolocate, communicate, collaborate, and divide work.
2016	Crow Search optimisation Algorithm	It is predicated on the way crows forage for food, conceal it from other crows, and retain their hiding spots.
2015	Ant Lion Optimisation algorithm	This algorithm simulates how ant-lions hunt in the wild.
2015	Elephant Herding Optimization algorithm	Elephants' herding behavior serves as its basis, with many groups of elephants living under a matriarch.

IFERP
ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 12, Issue 01, January 2025

Year	Algorithm proposed	Inspiration / method
2015	Moth-flame Optimization algorithm	It is based on transverse orientation, which is how moths navigate in the wild.
2014	Grey Wolf Optimisation algorithm	It imitates the natural hunting and dwelling hierarchy of grey wolves.
2014	Pigeon Optimization algorithm	It is modeled after how passenger pigeons swarm.
2014	Spider Monkey Optimization Algorithm	The Fission-Fusion social structure of spider monkeys during foraging serves as its model.
2013	Spider Optimization algorithm	It is predicated on social spiders' cooperative traits.
2012	Bacterial Colony Optimization algorithm	The life cycle of a bacterium called E. Coli serves as its basis.
2012	Zombie Survival Optimization algorithm	It is based on how zombies forage and discover a fictitious airborne remedy that heals their illnesses.
2010	Bat optimisation Algorithm	It is based on the echolocation of natural microbats, which have different loudness and emission pulse rates.
2010	Termite Colony Optimization algorithm	Its foundation is termites' clever behavior.
2010	Fireworks optimisation Algorithm	Two different kinds of explosion processes are carried out using this technology, and their diversity is maintained by using fireworks.
2009	Cuckoo Search algorithm	It draws inspiration from the way cuckoos lay their eggs in other species' nests.
2009	Gravitational Search Algorithm	Its foundation for finding answers is found in Newton's laws and the laws of gravity.
2009	Glowworm Swarm Optimization algorithm	It mimics the actions of glow worms or lighting worms.
2008	Fast Bacterial Swarming optimisation Algorithm	This algorithm combines the flocking mechanism of birds from Particle Swarm Algorithm with the foraging behavior of E. Coli from Bacteria Colony Algorithm.
2007	Firefly optimisation Algorithm	It draws inspiration from the natural fireflies.
2006	Cat Swarm Optimization algorithm	It consists of two subprocesses, seeking mode and tracing mode, and is modeled by the behavior of cats.
2005	Artificial Bee Colony algorithm	The employed, workers, and scouts in honey bee colonies are all simulated by this method.
2004	Honey Bee optimisation Algorithm	It is modeled after the natural foraging methods used by honey bees.
1995	Particle Swarm Optimization algorithm	The way flocks of birds travel throughout the planet and look for food, both individually and collectively, is the basis for this theory.

A. Google scholar

Google scholar search “particle swarm optimization” information retrieved on 14th May, 2024.

Table II: No of materials related to PSO on google scholar

Year/criterion	All materials (include citations only)	All materials (include Patients and citations	Review articles only (include patients and citations)
Since 2024	18,000	16,700	1,880
Since 2023	29,100	31,600	6,880
Since 2022	65,600	52,200	11,000
Since 2021	85,500	105,000	14,100
Since 2020	133,000	168,000	16,000

IFERP
Explore Your Research Journey...
ISSN (Online) 2394-2320
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)
Vol 12, Issue 01, January 2025

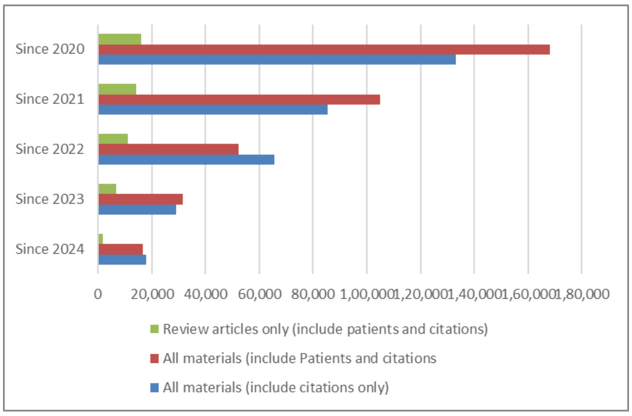


Fig. 1. Barchart representations of number of materials related to PSO on google scholar on Table 2

Table III: No of related material to PSO on google scholar

S/N	Year/criterion	All materials (include citations only)	All materials (include Patients and citations	Review articles only (include patients and citations)
1	2024(as at May)	18,000	16,700	1,880
2	2023	11,000	14,900	5,000
3	2022	36,500	20,600	4,120
4	2021	19,900	52,800	3,100
5	2000	47,500	63,000	1,900

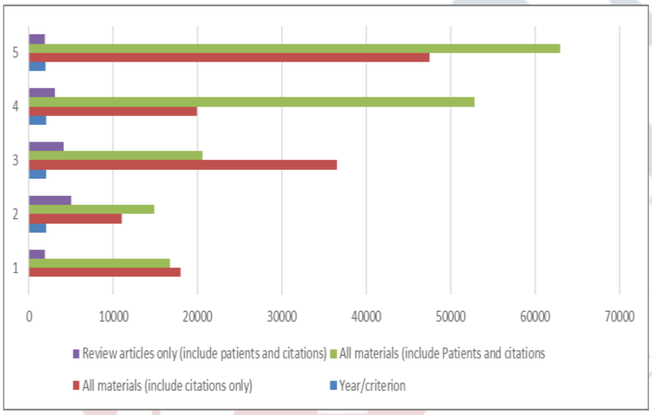


Fig. 2. Barchart representation of number of related material to PSO on google scholar on Table 3.

Also, searching for "particle swarm optimization" in mendeley website in may 2024 at www.mendeley.com give the following results. The related materials here include the following types of documents: journals, conference proceedings, book sections, genetics, thesis, report, web page, patents and working papers.

Table IV: No of related material on mendeley website. Source: www.mendely.com

Year	No of related materials
2024	2,733
2023	11,590
2022	10,946
2021	12,595
2020	9,121

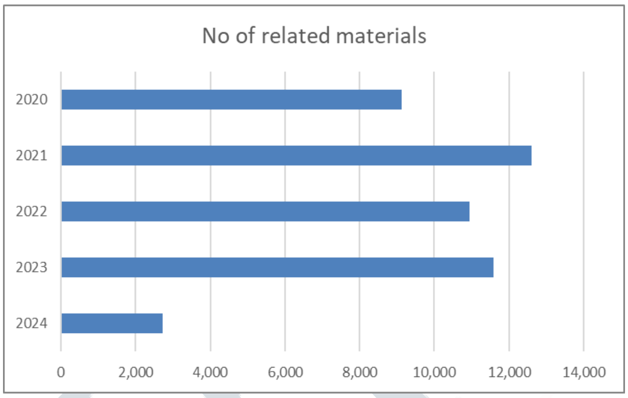


Fig. 3. Barchart representation of number of related material on mendeley website. Source: www.mendely.com

III. FUNDAMENTALS OF PSO

According to Kennedy and Eberhart (1995), PSO is a well-known technique for resolving single objective optimization problems. The swarm of particles explores the search region and surrounds the best particles, simulating the collective behavior of birds. In a particle swarm, each individual has location and velocity characteristics but no mass or volume. Fitness is the term used to describe the unique value assigned to each particle's location. The global optimum particle is the individual that occupies the best location among the particle population; the other particles are referred to as ordinary particles (Guo et al., 2023)

In recent years, PSO has garnered a lot of attention due to its unique performance. When it comes to handling complex optimization issues, it has the same drawbacks as other intelligent algorithms: it is prone to premature convergence and slowness in later rounds. By utilizing different adjustments to address the aforementioned issues, several researchers attempted to further enhance its performance and introduced numerous superior variations that outperform the original PSO2. The simplicity of use and limited number of fine-tuning parameters of the PSO are its primary benefits.

" Nonetheless, the primary disadvantages of an established PSO are its early convergence and the absence of equilibrium between exploration and exploitation searches inside the search area (Khan et al., 2021). PSO's exploration and exploitation behaviors Exploration is the process of looking for completely new areas inside a search space, whereas exploitation is the process of looking for areas that are close to the areas that have previously been visited and searched. Intelligent swarms Optimisation difficulties are resolved by algorithms like the PSO and meta-heuristics in general, which regulate the agents' degree of exploration vs exploitation. Exploration is the process by which population

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

agents look for previously untested regions of the fitness landscape. Exploitation is the process of searching regions of the search space that may contain possible optima in order to improve upon already-existing solutions (van Zyl & Andries, 2023).

Because unimodal benchmarks only have one global optimum, they are useful for assessing the potential for exploitation (Qian, Khishe, Huang, & Mirjalili, 2023). Individual dimensions can be used to determine if a swarm is diverging or congregating in a confined area. The difference between the dimension d values among swarm individuals increases as the algorithm is diverging, indicating that the search environment is dispersed with swarm members. In the context of metaheuristic study, this is known as exploration or diversification. However, as the swarm is converging, the disparity is reduced and the members of the swarm congregate in a smaller region.

We refer to this as intensification or exploitation. Different metaheuristic algorithms use various tactics to impose intensification and diversity among the swarm members during iterations (Hussain, Salleh, Cheng & Shi, 2018). There are generally two ways to manage exploration and exploitation in a metaheuristic algorithm. These make use of solution categorization and parametric values. When the update favors one of the two search behaviors depending on the parametric value, this is known as parametric value-based adjustment (Tilahun, 2019).

Because PSO only needs basic mathematical operators, less computational bookkeeping, and typically fewer lines of code than ant colony algorithms and Evolution Algorithms (EAs), it is computationally cheap in terms of speed and memory use. PSO is widely used because it is easy to implement and may rapidly converge on a solution that is deemed acceptable (Du & Swamy, 2016).

The figure 4 below illustrate the activities of particles in PSO

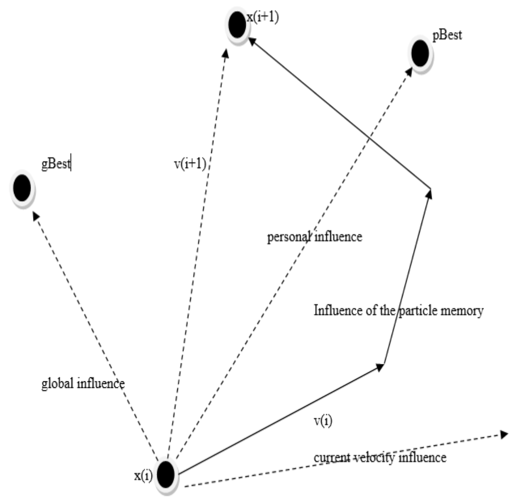


Fig. 4. Geometry representation of PSO

Updated position:

$X_i(t+1) = X_i(t) + V_i(t+1)$

Updated velocity:

$V_i(t+1) = wV_i(t) + c_1\beta_1(P_i(t) - X_i(t)) + c_2\beta_2(G(t)-X_1(t))$

Interpretation:

V_i=velocity of the ith particle

w= inertial weight of the particles

c_1 and c_2 = acceleration coefficients

β_1 and β_2 =random numbers

$wV_1(t)$=inertial term /momentum part

The weighting function w will be obtained using the following equation:

$w = w_{max} - (w_{max} - w_{min}) \times t/t_{max}$

where w_{max} and w_{min} are the maximum and minimum values of w.

$c_1\beta_1(P_i(t) - X_i(t))$ =cognitive component

$c_2r_2(G(t)-X_1(t))$ =social component

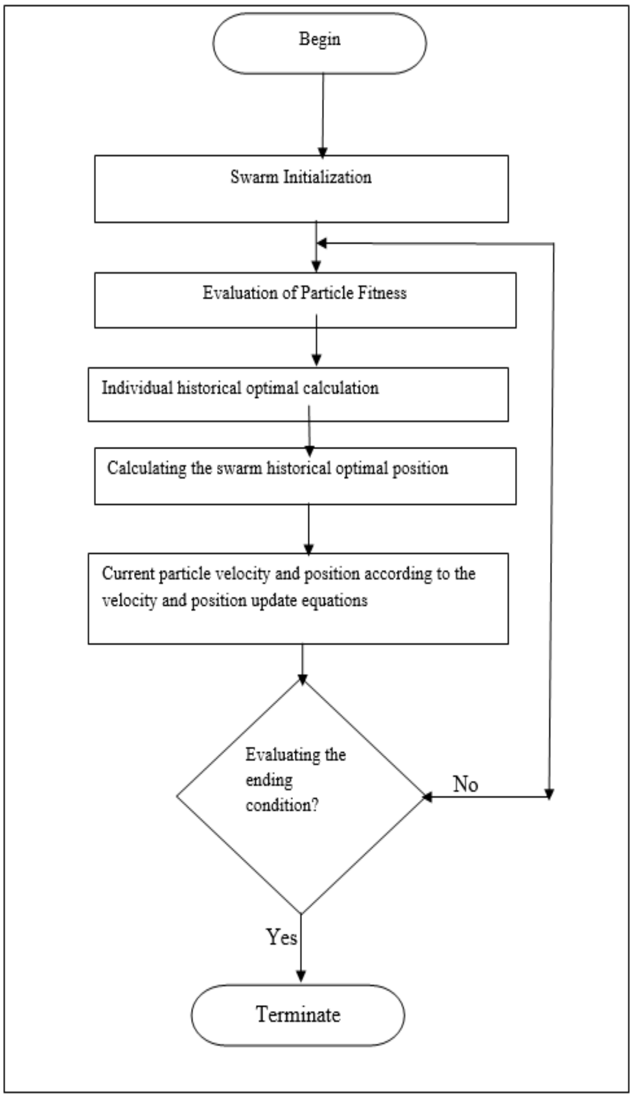


Fig. 5. Flowchart of the PSO algorithm

The algorithm for PSO is presented below:

Input: = $\{VM1, VM2, \ldots..\}$, $PM = \{pm1, pm2, \ldots..\}$, $Task = \{T1, T2, \ldots.., Tn\}$ Findings include the best possible task placement on the virtual machines (Gbest), the best possible VM placement on the PMs (Gbest), and the best possible average use on the VMs (Gbest).

To start:

1. Assign particle dimensions according to placement, average utilization, and the size of open positions.
2. Set the particle's position and velocity at random.
3. Apply a load balancing and placement algorithm to each particle to determine its position.
4. Determine the alpha value and fitness value for every particle. Set the current fitness value as the new pbest if (fitness value).
5. After Step 4, select the best particle out of all the particles.
6. Update each particle's position and calculate its velocity.
7. If the halting criteria or maximum iteration are not fulfilled, repeat Step 4. otherwise Return to gbest Stop now.

Fig. 6. The pseudo-code of particle swarm optimization algorithm

A. Benchmark Test Functions

Benchmark test functions are mathematical expressions of optimization problems in the form of numerical functions. These functions are optimized with a set of ideal parameter values to obtain the best answer, where D stands for the issue dimensions. With hills and valleys of all shapes and sizes, the challenging terrain obscures the best option among other less-than-ideal options. Metaheuristic algorithms, like other optimization algorithms, like to find the best answer as quickly as possible, albeit this isn't always the case.

Any metaheuristic's ability to search worldwide and converge locally determines how successful it is. Better global searchability algorithms are more challenging to trap in subpar (local minima or maxima) sites. It is difficult to overlook any optimal solution in the nearby regions while using metaheuristics with an effective convergence ability (Hussain, Salleh, Cheng & Naseem, 2017; Plevris & Solorzano, 2022).

Features of benchmarking functions According to Hussain, Salleh, Cheng, and Naseem (2017) and Plevris and Solorzano (2022), there are several classifications of benchmark functions in the literature. The primary classifications are as follows:

a. Separatability

Separability describes how to optimize a function's variables. Both separable and non-separable functions can be unimodal or multimodal. Functions that are separable allow for the independent optimization of each variable xi. This category of functions has simple solutions. Non-separable functions: These are those in which all of the variables have a close relationship with one another and cannot be optimized separately. Solving such functions is not so easy.

b. Dimensionality

This attribute defines the search space. As the intricacy of the environment increases, there are more poor locations. Most metaheuristic algorithms often perform well on small-dimensional functions, which are easy to solve. To give correct performance ratings, however, functions need to be highly dimensional.

c. Valleys

A valley is a long, thin area that is encircled by peaks. In essence, it is a narrowly moving region. Because local search wants to scan these areas carefully, the metaheuristic algorithms take longer. These troughs have different frequency and shapes due to unimodal and multimodal processes.

The basin an area of steep hills encircling a precipitous drop is called a basin. There are more basins of local minima than global minima in the multimodal functions. Inadequate algorithms frequently identify issues in local minima basins and fail to identify global minima basins.

d. Modality

The issue landscape's number of peaks is defined by the modality. The locations of local and global minima are formed by these peaks. The optimal solution for unimodal functions can only be located in one valley and one global minimum point. Even if moving and rotating these functions makes them more challenging to solve, they are still considered simple functions. These functions can be used to examine metaheuristic approaches for local searchability. Functions in the multimodal category preserve multiple solutions, even though there is only one final global best. There is only one real global minimum for these functions, however there are numerous local minima sites. To identify the genuine global optimal solution, any metaheuristic algorithm must search the entire landscape. These functions can be used to assess the global searchability of an algorithm because they are challenging to solve.

There are numerous sets of benchmark functions for researchers in the literature. Jamil & Yang (2013) have the largest collection of benchmark functions for global optimization problems, with 175 functions. Nonetheless, certain functions have similar characteristics. To test the performance of an algorithm, not all functions must be used.

For the purpose of this study six functions are selected, three unimodal and three multimodal functions.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

1. Sphere function

Sphere function also known as De Jong's function. Common properties are continuous, differentiable, separable, scalable, unimodal. The graphical illustration is given below:

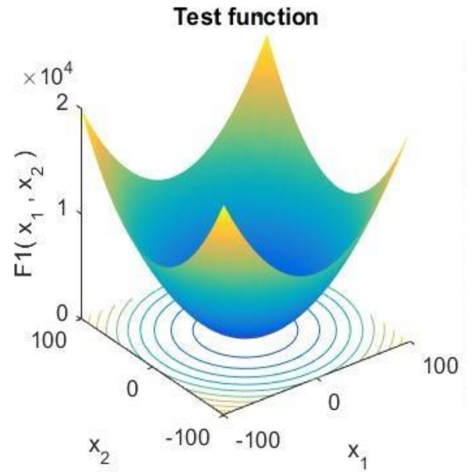


Fig. 7. Graphical representation of sphere function

The expression for sphere function is:

$$f(x) = \sum_{i=1}^{D} x_i^2$$

Subject to $-100 \le x_i \le 100$. The global minimum is located at $x^* = f(0,0,0,0\ldots\ldots\ldots,\ )$, $f(x)=0$

2. Rosembrock

The properties of Rosembrock function are continuous, differentiable, non-separable, scalable, Unimodal. it is also known as banana function. The graphical illustration of Rosembrock is below:

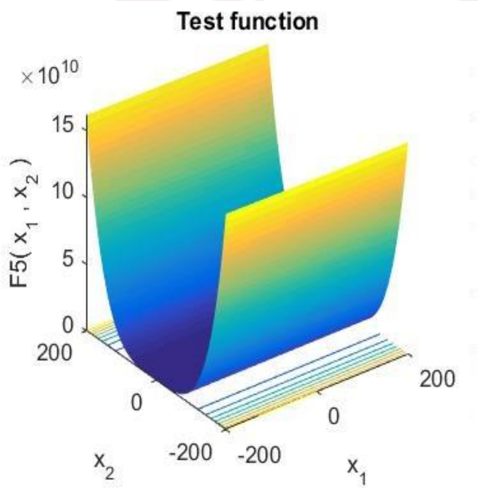


Fig. 8. Graphical representation of Rosembrock function

The Rosembock function can expressed below:

$$f(x) = \sum_{i=1}^{D-1} [100((x_{i+1} - x_i^2)^2) + (x_i - 1)^2]$$

Subject to $-30 \le x_i \le 30$.

The global minima is located at $x^* = f(1,1,\ \ldots\ldots\ldots1)$, $f(x^*) = 0$

3. Quartic function

Quartic function has the following properties: continuous, differentiable, separable, scalable and unimodal), the graphical illustration of quartic function is below:

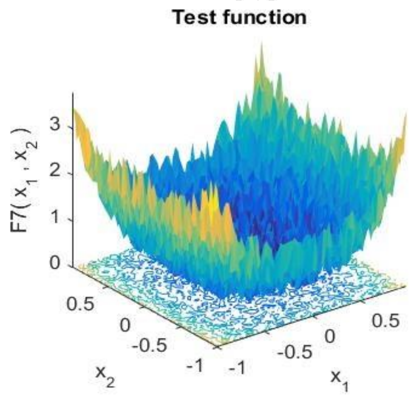


Fig. 9. Graphical representation of Quartic function

$$f(x) = \sum_{i=1}^{D} ix_i^4 + \text{random } [0,1]$$

Subject to $-1.28 \le x_i \le 1.28$

The global minima is located at $x^* = f(0,0,\ \ldots\ldots\ldots0)$, $f(x^*) = 0$

4. Schwefel 2.26

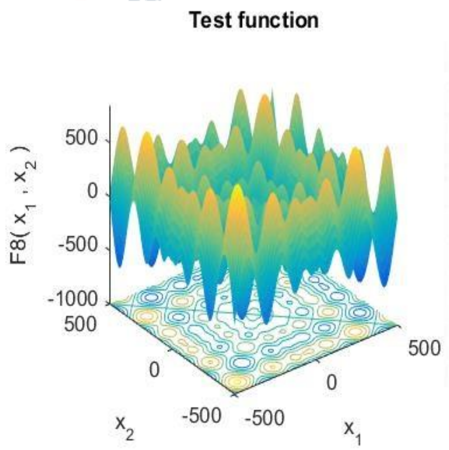


Fig. 10. graphical representation of Schwefel 2.26 function

5. Rastrigin's function

It is challenging to solve this multimodal function because it offers a large number of local minima sites where an optimization algorithm with limited exploratory capabilities is likely to become stuck. The domain of [-5.12,5.12] contains the function's lone globally optimal solution, 0: $f(x^*) = [0,0,\ldots,0]$. The following is a graphical representation:

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

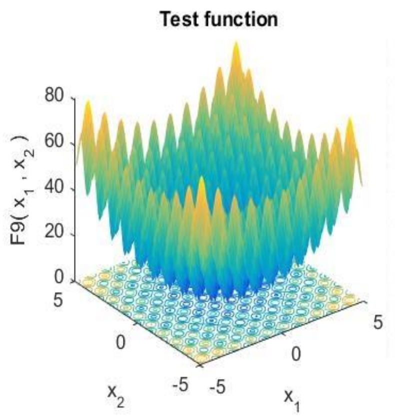


Fig. 11. Graphical representation of Rastrigin's function

The function is mathematically written as:

$$f(x) = \sum_{i=1}^{D}\left(x_i^2 - 10cos(2\pi x_i)\right) + 10D$$

subject to -5.12 $\leq x_i \leq$ 5.12

6. Ackley Function

Continuous, differentiable, non-separable, scalable, and multimodal are Ackley's characteristics. One of the most popular test functions for evaluating metaheuristic algorithms is this multimodal function. There are many local minima, but the deep, narrow basin in the center has the only global optimal solution. In domain [-32,32], the optimal solution 0 can be found at f(x*) = [0,0,…,0]. The following is a graphical representation:

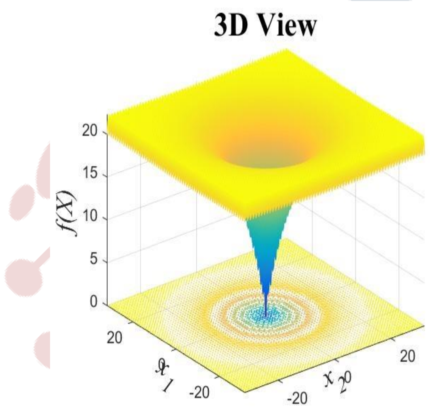


Fig. 12. Graphical representation of Ackley function

The function is mathematically written as:

$$f(x) = -20exp\left(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} x_i^2}\right) - \exp\left(\frac{1}{D}\sum_{i=1}^{D} cos(2\pi x_i)\right) + e + 20$$

IV. EXPERIMENTS

A. Experimental settings

The PSO`s overall performance may be superior with the aid of using adaptively balancing the swarm's exploration and exploitation capacities through the usage of range of parameter that assesses the dispersion in a swarm (Niu, Zhu, He, & Wu, 2007).

When range is low, the swarm can inspect greater specific solutions; while range is excessive, it may inspect a huge vicinity to save you untimely convergence. Accordingly, it's far encouraged to hold the swarm's range excessive withinside the early tiers of evolution so as to research a huge vicinity wherein the exceptional solution may exist, and to decrease it toward the quit of the evolution so one can refine the final results locally (Zhou & Wei, 2021).

B. Adjusting Inertial weight

The trade-off between PSO's exploration and exploitation skills is adjusted by inertia weight. PSO's exploring functionality may increase when the inertia weight decreases and vice versa. In order to focus the quest attempt specifically on exploration at the beginning ranges and on exploitation at the later ranges of the run, it is typically drastically lowered linearly at some point along the run's course (Jordehi & Jasni, 2013). The inertial weight is set at a minimum of 0.4 and a maximum of 1.2.

C. Particle Size

A swarm's population size is determined by the quantity of particles it contains. This parameter is essential for describing the convergence performance of PSO. Finding the ideal swarm size at which PSO can achieve its best convergence performance is the primary concern here. 20, 50, 70, 100, and 120 are the respective settings.

D. Stopping Requirements

To end the PSO run, two different kinds of ending criteria are typically applied. When the first stopping criterion is met, the PSO process ends after a predetermined number of iterations.

Both Beheshti and Shamsuddin (2014) and Ratnaweera, Halgamuge, and Watson (2004) have made substantial use of this criterion in their writings. Variability in feature evaluations (FEs) is the second preventing condition (Shami, El-Saleh, Alswaitti, Al-Tashi, Summakieh & Mirjalili, 2022).

FEs = S * T

wherein S is the swarm length and T is the most variety of iterations.

E. Controlling Parameters of PSO

PSO has 3 foremost controlling parameters:

1. Inertia weight w,
2. The cognitive element c1, and
3. The social element c2.

ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025

These parameters have a great effect on PSO performance, and employing the appropriate setting of those parameters will help achieve the best overall performance. Numerous studies have attempted to enhance the overall performance of PSO through the use of unique ways to tune those regulating parameters. The parameters employed in this picture are little different from those typically seen in literature. As shown in Table 5, a careful selection of criteria was made for the purpose of this painting.

These parameters have a great effect on PSO performance, and employing the appropriate setting of those parameters will help achieve the best overall performance. Numerous studies have attempted to enhance the overall performance of PSO through the use of unique ways to tune those regulating parameters. The parameters employed in this picture are little different from those typically seen in literature. As shown in Table 5, a careful selection of criteria was made for the purpose of this paper.

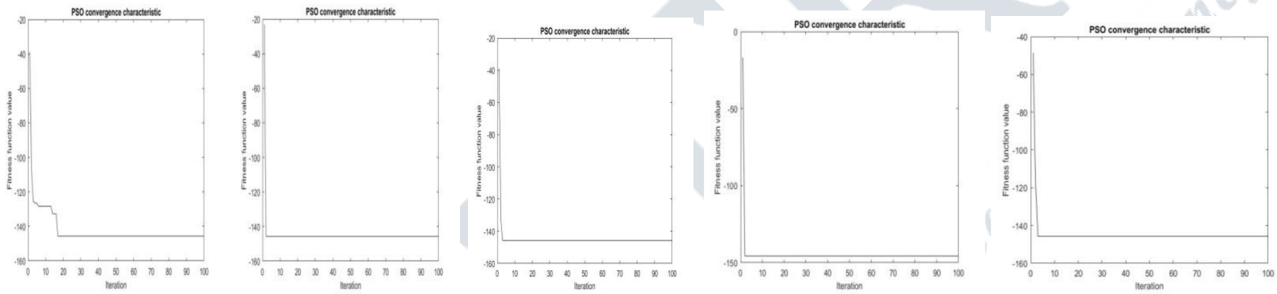
Table V: Parameter setting for PSO

Parameters	**Quantity**
Number of variables	5
Inertial weight	1.2
Inertial weight	0.4
Acceleration coefficient - A1	2.8
Acceleration coefficient -A2	1.3
Maximum no runs	30
Maximum no number of iterations	100
Population considered	20, 50, 70, 100, 120

The specifications of the system for this work is Intel(R) Core (TM) i3-1005G1 CPU @ 1.20GHz 1.19 GHz

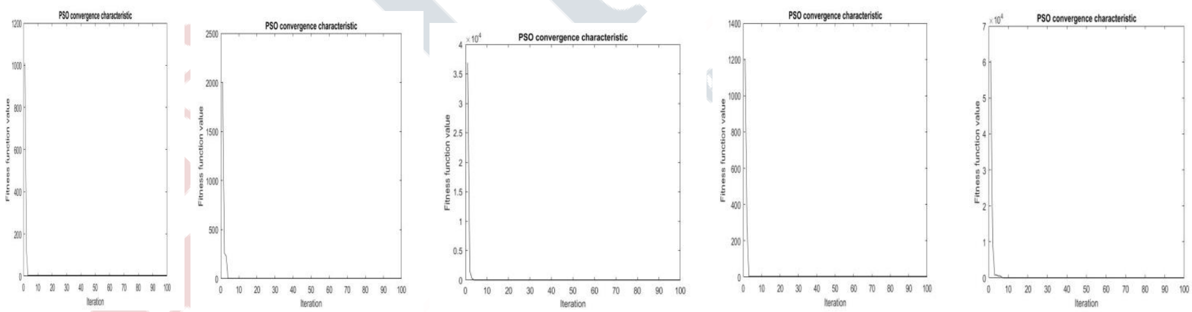
Installed RAM 4.00 GB (3.79 GB usable).

The figures below represent the results of the work using Matlab for PSO.



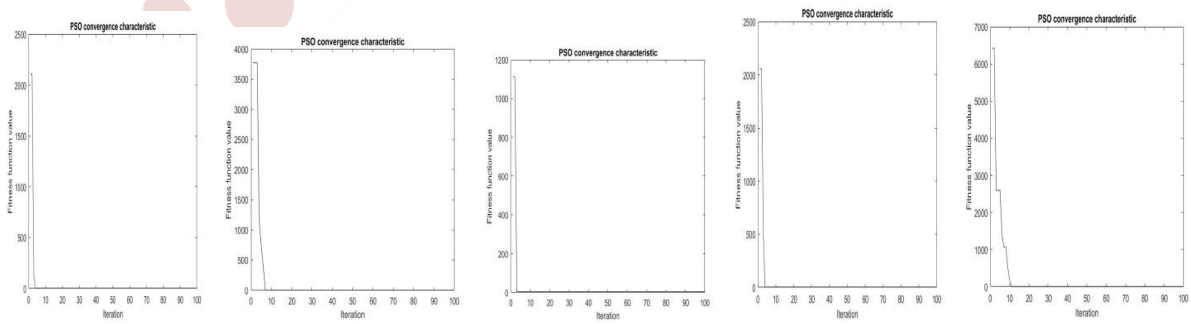
(a) (b) (c) (d) (e)

Fig. 13. Sphere function



(a) (b) (c) (d) (e)

Fig. 14. Rosembrock function



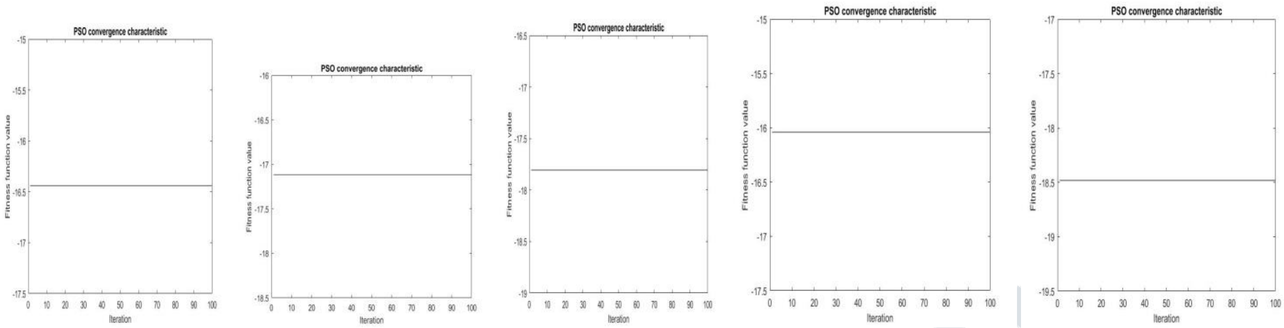
(a) (b) (c) (d) (e)

Fig. 15. Quartic function

ISSN (Online) 2394-2320

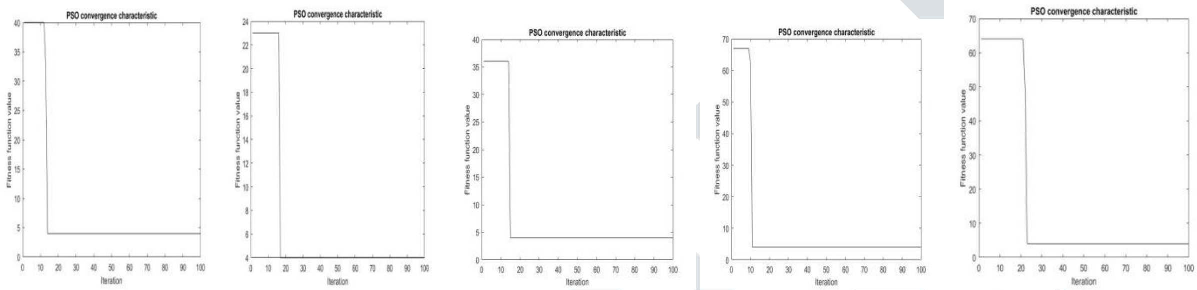
International Journal of Engineering Research in Computer Science and Engineering (IJERCSE)

Vol 12, Issue 01, January 2025



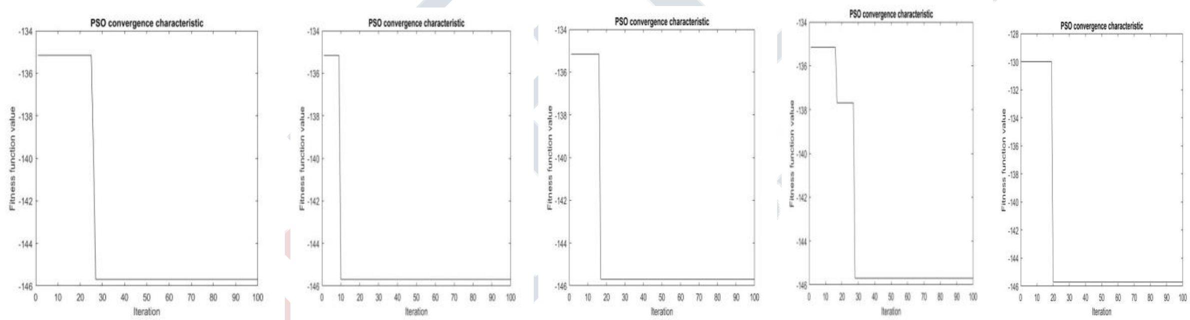
(a) (b) (c) (d) (e)

Fig. 16. Schwefel2.26 function



(a) (b) (c) (d) (e)

Fig. 17. Rastrigin function



(a) (b) (c) (d) (e)

Fig. 18. Ackley function

REFERENCES

[1] Abdollahzadeh, B.; Gharehchopogh, F.S.; Mirjalili, S. African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. Comput. Ind. Eng. 2021, 158, 107408.

[2] Beheshti, Z. and Shamsuddin, S. M. H. (2014). "CAPSO: Centripetal accelerated particle swarm optimization," Inf. Sci., vol. 258, pp. 54-79.

[3] Bhardwaj, S.; Kim, D.-S. Dragonfly-based swarm system model for node identification in ultra-reliable low-latency communication. Neural Comput. Appl. 2021, 33, 1837–1880.

[4] Biswas,A., Kalayci, C. B., & Mirjalili, S., 2023). Advances in Swarm Intelligence Variations and Adaptations for Optimization Problems. Series Editor. Kacprzyk, J., Studies in Computational Intelligence Vol. 1054. Springer. ISBN 978-3-031-09834-5 ISBN 978-3-031-09835-2 (eBook). https://doi.org/10.1007/978-3-031-09835-2.

[5] Bonyadi, M. R. & Michalewicz, Z. Particle Swarm Optimization for Single Objective Continuous Space Problems: A Review. Evolutionary Computation. (2016). 25(1): 1–54. doi:10.1162/EVCO_r_00180.

[6] Dhiman, G.; Garg, M.; Nagar, A.; Kumar, V.; Dehghani, M. A novel algorithm for global optimization: Rat Swarm Optimizer. J. Ambient. Intell. Humaniz. Comput. 2020, 12, 8457–8482.

[7] Du, K. & Swamy, M.N.S. (2016). Search and

Optimization by Metaheuristics Techniques and Algorithms Inspired by Nature. ISBN 978-3-319-41191-0 ISBN 978-3-319-41192-7 (eBook) DOI 10.1007/978-3-319-41192-7. Library of Congress Control Number: 2016943857. Mathematics Subject Classification (2010): 49-04, 68T20, 68W15 © Springer International Publishing Switzerland 2016.

[8] Gheitanchi, S., Ali, F. & Stipidis, E. (2010). Particle Swarm Optimization for Adaptive Resource Allocation in Communication Networks. EURASIP Journal on Wireless Communications and Networking. Hindawi Publishing Corporation. Volume 2010, Article ID 465632, 13 pages doi: 10.1155/2010/465632.

[9] Gad, A. G. (2022). Particle Swarm Optimization Algorithm and Its Applications:A Systematic Review. Archives of Computational Methods in Engineering (2022) 29:2531–2561. https://doi.org/10.1007/s11831-021-09694-4.

[10] Guo, J, zhou, G., Di, Y., Shi, B., Yan, K., and Sato, Y. (2023). A Bare-Bones Particle Swarm Optimization with Crossed Memory for Global Optimization. IEEE Access. Digital Object Identifier 10.1109/ACCESS.2023.3250228.

[11] Hussain, K., Mohd Salle, M. N., Cheng, S. and Shi, Y. (2018). On the exploration and exploitation in popular swarm-based metaheuristic algorithms. Neural Computing and Applications https://doi.org/10.1007/s00521-018-3592-0.

[12] Hussain, K, Salleh, M. N. M. Cheng, S. & Naseem, R. (2017). Common Benchmark Functions for Metaheuristic Evaluation: A Review. International Journal on Informatics Visualization (JOIV). Vol. 1 No 4(2), e-ISSN: 2549-9904 ISSN: 2549-9610.

[13] Jamil, M. & Yang, X. (2013). A literature survey of benchmark functions for global optimisation problems. Int. J. Mathematical Modelling and Numerical Optimisation, Inderscience Enterprises Ltd. Vol. 4(2)

[14] Jia, H.; Sun, K.; Zhang, W.; Leng, X. An enhanced chimp optimization algorithm for continuous optimization domains. Complex Intell. Syst. 2022, 8, 65–82.

[15] Jordehi, A. R., & J. Jasni, J. (2013) Parameter selection in particle swarm optimisation: a survey, Journal of Experimental & Theoretical Artificial Intelligence, 25:4, 527-542, DOI:10.1080/0952813X.2013.782348.

[16] Kennedy, J. and Eberhart, R., ''Particle swarm optimization,'' in Proc. Int. Conf. Neural Netw., vol. 4, Nov./Dec. 1995, pp. 1942–1948.

[17] Khan, S. Yang, S. Fahad, S. U. Khan, & Kalimullah (2021). A Modified Particle Swarm Optimization with a Smart Particle for Inverse Problems in Electromagnetic Devices. Digital Object Identifier 10.1109/ACCESS.2021.3095403. IEEE acess.

[18] Leren Qian, L., Khishe, M., Huang, Y., and Mirjalili, S. (2024). SEB-ChOA: an improved chimp optimization algorithm using spiral exploitation behavior. Neural Computing and Applications (2024) 36:4763–4786 https://doi.org/10.1007/s00521-023-09236-y.

[19] Mahamad Nabab Alam. MATlab Codes for Particle Swarm Optimisation. ResearchGate. (2016) https://www.researchgate.net/publication/296636431

[20] Martínez-Álvarez, F.; Asencio-Cortés, G.; Torres, J.F.; Gutiérrez-Avilés, D.; Melgar-García, L.; Pérez-Chacón, R.; Rubio-Escudero, C.; Riquelme, J.C.; Troncoso, A. Coronavirus optimization algorithm: A bioinspired metaheuristic based on the COVID-19 propagation model. Big Data 2020, 8, 308–322.

[21] MiarNaeimi, F., Azizyan, G. & Rashki, M. Horse herd optimization algorithm: A nature-inspired algorithm for high-dimensional optimization problems. Knowl. Based Syst. 2021, 213, 106711.

[22] Mirjalili, S. and Dong, J. S. (2020). Multi-Objective Optimization using Artificial Intelligence Techniques. Chapter 3 Multi-objective Particle Swarm Optimization. Series Editor, J. Kacprzyk, Springer Briefs in Applied Sciences and Technology. Computational Intelligence. http://www.springer.com/series/10618. ISBN 978-3-030-24835-2 (eBook) . https://doi.org/10.1007/978-3-030-24835-2.

[23] Mohammed, A.,Ngadi, M. A., Salihu I. D., Shafi'i M. A., & Mohammed J. U. (2019). "A survey of symbiotic organisms search algorithms and applications", Springer-Verlag London Ltd. Neural Computing and Applications (2019), https://doi.org/10.1007/s00521-019-04170-4.

[24] Mohamed, A.W., Hadi, A. A., & Mohamed, A.K. (2020). Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm. Int. J. Mach. Learn. Cybern., 11, 1501–1529.

[25] Niu, B., Zhu, Y., He, X., and Wu, H. (2007). MCPSO: A multi-swarm cooperative particle swarm optimizer. Appl. Math. Comput., 185, 1050–1062.

[26] Plevris, V. & Solorzano, G. A. (2022). Collection of 30 Multidimensional Functions for Global Optimization Benchmarking. Data-MDPI 7(46). https://doi.org/10.3390/data7040046.

[27] Priya, P., & Kamilu, S. S. (2023). Unmanned Aerial System Trajectory Tracking Based on Diversified Grey Wolf Optimization Algorithm. Digital Object Identifier 10.1109/ACCESS.2023.3346681. IEEE acess. Vol. 11(2023)

[28] Ratnaweera, A., Halgamuge, S. K. and Watson, H. C. (2004). "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," IEEE Trans. Evol. Comput., vol. 8, no. 3, pp.240-255.

[29] 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.

[30] Tilahun, S. L. (2019). Balancing the Degree of Exploration and Exploitation of Swarm Intelligence Using Parallel Computing. International Journal on Artificial Intelligence Tools. Vol. 28, No. 3. 1950014 (20 pages). World Scientific Publishing Company. DOI: 10.1142/S0218213019500143.

[31] Van Zyl, J. P. & Engelbrecht, A.P. (2023). Set-Based Particle Swarm Optimisation: A Review. Mathematics. https://www.mdpi.com/journal/mathematics. https://doi.org/10.3390/math11132980.

[32] Wang, D., Tan, D., and Liu, L. (2017). Particle swarm optimization algorithm: an overview. Spinger. Soft Comput. DOI 10.1007/s00500-016-2474-6

[33] Zhou, H. & Wei, X. (2021). Particle Swarm Optimization Based on a Novel Evaluation of Diversity. Algorithms, 14, 29. https://doi.org/10.3390/a14020029)., -volV)

[34] Zhao, W.; Wang, L.; Mirjalili, S. Artificial hummingbird algorithm: A new bio-inspired optimizer with its engineering applications. Comput. Methods Appl. Mech. Eng. 2022, 388, 114194.