Int J Syst Assur Eng Manag (August 2024) 15(8):3878–3893
https://doi.org/10.1007/s13198-024-02386-9

ORIGINAL ARTICLE

A novel hybrid pelican-particle swarm optimization algorithm (HPPSO) for global optimization problem

Amit Raj[1] · Parul Punia[1] · Pawan Kumar[1]

Received: 28 November 2023 / Revised: 2 May 2024 / Accepted: 23 May 2024 / Published online: 18 June 2024

Abstract Particle Swarm Optimization (PSO) has drawn attention due to its widespread use in scientific and engineering fields. However, it suffers from a major limitation which is its slow exploration capability leading to stagnation. To overcome this limitation, various algorithms have been hybridized to improve the exploration phase of PSO but still there is a need to improve it further. Keeping this in mind, this paper proposes a novel hybrid meta-heuristic algorithm called the Hybrid Pelican-Particle Swarm Optimization (HPPSO) for solving complex optimization problems. The purpose of hybridization is motivated by the excellent exploration capability of the Pelican Optimization Algorithm (POA). The performance of the proposed HPPSO has been tested on 33 standard benchmark functions in MATLAB (R2023a). For evaluation, the obtained results of proposed HPPSO algorithm are compared with conventional PSO and POA along with other numerous hybridized algorithms of PSO (PSOGSA, HFPSO, PSOBOA, and PSOGWO). The results are analyzed statistically through convergence curves, boxplot and a non-parametric Wilcoxon signed rank test. These analyses show that the proposed HPPSO algorithm achieves a better optimum than other algorithms used in the present paper.

Keywords Exploration · Exploitation · Meta-heuristic algorithms · Particle swarm optimization · Pelican optimization algorithm

✉ Pawan Kumar
drpawan@cuh.ac.in; chhoker.pawan@rediffmail.com

[1] Department of Mathematics, Central University of Haryana, Mahendergarh, Haryana 123031, India

Introduction

In recent years, optimization has become a fascinating area of research due to the increasing complexity and diversity of optimization problems across various fields such as engineering, wireless sensor network (Gokulraj et al. 2021; Abdulai et al. 2023; Vahabi et al. 2022; Raja and Mookhambika 2022; Navin Dhinnesh and Sabapathi 2022; Dao et al. 2023; Jain et al. 2023; Thalagondapati and Singh 2023; Verma and Jain 2023; Boyineni et al. 2024), forecasting (Kim and Moon 2019; Roy et al. 2020; Dong et al. 2022; Nayak et al. 2023; Singh and Rizwan 2023; Wang et al. 2023; Danandeh Mehr et al. 2023), search engine optimization (Sethuraman et al. 2019), science (Chakrabarti and Chakrabarty 2019; Gupta et al. 2020; Muruganantham and Gnanadass 2021; Avvari and Vinod Kumar 2022) etc. The progress in optimization problems has made it challenging for traditional optimization techniques to effectively solve them. These techniques typically rely on gradient-based approaches or assume convexity in the problem space (Hariharan et al. 2023).

The limitations of traditional optimization have prompted researchers to explore alternative strategies capable of navigating the complexities of contemporary optimization problems. This exploration has led to the development and application of meta-heuristic algorithms. These algorithms do not require the problem to be convex or differentiable and are capable of searching large and complex spaces more efficiently (Rao 2019). They provide a framework for developing solution strategies that are adaptable, robust, and capable of finding satisfactory solutions with less computational effort.

Most of these algorithms are applied in different areas such as artificial neural network (Movassagh et al. 2021), forecasting (Sengar and Liu 2020; Murali et al. 2020), malware detection (Alzubi et al. 2022a; b, 2023), economic load dispatch problem (Padhi et al. 2020), optical wavelength division multiplexing (WDM) system (Bansal et al. 2017; Bansal 2021) and wireless sensor network (Halllafi et al. 2023; Vasanthi and Prabakaran 2023; Saranraj et al. 2022; Srinivas and Amgoth 2023; Khalifa et al. 2023; Dash, 2023). The first and most well-known meta-heuristic algorithm is Genetic Algorithm (GA) (Holland, 1992; Goldberg 1989). GA algorithm is inspired by the process of natural selection in genetics and have applications in diverse fields including networking and scheduling (Shinde and Bichkar 2023).

Thereafter, many researchers came up with new and hybrid ways to find the best solution based on evolutionary, food-searching, and physical principles of the universe. A few of the widely used methods include Particle Swarm Optimization (PSO) (Kennedy and Eberhart 1995), Gravitational Search Algorithm (GSA) (Rashedi et al. 2009), Monkey Search (MS) (Sharma et al. 2016), Differential Evolution (DE) (Storn and Price 1997), Simulated Annealing (SA) (Zhang and Wang, 1993), Artificial Bee Colony (ABC) (Garg 2014), Multi-Objective Generalized Teacher-Learning-Based-Optimization Algorithm (Ram et al. 2022).

Out of these, PSO is particularly effective at optimizing problems with continuous variables and has a rapid convergence rate compared to earlier algorithms. However, it has limitations in the exploration phenomenon due to which it may get stuck in local optima, particularly for functions with multiple local optima. Over the years, researchers have proposed various methods of PSO such as variants, improvements and hybridization to deal with its limitations (Houssein et al. 2021; Gad 2022). The different variants of PSO, such as binary, chaotic and multi-objective have been developed to enhance the performance of PSO.

Hybridization of algorithms is another way to enhance the performance of an algorithm by combining their best parts. The need for such hybridization arises from the fact that there are many different types of functions ranging from simple to complex real-world problems that need to be optimized. Since a single algorithm is typically designed around a single logical strategy, it cannot optimize every type of function. Also, different types of functions require different search strategies. This is where the concept of hybridization becomes relevant. By merging two distinct approaches, it can enhance the performance of more functions than the individual approach.

One hybridization approach with PSO is PSOGSA, proposed by Mirjalili and Hashim (2010) in which the ability of exploration in GSA is combined with the ability of exploitation in PSO. Chopra et al. (2016) hybridized PSO with the grey wolf optimization (GWO) algorithm to solve the economic load dispatch problem by imitating the grey wolves' leadership hierarchy and hunting mechanism. Yang et al. (2020) proposed three strategies to enhance the global optimization ability of the Butterfly Optimization Algorithm (BOA) (Arora and Singh 2019). The strategies include initializing BOA using a chaotic cubic map, applying a nonlinear parameter control strategy to the power exponent, and combining BOA with the PSO algorithm in a hybrid approach. The goal of these strategies is to address some of the limitations of the basic BOA and improve its ability to find the global optimum. However, it is important to note that the effectiveness of these strategies may vary depending on the specific optimization problem at hand. The study suggests that making innovative modifications and hybridizing with other algorithms can potentially improve the optimization capabilities of an algorithm. This paper introduces a novel hybrid approach called hybrid pelican-particle swarm optimization algorithm (HPPSO) by combining the search principle of Pelican Optimization Algorithm (POA) with PSO that eliminates the stagnation effect of PSO. POA was proposed by Pavel Trojovsky and Mohammad Dehghani, taking inspiration from the foraging behavior of pelicans in search of food. It is highly effective at exploring and is particularly suitable for optimizing functions with a bowl-shaped structure. But there is no assurance that the optimization solutions obtained through the use of POA will always be the global optimum for all optimization problems. The hybridization process in this paper differs from previous works as it combines the exploration phase of POA and the exploitation phases of PSO algorithms to create a novel high-performing algorithm. More detailed information about the algorithm's search principle is explained in Sect. 4 of the paper. The main contributions of the paper are as follows:

- A novel hybrid optimization algorithm, called hybrid pelican-particle swarm optimization algorithm (HPPSO) has been proposed by combining two meta-heuristic algorithms PSO and POA.
- To validate the proposed HPPSO algorithm, it has been tested on 33 benchmark mathematical functions in MATLAB (R2023a).
- The obtained results of the proposed HPPSO algorithm are compared with conventional PSO and POA along with other numerous hybridized algorithms of PSO such as PSOGSA, HFPSO, PSOBOA and PSOGWO.
- The performance of the proposed HPPSO algorithm has been analyzed statistically through convergence curve, boxplot and a non-parametric Wilcoxon sign rank test.
- From the above analyses, the proposed hybrid algorithm performs better than other compared algorithms used in the paper.

The subsequent sections of the paper are structured as follows: Sect. 2 explains the working mechanisms of the PSO and POA algorithms. Section 3 presents the proposed HPPSO algorithm. Section 4 contains the results, including the performance evaluations and statistical analysis. Finally, Sect. 5 presents the conclusion and the future scope.

1 Related work

This section provides a concise explanation of the working principles and basic parameters of PSO and POA algorithms that are essential components of our proposed algorithm. Since POA is the latest algorithm and PSO is widely used, we will simply provide the basic concepts of these algorithms to facilitate a better understanding of our proposed algorithm.

1.1 Particle swarm optimization

Kennedy and Eberhart developed the PSO algorithm in 1995, drawing an inspiration from the social behavior of a swarm of particles moving in a search space. In the algorithm, each member of the swarm is referred to as a particle and has two essential features: velocity and position. These features are utilized in determining the optimal value. The algorithm starts by initializing a population of particles, each with a randomly generated position and velocity in a search space with 'd' dimensions and a swarm of 'N' particles and evaluates the objective (fitness) function at each position.

Then, the velocity and position of each particle are updated by using the Eqs. (1) and (2).

$$v_k(t+1) = \omega \times v_k(t) + c_1 \times r_1 \times (pbest_k - x_k(t)) + c_2 \times r_2 (gbest - x_k(t)) \tag{1}$$

$$x_k(t+1) = x_k(t) + v_k(t+1) \tag{2}$$

where:

$x_k(t)$: the current position of the k^{th} particle at time t
ω: the inertia weight
$v_k(t)$: the current velocity of the k^{th} particle at time t
$pbest_k$: the personal best position of the k^{th} particle
gbest: the global best position of any particle in the swarm
c_1: cognitive constant and c_2 is social constant
r_1 and r_2 are two random numbers that take values between 0 and 1.

The inertia weight controls the impact of the particle's previous velocity on its current velocity while c_1, c_2, r_1 and r_2 control the influence of the personal and global best positions on the particle's movement.

At each iteration, the fitness of each particle is evaluated based on the fitness function. If a particle's fitness is better than its personal best, it's personal best is updated. If the particle's fitness is better than the global best, the global best is updated. The algorithm terminates when a predefined stopping criterion is met, such as reaching a maximum number of iterations or finding a solution satisfying a specified fitness level. The final solution is the global best position found by any particle in the population.

Despite the successful implementation of PSO in various optimization problems, it still has some limitations. One major limitation of the PSO algorithm is the risk of premature convergence. This happens when the particles in the swarm converge to a suboptimal solution rather than exploring the entire search space, resulting in the inability to reach the global optimal solution.

1.2 Pelican optimization algorithm

Pelican optimization algorithm (POA) is also a population-based algorithm that takes inspiration from the natural behaviors of pelicans. The algorithm is designed to mimic the strategies and behavior that pelicans exhibit during hunting. The algorithm was proposed by Pavel Trojovsky and Dehghani (2022) in which pelicans are considered as the members of the population.

The approach of pelicans when they hunt for food is replicated to improve the candidate solution in two phases through simulation after initializing the position of pelicans randomly in the search area. These two phases are as follows:

Phase 1: Exploration phase (Moving towards food source): In this phase, the algorithm tries to explore the search space for pelicans to find their food source (prey) randomly. Once their prey is detected, the pelicans move towards them. The position of the i^{th} pelican candidate solution is updated in this phase using the following equations:

$$X_i^{New\_P1} = \begin{cases} X_i + rand.(L_P - I.X_i), & F_p < F_i \\ X_i + rand.(X_i - L_P), & \text{else} \end{cases} \tag{3}$$

where:

X_i: is the initial position of the candidate solution
L_P: is the location of prey
F_p: is the fitness function
I: is 1 or 2, selected randomly for each iteration

If the value of the fitness function at the new position is better than the value at the current position, then the pelican's new position is considered by using the following equation:

$$X_i = \begin{cases} X_i^{New\_P1}, & F_i^{P1} < F_i \\ X_i, & \text{else} \end{cases} \quad (4)$$

Phase 2: Exploitation phase (Winging on the water surface): After the exploration phase, the algorithm enters into the exploitation phase. In this phase, the pelicans use their wings to create a space over the water's surface, allowing the prey to move upwards. This process enhances the local search ability. The position of the i^{th} pelican candidate solution is updated in phase 2 using the following equations:

$$X_i^{New\_P2} = X_i + R.(1 - (c_{ite}/Max_{ite})).(2.rand - 1).X_i, \quad (5)$$

Where:

c_{ite}: is the current iteration
Max_{ite}: is the maximum number of iterations
R: is 0.2,

Then, the process of accepting or rejecting a new pelican position is utilized by the following equation:

$$X_i = \begin{cases} X_i^{New\_P2}, & F_i^{P2} < F_i \\ X_i, & \text{else} \end{cases} \quad (6)$$

2 The proposed HPPSO algorithm

This section describes the thought process that went into developing the proposed HPPSO algorithm and its structure and basic working principles.

2.1 Basic idea

Achieving a global optimum requires an optimization algorithm to maintain a balance between exploring and exploiting solutions. When trying to solve an optimization problem, exploration brings in variety while exploitation suggests intensity. We have already seen that PSO's stagnation effect arises when the algorithm's exploration and exploitation phases are out of balance. Therefore, the idea of hybridization with the excellent exploration capability of POA is used to overcome it. In the proposed algorithm, POA is used to generate a working solution by initially exploring the search space and then the PSO is applied to optimize the solution by improving upon the POA's output.

2.2 Implementation of the algorithm

In the proposed HPPSO algorithm, the initial positions of 'P' particles are randomly generated within the boundaries of a search area that has 'D' dimensions. After initializing the position of particles, the primary goal of the algorithm is to explore the search area thoroughly to identify the best possible solutions for a given problem. To achieve this, the algorithm uses the phase 1 mechanism of POA. This mechanism employs the algorithm to explore the search area efficiently. Once the exploration phase is complete, the algorithm moves into the exploitation phase. In this phase, the HPPSO algorithm passes the particles to the PSO technique as initial points for the exploitation phase (Fig. 1).

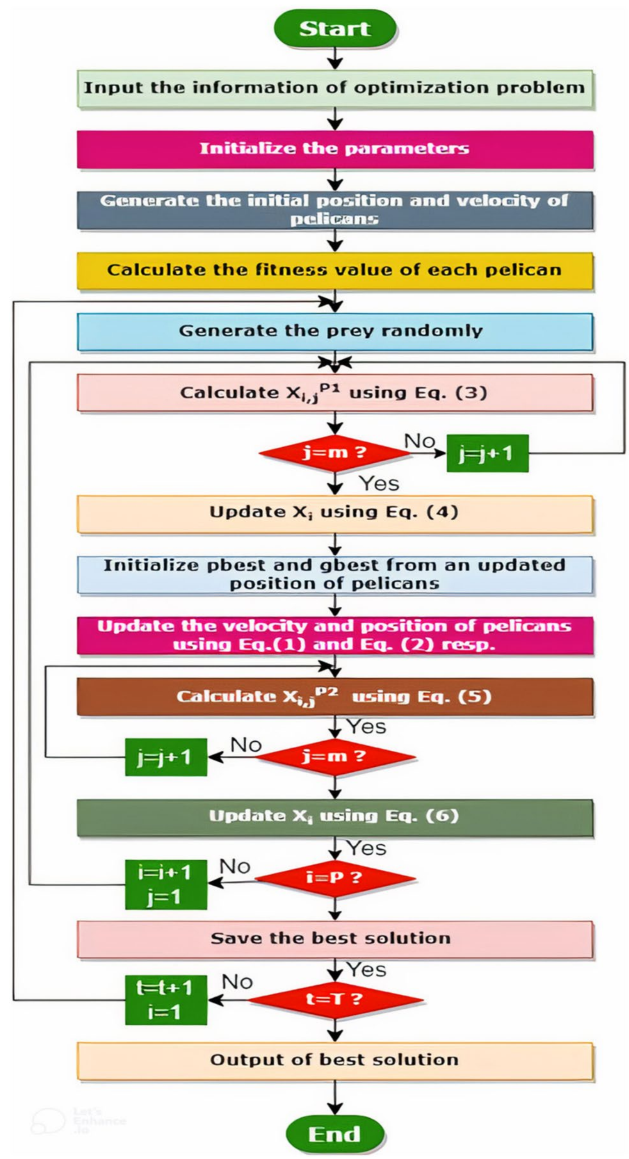


Fig. 1 Flowchart of the proposed HPPSO algorithm

Algorithm 1 Hybrid pelican particle optimization algorithm (HPPSO)

```
Input the details of an optimization problem
Initialize the parameters such as P, c1, c2, w and T.
Initialize the position and the velocity of the pelicans randomly.
Calculate the objective function (or fitness ) of each pelican
for t = 1 : T do
    Generate the position of the prey randomly
    for i = 1 : P do
        Calculate the position of pelicans by Eq. (3)
        Update the position of pelicans by Eq. (4)
        Initialize the pbest and gbest from an updated position of the pelicans
        Update the velocity and the position of the pelicans by Eq. (1) and (2)
        Calculate the objective value of each pelican.
        Calculate the new position of pelicans by Eq. (5) and update it by Eq. (6).
    end for
    Update best candidate solution
end for
Output best candidate solution obtained by HPPSO
```

Table 1 Parameters value of the compared algorithms

Algorithms	Parameter	Value
	Inertia weight	$W_{max} = 0.9, W_{min} = 0.2$
PSO	Velocity limit	10% of dimension range
	Cognitive constant (C1)	2
	social constant (C2)	1
	Number of particles	50
	Gravitational constant (G0)	1
PSOGSA	C1	2
	C2	1
	Search agents	50
	a	0.2
PSOGWO	c1	2
	c2	1
	a	0.2
HFPSO	B_0, γ	2, 1
	c1, c2	2, 1
	Search agents	50
	a, c(0)	0.1, 0.01
PSOBOA	p, c1, c2	0.6, 2, 1
	Search agents	50
	Search agents	50
POA	R	0.2
	Search agents	50
HPPSO	R	0.2
	c1	2
	c2	1

Table 2 Descriptions of 33 Standard benchmark functions

Functions	Dimension	Range	F_{min}
$F_1(x)$= Sphere	30	[−100 100]	0
$F_2(x)$= Schwefel 2.22	30	[−10 10]	0
$F_3(x)$= Schwefel 1.2	30	[−100, 100]	0
$F_4(x)$= Schwefel 2.21	30	[−100, 100]	0
$F_5(x)$= Rosenbrock	30	[−30, 30]	0
$F_6(x)$= Step	30	[−100, 100]	0
$F_7(x)$=Quartic	30	[−1.28, 1.28]	0
$F_8(x)$= Schwefel	30	[−500, 500]	-12569
$F_9(x)$= Rastrigin	30	[−5.12, 5.12]	0
$F_{10}(x)$= Ackley	30	[−32, 32]	0
$F_{11}(x)$= Griewank	30	[−600, 600]	0
$F_{12}(x)$=Penalized	30	[−50, 50]	0
$F_{13}(x)$= Penalized2	30	[−50, 50]	0
F_{14}= Foxholes	2	[−65.536, 65.536]	1
$F_{15}(x)$= Kowalik	4	[−5, 5]	0.0003
$F_{16}(x)$= Six Hump Camel	2	[−5, 5]	−1.0316
$F_{17}(x)$= Branin	2	[−5, 10] × [0, 15]	0.398
$F_{18}(x)$ = GoldStein-Price	2	[−2,2]	3
$F_{19}(x)$ = Hartman 3	3	[0, 1]	−3.86
$F_{20}(x)$ =Hartman 6	3	[0, 1]	−3.22
$F_{21}(x)$ = Shekel 5	3	[0, 10]	−10.1532
$F_{22}(x)$ = Shekel 7	3	[0, 10]	−10.4029
$F_{23}(x)$ = Shekel 10	3	[0, 10]	−10.5364
$F_{24}(x)$ = Colville	4	[−10, 10]	0
$F_{25}(x)$ = Zakharov	10	[−5, 10]	0
$F_{26}(x)$ = Schwefel's 2.20	30	[−100, 100]	0
$F_{27}(x)$ = Brown	30	[−1, 4]	0
$F_{28}(x)$ = Powell Singular	30	[−4, 5]	0
$F_{29}(x)$ = Sum Squares	30	[−10, 10]	0
$F_{30}(x)$ = Egg Crate	2	[−5, 5]	0
$F_{31}(x)$ = Alpine N. 1	30	[−10, 10]	0
$F_{32}(x)$ = Xin-She Yang	30	[−5, 5]	0
$F_{33}(x)$ = Trignometric 2	30	[−500, 500]	0

The PSO mechanism then utilizes the information gathered from the exploration phase to guide the particles towards the best-known solutions in the search space. It helps to improve the diversity of the population and avoid getting stuck in local minima.

3 Results and discussion

To validate the proposed HPPSO algorithm, 33 standard benchmark mathematical functions have been employed. The details of these benchmark functions and parameter settings of the compared algorithms are listed in Sect. 4.1. In section 4.2, a comparative analysis of the obtained results has been carried out to evaluate the performance of HPPSO algorithm with other existing hybridization algorithms of PSO including conventional PSO and POA. Section 4.3 introduces the result analysis by Wilcoxon signed rank test while Sect. 4.4 introduces the boxplot analysis of the obtained result. These statistical analyses has confirmed the effectiveness of the HPPSO Algorithm.

3.1 Experiment settings

The proposed HPPSO algorithm has been implemented in MATLAB R2023a on a computer system consisting of a core i5-11300 H CPU @ 3.10GHz, with 16 GB of RAM.

3.1.1 Parameter settings

In order to achieve the fairness of the test experiment, the parameters of the existing hybridization algorithms of PSO and conventional PSO and POA are same in the simulation experiments. The experiment has been conducted using a population size of 50 and a maximum iteration of 1000 with 20 independent runs for each function. Table 1 provides information about the control parameters of the compared algorithms.

3.1.2 Different benchmark mathematical functions

The proposed HPPSO algorithm's effectiveness is analyzed by testing its performance on three types of functions: Unimodal benchmark functions, Multimodal benchmark functions, and Fixed dimension multimodal functions. The detail description of these functions are listed in Table 2 (Digalakis and Margaritis 2001; Molga and Smutnicki 2005).

3.2 Statistical results and convergence curves analysis

This section discusses the results of implementing the HPPSO algorithm on 33 well-known mathematical functions. The obtained results are compared with recent existing hybridizing optimization algorithms of PSO including conventional PSO and POA. Table 3 demonstrates a comparative analysis of proposed HPPSO and other existing algorithms based on their mean value, standard deviation and best value. From this table, certain conclusions can be drawn:

The proposed HPPSO algorithm performs better than all other existing algorithms used in this study for functions F1-F5. Specifically, when applied to these functions, the HPPSO algorithm is able to achieve better optimization results and is stable due to the lowest standard deviation than other compared algorithms. For function F6, the

Table 3 Statistical results on 33 standard benchmark functions

		PSO	PSOGSA	PSOGWO	HFPSO	PSOBOA	POA	HPPSO
	MEAN	0.0005161	2.11E-18	494.9151	1.98E-11	2.47E-118	5.17E-206	2.45E-286
F_1	S.D	0.000704	3.50E-19	1001.757	2.48E-11	5.62E-118	0	0
	BEST	3.85E-06	1.53E-18	2.87E-49	5.22E-12	1.19E-122	1.75E-234	2.73E-312
	Rank	6	4	7	5	3	2	1
	MEAN	0.0150937	0.000314	4.006761	3.08E-06	6.67E-57	2.49E-106	5.43E-146
F_2	S.D	0.0183494	0.00099	4.678129	1.45E-06	7.70E-57	8.89E-106	2.38E-145
	BEST	0.0001369	5.48E-09	5.17E-27	1.56E-06	3.05E-58	1.19E-113	1.24E-154
	Rank	6	5	7	4	3	2	1
	MEAN	233.46943	374.1903	8919.653	0.002236	7.15E-117	4.09E-211	1.02E-276
F_3	S.D	100.84767	209.9918	11821.19	0.0012385	1.57E-116	0	0
	BEST	114.36928	29.70854	2.25E-22	0.0004725	5.80E-120	1.67E-229	5.80E-301
	Rank	5	6	7	4	3	2	1
	MEAN	2.6920692	20.20699	14.82922	0.0080782	1.14E-58	9.05E-108	1.42E-140
F_4	S.D	1.0795371	18.00711	21.39136	0.0032999	1.05E-58	3.41E-107	6.36E-140
	BEST	0.9111016	8.264264	7.42E-18	0.0033522	1.22E-59	9.98E-116	5.14E-156
	Rank	5	7	6	4	3	2	1
	MEAN	4762.5097	60.17998	401047	30.742626	28.966168	27.8370342	27.45452156
F_5	S.D	20093.846	71.72188	1161266	21.122566	0.0174197	0.96577439	1.104814793
	BEST	4.9739986	22.46966	25.77075	23.238835	28.9211	26.1944461	25.40712433
	Rank	6	5	7	4	3	2	1
	MEAN	0.35	2.15	1357.85	0	0	0	0
F_6	S.D	0.5871429	1.755443	1773.89	0	0	0	0
	BEST	0	0	0	0	0	0	0
	Rank	5	6	7	1	1	1	1
	MEAN	0.0110797	0.04229	0.775868	0.0052167	6.05E-05	6.75E-05	2.40E-05
F_7	S.D	0.0056345	0.018998	1.455158	0.0012868	4.53E-05	3.96E-05	1.29E-05
	BEST	0.004703	0.021024	0.000418	0.0026938	9.93E-06	8.40E-06	1.21E-06
	Rank	5	6	7	4	2	3	1
	MEAN	−8875.105	−7546.634	−5760.806	−7527.2	−2670.544	−7792.059	−9136.838
F_8	S.D	630.98379	781.4522	834.3606	736.42571	409.34915	831.05344	825.9307893
	BEST	−10239.3	−8873.55	−7484.829	−9072.439	−1974.696	−9271.239	−10298.183
	Rank	2	4	6	5	7	3	1
	MEAN	42.393799	112.0816	9.462196	37.261192	0	0	0
F_9	S.D	11.530523	30.56482	15.14683	19.452903	0	0	0
	BEST	20.950866	45.768	0	14.924381	0	0	0
Rank	6	7	4	5	1	1	1	
	MEAN	0.8342155	1.02E-09	2.322208	2.98E-06	4.44E-16	2.75E-15	3.29E-15
F_{10}	S.D	0.7379172	1.23E-10	3.664429	1.51E-06	0	1.74E-15	1.46E-15
	BEST	0.00261	8.45E-10	1.47E-14	1.18E-06	4.44E-16	4.44E-16	4.44E-16
	Rank	6	4	7	5	1	2	3
	MEAN	0.0139929	0.087059	2.412019	0.0122921	0	0	0
F_{11}	S.D	0.0146712	0.233615	4.56764	0.0152571	0	0	0
	BEST	5.25E-06	0	0	1.25E-12	0	0	0
	Rank	5	6	7	4	1	1	1
	MEAN	0.159362	1.235535	136054.4	7.18E-12	0.8103647	0.12895859	0.092189045
F_{12}	S.D	0.3001638	1.386297	557435	5.24E-12	0.2062796	0.04258804	0.04306566
	BEST	6.80E-06	1.52E-03	0.004557	1.40E-12	0.3714775	0.06068416	0.042464207
	Rank	4	6	7	1	5	3	2
	MEAN	0.0121459	0.000549	1282575	3.28E-10	2.9414463	2.51593926	2.760772756
F_{13}	S.D	0.0276929	0.002457	2059942	7.14E-10	0.137093	0.47814626	0.384015438

Table 3 (continued)

		PSO	PSOGSA	PSOGWO	HFPSO	PSOBOA	POA	HPPSO
	BEST	2.20E-06	3.45E-19	0.073977	4.67E-11	2.526936	1.7326217	1.967193528
	Rank	3	2	7	1	6	4	5
	MEAN	0.9980038	5.821959	1.585504	1.0477052	5.6463666	0.99800384	1.146910257
F_{14}	S.D	0	5.461505	2.205164	0.2222712	1.9272912	0	0.48565124
	BEST	0.9980038	0.998004	0.998004	0.9980038	1.9937841	0.99800384	0.998003838
	Rank	1	7	5	3	6	1	4
	MEAN	0.0003622	0.003733	0.000828	0.0004231	0.0010793	0.00030749	0.000313327
F_{15}	S.D	0.0002065	0.007172	0.000511	0.0003584	0.0007728	1.91E-10	2.61E-05
	BEST	0.0003075	0.000307	0.000308	0.0003075	0.0004358	0.00030749	0.000307486
	Rank	3	7	5	4	6	1	2
	MEAN	−1.0316	−1.0316	−1.0316	−1.0316	−1.0248	−1.0316	−1.0316
F_{16}	S.D	2.28E-16	1.76E-16	0.000262	1.97E-16	0.031864	1.84E-16	2.71E-12
	BEST	−1.0316	−1.0316	−1.0316	−1.0316	−0.9843	−1.0316	−1.0316
	Rank	1	1	1	1	7	1	1
	MEAN	0.39789	0.39789	0.39789	0.66303	0.4472	0.39789	0.39789
F_{17}	S.D	0	0	0	0.271411	0.14632	0	0
	BEST	0.39789	0.39789	0.39789	0.39789	0.39789	0.39789	0.39789
	Rank	1	1	1	7	6	1	1
	MEAN	3	3	3.003346	3	8.5597595	3	3
F_{18}	S.D	1.44E-16	2.20E-15	0.007782	5.94E-16	6.2621503	7.06E-16	9.26E-11
	BEST	3	3	3	3	3.2054761	3	3
	Rank	1	1	6	1	7	1	1
	MEAN	−3.8628	−3.8628	−3.8566	−3.8628	−3.4888	−3.8628	−3.8628
F_{19}	S.D	2.28E-15	2.07E-15	0.003675	2.15E-15	0.314389	2.06E-15	2.43E-08
	BEST	−3.862782	−3.862782	−3.862769	−3.862782	−3.045762	−3.86278215	−3.862782148
	Rank	1	1	6	1	7	1	1
	MEAN	−3.268	−3.214	−2.71	−3.298	−2.234	−3.321	−3.321
F_{20}	S.D	0.0606852	0.036595	0.443222	0.0487927	0.3419208	1.61E-06	4.48E-07
	BEST	−3.321	−3.321	−3.321	−3.321	−1.527	−3.321	−3.321
	Rank	4	5	6	3	7	1	1
	MEAN	−7.245892	−6.654923	−4.320569	−10.06604	−3.40539	−9.13359583	−9.643375573
F_{21}	S.D	3.0895351	3.656221	1.494203	0.2934338	0.4049457	2.09217412	1.569123775
	BEST	−10.1532	−10.1532	−5.055045	−10.1532	−4.520608	−10.1531997	−10.15319816
	Rank	4	5	6	1	7	3	2
	MEAN	−9.348117	−9.374134	−4.223858	−10.02024	−3.417585	−9.87141358	−9.871392102
F_{22}	S.D	2.1644523	2.551813	1.702318	1.7074982	0.2725789	1.63600516	1.635997915
	BEST	−10.40294	−10.40294	−5.087543	−10.40294	−3.877311	−10.4029406	−10.40293916
	Rank	5	4	6	1	7	2	3
	MEAN	−9.89188	−8.668747	−4.511911	−8.955182	−3.201786	−10.5364098	−10.53638437
F_{23}	S.D	2.014784	3.363911	1.704805	2.7987303	0.3049506	1.66E-07	2.53E-05
	BEST	−10.53641	−10.53641	−7.73189	−10.53641	−3.71992	−10.5364098	−10.53640807
	Rank	3	5	6	4	7	1	2
	MEAN	0.0015178	0.002368	3.129478	0.0032275	10.599785	3.06E-06	2.96E-06
F_{24}	S.D	0.0033074	0.00952	7.407005	0.0029984	5.0053614	1.29E-05	2.40E-06
	BEST	1.20E-05	7.66E-12	0.000242	3.09E-05	3.3755556	1.96E-24	2.00E-08
	Rank	3	4	6	5	7	2	1
	MEAN	3.77E-21	3.69E-19	0.093675	3.29E-18	4.35E-120	2.17E-219	2.66E-290
F_{25}	S.D	1.04E-20	1.66E-19	0.336384	4.64E-18	8.89E-120	0	0
	BEST	2.58E-28	1.87E-19	1.37E-98	4.07E-20	1.26E-122	2.92E-237	9.06E-316
	Rank	4	5	7	6	3	2	1

Table 3 (continued)

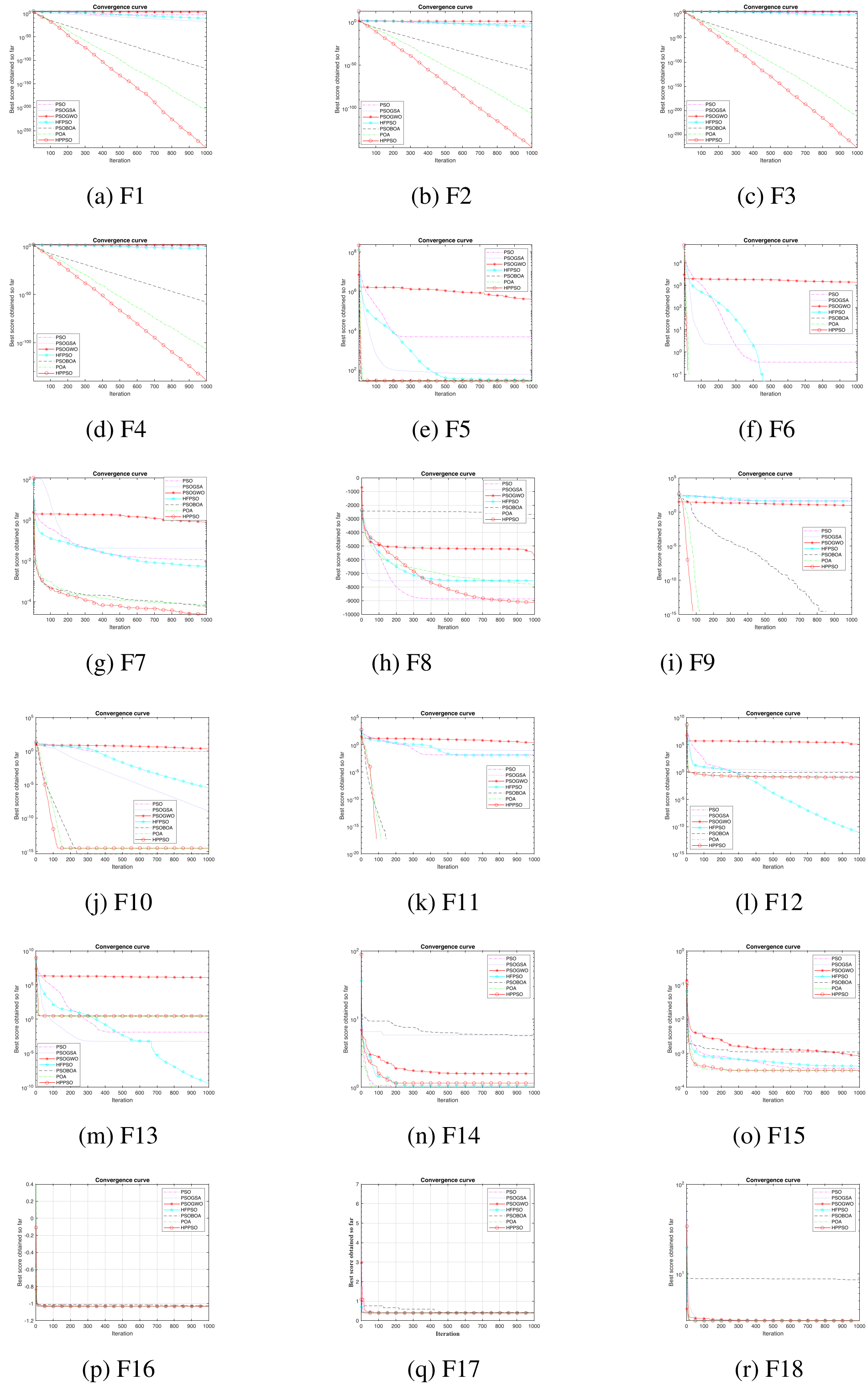
		PSO	PSOGSA	PSOGWO	HFPSO	PSOBOA	POA	HPPSO
	MEAN	0.1474196	6.00E-09	29.31669	5.78E-06	2.86E-56	3.04E-106	1.85E-143
F_{26}	S.D	0.1432545	6.15E-10	38.65453	2.28E-06	3.40E-56	1.26E-105	7.55E-143
	BEST	0.0054616	4.65E-09	4.60E-30	1.93E-06	1.16E-57	2.02E-117	1.56E-152
	Rank	6	4	7	5	3	2	1
	MEAN	4.7500001	4.41E-18	0.332907	7.42E-12	2.78E-119	2.54E-217	2.11E-289
F_{27}	S.D	7.6562669	7.51E-19	0.575162	5.36E-12	5.60E-119	0	0
	BEST	3.88E-09	2.91E-18	1.59E-59	1.20E-12	1.85E-122	9.95E-239	0.00E+00
	Rank	7	4	6	5	3	2	1
	MEAN	3.3391761	0.015115	62.06139	0.0010114	6.53E-117	1.67E-211	1.55E-271
F_{28}	S.D	14.78479	0.008336	115.2207	0.0004127	1.12E-116	0	0
	BEST	0.0021816	0.002628	9.54E-08	0.0003348	3.27E-119	1.28E-231	1.14E-303
	Rank	6	5	7	4	3	2	1
	MEAN	0.1108918	5.87E-17	70.61117	1.78E-10	4.90E-116	9.39E-214	1.77E-285
F_{29}	S.D	0.4955892	2.11E-17	173.0531	2.30E-10	9.54E-116	0	0
	BEST	4.01E-07	3.41E-17	5.50E-44	1.09E-11	1.71E-118	1.07E-236	2.67E-306
	Rank	6	4	7	5	3	2	1
	MEAN	2.46E-206	2.52E-21	0.002136	6.39E-51	2.35E-27	1.71E-307	0
F_{30}	S.D	8.43E-210	8.20E-23	1.11E-241	1.43E-59	4.29E-44	0	0
	BEST	0	2.84E-21	0.006468	2.85E-50	9.14E-27	0	0
	Rank	3	6	7	4	5	2	1
	MEAN	0.0316061	0.757986	1.217475	0.001761	1.44E-33	2.56E-106	8.53E-145
F_{31}	S.D	0.104602	0.778876	1.724801	0.0015251	6.43E-33	8.52E-106	3.63E-144
	BEST	0.0009144	0.029112	6.21E-33	8.17E-06	6.83E-46	5.94E-114	8.08E-157
	Rank	5	6	7	4	3	2	1
	MEAN	0.1937573	620.0419	499.5411	0.0002801	1.37E-10	3.96E-48	1.05E-54
F_{32}	S.D	0.7719138	1741.347	1722.59	0.0009142	2.74E-10	1.28E-47	4.68E-54
	BEST	2.26E-07	0.501897	5.13E-46	6.96E-11	4.36E-13	6.43E-58	7.84E-96
	Rank	5	7	6	4	3	2	1
	MEAN	45.066235	5562.847	19988.79	7.7704892	41.24734	35.2188775	33.8581526
F_{33}	S.D	20.375715	1966.107	36771.52	6.308942	17.620421	8.78848984	6.393683476
	BEST	15.427542	1480.224	16.87612	1.000003	27.300851	21.8281709	21.50883111
	Rank	5	6	7	1	4	3	2
	Average Rank	4.181818	4.727273	6.090909	3.515152	4.333333	1.878788	1.484848
	Overall Rank	4	6	7	3	5	2	1

results indicate that HPPSO displayed the global optimum solution. For function F7, the HPPSO algorithm performs superior to other algorithms with a low standard deviation that indicates the stability of the HPPSO algorithm. For function F8, the HPPSO algorithm performs better than all other compared algorithms. Additionally, the HPPSO algorithm produces the best result which is −10298.183.

The HPPSO algorithm obtained the global optimum solution for the function F9. For F10, the HPPSO algorithm produces the third-best optimal value while PSOBOA performs superior to all the algorithms in terms of their mean values. Additionally, the HPPSO algorithm produces the global optimum value. For function F11, the HPPSO algorithm obtains the global optimum value.

For function F12, the HPPSO algorithm produces the second-best optimal value while the HFPSO algorithm performs better than other algorithms. The HPPSO algorithm performs worse but better than PSOBOA and PSOGWO for the function F13. The PSO and POA provide better results that are close to the global optimal for function F14. For function F15, the HPPSO algorithm produces the second-best optimal value while the POA algorithm performs better than other algorithms. For functions F16-F20, HPPSO outperforms all other algorithms and reaches global optima. The HPPSO algorithm produces

Fig. 2 Convergence curves of all compared algorithms on 33 standard benchmark functions

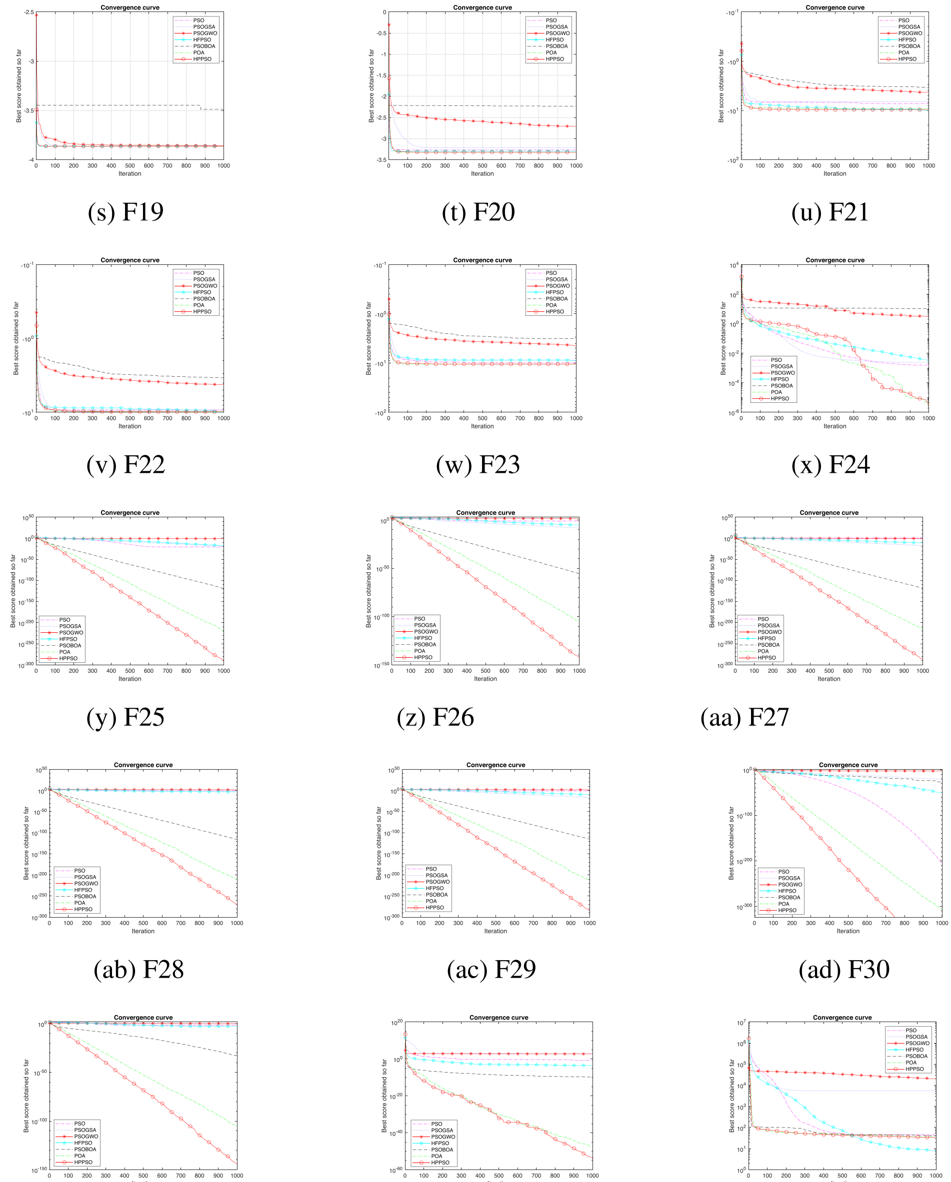


(a) F1 (b) F2 (c) F3
(d) F4 (e) F5 (f) F6
(g) F7 (h) F8 (i) F9
(j) F10 (k) F11 (l) F12
(m) F13 (n) F14 (o) F15
(p) F16 (q) F17 (r) F18

the second-best optimal value for the functions F21 and F23 and the third-best optimal value for the function F22.

For functions F24-F32, the HPPSO performs superior to other compared algorithms and second best for function F33. Thus, The HPPSO algorithm shows a balanced approach between exploring new possibilities and exploiting known solutions, which is demonstrated by its performance on the 33 standard benchmark functions. This balance enables the algorithm to provide the best results for almost all functions that no other algorithm could

Fig. 2 (continued)



(s) F19 (t) F20 (u) F21

(v) F22 (w) F23 (x) F24

(y) F25 (z) F26 (aa) F27

(ab) F28 (ac) F29 (ad) F30

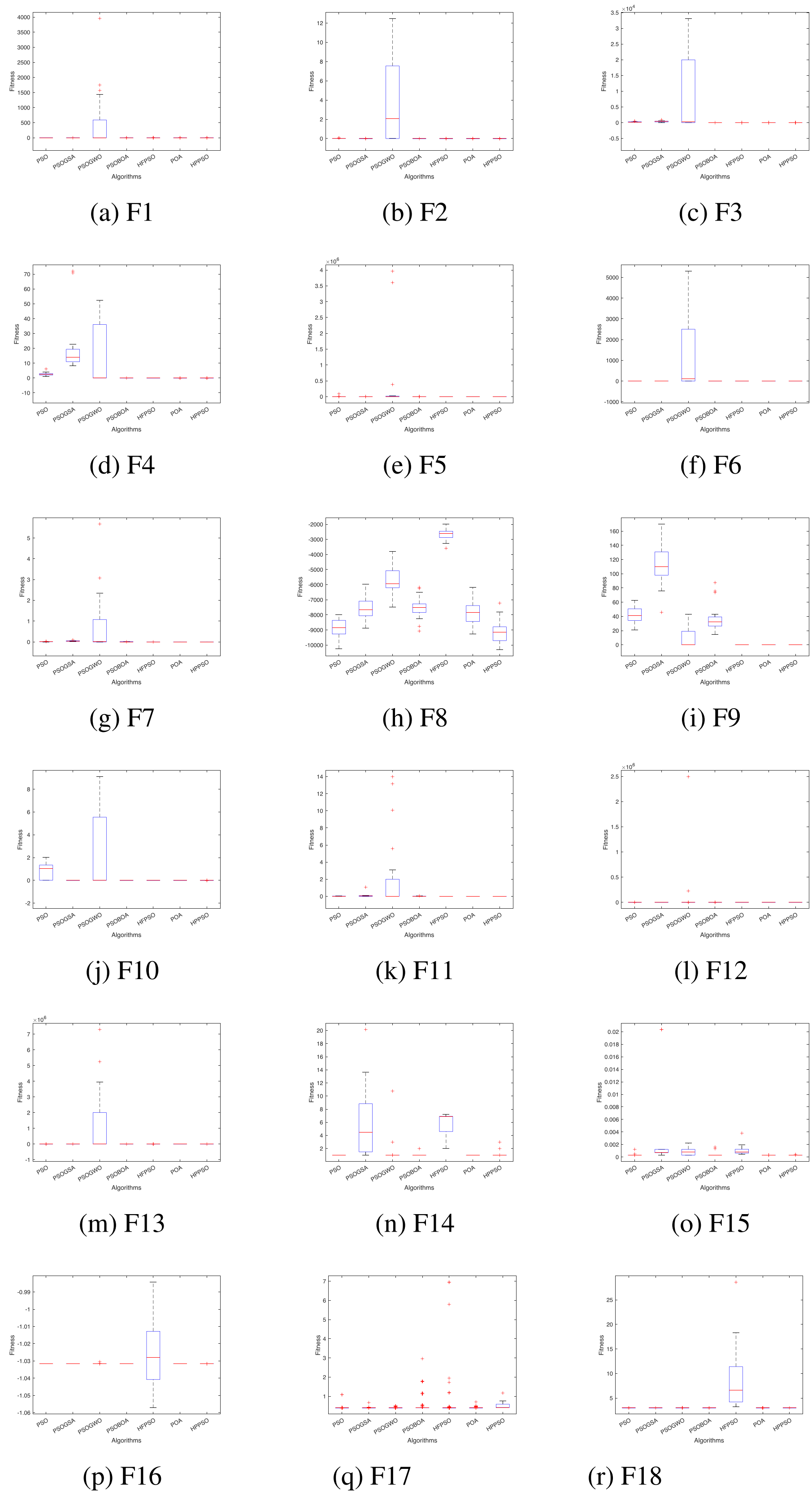
(ae) F31 (af) F32 (ag) F33

Table 4 *p*-value of Wilcoxon sign rank test

Functions	HPPSO vs PSO	HPPSO vs PSOGSA	HPPSO vs PSOGWO	HPPSO vs HFPSO	HPPSO vs PSOBOA	HPPSO vs POA
F1-F7	0.0156	0.0156	0.0156	0.0312	0.0312	0.0312
F8-F13	0.3125	0.2188	0.0312	0.5625	0.2500	0.8750
F14-F23	0.2188	0.0312	0.0156	0.9375	0.0039	0.6250
F24-F33	0.0020	0.0020	0.0020	0.0840	0.0020	0.0020

achieve. As a result, it is highly efficient in achieving an optimal value. So, the proposed HPPSO algorithm is more efficient than other algorithms in finding the optimal solution. This is because of its POA search mechanism which allows for better exploration and avoids local minima. Its performance on functions demonstrates that it is effective in managing complex tasks that require a balance between exploration and exploitation, as well as the ability to avoid local minima.

Fig. 3 Boxplot of all compared algorithms on 33 standard benchmark functions



(a) F1 (b) F2 (c) F3

(d) F4 (e) F5 (f) F6

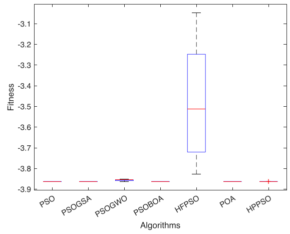
(g) F7 (h) F8 (i) F9

(j) F10 (k) F11 (l) F12

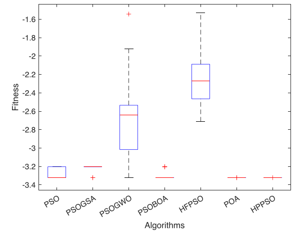
(m) F13 (n) F14 (o) F15

(p) F16 (q) F17 (r) F18

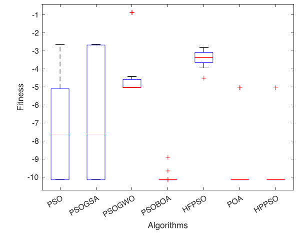
Fig. 3 (continued)



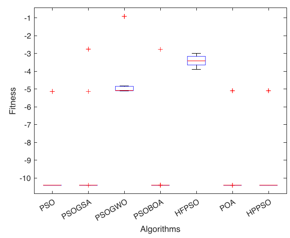
(s) F19



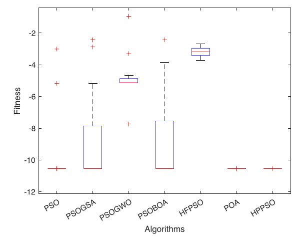
(t) F20



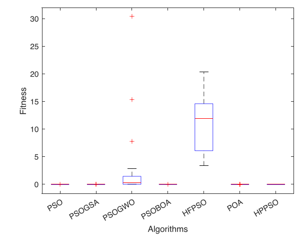
(u) F21



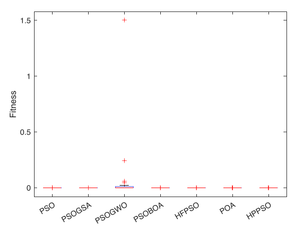
(v) F22



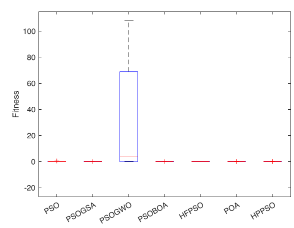
(w) F23



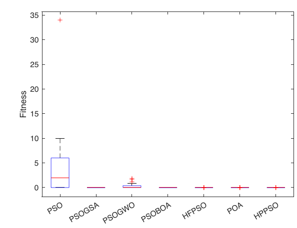
(x) F24



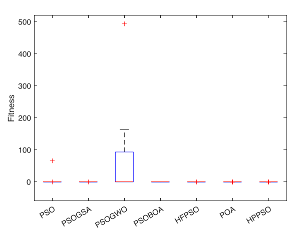
(y) F25



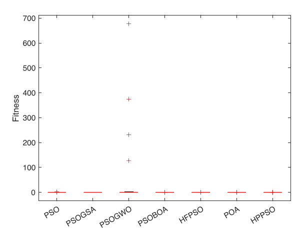
(z) F26



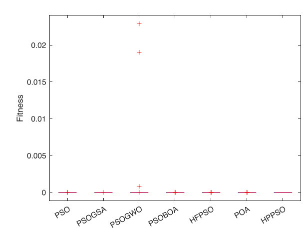
(aa) F27



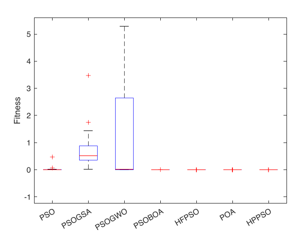
(ab) F28



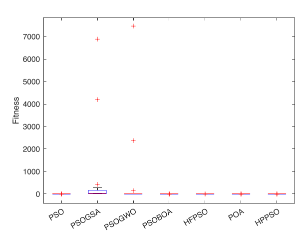
(ac) F29



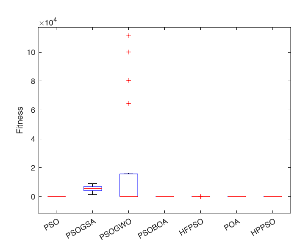
(ad) F30



(ae) F31



(af) F32



(ag) F33

Moreover, the convergence curves for the 33 benchmark functions optimized by the proposed HPPSO algorithm and other algorithms are depicted in Fig. 2. These curves illustrate that the HPPSO algorithm exhibits superior convergence capabilities. This fast convergence suggests that the algorithm has the ability to find optimal solutions quickly. Specifically, in the case of unimodal functions (F1-F7), the HPPSO algorithm consistently identifies better solutions and converges steadily. In contrast, the conventional PSO and POA algorithm tend to get stuck in local optima and are outperformed by the HPPSO algorithm. Even for F24-F33 functions, the convergence curves evident that the HPPSO algorithm frequently identifies superior solutions and converges rapidly. However, the convergence curves of HPPSO are not effective for multidimensional functions.

3.3 Results analysis by Wilcoxon signed rank test

In section 4.2, the performance of the HPPSO algorithm was evaluated in terms of mean and standard deviation on 33 benchmark mathematical functions. However, these values are not enough to validate the results obtained by the algorithms. A Wilcoxon signed-rank test (Wilcoxon, 1992) is used to validate the results by using the mean values obtained for 33 standard benchmark functions. The test is used to identify the statistically significant differences between the paired algorithms in terms of their mean value instead of ranking to the algorithm's performance.

In this test, the probability value (p-value) is used to determine if there are any significant differences between the results or not. The lower p-value indicates a greater level of significance and stronger evidence for rejecting the null hypothesis, implying that the performance of the two algorithms being compared has a statistically significant difference. The results of the Wilcoxon signed-rank test are presented in Table 4 and suggest that the performance of the proposed HPPSO algorithm is significantly different from other existing algorithms used in this study at a 5% level of significance.

3.4 Boxplot analysis

The boxplot analysis is used for evaluating and comparing the performance of the algorithms. This graphical approach facilitated the visualization of key statistics, including minimum and maximum data points (whisker edges) and the interquartile range (box width). By employing this analysis, the study is able to effectively assess the dispersion, central tendency, and data agreement characteristics of the algorithms. The boxplots of all compared algorithms for 33 benchmark mathematical functions are presented in Fig. 3. The results of this analysis highlight the consistently superior performance of the HPPSO algorithm in comparison to the other algorithms.

4 Conclusion and future scope

In the present paper, a novel Hybrid Pelican-Particle Swarm Optimization (HPPSO) algorithm has been presented using PSO and POA to efficiently solve complex optimization problems. HPPSO uses the good exploration capability of POA to overcome the stagnation effect of PSO. This is achieved by updating particles in exploitation phase of PSO obtained through exploration phase of POA. The performance of HPPSO has been tested on 33 benchmark mathematical functions and compared it with conventional PSO and POA along with other numerous hybridized algorithms of PSO (PSOGSA, HFPSO, PSOBOA and PSOGWO). The statistical analysis of HPPSO algorithm has been carried out through convergence curves, boxplot and a non-parametric wilcoxon signed rank test. These analyses indicate that HPPSO performs better than other algorithms in terms of achieving better optima. So, HPPSO is an effective algorithm that can handle complex optimization problems and avoid local optima. Thus, it is a promising choice for optimization problems.

However, it's important to acknowledge that the HPPSO algorithm introduced in this paper has certain limitations, one of which is its inability to effectively optimize multimodal functions (F10, F12-F14). In future, several research opportunities can be explored to improve the proposed HPPSO algorithm and address the above mention limitations. For instance, the robustness and accuracy of the proposed modification could be tested on different engineering, combinatorial optimization and real-world problems.

Author contributions Amit Raj contributed to methodology development, manuscript writing, and results compilation. Parul Punia contributed to the preparation of the figures and tables used in the manuscript. Pawan Kumar contributed towards the data analysis, supervision, writing review, and editing of the manuscript. All the authors read and approved the final manuscript.

Funding The first author is extremely grateful to the Council of Scientific & Industrial Research (CSIR) for providing Junior Research Fellowship (JRF) with file number 09/1152(0024)/2020-EMR-I and encouragement, which made this research possible.

Declarations

Conflict of interest There are no Conflict of interest to disclose by the authors in relation to the current study.

Ethical approval The author did not conduct any studies involving human participants or animals for this article.

Informed consent All the authors have approved the manuscript and agree with its submission to the International Journal of System Assurance Engineering and Management.

References

Abdulai JD, Adu-Manu KS, Katsriku FA, Engmann F (2023) A modified distance-based energy-aware (mDBEA) routing protocol in wireless sensor networks (WSNs). J Amb Intell Human Comput 14(8):10195–10217

Alzubi OA, Alzubi JA, Alazab M, Alrabea A, Awajan A, Qiqieh I (2022) Optimized machine learning-based intrusion detection system for fog and edge computing environment. Electronics 11(19):3007

Alzubi OA, Alzubi JA, Al-Zoubi AM, Hassonah MA, Kose U (2022) An efficient malware detection approach with feature weighting based on harris hawks optimization. Cluster Comput 25(4):2369–2387

Alzubi OA, Alzubi JA, Alzubi TM, Singh A (2023) Quantum mayfly optimization with encoder-decoder driven LSTM networks for malware detection and classification model. Mobile Netw Appl 28(2):795–807

Arora S, Singh S (2019) Butterfly optimization algorithm: a novel approach for global optimization. Soft Comput 23(3):715–734

Avvari RK, Vinod Kumar DM (2022) Multi-objective optimal power flow with efficient constraint handling using hybrid decomposition and local dominance method. J Inst Eng (India): Ser B 103(5):1643–1658

Aydilek IB (2018) A hybrid firefly and particle swarm optimization algorithm for computationally expensive numerical problems. Appl Soft Comput 66:232–249

Bansal S (2021) Nature-inspired hybrid multi-objective optimization algorithms in search of near-ogrs to eliminate fwm noise signals in optical wdm systems and their performance comparison. J Inst Eng (India): Ser B 102(4):743–769

Bansal S, Singh AK, Gupta N (2017) Optimal golomb ruler sequences generation for optical WDM systems: a novel parallel hybrid multi-objective bat algorithm. J Inst Eng (India): Ser B 98(1):43–64

Boyineni S, Kavitha K, Sreenivasulu M (2024) Rapidly-exploring random tree-based obstacle-aware mobile sink trajectory for data collection in wireless sensor networks. J Amb Intell Human Comput 15(1):607–621

Chakrabarti A, Chakrabarty K (2019) A proposal to adjust the timekeeping systems for savings in cycling operation and carbon emission. J Inst Eng (India) Ser B 100(6):541–550

Chopra N, Kumar G, Mehta S (2016) Hybrid GWO-PSO algorithm for solving convex economic load dispatch problem. Int J Res Adv Technol 4(6):37–41

Danandeh Mehr A, Rikhtehgar Ghiasi A, Yaseen ZM, Sorman AU, Abualigah L (2023) A novel intelligent deep learning predictive model for meteorological drought forecasting. J Amb Intell Human Comput 14(8):10441–10455

Dao TC, Tam NT, Quy NQ, Binh HTT (2023) An energy-efficient scheme for maximizing data aggregation tree lifetime in wireless sensor network. J Amb Intell Human Comput 14(9):12329–12344

Dash D (2023) Plane sweep algorithms for data collection for energy harvesting wireless sensor networks using mobile sink. J Amb Intell Human Comput 14(5):6307–6320

Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm Evolut Computat 1(1):3–18

Digalakis JG, Margaritis KG (2001) On benchmarking functions for genetic algorithms. Int J Comput Math 77(4):481–506

Dong X, Deng S, Wang D (2022) A short-term power load forecasting method based on k-means and SVM. J Amb Intell Human Comput 13(11):5253–5267

Faramarzi A, Heidarinejad M, Mirjalili S, Gandomi AH (2020) Marine predators algorithm: A nature-inspired metaheuristic. Expert Syst Appl 152:113377

Gad AG (2022) Particle swarm optimization algorithm and its applications: a systematic review. Arch Computat Methods Eng 29(5):2531–2561

Garg H (2014) Solving structural engineering design optimization problems using an artificial bee colony algorithm. J Ind Manag Opt 10(3):777–794

Gokulraj J, Senthilkumar J, Suresh Y, Mohanraj V (2021) Data prediction approaches for efficient data transmission using optimized Leibler distance matrix-based data aggregation in wireless sensor network. J Amb Intell Human Comput 3(26):1–8

Goldberg DE (1989) Genetic algorithms in search. Optimization and Machine Learning, Addison Wesley, Reading

Gupta S, Garg R, Singh A (2020) ANFIS-based control of multi-objective grid connected inverter and energy management. J Inst Eng Ser B 101(1):1–14

Hallafi A, Barati A, Barati H (2023) A distributed energy-efficient coverage holes detection and recovery method in wireless sensor networks using the grasshopper optimization algorithm. J Amb Intell Human Comput 14(10):13697–13711

Hariharan U, Rajkumar K, Akilan T, Ponmalar A (2023) A multi-hop protocol using advanced multi-hop Dijkstras algorithm and tree based remote vector for wireless sensor network. J Amb Intell Human Comput 14:6877–6895

Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73

Houssein EH, Gad AG, Hussain K, Suganthan PN (2021) Major advances in particle swarm optimization: theory, analysis, and application. Swarm Evolut Computat 63:100868

Jain D, Shukla PK, Varma S (2023) Energy efficient architecture for mitigating the hot-spot problem in wireless sensor networks. J Amb Intell Human Comput 14(8):10587–10604

Kennedy J, Eberhart R (1995) Particle swarm optimization. Proc Int Conf Neural Netw 4:1942–1948

Khalifa B, Al Aghbari Z, Khedr AM (2023) CAPP: coverage aware topology adaptive path planning algorithm for data collection in wireless sensor networks. J Amb Intell Human Comput 14(4):4537–4549

Kim J, Moon N (2019) BiLSTM model based on multivariate time series data in multiple field for forecasting trading area. J Amb Intell Human Comput 13:1–10

Laskar NM, Guha K, Chatterjee I, Chanda S, Baishnab KL, Paul PK (2019) HWPSO: a new hybrid whale-particle swarm optimization algorithm and its application in electronic design optimization problems. Appl Intell 49(1):265–291

Mirjalili S, Hashim SZM (2010) A new hybrid PSOGSA algorithm for function optimization. In: proceeding of ICCIA'10- international conference on computer and information application, pp 374-377

Molga M, Smutnicki C (2005) Test functions for optimization needs. Test Funct Optimizat Needs 101(48):01–43

Movassagh AA, Alzubi JA, Gheisari M, Rahimi M, Mohan S, Abbasi AA, Nabipour N (2021) Artificial neural networks training algorithm integrating invasive weed optimization with differential evolutionary model. J Amb Intell Human Comput 14:6017–6025

Murali P, Revathy R, Balamurali S, Tayade AS (2020) Integration of RNN with GARCH refined by whale optimization algorithm for yield forecasting: a hybrid machine learning approach. J Amb Intell Human Comput 11:1–13

Muruganantham B, Gnanadass R (2021) Solar integrated time series load flow analysis for practical distribution system. J Inst Eng Ser B 102(4):829–841

Navin Dhinnesh ADC, Sabapathi T (2022) Probabilistic neural network based efficient bandwidth allocation in wireless sensor networks. J Amb Intell Human Comput 13(4):2001–2012

Nayak JR, Shaw B, Sahu BK (2023) A fuzzy adaptive symbiotic organism search based hybrid wavelet transform-extreme learning machine model for load forecasting of power system: a case study. J Amb Intell Human Comput 14(8):10833–10847

Padhi S, Panigrahi BP, Dash D (2020) Solving dynamic economic emission dispatch problem with uncertainty of wind and load using whale optimization algorithm. J Inst Eng (India) Ser B 101(1):65–78

Raja J, Mookhambika N (2022) A novel energy harvesting with middle-order weighted probability (EHMoWP) for performance improvement in wireless sensor network (WSN). J Amb Intell Human Comput 13(12):5465–5476

Ram SDK, Srivastava S, Mishra KK (2022) A multi-objective generalized teacher-learning-based-optimization algorithm. J Inst Eng (India) Ser B 103(5):1415–1430

Rao SS (2019) Engineering optimization: theory and practice. Wiley, NJ, p 10

Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. Information Sci 179(13):2232–2248

Roy SS, Samui P, Nagtode I, Jain H, Shivaramakrishnan V, Mohammadi-Ivatloo B (2020) Forecasting heating and cooling loads of buildings: a comparative performance analysis. J Amb Intell Human Comput 11:1253–1264

Saranraj G, Selvamani K, Malathi P (2022) A novel data aggregation using multi objective based male lion optimization algorithm (DA-MOMLOA) in wireless sensor network. J Amb Intell Human Comput 13(12):5645–5653

Sengar S, Liu X (2020) Ensemble approach for short term load forecasting in wind energy system using hybrid algorithm. J Amb Intell Human Comput 11(11):5297–5314

Sethuraman J, Alzubi JA, Manikandan R, Gheisari M, Kumar A (2019) Eccentric methodology with optimization to unearth hidden facts of search engine result pages. Recent Patents Comput Sci 12(2):110–119

Sharma A, Sharma A, Panigrahi BK, Kiran D, Kumar R (2016) Ageist spider monkey optimization algorithm. Swarm Evolut Computat 100(28):58–77

Shinde AS, Bichkar RS (2023) Energy efficient active/sleep scheduling of sensor nodes in target based WSN using genetic algorithm with dither creeping mutation. J Amb Intell Human Comput 14(6):7649–7662

Singh U, Rizwan M (2023) Analysis of wind turbine dataset and machine learning based forecasting in SCADA-system. J Amb Intell Human Comput 14(6):8035–8044

Srinivas M, Amgoth T (2023) Delay-tolerant charging scheduling by multiple mobile chargers in wireless sensor network using hybrid GSFO. J Amb Intell Human Comput 14(12):16063–16079

Storn R, Price K (1997) Differential evolution-a simple and efficient heuristic for global optimization over continuous spaces. J Global Opt 11(4):341–359

Thalagondapati V, Singh MP (2023) A self-organized priority-based MAC protocol in wireless sensor networks based on SDR-RHSO optimal relay node selection and HL-ANN wake-up scheduling. J Amb Intell Human Comput 14(8):11093–11102

Trojovsky P, Dehghani M (2022) Pelican optimization algorithm: a novel nature-inspired algorithm for engineering applications. Sensors 22(3):855

Vahabi S, Mojab SP, Eslaminejad M, Dashti SE (2022) Eam: energy aware method for chain-based routing in wireless sensor network. J Amb Intell Human Comput 13(9):4265–4277

Vasanthi G, Prabakaran N (2023) Reliable network lifetime and energy-aware routing protocol for wireless sensor network using hybrid particle swarm-flower pollination search algorithm. J Amb Intell Human Comput 14(12):16183–16193

Verma RK, Jain S (2023) Energy and delay efficient data acquisition in wireless sensor networks by selecting optimal visiting points for mobile sink. J Amb Intell Human Comput 14(9):11671–11684

Wang X, Wang Y, Peng J, Zhang Z (2023) Multivariate long sequence time-series forecasting using dynamic graph learning. J Amb Intell Human Comput 14(6):7679–7693

Wilcoxon F (1945) Individual comparisons by ranking methods. Biometrics 1(6):80–83

Zhang C, Wang HP (1993) Mixed-discrete nonlinear optimization with simulated annealing. Eng Opt 21(4):277–291

Zhang M, Long D, Qin T, Yang J (2020) A chaotic hybrid butterfly optimization algorithm with particle swarm optimization for high-dimensional optimization problems. Symmetry 12(11):1800