Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/200624

HJSRSO: A Hybrid Algorithm Jellyfish Search and Rat Swarm Optimizer for Engineering Design Problems

A. M. Rashad[1], *Abeer A. Ahmed*[1], *Amr Elsonbaty*[2], *M. A. El-Shorbagy*[2,*], *Moutaz Ramadan*[2], *and Hamed A. EL-Mky*[1]

[1]Department of mathematics, Faculty of Science, Aswan University, Aswan 81528, Egypt
[2]Department of Mathematics, College of Science and Humanities in Al-Kharj, Prince Sattam bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia

Received: 2 Sep. 2026, Revised: 17 Oct. 2026, Accepted: 27 Oct. 2026
Published online: 1 Nov. 2026

Abstract: The study employs a novel technique called HJSRSO, which combines the jellyfish search optimizer (JS) with the rat swarm optimizer (RSO). The Jellyfish Search Optimization (JS) architecture serves as the foundation for the suggested hybrid HJSRSO method. However, the passive motion mechanism (Type A) used in JS is replaced with the Rat Swarm Optimizer (RSO) to successfully enhance the algorithm's exploitation capabilities. Additionally, a sine-based technique is incorporated into the Jellyfish Search algorithm's exploration phase to promote exploring behavior. This technique allows agents to follow oscillatory trajectories that boost population diversity and make it easier for the algorithm to escape local optima. Seven constrained engineering design problems (EDPs) and fifty common benchmark functions were used to thoroughly analyze the performance of the suggested HJSRSO algorithm. Rat Swarm Optimizer (RSO), the original Jellyfish Search (JS), and several contemporary cutting-edge metaheuristic algorithms were thoroughly compared. Each algorithm was independently run 30 times under the same experimental conditions for statistical reliability and repeatability. The significance of the suggested modifications to the original JS is demonstrated by the finding that, out of the 50-benchmark functions, HJSRSO obtained the best results in 28 functions with a percentage of 65%, whereas the JS obtained the best solutions in only 18 functions with a percentage of 36%. Key performance metrics, such as the best, worst, mean, median, and standard deviation of the objective function values, were calculated. Non-parametric statistical tests, such as the Friedman rank-based test and the Wilcoxon signed-rank test, were used to quantitatively assess the statistical significance of the observed performance differences. These analyses' findings verify that HJSRSO regularly achieves better optimal solutions than the original and rival algorithms, demonstrating statistically significant superiority in terms of solution quality, convergence stability, and robustness.

Keywords: Hybrid metaheuristic optimization; Jellyfish Search algorithm; Rat Swarm Optimization; Engineering design optimization; Benchmark functions; Statistical performance analysis

1 Introduction

Engineering design problems (EDPs) are regarded as a vital and complex research topic from both industrial and scientific viewpoints, particularly in engineering applications, where the objective is to obtain designs with improved accuracy, efficiency, and performance. EDPs are typically regarded as nonlinear constrained optimization problems (NCOPs). Solving NCOPs is frequently extremely challenging due to the complexity of constraints, and the issue feasible region may be a small portion of the search domain. It is impossible to pinpoint the global optimum to the NCOP [1].

Among the methods that have historically been employed to solve NCOPs are recursive quadratic programming, the projection approach, the extended reduced gradient method, the penalty method, and a multiplier method [2]. These methods necessitate that the objective function be differentiable and the feasible set be convex. Conventional methods are also inefficient since they may only compute local optima. It is therefore very challenging to apply these methods to problems where the feasible region is not convex or the goal function is not differentiable. Additionally, we find that most EDPs are discontinuity problems with a noise-filled search space, which prevents traditional methods from finding global solutions for this kind of problem. In general, we may say that conventional optimization

* Corresponding author e-mail: mohammed_shorbagy@yahoo.com

© 2026 NSP
Natural Sciences Publishing Cor.

methods depend on the existence of the derivative, which has a narrow scope, and are not robust enough in discontinuous, noisy, and multimodal search spaces [3].

Some optimization techniques have emerged that are categorized as unconventional or metaheuristic algorithms that are entirely different from the usual techniques. They are regarded as widely used techniques for resolving NCOPs. These techniques mimicked the traits and actions of biological, molecular, swarm, and neural systems. Additionally, they are able to overcome the limitations and challenges of traditional methods. Additionally, neither the objective function nor the constraints need to be continuous or derivable; these metaheuristic algorithms are conceptually simple, have a low probability of entering local optima, and require basic information about the optimization problem [4].To solve EDPs, researchers have turned to simulation-based metaheuristic algorithms. These algorithms' primary characteristic is their capacity to combine rules and randomness to replicate natural phenomena. Among the well-known metaheuristic algorithms currently in use are simulated annealing (SA) [5], differential evolution (DE) [6], beluga whale optimization (BWO) [7,8], genetic algorithms (GA) [9], jellyfish search optimizer (JS) [10], particle swarm optimization (PSO) [11], rat swarm optimizer (RSO) [12], ant colony optimization (ACO) [13], firefly algorithm (FA) [14], whale optimization algorithm (WOA) [15], sine cosine algorithm (SCA) [16], salp swarm algorithm [17], grey wolf optimizer (GWO) [18], equilibrium optimizer [19], monarch butterfly optimization [20], slime mould algorithm [21], moth-flame search algorithm (MFO) [22], and Harris hawks optimization (HHO) [23].

Metaheuristic algorithms may encounter problems such as getting stuck in a local optimum, a lack of diversity in solutions, and an imbalance between exploitation and exploration trends in some complex scenarios, despite the fact that they perform better than traditional methods. The majority of researchers have proposed hybridization strategies between metaheuristic algorithms to address these drawbacks, such as hybrid GA [24], hybrid ACO [25], hybrid PSO [26], hybrid GOA [27], hybrid FA [28], hybrid SCA [29], hybrid JS [30], hybrid BWO [30,31], etc. These hybrid approaches also have a number of advantages, including tackling large optimization problems and offering robust algorithms with faster performance. They are also among the most intriguing current optimization trends.

Jui-Sheng Chou and Dinh-Nhat Truong introduced a novel swarm-based method in 2021 [32] called the jellyfish search (JS) algorithm. It basically imitates how ocean jellyfish look for food. Because of the JS algorithm's excellent global hunting skills, high robustness, small number of parameters, and other positive aspects, we have conducted an increasing amount of in-depth research on this topic. The JS algorithm allows jellyfish to move in two ways: (1) inside the group and (2) in the ocean current. The time control notion states that jellyfish may alternate between several movement modes, which can enhance the optimization performance of the JS method. Because of its substantial superiority, this approach has been utilized to examine and solve a number of real-world engineering challenges. The JS optimizer method was used in [33] to address the economic load dispatch (ELD) in the electric power system. Gouda et al. [34] employed the JS approach to find undiscovered parameters in a previous PEM fuel cell. Youssef et al. [35] used the JS approach to predict the characteristics of a single-phase transformer. The Jellyfish Search (JS) algorithm was successfully used to solve the optimal voltage coordination problem in automated distribution systems [36], proving its relevance to power distribution optimization. Since then, studies have been conducted on the combination of quasi-reflected learning with multi-objective JS techniques [37]. Other researchers have refined the jellyfish search method [38,39,40,41].

In order to attain better solution quality for complicated engineering design problems (EDPs), this work suggests a hybrid optimization technique called HJSRSO, which combines the Jellyfish Search (JS) and Rat Swarm Optimization (RSO) algorithms. One of the most recent metaheuristic techniques is the Rat Swarm Optimizer (RSO), which is especially good at exploitation by directing agents toward promising areas of the search space to refine solutions through intensive local search. RSO is inspired by the collective foraging, pursuing, and attacking behaviors of rat populations [42]. The Jellyfish Search (JS) method, on the other hand, demonstrates high global exploration capabilities due to stochastic motions and ocean current-inspired movements that enable jellyfish agents to cover a large portion of the search field. A sine-based strategy is added to the JS exploration phase to further improve exploration. This allows agents to follow oscillatory trajectories that boost population diversity and make it easier for the algorithm to escape local optima. The hybrid HJSRSO strikes a balance between exploration and exploitation by fusing JS's sine-enhanced global search with RSO's intensive local exploitation. This allows for the preservation of diversity and coverage of several search space regions while speeding up convergence and increasing solution accuracy in subsequent optimization stages. This synergy produces a strong optimization framework that can effectively produce high-quality solutions for difficult EDPs.

The following succinctly describes this study's primary contributions:

1. The creation of HJSRSO, a hybrid algorithm intended to address challenging engineering design problems (EDPs).
2. Integration of Rat Swarm Optimizer (RSO) and Jellyfish Search (JS) into HJSRSO to combine RSO's intensive local exploitation capacity with JS's global exploration strength.
3. Using a sine-based approach that modifies the jellyfish agents' movement amplitude throughout the exploration stage, allowing for adaptive oscillatory motions to improve the exploration–exploitation balance and increase global search efficiency.

© 2026 NSP
Natural Sciences Publishing Cor.

4.Comprehensive testing of HJSRSO on standard benchmark problems as well as multiple real-world engineering design problems.
5.Application of rigorous statistical analyses, including the Wilcoxon signed-rank test and Friedman test, to validate the significance and reliability of the performance improvements achieved by HJSRSO.

This is how the rest of the paper is structured. The basic information about the issue at hand is given in Section 2. The suggested methodology is explained in detail in Section 3. Section 4 discusses the computational experiments and provides an in-depth discussion of the outcomes. The study is finally concluded in Section 5, which also suggests possible directions for further research.

2 Preliminaries

This section discusses how the EDPs are formulated. EDPs are frequently expressed as NCOPs, as was previously mentioned. NCOP's mathematical model is as follows:

$$\begin{aligned} &\text{Minimize (or Maximize) } f(X) \\ &\text{Subject to:} \\ &h_i(X) \le 0, \quad i = 1,\ldots,p, \qquad (1) \\ &g_j(X) = 0, \quad j = 1,\ldots,m, \qquad (2) \\ &L_k \le x_k \le U_k, \quad \forall k = 1,\ldots,n. \qquad (3) \end{aligned}$$

Where f, $h_1, \ldots, h_p$, $g_1, \ldots, g_m$ are functions defined on $\mathbb{R}_n$, $X = (x_1, \ldots, x_n)$ is a vector of n component subset of $\mathbb{R}_n$. The above problem must be solved for the values of the variables $x_1, \ldots, x_n$ that satisfy the constraints and minimize or (maximize) the function f. The function f is the objective function or the criterion function. If there are no constraints, the problem is called an unconstrained problem. Each of the constraints $h_i(X) \le 0$ for $i = 1, \ldots, p$ is called an inequality constraint, and each of the constraints $g_j(X) = 0$ for , $j = 1, \ldots, m$ is called an equality constraint. L_k represents the lower bounds and U_k the upper bounds for the decision variables x_k, $\forall k = 1, \ldots, n$.

While solving NCOP, we search for a global solution rather than relying on a local one. The optimal solution (maximal or minimum) within a close set of potential solutions is known as the local solution of an optimization problem. However, the global optimum is the best choice among all possible solutions, not only those found within a certain search region zone [43]. Definition 1 explains the differences between a local and global solution.

Definition 1: Let $X = (x_1, x_2, \ldots, x_n)$ be a feasible solution to a minimization problem with objective function $f(X)$ [43]. Then, X is

1. A global minimum if $f(X) \le f(Y)$ for every feasible point $Y = (y_1, y_2, \ldots, y_n)$.

2. A local minimum if $f(X) \le f(Y)$ for all feasible points $Y = (y_1, y_2, \ldots, y_n)$ sufficiently close to X.

3 Methodology

We give a quick synopsis of the JS and the RSO in this section. Next, a detailed description of the suggested algorithm is given.

3.1 Jellyfish search algorithm (JS)

The JS algorithm's primary source of inspiration is the way jellyfish look for food sources in the water, taking into account their intelligent movement patterns while determining the optimal position of food [32]. In JS modeling, three rules are used:

i. The time control device may move jellyfish in two different ways: either by moving with the swarm or by following the ocean current.

ii. When looking for food in the ocean, jellyfish gravitate toward the most nutrient-rich areas.

iii. The quantity and position of food that jellyfish find match the problem's objective function and its matching solution, respectively.

(a)Ocean current

The ocean current attracts jellyfish due to its abundance of food sources. In this scenario, the jellyfish use the following formula to update their positions:

$$\begin{aligned} X_i^{t+1} = X_i^t + r_1 \mu \left(X_{\text{best}}^t - \beta r_2\right), \\ i = 1,2,\ldots,N. \end{aligned} \qquad (4)$$

At iterations t and $(t+1)$, respectively, the jellyfish locations are denoted by X_i^t *and* X_i^{t+1}. The random numbers r_1 and r_2 fall inside the interval [0, 1]. X_{best}^t is The swarm's optimal global location at the moment. μ is the average position of every jellyfish, while the coefficient for distribution is denoted by β. There are N jellyfish. With β = 3, the ocean current direction and β are determined in [32].

(b)Jellyfish swarm

Jellyfish exhibit both passive (type A) and active (type B) movements in swarms. Jellyfish display the passive motion type (A) when the swarm first forms. They start to show more and more type (B) motion over time. Jellyfish move in type (A) motion around their own places, and each jellyfish's consistently updated location is determined by:

$$X_i^{t+1} = X_i^t + r_3 \times \gamma \times (ub - lb) \qquad (5)$$

Where the motion coefficient, γ=0.1, is related to the length of motion around the jellyfish sites, and *ub* and *lb* are the upper and lower bounds of search spaces, respectively.

To mimic type B motion, a jellyfish (j) that isn't the current jellyfish (i) is chosen at random, and the step and direction of the movement are determined by creating a distance vector between the two jellyfish. When the location of

© 2026 NSP
Natural Sciences Publishing Cor.

the selected jellyfish (j) has more food than the location of the jellyfish of interest (i), the latter moves in the direction of the former; if the amount of food available to the chosen jellyfish (j) is less than that of the jellyfish of interest (i), it moves straight away from the former. In order to locate the food in a swarm, each jellyfish moves in a better direction [32]. Since f is an objective function of position X in this case, the following equation is used to carry out the active motions (type B):

$$X_i^{t+1} = X_i^t + \overrightarrow{\text{step}} \tag{6}$$

Where

$$\overrightarrow{\text{step}} = r_4 \times \overrightarrow{\text{direction}} \tag{7}$$

$$\overrightarrow{\text{direction}} = \begin{cases} X_j^t - X_i^t, & \text{if } f(X_i) \geq f(X_j) \\ X_i^t - X_j^t, & \text{if } f(X_i) < f(X_j) \end{cases} \tag{8}$$

To alternate between the passive and active forms of motion, a time control mechanism is used as a control parameter. Additionally, it may be used to control jellyfish migration in the direction of the ocean current.

(c)Time control mechanism

A new jellyfish swarm is formed when the jellyfish relocate to a different ocean current at each degree of wind or temperature change. At the start of the search phase, the jellyfish in this swarm choose passive motion type A. The jellyfish gradually start to favor active motion (type B). By controlling the movement and alternating between jellyfish moving in a swarm and following the ocean current, a temporal control method is provided to mimic this behavior. A time control function that randomly varies over time between 0 and 1 is used to simulate this time control mechanism. The time control function is expressed as follows in equation (9):

$$C^t = \left(1 - \frac{t}{t_{\max}}\right) \times (2 \times r_5 - 1) \tag{9}$$

Where r_5 is a random number in the interval [0, 1] and $t_{\max}$ is the maximum number of iterations.

(d)Initialization of population

The starting jellyfish population is created at random. A logistic map provides more diverse starting populations than by random selection and a reduced chance of early convergence, is created in order to achieve more population diversity [32]. Therefore, based on the following, the logistic map may be applied to the initial population:

$$X_{i+1} = \eta X_i (1 - X_i), \quad 0 \leq X_0 \leq 1. \tag{10}$$

When X_0 is used to generate the starting jellyfish population, X_i is the chaotic logistic value of the i-th jellyfish's position, and η is set to 4.

(e)Restriction of boundaries

Jellyfish may travel outside the search space's bounds during swarm movement; in this scenario, the position is updated by going back to the opposite bound, using Equation (11):

$$X'_{i,d} = \begin{cases} (X_{i,d} - ub_d) + lb_d, & \text{if } X_{i,d} > ub_d \\ (X_{i,d} - lb_d) + ub_d, & \text{if } X_{i,d} < lb_d \end{cases} \tag{11}$$

Where $X_{i,d}$ is the position of the i-th jellyfish in the d-th dimension; $X'_{i,d}$ is the updated location following boundary constraint verification; and ub_d and lb_d are the upper and lower bounds of the d-th dimension in search spaces, respectively.

3.2 Rat swarm optimizer (RSO)

Rat Swarm Optimization (RSO), a novel metaheuristic algorithm, was inspired by the chasing and fighting behaviors of rats [42]. Rats are local creatures that coexist in swarms with both males and females. Rats frequently exhibit extremely aggressive behavior, which may cause some animals to die. This algorithm uses a mathematical model to simulate the chasing and fighting behaviors of rats [42]. The rat swarm optimizer begins with a set of random solutions that represent the rat's position in the search space, just like the other population-based optimization techniques. This random set is continuously assessed by an objective function and refined based on the chasing and fighting behaviors of rats. The beginning locations of viable solutions (rats' positions) in the search space are chosen at random in the original RSO approach as follows:

$$X_i = lb_i + r_6 \times (ub_i - lb_i), \quad i = 1, 2, \ldots, N \tag{12}$$

Where lb_i and ub_i represent the lower and upper bounds of the ith variable, correspondingly. Rats typically pursue the bait in groups by engaging in unpleasant social behaviors. In order to mathematically explain this rat performance, The most skilled search agent is thought to know where to place the bait. Consequently, the other search agents can share where they stand in relation to the top search agent. that has been found thus far. The following formula has been proposed to show how rats use bait to attack and generate the rats' most recent position [42]:

$$X_i^{t+1} = \left| X_{best}^t - \left(A \times X_i^t + C \times (X_{best}^t - X_i^t)\right) \right| \tag{13}$$

Where

$$A = R - t\left(\frac{R}{t_{\max}}\right), \quad C = 2 \times r_7 \tag{14}$$

© 2026 NSP
Natural Sciences Publishing Cor.

In this case, Random numbers between [1,5] and [0,2] are denoted by R and C, respectively [42]. $t_{\max}$ is the most iterations that can occur, while t is the optimization process's current iteration. Equation (10) preserves the optimal answer and modifies the search agents' placements.

3.3 The suggested strategy

3.3.1 The sine-based approach

A sine strategy's integration into an optimization algorithm's exploration phase is essential for increasing search diversity and boosting global exploration effectiveness. The movement of search agents becomes both stochastic and oscillatory when randomness is incorporated into the sine function, resulting in adaptive location updates that change smoothly in both amplitude and direction. Instead of following linear or entirely random trajectories, this method encourages exploration of both close and distant locations by enabling agents to explore the search space in a no uniform manner. Because of this, the population remains more diverse, which is crucial for identifying viable areas and preventing early stagnation.

Furthermore, exploration steps are kept under control by the bounded output of the sine function [-1,1], which avoids unstable or too large motions that can interfere with convergence. Agents are encouraged to alternately approach and move away from the current optimum solution because to the random sine oscillations' changing attraction and repulsion effects. Especially in the early and mid-iterations, this behavior decreases premature convergence and raises the likelihood of escaping local optima. Overall, the sine method with randomized input strengthens the exploratory capabilities while maintaining convergence stability by offering a flexible balance between randomness and guidance.

The expression for the sine-based exploration update is:

$$X_i^{t+1} = X_i^t + w\sin(\varphi)|X_{\text{best}}^t - X_i^t|, \quad i = 1,2,\ldots,N. \quad (15)$$

Where $w = 2 - t\left(\frac{2}{t_{\max}}\right)$, *and* $\varphi = 2 \times \pi \times r_8$. In order to facilitate the shift from global exploration in early iterations to local exploitation in later iterations, w is a linearly decreasing scaling factor that decreases from 2 to 0 over the course of iterations. Whereas r_8 yields a uniform random number in (0,1), φ yields a random angle in [0, 2π]. Search agents can travel toward or significantly away from the optimal solution thanks to the oscillatory movement pattern introduced by the $sin(\varphi)$. In order to promote adaptive movement magnitudes, the absolute difference $|X_{best}^t - X_i^t|$ guarantees that the step size is proportionate to the distance between the search agent and the global best.

3.3.2 HJSRSO

(a)**The exploration of HJSRSO**

During the exploration phase, the method uses two complementing movement directions: sine-based oscillatory motion and ocean current motion. A random number (r_9) from the interval (0,1) is used to dynamically switch between the two movement ways. This random number was chosen to increase search diversity and prevent the exploration process from becoming biased toward a specific movement pattern.

The search agent to update its position uses the ocean current technique, which mimics large-scale directed flow toward promising areas of the search space, if $r_9 < C_0$ (where C_0 is equal to 0.5). This motion promotes long-distance travel, allowing search agents to quickly investigate uncharted territory and enhance global coverage. On the other hand, the agent employs the sine method, which introduces nonlinear oscillatory movements with adaptive step sizes, if $r_9 \geq C_0$. Search agents can avoid local optima by exploring the neighborhood surrounding intriguing solutions while retaining randomness thanks to the sine-based mobility.

The exploration update can be stated mathematically as follows:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1\mu\left(X_{\text{best}}^t - \beta r_2\right), & \text{if } r_9 < C_0, \\ X_i^t + w\sin(\varphi) \times \left|X_{\text{best}}^t - X_i^t\right|, & \text{if } r_9 \geq C_0. \end{cases} \quad (16)$$

Sine-based oscillations improve local diversification and robustness against premature convergence, while ocean currents encourage wide-ranging global exploration. This random switching between ocean current and sine techniques guarantees a balanced and adaptive exploration process.

(b)**The exploitation of HJSRSO**

The suggested approach significantly from JS during the exploitation phase by substituting the Rat Swarm Optimizer (RSO) for passive motion. The RSO mechanism adds a goal-oriented intensification method that directly directs individuals toward the global best solution, in contrast to passive motion, which depends on undirected and weakly informative local disturbances. By eliminating unnecessary random movements, increasing convergence accuracy, and facilitating the more efficient use of acquired swarm knowledge, this substitution significantly increases exploitation efficiency.

To ensure a reliable exploitation process, a time-control system dynamically switches between the active interaction motion and the RSO-based motion. Overall, compared to conventional passive–active swarm frameworks, the incorporation of RSO in place of passive motion greatly enhances the exploitation potential of the suggested algorithm, resulting in faster convergence speed, greater solution quality, and increased stability. The pseudo code for the HJSRSO is shown in Algorithm 1.

© 2026 NSP
Natural Sciences Publishing Cor.

3.3.3 The HJSRSO Algorithm's Time Complexity

An algorithm is a collection of computer methods for handling data and resolving optimization issues. While multiple algorithms may yield comparable solutions for the same issue, their computing costs and execution times can vary greatly. As a result, temporal complexity is frequently used to assess and contrast the effectiveness of optimization methods by measuring the number of operations needed during execution. The population size N, the problem dimension D, and the maximum number of iterations $t_{\max}$ are the primary determinants of the temporal complexity of the suggested Hybrid Jellyfish–Rat Swarm Optimization (HJSRSO) algorithm. HJSRSO carries out a number of tasks that strike a balance between exploration and exploitation during each iteration. To improve global search capabilities, candidate solutions are updated during the exploration phase by using a sine-based technique and the ocean current mechanism. The program intensifies local search around potential spots during the exploitation phase by using the rat swarm optimization strategy in conjunction with jellyfish active motion. For every member of the population, these operations entail vector-based calculations in all dimensions.

Algorithm 1 The Pseudocode of the HJSRSO Algorithm

Require: $t_{\max}$, lb, ub, N, and D
Ensure: Best solution X_{best}
Initialize the population using the Logistic map
Evaluate the fitness of all search agents
Determine the best solution X_{best}
while $t \leq t_{\max}$ **do**
  for $i = 1$ to N **do**
    Calculate the control parameter C^t
    if $C^t > 0.5$ **then**
      Perform the exploration stage
    else
      Perform the exploitation stage
    end if
    Evaluate the new position
  end for
  Update the best solution
  $t \leftarrow t + 1$
end while
return X_{best}

Based on the above procedures, the time complexity of HJSRSO can be expressed as follows:

$$\begin{aligned} O(\text{HJSRSO}) = O\Big(t_{\max}\Big[&O(\text{ocean-current search} \\ &+ \text{sine-based exploration}) \\ &+ O(\text{rat swarm optimization} \\ &+ \text{active motion})\Big]\Big) \end{aligned} \tag{17}$$

$$\begin{aligned} O(\text{HJSRSO}) &= O\big(t_{\max}\big[O(ND) + O(ND)\big]\big) \\ &= O(t_{\max} ND) \end{aligned} \tag{18}$$

Thus, the time complexity of the HJSRSO algorithm grows linearly with the population size, problem dimensionality, and number of iterations. This confirms that the hybridization strategy enhances the search performance without increasing the asymptotic computational complexity, making HJSRSO computationally efficient and suitable for solving large-scale and complex engineering optimization problems.

4 Computational Experiments

In this section, the performance of the proposed HJSRSO algorithm is investigated using a suite of 50 benchmark test functions to assess its effectiveness in global optimization tasks [44]. Furthermore, the applicability of HJSRSO to real-world problems is examined through seven constrained engineering design problems selected from the CEC'2020 test suite [45]. The proposed algorithm is comparatively evaluated against several metaheuristic methods, including the Jellyfish Search Optimizer (JS), Particle Swarm Optimization (PSO), Rat Swarm Optimizer (RSO), Whale Optimization Algorithm (WOA), Sine Cosine Algorithm (SCA), Grey Wolf Optimizer (GWO), Multi-Verse Optimizer (MVO), and the hybrid Jellyfish Search–Particle Swarm Optimization (HJSPSO).

All algorithms were implemented in MATLAB (R2014a) and executed on a Windows 7 platform equipped with an AMD E1-2100 APU processor operating at 1 GHz and 4 GB of RAM. The parameter settings of the comparative algorithms were adopted from their original studies, as summarized in Table 1. To ensure a fair comparison, each algorithm was independently executed 30 times using a population size of 50 and a maximum of 1000 iterations. The Wilcoxon signed-rank test was used to verify the significance of pairwise comparisons between the rival algorithms, and the nonparametric Friedman test was used for statistical analysis of the data.

4.1 A computational experiment involving fifty benchmark functions

The performance of the suggested optimization approach is thoroughly assessed in this study using 50 benchmark test functions. The optimization literature often uses this classification, which divides these functions evenly into 25 unimodal and 25 multimodal issues [45]. Unimodal functions are mostly used to evaluate the exploitation capacity, convergence accuracy, and stability of optimization algorithms because they have only one global optimum and no local optima. Multimodal functions, on the other hand,

© 2026 NSP
Natural Sciences Publishing Cor.

Table 1: The parameter values employed for HJSRSO and the other algorithms

Algorithm	**Parameters**
HJSRSO	$\gamma = 0.1, \beta = 3, c_0 = 0.5$, ranges of R and C are set to $[1,5]$ and $[0,2]$, respectively
JS	$\gamma = 0.1, \beta = 3, c_0 = 0.5$
HJSPSO	$c_{\min} = 0.5; c_{\max} = 2.5; w_{\min} = 0.4; w_{\max} = 0.9; \gamma = 0.1; \beta = 0.1; c_0 = 0.5$
GWO	control parameter a linearly decreases from 2 to 0
PSO	$c_1 = 0.5; c_2 = 2.5; w_{\min} = 0.4; w_{\max} = 0.9$
MVO	Wormhole existence probability (WEP) $\in [1, 0.2]$
WOA	α variable decreases linearly from 2 to 0, $b = 1$.
RSO	ranges of R and C are set to $[1,5]$ and $[0,2]$, respectively
SCA	a decreases from 2 to 0, r_1, r_2, r_3, r_4 are random numbers

have several local optima in addition to the global optimum, which makes them especially useful for assessing the resilience of algorithms in preventing premature convergence, exploration capacity, and global search efficiency. A balanced and rigorous assessment framework is obtained by combining both unimodal and multimodal benchmark functions, guaranteeing that the suggested algorithm is successfully tested in terms of both local refinement and global search performance.

A summary of the results, such as average, worst, best, median, standard deviation, and rank, may be seen in the table 2. The optimization methods that yield the best result (i.e., first rank) are indicated in bold in the table. In eight of the twenty-five unimodal benchmark functions, the suggested approach obtained the top rank, demonstrating its strong capacity for exploitation and dependable convergence to the global optimum. In comparison, the RSO got the greatest performance in 15 unimodal functions, whereas JS method reached the top rank in 7 functions. Although RSO slightly outperforms it in the number of unimodal functions, the proposed algorithm shows competitive convergence behavior and stability, indicating that it may effectively improve solutions. The suggested approach fared better than its rivals for the multimodal benchmark functions, obtaining the best results in 20 out of 25 functions, while RSO and JS achieved the top rank in 9 and 11 functions, respectively. This benefit demonstrates the higher exploration capabilities of the suggested strategy, which enables it to effectively traverse intricate search spaces with numerous local optima and prevent premature convergence. These findings, when combined with its competitive performance in unimodal functions, show that the suggested algorithm offers a well-rounded search strategy that excels in both accurate local exploitation and reliable global exploration, which is essential for handling a variety of optimization problems. Table 2 clearly shows that the capability of HJSRSO is superior to its competitors in solving 28 out of 50 benchmark test functions with the best solution, while the other algorithms result as follows: HJSPSO (19/50), JS (18/50), PSO (19/50), MVO (1/50), WOA (11/50), GWO (4/50), RSO (24/50), and SCA (1/50). As a result, the proposed hybrid algorithm can improve the functionality of the JS and RSO algorithms. The hybrid metaheuristic algorithms are used to capitalize on the advantages of the algorithms. It is possible to observe that the JS and RSO algorithms are capable of exploration and exploitation efficiently in hybridization.

4.1.1 The Friedman and Wilcoxon signed-rank nonparametric tests for fifty benchmark functions outcomes

The Friedman test is employed to statistically assess and rank the performance significance of the compared algorithms [46]. The corresponding outcomes are summarized in Table 3. According to this analysis, HJSRSO achieves the highest rank, and the differences between its performance and that of the other algorithms are statistically significant, as indicated by a P-value below the 0.05 significance level ($P = 0.000$). The ranking distribution of HJSRSO and the rival techniques is shown in Figure 1, where shorter bars represent better performance and longer bars imply worse outcomes. As can be seen, SCA has the highest mean rank (8.02), whereas HJSRSO has the lowest mean rank (2.73), indicating that it is the best algorithm out of all those assessed.

Additionally, the statistical significance of the performance differences between HJSRSO and the other techniques is further examined using the Wilcoxon signed-rank test [47]. Table 4 presents the test's results, with P-R and N-R standing for the sums of positive and negative ranks, respectively. HJSRSO consistently produces P-R values that are higher than the equivalent N-R values in every pairwise comparison, proving its superiority over the rival algorithms. All of these statistical results confirm that the suggested HJSRSO is a very efficient and competitive optimization technique.

4.1.2 Convergence Performance Analysis

To evaluate convergence behavior, 24 benchmark functions were utilized, comprising six unimodal (F5, F8, F16, F19, F20, F25) and eighteen multimodal (F26–F28, F32–F41, F44, F47–F50) functions. As shown in Fig. 2, HJSRSO consistently achieves faster convergence, indicating an enhanced trade-off between exploration and exploitation.

© 2026 NSP
Natural Sciences Publishing Cor.

Table 2: Evaluation results of HJSRSO alongside other optimization algorithms on fifty standard benchmark functions

		HJSRSO	JS	HJSPSO	GWO	PSO	MVO	WOA	RSO	SCA
F1	**STD**	1.47E-80	2.43E-35	7.30E-97	3.43E-51	1.41E-05	0.269308	0	**0**	30.207100
	MEAN	8.88E-81	1.31E-35	3.36E-97	2.38E-51	1.81E-05	1.3266187	3.52E-171	**0**	31.037326
	MEDIAN	5.23E-85	3.89E-36	9.07E-99	3.46E-52	1.88E-05	1.296460	2.06E-181	**0**	36.103318
	WORST	3.39E-80	5.65E-35	1.64E-96	7.98E-51	3.46E-05	1.688095	1.76E-170	**0**	72.867438
	BEST	1.96E-86	1.36E-37	1.71E-99	6.20E-53	2.28E-06	1.0730545	7.74E-186	**0**	0.3983747
	RANK	4	6	3	5	7	9	2	**1**	8
F2	**STD**	0.0001581	0.0002054	0.0002486	0.0003099	45.721307	0.0150089	0.0005736	**0.0001386**	0.1261083
	MEAN	0.0009888	0.0008106	0.000678	0.0009945	123.64537	0.0445644	0.0006710	**0.000127**	0.1769190
	MEDIAN	0.0009196	0.0008633	0.0006886	0.0009198	104.81034	0.03954	0.0004055	**0.0001208**	0.1690984
	WORST	0.0012679	0.0010895	0.0010503	0.0014048	182.69483	0.062694	0.0015936	**0.0003477**	0.3835244
	BEST	0.0008912	0.000551	0.0003610	0.0006350	78.030497	0.0299292	0.0001599	**7.68E-06**	0.0544406
	RANK	6	4	3	5	9	7	2	**1**	8
F3	**STD**	**0**	2.35E-113	**0**	**0**	4.53E-09	4.97E-08	**0**	**0**	3.32E-05
	MEAN	1.38E-214	1.25E-113	9.73E-246	8.52E-217	2.13E-09	1.42E-07	4.92E-260	**0**	1.51E-05
	MEDIAN	6.96E-221	6.57E-116	3.66E-246	1.65E-221	1.73E-10	1.75E-07	1.53E-264	**0**	1.49E-07
	WORST	6.91E-214	5.40E-113	4.06E-245	4.26E-216	1.02E-08	1.84E-07	2.46E-259	**0**	7.45E-05
	BEST	1.41E-228	4.90E-116	1.46E-250	8.10E-228	6.33E-13	8.33E-08	1.19E-284	**0**	2.57E-12
	RANK	4	6	3	5	7	9	2	**1**	8
F4	**STD**	7.72E-45	8.57E-19	3.99E-50	6.51E-30	0.0159625	1.7301942	1.58E-107	**0**	0.0412000
	MEAN	6.09E-45	1.66E-18	3.08E-50	8.60E-30	0.0258734	1.4745737	8.73E-108	**0**	0.0296005
	MEDIAN	4.63E-45	1.46E-18	2.02E-50	5.62E-30	0.0334186	10.484739	2.17E-110	**0**	0.0146116
	WORST	1.88E-44	3.05E-18	9.80E-50	1.87E-29	0.0411806	13.199910	3.64E-107	**0**	0.1025390
	BEST	6.12E-48	9.07E-19	8.06E-52	2.75E-30	0.0049311	8.5564533	3.48E-115	**0**	0.0025895
	RANK	4	6	3	5	8	9	2	**1**	7
F5	**STD**	3.59E-34	6.91E-15	8.23E-40	2.60E-10	0.1980155	0.6733471	18.476538	**0**	10.5017965
	MEAN	1.62E-34	1.34E-14	4.18E-40	3.40E-10	2.1685018	3.2987987	79.089929	**6.15E-202**	51.6603067
	MEDIAN	1.11E-36	8.70E-15	4.05E-41	4.30E-10	2.0383946	3.5217807	90.020091	**0**	49.106986
	WORST	8.04E-34	2.22E-14	1.89E-39	6.31E-10	2.4188074	4.0768800	94.109723	**3.07E-201**	67.769819
	BEST	1.49E-43	8.13E-15	2.72E-41	1.56E-11	2.0072964	2.5588652	52.78007	**0**	39.573285
	RANK	2	4	3	5	6	7	9	**1**	8
F6	**STD**	0.0456765	0.1180529	0.1051982	0.6609552	**0.0001755**	0.2902762	0.0163483	1.2556436	16.915500
	MEAN	0.0299344	0.0549641	0.3068884	1.5000833	**0.0001183**	1.3081808	0.0525887	6.8582806	19.587285
	MEDIAN	0.0035792	0.0032097	0.2699476	1.2607904	**1.10E-05**	1.3810104	0.0493231	7.3068078	12.451722
	WORST	0.1075631	0.2661309	0.4937326	2.4998525	**0.0004093**	1.6060984	0.0699451	8.2355573	49.686543
	BEST	0.0007119	0.0003714	0.2467375	0.7483997	**3.67E-06**	0.8231683	0.0301853	5.1980521	9.7689182
	RANK	3	2	5	6	**1**	7	4	8	9
F7	**STD**	5.2915026	8.2945765	6.9065186	8.2280009	23.847431	8.6775572	**0**	**0**	13.838352
	MEAN	-236	-211.4	-237.8	-220.2	-195.8	-243.4	**-275**	**-275**	-154
	MEDIAN	-234	-211	-239	-224	-209	-244	**-275**	**-275**	-149
	WORST	-230	-204	-228	-210	-165	-230	**-275**	**-275**	-144
	BEST	-245	-224	-244	-229	-220	-251	**-275**	**-275**	-178
	RANK	4	7	5	6	8	3	**1**	**1**	9
F8	**STD**	4.20E-53	0.0010555	1.62E-18	3.86E-10	69.7884211	89.0552602	36540.0054	**0**	19801.7884
	MEAN	1.90E-53	0.0012236	1.17E-18	2.33E-10	361.601031	729.29884	183360.918	**0**	28183.6209
	MEDIAN	6.39E-56	0.0008553	2.62E-20	7.98E-11	340.91535	732.077315	163909.268	**0**	22994.7009
	WORST	9.41E-53	0.002756	3.33E-18	9.20E-10	441.332265	822.388302	244386.289	**0**	58910.2318
	BEST	2.65E-61	0.0002013	3.00E-23	2.04E-12	279.056683	589.51629	158579.213	**0**	11222.3995
	RANK	2	5	3	4	6	7	9	**1**	8
F9	**STD**	1.09E-50	2.32E-18	3.18E-42	2.56E-29	0.12437244	9.56E+49	1.09E-107	**0**	0.02796299
	MEAN	1.09E-50	2.38E-18	1.55E-42	4.88E-29	0.22560673	4.28E+49	4.89E-108	**7.35E-283**	0.01622078
	MEDIAN	8.98E-51	1.62E-18	4.54E-45	4.55E-29	0.19230789	9.40E+46	1.14E-113	**0**	0.00451502
	WORST	2.86E-50	6.42E-18	7.23E-42	8.17E-29	0.44213714	2.14E+50	2.45E-107	**3.68E-282**	0.06575115
	BEST	1.05E-51	7.66E-19	1.56E-51	1.65E-29	0.12729109	2.05E+36	1.85E-116	**0**	0.00013684
	RANK	3	6	4	5	8	9	2	**1**	7
F10	**STD**	**0**	4.03E-161	0	0	0.00595097	6.77E-10	**0**	**0**	3142555.45
	MEAN	3.61E-321	1.95E-161	0	2.86E-168	0.00317397	4.43E-10	**0**	**0**	1554658.69
	MEDIAN	**0**	1.88E-162	0	1.51E-168	0.00066639	2.07E-10	**0**	**0**	253265.205
	WORST	1.80E-320	9.15E-161	0	6.55E-168	0.01376837	1.64E-09	**0**	**0**	7169554.56
	BEST	**0**	3.57E-166	0	3.66E-170	4.14E-06	2.91E-11	**0**	**0**	1103.39289
	RANK	**1**	6	**1**	5	8	7	**1**	**1**	9
F11	**STD**	0.2587203	0.065395	1.57413752	0.42947269	63.3255498	99.207218	0.32323374	**0.0108305**	57673.0527
	MEAN	45.673498	**0.0825822**	45.7366328	46.6455694	128.594339	166.380021	46.9721937	48.757438	50981.6703
	MEDIAN	45.76603	**0.0495198**	44.9278189	46.7194002	135.452305	196.081573	46.8818581	48.762234	40480.3805
	WORST	45.975464	**0.1790078**	48.5157926	47.0654628	207.666025	281.742413	47.4811106	48.76751	141650.641
	BEST	44.842603	**0.0193065**	45.401775	46.1747927	44.9927901	60.3071864	46.6156489	48.739848	2522.42991
	RANK	2	**1**	4	5	3	8	6	7	9
F12	**STD**	1.88E-84	2.49E-38	2.20E-99	4.81E-55	327.892513	0.00060751	**0**	**0**	0.00941336
	MEAN	8.43E-85	2.15E-38	1.62E-99	1.02E-54	519	0.00368598	1.37E-173	**0**	0.00565441
	MEDIAN	3.37E-95	1.13E-38	1.35E-100	1.03E-54	485	0.00364282	4.94E-177	**0**	0.0008964
	WORST	4.20E-84	6.32E-38	4.79E-99	1.77E-54	948	0.00448445	6.84E-173	**0**	0.02204169
	BEST	1.63E-97	5.77E-40	5.03E-102	5.68E-55	79	0.00279186	1.79E-183	**0**	1.02E-05
	RANK	4	6	3	5	9	8	2	**1**	7
F13	**STD**	**1.66E-08**	0.0047427	2.72E-07	2.20E-07	171407.33	16.9040946	1.09E-05	1.15E-06	777.616748
	MEAN	0.6666667	**0.0127358**	0.66666682	0.66666756	128320.403	15.7653328	0.66668599	0.6666681	846.243698
	MEDIAN	0.6666667	**0.0125567**	0.66666669	0.66666745	72590.5166	9.56163308	0.66669033	0.6666675	1009.92938
	WORST	0.6666667	**0.0183155**	0.66666731	0.66666788	412429.715	42.6216417	0.66669463	0.6666697	1804.59295
	BEST	0.6666667	**0.0064638**	0.66666668	0.66666733	3.06753831	1.76479056	0.66666799	0.6666671	12.6609924
	RANK	2	**1**	3	5	8	7	6	4	9
F14	**STD**	1.36E-36	2.56E-09	5.30E-25	7.04E-06	3760.97561	1.24472242	1.34E-11	**0**	81.1045336
	MEAN	6.13E-37	1.18E-09	2.37E-25	8.32E-06	6711.0118	3.35008161	6.00E-12	**0**	53.1651919

© 2026 NSP
Natural Sciences Publishing Cor.

Table 2 (continued)

		HJSRSO	JS	HJSPSO	GWO	PSO	MVO	WOA	RSO	SCA
	MEDIAN	1.35E-49	3.16E-11	2.54E-38	7.13E-06	7419.98254	2.77288935	9.62E-24	**0**	20.1609024
	WORST	3.05E-36	5.75E-09	1.18E-24	1.92E-05	11954.8929	5.43392206	3.00E-11	**0**	197.539893
	BEST	2.96E-80	7.34E-15	7.56E-48	3.43E-07	2453.43959	2.36295824	2.74E-75	**0**	8.96894457
	RANK	2	5	4	6	9	7	3	**1**	8
	STD	6.77E-46	0.1136268	1.32E-08	1.42E-11	231.055017	0.42296938	64.2324539	**0**	39.146521
	MEAN	3.03E-46	1.0448706	5.98E-09	6.51E-12	619.660259	1.02579908	1186.79890	**0**	61.1653868
F15	**MEDIAN**	4.01E-49	1.0559238	1.48E-11	3.46E-14	717.900656	0.96580033	1191.39599	**0**	67.9052534
	WORST	1.52E-45	1.1694385	2.95E-08	3.20E-11	833.253497	1.70169685	1241.48834	**0**	116.047517
	BEST	1.05E-61	0.8670891	7.09E-13	1.90E-15	362.810245	0.67102324	1079.29695	**0**	23.1100462
	RANK	2	6	4	3	8	5	9	**1**	7
	STD	0.5477226	1.37E-130	4.94E-144	**0**	**0**	**0**	**0**	**0**	**0**
	MEAN	**-0.4**	6.36E-131	2.21E-144	4.21E-198	0	7.80E-270	0	0	9.94E-289
F16	**MEDIAN**	1.35E-200	3.47E-142	1.70E-181	9.69E-210	**0**	1.84E-286	**0**	**0**	3.95E-302
	WORST	8.42E-156	3.09E-130	1.10E-143	2.10E-197	**0**	3.90E-269	**0**	**0**	4.97E-288
	BEST	**-1**	7.72E-156	1.39E-190	1.59E-254	0	0	0	0	0.00E+00
	RANK	**1**	9	8	7	2	2	2	2	2
	STD	3.6851842	**0.1262191**	0.87472474	12.0980297	0.19171658	0.17553264	753.941908	2268.0097	20.587053
	MEAN	1.8367929	**0.1631635**	1.28537852	20.8746983	0.30039497	0.24158091	455.406927	2228.4898	23.7308553
F17	**MEDIAN**	0.2438147	0.**1494584**	1.24532602	24.7061904	0.32317986	0.28079238	93.7184142	891.8814	16.3663743
	WORST	8.4254835	**0.3571221**	2.54457557	32.6449789	0.54261322	0.42190907	1790.83638	4877.3589	59.7878466
	BEST	0.0411881	0.0203829	0.07650216	6.2218127	**0.00500504**	0.00971014	16.8249443	415.29913	8.75601036
	RANK	4	3	5	6	**1**	2	8	9	7
	STD	**0**	**0**	**0**	**0**	6.57E-130	1.31E-08	**0**	**0**	1.31E-150
	MEAN	6.97E-272	7.94E-245	1.28E-257	**0**	3.38E-130	2.35E-08	8.43E-197	**0**	5.89E-151
F18	**MEDIAN**	2.81E-278	8.28E-246	2.34E-259	**0**	3.21E-131	2.49E-08	3.76E-199	**0**	4.55E-156
	WORST	3.49E-271	3.78E-244	5.95E-257	**0**	1.51E-129	3.92E-08	3.47E-196	**0**	2.94E-150
	BEST	1.10E-278	1.16E-253	4.94E-262	**0**	6.01E-132	8.54E-09	6.41E-203	**0**	1.06E-159
	RANK	3	5	4	**1**	8	9	6	**1**	7
	STD	**0**	**0**	**0**	6.17E-08	**0**	0.34080791	3.00E-11	0.0004943	2.72E-05
	MEAN	**0**	**0**	**0**	5.34E-08	**0**	0.15241394	1.51E-11	0.0005221	2.46E-05
F19	**MEDIAN**	**0**	**0**	**0**	3.53E-08	**0**	8.81E-09	2.55E-12	0.0004273	1.66E-05
	WORST	**0**	**0**	**0**	1.50E-07	**0**	0.76206965	6.86E-11	0.0013684	7.20E-05
	BEST	**0**	**0**	**0**	2.79E-09	**0**	2.70E-09	1.95E-13	0.0001261	4.40E-06
	RANK	**1**	**1**	**1**	7	**1**	6	5	9	8
	STD	**0**	**0**	**0**	4.89E-08	**0**	1.09E-07	1.87E-05	0.0030443	0.00023888
	MEAN	**0**	**0**	**0**	4.22E-08	**0**	9.32E-08	3.19E-05	0.0028664	0.00028756
F20	**MEDIAN**	**0**	**0**	**0**	1.21E-08	**0**	4.01E-08	3.50E-05	0.0015858	0.00027332
	WORST	**0**	**0**	**0**	9.91E-08	**0**	2.83E-07	5.38E-05	0.0082359	0.00065117
	BEST	**0**	**0**	**0**	2.50E-09	**0**	2.69E-08	1.17E-05	0.0008841	2.96E-05
	RANK	**1**	**1**	**1**	5	**1**	6	7	9	8
	STD	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
	MEAN	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**
F21	**MEDIAN**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**
	WORST	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**
	BEST	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**	**1.38E-87**
	RANK	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
	STD	**0**	**0**	**0**	**0**	5.66E-93	6.54E-09	**0**	**0**	5.76E-126
	MEAN	1.47E-223	1.10E-170	2.95E-186	1.36E-285	3.80E-93	6.16E-09	**0**	**0**	2.59E-126
F22	**MEDIAN**	2.98E-229	6.03E-171	1.68E-186	4.34E-292	2.16E-94	6.67E-09	**0**	**0**	1.12E-129
	WORST	7.33E-223	3.14E-170	1.08E-185	6.82E-285	1.29E-92	1.58E-08	**0**	**0**	1.29E-125
	BEST	1.27E-229	5.08E-177	3.20E-191	3.29E-304	4.67E-98	7.94E-12	**0**	**0**	3.69E-138
	RANK	4	6	5	3	8	9	**1**	**1**	7
	STD	**2.78E-17**	2.78E-17	2.78E-17	2.75E-08	4.81E-17	8.96E-08	1.77E-06	2.47E-08	2.82E-09
	MEAN	**0.2925786**	0.2925786	0.29257863	0.29257866	0.29257863	0.29257876	0.29257996	0.2925787	0.29257864
F23	**MEDIAN**	**0.2925786**	0.2925786	0.29257863	0.29257867	0.29257863	0.29257873	0.29257877	0.2925786	0.29257864
	WORST	**0.2925786**	0.2925786	0.29257863	0.2925787	0.29257863	0.29257888	0.29258257	0.2925787	0.29257864
	BEST	**0.2925786**	**0.2925786**	**0.29257863**	0.29257863	**0.29257863**	0.29257866	0.29257866	0.2925786	0.29257864
	RANK	**1**	**1**	**1**	5	**1**	8	9	6	7
	STD	2.51E-15	1.78E-15	**0**	2.06E-06	3.55E-15	0.00044704	7.52E-05	0.1412279	0.00387264
	MEAN	**19.10588**	19.10588	19.1058798	19.1058823	19.1058798	19.1065338	19.1059865	19.255034	19.1110526
F24	**MEDIAN**	**19.10588**	19.10588	19.1058798	19.1058834	19.1058798	19.1065016	19.1059912	19.236385	19.1105334
	WORST	**19.10588**	19.10588	19.1058798	19.1058843	19.1058798	19.1071776	19.1060932	19.455921	19.1153133
	BEST	**19.10588**	**19.10588**	**19.1058798**	19.1058799	**19.1058798**	19.1059393	19.1058879	19.111307	19.1069857
	RANK	**1**	**1**	**1**	5	**1**	7	6	9	8
	STD	**0**	3.65E-24	2.87E-23	6.30E-08	3.68E-22	4.18E-08	1.40E-06	0.0030189	9.16E-05
	MEAN	**0**	2.95E-24	1.47E-23	5.98E-08	4.53E-22	5.43E-08	1.49E-06	0.0025073	0.00011962
F25	**MEDIAN**	**0**	1.05E-24	9.96E-25	2.36E-08	4.15E-22	4.51E-08	1.34E-06	0.0011647	7.78E-05
	WORST	**0**	8.44E-24	6.58E-23	1.67E-07	8.44E-22	1.25E-07	2.99E-06	0.0076172	0.0002235
	BEST	**0**	8.66E-26	6.93E-25	2.03E-08	1.32E-24	1.50E-08	6.20E-08	0.0002453	3.79E-05
	RANK	**1**	2	3	6	4	5	7	9	8
	STD	29.169573	53.795181	11.8513332	20.7241408	19.789797	12.0573781	41.2770154	37.062251	**7.50918163**
	MEAN	**86.902013**	175.5321	284.959725	227.257418	158.586962	140.050131	94.2580925	273.56462	313.267242
F26	**MEDIAN**	**100.8264**	155.37237	288.559375	218.998884	154.763475	133.804015	118.43834	282.58451	313.150979
	WORST	**111.91754**	266.43158	299.386157	264.009347	188.330062	155.766404	123.067194	309.15637	320.682626
	BEST	**25.364444**	134.73028	272.497398	215.191027	133.447194	127.308313	45.4752278	221.45917	303.056104
	RANK	**1**	5	8	6	4	3	2	7	9
	STD	**0**	6.4379445	**0**	2.54E-14	69.2950355	48.0079477	**0**	**0**	33.9547226
	MEAN	**0**	2.8791363	**0**	1.14E-14	274.003116	263.543326	**0**	**0**	71.734121
F27	**MEDIAN**	**0**	**0**	**0**	**0**	238.941498	256.407262	**0**	**0**	66.1873883
	WORST	**0**	14.395682	**0**	5.68E-14	359.764183	336.931179	**0**	**0**	105.650292
	BEST	**0**	**0**	**0**	**0**	202.382895	216.436856	**0**	**0**	23.5236298

© 2026 NSP
Natural Sciences Publishing Cor.

Table 2 (continued)

		HJSRSO	JS	HJSPSO	GWO	PSO	MVO	WOA	RSO	SCA
	RANK	**1**	**1**	**1**	**1**	8	9	**1**	**1**	7
	STD	**0**	0.6704422	5.51717861	0.20365056	1.34001122	0.00205575	0.07489078	2.2370197	1.07067925
	MEAN	**0.9**	13.916638	10.7315734	1.5480303	5.43938454	1.01378007	0.93349217	2.5178445	12.0708314
F28	**MEDIAN**	**0.9**	14.147588	13.0088135	1.51346029	5.43938454	1.01384174	**0.9**	**0.9**	12.0505582
	WORST	**0.9**	14.546464	13.8673884	1.81211547	6.62322041	1.01609627	1.06746088	5.3841406	13.6807673
	BEST	**0.9**	13.160476	**0.9**	1.28815262	3.36767178	1.01104401	**0.9**	**0.9**	10.7349301
	RANK	**1**	9	**1**	6	7	5	**1**	**1**	8
	STD	80.987273	49.902928	769.397313	2446.5792	**0.00159368**	2005.6399	1216.58224	6143.3969	58979444.1
	MEAN	55.131414	39.725402	399.326995	9688.15965	**0.00159684**	3495.14646	2028.96038	22166.018	29155435.3
F29	**MEDIAN**	18.573594	18.275427	41.5874855	11098.6758	**0.000676**	2762.49594	1674.74894	22584.806	2588331.63
	WORST	197.72162	118.08998	1772.10359	11960.9134	**0.00342468**	6997.41891	4112.79072	29514.296	134503927
	BEST	2.1598181	1.1626412	8.45843523	6197.01859	**0.00028774**	1978.655	1171.70558	13002.185	33678.953
	RANK	3	2	4	7	**1**	6	5	8	9
	STD	1.08E-52	2.36E-20	1.01E-44	2.15E-05	2.43266819	1.9287519	1.23E-106	**0**	0.35483508
	MEAN	2.35E-52	3.30E-20	4.51E-45	9.63E-06	1.78413647	11.1671481	5.52E-107	**5.70E-263**	0.2857372
F30	**MEDIAN**	2.65E-52	2.37E-20	8.16E-48	9.16E-29	0.00820507	11.3923424	2.31E-111	**0**	0.0654621
	WORST	3.37E-52	7.25E-20	2.25E-44	4.82E-05	4.45107035	13.2227293	2.76E-106	**2.85E-262**	0.84129523
	BEST	6.96E-53	1.21E-20	4.06E-52	1.18E-29	0.00694378	8.0348992	7.89E-118	**0**	0.01844851
	RANK	3	6	4	5	7	9	2	**1**	8
	STD	2.25E-67	8.64E-14	1.61E-34	6.09E-70	4.77E+16	998.827858	2.66E-11	**1.28E-88**	133313.882
	MEAN	1.31E-67	3.87E-14	7.49E-35	2.73E-70	2.77E+16	475.903638	1.26E-11	**5.71E-89**	59624.1098
F31	**MEDIAN**	3.72E-68	4.13E-19	4.64E-37	1.42E-75	2.5595E+14	11.3424889	5.75E-16	**0**	6.88270316
	WORST	5.28E-67	1.93E-13	3.63E-34	1.36E-69	1.10E+17	2261.33211	6.02E-11	**2.86E-88**	298103.232
	BEST	1.45E-73	8.81E-30	4.59E-43	9.26E-81	1.2264E+11	2.00839212	7.51E-24	**0**	0.0106635
	RANK	3	5	4	2	9	8	6	**1**	7
	STD	**0**	1.59E-15	1.59E-15	3.18E-15	0.54765422	9.94387866	1.59E-15	1.59E-15	0.07248031
	MEAN	**-8.88E-16**	5.51E-15	1.95E-15	2.61E-14	0.37396507	9.00104423	1.95E-15	-1.78E-16	20.3970598
F32	**MEDIAN**	**-8.88E-16**	6.22E-15	2.66E-15	2.75E-14	0.0466073	2.63455631	2.66E-15	-8.88E-16	20.4092375
	WORST	**-8.88E-16**	6.22E-15	2.66E-15	2.75E-14	1.26983513	19.8876	2.66E-15	2.66E-15	20.4509639
	BEST	**-8.88E-16**	2.66E-15	-8.88E-16	2.04E-14	0.00565522	1.23751856	-8.88E-16	-8.88E-16	20.2736662
	RANK	**1**	5	**1**	6	7	8	**1**	**1**	9
	STD	**3.7063399**	5.5644468	16.819196	13.3407152	19.1169192	74.5762369	37.0059818	5.1730107	1852.80663
	MEAN	**25.498396**	26.954785	44.5975165	56.5870821	48.0151479	431.207206	80.0259369	130.48001	1681.95025
F33	**MEDIAN**	**27.727651**	28.294284	37.7492951	52.6167686	50.8181678	438.849209	92.7137738	130.93041	1086.79994
	WORST	**30.4153**	31.142615	74.0001398	75.3550982	67.8614825	517.044226	113.671197	137.35765	4904.82118
	BEST	**18.146123**	21.887591	32.5849751	42.7016627	21.6713256	324.608436	19.0624766	122.77936	240.168397
	RANK	**1**	4	5	6	3	9	2	7	8
	STD	**3.68E-19**	0.0547723	3.39E-09	0.04472136	0.04472163	0.07071068	0.1	0.0446647	0.56001576
	MEAN	**2.48E-19**	0.1398734	0.09987335	0.17987335	0.67987383	1.09987345	0.19987334	0.0199747	1.67403986
F34	**MEDIAN**	8.15E-22	0.0998733	0.09987335	0.19987335	0.69987335	1.09987341	0.19987334	**0**	1.30959921
	WORST	**8.23E-19**	0.1998734	0.09987335	0.19987335	0.69987577	1.19987335	0.29987334	0.0998733	2.4022713
	BEST	**0**	0.0998733	0.09987335	0.09987335	0.59987335	0.99987335	0.09987334	**0**	1.20167668
	RANK	**1**	5	4	6	7	8	3	**1**	9
	STD	**6.324669**	28.868334	81.4495874	75.12302	40.7264412	36.5807691	90.4893237	145.24391	40.8598641
	MEAN	**-1955.4075**	-1748.915	-1694.7287	-1365.6343	-1726.4639	-1630.2626	-1899.12734	-1042.084	-812.17303
F35	**MEDIAN**	**-1958.2386**	-1746.179	-1689.4359	-1355.7682	-1703.8469	-1618.9566	-1942.11032	-1004.986	-795.82724
	WORST	**-1944.0943**	-1703.777	-1602.7979	-1299.0703	-1689.7106	-1590.6969	-1739.80938	-904.2787	-764.90241
	BEST	**-1958.2981**	-1774.379	-1802.4095	-1489.0054	-1788.6673	-1675.5022	-1957.53994	-1214.097	-866.61999
	RANK	**1**	5	3	7	4	6	2	8	9
	STD	**0**	**0**	**0**	**0**	2.87E-07	0.05415026	**0**	**0**	0.25895982
	MEAN	**0**	**0**	**0**	**0**	5.42E-07	0.1338762	**0**	**0**	0.62633698
F36	**MEDIAN**	**0**	**0**	**0**	**0**	4.95E-07	0.11023593	**0**	**0**	0.60827007
	WORST	**0**	**0**	**0**	**0**	1.00E-06	0.22923115	**0**	**0**	1.02147999
	BEST	**0**	**0**	**0**	**0**	2.89E-07	0.09782032	**0**	**0**	0.31444865
	RANK	**1**	**1**	**1**	**1**	7	8	**1**	**1**	9
	STD	**4.74E-25**	1.64E-22	8.24E-24	4.10E-16	2.30E-21	5.45E-24	0.44721359	0.5477226	4.58E-16
	MEAN	**-1**	1.05E-22	1.68E-23	2.86E-24	1.27E-21	1.85E-23	-0.2	-0.4	4.19E-16
F37	**MEDIAN**	**-1**	4.36E-23	1.69E-23	2.83E-24	3.06E-25	1.72E-23	1.48E-21	1.13E-18	1.70E-16
	WORST	**-1**	3.96E-22	2.60E-23	3.64E-24	5.30E-21	2.79E-23	2.53E-21	6.08E-16	9.86E-16
	BEST	**-1**	4.19E-24	6.71E-24	2.42E-24	4.05E-26	1.38E-23	**-1**	**-1**	8.64E-17
	RANK	**1**	6	7	5	4	8	**1**	**1**	9
	STD	**3.21E-22**	1.96E-21	5.57E-22	5.53E-12	2.60E-21	2.54E-19	7.78E-21	2.70E-21	8.07E-17
	MEAN	**1.26E-20**	1.32E-20	1.27E-20	3.74E-12	1.63E-19	3.85E-19	1.60E-20	1.69E-19	1.72E-16
F38	**MEDIAN**	1.24E-20	1.24E-20	1.26E-20	9.21E-16	1.62E-19	3.65E-19	1.29E-20	1.68E-19	1.37E-16
	WORST	1.31E-20	1.67E-20	1.36E-20	1.23E-11	1.65E-19	7.57E-19	2.99E-20	1.73E-19	2.62E-16
	BEST	1.21E-20	1.21E-20	1.22E-20	1.83E-17	1.59E-19	9.70E-20	1.22E-20	1.66E-19	9.15E-17
	RANK	**1**	2	3	8	6	5	4	7	9
	STD	**0.0060511**	0.0089827	0.76975219	0.23823655	0.04497707	0.05128378	0.07231713	0.0390635	61126.72
	MEAN	**0.0044318**	0.0091246	1.20434051	1.80742032	0.04099934	0.14812677	0.16175374	4.8595999	27640.5156
F39	**MEDIAN**	**1.99E-05**	0.0110994	1.28779162	1.87209068	0.02333169	0.14585299	0.16173285	4.8414351	120.139626
	WORST	**0.0111259**	0.0220397	2.17502898	1.98297061	0.1140124	0.20673692	0.24985220	4.9015634	136984.573
	BEST	**6.51E-06**	0.0004647	0.33365788	1.39738441	0.00106926	0.08580439	0.08499222	4.8149214	5.69130153
	RANK	**1**	2	6	7	3	5	4	8	9
	STD	0.0001038	0.0017066	0.00232082	0.01743931	**2.48E-07**	0.49346362	0.00087997	0.122262	384361.038
	MEAN	9.51E-05	0.000801	0.0013488	0.06392671	**2.60E-07**	2.70038045	0.00166064	0.5071396	174887.72
F40	**MEDIAN**	9.07E-05	2.79E-05	0.0002961	0.06236331	**2.22E-07**	2.56707344	0.00142096	0.4740897	1908.67936
	WORST	0.0002638	0.0038535	0.00549747	0.09303576	**6.52E-07**	3.50239229	0.00320216	0.6967617	862413.384
	BEST	7.38E-06	1.60E-05	0.00023632	0.04907658	**1.01E-08**	2.234318	0.00100740	0.373493	11.9186286
	RANK	2	3	4	6	**1**	8	5	7	9
	STD	0	0	0	**0**	8.80E-129	2.06E-07	**0**	**0**	2.96E-153
	MEAN	1.35E-282	1.73E-244	6.67E-257	**0**	4.93E-129	2.69E-07	2.20E-282	**0**	1.36E-153
F41										

© 2026 NSP
Natural Sciences Publishing Cor.

Table 2 (continued)

		HJSRSO	JS	HJSPSO	GWO	PSO	MVO	WOA	RSO	SCA
	MEDIAN	9.12E-287	3.80E-248	3.61E-259	**0**	6.86E-130	2.16E-07	2.05E-292	**0**	1.32E-155
	WORST	6.71E-282	7.15E-244	3.28E-256	**0**	2.05E-128	5.81E-07	1.09E-281	**0**	6.65E-153
	BEST	1.67E-292	2.30E-253	3.70E-262	**0**	5.84E-133	6.05E-08	**0.00E+00**	**0**	9.57E-159
	RANK	4	6	5	**1**	8	9	**1**	**1**	7
	STD	**0**	**0**	**0**	1.25E-08	**0**	2.05E-06	1.12E-09	3.30E-06	4.37E-05
F42	**MEAN**	**-195.62903**	**-195.629**	**-195.62903**	-195.62903	**-195.62903**	-195.62902	**-195.629028**	-195.629	-195.62897
	MEDIAN	**-195.62903**	**-195.629**	**-195.62903**	-195.62903	**-195.62903**	-195.62902	**-195.629028**	-195.629	-195.62897
	WORST	**-195.62903**	**-195.629**	**-195.62903**	-195.62903	**-195.62903**	-195.62902	**-195.629028**	-195.629	-195.62891
	BEST	**-195.62903**	**-195.629**	**-195.62903**	-195.62903	**-195.62903**	-195.62903	**-195.629028**	-195.629	-195.62902
	RANK	**1**	**1**	**1**	6	**1**	8	**1**	7	9
	STD	**0**	**0**	**0**	4.13E-13	**0**	2.08E-11	2.22E-16	2.22E-08	4.93E-10
	MEAN	**-2.0218068**	**-2.021807**	**-2.0218068**	-2.0218068	**-2.0218068**	-2.0218068	**-2.02180678**	-2.021807	-2.0218068
F43	**MEDIAN**	**-2.0218068**	**-2.021807**	**-2.0218068**	-2.0218068	**-2.0218068**	-2.0218068	**-2.02180678**	-2.021807	-2.0218068
	WORST	**-2.0218068**	**-2.021807**	**-2.0218068**	-2.0218068	**-2.0218068**	-2.0218068	**-2.02180678**	-2.021807	-2.0218068
	BEST	**-2.0218068**	**-2.021807**	**-2.0218068**	-2.0218068	**-2.0218068**	-2.0218068	**-2.02180678**	-2.021807	-2.0218068
	RANK	**1**	**1**	**1**	7	**1**	6	**1**	9	8
	STD	**7.11E-15**	2.01E-14	1.00E-14	1.79E-05	2.36E-14	1.22E-06	4.44E-08	0.0577471	0.02520044
	MEAN	**-106.76454**	-106.7645	-106.76454	-106.76452	-106.76454	-106.76453	-106.764537	-106.7237	-106.73055
F44	**MEDIAN**	**-106.76454**	-106.7645	-106.76454	-106.76453	-106.76454	-106.76453	-106.764537	-106.7629	-106.73641
	WORST	**-106.76454**	-106.7645	-106.76454	-106.76449	-106.76454	-106.76453	-106.764537	-106.6391	-106.69283
	BEST	**-106.76454**	-106.7645	-106.76454	-106.76454	**-106.76454**	-106.76454	-106.764537	-106.7645	-106.76239
	RANK	**1**	**1**	**1**	6	**1**	7	5	8	9
	STD	**0**	**0**	**0**	7.03E-09	**0**	6.98E-08	1.75E-11	6.39E-06	4.34E-06
	MEAN	**-1.0316285**	**-1.031628**	**-1.0316285**	-1.0316284	**-1.0316285**	-1.0316284	-1.03162845	-1.031622	-1.0316221
F45	**MEDIAN**	**-1.0316285**	**-1.031628**	**-1.0316285**	-1.0316285	**-1.0316285**	-1.0316284	-1.03162845	-1.031624	-1.0316231
	WORST	**-1.0316285**	**-1.031628**	**-1.0316285**	-1.0316284	**-1.0316285**	-1.0316283	-1.03162845	-1.031612	-1.0316163
	BEST	**-1.0316285**	**-1.031628**	**-1.0316285**	-1.0316285	**-1.0316285**	-1.0316284	-1.03162845	-1.031628	-1.0316263
	RANK	**1**	**1**	**1**	**6**	**1**	**7**	**5**	**8**	**9**
	STD	**0**	**0**	**0**	8.24E-08	**0**	1.36E-08	4.10E-08	0.0113961	0.00012954
	MEAN	**0.3978874**	**0.3978874**	**0.39788736**	0.39788748	**0.39788736**	0.39788738	0.39788739	0.4041164	0.39805947
F46	**MEDIAN**	**0.3978874**	**0.3978874**	**0.39788736**	0.3978875	**0.39788736**	0.39788738	0.39788737	0.3995609	0.39809077
	WORST	**0.3978874**	**0.3978874**	**0.39788736**	0.39788756	**0.39788736**	0.3978874	0.39788746	0.4244641	0.39820021
	BEST	**0.3978874**	**0.3978874**	**0.39788736**	0.39788736	**0.39788736**	0.39788736	0.39788736	0.3982143	0.39789322
	RANK	**1**	**1**	**1**	**6**	**1**	**5**	**7**	**9**	**8**
	STD	**0.00E+00**	**0**	**0**	6.83E-06	9.93E-16	1.51E-07	4.12E-06	1.75E-06	1.35E-06
	MEAN	**3**	**3**	**3**	3.00000503	**3**	3.00000041	3.00000206	3.0000018	3.00000165
F47	**MEDIAN**	**3**	**3**	**3**	3.00000278	**3**	3.00000042	3.00000010	3.000002	3.00000196
	WORST	**3**	**3**	**3**	3.00001717	**3**	3.00000064	3.00000943	3.0000035	3.00000352
	BEST	**3**	**3**	**3**	3.00000117	**3**	3.00000023	3.00000003	3	3.00000027
	RANK	**1**	**1**	**1**	9	**1**	7	6	5	8
	STD	**0**	**0**	**0**	3.63E-06	**0**	8.19E-08	0.0003722	0.204933	0.000301
	MEAN	**-3.8627821**	**-3.862782**	**-3.8627821**	-3.8627785	**-3.862782**	-3.862782	-3.862486	-3.530367	-3.85457
F48	**MEDIAN**	**-3.8627821**	**-3.862782**	**-3.8627821**	-3.86278	**-3.862782**	-3.862782	-3.862714	-3.439488	-3.85465
	WORST	**-3.8627821**	**-3.862782**	**-3.8627821**	-3.8627741	**-3.862782**	-3.862782	-3.861926	-3.330277	-3.85407
	BEST	**-3.8627821**	**-3.862782**	**-3.8627821**	-3.8627818	**-3.862782**	-3.862782	-3.862760	-3.825458	-3.85482
	RANK	**1**	**1**	**1**	6	**1**	5	7	9	8
	STD	**2.22E-16**	**2.22E-16**	**2.22E-16**	1.13E-06	0.0872102	0.065198	0.0936957	0.5495913	0.3735130
	MEAN	**-3.32199**	**-3.321995**	**-3.321995**	-3.321992	-3.260202	-3.27437	-3.180826	-1.678526	-2.919089
F49	**MEDIAN**	**-3.321995**	**-3.321995**	**-3.321995**	-3.321993	-3.321995	-3.32199	-3.193982	-1.413167	-3.10487
	WORST	**-3.321995**	**-3.321995**	**-3.321995**	-3.321991	-3.137641	-3.20293	-3.084736	-1.305204	-2.25508
	BEST	**-3.321995**	**-3.321995**	**-3.321995**	-3.321994	**-3.321995**	-3.32199	-3.321939	-2.604511	-3.11186
	RANK	**1**	**1**	**1**	6	**1**	5	7	9	8
	STD	**0**	**0**	**0**	1.40E-06	**0**	2.87E-09	2.50E-11	1.19E-05	3.50E-06
	MEAN	**-2.062611**	**-2.062612**	**-2.062611**	-2.062611	**-2.062611**	-2.062611	-2.062611	-2.062606	-2.06260
F50	**MEDIAN**	**-2.062611**	**-2.062612**	**-2.062611**	-2.062611	**-2.062611**	-2.062611	-2.062611	-2.062611	-2.06260
	WORST	**-2.062611**	**-2.06261**	**-2.06261**	-2.06260	**-2.06261**	-2.06261	-2.062611	-2.0625	-2.06260
	BEST	**-2.062611**	**-2.06261**	**-2.06261**	-2.062611	**-2.062611**	-2.06261	**-2.062611**	-2.0626	-2.06261
	RANK	**1**	**1**	**1**	6	**1**	7	**1**	8	9

© 2026 NSP
Natural Sciences Publishing Cor.

Table 3: Friedman statistical test results for 50 benchmark functions.

Algorithm	Mean Rank	Algorithm	Mean Rank
HJSRSO	**2.73**	PSO	5.13
HJSPSO	3.71	MVO	6.78
JS	4.17	SCA	8.02
WOA	4.34	GWO	5.33
RSO	4.79		
Test Statistics			
Sample Size (N)			50
Chi-Square (χ^2)			147.267
Degrees of freedom (df)			8
Asymp. Sig. (p-value)			0.000

HJSRSO demonstrates a good exploitation capability for unimodal benchmark functions, specifically F16, F19, F20, and F25, allowing it to effectively improve solutions and approach the global optimum with high precision. This behavior demonstrates how well the algorithm works to enhance the search around potential spots once they are found. Despite the existence of several local minima, HJSRSO shows a strong capacity to avoid local optima in the case of multimodal functions, achieving quick and steady convergence.

It is noteworthy that because of its robust exploration mechanism, which enables wide coverage of the search space, the original Jellyfish Search (JS) method performs well on multimodal problems. On the other hand, because it guides search agents toward the best solutions found thus far, the Rat Swarm Optimization (RSO) algorithm performs competitively on unimodal functions like F5 and F8, where exploitation is essential. The suggested HJSRSO greatly increases overall convergence speed, solution correctness, and robustness across various problem landscapes by combining the exploration strength of JS with the exploitation efficiency of RSO and further refining this hybrid framework with additional techniques.

4.2 Outcomes for problems involving engineering design

Examining intelligent optimization algorithms' universality and practical usefulness in real-world engineering contexts is crucial as these algorithms continue to advance. An algorithm's efficacy can be reliably determined by evaluating it on typical engineering applications, since the majority of engineering design tasks can be phrased as optimization problems. In order to verify the robustness and broad applicability of the suggested approach, seven different engineering optimization problems are chosen for this study. These problems cover a variety of optimization properties including constrained formulations, continuous and mixed design variables, in addition to weakly constrained, strongly constrained, and implicit criteria.

This section evaluates and contrasts the HJSRSO algorithm's performance with a number of current optimization techniques that can handle real-world engineering issues. Tension/compression spring design, gas transmission compressor design, cantilever design, pressure vessel design, welded beam design, gear train design, and speed reducer design are some of the chosen case studies that show how HJSRSO may be applied and used to solve a variety of engineering design problems. The seven case studies are all nonlinear constrained optimization problems with intricate constraint connections and very nonlinear objective functions. The penalty function approach, a popular and efficient method for converting limited optimization problems into unconstrained ones, is utilized to deal with these limitations. This is accomplished by adding penalty terms related to constraint violations to the original objective function, allowing intelligent optimization algorithms like HJSRSO to provide an effective solution.

(a)**The tension/compression spring design**

The objective of this optimization issue is to find the best design that satisfies a set of nonlinear mechanical and geometric constraints while minimizing the overall weight of a tension/compression spring. These constraints include allowable shear stress limits, minimum deflection requirements, and surge frequency restrictions to avoid resonance, outer diameter bounds, and permissible ranges of the design variables. As depicted in Figure 3 [14], the spring design is defined by three continuous decision variables: the wire diameter d(x_1), the mean coil diameter D(x_2), and the number of active coils P(x_3).

The mathematical formulation of the objective function and the associated constraint equations is expressed as follows:

$$
\begin{aligned}
\text{Minimize} \quad & f(x) = (x_3+2)x_2x_1^2 \\
\text{Subject to:} \quad & 1-\frac{x_2^3x_3}{71785x_1^4} \le 0, \\
& \frac{4x_2^2-x_1x_2}{12566(x_2x_1^3-x_1^4)}+\frac{1}{5108x_1^2}-1 \le 0, \\
& 1-\frac{140.45x_1}{x_2^2x_3} \le 0, \\
& \frac{x_1+x_2}{1.5}-1 \le 0, \\
& 0.05 \le x_1 \le 2, \\
& 0.25 \le x_2 \le 1.3, \\
& 2 \le x_3 \le 15.
\end{aligned}
\tag{19}
$$

The complete statistical results obtained by JS, HJSPSO, GWO , RSO, WOA [48], MFO [49], Spotted Hyena Optimizer (SHO) [42], MVO [42], GA [42], harmony search (HS) [42], PSO [42], gravitational search algorithm (GSA) [42],SCA [42], HHO [50], and the proposed HJSRSO algorithm for the engineering design problem illustrated in

© 2026 NSP
Natural Sciences Publishing Cor.

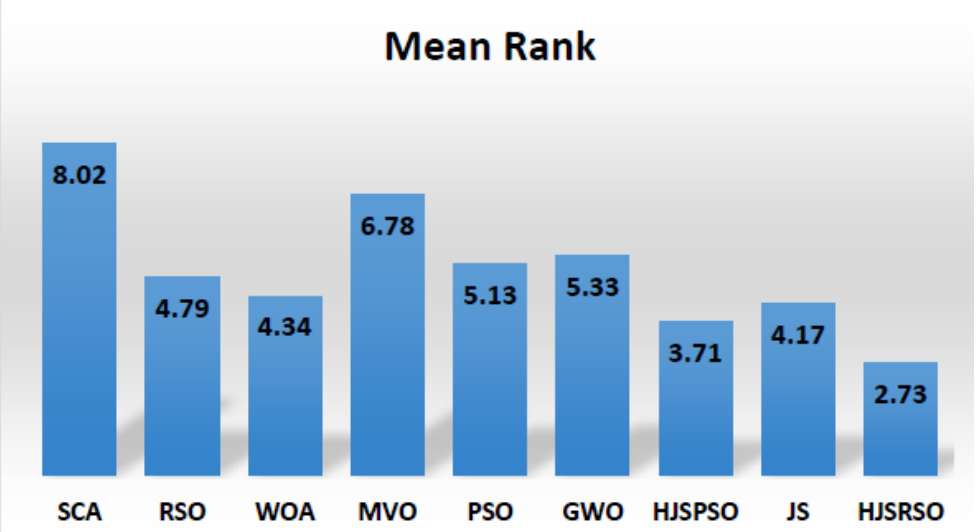


Fig. 1: Friedman test results showing mean ranks for HJSRSO and eight competing algorithms.

Table 4: Outcomes of the Wilcoxon signed-rank test applied to fifty benchmark functions

Test statistics			**Ranks** N	M-R	S-R	
JS - HJSRSO		N-R	6^{a}	23.83	143	a. JS < HJSRSO
Z	-2.641-^{b}	P-R	28^{b}	16.14	452	b. JS > HJSRSO
Asymp. Sig. (2-tailed)	0.008	Ties	16^{c}			c. JS = HJSRSO
b. Depend on N-R.		Total	50			
HJSPSO - HJSRSO		N-R	6^{d}	9.83	59.00	d. HJSPSO < HJSRSO
Z	-3.704-^{b}	P-R	25^{e}	17.48	437.00	e. HJSPSO > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	19^{f}			f. HJSPSO = HJSRSO
b. Depend on N-R.		Total	50			
GWO - HJSRSO		N-R	5^{g}	9.00	45.00	g. GWO < HJSRSO
Z	-5.492-^{b}	P-R	42^{h}	25.79	1083.00	h. GWO > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	3^{i}			i. GWO = HJSRSO
b. Depend on N-R.		Total	50			
PSO - HJSRSO		N-R	4^{j}	15.00	60.00	j. PSO < HJSRSO
Z	-4.289-^{b}	P-R	32^{k}	18.94	606.00	k. PSO > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	14^{l}			l. PSO = HJSRSO
b. Depend on N-R		Total	50			
MVO - HJSRSO		N-R	2^{m}	31.00	62.00	m. MVO < HJSRSO
Z	-5.476-^{b}	P-R	47^{n}	24.74	1163.00	n. MVO > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	1^{o}			o. MVO = HJSRSO
b. Depend on N-R.		Total	50			
WOA - HJSRSO		N-R	10^{p}	10.5	105.00	p. WOA < HJSRSO
Z	-4.100-^{b}	P-R	30^{q}	23.83	715.00	q. WOA > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	10^{r}			r. WOA = HJSRSO
b. Depend on N-R.		Total	50			
RSO - HJSRSO		N-R	16^{s}	10.56	169.00	s. RSO < HJSRSO
Z	-3.532-^{b}	P-R	26^{t}	28.23	734.00	t. RSO > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	8^{u}			u. RSO = HJSRSO
b. Depend on N-R.		Total	50			
SCA - HJSRSO		N-R	0^{v}	0.00	0.00	v. SCA < HJSRSO
Z	-6.093-^{b}	P-R	49^{w}	25.00	1225.00	w. SCA > HJSRSO
Asymp. Sig. (2-tailed)	0.000	Ties	1^{x}			x. SCA = HJSRSO
b. Depend on N-R.		Total	50			

Figure 3 are presented in Tables 5 and 6. These tables report the optimized design variable values along with the corresponding statistical performance indicators, including the best (minimum), mean, worst, and standard deviation of the spring weight, calculated over 30 independent runs for each algorithm. The best values of each indicator are highlighted in bold for clarity.

© 2026 NSP
Natural Sciences Publishing Cor.

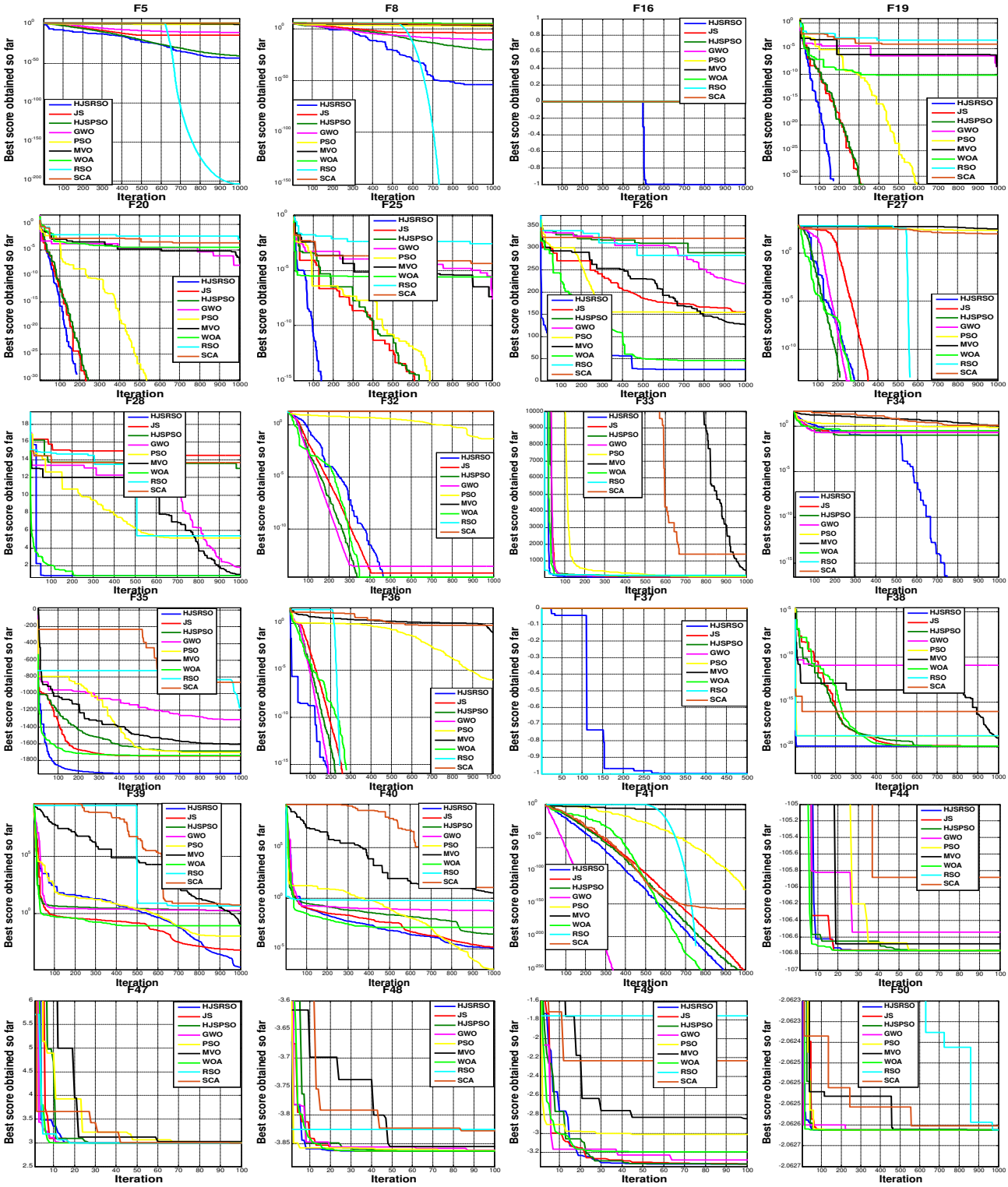


Fig. 2: Convergence curves of all algorithms on the twenty-four benchmark test functions

As observed from Table 6, the HJSRSO algorithm consistently achieves superior performance across all statistical metrics when compared with the competing algorithms. Specifically, HJSRSO exhibits great robustness, dependability, and solution stability by achieving the lowest minimum and mean spring weights while preserving the smallest standard deviation. These findings support the usefulness and efficiency of HJSRSO in resolving real-world engineer-

© 2026 NSP
Natural Sciences Publishing Cor.

Table 5: Performance comparisons on the tension/compression spring problem

Algorithms	Optimum variables d	D	N	Optimum cost
HJSRSO	**0.05153638**	**0.353055741**	**11.50693135**	**0.012665659**
JS	0.052375556	0.373391304	10.37696337	0.012677558
HJSPSO	0.051786454	0.359057161	11.15905503	0.012671284
GWO	0.05040596	0.32657162	13.31029772	0.012703572
RSO	0.069094	0.937540	2.0181	0.012878
WOA	0.060649	0.61304	4.2157	0.012687
MFO	0.058979	0.55879	4.9783	0.012666
SHO	0.051144	0.343751	12.0955	0.012674
MVO	0.05	0.315956	14.22623	0.01281693
SCA	0.05078	0.334779	12.72269	0.012709667
GSA	0.05	0.317312	14.22867	0.012873881
GA	0.0501	0.310111	14	0.013036251
HS	0.05025	0.316351	15.2396	0.012776352
PSO	0.05	0.310414	15	0.01319258

Table 6: Evaluation of statistical outcomes for the tension/compression spring problem

Algorithm	WORST	MEAN	BEST	STD
HJSRSO	**0.012716522**	**0.01268314**	**0.012665659**	**7.64E-06**
JS	0.012782589	0.012714874	0.012677558	3.24E-05
HJSPSO	0.012802911	0.012717853	0.012671284	3.90E-05
GWO	0.012729056	0.012721402	0.012703572	1.69E-05
RSO	0.018036	0.017167	0.012878	2.4197E-05
WOA	0.017329	0.013813	0.012687	0.0000012431
MFO	0.014735	0.012969	0.012666	0.00000022056
SHO	0.012715185	0.012684106	0.012674	0.000027
MVO	0.017839737	0.014464372	0.01281693	0.001622
SCA	0.012998448	0.012839637	0.012709667	0.000078
GSA	0.014211731	0.013438871	0.012873881	0.000287
GA	0.016251423	0.014036254	0.013036251	0.002073
HS	0.01521423	0.013069872	0.012776352	0.000375
PSO	0.01319258	0.014817181	0.017862507	0.002272

ing design issues as well as its capacity to find the best configurations for design variables at the lowest possible cost.

(b)The welded beam design problem

The goal of this design challenge is to reduce the cost of fabricating a welded beam while meeting certain geometric and mechanical limitations, such as end deflection (δ), bending stress (σ), shear stress (τ), and other side constraints [51]. Four design variables define the welded beam problem, as shown in Figure 4: the weld thickness ($h(x_1)$), the attached bar length ($l(x_2)$), the bar height ($t(x_3)$), and the bar thickness ($b(x_4)$) [52].

The mathematical formulation of the optimization problem is expressed as follows:

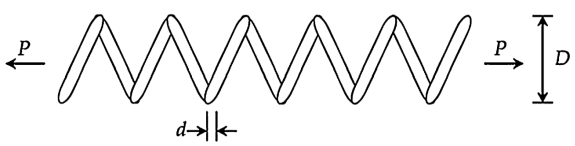


Fig. 3: Tension/compression spring design problem [14].

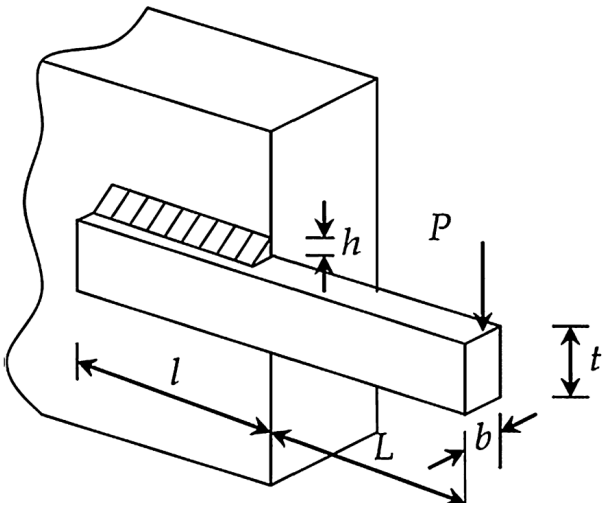


Fig. 4: The welded beam design problem [52].

$$
\begin{aligned}
\text{Minimize} \quad & f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14+x_2) \\
\text{Subject to:} \quad & \tau(x) - 13600 \le 0, \quad \sigma(x) - 30000 \le 0, \\
& \delta(x) - 0.25 \le 0, \quad 6000 - P_c(x) \le 0, \\
& x_1 - x_4 \le 0, \quad 0.125 - x_1 \le 0, \\
& 0.10471x_1^2 + 0.04811x_3x_4(14+x_2) - 5.0 \le 0, \\
\text{where} \quad & \tau(x) = \sqrt{(\tau')^2 + 2\tau'\tau''\frac{x_2}{2R} + (\tau'')^2}, \\
& \tau' = \frac{6000}{\sqrt{2}x_1x_2}, \quad \tau'' = \frac{MR}{J}, \\
& M = 6000\left(14 + \frac{x_2}{2}\right), \\
& R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2}, \\
& J = 2\sqrt{2}x_1x_2\left[\frac{x_2^2}{4} + \left(\frac{x_1+x_3}{2}\right)^2\right], \\
& \sigma(x) = \frac{504000}{x_3^2x_4}, \quad \delta(x) = \frac{65856000}{(30\times10^6)x_3^2x_4}, \\
& P_c(x) = \frac{4.013(30\times10^6)}{196}\sqrt{\frac{x_3^2x_4^6}{36}} \\
& \quad \times\left(1 - \frac{x_3}{28}\sqrt{\frac{30\times10^6}{4(12\times10^6)}}\right), \\
& 0.1 \le x_1 \le 2, \quad 0.1 \le x_2 \le 10, \\
& 0.1 \le x_3 \le 10, \quad 0.1 \le x_4 \le 2.
\end{aligned}
\tag{20}
$$

Where τ' denotes the primary shear stress, τ'' represents the secondary shear stress, P_c refers to the critical buckling load of the bar, M is the bending moment, R is the distance from the weld-group centroid to the critical point, and J is the polar moment of inertia.

© 2026 NSP
Natural Sciences Publishing Cor.

This benchmark problem has been addressed by numerous optimization methods, including JS, HJSPSO, GWO, RSO, the hybrid genetic–firefly algorithm (H-GA-FA) [14], the society and civilization (SC) algorithm [53], as well as MVO [42], SCA [42], GSA [42], GA [42], HS [42], and PSO [42].

The comparison of the best solutions obtained by HJSRSO and the other optimization algorithms is presented in Tables 7 and 8. The proposed HJSRSO algorithm achieves the highest performance, obtaining the best solution $f(x) = 1.724852309$, followed by MVO in second place and HJSPSO in third. Table 8 summarizes the statistical results, including the best, worst, median, mean, and standard deviation of the solutions. These indicators demonstrate that HJSRSO consistently delivers more accurate and stable solutions compared to the other algorithms, confirming its superior reliability and overall effectiveness in solving the welded beam problem.

(C)pressure vessel design problem

This design issue is specific to a cylindrical vessel with hemispherical heads at both ends, as Figure 5 illustrates [14]. It was suggested in [54] and seeks to reduce the whole cost, which includes material, welding, and forming expenses. The vessel's cylindrical section length L (x_4), inner radius R (x_3), head thickness T_h (x_2), and shell thickness T_s (x_1) are its four design variables. While the other two variables, T_s and T_h, are anticipated to be integer multiples of 0.0625 inches, the available thickness of the rolled steel plates, the two variables, R and L are continuous. The mathematical expression for this problem is:

$$\begin{aligned}
\text{Minimize} \quad & f(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 \\
& + 3.1661x_1^2x_4 + 19.84x_1^2x_3 \\
\text{Subject to:} \quad & -x_1 + 0.0193x_3 \le 0, \quad -x_2 + 0.00954x_3 \le 0, \\
& -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0, \\
& x_4 - 240 \le 0, \\
& 1 \times 0.0625 \le x_1, x_2 \le 99 \times 0.0625, \\
& 10 \le x_3, x_4 \le 200.
\end{aligned} \tag{21}$$

Table 9 presents the comprehensive statistical results of the JS, HJSPSO, GWO, RSO, WOA [48], hybrid PSO (HPSO) [55], HHO [50], MVO [42], SCA [42], GSA [42], GA [42], HS [42], PSO [42], and HJSRSO algorithms for the pressure vessel design problem illustrated in Figure 5. For clarity, Tables 9 and 10 summarize the optimized design variables along with the evaluation metrics, including the best (optimal), mean, worst, median, rank, and standard deviation of the total cost obtained from 30 independent runs of each algorithm. The best values for each performance indicator are underlined in bold. The results unequivocally demonstrate the HJSRSO algorithm's great capacity to provide stable and high-quality solutions by consistently outperforming the rival approaches across all statistical criteria. GA comes in second among the remaining algorithms, with PSO coming in third. Because the HJSRSO algorithm consistently finds the best solution in this situation, these results further validate the method's efficacy, robustness, and superiority in solving the pressure vessel design problem.

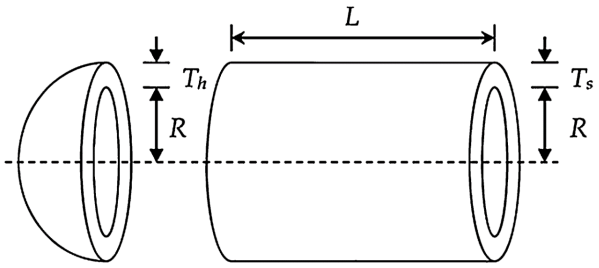


Fig. 5: Pressure vessel design problem [14].

(D)Speed reducer design problem

The objective of this design problem is to minimize the overall weight of the speed reducer while satisfying multiple design constraints, including limitations on gear tooth bending stress, surface contact stress, transverse deflections of the shafts, and shaft stress levels. The problem involves seven design variables, the width of face $b\,(x_1)$, the teeth module $m\,(x_2)$, number of the teeth in the pinion $z\,(x_3)$, the first shaft length between bearings $l_1\,(x_4)$, the second shaft length between bearings $l_2\,(x_5)$, the first shaft diameter $d_1\,(x_6)$ and the second shaft $d_2\,(x_7)$ as in Figure 6 [52]. Mathematically, this problem can be described as in Equation 22.

The speed reducer design problem has been addressed using several well-known optimization algorithms, including JS, HJSPSO, GWO, RSO, SHO [42], MVO [42], SCA [42], GSA [42], GA [42], HS [42], and PSO [42]. Table 11 provides a comparative analysis of the best solutions obtained by these methods and the proposed HJSRSO algorithm. In addition, the comprehensive statistical performance of all algorithms is reported in Table 12.

As observed in Table 11, the proposed HJSRSO approach achieves the minimum objective function value, with a best solution of $f(x) = 2994.471066$, outperforming all competing algorithms. Additionally, Table 12's statistical results, which include metrics like the best, mean, worst, median, and standard deviation, show that HJSRSO regularly yields more accurate, stable, and dependable solutions than the other approaches. These findings confirm the superior optimization capability and robustness of the proposed algorithm in solving the speed reducer design problem.

(E)Design of gas transmission compressor problem

The main goal of the gas transmission compressor design problem is to minimize the objective function by optimizing four important design variables, which are as fol-

© 2026 NSP
Natural Sciences Publishing Cor.

$$\text{Minimize} \quad f(x) = 0.7854x_1x_2^2(3.3333x_3^2 + 14.9334x_3 - 43.0934) - 1.508x_1(x_6^2 + x_7^2) + 7.4777(x_6^3 + x_7^3) + 0.7854(x_4x_6^2 + x_5x_7^2)$$

$$\text{Subject to:} \quad \frac{27}{x_1x_2^2x_3} - 1 \le 0, \quad \frac{397.5}{x_1x_2^2x_3^2} - 1 \le 0, \quad \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \le 0, \quad \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \le 0,$$

$$\frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6}}{110x_6^3} - 1 \le 0, \quad \frac{\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6}}{85x_7^3} - 1 \le 0,$$

$$\frac{x_2x_3}{40} - 1 \le 0, \quad \frac{5x_2}{x_1} - 1 \le 0, \quad \frac{x_1}{12x_2} - 1 \le 0, \quad \frac{1.5x_6 + 1.9}{x_4} - 1 \le 0, \quad \frac{1.1x_7 + 1.9}{x_5} - 1 \le 0,$$

$$2.6 \le x_1 \le 3.6, \quad 0.7 \le x_2 \le 0.8, \quad 17 \le x_3 \le 28, \quad 7.3 \le x_4 \le 8.3,$$

$$7.3 \le x_5 \le 8.3, \quad 2.9 \le x_6 \le 3.9, \quad 5 \le x_7 \le 5.5. \tag{22}$$

Table 7: Performance comparisons on the welded beam design problem

Algorithms	Optimum variables				Optimum cost
	x_1	x_2	x_3	x_3	
HJSRSO	**0.2057**	**3.4705**	**9.0366**	**0.2057**	**1.724852309**
JS	0.205678	3.475403	9.036964	0.206229	1.726995
HJSPSO	0.205563	3.474846	9.035799	0.205811	1.725661
GWO	0.205	3.4875	9.0365	0.2057	1.726062783
RSO	0.2032	4.3711	7.6861	0.2897	2.167135267
H-GA-FA	0.244368976	5.310427969	8.29147139	0.244368976	2.232693776
SC algorithm	0.244438276	6.237967234	8.288576143	0.244566182	2.3854347
MVO	0.205611	3.472103	9.040931	0.205709	1.725472
SCA	0.204695	3.536291	9.00429	0.210025	1.759173
GSA	0.147098	5.490744	10	0.217725	2.172858
GA	0.164171	4.032541	10	0.223647	1.873971
HS	0.206487	3.635872	10	0.203249	1.83625
PSO	0.197411	3.315061	10	0.201395	1.820395

Table 8: Evaluation of statistical outcomes for the welded beam design problem

Algorithms	WORST	MEAN	BEST	STD	MEDIAN	Rank
HJSRSO	**1.724852309**	**1.724852309**	**1.724852309**	**0**	**1.724852309**	1
JS	1.727563	1.72712801	1.726994	0.00115711	1.72708701	5
HJSPSO	1.726064	1.725828	1.725661	0.000287	1.725787	3
GWO	1.729081721	1.726836052	1.726062783	0.000843043	1.72654123	4
RSO	6.086111044	3.591197935	2.167135267	1.254390129	3.268260016	10
H-GA-FA	2.23304876	2.232887524	2.232693776	9.91E-03	Not available	12
SC algorithm	6.3996785	3.2551371	2.3854347	9.59E-01	Not available	13
MVO	1.741651	1.72968	1.725472	0.004866	1.72742	2
SCA	1.873408	1.817657	1.759173	0.027543	1.820128	6
GSA	3.003657	2.544239	2.172858	0.255859	2.495114	11
GA	2.320125	2.11924	1.873971	0.03482	2.097048	9
HS	2.035247	1.363527	1.83625	0.139485	1.9357485	8
PSO	3.048231	2.23031	1.820395	0.324525	2.244663	7

lows: length between compressor stations $L = x_1$, compression ratio, which indicates the compressor's inlet pressure $r = x_2$, and inner pipe diameter $D = x_3$. This problem is shown and formulated in Figure 7 and Equation (23).

$$\text{Minimize} \quad f(x) = 8.61 \times 10^5 x_1^{0.5} x_2 x_3^{-2/3} x_4^{-0.5} + 3.69 \times 10^4 x_3 + \frac{7.72 \times 10^8 x_2^{0.219}}{x_1} - \frac{765.43 \times 10^6}{x_1}$$

$$\text{Subject to:} \quad x_4x_2^{-2} + x_2^{-2} - 1 \le 0,$$

$$20 \le x_1 \le 50, \quad 1 \le x_2 \le 10,$$

$$20 \le x_3 \le 45, \quad 0.1 \le x_4 \le 60. \tag{23}$$

© 2026 NSP
Natural Sciences Publishing Cor.

Table 9: Evaluation results for the pressure vessel design problem

Algorithms	Optimum variables				Optimum cost
	x_1	x_2	x_3	x_4	
HJSRSO	**0.7855**	**0.3883**	**40.6952**	**194.8443**	**5889.3374**
JS	0.7803	0.3866	40.4306	198.5234	5893.016255
HJSPSO	0.7818	0.3868	40.5018	197.5428	5894.81735
GWO	0.7792	0.3854	40.3658	199.3784	5898.473046
RSO	1.0025	2.2091	51.0646	90.1176	12453.39708
WOA	0.8519145	0.5603772	43.42803	160.8293	6314.9267
HPSO	0.8125	0.4375	42.0984	176.6366	6059.7143
HHO	1.0860800	0.5215510	54.99250	63.0875	5972.4547
MVO	0.845719	0.418564	43.81627	156.38164	6011.5148
SCA	0.817577	0.417932	41.74939	183.5727	6137.3724
GSA	1.0858	0.949614	49.345231	169.48741	11550.2976
GA	0.752362	0.39954	40.452514	198.00268	5890.3279
HS	1.099523	0.906579	44.456397	179.65887	6550.023
PSO	0.778961	0.384683	40.320913	200	5891.3879

Table 10: Evaluation of statistical outcomes for the pressure vessel design problem

Algorithms	WORST	MEAN	BEST	STD	MEDIAN	RANK
HJSRSO	**5963.799063**	**5910.58132**	**5889.3374**	**20.53654787**	**5895.006821**	**1**
JS	6016.397699	5933.686035	5893.016255	33.8185511	5903.285577	4
HJSPSO	5965.146769	5913.876408	5894.81735	23.04890482	5903.259963	5
GWO	6799.634628	5988.29517	5898.473046	285.2577728	5929.438322	6
RSO	199679.8321	67555.982	12453.39708	61595.21114	48129.81281	14
WOA	8662.6477	7300.9278	6314.9267	478,781.6422	Not available	11
HPSO	6288.677	6099.9323	6059.7143	86.2	Not available	9
HHO	7306.5959	6715.7933	5972.4547	175,488.7714	Not available	7
MVO	7250.917	6477.305	6011.5148	327.007	6397.4805	8
SCA	6512.3541	6326.7606	6137.3724	126.609	6318.3179	10
GSA	33226.2526	23342.2909	11550.2976	5790.625	24010.0415	13
GA	7005.75	6264.0053	5890.3279	496.128	6112.6899	2
HS	8005.4397	6643.987	6550.023	657.523	7586.0085	12
PSO	7394.5879	6531.5032	5891.3879	534.119	6416.1138	3

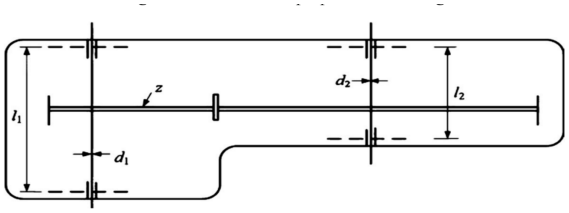


Fig. 6: Speed reducer design problem [52].

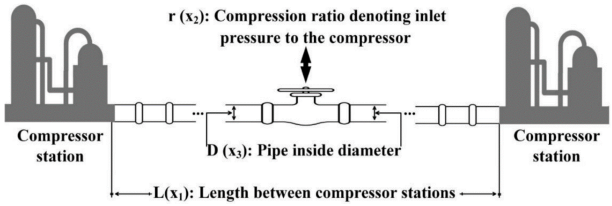


Fig. 7: Gas transmission compressor problem design.

This problem is solved by many algorithms such as JS, HJSPSO, GWO, RSO, continuous genetic algorithm (CGA) [56], whale optimization algorithm (WOA) [56], Lévy-flight moth-flame optimization (LMFO) [56], water-cycle–moth-flame optimization (WCMFO) [56], chimp optimization algorithm (ChOA) [56], arithmetic optimization algorithm (AOA) [56], sine-cosine moth-flame optimization (SMFO) [56], improved moth-flame optimization (I-MFO) [56], fruit fly algorithm (FFA) [56], and chaotic fruit fly algorithm (CFFA) [56].

Table 13 presents a comparison of the best solutions obtained by HJSRSO and the competing algorithms. As shown, the proposed method achieves the lowest objective function value, $f(x) = 2964895.427$, outperforming all other approaches. In addition, Table 14 reports the statistical performance metrics for the optimization process, including the best, worst, mean, median, rank, and standard deviation values.

The proposed algorithm demonstrates dominant performance across all evaluation criteria by achieving the best values in terms of mean, standard deviation, worst, median, and best results. This consistent superiority in both central tendency and dispersion measures highlights the algorithm's strong robustness, and reduced solution variability.

(F)The three-bar truss design problem

Finding the ideal cross-sectional areas $A_1 = A_3 = x_1$ and $A_2 = x_2$ that minimize the volume $f(x)$ of a statically loaded truss structure, subject to acceptable stress limitations σ, is the goal of the three-bar truss design problem, a classical engineering optimization job. Three constraint functions and two choice variables are used in the mathematical modeling

© 2026 NSP
Natural Sciences Publishing Cor.

Table 11: Evaluation results for speed reducer design problem

Algorithms	Optimum variables							Optimum cost
	x_1	x_2	x_3	x_4	x_5	x_6	x_7	
HJSRSO	**3.5**	**0.7**	**17**	**7.3**	**7.7153**	**3.3502**	**5.2867**	**2994.471066**
JS	3.5	0.7	17	7.3	7.8	3.350214	5.2866832	2996.348167
HJSPSO	3.5	0.7	17	7.3	7.715772	3.350218	5.286654	2994.482453
GWO	3.5029	0.7	17	7.3246	7.733	3.3509	5.2897	2998.323842
RSO	3.5786	0.7094	23.4678	8.0858	7.8499	3.4712	5.4754	1.19715E+12
SHO	3.50159	0.7	17	7.3	7.8	3.35127	5.28874	2998.5507
MVO	3.508502	0.7	17	7.392843	7.816034	3.358073	5.286777	3002.928
SCA	3.508755	0.7	17	7.3	7.8	3.46102	5.289213	3030.563
GSA	3.6	0.7	17	8.3	7.8	3.369658	5.289224	3051.12
GA	3.510253	0.7	17	8.35	7.8	3.362201	5.287723	3067.561
HS	3.520124	0.7	17	8.37	7.8	3.36697	5.288719	3029.002
PSO	3.500019	0.7	17	8.3	7.8	3.352412	5.286715	3005.763

Table 12: Evaluation of statistical outcomes for speed reducer design problem

Algorithms	WORST	MEAN	BEST	STD	MEDIAN	RANK
HJSRSO	**2994.471066**	**2994.471066**	**2994.471066**	**4.55E-13**	**2994.471066**	**1**
JS	2996.348204	2996.348174	2996.348167	6.40E-06	2996.35	3
HJSPSO	2999.652444	2996.769019	2994.482453	1.56E+00	2996.589	2
GWO	3006.921362	3003.035261	2998.323842	2.515505359	3003.449058	4
RSO	4.87703E+12	1.37815E+12	1.19715E+12	1.4306E+12	86561955858	12
SHO	3003.889	2999.64	2998.5507	1.93193	2999.187	5
MVO	3060.958	3028.841	3002.928	13.0186	3027.031	6
SCA	3104.779	3065.917	3030.563	18.0742	3065.609	9
GSA	3363.873	3170.334	3051.12	92.5726	3156.752	10
GA	3313.199	3186.523	3067.561	17.1186	3198.187	11
HS	3619.465	3295.329	3029.002	57.0235	3288.657	8
PSO	3211.174	3105.252	3005.763	79.6381	3105.252	7

Table 13: Evaluation results for gas transmission compressor problem design

Algorithms	Optimum variables				Optimum cost
	x_1	x_2	x_3	x_4	
HJSRSO	**49.99999652**	**1.178255198**	**24.5935508**	**0.388285316**	**2964895.427**
JS	49.98393722	1.178308334	24.57958165	0.388410217	2964901.191
HJSPSO	49.99844832	1.178261777	24.60415228	0.388300796	2964896.092
GWO	49.91713522	1.178008935	24.6142145	0.387702178	2964927.486
RSO	20	1.049320293	20	0.1	3209716.735
CGA	49.97	20.01	31.47	49.83	17350230
WOA	50	1.18	24.86	0.39	2965002
LMFO	49.46	1.18	24.64	0.39	2965456
WCMFO	50	1.18	24.61	0.39	2964897
AOA	50	1.23	20	0.51	3014615
SMFO	23.66	1.09	23.66	0.19	3052254
I-MFO	50	1.18	24.6	0.39	2964896
FFA	49.866	1.1832	23.5344	0.3999	2966272
CFFA	49.9999	1.1782	24.5936	0.3882	2964895.472

Table 14: Evaluation of statistical outcomes for gas transmission compressor problem design

Algorithms	WORST	MEAN	BEST	STD	MEDIAN	RANK
HJSRSO	**2964929.713**	**2964904.343**	**2964895.427**	**10.32350807**	**2964901.244**	**1**
JS	2964936.088	2964911.995	2964901.191	12.51831268	2964905.726	3
HJSPSO	2964939.257	2964917.318	2964896.092	13.39927008	2964919.762	2
GWO	2965245.183	2965071.775	2964927.486	124.119364	2965060.422	4
RSO	3276008.555	3226954.107	3209716.735	21180.02902	3219549.423	5

of the problem. The problem's full mathematical formulation and comprehensive schematic are shown in Equation (24) and Figure 8, respectively.

Many algorithms such as JS, HJSPSO, GWO, RSO, CGA [56], WOA [56], LMFO [56], WCMFO [56], ChOA [56], AOA [56], SMFO [56], FFA [56], and CFFA [56] solve this problem.

Table 15 presents a comparison of the best solutions obtained for the three-bar truss design problem by HJSRSO and other reported algorithms, while Table 16 summarizes the corresponding statistical results. The findings indicate that HJSRSO outperforms all competing methods, achieving a lower total cost than any of the other algorithms and demonstrating its strong capability in identifying the

© 2026 NSP
Natural Sciences Publishing Cor.

Table 15: Evaluation results for three-bar truss problem design

Algorithms	Optimum variables x_1	x_2	Optimum cost
HJSRSO	**0.7888**	**0.408**	**263.8958453**
JS	0.7887	0.4083	263.8958533
HJSPSO	0.7886	0.4083	263.8958457
GWO	0.7881	0.4098	263.8963361
RSO	0.7743	0.4508	264.0785929
CGA	0.792428	0.397752	263.9077
WOA	0.787713	0.410977	263.89653
LMFO	0.791713	0.399909	263.92114
WCMFO	0.788472	0.408822	263.89589
AOA	0.792789	0.396906	263.92526
SMFO	0.792044	0.398859	263.90973
I-MFO	0.788792	0.407919	263.89585
FFA	0.779348	0.435289	263.9619
CFFA	0.788661	0.408286	263.89584

global optimum. Furthermore, all statistical criteria, including mean, standard deviation, worst, median, and best values, consistently show that the suggested approach performs better. This consistent superiority highlights not only higher solution quality but also reduced variability and enhanced robustness, and confirming the algorithm's reliable.

$$\begin{aligned} \text{Minimize} \quad & f(x) = H(x_2 + 2\sqrt{2}x_1) \\ \text{Subject to:} \quad & \frac{x_2}{2x_1x_2 + \sqrt{2}x_1^2}P - \sigma \le 0, \\ & \frac{x_2 + \sqrt{2}x_1}{2x_1x_2 + \sqrt{2}x_1^2}P - \sigma \le 0, \\ & \frac{1}{x_1 + \sqrt{2}x_2}P - \sigma \le 0, \\ & H = 100\,\text{cm}, \quad P = 2\,\text{KN/cm}^2, \\ & \sigma = 2\,\text{KN/cm}^2, \\ & 0 \le x_1, x_2 \le 1. \end{aligned} \tag{24}$$

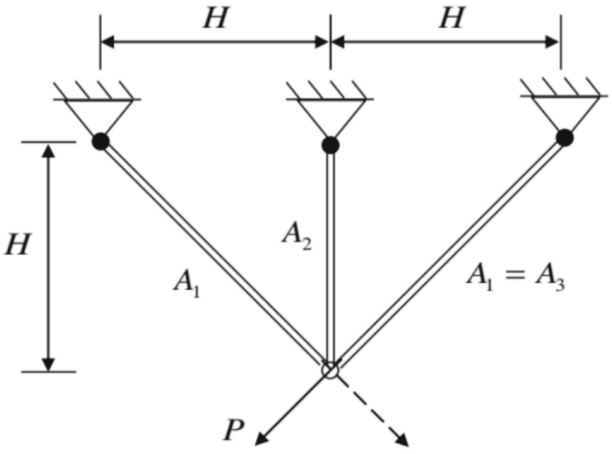


Fig. 8: Three-bar truss problem design.

(G)**Cantilever beam design problem**

This design challenge comprises five hollow parts with a square cross-section, as illustrated in Figure 9 [57]. While the thickness is regarded as constant, every constituent is expressed as a variable. Thus, there is only one perpendicular displacement constraint and five structural parameters. Figure 9 shows that node 1 is rigorously supported and node 6 is subject to a perpendicular stress. The goal of this issue is to minimize the beam's weight while adhering to a single constraint, which must be met by the final optimal design. This problem is expressed mathematically as follows:

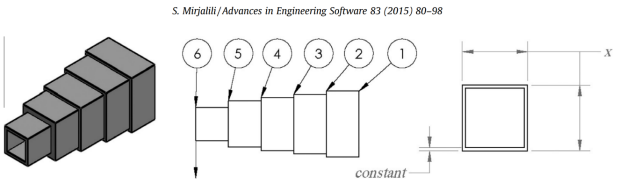


Fig. 9: Cantilever beam design problem [57].

$$\begin{aligned} \text{Minimize} \quad & f(x) = 0.6224x_1 + x_2 + x_3 + x_4 + x_5 \\ \text{Subject to:} \quad & \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} \le 1, \\ & 0.01 \le x_i \le 100, \quad i = 1,2,\ldots,5. \end{aligned} \tag{25}$$

Many algorithms such as JS, HJSPSO, GWO, RSO, H-GA-FA [14], enhanced leader PSO (ELPSO) [58], WOA [59], improved WOA based on a Levy flight trajectory (LWOA) [59].

Table 17 presents a comparison of the optimal design variables and the best objective function values obtained by HJSRSO and the other algorithms, while Table 18 reports the statistical performance of these methods for this problem. Although H-GA-FA achieves a slightly better best value, the proposed HJSRSO algorithm demonstrates superior overall performance in statistical terms, attaining the lowest mean, standard deviation, worst, and median values. This indicates reduced variability, enhanced robustness, and more convergence across multiple independent runs.

4.2.1 Convergence Performance Analysis of Seven Engineering Design Problems

In addition to the statistical analysis, the convergence curves further illustrate the performance differences among HJSRSO, JS, and RSO in Figure 10. Compared to both JS and RSO, HJSRSO converges more quickly and persistently toward high-quality solutions, according to the convergence behavior. Although JS has a great exploration capability in the early iterations, its comparatively weaker exploitation

© 2026 NSP
Natural Sciences Publishing Cor.

Table 16: Evaluation of statistical outcomes for three-bar truss problem design

Algorithm	WORST	MEAN	BEST	STD	MEDIAN	RANK
HJSRSO	**263.8959**	**263.8959**	**263.8958**	**2.21E-05**	**263.8959**	**1**
JS	263.896	263.8959	263.8959	6.64E-05	263.8959	3
HJSPSO	263.896	263.8959	263.8958	4.94E-05	263.8959	2
GWO	263.9018	263.8981	263.8963	0.00202	263.8973	4
RSO	267.8053	265.6341	264.0786	1.485546	265.4219	5

Table 17: Evaluation results for the cantilever design problem.

Algorithms	Optimum variables					Optimum cost
	x_1	x_2	x_3	x_4	x_5	
HJSRSO	**6.019**	**5.3143**	**4.4907**	**3.5001**	**2.1496**	**1.339956513**
JS	6.0138	5.308	4.4965	3.5009	2.1546	1.339956925
HJSPSO	6.017	5.3094	4.4927	3.5016	2.1529	1.339958384
GWO	6.0077	5.3007	4.5105	3.502	2.1529	1.339971141
RSO	7.4251	6.3025	3.096	6.3692	3.3502	1.656284797
H-GA-FA	6.017563478	5.311392298	4.497733036	3.494192221	2.152821173	1.336523225
ELPSO	6.016	5.3092	4.4943	3.5015	2.1527	1.34
WOA	5.9518	5.0649	4.6744	3.5476	2.2916	1.3435
LWOA	6.0105	5.347	4.4468	3.5165	2.1546	1.34

Table 18: Evaluation of statistical outcomes for the cantilever design problem.

Algorithms	WORST	MEAN	BEST	STD	MEDIAN	RANK
HJSRSO	**1.339961753**	**1.339958341**	1.339956513	**1.02E-06**	**1.339957703**	2
JS	1.33996297	1.339959994	1.339956925	1.72E-06	1.33996042	3
HJSPSO	1.339961971	1.339960077	1.339958384	2.53E-06	1.339960237	4
GWO	1.340076403	1.3400211	1.339971141	3.61E-05	1.340014551	5
RSO	3.947530547	2.68318955	1.656284797	0.726042981	2.82259472	9
H-GA-FA	1.34367893	1.340121354	**1.336523225**	0.0008745	Not available	**1**
ELPSO	1.34	1.34	1.34	0.00001755	Not available	6
WOA	1.4865	1.3668	1.3435	0.027	Not available	8
LWOA	1.349137	1.34101	1.34	0.002208	Not available	6

causes its convergence rate to slow down in later phases. RSO, on the other hand, exhibits quick exploitation behavior but is more prone to early convergence, particularly when stuck in local optima. HJSRSO keeps the search process balanced throughout the optimization run by skillfully combining the exploration power of JS with the exploitation effectiveness of RSO. Consequently, HJSRSO achieves the best convergence performance, reaching lower objective values in fewer iterations and demonstrating superior convergence accuracy compared with its constituent algorithms.

4.2.2 Engineering Design Problems' Friedman Test and Wilcoxon signed-rank Test

In this section, the Friedman test [46] is applied to evaluate and compare the performance of the proposed and competing algorithms on the engineering design problems (EDPs). The Friedman test is a nonparametric statistical method that assesses algorithm performance by assigning average ranks across multiple problems, where a smaller mean rank indicates better overall performance. In addition, the test provides a p-value to determine whether the observed differences among algorithms are statistically significant. When the p-value is less than 0.05 ($p < 0.05$), it suggests that the performance variations are not due to random chance.

The results of the Friedman test for the EDPs are summarized in Table 19. As shown in the table, the p-values for all considered EDPs are below the 0.05 threshold, confirming the presence of statistically significant differences among the evaluated algorithms. Additionally, the suggested HJSRSO algorithm consistently achieves the lowest mean rank across all EDPs, demonstrating it is better and more reliable performance in comparison to the other optimization techniques.

Additionally, for every engineering design problem (EDP), the pairwise statistical differences between the suggested HJSRSO algorithm and the other competing approaches are examined using the Wilcoxon signed-rank test [47]. By comparing two algorithms at a time, this nonparametric test establishes if the observed performance differences are statistically significant using the corresponding p-values. Table 20 shows the results of the Wilcoxon signed-rank test for the EDPs, where P-R stands for the total of positive ranks and N-R for the sum of negative ranks. Some algorithm comparisons result in p-values greater than 0.05, as shown in Table 20. This can be explained by inadequate or missing performance data supplied by some competing algorithms. In spite of this, it is evident from the results that HJSRSO consistently obtains higher P-R values than N-R in all EDPs. In the majority of pairwise comparisons, HJSRSO performs better than the other algorithms, as evi-

© 2026 NSP
Natural Sciences Publishing Cor.

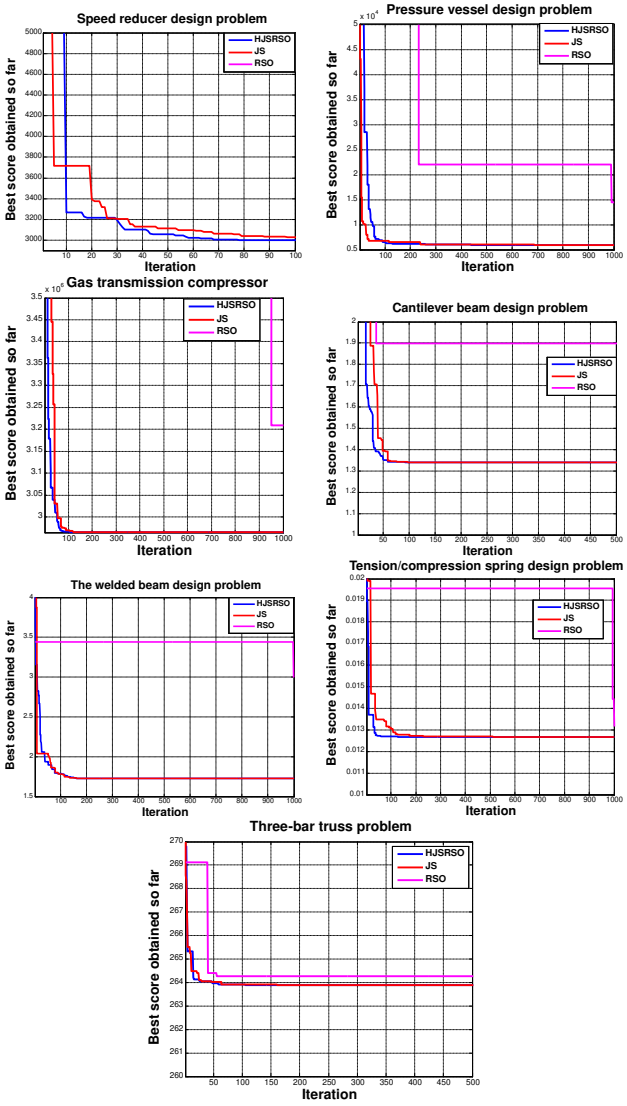


Fig. 10: Convergence curves for seven engineering design problems.

denced by the dominance of positive ranks. Consequently, it can be inferred from the Wilcoxon signed-rank test results that HJSRSO performs better than most of the algorithms used to solve the engineering design challenges.

5 Conclusion and Future Work

This paper presents a new hybrid metaheuristic algorithm called HJSRSO that combines the Rat Swarm Optimizer (RSO) and the Jellyfish Search Optimizer (JS). Overall optimization performance is intended to be improved by the proposed hybrid framework, which compensates for the inherent shortcomings of the different approaches. Specifically, the RSO mechanism, which has shown better exploitation performance than many other metaheuristics now in use, effectively addresses the weak exploitation capabilities of the JS algorithm. Additionally, JS's exploration phase incorporates a sine-based method to enhance the algorithm's capacity for global search. The efficacy of the suggested method is confirmed by extensive trials carried out on a wide range of benchmark problems. Seven engineering design problems (EDPs) and fifty well-known benchmark functions are included in the test suite. Eight well-known metaheuristic algorithms—JS, RSO, HJSPSO, WOA, GWO, MVO, SCA, and PSO—were compared against HJSRSO for the benchmark functions.

Rank-based analysis and comparison results show that HJSRSO outperformed all competing algorithms, achieving the top rank in 28 out of 50 benchmark functions. Additionally, by contrasting HJSRSO with a number of cutting-edge and competitive optimization approaches, such as H-GA-FA, LWOA, WCMFO, CFAA, and SMFO, its performance on engineering design issues was further evaluated. The suggested method obtained the highest overall rank and produced better solutions in six of the seven EDPs. These findings unequivocally show that HJSRSO offers a reliable and effective optimization mechanism with seamless position updating, making it ideal for resolving challenging engineering and practical optimization issues. Lastly, the Friedman test and Wilcoxon signed rank test were used to examine the statistical efficiency of the HJSRSO, and the results showed that the suggested HJSRSO performed better than alternative algorithms.

Because meta-heuristic methods are random approaches, the proposed method HJSRSO has the potential drawback of not guaranteeing an improvement in computing speed or accuracy while solving any optimization problem, similar to other meta-heuristic algorithms where the computational effectiveness and solution quality of the HJSRSO depend on the nature and complexity of the problem. Although this study tackles some of JS's shortcomings and enhances its efficiency, it is unable to determine the global optimum for some functions.

Three directions will be the focus of our future work: (i) expanding HJSRSO to solve many-objective optimization problems; (ii) Numerous significant optimization issues, including resource allocation, economical load transfer, unit commitment, wind farm optimization of wind turbines, and real-time applications, can be resolved with the proposed method; and (iii) creating fresh hybrid optimization methods that draw inspiration from the suggested framework.

Conflict of Interests

The authors declare that they have no competing or conflict of interests.

© 2026 NSP
Natural Sciences Publishing Cor.

Table 19: Engineering Design Problems' Friedman test

Tension/Compression Spring Design Problem			
Algorithm	Men Rank		
HJSRSO	1.25		
JS	4.75		
HJSPSO	5		
GWO	5	**Test Statistics**	
RSO	10.75	N	4
WOA	8	Chi-Square	36.771
MFO	5	df	13
SHO	3.25	**p-value**	0.000
MVO	11.75		
SCA	7.25		
GSA	9.5		
GA	12		
HS	9.5		
PSO	12		
Welded Beam Design Problem			
Algorithm	Mean Rank		
HJSRSO	1.25		
JS	4.25		
HJSPSO	2.50		
GWO	3.75	**Test Statistics**	
RSO	12.00	N	4
H-GA-FA	9.00	Chi-Square	41.802
SC algorithm	12.50	df	12
MVO	4.50	**p-value**	0.000
SCA	6.50		
GSA	10.50		
GA	8.50		
HS	6.25		
PSO	9.50		
Three-bar truss problem design			
Algorithm	Mean Rank		
HJSRSO	1.00	**Test Statistics**	
JS	2.20	N	5
HJSPSO	2.80	Chi-Square	19.360
GWO	4.00	df	4
RSO	5.00	**p-value**	0.001
gas transmission compressor design			
Algorithm	Mean Rank		
HJSRSO	1.00	**Test Statistics**	
JS	3.00	N	5
HJSPSO	2.00	Chi-Square	20.000
GWO	4.00	df	4
RSO	5.00	**p-value**	0.000

Pressure Vessel Design Problem			
Algorithm	Mean Rank		
HJSRSO	1.00		
JS	3.25		
HJSPSO	2.75		
GWO	5.50	**Test Statistics**	
RSO	13.50	N	4
WOA	12.25	Chi-Square	44.571
HPSO	5.50	df	13
HHO	10.00	**p-value**.	0.000
MVO	7.75		
SCA	6.75		
GSA	12.50		
GA	5.75		
HS	10.75		
PSO	7.75		
Speed reducer design problem			
Algorithm	Mean Rank		
HJSRSO	1.00		
JS	2.20		
HJSPSO	2.80	**Test Statistics**	
GWO	4.80	N	5
RSO	12.00	Chi-Square	52.600
SHO	4.20	df	11
MVO	6.00	**p-value**	0.000
SCA	7.60		
GSA	9.80		
GA	9.40		
HS	10.00		
PSO	8.20		
Cantilever design problem			
Algorithm	Mean Rank		
HJSRSO	1.25		
JS	2.50		
HJSPSO	3.00	**Test Statistics**	
GWO	5.00	N	4
RSO	9.00	Chi-Square	28.342
H-GA-FA	4.75	df	8
ELPSO	4.63	**p-value**	0.000
WOA	8.00		
LWOA	6.88		

Acknowledgments

The authors extend their appreciation to Prince Sattam bin Abdulaziz University for funding this research work through the project number (PSAU/2025/01/36283).

References

[1] Michael, B.-B. (2008). *Nonlinear optimization with engineering applications (Springer Optimization and Its Applications Book Series)*. Springer-Verlag.

[2] Rao, S. S. (2009). *Engineering optimization: Theory and practice*(3rd ed.). Wiley.

[3] Michalewicz, Z. (1994). Evolutionary computation techniques for nonlinear programming problems. *International Transactions in Operational Research*, *1*, 223–240.

[4] Onwubolu, G. C., & Babu, B. V. (2004). *New optimization techniques in engineering*(Vol. *141*). Springer Science & Business Media.

[5] Alrefaei, M. H., & Diabat, A. H. (2009). A simulated annealing technique for multi-objective simulation optimization. *Applied Mathematics and Computation*, *215*, 3029–3035.

[6] Parouha, R. P., & Das, K. N. (2016). A memory-based differential evolution algorithm for unconstrained optimization. *Applied Soft Computing*, *38*, 501–517.

[7] El-Shorbagy, M. A., Rashad, A. M., EL-Mky, H. A., & Ahmed, A. A. (2025). Enhanced Beluga Whale Optimization Algorithm for Handling Global Optimization and Engineering Applications. *International Journal of Applied and Computational Mathematics*, *11*(4), 122.

[8] El-Shorbagy, M. A., & Ahmed, A. A. (2025). Studying Complex Nonlinear Systems of Equations by Modified Beluga Whale Optimization Algorithm with Analysis. *Results in Engineering*, 105236.

[9] El-Shorbagy, M. A., Ayoub, A. Y., Mousa, A. A., & El-Desoky, I. M. (2019). An enhanced genetic algorithm with new mutation for cluster analysis. *Computational Statistics*, *34*, 1355–1392.

[10] Aljaidi, M., Jangir, P., Arpita, Agrawal, S. P., Pandya, S. B., Parmar, A., ... & Jangid, R. (2025). Hybrid Beluga whale and jellyfish search optimizer for optimizing proton exchange membrane fuel cell parameter estimation. *Ionics*, 1-33.

[11] El-Shorbagy, M. A., & Mousa, A. A. (2017). Chaotic particle swarm optimization for imprecise combined economic and

© 2026 NSP
Natural Sciences Publishing Cor.

Table 20: Engineering Design Problems' Wilcoxon Signed-rank Test

Design problem	Compared Algorithms		Evaluations of Solution				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	4	0.068	≈
	HJSPSO	HJSRSO	0	0	4	0.068	≈
	GWO	HJSRSO	0	0	4	0.068	≈
	RSO	HJSRSO	0	0	4	0.068	≈
	WOA	HJSRSO	0	0	4	0.068	≈
	MFO	HJSRSO	0	0	4	0.068	≈
Tension/compression spring	SHO	HJSRSO	1	0	3	0.273	≈
	MVO	HJSRSO	0	0	4	0.068	≈
	SCA	HJSRSO	0	0	4	0.068	≈
	GSA	HJSRSO	0	0	4	0.068	≈
	GA	HJSRSO	0	0	4	0.068	≈
	HS	HJSRSO	0	0	4	0.068	≈
	PSO	HJSRSO	0	0	4	0.068	≈
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043	HJSRSO
	HJSPSO	HJSRSO	0	0	5	0.043	HJSRSO
	GWO	HJSRSO	0	0	5	0.043	HJSRSO
	RSO	HJSRSO	0	0	5	0.043	HJSRSO
	H-GA-FA	HJSRSO	0	0	4	≈	HJSRSO
Welded beam	SC algorithm	HJSRSO	0	0	4	≈	HJSRSO
	MVO	HJSRSO	0	0	5	0.043	HJSRSO
	SCA	HJSRSO	0	0	5	0.043	HJSRSO
	GSA	HJSRSO	0	0	5	0.043	HJSRSO
	GA	HJSRSO	0	0	5	0.043	HJSRSO
	HS	HJSRSO	1	0	45	≈	HJSRSO
	PSO	HJSRSO	0	0	5	0.043	HJSRSO
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043	HJSRSO
	HJSPSO	HJSRSO	0	0	5	0.043	HJSRSO
	GWO	HJSRSO	0	0	5	0.043	HJSRSO
	RSO	HJSRSO	0	0	5	0.043	HJSRSO
	WOA	HJSRSO	0	0	4	0.068	≈
Pressure vessel	HPSO	HJSRSO	0	0	4	0.068	≈
	HHO	HJSRSO	0	0	4	0.068	≈
	MVO	HJSRSO	0	0	5	0.043	HJSRSO
	SCA	HJSRSO	0	0	5	0.043	HJSRSO
	GSA	HJSRSO	0	0	5	0.043	HJSRSO
	GA	HJSRSO	0	0	5	0.043	HJSRSO
	HS	HJSRSO	0	0	5	0.043	HJSRSO
	PSO	HJSRSO	0	0	5	0.043	HJSRSO
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043	HJSRSO
	HJSPSO	HJSRSO	0	0	5	0.043	HJSRSO
	GWO	HJSRSO	0	0	5	0.043	HJSRSO
	RSO	HJSRSO	0	0	5	0.043	HJSRSO
Speed reducer	SHO	HJSRSO	0	0	5	0.043	HJSRSO
	MVO	HJSRSO	0	0	5	0.043	HJSRSO
	SCA	HJSRSO	0	0	5	0.043	HJSRSO
	GSA	HJSRSO	0	0	5	0.043	HJSRSO
	GA	HJSRSO	0	0	5	0.043	HJSRSO
	HS	HJSRSO	0	0	5	0.043	HJSRSO
	PSO	HJSRSO	0	0	5	0.043	HJSRSO
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043114447	HJSRSO
gas transmission compressor	HJSPSO	HJSRSO	0	0	5	0.043114447	HJSRSO
	GWO	HJSRSO	0	0	5	0.043114447	HJSRSO
	RSO	HJSRSO	0	0	5	0.043114447	HJSRSO
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043114447	HJSRSO
Three-bar truss	HJSPSO	HJSRSO	0	0	5	0.043114447	HJSRSO
	GWO	HJSRSO	0	0	5	0.043114447	HJSRSO
	RSO	HJSRSO	0	0	5	0.043114447	HJSRSO
	Compared Algorithms		**Evaluations of Solution**				
	Algorithm 1	Algorithm 2	**N-R**	≈	**P-R**	P-Value	**Best algorithm**
	JS	HJSRSO	0	0	5	0.043	HJSRSO
	HJSPSO	HJSRSO	0	0	5	0.043	HJSRSO
	GWO	HJSRSO	0	0	5	0.043	HJSRSO
Cantilever	RSO	HJSRSO	0	0	5	0.043	HJSRSO
	H-GA-FA	HJSRSO	1	0	3	0.465	≈
	ELPSO	HJSRSO	0	0	4	0.068	≈
	WOA	HJSRSO	0	0	4	0.068	≈
	LWOA	HJSRSO	0	0	4	0.068	≈

© 2026 NSP
Natural Sciences Publishing Cor.

emission dispatch problem. *Review of Information Engineering and Applications*, *4*, 20–35.

[12] Ni, R., Chen, H., Liang, X., He, M., Xia, Y., & Sun, L. (2025). Elman Network Classifier Based on Hyperactivity Rat Swarm Optimizer and Its Applications for AlSi10Mg Process Classification. *Processes*, *13*(9), 2802.

[13] Ashraf, A., Zhao, Q., Bangyal, W. H., Raza, M., & Iqbal, M. (2025). Female autism categorization using CNN based NeuroNet57 and ant colony optimization. *Computers in Biology and Medicine*, *189*, 109926.

[14] El-Shorbagy, M. A., & El-Refaey, A. M. (2022). A hybrid genetic–firefly algorithm for engineering design problems. *Journal of Computational Design and Engineering*, *9*(2), 706-730.

[15] Nadimi-Shahraki, M. H., Zamani, H., Asghari Varzaneh, Z., & Mirjalili, S. (2023). A systematic review of the whale optimization algorithm: theoretical foundation, improvements, and hybridizations. *Archives of Computational Methods in Engineering*, *30*(7), 4113-4159.

[16] Mousa, A. A., El-Shorbagy, M. A., & Farag, M. A. (2020). Steadystate sine cosine genetic algorithm-based chaotic search for nonlinear programming and engineering applications. *IEEE Access*, *8*, 212036–212054.

[17] El-Shorbagy, M. A., Eldesoky, I. M., Basyouni, M. M., Nassar, I., & El-Refaey, A. M. (2022). Chaotic search-based salp swarm algorithm for dealing with system of nonlinear equations and power system applications. *Mathematics*, *10*(9), 1368.

[18] Makhadmeh, S. N., Al-Betar, M. A., Doush, I. A., Awadallah, M. A., Kassaymeh, S., Mirjalili, S., & Zitar, R. A. (2023). Recent advances in Grey Wolf Optimizer, its versions and applications. *Ieee Access*, *12*, 22991-23028.

[19] Mousa, A. A., El-Shorbagy, M. A.,Mustafa, I., & Alotaibi, H. (2021). Chaotic search-based equilibrium optimizer for dealing with nonlinear programming and petrochemical application. *Processes*, *9*, 200.

[20] Mukilan, K., Rameshbabu, C., Baranitharan, B., Muthusamy, S., Ramamoorthi, P., Sadasivuni, K. K., ... & Khan, A. (2025). An efficient claim management assurance system using EPC contract based on improved monarch butterfly optimization models. *Neural Computing and Applications*, *37*(1), 169-184.

[21] Li, T., Meng, H., Wang, D., Fu, B., Shao, Y., & Liu, Z. (2025). An enhanced slime mould algorithm based on best–worst management for numerical optimization problems. *Biomimetics*, *10*(8), 504.

[22] Sun, Z., Sun, J., Xie, X., An, Z., Hong, Y., & Sun, Z. (2025). Multi-population mutative moth-flame optimization algorithm for modeling and the identification of PEMFC parameters. *Renewable Energy*, *240*, 122238.

[23] Zhang, Q., Sheng, J., Zhang, Q., Yang, Z., Xin, Y., Wang, B., ... & Alzheimer's Disease Neuroimaging Initiative. (2025). Optimizing deep learning with improved Harris Hawks optimization for Alzheimer's disease detection. *Artificial Intelligence Review*, *58*(10), 301.

[24] Al Malki, A., Rizk, M. M., El-Shorbagy, M. A., & Mousa, A. A. (2016). Hybrid genetic algorithm with K-means for clustering problems. *Open Journal of Optimization*, *5*, 71–83.

[25] Goel, R., & Maini, R. (2018). A hybrid of ant colony and firefly algorithms (HAFA) for solving vehicle routing problems. *Journal of Computational Science*, *25*, 28–37.

[26] El-Shorbagy, M. A.,Mousa, A. A., & Fathi,W. (2011). *Hybrid particle swarm algorithm for multiobjective optimization: Integrating particle swarm optimization with genetic algorithms formultiobjective optimization*. Lambert Academic Publishing.

[27] El-Shorbagy, M. A., & El-Refaey, A. M. (2020). Hybridization of grasshopper optimization algorithm with genetic algorithm for solving system of non-linear equations. *IEEE Access*, *8*, 220944–220961.

[28] Ekinci, S., Hekimo glu, B., Eker, E.,&Sevim, D. (2019). Hybrid firefly and particle swarm optimization algorithm for PID controller design of buck converter. In *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*(pp. 1–6).

[29] Wang, J., Yang, W., Du, P., & Niu, T. (2018). A novel hybrid forecasting system of wind speed based on a newly developed multi-objective sine cosine algorithm. *Energy Conversion and Management*, *163*, 134–150.

[30] Aljaidi, M., Jangir, P., Arpita, Agrawal, S. P., Pandya, S. B., Parmar, A., ... & Jangid, R. (2025). Hybrid Beluga whale and jellyfish search optimizer for optimizing proton exchange membrane fuel cell parameter estimation. *Ionics*, 1-33.

[31] Yuan, X., Hu, G., Zhong, J., & Wei, G. (2023). HBWO-JS: Jellyfish search boosted hybrid beluga whale optimization algorithm for engineering applications. *Journal of Computational Design and Engineering*, *10*(4), 1615-1656.

[32] Chou, J.-S.; Truong, D.-N. A novel metaheuristic optimizer inspired by behavior of jellyfish in ocean. Appl. Math. Comput. **2021**, 389, 125535.

[33] Rai, H.; Verma, H.K. Economic load dispatch using jellyfish search optimizer. In Proceedings of the 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT), Bhopal, India, 18–19 June 2021.

[34] Gouda, E.A.; Kotb, M.F.; El-Fergany, A.A. Jellyfish search algorithm for extracting unknown parameters of PEM fuel cell models: Steady-state performance and analysis. Energy **2021**, 221, 119836.

[35] Youssef, H.; Hassan, M.H.; Kamel, S.; Elsayed, S.K. Parameter estimation of single phase transformer using jellyfish search optimizer algorithm. In Proceedings of the 2021 IEEE International Conference on Automation/XXIV Congress of the Chilean
Association of Automatic Control (ICA-ACCA), Online, 22–26 March 2021; pp. 1–4.

[36] Shaheen, A.M.; Elsayed, A.M.; Ginidi, A.R.; Elattar, E.E.; El-Sehiemy, R.A. Effective automation of distribution systems with joint integration of DGs/ SVCs considering reconfiguration capability by jellyfish search algorithm. IEEE Access **2021**, 9, 92053–92069.

[37] Shaheen, A.M.; El-Sehiemy, R.A.; Alharthi, M.M.; Ghoneim, S.S.; Ginidi, A.R. Multi-objective jellyfish search optimizer for efficient power system operation based on multi-dimensional OPF framework. Energy **2021**, 237, 121478.

[38] Barshandeh, S.; Dana, R.; Eskandarian, P. A learning automata-based hybrid MPA and JS algorithm for numerical optimization problems and its application on data clustering. Knowl.-Based Syst. **2021**, 236, 107682.

[39] Manita, G.; Zermani, A. A modified jellyfish search optimizer with orthogonal learning strategy. Procedia Comput. Sci. **2021**, 192, 697–708.

© 2026 NSP
Natural Sciences Publishing Cor.

[40] Abdel-Basset, M.; Mohamed, R.; Chakrabortty, R.; Ryan, M.; El-Fergany, A. An improved artificial jellyfish search optimizer for parameter identification of photovoltaic models. Energies **2021**, 14, 1867.
[41] Abdel-Basset, M.; Mohamed, R.; Abouhawwash, M.; Chakrabortty, R.K.; Ryan, M.J.; Nam, Y. An improved jellyfish algorithm for multilevel thresholding of magnetic resonance brain image segmentations. Comput. Mater. Con. **2021**, 68, 2961–2977.
[42] Dhiman, G., Garg, M., Nagar, A., Kumar, V., & Dehghani, M. (2021). A novel algorithm for global optimization: Rat Swarm Optimizer. *Journal of Ambient Intelligence and Humanized Computing*, *12*(8), 8457-8482.
[43] Michalewicz, Z. (1994). Evolutionary computation techniques for nonlinear programming problems. *International Transactions in Operational Research*, *1*, 223–240.
[44] Ayyarao, T. S., Ramakrishna, N. S. S., Elavarasan, R. M., Polumahanthi, N., Rambabu, M., Saini, G., ... & Alatas, B. (2022). War strategy optimization algorithm: a new effective metaheuristic algorithm for global optimization. *Ieee Access*, *10*, 25073-25105.
[45] A. Kumar, G. Wu, M. Z. Ali, R. Mallipeddi, P. N. Suganthan, and S. Das, "A test-suite of non-convex constrained optimization problems from the real-world and some baseline results," *Swarm and Evolutionary Computation*, vol. 56, Article ID 100693, 2020.
[46] Derrac, Joaquín, Salvador García, Daniel Molina, and Francisco Herrera. "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms." *Swarm and Evolutionary Computation* 1, no. 1 (2011): 3-18.
[47] García, Salvador, Alberto Fernández, Julián Luengo, and Francisco Herrera. "Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power." *Information sciences* 180, no. 10 (2010): 2044-2064.
[48] Mirjalili, S.; Lewis, A. The whale optimization algorithm. Adv. Eng. Softw. **2016**, 95, 51–67.
[49] Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowl.-Based Syst. **2015**, 89, 228–249.
[50] Heidari, A.A.; Mirjalili, S.; Faris, H.; Aljarah, I.; Mafarja, M.; Chen, H.L. Harris hawks optimization: Algorithm and applications. Future Gener. Comput. Syst. **2019**, 97, 849–872.
[51] Cheng, M.-Y., & Prayogo, D. (2014). Symbiotic organisms search: A new metaheuristic optimization algorithm. *Computers and Structures*, *139*, 98–112.
[52] A.Sadollah, A. Bahreininejad, H. Eskandar, M. Hamdi, Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems, Applied Soft Computing 13(5) (2013) 2592–2612.
[53] Ray, T., Liew, K.M.: Society and civilization: an optimization algorithm based on the simulation of social behavior. IEEE Trans. Evol. Comput. 7(4), 386–396 (2003).
[54] B. Kannan, S. N. Kramer, An augmented Lagrange multiplier based method for mixed integer discrete continuous optimization and its applications to mechanical design, Journal of Mechanical Design 116 (1994) 405–411.
[55] Q. He, L. Wang, A hybrid particle swarm optimization with a feasibility-based rule for constrained optimization, Applied Mathematics and Computation 186 -(2007) 1407–1422.
[56] El-Shorbagy, M. A. (2022). Chaotic fruit fly algorithm for solving engineering design problems. *Complexity*, *2022*(1), 6627409.
[57] S. Mirjalili, The ant lion optimizer, Advances in Engineering Software 83(1) (2015) 80–98.
[58] Jordehi, A. R. (2015). Enhanced leader PSO (ELPSO), a new PSO variant for solving global optimization problems. *Applied Soft Computing Journal*, *26*, 401–417.
[59] Zhou, Y., Ling, Y., & Luo, Q. (2018). Lévy flight trajectory-based whale optimization algorithm for engineering optimization. *Engineering Computations*, *35*(7), 2406-2428.

© 2026 NSP
Natural Sciences Publishing Cor.