A Hybrid Approach of Spotted Hyena Optimization Integrated with Quadratic Approximation for Training Wavelet Neural Network

Nibedan Panda1,2 · Santosh Kumar Majhi1 · Rosy Pradhan3

Received: 25 August 2021 / Accepted: 29 December 2021 / Published online: 1 February 2022 © King Fahd University of Petroleum & Minerals 2022

Abstract

Spotted hyena optimization (SHO) is one of the newly evolved swarm-based metaheuristic optimization methods based on the social life cycle of hyenas. In recent times SHO is being applied to various engineering applications as well as to solve real-life complications. In this paper, we have hybridized SHO with quadratic approximation operator (QAO), termed as QASHO. The proposed QASHO has been scrutinized to enhance the exploitation ability, aiming to achieve global optimum, as QAO performs better within the local confinement region. Furthermore, the proposed approach shows improved strength in terms of escaping from the local minima trap, as in each iteration we discard some of the worst individuals by some suitable ones. To validate the proficiency of the proposed QASHO approach, 28 standard problems have been preferred in connection with IEEE-CEC-2017. The outcome observed from the suggested method has also equated with contemporary metaheuristic approaches. To prove the statistical significance, a nonparametric test has also been accomplished. Additionally as a real-life application, the suggested approach QASHO has utilized to train wavelet higher-order neural networks (HONN) by choosing datasets from the UCI store. The above correlations reveal that QASHO can deal with complex optimization tasks.

Keywords  Quadratic approximation (QA) · Spotted hyena optimizer (SHO) · Global optimization · Swarm-based algorithm · Metaheuristic approach · Wavelet neural network

1 Introduction

Optimization is considered as one of the fundamental operations in numerous applications, starting from engineering, business exercises, planning of vacations, and network routing to GPS navigation systems. The need for optimization is a demand of time as we have limited resources, and in return expect optimum profit, outcome, efficiency, and performance [1, 2]. Broadly all optimization algorithms are classified into 2 categories. The earlier one is termed gradient-based or trajectory-based or deterministic. Later is termed as gradient-free, population-based, or stochastic. A deterministic or trajectory-based method follows a single solution and continues with the same path by multiple interactions without involving any randomness. A popular example of such a method is hill climbing. The limitation associated with such kinds of optimization methods is the lack of attaining global optimum value within the stipulated period due to problem complexity [3]. On the other hand, randomness is involved with the stochastic or population-based approach. So multiple candidate solutions will follow various paths over iterations. Such methods fall into the category of contemporary nature-inspired metaheuristics [4–7].

The meaning of meta is upper level and the meaning of heuristic is to ascertain by experimental basis. So in simple terms metaheuristics means which performs superior as compared to simple heuristic [Glover 1986] which guides the search process to attain better outcomes without falling into the local optima trap. Nature has full of complications and simultaneously also a huge source of
stimulation to elucidate such by (the help of) observing the same, especially biological nature, for which new methods are emerging. Such methods are providing a smarter way for innovation and modern tools for analysis. In addition to this, these methods are simple, flexible, and efficient to deal with a vast range of applied complexities. Also, such methods are free from initial assumptions and suitable to handle nonlinear, multimodal, and complex problems. Some broad applications of nature-stimulated methods are in the fields of swarm intelligence [8], neural networks [9], fuzzy classifications [10], and evolutionary programming [11]. Still, the nature-inspired mechanism has to face a lot of challenges such as parameter setting, choosing, and controlling of the parameter. Such nature-inspired methods are normally centred on swarm intelligence, bio-inspired organisms, physics, and biochemical-based arrangements. Popular population-based algorithms which are evolved by observing the social behaviour of various creatures are PSO [12], GA [13], SSA [14], SHO [15], ACO [16], ABC [17], Big Bang Big-Crunch [18], and GWO [19]. According to no free lunch theorem stated by Wolpert and Macready expresses that there exists no particular approach which can be uniformly applied to all varieties of problems and returns optimum outcome successfully [20]. This is the cause for the evolvement of modern metaheuristic algorithms to resolve complicated real-world problems. At the same time, momentum has started for hybrid algorithms by researchers over years. The popularity of hybridization is due to its underlying capability, by combining the abilities of better/strong characteristics of two approaches. So that we can adopt the best.

The cause for the evolvement of novel hybrid algorithms is to intensify the performance in terms of better exploration and exploitation competency [21–23]. In this work, we propose an approach by hybridizing SHO and quadratic approximation operator (QAO) to improve the exploitability strength of the original SHO, which is termed as quadratic approximation spotted hyena optimization (QASHO). In the suggested hybrid approach, the QAO is castoff to alter a portion of search agents, while the rest of the search agents are updated by SHO customarily. Proposed QASHO and standard SHO are assessed in terms of performance over IEEE-CEC-2017 problems. Simultaneously the statistical significance has also been established by considering popular nonparametric measures such as Friedman and Holm’s assessment. Furthermore, as an application in real life, the suggested QASHO is utilized to train wavelet neural networks. In the past decade, modern metaheuristics approaches are widely applied over diversified domains to support complicated decision-making. Observing traditional human practices can lead to the development of new metaheuristics. Some of the popular applications of metaheuristic approaches are optimal routing in transportation, bioinformatics, and communication networks. Apart from these traditional applications, recently these approaches are utilized for various realistic applications such as portfolio optimization, index tracing, increased indexation, credit risk, stock pricing, fiscal assignment planning, attribute assortment, insolventcy, and fiscal hardship forecast. The field of structural design is one where metaheuristics have gained traction and pace. No known analytical strategies have yet been devised to solve this challenge due to the highly nonlinear design restrictions and onerous objective functions connected with them. Some of the most interesting applications are how a fisherman goes about maximizing his fishing results, scheduling of timetable for educational institutions, and assigning apprentices to industry engagements.

The remaining part is formulated as follows: The literature related to QAO and recent applications of SHO is presented in segment 2. Details about the SHO algorithm and QAO are precised in segment 3. Segment 4 represents the suggested approach QASHO. The ambient for experimentation and utilized standard functions set with required constraints as described in segment 5. Scrutiny of the reported results, comparison among recent methods, complexity, convergence analysis, and statistical significance of suggested approach is signified in segment 6. As a realistic application, the said approach is castoff as a classifier to train HONN (WNN) with the help of chosen benchmark datasets is signified in segment 7. The concluding observations are presented in segment 8.

2 Literature Review

Many hybrid algorithms are available in recent times due to their better adaptability for solving tasks. We have integrated the quadratic approximation operator (QAO) with SHO and presented some of its variants. In the year of 2008 Deep et al. hybridized QAO with GA. The 4 types of operators are hybridized and the same is compared with normal GA’s by considering a test set of 22 problems. As an outcome ranking is accomplished among all 4 types of hybrid GA’s and which GA (castoff with QAO) is best according to performance is determined [24]. Again in 2009 same author Deep et al. castoff QAO with PSO as a hybrid algorithm. The hybridization has performed over 2 popular alternates of PSO like PSO-W and PSO-C. Afterwards, the effectiveness of the proposed 2 approaches is compared with the existing two, over 15 standard problems and 3 applied problems. From the experimental observation, it is evident that the proposed two approaches are effective for achieving optimum for nonlinear complications [25]. Gupta et al. in the year 2017 castoff QAO with spider monkey optimization also, to enhance the local search capability, for solving unconstrained problems. Also, the improved method has been compared with existing SMAO in terms of various efficiency measures and used to solve LJ problems [26]. Chen et al. used QAO
with a teacher-learning-based optimization method to resolve dynamic optimization complications. Also, the usefulness of said method has been equated with the contemporary 11 approaches. The results revealed that by integrating QAO with TLBO, the exploration and exploitation competencies of the existing method amplified to a better extent [27]. In 2016, Turgeon et al. integrated QAO with the artificial cooperative search (ACS) method to constraint electrical load balancing problem including various generation modules. The study and results presented have shown that the suggested approach gives better efficiency compared to existing ACS [28]. Naidu et al. incorporated QAO with the invasive weed optimization (IWO) method, to investigate the competence of both methods. Observation for the results discloses that the suggested approach is superior to the existing IWO and population algorithm GA [29]. A large amount of SHO variations are available and applied across various disciplines of society. Though SHO is a recent algorithm, it has been widely used to resolve various engineering problems [30–36]. Rather than engineering problems, SHO is also castoff to resolve numerous mechanical and electrical glitches such as reduction of load for brake constituents belonging to automobiles, PID factor optimization for AVR systems, soil strength estimation, economic load dispatch problem, and improvement in software project quality also [37–41]. The same algorithm has also been used with neural networks, to solve feature selection and classification problems [42–46].

Recently metaheuristic algorithms are utilized across diversified domains for resolving real-time complications. The problem areas may also be scattered from the game-based optimization approach, management of game governing rules, granting of AI-based patent for the economic benefit of the nation, evaluating fuzzy reliability, and covid-19 data analysis over supervised algorithms [47–51]. Dhiman et al. suggested (STOA) a bio-inspired algorithm for tackling constrained engineering issues. The main source of inspiration for this approach is the natural movement and assault actions of the sea bird sooty tern. It is then used to solve six real-time problems to illustrate its applicability [52]. The same author suggested EPO approach as an evolutionary method. The emperor penguin’s huddling behaviour is the inspiration for this algorithm [53]. Another enhanced BEPO approach is suggested by Dhiman et al. to resolve feature selection tasks efficiently [54]. SOA by Dhiman et al., an evolutionary algorithm for solving computationally demanding tasks. The main source of inspiration for this method is a seagull’s natural movement and attacking activities. To emphasize exploration and exploitation in a particular search space, these behaviours are mathematically represented and implemented [55]. Dhiman et al. suggested RSO following the natural behaviour pattern of rats and ESA hybrid approach by integrating the SSA with EPO and utilized both approaches to resolve engineering complications [56, 57]. Kaur et al. and Dehghani et al. suggested two bio-inspired methods as TSA and SSA and utilized both the approaches to resolve constrained and unconstrained engineering complications [58, 59]. Due to the diverse application of metaheuristic methods, we have considered SHO a recently suggested to incorporate with QAO to intensify the performance of existing algorithms.

3 Basic Preliminaries

The successive segment describes the societal and haunting pattern of spotted hyenas along with the quadratic approximation operator QAO).

3.1 Outline of Spotted Hyena Optimizer

SH also termed as laughing hyena is considered the biggest among all hyena species. These hyenas are large carnivores at present innate to Africa, possibly originated in Asia, and spread across Europe till close to the Pleistocene. This species shows high societal attachment, complicated behaviour, and biggest clan size, competitive social pattern. The clan is headed by females, male hyenas are smaller than females and get importance in the clan. As they live largest in number in a clan, they are considered as the great prosperous species due to their adaptableness and cunning nature. The spotted hyena shows superior flexibility during a hunt and exploring pattern in comparison with the rest of the carnivores. The carnivore can hunt alone, with fewer members, or with large groups. They can chase their selected prey with the pack for a longer distance, maybe to some kilometres to death. The overall behaviour pattern of spotted hyenas is mapped mathematically to resolve numerous optimization complications [15]. In the SHO implementation, prey position is treated as instantaneous best for the sake of nearer to optimal as search region is unexplored beforehand. The remaining search agents alter their location after fixing the best search agent around the finest optimum solution. Numerical demonstration of behaviour is expressed by Eqs. (1) and (2).

\[
\overrightarrow{D_{th}} = \left| \overrightarrow{B_{cov}} \ast \overrightarrow{P_{v}} \ast I - \overrightarrow{P_{S}}(CUI) \right| 
\]

\[
\overrightarrow{P_{S}}(I + 1) = \overrightarrow{P_{v}}(I) - E_{cov} \ast \overrightarrow{D_{th}} 
\]

where $\overrightarrow{D_{th}}$ symbolizes the interspace between hyena and prey, $CUI$ symbolizes in progress repetition, $\overrightarrow{P_{v}}$ symbolizes prey vector position, $\overrightarrow{P_{S}}$ symbolizes hyena position vector, $B_{cov}$ and $E_{cov}$ symbolize coefficient vectors and represented
companions and grasps the superior magnitudes attained till the agents (hyenas) articulates an assembly of trustworthy (hyena) which in turn is treated as optimum. The rest of follows, prey position is recognized to a superlative agent vector considered randomly, within limit \([0, 1]\), and grs can be computed by Eq. (6).

bolizes a pool of candidate solutions. The optimum outcome PS \(\rightarrow\) the search region. symbolizes the spot of remaining agents surrounded within limit \([0, 1]\), and POrs symbolizes the optimum agent (hyena) and POrs \(\rightarrow\) the population. Two are chosen randomly and the third one is considered for searching in the global region. In the first step of SHO, we may achieve superior global optimum by exploiting prior exploitation ability. The QAO is considered a better tool for any specified task. Using QAO we can generate another new term, by considering 3 individuals. Among 3 individuals, the best individual is picked from the population as first, and the rest 2 are chosen randomly. By utilizing these 3 picked individuals QAO finds the point of optima by using the quadratic surface and approximates the tested functions [27]. Hence, the extreme optimum can be computed by the following equation. To generate a new individual by using QAO, let the 3 elements are

\[
\begin{align*}
Z_1 &= (Z_1^1, Z_1^2, Z_1^3, \ldots, Z_1^D) \\
Z_2 &= (Z_2^1, Z_2^2, Z_2^3, \ldots, Z_2^D) \\
Z_3 &= (Z_3^1, Z_3^2, Z_3^3, \ldots, Z_3^D)
\end{align*}
\]

Then it may be computed as follows:

\[
Z_{QA}^D = 0.5 \times \left[\left((Z_3^D - Z_2^D)^2 \times F(Z_1)\right) + \left((Z_1^D - Z_2^D)^2 \times F(Z_2)\right) + \left((Z_2^D - Z_1^D)^2 \times F(Z_3)\right)\right]
\]

where \(F_{sa}^\rightarrow\) symbolizes the optimum agent (hyena) and POrs \(\rightarrow\) the population. Two are chosen randomly and the third one is considered for searching in the local search region. \(A_{om}^\rightarrow\) outlines a pool of entire finest consequences.

\[
\begin{align*}
\overrightarrow{TP_{th}} &= \text{Calcgrs}\left(\overrightarrow{F_{sa}}, \overrightarrow{F_{sa+1}}, \overrightarrow{F_{sa+2}} + \cdots + \overrightarrow{F_{sa + RV}}\right)
\end{align*}
\]

where \(TP_{th}\) entitles population size of spotted hyenas, RV is a vector considered randomly, within limit \([0, 1]\), and grs symbolizes a pool of candidate solutions. The optimum outcome can be computed by Eq. (6).

\[
\overrightarrow{PS}(CUI + 1) = \frac{A_{om}^\rightarrow}{TP_{th}}
\]

where \(\overrightarrow{PS}(CUI + 1)\) illustrates the best significant outcomes which are conformed from formerly perceived value accrued from diverse repetition and remaining agents will alter their positions based on observing the best consequence attained.

where \(F(Z_1), F(Z_2), \) and \(F(Z_3)\) are objective function fitness outcomes at \(Z_1, Z_2, \) and \(Z_3\). After observing computed \(Z_{QA}\), the same is contrasted with the worst individuals \(Z\) worst from SHO population. If the observed \(Z_{QA}\) is superior to the worst individuals, \(Z\) worst, then replace the values of \(Z_{QA}\) with \(Z\) worst.

4 Proposed Quadratic Approximation-Based Spotted Hyena Optimization (QASHO)

This segment will introduce the proposed quadratic approximation-based spotted hyena optimization algorithm (QASHO). The method QASHO incorporates quadratic approximation techniques denoted in Eq. (7) on SHO, expecting intensification in performance issues like superior exploitation ability. The QAO is considered a better tool inside the local search region. So by integrating QAO with SHO, we may achieve superior global optimum by exploiting the parameters in the local region, while maintaining balance for searching in the global region. In the first step of hybridization, we have to pick 3 individuals from the hyena population. Two are chosen randomly and the third one is picked from the population, which is the best candidate. The 3 individuals are \(Z_1, Z_2, \) and \(Z_3\). The new candidate \(Z_{QA}\)
Table 1 Proposed quadratic approximation operator-based spotted hyena optimization algorithm (QASHO)

| Pseudocode of QASHO algorithm |
|-------------------------------|
| Assert whole parameters and declare $l_{r_1}, b_{cov}, E_{cov}$ and $TP_{sh}$ |
| Assess fitness of each agent |
| Synchronously assess by replacing the worst individuals with superior elements attained from $Z_{QA}$ model as shown in Eq. (7) |
| Assign $F_{sa} \leftarrow$ best search agent |
| Make a group of finest outcomes, symbolized as $A_{gm}$ |
| while (counter $<$ maximum number of reiterations) do |
| for each distinct search agent in the population do |
| Modify position of ongoing search agents within population by Eq. (6) |
| end for |
| Update elements: $l_{r_1}, b_{cov}, E_{cov}$ and $TP_{sh}$ |
| Check for crossing search region limits by any individual: if so adjust |
| Define fitness of revised search agents independently |
| Modify $F_{sa}$ value, if better outcome attained in contrast to former finest outcome |
| Amend cluster $A_{gm}$ conforming to $F_{sa}$ |
| counter $\leftarrow$ counter $+$ 1 |
| end while |
| return $F_{sa}$ |

will be generated by using Eq. 7. After generating the new term, it will be equated with the poorest candidates present in the populace. If the perceived value of $Z_{QA}$ is superior to the worst individual $Z_{worst}$, then replace the values of $Z_{QA}$ with $Z_{worst}$. In case of minimization problem, if perceived $Z_{QA}$ value is less than the highest values in population then alter the worst individuals. As QAO performs better as a local search agent, it utilizes the parameter available locally and exhibits enhanced exploitation capability, which in return attain a smoother balance between search regions to acquire global optimum. The pseudocode for the proposed QASHO method is represented in Table 1.

5 Benchmark Functions and Experiment Settings

The technique of replicating a scientific conclusion achieved by computational means by doing research that does not rely heavily on computing resources is the main concern behind experimental validation. We have considered the IEEE-CEC-2017 standard problem set for the assessment of the proposed QASHO along with the existing SHO. The set contains 28 well-proven constrained optimization tasks with problem dimension size 10 [60]. The reported results are considered over 500 executions for one run/trial. Twenty-five such trials are considered for the accuracy of outcomes. The search region is in accordance with IEEE-CEC-2017 and the specified range is as: $[-10, 10], [-20, 20], [-50, 50]$ and $[-100, 100]$. For all algorithms, the number of search agents considered is set at 20, because 20 is a rational number for solving optimization problems and obtaining a global optimum. The environment we have followed for implementation is MATLAB 2016a with processor i7-10th generation, speed 4.1 GHz, 8 GB RAM, 64-bit OS, and 1 TB HDD.

6 Analysis of Consequences

This subsection presents the efficiency of QASHO in contrast with SHO, verified in terms of 28 constraints standard functions with dimension 10 and belongs to IEEE-CEC-2017. Benchmark functions employing benchmark indices that only focus on the method’s capability to search within the solution region are a good way to evaluate the performance of evolutionary metaheuristic algorithms. As a result, function evaluation is the most efficient metric for metaheuristic algorithms. The acquired results from QASHO and SHO are reported in Table 2. Various efficacy metrics such as minimum, maximum, mean, median, and standard deviation values are attained by executing over CEC-2017. The CEC-2017 standard test functions comprise unimodal, multimodal, hybrid, and composite functions. The unimodal and multimodal functions are utilized to verify the exploration and exploitation ability. Algorithms that follows a stochastic approach will be well regarded if it can avoid confined optima from becoming stuck by increasing exploitative strength inside the search region, endorses its enhanced diversification capability, which can be best examined using
Table 2  Consequences found from numerous objective functions belonging to IEEE-CEC-2017 with problem magnitude 10 over SHO and proposed QASHO

| Benchmark Functions | Optimization technique | Mean   | Median   | Std Dev   | Lowest value | Highest value |
|---------------------|------------------------|--------|----------|-----------|--------------|---------------|
| 1                   | SHO                    | 7.74E + 03 | 7.66E + 03 | 460.6879  | 7.31E + 03   | 8.31E + 03   |
|                     | QASHO                  | 7.16E + 03 | 7.14E + 03 | 1030.146  | 6.09E + 03   | 8.22E + 03   |
| 2                   | SHO                    | 8.96E + 03 | 9.49E + 03 | 1044.813  | 7.76E + 03   | 9.64E + 03   |
|                     | QASHO                  | 5.85E + 03 | 5.08E + 03 | 2641.218  | 3.24E + 03   | 8.45E + 03   |
| 3                   | SHO                    | 1.13E + 04 | 1.12E + 04 | 636.3961  | 1.08E + 04   | 1.20E + 04   |
|                     | QASHO                  | 1.01E + 04 | 1.01E + 04 | 1610.807  | 1.01E + 04   | 1.02E + 04   |
| 4                   | SHO                    | 2.43E + 02 | 2.43E + 02 | 41.01219  | 2.14E + 02   | 2.72E + 02   |
|                     | QASHO                  | 2.05E + 02 | 2.06E + 02 | 41.01219  | 2.12E + 02   | 2.73E + 02   |
| 5                   | SHO                    | 7.25E + 04 | 7.23E + 04 | 2653.928  | 6.99E + 04   | 7.52E + 04   |
|                     | QASHO                  | 7.16E + 04 | 7.12E + 04 | 2722.132  | 6.91E + 04   | 7.45E + 04   |
| 6                   | SHO                    | 6.20E + 02 | 6.20E + 02 | 14.1414  | 6.09E + 02   | 6.30E + 02   |
|                     | QASHO                  | 6.10E + 02 | 6.11E + 02 | 12.72792  | 6.01E + 02   | 6.19E + 02   |
| 7                   | SHO                    | –1.84E + 02 | –1.84E + 02 | 57.98276  | –2.25E + 02  | –1.43E + 02  |
|                     | QASHO                  | –1.94E + 02 | –1.94E + 02 | 57.98276  | –2.35E + 02  | –1.53E + 02  |
| 8                   | SHO                    | 1.59E + 01 | 1.59E + 01 | 3.323402  | 1.35E + 01   | 1.82E + 01   |
|                     | QASHO                  | 1.53E + 01 | 1.53E + 01 | 3.50725  | 1.29E + 01   | 1.78E + 01   |
| 9                   | SHO                    | 5.11E + 00 | 5.11E + 00 | 0.494975  | 4.76E + 00   | 5.46E + 00   |
|                     | QASHO                  | 4.88E + 00 | 4.88E + 00 | 0.806102  | 4.31E + 00   | 5.45E + 00   |
| 10                  | SHO                    | 2.91E + 01 | 2.92E + 01 | 6.017479  | 2.48E + 01   | 3.35E + 01   |
|                     | QASHO                  | 2.63E + 01 | 2.62E + 01 | 4.928534  | 2.28E + 01   | 2.99E + 01   |
| 11                  | SHO                    | –2.21E + 02 | –2.11E + 02 | 42.88356  | –2.68E + 02  | –1.84E + 02  |
|                     | QASHO                  | –2.21E + 02 | –2.11E + 02 | 42.88356  | –2.68E + 02  | –1.84E + 02  |
| 12                  | SHO                    | 6.73E + 03 | 6.82E + 03 | 2099.563  | 4.63E + 03   | 8.64E + 03   |
|                     | QASHO                  | 6.15E + 03 | 6.19E + 03 | 2267.442  | 4.09E + 03   | 8.12E + 03   |
| 13                  | SHO                    | 3.22E + 09 | 4.51E + 09 | 2.35E + 09 | 4.99E + 08   | 4.64E + 09   |
|                     | QASHO                  | 2.70E + 08 | 2.59E + 08 | 68.624.583 | 2.07E + 08   | 3.43E + 08   |
| 14                  | SHO                    | 2.13E + 01 | 2.13E + 01 | 0.098489  | 2.12E + 01   | 2.14E + 01   |
|                     | QASHO                  | 2.08E + 01 | 2.08E + 01 | 0.079373  | 2.07E + 01   | 2.09E + 01   |
| 15                  | SHO                    | 4.39E + 01 | 4.30E + 01 | 11.2207  | 3.36E + 01   | 5.59E + 01   |
|                     | QASHO                  | 3.66E + 01 | 3.66E + 01 | 9.290579  | 2.74E + 01   | 4.58E + 01   |
| 16                  | SHO                    | 3.35E + 02 | 3.68E + 02 | 105.0444  | 2.17E + 02   | 4.19E + 02   |
|                     | QASHO                  | 3.04E + 02 | 3.22E + 02 | 98.78428  | 1.97E + 02   | 3.92E + 02   |
| 17                  | SHO                    | 3.13E + 00 | 3.41E + 00 | 0.601082  | 2.44E + 00   | 3.54E + 00   |
|                     | QASHO                  | 2.43E + 00 | 2.23E + 00 | 0.50659  | 2.06E + 00   | 3.01E + 00   |
| 18                  | SHO                    | 6.54E + 03 | 6.27E + 03 | 2242.326  | 4.39E + 03   | 9.24E + 03   |
|                     | QASHO                  | 4.88E + 03 | 4.75E + 03 | 993.0449  | 3.81E + 03   | 6.20E + 03   |
| 19                  | SHO                    | 3.66E + 01 | 3.88E + 01 | 4.042689  | 3.19E + 01   | 3.91E + 01   |
|                     | QASHO                  | 2.84E + 01 | 2.79E + 01 | 1.851945  | 2.69E + 01   | 3.05E + 01   |
| 20                  | SHO                    | 2.13E + 00 | 2.23E + 00 | 0.19975  | 1.90E + 00   | 2.26E + 00   |
|                     | QASHO                  | 1.85E + 00 | 1.99E + 00 | 0.28688  | 1.52E + 00   | 2.04E + 00   |
multimodal functions. The better acceptance of any metaheuristic algorithm is dependent upon its attaining global peak while preventing local minima trap, which is better judged by hybrid and composite functions. Analysing the results from Table 2 QASHO outperforms over SHO in all 4 categories of tested functions available for F1 to F28 in CEC-2017.

### 6.1 Consideration with Other Recent Metaheuristic Algorithms

Though swarm-based metaheuristic algorithms are different from each other due to their involved randomness and varying social behaviour patterns of various species in nature. So, to compare one swarm-based algorithm to another is not a concrete way to measure the difference. At the same time, if we are going to compare some contemporary approaches, then we can observe the behaviour of the different algorithms in terms of various constraints functions. From which we can conclude the exploration, exploitation, and convergence rate of any considered method. Hence, we have chosen the QASHO algorithm to compare with recent methods such as SCA [61], MFO [62], GWO [19], WOA [63], ALO [64], and SHO [15]. The compared consequence is presented in Table 3. By scrutinizing the outcome it is obvious that QASHO performs better as compared to others.

### 6.2 Analysis of Complexity and Concurrence rate

The introduction of the populace in SHO alongside QASHO takes time as $O(p \ast d)$. Here the variable $p$ and $d$ symbolizes the entire repetitions count required for generating a random population within the search region and magnitude of the problem, respectively. Computing fitness for individual search agents consumes time in its worst case is $O(\text{maxrepetn } \ast p \ast d)$, where maxrepetn symbolizes the overall count of repetitions to simulate the suggested approach. To make a group of optimal agents requires $O(\text{maxrepetn } \ast \text{cnt})$ time, where cnt symbolizes total hyena (agent) count. The entire process will repeat until it reaches the allowed number of reiterations to achieve desired consequence, which requires $O(m)$ time, where $m$ symbolizes the entire count of reiterations. So the proposed approach QASHO requires $O(\text{maxrepetn } \ast \text{cnt})$ time in its worst case. The convergence curve of the proposed QASHO approach and original SHO approach is shown in Fig. 1. The CEC-2017 benchmark functions with problem magnitude 10 have been considered for performance evaluation of QASHO and SHO. In the presented graph, the $X$-axis symbolizes the entire count of repetitions required for simulation and the $Y$-axis symbolizes standard functions value. By analysing the curve, it is obvious that the convergence curve of QASHO is superior as compared to SHO, as the curve approaches towards the minimum value. At the same time, it illustrates
enhanced ability towards accomplishing global peak by efficaciously avoiding local entrapment.

### 6.3 Statistical Procedures for Performance Assessment

The significance of statistical methodology is to confirm the supremacy of a projected method over state-of-the-art methods. The evolved algorithm must be statistically substantial concerning performance metrics. We have used Friedman’s test (Milton Friedman, 1937–1940) in this work to establish the variances among numerous considered classifiers [65, 66]. For each classifier observed values have been assigned with a definite rank. Here we have considered $\alpha = 0.05$ and $N$ is the count of datasets castoff.

Average rank $AV_{rk}$ can be appraised by Eq. (8) and presented in Table 4.

$$AV_{rk} = \frac{\text{Summation of ranks found from considered classifiers}}{\text{Datasets count}}$$

**Table 3** Assessment of outcomes obtained from proposed QASHO with state-of-the-art approaches appropriate to IEEE-CEC-2017 with problem magnitude 10

| Benchmark Functions | MFO     | GWO     | WOA     | SCA     | ALO     | SHO     | QASHO   |
|---------------------|---------|---------|---------|---------|---------|---------|---------|
| 1                   | 1.38E + 04 | 2.02E + 04 | 1.31E + 04 | 1.38E + 04 | 2.52E + 04 | 7.73E + 03 | 7.16E + 03 |
| 2                   | 1.73E + 04 | 2.52E + 04 | 1.12E + 04 | 1.07E + 04 | 1.65E + 04 | 8.96E + 03 | 5.58E + 03 |
| 3                   | 2.24E + 04 | 2.34E + 04 | 1.04E + 04 | 2.29E + 04 | 1.08E + 04 | 1.12E + 04 | 1.01E + 04 |
| 4                   | 2.12E + 02 | 2.66E + 02 | **1.73E + 02** | 2.87E + 02 | 2.16E + 02 | 2.43E + 02 | 2.05E + 02 |
| 5                   | 4.81E + 05 | 7.37E + 05 | **1.87E + 05** | 3.86E + 05 | 3.72E + 05 | 7.25E + 04 | 7.16E + 04 |
| 6                   | 1.21E + 03 | 6.82E + 02 | 8.62E + 02 | 9.62E + 02 | 8.15E + 02 | 6.20E + 02 | **6.10E + 02** |
| 7                   | 5.41E + 01 | −7.51E + 00 | −2.13E + 02 | −1.44E + 02 | −1.18E + 02 | −1.84E + 02 | −1.94E + 02 |
| 8                   | 2.30E + 01 | 5.52E + 01 | −2.05E + 01 | 5.30E + 01 | 2.19E + 01 | 1.59E + 01 | 1.53E + 01 |
| 9                   | 7.36E + 00 | 8.39E + 00 | −0.61E + 00 | 9.49E + 00 | 5.10E + 00 | 5.11E + 00 | 4.88E + 00 |
| 10                  | 4.80E + 01 | **2.02E + 01** | 4.02E + 01 | 4.05E + 01 | 4.19E + 01 | 2.91E + 01 | 2.63E + 01 |
| 11                  | −1.99E + 02 | −1.04E + 02 | −2.09E + 02 | −1.50E + 02 | −3.74E + 02 | −2.21E + 02 | −2.21E + 02 |
| 12                  | 1.25E + 04 | 1.98E + 04 | 1.12E + 04 | 1.62E + 04 | 2.18E + 04 | 6.73E + 03 | 6.15E + 03 |
| 13                  | 1.69E + 09 | 1.05E + 10 | 1.62E + 09 | 7.02E + 09 | 5.43E + 09 | 3.22E + 09 | 2.70E + 08 |
| 14                  | 2.12E + 01 | 2.12E + 01 | 2.14E + 01 | 2.13E + 01 | 2.09E + 01 | 2.13E + 01 | 2.08E + 01 |
| 15                  | 7.10E + 01 | 6.21E + 01 | 7.96E + 01 | 7.59E + 01 | 8.46E + 01 | 4.39E + 01 | 3.66E + 01 |
| 16                  | 3.26E + 02 | 3.22E + 02 | 3.25E + 02 | 3.56E + 02 | **2.76E + 02** | 3.35E + 02 | 3.04E + 02 |
| 17                  | 4.75E + 00 | 3.19E + 00 | 4.59E + 00 | 5.10E + 00 | 4.34E + 00 | 3.13E + 00 | **2.43E + 00** |
| 18                  | 1.46E + 04 | 9.56E + 03 | 1.06E + 04 | 1.16E + 04 | 2.25E + 04 | 6.54E + 03 | 4.88E + 03 |
| 19                  | 3.96E + 01 | 3.69E + 01 | 3.37E + 01 | 3.38E + 01 | 3.28E + 01 | 3.66E + 01 | 2.84E + 01 |
| 20                  | 2.51E + 00 | 3.35E + 00 | 3.02E + 00 | 3.10E + 00 | 2.98E + 00 | 2.13E + 00 | 1.85E + 00 |
| 21                  | 1.51E + 04 | 1.10E + 04 | 1.56E + 04 | 1.56E + 04 | 5.68E + 03 | 4.81E + 03 | 4.11E + 03 |
| 22                  | 4.46E + 09 | 2.09E + 09 | 2.76E + 09 | 2.40E + 09 | 3.89E + 09 | 5.16E + 08 | **5.05E + 08** |
| 23                  | 2.23E + 01 | 2.20E + 01 | 2.17E + 01 | 2.21E + 01 | 2.14E + 01 | 2.13E + 01 | **2.05E + 01** |
| 24                  | 5.84E + 01 | 5.86E + 01 | 6.91E + 01 | 6.66E + 01 | 7.47E + 01 | 4.08E + 01 | **3.35E + 01** |
| 25                  | 3.70E + 02 | **2.60E + 02** | 3.29E + 02 | 3.10E + 02 | 3.58E + 02 | 2.92E + 02 | 2.78E + 02 |
| 26                  | 4.95E + 00 | 4.85E + 00 | 4.26E + 00 | 4.10E + 00 | 5.98E + 00 | 2.92E + 00 | **2.20E + 00** |
| 27                  | 1.11E + 04 | 1.45E + 04 | 1.78E + 04 | 1.56E + 04 | 1.29E + 04 | 5.45E + 03 | 4.04E + 03 |
| 28                  | 3.61E + 01 | 3.87E + 01 | 3.46E + 01 | 3.65E + 01 | 3.15E + 01 | 3.24E + 01 | **3.05E + 01** |

Bold sign indicates the superior outcome.
Fig. 1 Graphical representation of concurrence curve accomplished from objective functions of IEEE-CEC-2017 over magnitude 10
Fig. 1 continued
Table 4 Calculation of average rank by utilizing state-of-the-art methods concentrated on attained from standard functions of IEEE-CEC-2017

| Employed Functions | MFO     | GWO     | WOA     | SCA     | ALO     | SHO     | QASHO   |
|--------------------|---------|---------|---------|---------|---------|---------|---------|
| 1                  | 1.38E+04 (4.5) | 2.02E+04 (6) | 1.31E+04 (3) | 1.38E+04 (4.5) | 2.52E+04 (7) | 7.73E+03 (2) | 7.16E+03 (1) |
| 2                  | 1.73E+04 (6)   | 2.52E+04 (7) | 1.12E+04 (4)   | 1.07E+04 (3)   | 1.65E+04 (5) | 8.96E+03 (2) | 5.58E+03 (1) |
| 3                  | 2.24E+04 (5)   | 2.34E+04 (7) | 1.94E+04 (6)   | 2.29E+04 (6)   | 1.09E+04 (2) | 1.13E+04 (3) | 1.01E+04 (1) |
| 4                  | 2.13E+02 (3)   | 2.67E+02 (6) | **1.73E+02 (2)** | 2.87E+02 (7) | 2.16E+02 (4) | 2.43E+02 (5) | 2.05E+02 (2) |
| 5                  | 4.81E+05 (6)   | 7.37E+05 (7) | 1.87E+05 (3)   | 3.86E+05 (5)   | 3.72E+05 (4) | 7.25E+04 (2) | 7.16E+04 (1) |
| 6                  | 1.21E+03 (7)   | 6.82E+02 (3) | 8.63E+02 (5)   | 9.63E+02 (6)   | 8.16E+02 (4) | 6.20E+02 (2) | **6.10E+02 (1)** |
| 7                  | 5.41E+01 (7)   | 7.51E+00 (6) | **2.13E+02 (1)** | 1.44E+02 (4) | 1.18E+02 (5) | 1.84E+02 (3) | 1.94E+02 (2) |
| 8                  | 2.30E+01 (5)   | 5.52E+01 (7) | **2.05E+01 (1)** | 5.30E+01 (6) | 2.20E+01 (4) | 1.60E+01 (3) | 1.54E+01 (2) |
| 9                  | 7.36E+00 (5)   | 8.39E+00 (6) | **0.61E+00 (1)** | 9.49E+00 (7) | 5.10E+00 (4) | 5.11E+00 (3) | 4.88E+00 (2) |
| 10                 | 4.80E+01 (7)   | **2.02E+01 (1)** | 4.02E+01 (4) | 4.05E+01 (5) | 4.19E+01 (6) | 2.91E+01 (3) | 2.63E+01 (2) |
| 11                 | **1.99E+02 (5)** | **1.05E+02 (7)** | **2.10E+02 (4)** | **1.51E+02 (6)** | **3.74E+02 (1)** | **2.21E+02 (2.5)** | **2.21E+02 (2.5)** |
| 12                 | 1.26E+04 (4)   | 1.99E+04 (6) | 1.13E+04 (3) | 1.63E+04 (5) | 2.18E+04 (7) | 6.73E+03 (2) | **6.15E+03 (1)** |
| 13                 | 1.69E+09 (3)   | 1.05E+10 (7) | 1.62E+09 (2) | 7.02E+09 (6) | 5.43E+09 (5) | 3.22E+09 (4) | 2.70E+08 (1) |
| 14                 | 2.12E+01 (4.5) | 2.12E+01 (4.5) | 2.14E+01 (7) | 2.13E+01 (6) | 2.09E+01 (3) | 2.13E+01 (2) | **2.08E+01 (1)** |
| 15                 | 7.10E+01 (4)   | 6.21E+01 (3) | 7.96E+01 (6) | 7.59E+01 (5) | 8.46E+01 (7) | 4.39E+01 (2) | **3.66E+01 (1)** |
| 16                 | 3.26E+02 (6)   | 3.22E+02 (4) | 3.25E+02 (5) | 3.56E+02 (7) | **2.76E+02 (1)** | 3.35E+02 (3) | 3.04E+02 (2) |
| 17                 | 4.75E+00 (6)   | 3.19E+00 (3) | 4.59E+00 (5) | 5.10E+00 (7) | 4.34E+00 (4) | 3.13E+00 (2) | **2.43E+00 (1)** |
| 18                 | 1.46E+04 (5)   | 9.56E+03 (7) | 1.07E+04 (3) | 1.17E+04 (4) | 2.25E+04 (6) | 6.55E+03 (2) | **4.88E+03 (1)** |
| 19                 | 3.96E+01 (7)   | 3.69E+01 (6) | 3.37E+01 (4) | 3.38E+01 (5) | 3.29E+01 (3) | 3.67E+01 (2) | **2.84E+01 (1)** |
| 20                 | 2.51E+00 (3)   | 3.35E+00 (7) | 3.02E+00 (5) | 3.10E+00 (6) | 2.98E+00 (4) | 2.13E+00 (2) | **1.85E+00 (1)** |
| 21                 | 1.51E+04 (4)   | 1.10E+04 (3) | 1.56E+04 (5.5) | 1.56E+04 (5.5) | 5.68E+03 (7) | 4.81E+03 (2) | **4.11E+03 (1)** |
| 22                 | 4.46E+09 (7)   | 2.09E+09 (3) | 2.76E+09 (5) | 2.40E+09 (4) | 3.89E+09 (6) | 5.16E+08 (2) | **5.05E+08 (1)** |
| 23                 | 2.24E+01 (7)   | 2.21E+01 (5) | 2.18E+01 (4) | 2.21E+01 (6) | 2.14E+01 (3) | 2.13E+01 (2) | **2.05E+01 (1)** |
| 24                 | 5.84E+01 (4)   | 5.86E+01 (3) | 6.91E+01 (6) | 6.66E+01 (5) | 7.47E+01 (7) | 4.08E+01 (2) | **3.55E+01 (1)** |
| 25                 | 3.70E+02 (7)   | **2.60E+02 (1)** | 3.29E+02 (5) | 3.10E+02 (4) | 3.58E+02 (6) | 2.92E+02 (3) | 2.78E+02 (2) |
| 26                 | 4.95E+00 (6)   | 4.85E+00 (5) | 4.26E+00 (4) | 4.10E+00 (3) | 5.98E+00 (7) | 2.92E+00 (2) | **2.20E+00 (1)** |
| 27                 | 1.12E+04 (3)   | 1.46E+04 (4) | 1.78E+04 (7) | 1.57E+04 (6) | 1.29E+04 (4) | 5.45E+03 (2) | **4.04E+03 (1)** |
| 28                 | 3.61E+01 (5)   | 3.88E+01 (7) | 3.47E+01 (4) | 3.65E+01 (6) | 3.15E+01 (3) | 3.24E+01 (2) | **3.05E+01 (1)** |
| Avg. Rank          | 5.21     | 5.05     | 3.96     | 5.35     | 4.6     | 2.44     | 1.3     |

Bold sign indicates the superior outcome
Friedman statistics can be figured by Eq. (9)

\[
FR_s = \frac{(N-1)Y_F^2}{N(NC-1) - Y_F^2}
\]  

(9)

where \( Y_F^2 = \frac{12N}{NC(NC+1)} \sum_i A\psi_{ik}^2 - \frac{NC(NC+1)^2}{4} \).

Here \( NC \) symbolizes castoff classifiers count. Friedman statistics \( FR_s \) is dispersed deliberating to \( F \)-distribution employing \( NC-1 \) and \( NC-1 \ast (N-1) \) degree of freedom. Based on 6 classifiers and 28 functions the degree of freedom is figured to be 6 to 162. Critical value for \( F \) (6162) taking \( \alpha \) as 0.05 is 3.6689 [67]. The anticipated null hypothesis is acknowledged if and only if the figured assessment of \( FR_s \) is lower than an experiential critical value, or else the hypothesis is excluded. Figured \( FR_s \) assessment is 26.01 which is considerably larger than experiential critical assessment. This proceeds towards the dismissal of the well-thought-out null hypothesis. By the denial of the null hypothesis, it endorses substantial variance among considered classifiers. After substantiating variance by Friedman’s approach next approach is carried out as Holms approach, to endorse the proposed approach accomplishes superior to other classifiers. Hence, the null hypothesis can be framed as, \( H_0 \), the classifiers equated in pair perform alike. To continue with Holm’s approach we necessitate \( Z \)-value which may be calculated as Eq. (10)

\[
Z = \frac{R_i - R_j}{SE}
\]  

(10)

where \( SE = \sqrt{\frac{NC(NC+1)}{6N}} \).

The likelihood \( p_r \) value can originate from the normal distribution table in connection with computed \( Z \)-value [68]. The hypothesis is acknowledged or excluded depending upon \( p_r \) and \( \frac{\alpha}{NC-1} \) value. Witnessing reported outcomes from Table 5, \( p_r \) values are smaller in comparison with \( \frac{\alpha}{NC-1} \) values proceed to denial of the hypothesis. Hence, denial of all considered classifiers with QASHO as a regulatory classifier authorizes its dominance in terms of performance among others (Fig. 2).

# 7 Wavelet Neural Network

In 1980, the concept of wavelet theory was first established by Grossman and Morlet [69]. Later the combination of wavelet theory with neural network leads to the evolvement of wavelet neural network (WNN). This is established in 1992 by Jhang [70]. Usually, WNN is a higher-order neural network that entails a feed-forward network having 3 layers as input, hidden, and output. The input layer is responsible for absorbing single or multiple input variables. The middle layer recognized as the hidden layer comprises neurons termed as wavelons similarly to sigmoid-based networks. The activation function for these wavelons is strained from a wavelet foundation [71]. A lot of wavelet functions are available in the literature without explicit manifestation. So 7 mother wavelet functions are considered with their typical manifestations [72]. The output layer approximates the target outcome by utilizing summers or linear combiners. Sigmoid is incorporated as an activation function in the final layer. Out of 7 functions we have considered Morlet wavelet functions which are depicted in Eq. 8. The cause behind choosing Morlet wavelet functions is its outstanding lower RMSE value as compared with Gaussian wavelet functions [73].

From an application point of view, we have castoff the proposed classifier QASHO to train wavelet higher-order neural network. To achieve the purpose we have considered 12 standard datasets from the UCI warehouse [74]. The specifics of the datasets and rearrangement of WNN concerning training and testing samples are presented in Table 6. The acquired outcomes from the suggested QASHO approach along with the contemporary metaheuristic approach including DE, GA, PSO, GWO, and SHO have been reported concerning numerous performance metrics in Tables 7 and 8. The metrics RMSE symbolized in Eq. (12) can be measured by refereeing the difference between anticipated consequence and assessed consequence perceived from WNN. Observing reduced RMSE, lesser standard deviation, and greater average values from Table 7 confirms better exploitable competence of suggested classifier QASHO. The suitability of a classifier for classification is also judged by sensitivity and specificity analysis. Here from the reported consequence suggested QASHO confirms its ability as a better classifier. Further higher accuracy reported in Table 8 signifies its
Table 5. Consequence reported from Holm’s Test

| Rank | Classifiers | z value | p value | ∞ \( p_{\text{Holm}} \) | Hypothesis |
|------|-------------|---------|---------|----------------|------------|
| 1    | SCA         | -7.01   | 0.00004 | 0.0083         | Excluded   |
| 2    | MFO         | -6.77   | 0.00004 | 0.01           | Excluded   |
| 3    | GWO         | -6.49   | 0.00004 | 0.0125         | Excluded   |
| 4    | ALO         | -5.71   | 0.00004 | 0.0166         | Excluded   |
| 5    | WOA         | -4.61   | 0.00004 | 0.025          | Excluded   |
| 6    | SHO         | -1.97   | 0.02559 | 0.05           | Excluded   |

Table 6. Structure of WNN for different datasets and considered samples for training and testing

| Datasets for classification | Attributes count | WNN structure | Training samples count | Testing samples count | Count of classes |
|----------------------------|-----------------|---------------|------------------------|-----------------------|------------------|
| Glass                      | 9               | 9–9–1         | 214                    | 42                    | 6                |
| Iris                       | 4               | 4–4–1         | 150                    | 30                    | 3                |
| Seed                       | 7               | 7–7–1         | 210                    | 42                    | 3                |
| Yeast                      | 8               | 8–8–1         | 1484                   | 185                   | 10               |
| Wine                       | 13              | 13–13–1       | 178                    | 51                    | 3                |
| Diabetes                   | 8               | 8–8–1         | 768                    | 150                   | 2                |
| Heart                      | 13              | 13–13–1       | 270                    | 60                    | 2                |
| Balloon                    | 4               | 4–4–1         | 20                     | 20                    | 2                |
| Ionosphere                 | 10              | 10–10–1       | 351                    | 70                    | 2                |
| Cancer                     | 9               | 9–9–1         | 683                    | 120                   | 2                |
| XOR                        | 3               | 3–3–1         | 8                      | 8                     | 2                |
| Liver                      | 6               | 6–6–1         | 345                    | 70                    | 2                |

improved diversification skill to achieve the finest optimal magnitude with successfully handling local trap problems. With all the substantial evidence the suggested approach QASHO has the ability for an appropriate classifier.

\[
\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{M} (T_i^y - Q_i^y)^2}{M}} \quad (12)
\]

8 Conclusion

In this study, we have proposed a hybrid algorithm as QASHO by hybridizing SHO with a quadratic approximation operator (QAO). Integration of SHO and QAO can attain the finest peak by preserving a smooth balance between exploration and exploitation. Because SHO might get stuck in local minima’s with slowed convergence and a smaller range of solutions, it sometimes fails to reach a global peak. The efficacy of the said method is validated by 28 standard problems from CEC-2017 considered for magnitude 10. By analysing the reported results from CEC-2017 standard functions as well as comparing the results from contemporary approaches, the proposed QASHO outperforms all, in several facets such as error assessment acknowledged from objective functions, other metrics like reduced mean square error, enhanced accuracy, lesser standard deviation, and higher average. The observed supremacy in reported metrics of suggested QASHO confirms its seamless balance between exploration and exploitation, which leads to attaining a global peak with a superior convergence rate. In the light of statistical significance, Friedman’s and Holm’s test endorses its uniqueness. From the application point of view as a better trainer algorithm, QASHO reveals a lower error value, and improved accuracy confirms its solid acceptance. In entirety, we may conclude that hybrid approaches are considered as a more pertinent technique to deal with complicated applied problems, as it requires a lesser number of assessments along with an increasing rate of convergence towards optimum. Many difficulties have arisen throughout our research and testing of hybrid metaheuristic techniques and their application to data classification. These issues raise the possibility of additional examination in this field of study such as how can algorithm-dependent parameters (controlling parameters) be fine-tuned to achieve maximum efficiency. From our observed experimentation we found that the suggested
| Dataset/Classifier | DE | GA | GWO | PSO | SHO | QASHO |
|-------------------|----|----|-----|-----|-----|-------|
| Balloon LOWEST RMSE | 0.2482 | 0.2617 | 0.0805 | 0.8937 | 0.1501 | 0.0777 |
| AVERAGE | 0.2608 | 0.2692 | 0.1581 | 1.0937 | 0.1751 | 0.1928 |
| STD | 0.0150 | 0.0269 | 0.0932 | 0.1825 | 0.0310 | 0.0961 |
| SPECIFICITY | 0.6 | 0.5 | 0.6 | 0.4 | 0.6 | 0.6 |
| SENSITIVITY | 0.8 | 0.8 | 0.8 | 0.6 | 0.8 | 0.8 |
| PREVALENCE | 40 | 40 | 40 | 30 | 40 | 40 |
| Cancer LOWEST RMSE | 0.0636 | 0.0474 | 0.0055 | 0.9874 | 0.0061 | 0.0052 |
| AVERAGE | 0.1212 | 0.0999 | 0.0784 | 1.2154 | 0.1754 | 0.1841 |
| STD | 0.1010 | 0.0974 | 0.1254 | 0.2378 | 0.0817 | 0.0729 |
| SPECIFICITY | 0.48 | 0.35 | 0.51 | 0.73 | 0.78 | 0.78 |
| SENSITIVITY | 0.76 | 0.66 | 0.6 | 1 | 0.96 | 1 |
| PREVALENCE | 38.33 | 33.33 | 30 | 50 | 48.33 | 50 |
| Diabetes LOWEST RMSE | 0.0496 | 0.0598 | 0.0349 | 0.9901 | 0.0071 | 0.0049 |
| AVERAGE | 0.1269 | 0.1351 | 0.1012 | 1.4554 | 0.0985 | 0.0835 |
| STD | 0.0764 | 0.0845 | 0.0787 | 0.4618 | 0.0561 | 0.0498 |
| SPECIFICITY | 0.65 | 0.54 | 0.25 | 0.45 | 0.4 | 0.37 |
| SENSITIVITY | 0.74 | 0.57 | 0.84 | 0.54 | 0.81 | 0.85 |
| PREVALENCE | 37.14 | 43.57 | 42.14 | 27.14 | 40.71 | 42.85 |
| Glass LOWEST RMSE | 0.0063 | 0.0122 | 0.0038 | 0.8163 | 0.0027 | 0.0017 |
| AVERAGE | 0.1654 | 0.0987 | 0.0635 | 0.9933 | 0.0979 | 0.1087 |
| STD | 0.0917 | 0.0621 | 0.0489 | 0.3291 | 0.0632 | 0.0758 |
| Iris LOWEST RMSE | 0.1422 | 0.2130 | 0.0501 | 0.7957 | 0.1187 | 0.0457 |
| AVERAGE | 0.3226 | 0.3823 | 0.190 | 1.6044 | 0.1475 | 0.1505 |
| STD | 0.1603 | 0.1189 | 0.1454 | 1.1202 | 0.0487 | 0.0620 |
| Liver LOWEST RMSE | 0.0814 | 0.1257 | 0.0906 | 0.9890 | 0.0168 | 0.0132 |
| AVERAGE | 0.2052 | 0.3126 | 0.2071 | 2.0098 | 0.1120 | 0.1405 |
| STD | 0.1378 | 0.1423 | 0.1358 | 0.9331 | 0.0711 | 0.0890 |
| SPECIFICITY | 0.67 | 0.47 | 0.29 | 0.5 | 0.55 | 0.44 |
| SENSITIVITY | 0.67 | 0.64 | 0.82 | 0.55 | 0.58 | 0.82 |
| PREVALENCE | 33.82 | 29.41 | 41.17 | 27.94 | 29.41 | 41.17 |
| Seed LOWEST RMSE | 0.0875 | 0.0770 | 0.0402 | 0.9535 | 0.0075 | 0.0095 |
| AVERAGE | 0.2341 | 0.2913 | 0.3200 | 1.4721 | 0.1978 | 0.1998 |
| STD | 0.1910 | 0.2101 | 0.3189 | 0.5312 | 0.1761 | 0.1432 |
| XOR LOWEST RMSE | 0.1905 | 0.1741 | 0.1519 | 0.3935 | 0.0846 | 0.0850 |
| AVERAGE | 0.3148 | 0.2967 | 0.2967 | 0.5286 | 0.2412 | 0.1957 |
| STD | 0.2143 | 0.1710 | 0.1701 | 0.2205 | 0.2145 | 0.1432 |
| SPECIFICITY | 0.75 | 0.75 | 0.5 | 0.25 | 0.5 | 37.5 |
| SENSITIVITY | 0.75 | 0.75 | 0.75 | 1 | 0.75 | 1 |
| PREVALENCE | 37.5 | 37.5 | 37.5 | 50 | 37.5 | 50 |
| Yeast LOWEST RMSE | 0.0330 | 0.0466 | 0.0260 | 0.9717 | 0.0764 | 0.0062 |
| AVERAGE | 0.2701 | 0.2957 | 0.2073 | 1.3874 | 0.1910 | 0.2014 |
| STD | 0.3110 | 0.3616 | 0.1870 | 0.4472 | 0.1412 | 0.1571 |
| Heart LOWEST RMSE | 0.0023 | 0.0019 | 0.5142 | 0.9930 | 0.0012 | 0.0011 |
| AVERAGE | 0.7893 | 0.9606 | 1.4652 | 2.0812 | 0.1047 | 0.1280 |
### Table 7 (continued)

| Dataset/Classifier | DE  | GA  | GWO | PSO | SHO | QASHO |
|--------------------|-----|-----|-----|-----|-----|-------|
| STD                | 0.8123 | 1.1240 | 1.1957 | 0.9637 | 0.1100 | 0.1498 |
| SPECIFICITY       | 0.51 | 0.51 | 0.55 | 0.37 | 0.22 | 0.40 |
| SENSITIVITY       | 0.96 | 0.88 | 0.70 | 0.77 | 0.85 | 0.92 |
| PREVALENCE        | 48.14 | 44.44 | 35.18 | 38.88 | 42.59 | 46.29 |
| Iono LOWEST RMSE  | 0.0049 | 0.0085 | 0.0099 | 0.9916 | 0.0069 | **0.0024** |
| AVERAGE           | 0.8231 | 0.0174 | 0.0145 | 1.7578 | 0.1428 | 0.1985 |
| STD               | 0.0976 | 0.0041 | 0.0046 | 0.5357 | 0.0758 | 0.0992 |
| SPECIFICITY       | 0.22 | 0.2 | 0.37 | 0.25 | 0.31 | 0.34 |
| SENSITIVITY       | 0.8 | 0.88 | 0.91 | 0.97 | 1 | 1 |
| PREVALENCE        | 40 | 44.28 | 45.71 | 48.57 | 50 | 50 |
| Wine LOWEST RMSE  | 0.0249 | 0.0012 | 0.0070 | 0.9982 | 0.0014 | **0.0012** |
| AVERAGE           | 0.0875 | 0.0519 | 0.0564 | 1.615 | 0.0798 | 0.0985 |
| STD               | 0.0589 | 0.056 | 0.1172 | 0.8009 | 0.0128 | 0.0119 |

Bold sign indicates the superior outcome

### Table 8 Error Rate and Accuracy conquered from numerous datasets

| Dataset/Classifier | DE  | GA  | GWO | PSO | SHO | QASHO |
|--------------------|-----|-----|-----|-----|-----|-------|
| Balloon Error Rate (%) | 30  | 35  | 30  | 50  | 30  | 30  |
| Accuracy (%)      | **70** | 65  | **70** | 50  | **70** | **70** |
| Cancer Error Rate (%) | 37.5 | 49.16 | 35  | 13.33 | 12.5 | 10.83 |
| Accuracy (%)      | 62.5 | 50.84 | 65  | 86.67 | 87.5 | **89.17** |
| Diabetes Error Rate (%) | 30  | 29.28 | 45  | 50  | 39.28 | 38.57 |
| Accuracy (%)      | 70  | **70.72** | 55  | 60.72 | 61.43 | 61.43 |
| Glass Error Rate (%) | 16.66 | 16.66 | 19.04 | 26.19 | 16.66 | 14.28 |
| Accuracy (%)      | 83.34 | 83.34 | 80.96 | 73.80 | 83.34 | **85.72** |
| Iris Error Rate (%) | 23.33 | 33.33 | 30  | 53.33 | 16.66 | 16.66 |
| Accuracy (%)      | 76.67 | 66.67 | 70  | 46.67 | 83.34 | **83.67** |
| Liver Error Rate (%) | 32.35 | 47.05 | 32.35 | 47.05 | 32.35 | 36.76 |
| Accuracy (%)      | **67.65** | 52.95 | **67.65** | 52.95 | **67.65** | 63.24 |
| Seed Error Rate (%) | 38.46 | 23.07 | 35.89 | 33.33 | 38.46 | 23.07 |
| Accuracy (%)      | 61.54 | 76.93 | 64.11 | 66.67 | 61.54 | **76.93** |
| XOR Error Rate (%) | 25  | 25  | 25  | 37.5 | 25  | 12.5 |
| Accuracy (%)      | 75  | 75  | 75  | 62.5 | 75  | **87.5** |
| Yeast Error Rate (%) | 9.18 | 8.1  | 3.78 | 11.35 | 8.64 | 1.62 |
| Accuracy (%)      | 90.82 | 91.9 | 96.22 | 88.65 | 91.36 | **98.38** |
| Heart Error Rate (%) | 25.92 | 29.62 | 33.33 | 42.59 | 33.33 | 33.33 |
| Accuracy (%)      | **74.08** | 70.38 | 66.67 | 57.41 | 66.67 | 66.67 |
| Wine Error Rate (%) | 54.90 | 35.29 | 37.25 | 54.29 | 27  | 41.17 |
| Accuracy (%)      | 45.10 | 64.71 | 62.5 | 45.10 | **73** | 58.83 |
| Iono Error Rate (%) | 48.57 | 45.71 | 35.71 | 38.57 | 34.28 | 34.28 |
| Accuracy (%)      | 51.43 | 54.29 | 64.29 | 61.43 | 65.72 | **65.72** |

Bold sign indicates the superior outcome
approach shows the deprived performance when the problem dimension is very high.

Furthermore, this study recommends the use of meta-heuristic algorithms to solve other complex restricted optimization issues that are important in practice. The enhanced SHO can be used in the future to solve real-world engineering design challenges such as truss bar design, aileron design, maritime paddle-wheel design, pressure liner design, and tension string design.

Acknowledgements This work is supported by the All India Council of Technical Education under RPS project No. 8-83/FDC/RPS (POLICY-1) 2019-20.

References
1. Dhiman, G.: SSC: A hybrid nature-inspired meta-heuristic optimization algorithm for engineering applications. Knowl.-Based Syst. 222, 106926 (2021)
2. Zheng, X.; Qin, A.K.; Gong, M.; Zhou, D.: Self-regulated evolutionary multitask optimization. IEEE Trans. Evol. Comput. 24(1), 16–28 (2019)
3. Quwaider, M.; Shatnawi, Y.: Neural network model as internet of things congestion control using PID controller and immune-hill-climbing algorithm. Simul Model Pract Theory 101, 102022 (2020)
4. Faramarzi, A.; Heidarinejad, M.; Mirjalili, S.; Gandomi, A.H.: Marinepredatorsalgorithm: a nature-inspired metaheuristic. Expert Syst Appl 152, 113377 (2020)
5. Fatollahi-Fard, A.M.; Hajighaee-Keshteli, M., Tahvakkol-Moghadam, R.: Reddeeralgorithm (RDA): a new nature-inspired meta-heuristic. Soft Comput. 1–29 (2020)
6. Abualigah, L.: Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. Neural Comput. Appl. 1–24 (2020)
7. Krishna, M.M., Panda, N. and Majhi, S.K.: Solving traveling salesman problem using hybridization of rider optimization and spotted hyena optimization algorithm. Expert Syst. Appl. 115353 (2021)
8. Slowik, A.; Kwasnicka, H.: Nature inspired methods and their industry applications—swarm intelligence algorithms. IEEE Trans. Industr. Inf. 14(3), 1004–1015 (2017)
9. Panda, N.; Majhi, S.K.: How effective is the salp swarm algorithm in data classification. In: Computational intelligence in pattern recognition, pp. 579–588. Springer, Singapore (2020)
10. Das, H., Naik, B., Behera, H.S., Jaiswal, S., Mahato, P. and Rout, M.: Biomedical data analysis using neuro-fuzzy model with post-feature reduction. J. King Saud Univ. Comput. Inf. Sci. (2020)
11. Lee, C.Y.; Yao, X.: Evolutionary programming using mutations based on the Lévy probability distribution. IEEE Trans. Evol. Comput. 8(1), 1–13 (2004)
12. Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings of ICNN’95-International Conference on Neural Networks, vol. 4, pp. 1942–1948. IEEE (1995)
13. Mirjalili, S.: Genetic algorithm. In: Evolutionary Algorithms and Neural Networks, pp. 43–55. Springer, Cham (2019)
14. Mirjalili, S.; Gandomi, A.H.; Mirjalili, S.Z.; Saremi, S.; Faris, H.: Mirjalili, S.M.: Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv. Eng. Softw. 114, 163–191 (2017)
15. Dhiman, G.; Kumar, V.: Spotted hyena optimizer: a novel bio-inspired based metaheuristic technique for engineering applications. Adv. Eng. Softw. 114, 48–70 (2017)
16. Dorigo, M.; Birattari, M.; Stützle, T.: Ant colony optimization. IEEE Comput. Intell. Mag. 1(4), 28–39 (2006)
17. Karaboga, D.; Akay, B.: A comparative study of artificial bee colony algorithm. Appl. Math. Comput. 214(1), 108–132 (2009)
18. Erol, O.K.; Eksin, I.: A new optimization method: big bang–big crunch. Adv. Eng. Softw. 37(2), 106–111 (2006)
19. Mirjalili, S.; Mirjalili, S.M.; Lewis, A.: Grey wolf optimizer. Adv. Eng. Softw. 69, 46–61 (2014)
20. Wolpert, D.H.; Macready, W.G.: No free lunch theorems for optimization. IEEE Trans. Evol. Comput. 1(1), 67–82 (1997)
21. Panda, N.; Majhi, S.K.: Improved salp swarm algorithm with space transformation search for training neural network. Arab. J. Sci. Eng. 45(4), 2743–2761 (2020)
22. Panda, N.; Majhi, S.K.: Improved spotted hyena optimizer with space transformational search for training pi-sigma higher order neural network. Comput. Intell. 36(1), 320–350 (2020)
23. Panda, N., Majhi, S.K.: Oppositional salp swarm algorithm with mutation operator for global optimization and application in training higher order neural networks. Multimed. Tools Appl. 1–25 (2021)
24. Deep, K.; Das, K.N.: Quadratic approximation based hybrid genetic algorithm for function optimization. Appl. Math. Comput. 203(1), 86–98 (2008)
25. Deep, K.; Bansal, J.C.: Hybridization of particle swarm optimization with quadratic approximation. Opsearch 46(1), 3–24 (2009)
26. Gupta, K.; Deep, K.; Bansal, J.C.: Improving the local search ability of spider monkey optimization algorithm using quadratic approximation for unconstrained optimization. Comput. Intell. 33(2), 210–240 (2017)
27. Chen, X.; Mei, C.; Xu, B.; Yu, K.; Huang, X.: Quadratic interpolation based teaching-learning-based optimization for chemical dynamic system optimization. Knowl.-Based Syst. 145, 250–263 (2018)
28. Turgut, M.S.; Demir, G.K.: Quadratic approximation-based hybrid Artificial Cooperative Search algorithm for economic emission load dispatch problems. Int. Trans. Electr. Energy Syst. 27(4), e2284 (2017)
29. Naidu, Y.R.; Ojha, A.K.: A hybrid version of invasive weed optimization with quadratic approximation. Soft. Comput. 19(12), 3581–3598 (2015)
30. Jia, H.; Li, J.; Song, W.; Peng, X.; Lang, C.; Li, Y.: Spotted hyena optimization algorithm with simulated annealing for feature selection. IEEE Access 7, 71943–71962 (2019)
31. Dhiman, G.; Kaur, A.: A hybrid algorithm based on particle swarm and spotted hyena optimizer for global optimization. In: Soft Computing for Problem Solving, pp. 599–615. Springer, Singapore (2019)
32. Kaur, S., Awasthi, L.K. and Sangal, A.L.: HMOSSSA: a hybrid meta-heuristic approach for solving constrained optimization problems. Eng. Comput. 1–37 (2020)
33. Dhiman, G.; Kumar, V.: Spotted hyena optimizer for solving complex and non-linear constrained engineering problems. In: Harmony Search and nature Inspired Optimization Algorithms, pp. 857–867. Springer, Singapore (2019)
34. Dhiman, G.; Kumar, V.: Multi-objective spotted hyena optimizer: a multi-objective optimization algorithm for engineering problems. Knowl.-Based Syst. 150, 175–197 (2018)
35. Dhiman, G.; Kaur, A.: Optimizing the design of airfoil and optical buffer problems using spotted hyena optimizer. Designs 2(3), 28 (2018)
36. Dhiman, G. and Kaur, A.: Spotted hyena optimizer for solving engineering design problems. In: 2017 International Conference on Machine Learning and Data Science (MLDS), pp. 114–119. IEEE (2017)
37. Yıldız, B.S.: The spotted hyena optimization algorithm for weight-reduction of automobile brake components. Mater. Test. 62(4), 383–388 (2020)
38. Zhou, G.; Li, J.; Tang, Z.; Luo, Q.; Zhou, Y.: An improved spotted hyena optimizer for PID parameters in an AVR system. Math. Biosci. Eng. 17(4), 3767 (2020)
39. Moayedi, H.; Tien Bui, D.; Anastasios, D.; Kalantar, B.: Spotted hyena optimizer and ant lion optimization in predicting the shear strength of soil. Appl. Sci. 9(22), 4738 (2019)
40. Dhiman, G.; Guo, S.; Kaur, S.: ED-SHO: A framework for solving nonlinear economic load power dispatch problem using spotted hyena optimizer. Mod. Phys. Lett. A 33(40), 1850239 (2018)
41. Elsabagh, M.A.; Farhan, M.S.; Gafar, M.G.: Cross-projects software defect prediction using spotted hyena optimizer algorithm. SN Appl. Sci. 2(4), 1–15 (2020)
42. Panda, N.; Majhi, S.K.: How effective is spotted hyena optimizer for training multilayer Perceptrons. Int. J. Rec. Technol. Eng. 8, 4915–4927 (2019)
43. Li, J.; Luo, Q.; Liao, L. and Zhou, Y.: Using spotted hyena optimizer for training feedforward neural networks. In: International Conference on Intelligent Computing, pp. 828–833. Springer, Cham
44. Kumar, V. and Kaur, A.: Binary spotted hyena optimizer and its application to feature selection. J. Ambient Intell. Hum. Comput. 1–21 (2019)
45. Luo, Q.; Li, J.; Zhou, Y.: Spotted hyena optimizer with lateral inhibition for image matching. Multimed. Tools Appl. 78(24), 34277–34296 (2019)
46. Divya, S.; Kiran, E.L.; Rao, M.S.; Venmulapati, P.: Prediction of gene selection features using improved multi-objective spotted hyena optimization algorithm. In: Data Communication and Networks, pp. 59–67. Springer, Singapore (2020)
47. Dehghani, M.; Montazeri, Z.; Givi, H.; Guerrero, J.M.; Dhiman, G.: Darts game optimizer: a new optimization technique based on darts game. Int. J. Intell. Eng. Syst 13(5), 286–294 (2020)
48. Dehghani, M.; Montazeri, Z.; Malik, O.P.; Dhiman, G.; Kumar, V.: BOSA: binary orientation search algorithm. Int. J. Innov. Technol. Explor. Eng. (IJITEE) 9, 5306–5310 (2019)
49. Chatterjee, I.: Artificial intelligence and patentability: review and discussions. Int. J. Mod. Res. I(1), 15–21 (2021)
50. Kumar, R.; Dhiman, G.: A comparative study of fuzzy optimization through fuzzy number. Int. J. Mod. Res. I(1), 1–14 (2021)
51. Vaishnav, P.K.; Sharma, S.; Sharma, P.: Analytical review analysis for screening COVID-19 disease. Int. J. Mod. Res. I(1), 22–29 (2021)
52. Dhiman, G.; Kaur, A.: STOA: a bio-inspired based optimization algorithm for industrial engineering problems. Eng. Appl. Artif. Intell. 82, 148–174 (2019)
53. Dhiman, G.; Kumar, V.: Emperor penguin optimizer: a bio-inspired algorithm for engineering problems. Knowl.-Based Syst. 159, 20–50 (2018)
54. Dhiman, G.; Oliva, D.; Kaur, A.; Singh, K.K.; Vimal, S.; Sharma, A.; Cengiz, K.: BEPO: a novel binary emperor penguin optimizer for automatic feature selection. Knowl.-Based Syst. 211, 106560 (2021)
55. Dhiman, G.; Kumar, V.: Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems. Knowl.-Based Syst. 165, 169–196 (2019)
56. Dhiman, G.; Garg, M.; Nagar, A.; Kumar, V.; Dehghani, M.: A novel algorithm for global optimization: rat swarm optimizer. J. Ambient Intell. Hum. Comput. 12(8), 8457–8482 (2021)
57. Dhiman, G.: ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems. Eng. Comput. 37(1), 323–353 (2021)
58. Dehghani, M.; Montazeri, Z.; Dhiman, G.; Malik, O.P.; Morales-Menendez, R.; Ramirez-Mendoza, R.A.; Dehghani, A.; Guerrero, J.M.; Parra-Arroyo, L.: A spring search algorithm applied to engineering optimization problems. Appl. Sci. 10(18), 6173 (2020)
59. Kaur, S.; Awasthi, L.K.; Sangal, A.L.; Dhiman, G.: Tunealt swarm algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng. Appl. Artif. Intell. 90, 103541 (2020)
60. Wu, G., Mallipeddi, R., Suganthan, P.N.: Problem definitions and evaluation criteria for the CEC 2017 competition on constrained real-parameter optimization. In: National University of Defense Technology, Changsha, Hunan, PR China and Kyungpook National University, Daegu, South Korea and Nanyang Technological University, Singapore, Technical Report (2017)
61. Mirjalili, S.: SCA: a sine cosine algorithm for solving optimization problems. Knowl.-Based Syst. 96, 120–133 (2016)
62. Mirjalili, S.: Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl.-Based Syst. 89, 228–249 (2015)
63. Mirjalili, S.; Lewis, A.: The whale optimization algorithm. Adv. Eng. Softw. 95, 51–67 (2016)
64. Mirjalili, S.: The ant lion optimizer. Adv. Eng. Softw. 83, 80–98 (2015)
65. Friedman, M.: The use of ranks to avoid the assumption of normality implicit in the analysis of variance. J. Am. Stat. Assoc. 32(200), 675–701 (1937)
66. Friedman, M.: A comparison of alternative tests of significance for the problem of m rankings. Ann. Math. Stat. 11(1), 86–92 (1940)
67. F Distribution Table, March 2018. Retrieved from http://www.socr.ucla.edu/applets.dir/f_table.html
68. Normal Distribution Table. Retrieved from http://math.arizona.edu/~rsims/ma464/standardnormaltable.pdf
69. Grossmann, A.; Morlet, J.: Decomposition of Hardy functions into square integrable wavelets of constant shape. SIAM J. Math. Anal. 15(4), 723–736 (1984)
70. Zhang, Q.; Benveniste, A.: Wavelet networks. IEEE Trans. Neural Netw. 3(6), 889–898 (1992)
71. Alexandridis, A.K.; Zapranis, A.D.: Wavelet neural networks: a novel algorithm for global optimization: rat swarm optimizer. J. Ambient Intell. Hum. Comput. 12(8), 8457–8482 (2021)
72. Wang, G., Guo, L. and Duan, H.: Wavelet neural network using multiple wavelet functions in target threat assessment. Sci. World J. (2013)
73. Panda, N., Majhi, S.K., Singh, S., Khanna, A.: Oppositional spotted hyena optimizer algorithm: a new bio-inspired based metaheuristic paradigm for global optimization. Eng. Appl. Artif. Intell. 90, 103541 (2020)
74. Bache, K. and Lichman, M.: UCI machine learning repository. http://archive.ics.uci.edu/ml. Irvine, CA: University of California, School of information and computer science 28