Received: January 2, 2026. Revised: February 8, 2026.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Hoatzin Optimization Algorithm (HOA): A Novel Nature-inspired Metaheuristic for Engineering Applications

Tareq Hamadneh[1] Omar Alsayyed[2] Mohanned Saadi Alsaudi[3] Irina Leonova[4] Aseel Smerat[5] Zeinab Montazeri[6] Mohammad Dehghani[6]* Om Parkash Malik[7] Kei Eguchi[8]

[1]*Department of Mathematics, Al Zaytoonah University of Jordan, Amman 11733, Jordan*
[2]*Department of Mathematics, Faculty of Science, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan*
[3]*School of Mathematical and Statistical Sciences, Southern Illinois University Carbondale, Carbondale, IL, USA*
[4]*Faculty of Educational Sciences and Humanities, International University of la Rioja (UNIR), Spain*
[5]*Hourani Center for Applied Scientific Research, Al-Ahliyya Amman University, Amman 19328, Jordan*
[6]*Department of Electrical and Electronics Engineering, Shiraz University of Technology, Shiraz 7155713876, Iran*
[7]*Department of Electrical and Software Engineering, University of Calgary, Calgary, AB T2N 1N4, Canada*
[8]*Department of Information Electronics, Fukuoka Institute of Technology, Japan*
* Corresponding author's Email: adanbax@gmail.com

Abstract: In this study, a novel swarm-based metaheuristic algorithm, termed the Hoatzin Optimization Algorithm (HOA), is proposed, which emulates the natural behaviours of the hoatzin bird in foraging and exploiting food resources. The core design of HOA is inspired by two key behaviours of this species: (i) sudden, extensive movements to identify new food habitats and (ii) slow, small-scale displacements near discovered resources to maximize exploitation. In the algorithm, the theoretical and abstract concepts of hoatzin behavior are first described and subsequently mathematically modeled in two phases—exploration and exploitation. The exploration phase simulates the bird's searching behavior for locating food sources, while the exploitation phase models the focused utilization of the discovered resources. The performance of HOA is evaluated on four constrained engineering design problems and compared against nine well-established metaheuristic algorithms. The results indicate that HOA achieves high-quality solutions by maintaining an optimal balance between exploration and exploitation, consistently ranking first across all problems and exhibiting performance that surpasses competing algorithms by 100%. These findings demonstrate that HOA, with its strong exploratory and exploitative capabilities and its ability to balance these processes, represents a powerful and effective metaheuristic algorithm with broad potential applications in both scientific and real-world engineering optimization problems.

Keywords: Hoatzin, Nature-inspired, Hoatzin optimization algorithm, Exploration, Exploitation, Optimization, Metaheuristic.

1. Introduction

Metaheuristic algorithms are among the most widely used and influential stochastic optimization approaches, playing a pivotal role in solving optimization problems across diverse domains of science, engineering, industry, and other real-world applications [1]. Unlike traditional and classical mathematical approaches, whose efficiency often depends on the availability of an explicit analytical model, continuity, and differentiability of the problem [2], metaheuristic algorithms, relying on directed random search mechanisms in the solution space, are able to extract competitive and high-quality solutions for complex, nonlinear, and multifaceted problems without the need for explicit knowledge of the mathematical structure of the problem [3]. The search process in metaheuristic algorithms is fundamentally governed by two core concepts: exploration and exploitation [4]. Exploration refers to the algorithm's ability to perform a broad and global search of the solution space to identify promising regions, whereas exploitation denotes its capability to conduct an intensive and accurate local search around the discovered high-quality solutions. The key to the success of metaheuristic algorithms lies in establishing a dynamic and effective balance between exploration and exploitation throughout successive iterations, a balance that directly influences convergence quality and the ability to avoid premature convergence to local optima [5].

Among the major advantages of metaheuristic algorithms are their conceptual simplicity, ease of implementation, problem independence, lack of requirement for derivative information, and high flexibility across a wide range of applications. These characteristics have led to the development of a large number of novel metaheuristic algorithms over recent decades and their extensive adoption in various scientific fields [6].

From the perspective of inspiration sources, metaheuristic algorithms can generally be classified into four main categories [7]:

Swarm-based algorithms, which are inspired by the collective behaviors of living organisms in nature, such as foraging, migration, social interaction, pursuit–evasion, and coordinated movement. Representative examples of this category include Particle Swarm Optimization (PSO) [8], Draco Lizard Optimizer (DLO) [9], Ant Colony Optimization (ACO), Butterfly Optimization Algorithm (BOA) [10], and Pelican Optimization Algorithm (POA) [11].

Evolutionary-based algorithms, which originate from biological and genetic principles and simulate processes such as reproduction, mutation, and natural selection. Well-known algorithms such as Genetic Algorithm (GA) [12] and Differential Evolution (DE) [13] belong to this family.

Physics-based algorithms, which are developed through the simulation of physical laws, forces, and phenomena. Algorithms such as Gravitational Search Algorithm (GSA) [14], Equilibrium Optimizer (EO) [15], Simulated Annealing (SA) [16], and Multi-Verse Optimizer (MVO) [17] are notable members of this group.

Human-based algorithms, which are designed by mimicking human decision-making processes, social interactions, learning mechanisms, and cognitive behaviors. Algorithms such as Mother Optimization Algorithm (MOA) [18], Hiking Optimization Algorithm (HOA) [19], and Teaching–Learning-Based Optimization (TLBO) [20] fall into this category.

In addition to the aforementioned approaches, numerous metaheuristic algorithms have been introduced in the literature in recent years. However, according to the No Free Lunch (NFL) theorem [21], it cannot be claimed that a single metaheuristic algorithm delivers superior performance across all optimization problems. This theorem states that due to the stochastic nature of search processes and the inherent uncertainty in reaching the global optimum, no algorithm can uniformly outperform others over all possible problems. This fundamental limitation has motivated researchers to continuously propose new algorithms and enhance search mechanisms to achieve improved performance for specific problems or particular classes of optimization tasks.

Based on a comprehensive review of the existing literature and to the best knowledge of the authors, no metaheuristic algorithm inspired by the natural behaviors and strategies of the hoatzin bird has been reported so far. This is despite the fact that hoatzin exhibits intelligent and adaptive behaviors in searching for and exploiting food resources, making it a highly promising and novel source of inspiration for the development of a new metaheuristic algorithm.

To locate food resources, the hoatzin performs exploratory movements with large displacement ranges, which, from an algorithmic perspective, correspond to the exploration phase and global search within the solution space. Conversely, once a food source has been identified, the bird remains in the same area and performs small, slow, and precise movements to maximize exploitation of the discovered resource. This behavior is algorithmically equivalent to the exploitation phase and local search in promising regions of the solution space.

To address this research gap, this study introduces a new metaheuristic algorithm, termed the Hoatzin Optimization Algorithm (HOA), inspired by the intelligent movement patterns of the hoatzin during food searching and exploitation processes. The main contributions of this paper can be summarized as follows:

- Proposing and developing a novel nature-inspired metaheuristic algorithm named the Hoatzin Optimization Algorithm (HOA).
- Deriving the core idea of HOA from the exploratory behavior of the hoatzin in searching for food resources and its exploitative behavior after food discovery.
- Providing a detailed explanation of the theory, concepts, and distinctive behaviors of

the hoatzin and establishing their correspondence with the algorithmic components of HOA.
- Formulating the mathematical model of HOA in two phases: exploration (based on simulating the large-scale movements of the hoatzin) and exploitation (based on simulating its small and precise movements).
- Evaluating the performance of HOA on four real-world engineering design optimization problems.
- Comparing the performance of HOA with nine well-established metaheuristic algorithms to assess the competitive capability of the proposed approach.

The remainder of this paper is organized as follows: Section 2 presents a comprehensive description and mathematical modeling of the proposed HOA. Section 3 is devoted to simulation studies and performance evaluation of HOA on real-world applications. Finally, Section 4 concludes the paper and outlines potential directions for future research.

2. Hoatzin optimization algorithm (HOA)

In this section, the source of inspiration, underlying theory, and abstract concepts employed in the design of the proposed metaheuristic algorithm, the Hoatzin Optimization Algorithm (HOA), are first elucidated. Subsequently, for the application of the algorithm to optimization problems, its mathematical modelling is presented across the initialization phase, the exploration phase (based on simulating the hoatzin's foraging strategy for locating food resources), and the exploitation phase (based on simulating the hoatzin's behaviour while utilizing discovered resources).

2.1 Behavioural analysis and key features of the hoatzin in HOA design

The hoatzin bird, due to its distinctive foraging and resource-utilization behaviors, serves as the primary inspiration for the design of the HOA metaheuristic. A detailed analysis of its behavior reveals that two principal strategies of the bird can be directly mapped onto the two core phases of the algorithm: exploration and exploitation.

The first behavior involves sudden and extensive movements that the hoatzin performs when leaving low-quality areas to search for new feeding habitats. These displacements are often non-directional and rely partly on the bird's prior experiences. The primary purpose of this behavior is to identify new food resources and increase the likelihood of encountering favorable environments. The bird's broad and sometimes irregular movements allow it to move away from constrained, low-benefit areas and discover potential opportunities across larger regions. In other words, this characteristic behavior effectively demonstrates the hoatzin's global search ability and its capacity to explore new environments. The second behavior consists of slow and limited movements around discovered food resources. Once the hoatzin accesses a tree or branch rich in food, instead of exploring new areas, it remains in place and performs small, deliberate movements to maximize utilization of the available resource. This behavior exemplifies local search and exploitation, enabling the bird to efficiently capitalize on surrounding opportunities without expending unnecessary energy and time exploring uncertain environments.

Analysis of these two behaviors indicates that the hoatzin naturally balances two critical strategies: extending the search domain to discover new resources and concentrating on optimal exploitation of existing resources. These strategies not only ensure the bird's survival but also provide a natural template for developing effective optimization algorithms.

Accordingly, the Hoatzin Optimization Algorithm (HOA) is designed by drawing inspiration from these distinctive behaviors. The bird's broad and scattered exploratory movements form the basis of the algorithm's exploration phase, while its slow, precise displacements around discovered resources inspire the exploitation phase, enabling focused local search and efficient utilization of existing solutions. This natural integration of the two key behaviors allows HOA to maintain an effective balance between global exploration and local exploitation, thereby delivering robust performance across diverse optimization problems.

2.2 Initialization process and population positioning in the solution space

The first step in the mathematical modeling of the Hoatzin Optimization Algorithm (HOA) is the initialization of the population and the assignment of member positions within the solution space. From an algorithmic perspective, each hoatzin in nature corresponds to a population member in the solution space, and its position represents a candidate solution to the given optimization problem.

In HOA, the position of each population member is represented by a vector of decision variables. The entire population of hoatzins, representing the algorithmic population in HOA, is encoded using a population matrix X:

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,d} & \cdots & x_{1,m} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ x_{i,1} & \cdots & x_{i,d} & \cdots & x_{i,m} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ x_{N,1} & \cdots & x_{N,d} & \cdots & x_{N,m} \end{bmatrix}_{N\times m} \quad (1)$$

The position of each population member is initialized randomly within the predefined bounds for each problem variable:

$$x_{i,d} = lb_d + r \cdot (ub_d - lb_d) \quad (2)$$

where:

- X denotes the population matrix of hoatzins.
- X_i represents the position of the i-th hoatzin.
- $x_{i,d}$ is the position of the i-th member in the d-th dimension of the solution space.
- N is the population size, and m is the number of decision variables.
- r is a uniformly distributed random number in the interval $[0, 1]$.
- ub_d and lb_d denote the upper and lower bounds of the d-th dimension, respectively.

Once all members' positions are determined, each is evaluated as a candidate solution using the objective function of the problem. The corresponding objective function values for the entire population are represented by a vector:

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (3)$$

where $F_i = F(X_i)$ denotes the evaluated objective function value of the i-th member.

Following the evaluation, the best member of the population is identified based on the objective values and stored. This member serves as a reference point for updating the positions of other members in subsequent iterations.

Upon completion of the initialization process, the algorithm proceeds into the iterative update cycle. In HOA, population member positions are updated according to the two key phases: exploration and exploitation. Each of these phases is subsequently described and mathematically modeled in detail, ensuring a direct mapping to the distinctive behaviors of the hoatzin in nature and maintaining a balance between global search and local exploitation.

2.3 Exploration phase: Simulating the hoatzin's foraging behaviour in locating food resources

The exploration phase of the Hoatzin Optimization Algorithm (HOA) is designed based on the sudden and relatively long-range movements of the hoatzin, a behavior exhibited by the bird when leaving low-quality habitats and searching for new feeding grounds. In the wild, the hoatzin performs extensive, directionally unstructured movements to access fresh food sources, while partially relying on prior experiences to instinctively identify promising locations.

In HOA, this natural behavior is modeled as the broad movement of the hoatzin for resource search within the solution space. Accordingly, the new position of each population member is calculated as follows:

$$X_i^{P1} = X_i + \delta_t \cdot r \cdot (X_{best} - I \cdot X_i) + \Delta X_i^{\varepsilon} \quad (4)$$

where:

- X_i^{P1} denotes the new position of the i-th member after the exploration phase.
- δ_t represents the displacement intensity, which gradually decreases as iterations progress; initial movements are rapid and wide-ranging, whereas later ones become more cautious and localized.
- r is a uniformly distributed random number in $[0, 1]$.
- X_{best} represents the position of the best member in the population.
- $I \in \{1,2\}$ is a selection coefficient that determines the movement direction toward potential food sources.
- ΔX_i^{ε} is a limited random perturbation in the hoatzin's movement, controlled by $P_\varepsilon \in \{0,1\}$:

$$\Delta X_i^{\varepsilon} = P_\varepsilon \cdot (1 - 2 \cdot r) \cdot \frac{|UB - LB|}{t + 1} \quad (5)$$

In these expressions, UB and LB denote the upper and lower bounds of each decision variable, and t is the current iteration index. The stochastic perturbation ΔX_i^{ε} adds flexibility to the population's movements and helps prevent premature convergence to local minima.

After computing the new positions, the objective function is evaluated at these positions. A new position is adopted only if it improves the objective function value:

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} < F_i \\ X_i, & else \end{cases} \quad (6)$$

where $F_i^{P1} = F(X_i^{P1})$ represents the objective function evaluated at the new position. This mechanism ensures that each exploratory move is accepted only if it enhances solution quality, while the hoatzin's extensive and stochastic movements enable the algorithm to discover new regions within the solution space.

Thus, the HOA exploration phase represents a carefully balanced combination of broad and directed search, accurately reflecting the natural behavior of the hoatzin in locating new food resources and providing a robust foundation for the optimization process.

2.4 Exploitation phase: Simulating the hoatzin's local search behaviour on food sources

The exploitation phase of the Hoatzin Optimization Algorithm (HOA) is inspired by the hoatzin's local search behavior when feeding on a rich tree or branch. In nature, once the hoatzin reaches a suitable food source, it remains on the same branch and restricts its movements to short, slow, and precise adjustments around that location. The primary goal of this behavior is to maximize the utilization of the discovered resource rather than exploring new areas. This behavior represents the local exploitation strategy of the hoatzin and forms the algorithmic basis for focused local search and convergence toward improved solutions.

In HOA, this natural behavior is modeled as the precise and small-scale movements of the hoatzin near the discovered resource. In this model, the new position of each population member is calculated as follows:

$$X_i^{P2} = X_i + r \cdot (X_{best} - I \cdot X_i) \cdot e^{-\frac{t}{T}} + \Delta X_i^{\varepsilon} \quad (7)$$

where:

- X_i^{P2} denotes the new position of the i-th member after the exploitation phase.
- X_{best} represents the best position in the population.
- $I \in \{1,2\}$ is a coefficient determining the movement direction toward the resource.
- r is a uniformly distributed random number in $[0,1]$.
- $e^{-\frac{t}{T}}$ represents the gradual reduction in displacement magnitude as iterations progress; thus, initial movements are larger and more dynamic, while later movements become finer and more precise.
- ΔX_i^{ε} is a limited random perturbation applied to the displacement, with occurrence probability controlled by (P_ε).
- T is the maximum number of iterations.

Once the new positions are computed, the objective function is evaluated at these positions. As in the exploration phase, a new position is accepted only if it improves the objective function value:

$$X_i = \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i, & else \end{cases} \quad (8)$$

where $F_i^{P2} = F(X_i^{P2})$ represents the objective function evaluated at the new position in the exploitation phase.

In this way, the HOA exploitation phase accurately reflects the hoatzin's precise and targeted local search aimed at maximizing the utilization of discovered resources. This phase complements the exploration phase and establishes a balance between global search and local exploitation, enabling the population to effectively progress toward improved solutions while fully leveraging the valuable resources already identified.

2.5 Implementation procedure and operation of the hoatzin optimization algorithm (HOA)

The proposed Hoatzin Optimization Algorithm (HOA) is a population-based metaheuristic capable of efficiently exploring the solution space through stochastic search to identify high-quality solutions for optimization problems.

The optimization process in HOA begins with population initialization (Eqs. (1)–(3)). At this stage, each member of the population is assigned a position within the solution space, and the corresponding objective function values are computed and evaluated. The best member of the population, representing the optimal objective value identified at this stage, is then determined and stored.

Once initialization is complete, the algorithm enters an iterative cycle aimed at progressively improving the population's positions. In HOA, the position of each population member is updated first through the exploration phase (Eqs. (4)–(6)) and subsequently through the exploitation phase (Eqs. (7)–(8)). After all members' positions are updated, the best member of the population is re-identified and stored as the current best solution for that iteration.

At the end of each iteration, the algorithm proceeds to the next cycle, continuing the update process based on the exploration and exploitation phases until the final iteration is completed. Upon completion of the entire optimization procedure, the best position within the population is returned as the optimal solution obtained by HOA.

To illustrate the algorithm's operational flow and stepwise procedure, a flowchart of HOA is presented in Fig. 1. This flowchart visually demonstrates how the algorithm progresses from population initialization, through the exploration and exploitation phases, toward identifying the best solution, providing a clear depiction of the overall optimization process.

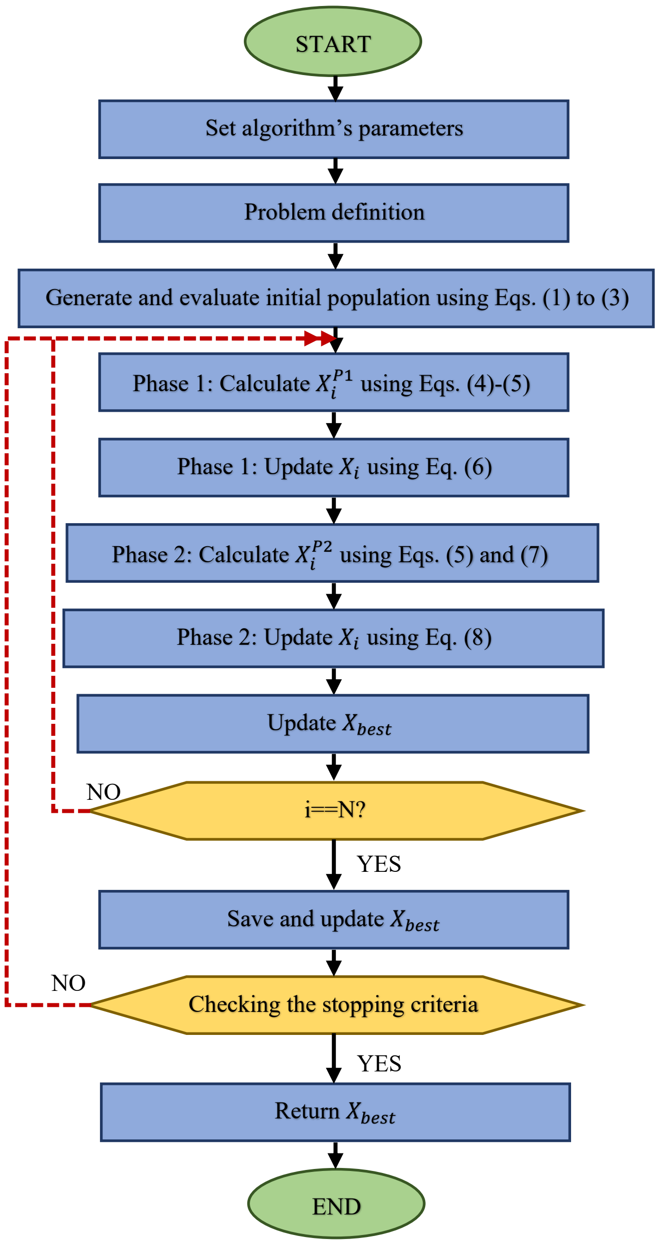

Figure. 1 Flowchart of HOA

2.6 Computational complexity assessment of the hoatzin optimization algorithm (HOA)

Assessing the computational complexity provides a precise perspective on the performance of the Hoatzin Optimization Algorithm (HOA) in solving optimization problems, and assists algorithm designers in evaluating its computational requirements and scalability.

In general, the structure of HOA comprises two main components:

1. Population initialization at the beginning of the algorithm execution.
2. Iterative procedures for updating the positions of population members.

During the initialization phase, the position of each hoatzin is randomly assigned within the solution space, and the corresponding objective function values for all population members are computed. This process has a computational complexity of ($O(N \cdot m)$), where (N) represents the number of hoatzins (population size) and (m) denotes the number of decision variables in the problem.

Following initialization, the algorithm enters an iterative cycle that consists of two independent phases: exploration and exploitation. In each phase, the new position of each population member is calculated, the corresponding objective function is evaluated, and the member's position is updated. This procedure is executed for all (N) population members over (T) iterations of the algorithm. Consequently, the computational complexity of each of the exploration and exploitation phases is ($O(N \cdot m \cdot T)$).

Based on the above, the overall computational complexity of HOA can be expressed as:

$$O_{total} = O(N \cdot m) + O(N \cdot m \cdot T) + O(N \cdot m \cdot T)$$
$$O_{total} = O\big((N \cdot m) \cdot (1 + 2 \cdot T)\big)$$

Therefore, the total complexity of HOA is linear with respect to the number of decision variables and population size, and also linear with respect to the number of iterations per phase. This analysis demonstrates that HOA, despite incorporating two distinct update phases, is computationally efficient and scalable, making it suitable for medium- to large-scale optimization problems, provided that the population size and number of iterations are appropriately selected according to available computational resources.

Furthermore, this complexity assessment provides a clear perspective on computational resource allocation and execution time for optimization problems of various dimensions,

Received: January 2, 2026. Revised: February 8, 2026.

enabling designers to achieve a balanced trade-off between solution accuracy and computational cost.

3. Simulation studies and performance analysis of the HOA algorithm on real-world engineering design problems

A realistic assessment of the effectiveness of metaheuristic algorithms necessitates examining their performance on real-world engineering optimization problems—problems that are typically characterized by nonlinear objective functions, complex constraints, continuous search spaces, and strong dependencies among design variables. These characteristics render the search process inherently challenging, often leading to issues such as convergence instability and high sensitivity to algorithmic settings. As a result, such problems provide a rigorous testbed in which the distinction between efficient and inefficient optimization algorithms can be clearly and reliably identified.

Accordingly, this section investigates the performance of the proposed Hoatzin Optimization Algorithm (HOA) in solving four well-established engineering design problems, namely the tension/compression spring design, welded beam design, speed reducer design, and pressure vessel design problems. These problems are widely regarded as benchmark cases in the engineering optimization literature and, owing to their stringent constraints and nonconvex structures, offer a suitable and demanding framework for evaluating the practical capabilities of metaheuristic algorithms.

For a thorough and well-documented assessment, the results obtained by the HOA algorithm are compared with those of several prominent metaheuristic algorithms and analyzed using standard statistical performance indicators along with appropriate visual representations. These analyses facilitate an in-depth examination of solution quality, performance robustness, and convergence behavior of the proposed algorithm, thereby providing a solid basis for discussing its advantages as well as its potential limitations in real-world engineering applications.

3.1 Studied problems, competing algorithms, and experimental settings

In this study, four classical engineering design problems are selected, which have been extensively employed in previous research to evaluate the performance of optimization algorithms. These problems involve nonlinear constraints, continuous search spaces, and high sensitivity to variations in design variables, and thus effectively reflect an algorithm’s capability to achieve a proper balance between exploration and exploitation.

To ensure a fair and meaningful comparison, the performance of the HOA algorithm is evaluated against nine well-known metaheuristic algorithms, including the Genetic Algorithm (GA) [22], Particle Swarm Optimization (PSO) [8], Gravitational Search Algorithm (GSA) [14], Whale Optimization Algorithm (WOA) [23], Teaching–Learning-Based Optimization (TLBO) [20], Multi-Verse Optimizer (MVO) [17], Tunicate Swarm Algorithm (TSA) [24], Reptile Search Algorithm (RSA) [25], and African Vultures Optimization Algorithm (AVOA) [26]. This set of algorithms represents a broad spectrum of evolutionary, population-based, and nature-inspired optimization paradigms.

All algorithms are implemented under identical experimental conditions, with 20 independent runs and 1000 iterations for each problem. To enable a comprehensive evaluation and mitigate the effects of stochastic behavior, the results are reported using six statistical indicators, namely the mean, best value, worst value, standard deviation, median, and final rank. These metrics collectively allow for a simultaneous comparison of solution quality, performance stability, and the relative superiority of the algorithms.

3.2 Tension/compression spring design

The tension/compression spring design problem is a fundamental and widely studied issue in mechanical component design, playing a critical role in the performance, safety, and efficiency of engineering systems. Such springs are extensively utilized across automotive, aerospace, industrial equipment, suspension systems, and energy absorption mechanisms.

The primary objective of this problem is to minimize the spring’s weight or effective volume while fully satisfying all mechanical and geometric constraints, including shear stress limits, allowable deflection, structural stability, and dimensional requirements. The strong nonlinearity of the constraints and the interdependence among design variables create a complex, constrained, and highly irregular search space. Consequently, this problem serves as a reliable benchmark for assessing an algorithm’s ability to handle constraints effectively and achieve stable, high-quality solutions. The mathematical model of this problem is presented as follows [23]:

$$Consider\ X = [x_1, x_2, x_3\ ] = [d, D, P].$$

Table 1. Statistical results of the optimization algorithms for the tension/compression spring design

Algorithm	mean	best	worst	std	median	Rank
HOA	0.012858	0.012665	0.01365	0.000727	0.012803	1
AVOA	0.012968	0.012669	0.013723	0.001068	0.012751	3
RSA	2.06E+12	0.013196	2.7E+13	1.94E+13	0.013307	10
TSA	0.012893	0.012707	0.013265	0.00049	0.01282	2
MVO	0.016814	0.012778	0.018125	0.005913	0.017891	6
WOA	0.013334	0.012665	0.015041	0.002211	0.01307	5
TLBO	0.018523	0.017856	0.019745	0.001444	0.018441	8
GSA	0.01712	0.013875	0.022308	0.00614	0.016979	7
PSO	0.013255	0.012685	0.015372	0.001937	0.013083	4
GA	0.024614	0.014409	0.033446	0.017917	0.024628	9

$$Minimize\ f(x) = (x_3 + 2)x_2x_1^2$$

subject to:

$$g_1(x) = 1 - \frac{x_2^3 x_3}{71785x_1^4} \leq 0,$$

$$g_2(x) = \frac{4x_2^2 - x_1x_2}{12566(x_2x_1^3)} + \frac{1}{5108x_1^2} - 1 \leq 0,$$

$$g_3(x) = 1 - \frac{140.45x_1}{x_2^2x_3} \leq 0,$$

$$g_4(x) = \frac{x_1 + x_2}{1.5} - 1 \leq 0$$

with

$$0.05 \leq x_1 \leq 2, 0.25 \leq x_2 \leq 1.3\ and\ 2 \leq x_3 \leq 15.$$

Statistical results obtained from 20 independent runs of various algorithms on this problem are reported in Table 1. Analysis of the mean indicator reveals that the proposed HOA algorithm achieved the best overall performance with a mean value of 0.012858, securing the top rank among all compared algorithms. This superiority is further underscored by a standard deviation of 0.000727, indicating high stability and very low sensitivity to initial conditions and independent runs. Moreover, the closeness of the median (0.012803) to the mean reflects a uniform distribution of results and the absence of significant fluctuations during the HOA convergence process.

Among the competing algorithms, TSA and AVOA demonstrated relatively competitive performance. TSA achieved a mean value of 0.012893 with relatively low dispersion, earning second place; however, in terms of solution quality and performance stability, it remains inferior to HOA. AVOA, despite attaining best values close to HOA, exhibited higher standard deviation, reflecting greater variability across independent runs and, consequently, lower reliability compared to HOA.

Other algorithms such as PSO and WOA occasionally produced reasonably good solutions, but their higher mean values and wider result dispersion indicate challenges in effectively balancing exploration and exploitation within the constrained search space of this problem. Algorithms like RSA and TLBO performed noticeably worse; RSA, in particular, reported very large worst-case values and high standard deviation, demonstrating severe instability and inability to handle the problem's nonlinear constraints.

Overall, the results presented in Table 1 unequivocally show that HOA outperforms all competing algorithms not only in terms of solution quality but also regarding stability, uniformity, and reliability. This superior performance highlights the strong capability of HOA to navigate complex search spaces and effectively manage the stringent constraints of the tension/compression spring design problem.

3.3 Welded beam design

The welded beam design problem is a classic and widely used benchmark in mechanical structural design and manufacturing engineering, frequently employed to evaluate the performance of optimization algorithms. The objective in this problem is to minimize the total fabrication cost of the welded beam while simultaneously satisfying multiple constraints, including shear stress, bending stress, buckling, end displacement, and geometric requirements. The problem's complexity arises from the nonlinear nature of the constraints, the strong interactions among design variables, and the sensitivity of the structural response to small parameter variations. Consequently, optimal welded beam design serves as a challenging benchmark for testing a metaheuristic algorithm's capability to manage stringent constraints and achieve stable and cost-effective solutions [23]:

$$Consider\ X = [x_1, x_2, x_3, x_4] = [h, l, t, b].$$

$$Minimize\ f(x) = 1.10471x_1^2x_2 + 0.04811x_3x_4(14.0 + x_2)$$

subject to:

Table 2. Statistical results of the optimization algorithms for the welded beam design problem

Algorithm	mean	best	worst	std	median	Rank
HOA	1.724881	1.724852	1.72524	0.000251	1.724853	1
AVOA	1.767854	1.72557	1.862205	0.113529	1.753791	4
RSA	2.283173	1.865514	3.19361	0.959837	2.174484	6
TSA	1.745918	1.735299	1.758199	0.015527	1.746218	3
MVO	1.740515	1.7268	1.800293	0.045362	1.737614	2
WOA	2.377497	1.800833	3.566042	1.477028	2.204143	7
TLBO	3.58E+12	3.095729	5.18E+13	3.54E+13	5.153188	10
GSA	2.50943	1.888788	3.04098	0.67517	2.482199	8
PSO	1.938663	1.724852	2.772051	0.866187	1.819975	5
GA	2.660471	2.006561	5.248906	2.173192	2.405594	9

$$g_1(x) = \tau(x) - 13600 \leq 0,$$
$$g_2(x) = \sigma(x) - 30000 \leq 0,$$
$$g_3(x) = x_1 - x_4 \leq 0,$$
$$g_4(x) = 0.10471x_1^2 + 0.04811x_3x_4(14 + x_2) - 5.0 \leq 0,$$
$$g_5(x) = 0.125 - x_1 \leq 0,$$
$$g_6(x) = \delta(x) - 0.25 \leq 0,$$
$$g_7(x) = 6000 - p_c(x) \leq 0$$

where

$$\tau(x) = \sqrt{\tau' + (2\tau\tau')\frac{x_2}{2R} + (\tau'')^2},$$
$$\tau' = \frac{6000}{\sqrt{2}x_1x_2},$$
$$\tau'' = \frac{MR}{J},$$
$$M = 6000\left(14 + \frac{x_2}{2}\right),$$
$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2},$$
$$J = 2\left\{x_1x_2\sqrt{2}\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\},$$
$$\sigma(x) = \frac{504000}{x_4x_3^2}$$
$$\delta(x) = \frac{65856000}{(30 \cdot 10^6)x_4x_3^3},$$
$$p_c(x) = \frac{4.013(30 \cdot 10^6)\sqrt{\frac{x_3^2x_4^6}{36}}}{196}\left(1 - \frac{x_3}{28}\sqrt{\frac{30 \cdot 10^6}{4(12 \cdot 10^6)}}\right)$$

with

$$0.1 \leq x_1, x_4 \leq 2 \text{ and } 0.1 \leq x_2, x_3 \leq 10.$$

Statistical results from 20 independent runs of the algorithms are presented in Table 2. Analysis of the mean indicator shows that the HOA algorithm achieved the best performance with a mean value of 1.724881, securing the top rank. Its extremely low standard deviation (0.000251) reflects high stability, uniform convergence, and minimal sensitivity to initial conditions. Furthermore, the proximity of the median (1.724853) to the mean indicates reproducibility and consistency of HOA across all runs.

Among the competing algorithms, MVO and TSA exhibited relatively competitive performance, ranking second and third, respectively; however, their higher standard deviations compared to HOA indicate greater variability and less uniformity across independent runs. AVOA, while achieving acceptable results in some runs, showed higher dispersion and therefore less stable performance than HOA.

Algorithms such as PSO and RSA demonstrated moderate performance, with higher mean values and greater fluctuations, indicating difficulty in simultaneously managing stress and displacement constraints. Other algorithms, including WOA, GSA, and TLBO, performed noticeably worse; TLBO, in particular, recorded extremely high worst-case values and standard deviations, highlighting severe instability and an inability to effectively guide the optimization process.

Overall, the results in Table 2 clearly demonstrate that HOA surpasses all competing algorithms in solution quality, stability, and reliability, confirming its capability to effectively navigate complex search spaces and nonlinear constraints in the welded beam design problem.

3.4 Speed reducer design

The speed reducer design problem is a key challenge in mechanical engineering and power transmission component design. The objective is to optimize the dimensions and mechanical parameters of the gearbox to minimize weight or total manufacturing cost, while satisfying performance

Table 3. Statistical results of the optimization algorithms for the speed reducer design problem

Algorithm	mean	best	worst	std	median	Rank
HOA	2996.348	2996.348	2996.348	1.97E-06	2996.348	1
AVOA	3003.191	2996.395	3013.157	15.15796	3002.484	2
RSA	3232.067	3194.518	3363.873	144.4699	3213.363	7
TSA	3034.359	3023.362	3049.132	23.32374	3032.698	3
MVO	3039.709	3006.23	3080.551	50.87287	3042.098	4
WOA	3250.492	3032.029	4056.46	875.1684	3147.242	8
TLBO	1.1E+14	4554.554	4.95E+14	4.02E+14	5.11E+13	10
GSA	3531.903	3131.569	4388.401	1036.742	3389.015	9
PSO	3045.007	2996.348	3188.325	151.8323	3035.626	5
GA	3166.619	3067.264	3213.834	99.70352	3175.633	6

and mechanical requirements, such as allowable stress limits, gear strength, and spatial constraints. The problem's complexity stems from nonlinear relationships among design parameters, high sensitivity to small dimensional variations in gears and shafts, and strong variable interactions. Accordingly, it is widely regarded as a standard benchmark for assessing the ability of metaheuristic algorithms to manage stringent and simultaneous constraints [37, 38].

Consider $X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7] = [b, m, p, l_1, l_2, d_1, d_2]$.

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

subject to:

$$g_1(x) = \frac{27}{x_1x_2^2x_3} - 1 \le 0,$$
$$g_2(x) = \frac{397.5}{x_1x_2^2x_3} - 1 \le 0,$$
$$g_3(x) = \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \le 0,$$
$$g_4(x) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \le 0,$$
$$g_5(x) = \frac{1}{110x_6^3}\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6} - 1 \le 0,$$
$$g_6(x) = \frac{1}{85x_7^3}\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6} - 1 \le 0,$$
$$g_7(x) = \frac{x_2x_3}{40} - 1 \le 0,$$
$$g_8(x) = \frac{5x_2}{x_1} - 1 \le 0,$$
$$g_9(x) = \frac{x_1}{12x_2} - 1 \le 0,$$
$$g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \le 0,$$
$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \le 0$$

with

$$2.6 \le x_1 \le 3.6, 0.7 \le x_2 \le 0.8, 17 \le x_3 \le 28, 7.3 \le x_4 \le 8.3, 7.8 \le x_5 \le 8.3, 2.9 \le x_6 \le 3.9, and\ 5 \le x_7 \le 5.5 .$$

Results from 20 independent algorithm runs are presented in Table 3. The HOA algorithm achieved the best performance with a mean of 2996.348 and a standard deviation of 1.97×10^{-6}, securing the top rank. The closeness of the median to the mean indicates uniform, stable, and reliable convergence across all runs.

Among competitors, AVOA and TSA showed relatively competitive results, but higher mean values and wider dispersion indicate lower stability and solution accuracy compared to HOA. Algorithms such as PSO, MVO, and WOA exhibited higher mean values and greater variability, reflecting challenges in balancing global and local search in the highly constrained and complex design space. RSA and GSA displayed poor convergence and severe instability, while TLBO, with extreme values and broad result dispersion, failed to manage constraints and search space effectively.

Overall, Table 3 demonstrates that HOA achieved the lowest objective values, highest stability, and superior reproducibility in the speed reducer design problem, highlighting its strong ability to integrate exploration and exploitation phases, manage optimal regions, and avoid entrapment in local minima.

3.5 Pressure vessel design

The pressure vessel design problem is a classic and challenging benchmark in mechanical

Table 4. Statistical results of the optimization algorithms for the pressure vessel design problem

Algorithm	mean	best	worst	std	median	Rank
HOA	6252.908	6017.5	6765.288	719.1787	6172.288	1
AVOA	6421.503	5885.338	7294.268	1357.486	6311.954	3
RSA	10479.53	6654.496	19918.35	8706.108	9795.543	8
TSA	6434.016	5935.573	7356.848	1306.753	6348.912	4
MVO	6710.232	5901.061	7333.978	1245.159	6698.786	5
WOA	8729.401	6070.274	13310.05	5857.135	8519.007	6
TLBO	30121.37	13304.53	52365.5	31935.67	31105.39	10
GSA	26917.32	13301.08	55128.82	30007.72	26559.04	9
PSO	6331.13	5886.328	7017.143	861.6094	6317.543	2
GA	10070.89	8618.778	12237.38	2891.603	9977.063	7

engineering and industrial component design. The main objective is to minimize the total manufacturing cost of the vessel while satisfying constraints related to wall thickness, volume, mechanical strength, and safety requirements. The complexity arises from the simultaneous presence of continuous and discrete variables and strong interactions among design parameters, where even small changes in wall thickness, radius, or height can significantly affect structural performance and cost. Consequently, this problem serves as a sensitive benchmark for evaluating a metaheuristic algorithm's ability to handle strict constraints, limited search spaces, and potentially unstable solutions.

The design problem involves four main variables: shell thickness, head thickness, base radius, and vessel height. These variables are subject to geometric and mechanical constraints while directly impacting total cost. Thus, the optimization algorithm must strike an effective balance between exploring new regions of the search space and exploiting promising solutions to achieve a stable, low-cost, and reliable outcome [27]:

Consider $X = [x_1, x_2, x_3, x_4] = [T_s, T_h, R, L]$.
Minimize $f(x) = 0.6224x_1x_3x_4 + 1.778x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$.

Subject to:
$g_1(x) = -x_1 + 0.0193x_3 \leq 0,$
$g_2(x) = -x_2 + 0.00954x_3 \leq 0,$
$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0,$

With
$0 \leq x_1, x_2 \leq 100, and\ 10 \leq x_3, x_4 \leq 200.$

The pressure vessel design problem is a classic and challenging benchmark in mechanical engineering and industrial component design. The main objective is to minimize the total manufacturing cost of the vessel while satisfying constraints related to wall thickness, volume, mechanical strength, and safety requirements. The complexity arises from the simultaneous presence of continuous and discrete variables and strong interactions among design parameters, where even small changes in wall thickness, radius, or height can significantly affect structural performance and cost. Consequently, this problem serves as a sensitive benchmark for evaluating a metaheuristic algorithm's ability to handle strict constraints, limited search spaces, and potentially unstable solutions.

The design problem involves four main variables: shell thickness, head thickness, base radius, and vessel height. These variables are subject to geometric and mechanical constraints while directly impacting total cost. Thus, the optimization algorithm must strike an effective balance between exploring new regions of the search space and exploiting promising solutions to achieve a stable, low-cost, and reliable outcome.

3.5 Visual performance analysis of algorithms based on boxplot charts

Fig. 2 presents the Boxplot charts of all algorithms' results across the four engineering design problems, providing a visual confirmation of the statistical outcomes reported in the previous tables. Across all problems, the boxes corresponding to the HOA algorithm clearly exhibit the narrowest interquartile ranges, the lowest overall dispersion, and the fewest outliers. These characteristics indicate rapid and stable convergence, high reproducibility, and reliable performance across independent runs. Moreover, the proximity of the median line to the mean in HOA's Boxplots demonstrates that the algorithm's results are consistently concentrated within the optimal region, without significant fluctuations caused by initial conditions or the stochastic nature of the search process.

Received: January 2, 2026. Revised: February 8, 2026.

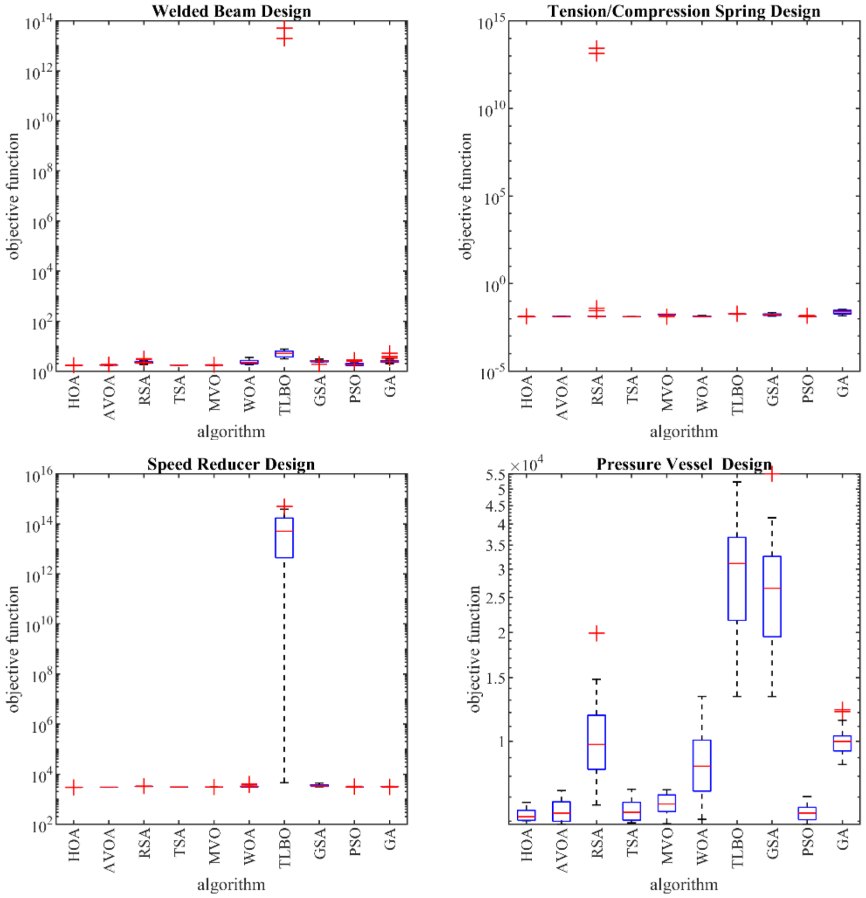

Figure. 2 Boxplot diagrams obtained from HOA and competing algorithms on studied optimization problems

In contrast, algorithms such as TLBO, RSA, and WOA display wide boxes, high dispersion, and numerous outliers, reflecting their inability to maintain uniform convergence and stable performance. While these algorithms may occasionally reach near-optimal solutions, the presence of extreme outliers in the Boxplots highlights their high sensitivity to initial values and instability during the optimization process.

Algorithms including PSO, MVO, and GA show variable performance; although some runs yield acceptable results, their interquartile ranges and box lengths are noticeably larger than HOA's, indicating an incomplete capability to balance exploration and exploitation in the highly constrained and complex search spaces. This effect is particularly pronounced in problems with strict constraints and strong nonlinear interactions, negatively impacting solution stability and reliability.

Overall, the visual analysis of Fig. 2 emphasizes that HOA successfully achieves an optimal balance between exploration and exploitation across all four design problems, delivering stable and dependable convergence. The compact boxes and limited outliers of HOA clearly demonstrate its superiority over competing algorithms in producing optimal, stable, and reproducible solutions. These features establish HOA as a robust and reliable tool for solving complex engineering design problems with stringent constraints.

3.7 Final discussion, advantages, and limitations of HOA

The results presented in the statistical tables and Boxplot charts clearly demonstrate that the proposed Hoatzin Optimization Algorithm (HOA) delivers superior and consistently stable performance across all four examined engineering design problems, achieving the top rank compared to all competing algorithms. This superiority is evident not only in terms of solution quality (mean and best indicators) but also in terms of stability, uniformity, and reproducibility of results (std and median), confirming the algorithm's rapid and reliable convergence.

The primary reason for this outstanding performance can be attributed to the hoatzin-inspired behavioral mechanism and the intelligent design of the exploration and exploitation phases. During the exploration phase, broad, sudden, and semi-directed displacements inspired by hoatzin movements enable effective search across the solution space, preventing entrapment in local minima. Gradual reduction of displacement intensity combined with controlled stochastic perturbations maintains a balance between diversity and search guidance, ensuring stable and purposeful exploration.

In the exploitation phase, simulating the hoatzin's local foraging behavior on a nutrient-rich branch facilitates precise and targeted local search around the best solutions. This mechanism enhances convergence accuracy, improves the quality of final solutions, and minimizes result dispersion, particularly in the later stages of the algorithm where attaining optimal solutions is critical.

Despite these advantages, HOA, like other metaheuristic algorithms, has certain limitations. These include sensitivity to initial population settings and increased computational cost for very high-dimensional problems. Nevertheless, comprehensive analysis of the results indicates that the behavioral design and intelligent structuring of the exploration and exploitation phases substantially mitigate these limitations. Consequently, HOA emerges as a powerful and reliable tool for solving complex engineering design problems characterized by stringent constraints and highly constrained search spaces.

4. Conclusion and future research directions

In this study, a novel metaheuristic algorithm, termed the Hoatzin Optimization Algorithm (HOA), was developed, inspired by the natural behaviors of the hoatzin bird in its habitat. The core concept of HOA is based on two distinctive behavioral patterns of the hoatzin:

- **Sudden and extensive movements** for global exploration to identify new food sources, corresponding to the exploration mechanism in global search within the algorithm.
- **Slow and localized movements** near discovered food sources, representing exploitation and local search for optimal utilization of the available resources.

The theoretical foundations and underlying concepts of HOA were thoroughly described and subsequently mathematically modeled in two phases: exploration and exploitation. The exploration phase simulates the hoatzin's wide-ranging natural movements to discover new resources, whereas the exploitation phase models the bird's precise, incremental movements around discovered resources to maximize the utilization of the current solutions.

The performance of HOA was evaluated on real-world optimization problems, including four constrained engineering design cases: tension/compression spring design, welded beam design, speed reducer design, and pressure vessel design. The results demonstrated that HOA, with its robust balance between exploration and exploitation, is capable of achieving high-quality, practical solutions for these engineering problems.

Moreover, comparative analysis against nine well-established metaheuristic algorithms revealed that HOA consistently attained the first rank across all four design problems, exhibiting a 100% superior performance relative to its competitors. These findings underscore that HOA is a powerful and reliable optimization tool with significant potential for application in diverse scientific, engineering, and industrial domains.

The introduction of HOA also opens several avenues for future research:

- Development of binary and multi-objective variants of HOA for application in discrete and multi-objective optimization problems.
- Application of HOA to a broader spectrum of optimization problems in science, engineering, industry, technology, and other real-world domains.
- Integration of HOA with other algorithms or machine learning frameworks to enhance performance in complex and large-scale problems.

In summary, the Hoatzin Optimization Algorithm, inspired by the natural behaviors of the hoatzin, provides a robust, flexible, and effective optimization framework and establishes a foundation for extensive future research in metaheuristic optimization.

Conflicts of Interest

The authors declare no conflict of interest.

Author Contributions

Conceptualization, T.H, O.A, M.S.A, and A.S; methodology, M.D, Z.M, I.L, and A.S; software, K.E, I.L, and O.P.M; validation, K.E, M.S.A, M.D, and O.P.M; formal analysis, T.H, A.S, Z.M, O.A, and O.P.M; investigation, O.A, M.S.A, A.S, and T.H; resources, Z.M, I.L, and T.H; data curation, Z.M and M.S.A; writing—original draft preparation, O.A, I.L, and A.S; writing—review and editing, T.H, M.S.A, O.P.M, and K.E; visualization, K.E, I.L, and O.P.M; supervision, M.D and Z.M; project administration, K.E, O.P.M, and O.A; funding acquisition, K.E.

Received: January 2, 2026. Revised: February 8, 2026.

References

[1] R. Rai, B. Sasmal, A. Das, T. Bharasa, K. G. Dhal, and P. K. Naskar, "Opposition based learning for metaheuristic algorithms: Theory, variants, applications, and performance evaluation", *Swarm and Evolutionary Computation,* Vol. 100, p. 102271, 2026, doi: 10.1016/j.swevo.2025.102271.

[2] A. Alshanty, W. G. Alshanti, and A. Zraiqat, "Early Detection of Cyberattacks in Banking Networks via a Fractional Partial Differential Equation Model", *Journal of Applied Mathematics,* Vol. 2026, No. 1, p. 4338391, 2026, doi: 10.1155/jama/4338391.

[3] H. Jia, X. Zhou, J. Zhang, and S. Mirjalili, "Superb Fairy-wren Optimization Algorithm: a novel metaheuristic algorithm for solving feature selection problems", *Cluster Computing,* Vol. 28, No. 4, p. 246, 2025, doi: 10.1007/s10586-024-04901-w.

[4] F. H. Rizk, K. S. Gaber, M. M. Eid, D. S. Khafaga, A. A. Alhussan, and E.-S. M. El-kenawy, "Dynamic binary swordfish movement optimization algorithm for feature selection", *Journal of Big Data,* Vol. 13, No. 1, p. 8, 2026, doi: 10.1186/s40537-025-01328-x.

[5] R. A. Zitar *et al.*, "Solenodon Optimization Algorithm (SOA): A Bio-inspired Metaheuristic for Efficient Global Optimization", *International Journal of Intelligent Engineering & Systems,* Vol. 19, No. 2, pp. 636-652, 2026, doi: 10.22266/ijies2026.0228.40.

[6] M. Dehghani, Z. Montazeri, E. Trojovská, and P. Trojovský, "Coati Optimization Algorithm: A new bio-inspired metaheuristic algorithm for solving optimization problems", *Knowledge-Based Systems,* Vol. 259, p. 110011, 2023, doi: 10.1016/j.knosys.2022.110011.

[7] C. Zhong, G. Li, Z. Meng, H. Li, A. R. Yildiz, and S. Mirjalili, "Starfish optimization algorithm (SFOA): a bio-inspired metaheuristic algorithm for global optimization compared with 100 optimizers", *Neural Computing and Applications,* Vol. 37, No. 5, pp. 3641-3683, 2025, doi: 10.1007/s00521-024-10694-1.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization", in *Proceedings of ICNN'95 - International Conference on Neural Networks*, 27 Nov.-1 Dec. 1995 1995, Vol. 4, pp. 1942-1948 Vol.4, doi: 10.1109/ICNN.1995.488968.

[9] X. Wang, "Draco lizard optimizer: a novel metaheuristic algorithm for global optimization problems", *Evolutionary Intelligence,* Vol. 18, No. 1, pp. 1-20, 2025.

[10] S. Arora and S. Singh, "Butterfly optimization algorithm: a novel approach for global optimization", *Soft Computing,* Vol. 23, No. 3, pp. 715-734, 2019, doi: 10.1007/s00500-018-3102-4.

[11] P. Trojovský and M. Dehghani, "Pelican Optimization Algorithm: A Novel Nature-Inspired Algorithm for Engineering Applications", *Sensors,* Vol. 22, No. 3, p. 855, 2022.

[12] D. E. Goldberg and J. H. Holland, "Genetic Algorithms and Machine Learning", *Machine Learning,* Vol. 3, No. 2, pp. 95-99, 1988, doi: 10.1023/A:1022602019183.

[13] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces", *Journal of Global Optimization,* Vol. 11, No. 4, pp. 341-359, 1997.

[14] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A Gravitational Search Algorithm", *Information Sciences,* Vol. 179, No. 13, pp. 2232-2248, 2009, doi: 10.1016/j.ins.2009.03.004.

[15] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm", *Knowledge-Based Systems,* Vol. 191, p. 105190, 2020, doi: 10.1016/j.knosys.2019.105190.

[16] S. Kirkpatrick, C. D. Gelatt Jr, and M. P. Vecchi, "Optimization by simulated annealing", *Science,* Vol. 220, No. 4598, pp. 671-680, 1983.

[17] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization", *Neural Computing and Applications,* Vol. 27, No. 2, pp. 495-513, 2016, doi: 10.1007/s00521-015-1870-7.

[18] I. Matoušová, P. Trojovský, M. Dehghani, E. Trojovská, and J. Kostra, "Mother optimization algorithm: a new human-based metaheuristic approach for solving engineering optimization", *Scientific Reports,* Vol. 13, No. 1, p. 10312, 2023, doi: 10.1038/s41598-023-37537-8.

[19] S. O. Oladejo, S. O. Ekwe, and S. Mirjalili, "The Hiking Optimization Algorithm: A novel human-based metaheuristic approach", *Knowledge-Based Systems,* Vol. 296, p. 111880, 2024, doi: 10.1016/j.knosys.2024.111880.

[20] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems", *Computer-Aided Design,* Vol. 43, No. 3, pp. 303-315, 2011, doi: 10.1016/j.cad.2010.12.015.

Received: January 2, 2026. Revised: February 8, 2026.

[21] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", *IEEE Transactions on Evolutionary Computation,* Vol. 1, No. 1, pp. 67-82, 1997.

[22] J. H. Holland, "Genetic Algorithms", *Scientific American,* Vol. 267, No. 1, pp. 66-73, 1992.

[23] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm", *Advances in Engineering Software,* Vol. 95, pp. 51-67, 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[24] S. Kaur, L. K. Awasthi, A. L. Sangal, and G. Dhiman, "Tunicate Swarm Algorithm: A new bio-inspired based metaheuristic paradigm for global optimization", *Engineering Applications of Artificial Intelligence,* Vol. 90, p. 103541, 2020, doi: 10.1016/j.engappai.2020.103541.

[25] L. Abualigah, M. A. Elaziz, P. Sumari, Z. W. Geem, and A. H. Gandomi, "Reptile Search Algorithm (RSA): A nature-inspired meta-heuristic optimizer", *Expert Systems with Applications,* Vol. 191, p. 116158, 2022, doi: 10.1016/j.eswa.2021.116158.

[26] B. Abdollahzadeh, F. S. Gharehchopogh, and S. Mirjalili, "African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems", *Computers & Industrial Engineering,* Vol. 158, p. 107408, 2021, doi: 10.1016/j.cie.2021.107408.

[27] B. Kannan and 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,* Vol. 116, No. 2, pp. 405-411, 1994.