International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Addax Optimization Algorithm: A Novel Nature-Inspired Optimizer for Solving Engineering Applications

Tareq Hamadneh[1] Khalid Kaabneh[2] Omar Alssayed[3] Kei Eguchi[4]* Saikat Gochhait[5,6] Irina Leonova[7] Mohammad Dehghani[8]

[1]Department of Matematics, Al Zaytoonah University of Jordan, Amman 11733, Jordan
[2]Faculty of Information Technology, Al-Ahliyya Amman University, Amman, 19328, Jordan
[3]Department of Mathematics, Faculty of Science,
The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan
[4]Department of Information Electronics, Fukuoka Institute of Technology, Japan
[5]Symbiosis Institute of Digital and Telecom Management,
Constituent of Symbiosis International Deemed University, Pune 412115, India
[6]Neuroscience Research Institute, Samara State Medical University, 89 Chapaevskaya str., 443001 Samara, Russia
[7]Faculty of Social Sciences, Lobachevsky University, 603950 Nizhny Novgorod, Russia
[8]Author's Department of Power and Control Engineering, School of Electrical and Computer Engineering, Shiraz University, Shiraz 71557-13876, Iran
* Corresponding author's Email: eguti@fit.ac.jp

Abstract: This paper introduces a novel nature-inspired optimization algorithm called the Addax Optimization Algorithm (AOA), which emulates the natural behavior of addax in the wild. The core inspiration for AOA is drawn from the addax's foraging strategy and digging skills. The theoretical foundation of AOA is expounded and mathematically modeled in two phases: (i) exploration based on modeling addax position change during foraging and (ii) exploitation based on addax position change modeling during digging. The efficiency of AOA in handling real-world engineering applications is evaluated on four engineering design problems. The optimization results show that AOA is achieved effective solutions for optimization problems with its high ability in exploration, exploitation, and establishing a balance between them during the search process. The outcomes derived from applying AOA are compared with the performance of twelve well-known optimization algorithms. The simulation results show that AOA is provided superior performance compared to competitor algorithms, by achieving better results and ranking as the first best optimizer. The simulation findings show that the proposed AOA approach has an effective performance for handling optimization tasks in engineering applications.

Keywords: Optimization, Optimization algorithm, Engineering application, Bio-inspired, addax, Exploration, Exploitation.

1. Introduction

Optimization problems encompass a variety of potential solutions, with the ultimate aim of attaining the best feasible outcome known as the global optimum [1]. In mathematical terms, these problems typically consist of decision variables, constraints, and an objective function. The objective in optimization is to set the decision variables within the constraints so that the objective function reaches its optimal value, whether it be a maximum or minimum [2]. Across mathematics, engineering, industry, real-world scenarios, and other scientific disciplines, a multitude of optimization challenges exist that require specialized techniques for solution [3, 4].

Metaheuristic algorithms stand out as highly effective stochastic methods for addressing optimization challenges. By employing random search within the problem-solving realm, along with random operators and trial and error processes, these algorithms can uncover suitable solutions [5]. The nature of this random optimization approach means that there is no assurance or promise of reaching the global optimum with the use of metaheuristic algorithms. Hence, the solutions achieved through these methods are deemed quasi-optimal. This realization, coupled with the aspiration to attain improved quasi-optimal results, serves as the primary driving force and motivation for researchers working on advancing various metaheuristic algorithms [6].

Two crucial concepts that play a key role in the effective implementation of metaheuristic algorithms for navigating random searches within problem-solving spaces are exploration and exploitation. Exploration refers to the algorithm's capacity for conducting broad searches across the problem-solving space, allowing it to avoid getting trapped in local solutions and to pinpoint global optimal regions through thorough exploration. Exploitation, on the other hand, refers to the algorithm's ability to conduct focused searches around identified solutions and promising areas within the problem-solving space, aiming to yield improved solutions. Alongside the important roles of exploration and exploitation, a critical factor contributing to the success of metaheuristic algorithms in optimization is the skillful balance between these two aspects throughout the search process [7].

In the realm of metaheuristic algorithms, a key question arises regarding the need for continued development and introduction of new algorithms, despite the plethora of existing ones. The No Free Lunch (NFL) theorem [8] addresses this by highlighting that the effectiveness of a metaheuristic algorithm in solving specific optimization problems does not guarantee similar success across all problems. This implies that no single algorithm can be universally deemed the best optimizer for all optimization tasks. The NFL theorem underscores the unpredictability of implementing a metaheuristic algorithm in any given optimization scenario, fueling ongoing research efforts to devise more efficient solutions through the innovation of new metaheuristic algorithms.

This paper introduces a novel and innovative contribution through the development of the Addax Optimization Algorithm (AOA), a bio-inspired metaheuristic designed to tackle optimization challenges across diverse scientific fields and engineering applications. The distinctive contributions of this research are summarized as follows:

- AOA is crafted by emulating the natural behaviors of addax in their habitat.
- The basic inspiration of AOA is taken from addax's foraging and digging capabilities.
- The theory of AOA is articulated and mathematically modeled in two phases: (i) exploration based on the simulation of the foraging process and (ii) exploitation based on the simulation of the drilling skill.
- The effectiveness of AOA in handling real-world engineering applications is challenged to optimize four engineering design problems.
- The outcomes obtained from AOA are compared with the performance of twelve widely recognized metaheuristic algorithms.

The paper is organized as follows: Section 2 presents the literature review, followed by the introduction and modeling of the proposed Addax Optimization Algorithm (AOA) in Section 3. The efficiency of AOA in addressing real-world engineering applications is explored in Section 4. Section 5 concludes the paper with recommendations for future research.

2. Literature review

Inspired by natural processes, living organisms, physics laws, genetics, biology, human behavior, and other natural evolutionary phenomena, metaheuristic algorithms have been created. They are classified into five groups based on their design inspiration: swarm-based, evolutionary-based, physics-based, human-based, and game-based approaches.

Swarm-based metaheuristic algorithms mimic the collective behavior seen in various creatures in nature such as birds, animals, insects, reptiles, and aquatic organisms. Some of the well-known examples of these algorithms include Ant Colony Optimization (ACO) [9], Particle Swarm Optimization (PSO) [10], Artificial Bee Colony (ABC) [11], and Firefly Algorithm (FA) [12]. ACO is inspired by the effective communication strategy of ant swarms in finding the shortest path to food sources. PSO draws from the group movements of fish and birds to effectively locate food. ABC models the hierarchical behaviors within a bee colony to efficiently gather food. FA takes its cue from the coordination and exchange of information among fireflies. The diverse abilities and behaviors seen in wildlife, like hunting, foraging, migrating, and digging, are harnessed in developing metaheuristic algorithms such as: Termite Alate Optimization Algorithm (TAOA) [13], Reptile Search Algorithm (RSA) [14], Tunicate Swarm Algorithm (TSA) [15], Electric Eel Foraging Optimization (EEFO) [16], Grey Wolf Optimizer (GWO) [17], African Vultures Optimization Algorithm (AVOA) [18], Whale Optimization

Algorithm (WOA) [19], and White Shark Optimizer (WSO) [20].

Evolutionary-inspired metaheuristic algorithms are created by mimicking principles and concepts from biological and genetic sciences. These algorithms are influenced by ideas like natural selection, survival of the fittest, Darwin's theory of evolution, and genetic principles. Genetic Algorithm (GA) [21] and Differential Evolution (DE) [22] are among the most well-known algorithms in this category, developed by replicating the natural reproduction process and incorporating genetic concepts like mutation, crossover, and selection. Artificial Immune System (AIS) design is based on mimicking the defense mechanisms of the human body against pathogens and illnesses [23].

Physic-based metaheuristic algorithms involve mimicking various phenomena, cycles, processes, laws, forces, and other concepts from the field of physics. Simulated Annealing (SA) is a well-known example of such algorithms, which mirrors the melting and freezing process of annealing metals, where metals are heated to a liquid state and then slowly cooled to form crystalline structures [24]. Physical forces and Newton's laws of motion are employed in designing algorithms such as: Spring Search Algorithm (SSA) by imitating Hooke's law and the tension force of a spring in a system consisting of weights connected to springs [25], Gravitational Search Algorithm (GSA) by imitating the gravitational attraction between masses that are at different distances from each other [26], and Momentum Search Algorithm (MSA) by imitating the force resulting from the impact of two balls [27].

Human-based metaheuristic algorithms are created by mimicking the behaviors, choices, thoughts, decisions, communication, interactions, and other activities exhibited by humans in their personal and social lives. Among these algorithms, Teaching-Learning Based Optimization (TLBO) stands out as it replicates the educational dynamics between teachers and students working together in a classroom setting [28]. The Mother Optimization Algorithm (MOA) draws its inspiration from Eshrat's attentive care towards her children [6]. Similarly, the Election-Based Optimization Algorithm (EBOA) models the strategies employed in elections and voting systems [29]. Reflecting on the importance of collaboration in team settings, the Teamwork Optimization Algorithm (TOA) focuses on emphasizing cooperation and teamwork to attain common goals [30]. Other examples of human-inspired algorithms include the: Random Selected Leader Based Optimizer (RSLBO) [31], Multi leader optimizer (MLO) [32], and Mixed Leader Based Optimizer (MLBO) [33].

Game-based metaheuristic algorithms are developed by imitating the rules of individual and group games as well as the behavior of players, coaches, referees and other influential people in these games. Darts Game Optimizer (DGO) is one of the most prominent game-based metaheuristic algorithms, which is inspired by the efforts of players in the game of darts in order to collect more points in their throws towards the scoreboard [34]. The design of Golf Optimization Algorithm (GOA) is imitated the skills of players in hitting the ball in the game of golf to put the ball in the hole [35]. Hide Object Game Optimizer (HOGO) is introduced based on players' efforts to find the hidden object on the playing field [36]. Holding league matches and competition between clubs has been a source of inspiration in designing algorithms such as: Football Game Based Optimization (FGBO) [37] and Volleyball Premier League (VPL) [38]. Some other game-based metaheuristic algorithms are Shell game optimization (SGO) [39], Orientation Search Algorithm (OSA) [40], Puzzle Optimization Algorithm (POA) [41], Ring toss game based optimization (RTGBO) [42], and Dice Game Optimizer (DGO) [43].

According to the literature review, there hasn't been any metaheuristic algorithm developed yet that is influenced by the natural behavior of addax in their habitat. Nevertheless, the foraging strategy and digging prowess displayed by addax showcase intelligent behaviors that could serve as the foundation for a novel optimizer. To fill this research gap on metaheuristic algorithms, this paper introduces a novel algorithm that is inspired by the mathematical representation of addax natural behaviors in the wild, as elaborated in the subsequent section.

3. Addax optimization Algorithm

In this section, first, the basic inspiration employed in designing the proposed Addax Optimization Algorithm (AOA) approach is described, then its implementation steps are mathematically modeled.

3.1 Inspiration of AOA

The addax, also known as the screwhorn antelope and white antelope, is a species of antelope that native to the Sahara Desert. This animal's coat varies in color based on the season, with a nearly all-white or sandy blonde appearance in the summer and a grayish-brown color with white hindquarters and legs in the winter, along with long, brown hair on the head, shoulders, and neck [44]. Addax has prominent red nostrils and scraggly beards. The head of this animal has black and brown patches. Black and long hair protrudes between the spiral and curved horns of the addax and ends in a short mane on the neck [45]. Females' addaxes stand from 95 to 110 cm at the shoulder, with male from 105 to 115 cm. The body and head length are 120 to 130 cm in both sexes, with a long tail from 25 to 35 cm. The weight of females varies from 60 to 90 kg, and that of males varies from 100 to 125 kg [44]. An image of the addax is shown in Fig. 1.

Addax's diet mainly consists of grasses, and leaves of any available shrubs, leguminous herbs and bushes. Considering that addax can live without water for a long time, this animal is well adapted to live in its desert habitat. Addax herds consist of five to twenty members of both female and male species, led by the oldest female [46]. With its good digging ability, Addax digs inside the sand in order to rest and deal with the heat. Also, these dug depressions protect addax from sandstorms. Addax usually rests in shady and dug-out places and wander widely only to forage. This animal has a good ability to track rainfall and as a result moves to areas with more vegetation [47].

Within the natural behaviors exhibited by addaxes in their habitat, (i) the foraging strategy and (ii) digging skills stand out prominently. The proposed Addax Optimization Algorithm (AOA) leverages mathematical modeling of these intelligent activities in nature, as elaborated below.

3.2 Algorithm Initialization

The proposed AOA methodology presents a population-based metaheuristic algorithm where addaxes form the population. AOA excels in offering effective solutions to optimization challenges through an iterative process, harnessing the search capabilities of its members. Each addax defines values for design variables based on its location in the problem-solving space. Consequently, the position of each addax within the population serves as a potential solution for the given problem. In the AOA framework, the habitat of addaxes mirrors the problem-solving space, with the position of each addax representing a candidate solution. Mathematically, the position of each addax, considered a candidate solution, can be represented as a vector with a length equal to the number of decision variables. Additionally, the AOA population, comprising addaxes and their associated vectors, can be formally modeled as a matrix using Eq. (1). The initial placement of addaxes in the problem-solving space is randomly initialized according to Eq. (2).

Figure. 1 Addax taken from: free media Wikimedia Commons

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} = \begin{bmatrix} x_{1,1} \cdots x_{1,d} \cdots x_{1,m} \\ \vdots \quad \ddots \quad \vdots \quad \iddots \quad \vdots \\ x_{i,1} \cdots x_{i,d} \cdots x_{i,m} \\ \vdots \quad \iddots \quad \vdots \quad \ddots \quad \vdots \\ x_{N,1} \cdots x_{N,d} \cdots x_{N,m} \end{bmatrix}_{N\times m} \quad (1)$$

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

In this context, X denotes the population matrix of AOA, X_i refers to the ith addax (candidate solution), and $x_{i,d}$ signifies its dth dimension in the search space (decision variable). N represents the number of addax in the population, m is the count of decision variables, r is a random number within the interval $[0,1]$, while lb_d and ub_d stand for the lower and upper bounds of the dth decision variable, respectively.

Given that the position of each addax serves as a candidate solution, the corresponding objective function for each addax can be assessed. The values evaluated for the objective function form a vector as per Eq. (3).

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

In this context, F represents the vector of assessed objective functions, where F_i denotes the evaluation of the objective function based on the ith addax.

The computed values of the objective function serve as effective indicators for assessing the performance of each addax in offering a potential solution. In line with this, the best calculated value of the objective function aligns with the best member of

Received: March 13, 2024. Revised: April 19, 2024.

the addax group, while the worst calculated value corresponds to the worst member. Given that AOA operates on an iterative basis with the addax's position being adjusted in every iteration, it becomes imperative to update and retain the best member as the prime candidate solution in each iteration.

3.3 Mathematical modelling of AOA

The proposed AOA approach is a population-based metaheuristic algorithm that imitates the natural behavior of addax in the wild in order to update the position of its population members in the problem solving space. Addaxes have a high ability to track rain and identify areas with more vegetation in order to foraging. In addition, this animal has a high skill in digging in order to create suitable depressions in order to rest and deal with sandstorms. These clever addax behaviors are the main sources of inspiration in the design of AOA. According to this, in AOA design, the position of each addax in the problem solving space is updated in each iteration based on two phases (i) exploration based on foraging process simulation and (ii) exploitation based on digging skill simulation. Each of these update phases is described in detail below and is mathematically modeled.

3.3.1 Phase 1: foraging process (exploration phase)

During the first phase of AOA, the adjustment of the population members' positions in the problem-solving space is determined by mathematically modeling the shifts observed in the addaxes' positions during foraging. Residing in arid landscapes, addaxes primarily feed on grasses, shrub leaves, leguminous plants, herbs, and existing bushes. Their adept foraging process stems from their ability to track rainfall and locate areas with abundant vegetation. Despite being part of a herd, addaxes conduct individual and extensive searches to locate food sources. The extensive positional changes in addaxes during foraging are modeled in AOA, resulting in significant shifts in the positions of population members within the problem-solving space. Consequently, this enhances the algorithm's exploration capability for global search management.

In the AOA design, each addax considers the positions of other addaxes with better objective function values as suitable foraging areas, as defined by Eq. (4).

$$CA_i = \{X_k : F_k < F_i \ and \ k \neq i\},\ \text{where } i = 1,2,\ldots,N \text{ and } k \in \{1,2,\ldots,N\} \tag{4}$$

In this context, CA_i represents the set of potential areas with appropriate vegetation for foraging for the *i*th addax. X_k denotes the population member with a better objective function value compared to the *i*th addax, and F_k represents its objective function value.

In the design of AOA, it is assumed that each addax randomly chooses one of these designated areas in order to forage. Based on the modeling of the movement of addax towards suitable vegetation areas in the foraging process, a new position for each member of the AOA population has been calculated using Eq. (5). Subsequently, if this new position results in an enhancement in the objective function value, it supersedes the previous position of the corresponding member as outlined in Eq. (6).

$$x_{i,j}^{P1} = x_{i,j} + r_{i,j} \cdot (SA_{i,j} - I_{i,j} \cdot x_{i,j}), \tag{5}$$

$$X_i = \begin{cases} X_i^{P1}, & F_i^{P1} \leq F_i, \\ X_i, & else, \end{cases} \tag{6}$$

In this context, SA_i denotes the chosen area for foraging by the *i*th addax, $SA_{i,j}$ represents its jth dimension in that area. X_i^{P1} signifies the newly computed position for the ith addax, derived from the foraging phase of the proposed AOA, where $x_{i,j}^{P1}$ denotes its jth dimension. F_i^{P1} corresponds to the objective function value for the *i*th addax in this phase. Additionally, $r_{i,j}$ represents random numbers drawn from the interval $[0, 1]$, and $I_{i,j}$ denotes randomly selected numbers, taking values of 1 or 2.

3.3.2 Phase 2: digging skill (exploitation phase)

In the second phase of AOA, the adjustment of population members' positions in the problem-solving space is determined by mathematically modeling the shifts in the addaxes' positions during the sand-digging process. Addaxes start digging in shady places during the day and rest in the depressions created. This digging process and depressions also protect addaxes from sandstorms. As it is known, during the drilling process, the position of addaxes will have small changes. The modeling of these minor positional adjustments during digging leads to corresponding slight changes in the positions of AOA population members within the problem-solving space, thereby enhancing the algorithm's exploitation power for effective local search management.

In the AOA design, informed by the modeled changes in addax positions during digging, a new position is computed for each AOA population member using Eq. (7). Subsequently, if this new position results in an improvement in the objective function value, it supersedes the previous position of the corresponding member in accordance with Eq. (8).

$$x_{i,j}^{P2} = x_{i,j} + \left(1 - 2\, r_{i,j}\right) \cdot \frac{ub_j - lb_j}{t} \quad (7)$$

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

In this context, X_i^{P2} denotes the newly calculated position for the *i*th addax, derived from the digging phase of the proposed AOA, $x_{i,j}^{P2}$ represents its jth dimension, F_i^{P2} corresponds to its objective function value, $r_{i,j}$ represents random numbers drawn from the interval $[0, 1]$, and t stands for the iteration counter.

3.4 Repetition process, pseudocode, and flowchart of AOA

The first iteration of AOA is completted by adjusting the location of all addaxes within the problem solution space according to the foraging and digging stages. Subsequently, the algorithm progresses to the next iteration with the revised addax positions and updated objective function values, utilizing Eqs. (4) to (8) for further enhancements until the final iteration. Each iteration updates and saves the best solution obtained up to that point. Upon the algorithm's completion, the best solution recorded throughout the iterations is identified as the ultimate AOA solution for the given problem. Algorithm 1 presents the pseudocode of AOA implementation.

4. AOA for real-world engineering applications

One of the key applications of metaheuristic algorithms lies in their ability to address optimization challenges in real-world engineering applications. In this section, we have chosen four engineering design problems with the objective of assessing the effectiveness of AOA in tackling practical applications. The results obtained from AOA are compared with the performance of twelve well-known algorithms: Genetic Algorithm (GA) [21], Particle Swarm Optimization (PSO) [10], Gravitational Search Algorithm (GSA) [26], Teaching-Learning Based Optimization (TLBO) [28], Grey Wolf Optimizer (GWO) [17], Multi-Verse Optimizer (MVO) [48], Marine Predator Algorithm (MPA) [49], Reptile Search Algorithm (RSA) [14], Tunicate Swarm Algorithm (TSA) [15], African Vultures Optimization Algorithm (AVOA) [18], Whale Optimization Algorithm (WOA) [19], and White Shark Optimizer (WSO) [20].

Algorithm 1. Pseudocode of AOA.

Start AOA.

1. Input problem information: variables, objective function, and constraints.
2. Set AOA population size (N) and iterations (T).
3. Generate the initial population matrix at random using Eq. (2). $x_{i,d} \leftarrow lb_d + r \cdot (ub_d - lb_d)$
4. Evaluate the objective function.
5. For $t = 1$ to T
6. For $i = 1$ to N
7. Phase 1: foraging (exploration phase)
8. Determine the candidate foraging area set for the *i*th addax using Eq. (4). $CA_i \leftarrow \{X_{k_i}: F_{k_i} < F_i \text{ and } k_i \neq i\}$
9. Select the target foraging area for the *i*th addax at random.
10. Calculate new position of *i*th addax using Eq. (5). $x_{i,d}^{P1} \leftarrow x_{i,d} + r \cdot \left(SA_{i,d} - I \cdot x_{i,d}\right)$
11. Update *i*th addax using Eq. (6). $X_i \leftarrow \begin{cases} X_i^{P1}, & F_i^{P1} < F_i \\ X_i, & else \end{cases}$
12. Phase 2: digging (exploitation phase)
13. Calculate new position of *i*th addax using Eq. (7). $x_{i,d}^{P2} \leftarrow x_{i,d} + (1 - 2r) \cdot \frac{(ub_d - lb_d)}{t}$
14. Update *i*th addax using Eq. (8). $X_i \leftarrow \begin{cases} X_i^{P2}, & F_i^{P2} < F_i \\ X_i, & else \end{cases}$
15. end
16. Save the best candidate solution so far.
17. end
18. Output the best quasi-optimal solution obtained with the AOA.

End AOA.

Received: March 13, 2024. Revised: April 19, 2024.

4.1 Pressure vessel design problem

Pressure vessel design is proposed as an optimization challenge in engineering where minimizing construction cost is its main goal. The mathematical model of this design is as follows [50]:

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:

$g_1(x) = -x_1 + 0.0193x_3 \leq 0,$

$g_2(x) = -x_2 + 0.00954x_3 \leq 0,$

$g_3(x) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \leq 0,$

$g_4(x) = x_4 - 240 \leq 0.$

With

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

The results of applying the AOA and alternative algorithms to the pressure vessel design problem are illustrated in Table 1.

Based on the simulation findings, AOA has successfully attained the optimal design, characterized by specific values for the design variables—namely, (0.7780271, 0.3845792, 40.312284, 200)—and an associated objective function value of (5882.8955). Examination of the simulation results suggests that AOA surpasses its competitors in optimizing pressure vessel design, yielding superior outcomes.

4.2 Speed reducer design problem

Speed reducer design is proposed as an optimization challenge in engineering where minimizing the weight of the speed reducer is its main goal. The mathematical model of this design is as follows [51]:

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:

$$g_1(x) = \frac{27}{x_1x_2^2x_3} - 1 \leq 0,$$

$$g_2(x) = \frac{397.5}{x_1x_2^2x_3} - 1 \leq 0,$$

$$g_3(x) = \frac{1.93x_4^3}{x_2x_3x_6^4} - 1 \leq 0,$$

$$g_4(x) = \frac{1.93x_5^3}{x_2x_3x_7^4} - 1 \leq 0,$$

$$g_5(x) = \frac{\sqrt{\left(\frac{745x_4}{x_2x_3}\right)^2 + 16.9 \times 10^6}}{110x_6^3} - 1 \leq 0,$$

$$g_6(x) = \frac{\sqrt{\left(\frac{745x_5}{x_2x_3}\right)^2 + 157.5 \times 10^6}}{85x_7^3} - 1 \leq 0,$$

$$g_7(x) = \frac{x_2x_3}{40} - 1 \leq 0,$$

$$g_8(x) = \frac{5x_2}{x_1} - 1 \leq 0,$$

$$g_9(x) = \frac{x_1}{12x_2} - 1 \leq 0,$$

$$g_{10}(x) = \frac{1.5x_6 + 1.9}{x_4} - 1 \leq 0,$$

$$g_{11}(x) = \frac{1.1x_7 + 1.9}{x_5} - 1 \leq 0.$$

With

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

Table 2 exhibits the outcomes of implementing the AOA and rival algorithms for the speed reducer design problem. Based on the simulation findings, AOA has successfully attained the optimal design, characterized by specific values for the design variables—namely, (3.5, 0.7, 17, 7.3, 7.8, 3.3502147, 5.2866832)—and an associated objective function value of (2996.3482). Upon comparing the simulation results, it can be inferred that AOA excels in optimizing the speed reducer design compared to its competitors. This superiority is evident in the improved values achieved for both the objective function and design variables.

4.3 Welded beam design problem

The welded beam design is suggested as an engineering optimization issue, with the primary objective of decreasing the welded beam's fabrication cost. The mathematical model for this design is as follows [19]:

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:

$g_1(x) = \tau(x) - 13600 \leq 0,$

$g_2(x) = \sigma(x) - 30000 \leq 0,$

$g_3(x) = x_1 - x_4 \leq 0,$

$g_4(x) = 0.10471x_1^2 + 0.04811x_3x_4(14 + x_2) - 5.0 \leq 0,$

$g_5(x) = 0.125 - x_1 \leq 0,$

$g_6(x) = \delta (x) - 0.25 \leq 0,$

$g_7(x) = 6000 - p_c (x) \leq 0.$

Where

$$\tau(x) = \sqrt{(\tau')^2 + (2\tau\tau')\frac{x_2}{2R} + (\tau")^2},$$

$$\tau' = \frac{6000}{\sqrt{2}x_1x_2},$$

$$\tau" = \frac{MR}{J}, M = 6000\left(14 + \frac{x_2}{2}\right),$$

$$R = \sqrt{\frac{x_2^2}{4} + \left(\frac{x_1 + x_3}{2}\right)^2},$$

$$J = 2\left\{x_1x_2\sqrt{2}\left[\frac{x_2^2}{12} + \left(\frac{x_1 + x_3}{2}\right)^2\right]\right\},$$

$$\sigma(x) = \frac{504000}{x_4x_3^2},$$

$$\delta\,(x) = \frac{65856000}{(30 \cdot 10^6)x_4x_3^3},$$

$$p_c\,(x) = \frac{4.013(30 \cdot 10^6)\sqrt{\frac{x_3^2x_4^6}{36}}}{196}\left(1 - \frac{x_3}{28}\sqrt{\frac{30 \cdot 10^6}{4(12 \cdot 10^6)}}\right).$$

With

$0.1 \le x_1, x_4 \le 2$ and $0.1 \le x_2, x_3 \le 10.$

The outcomes of addressing the welded beam design challenge using the AOA and alternative techniques are presented in Table 3. According to the simulation results, AOA has produced the optimal design, characterized by specific values for the design variables—namely, (0.2057296, 3.4704887, 9.0366239, 0.2057296)—and an associated objective function value of (1.7246798). The results indicate that AOA surpasses competing algorithms in tackling the welded beam design challenge, yielding superior values for both design variables and the objective function.

4.4 Tension/compression spring design problem

The tension/compression spring is an optimization challenge in engineering with the main goal of minimizing construction costs. The mathematical model used for this design is outlined in [19]:

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

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

Subject to:

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

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

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

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

With

$0.05 \le x_1 \le 2, 0.25 \le x_2 \le 1.3$

and $2 \le x_3 \le 15.$

Results shown in Table 4 illustrate how the AOA outperformed competitor algorithms when tackling the tension/compression spring design problem. Upon analyzing the simulation outcomes, it is clear that AOA is delivered the optimal design, with design variable values of (0.0516891, 0.3567177, 11.288966) and objective function value of (0.0126019). These results indicate that AOA outshines its competitors by providing better values for design variables and demonstrating superior performance in optimizing tension/compression spring design.

5. Conclusions and future Works

This paper introduces a novel bio-inspired metaheuristic algorithm named the Addax Optimization Algorithm (AOA), which mimics the behavior of the addax in its natural environment. Drawing on the natural foraging and sand-digging strategies of the addax, the AOA algorithm was developed. The approach of AOA was delineated and further formalized into two key phases: (i) exploration based on the modeling of addax position change during foraging and (ii) exploitation based on the addax position change modeling during digging. To assess its effectiveness in real-world engineering applications, AOA was applied to address four engineering design challenges. The optimization outcomes revealed that AOA excels in exploration, exploitation, and maintaining a balance between them throughout the search process, leading to effective solutions for optimization problems. The performance of AOA was then benchmarked against twelve well-known metaheuristic algorithms to gauge its efficacy. The findings from the simulation revealed that AOA outperformed the other algorithms in the competition by delivering superior results across all four studied engineering design and securing the overall top position as the best optimizer.

The obtained results showed that AOA, while providing better results compared to competitor algorithms, has effective performance for handling optimization tasks in real-world engineering applications.

Received: March 13, 2024. Revised: April 19, 2024.

Table 1. Optimization results of pressure vessel design

Algorithm	Optimum cost	Optimum variables			
		T_s	T_h	R	L
AOA	5882.8955	0.7780271	0.3845792	40.312284	200
WSO	5882.9778	0.7858894	0.3884649	40.719594	202.01889
AVOA	5882.9669	0.7858899	0.3884658	40.719684	202.01732
RSA	7408.5687	1.1584247	0.6097612	58.998619	57.111334
MPA	5882.9622	0.7858872	0.3884644	40.719541	202.01933
TSA	5953.9718	0.798104	0.4111007	41.352501	193.93073
WOA	6170.1676	0.8857063	0.4381857	45.145186	152.54675
MVO	6087.5684	0.8859536	0.4375393	45.829139	143.13338
GWO	5900.966	0.7932495	0.3935973	41.093706	196.90515
TLBO	9902.6437	1.49906	0.5234439	52.808242	95.565562
GSA	10421.604	1.0612011	0.9725151	44.186921	187.04202
PSO	9295.3223	1.4879777	0.6285249	64.254793	40.025537
GA	9829.8871	1.342597	0.7295122	58.633793	67.896633

Table 2. Optimization results of speed reducer design

Algorithm	Optimum cost	Optimum variables						
		b	M	p	l_1	l_2	d_1	d_2
AOA	2996.3482	3.5	0.7	17	7.3	7.8	3.3502147	5.2866832
WSO	2996.3783	3.5353539	0.7	17	7.3737433	7.8787892	3.3840554	5.3400841
AVOA	2996.3782	3.5353535	0.7	17	7.373738	7.8	3.3840552	5.340084
RSA	3119.3107	3.5960727	0.7	17	8.1226301	8.1824999	3.387919	5.4696083
MPA	2996.3782	3.5353535	0.7	17	7.3737375	7.8	3.3840552	5.340084
TSA	3008.0756	3.5438507	0.7	17	7.3900271	8.1876773	3.3843004	5.3424133
WOA	3023.7141	3.5929776	0.7	17	7.3745971	8.0166892	3.3915647	5.3401318
MVO	3005.3188	3.5368369	0.7	17	7.3743303	8.1099271	3.3968232	5.3402335
GWO	2999.8997	3.5353698	0.7	17.17284	7.5673929	7.903132	3.3878193	5.3403164
TLBO	4479.4184	3.5723089	0.709704	23.313923	7.9016629	8.1061982	3.5903956	5.3747847
GSA	3110.0138	3.5504465	0.7088844	17.4149	7.7183329	7.9626496	3.4226368	5.4054777
PSO	3198.2916	3.540745	0.7071182	17.893511	7.4393297	8.0244587	3.5456123	5.3778941
GA	3224.9986	3.5867475	0.7107371	17.707845	7.6669631	7.9157848	3.6155179	5.3793806

Table 3. Optimization results of welded beam design

Algorithm	Optimum cost	Optimum variables			
		h	l	t	b
AOA	1.7246798	0.2057296	3.4704887	9.0366239	0.2057296
WSO	1.7251719	0.2075889	3.5102894	9.1278723	0.2078091
AVOA	1.7258677	0.2070851	3.5212134	9.127802	0.2078123
RSA	1.8939614	0.1981651	3.6348241	9.7123911	0.2157764
MPA	1.7251719	0.2075889	3.5102895	9.1278723	0.2078091
TSA	1.7339452	0.2061272	3.5370761	9.1432889	0.2084467
WOA	1.7957382	0.2012686	3.4464077	9.7117063	0.2065802
MVO	1.7303824	0.2055038	3.5511236	9.1477802	0.207753
GWO	1.7265182	0.2070084	3.5242211	9.1288908	0.2078362
TLBO	2.5751418	0.2774583	4.1882004	7.6537322	0.3522537
GSA	1.9754131	0.258731	3.1343445	8.0354119	0.277623
PSO	3.2306824	0.312711	3.5950047	8.0138356	0.4502653
GA	2.4072838	0.2155475	5.8810755	8.2895482	0.2727565

Table 4. Optimization results of tension/compression spring design

Algorithm	Optimum cost	Optimum variables		
		d	D	P
AOA	0.0126019	0.0516891	0.3567177	11.288966
WSO	0.0126794	0.0517918	0.3505866	12.088642
AVOA	0.01267001	0.0517415	0.3491414	12.094713
RSA	0.0130152	0.0507401	0.322288	14.378692
MPA	0.0126667	0.0520683	0.3569197	11.613394
TSA	0.0127212	0.0515446	0.3441149	12.513285
WOA	0.0126901	0.052371	0.3648092	11.3348
MVO	0.0129838	0.0529374	0.3830476	11.677867
GWO	0.0127019	0.051053	0.3332558	13.263881
TLBO	0.0158164	0.0623273	0.7021824	6.4851944
GSA	0.0129328	0.0543215	0.4125033	9.3291759
PSO	0.0157373	0.0625764	0.7071115	5.972288
GA	0.0164115	0.0654431	0.7852704	3.8973553

The introduction of AOA opens up several directs for future research endeavors. This study particularly highlights the potential for developing binary and multi-objective versions of AOA. Also, using AOA to address optimization issues in various sciences and real-world applications are among the other research proposals of this paper for further work in the future.

Conflicts of Interest

"The authors declare no conflict of interest."

Author Contributions

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

Acknowledgments

Acknowledgments are to show that the article is supported by what organization. For example, "This work was supported by the National Nature Science Foundation under Grant No. 405".

References

[1] A. Taheri, K. RahimiZadeh, A. Beheshti, J. Baumbach, R. V. Rao, S. Mirjalili, and A. H. Gandomi, "Partial reinforcement optimizer: An evolutionary optimization algorithm", *Expert Systems with Applications,* vol. 238, pp. 122070, 2024.

[2] N. Singh, X. Cao, S. Diggavi, and T. Başar, "Decentralized multi-task stochastic optimization with compressed communications," *Automatica,* vol. 159, pp. 111363, 2024.

[3] W. G. Alshanti, I. M. Batiha, M. m. A. Hammad, and R. Khalil, "A novel analytical approach for solving partial differential equations via a tensor product theory of Banach spaces", *Partial Differential Equations in Applied Mathematics,* vol. 8, pp. 100531, 2023.

[4] A. A. Al-Nana, I. M. Batiha, and S. Momani, "A numerical approach for dealing with fractional boundary value problems", *Mathematics,* Vol. 11, No. 19, pp. 4082, 2023.

[5] G. Kumar, R. Saha, M. Conti, T. Devgun, and R. Thomas, "GREPHRO: Nature-inspired optimization duo for Internet-of-Things", *Internet of Things*, pp. 101067, 2024.

[6] 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, pp. 10312, 2023.

[7] M. Qaraad, S. Amjad, N. K. Hussein, M. Farag, S. Mirjalili, and M. A. Elhosseini, "Quadratic interpolation and a new local search approach to improve particle swarm optimization: Solar photovoltaic parameter estimation", *Expert Systems with Applications,* Vol. 236, pp. 121417, 2024.

[8] 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.

[9] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents", *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics),* Vol. 26, No. 1, pp. 29-41, 1996.

[10] J. Kennedy, and R. Eberhart, "Particle swarm optimization", In: *Proc. of ICNN'95-international conference on neural networks,* vol. 4, pp. 1942-1948, 1995.

[11] D. Karaboga, and B. Basturk, "Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems", In: *Proc. of International fuzzy systems association world congress*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 789-7982007.

[12] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation", *International Journal of Bio-inspired Computation,* Vol. 2, No. 2, pp. 78-84, 2010.

[13] A. Majumder, "Termite alate optimization algorithm: a swarm-based nature inspired algorithm for optimization problems", *Evolutionary Intelligence,* Vol. 16, No. 3, pp. 997-1017, 2023.

[14] L. Abualigah, M. Abd 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, pp. 116158, 2022.

[15] 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, pp. 103541, 2020.

[16] W. Zhao, L. Wang, Z. Zhang, H. Fan, J. Zhang, S. Mirjalili, N. Khodadadi, and Q. Cao, "Electric eel foraging optimization: A new bio-inspired optimizer for engineering applications", *Expert Systems with Applications*, Vol. 238, pp. 122200, 2024.
[17] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer", *Advances in Engineering Software*, Vol. 69, pp. 46-61, 2014.
[18] 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, pp. 107408, 2021.
[19] S. Mirjalili, and A. Lewis, "The whale optimization algorithm", *Advances in Engineering Software*, Vol. 95, pp. 51-67, 2016.
[20] M. Braik, A. Hammouri, J. Atwan, M. A. Al-Betar, and M. A. Awadallah, "White Shark Optimizer: A novel bio-inspired meta-heuristic algorithm for global optimization problems", *Knowledge-Based Systems*, pp. 108457, 2022.
[21] D. E. Goldberg, and J. H. Holland, "Genetic Algorithms and Machine Learning," *Machine Learning*, Vol. 3, No. 2, pp. 95-99, 1988.
[22] 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.
[23] L. N. De Castro, and J. I. Timmis, "Artificial immune systems as a novel soft computing paradigm", *Soft computing*, Vol. 7, No. 8, pp. 526-544, 2003.
[24] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing", *Science*, Vol. 220, No. 4598, pp. 671-680, 1983.
[25] M. Dehghani, Z. Montazeri, G. Dhiman, O. Malik, R. Morales-Menendez, R. A. Ramirez-Mendoza, A. Dehghani, J. M. Guerrero, and L. Parra-Arroyo, "A spring search algorithm applied to engineering optimization problems", *Applied Sciences*, Vol. 10, No. 18, pp. 6173, 2020.
[26] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm", *Information Sciences*, Vol. 179, No. 13, pp. 2232-2248, 2009.
[27] M. Dehghani and H. Samet, "Momentum search algorithm: A new meta-heuristic optimization algorithm inspired by momentum conservation law", *SN Applied Sciences*, Vol. 2, No. 10, pp. 1-15, 2020.
[28] R. V. Rao, V. J. Savsani, and D. 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.
[29] P. Trojovský, and M. Dehghani, "A new optimization algorithm based on mimicking the voting process for leader selection", *PeerJ Computer Science*, Vol. 8, pp. e976, 2022.
[30] M. Dehghani, and P. Trojovský, "Teamwork Optimization Algorithm: A New Optimization Approach for Function Minimization/Maximization", *Sensors*, Vol. 21, No. 13, pp. 4567, 2021.
[31] F. A. Zeidabadi, M. Dehghani, and O. P. Malik, "RSLBO: Random Selected Leader Based Optimizer", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 5, pp. 529-538, 2022, doi: 10.22266/ijies2021.1031.46.
[32] M. Dehghani, Z. Montazeri, A. Dehghani, R. A. Ramirez-Mendoza, H. Samet, J. M. Guerrero, and G. Dhiman, "MLO: Multi leader optimizer", *International Journal of Intelligent Engineering and Systems*, Vol. 13, pp. 364-373, 2020, doi: 10.22266/ijies2020.1231.32.
[33] F. A. Zeidabadi, S. A. Doumari, M. Dehghani, and O. P. Malik, "MLBO: Mixed Leader Based Optimizer for Solving Optimization Problems", *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 4, pp. 472-479, 2021, doi: 10.22266/ijies2021.0831.41.
[34] M. Dehghani, Z. Montazeri, H. Givi, J. M. Guerrero, and G. Dhiman, "Darts game optimizer: A new optimization technique based on darts game", *International Journal of Intelligent Engineering and Systems*, Vol. 13, pp. 286-294, 2020, doi: 10.22266/ijies2020.1031.26.
[35] Z. Montazeri, T. Niknam, J. Aghaei, O. P. Malik, M. Dehghani, and G. Dhiman, "Golf Optimization Algorithm: A New Game-Based Metaheuristic Algorithm and Its Application to Energy Commitment Problem Considering Resilience", *Biomimetics*, 2023.
[36] M. Dehghani, Z. Montazeri, S. Saremi, A. Dehghani, O. P. Malik, K. Al-Haddad, and J. M. Guerrero, "HOGO: Hide objects game optimization", *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 10, 2020, doi: 10.22266/ijies2020.0831.19.
[37] M. Dehghani, M. Mardaneh, J. M. Guerrero, O. Malik, and V. Kumar, "Football game based optimization: An application to solve energy commitment problem", *International Journal of Intelligent Engineering and Systems*, Vol. 13, pp. 514-523, 2020, doi: 10.22266/ijies2020.1031.45.

[38] R. Moghdani, and K. Salimifard, “Volleyball premier league algorithm”, *Applied Soft Computing,* Vol. 64, pp. 161-185, 2018.
[39] M. Dehghani, Z. Montazeri, O. P. Malik, H. Givi, and J. M. Guerrero, “Shell game optimization: A novel game-based algorithm”, *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 3, pp. 246-255, 2020, doi: 10.22266/ijies2020.0630.23.
[40] M. Dehghani, Z. Montazeri, O. P. Malik, A. Ehsanifar, and A. Dehghani, “OSA: Orientation search algorithm”, *International Journal of Industrial Electronics, Control and Optimization*, Vol. 2, No. 2, pp. 99-112, 2019.
[41] F. A. Zeidabadi, and M. Dehghani, “POA: Puzzle Optimization Algorithm”, *International Journal of Intelligent Engineering and Systems*, Vol. 15, No. 1, pp. 273-281, 2022, doi: 10.22266/ijies2022.0228.25.
[42] S. A. Doumari, H. Givi, M. Dehghani, and O. P. Malik, “Ring Toss Game-Based Optimization Algorithm for Solving Various Optimization Problems”, *International Journal of Intelligent Engineering and Systems*, Vol. 14, No. 3, pp. 545-554, 2021, doi: 10.22266/ijies2021.0630.46.
[43] M. Dehghani, Z. Montazeri, and O. P. Malik, “DGO: Dice game optimizer”, *Gazi University Journal of Science,* Vol. 32, No. 3, pp. 871-882, 2019.
[44] P. R. Krausman, and A. L. Casey, “Addax nasomaculatus”, *Mammalian Species*, No. 807, pp. 1-4, 2007.
[45] M. Burton, and R. Burton, *The Funk & Wagnalls Wildlife Encyclopedia: Vol 1 AAR-BAD. Funk and Wagnalls, Incorporated, 1969.*
[46] J. Hummel, P. Steuer, K.-H. Südekum, S. Hammer, C. Hammer, W. J. Streich, and M. Clauss, “Fluid and particle retention in the digestive tract of the addax antelope (Addax nasomaculatus)—adaptations of a grazing desert ruminant”, *Comparative Biochemistry and Physiology Part A: Molecular & Integrative Physiology*, Vol. 149, No. 2, pp. 142-149, 2008.
[47] E. M. Spevak, E. S. Blumer, and T. L. Correll, “Species survival plan contributions to research and reintroduction of addax Addax nasomaculatus”, *International Zoo Yearbook,* Vol. 32, No. 1, pp. 91-98, 1993.
[48] 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.
[49] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, “Marine Predators Algorithm: A nature-inspired metaheuristic”, *Expert Systems with Applications,* Vol. 152, pp. 113377, 2020.
[50] 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.
[51] A. H. Gandomi, and X.-S. Yang, "Benchmark problems in structural optimization", *Computational Optimization, Methods and Algorithms*, pp. 259-281, 2011.