Archives of Computational Methods in Engineering
https://doi.org/10.1007/s11831-023-09945-6

SURVEY ARTICLE

A Comprehensive Survey on Aquila Optimizer

Buddhadev Sasmal[1] · Abdelazim G. Hussien[2,3,4] · Arunita Das[1] · Krishna Gopal Dhal[1]

Received: 12 April 2023 / Accepted: 17 May 2023

Abstract

Aquila Optimizer (AO) is a well-known nature-inspired optimization algorithm (NIOA) that was created in 2021 based on the prey grabbing behavior of Aquila. AO is a population-based NIOA that has demonstrated its effectiveness in the field of complex and nonlinear optimization in a short period of time. As a result, the purpose of this study is to provide an updated survey on the topic. This survey accurately reports on the designed enhanced AO variations and their applications. In order to properly assess AO, a rigorous comparison between AO and its peer NIOAs is conducted over mathematical benchmark functions. The experimental results show the AO provides competitive outcomes.

1 Introduction

Finding the finest values for a system's particular parameters in order to satisfy the system design efficiently is known as an optimization procedure [1]. Practical applications and challenges in artificial intelligence (AI) are often discontinuous or unrestricted in nature, which is why these issues are seen as complicated and challenging engineering problems. Conventional approaches to optimization, such as Newton's method and gradient descent, suffer from rigid constraints on the state space and poor searching efficacy, which make them ineffective for addressing complex optimization issues [2]. Furthermore, they have a limited range of applications and mostly concentrate on tackling a few set problems. So, a key area of research right now is how to develop better optimization techniques.

Recent times, Nature-Inspired Optimization algorithms (NIOAs) [3–5] have become increasingly important in a variety of domains, including computational intelligence (CI), AI, machine learning (ML), and others. As a result, NIOAs are extensively used to solve issues in the real world, and have shown their effectiveness by offering positive solutions to these issues. Recent times, NIOAs have become increasingly important in a variety of fields, including computational intelligence (CI), AI, machine learning (ML), and others. As a result, NIOAs are widely employed to address issues in the real world, and have shown their effectiveness by offering positive solutions to these issues. But there is no guarantee that the best options will be considered. The goal is to create an effective NIOA that is practical, works most of the time, and can generate high-caliber results. Some of the highest quality solutions are likely to be close to being optimal; however, there is again no assurance of this. Each NIOA has two phases, namely (i) exploration or global search or diversification, and (ii) exploitation or local search or intensification [6]. The NIOA explores the search space regions during the exploration. In exploitation, the optimization approach looks across the search space for the finest solution. To prevent being trapped at local optima, an effective NIOA must have balanced exploration and exploitation. The No Free Lunch (NFL) theorem states that an algorithm's performance on one problem category does not assurance its performance on other categories [7]. Hence, the efficacy and supremacy of one NIOA over others is problem dependent. Nonetheless, it is evident that the efficiency of any form of algorithm, including NIOA, is profoundly impacted by the algorithm's design viewpoint such as

✉ Krishna Gopal Dhal
krishnagopal.dhal@midnaporecollege.ac.in

Buddhadev Sasmal
buddhadev.sasmal@midnaporecollege.ac.in

Abdelazim G. Hussien
aga08@fayoum.edu.eg

Arunita Das
arunita.das@midnaporecollege.ac.in

1 Department of Computer Science and Application, Midnapore College (Autonomous), Paschim Medinipur, West Bengal, India

2 Department of Computer and Information Science, Linköping University, 58183 Linköping, Sweden

3 Faculty of Science, Fayoum University, Fayoum, Egypt

4 MEU Research Unit, Middle East University, Amman, Jordan

Published online: 07 June 2023

optimal mixing of exploration and exploitation. Beside that current research describes a large number of NIOAs that have been developed based on physical, social, and biological factors in addition to natural laws up to the year 2023 and they have their usefulness and also limitations.

An NIOA named Aquila Optimizer (AO) [1], which is modeled after the natural behaviors of Aquila when catching their prey, was proposed by Abualigah in 2021. Since the AO was first introduced, it has drawn a large number of academics to undertake in-depth investigation on it in a variety of areas, and in the last 2 years, many scholarly outcomes have been attained. Google Scholar data dated 03.04.2023 shows that AO has received 884 citations overall. With an emphasis on research from 2021 to the present, this article seeks to offer a comprehensive up-to-date assessment of AO, its variants, and applications in several domains.

To the best of the knowledge, no detailed review or survey paper on AO has been published to date, and this is the primary motivation for this effort. As a result, the purpose of this study is to assemble and analyze existing research on AO. This study is significant since it identifies, classifies, and examines the AO and its enhanced variations, which are employed to address various optimization issues. This research thoroughly examines the application of AO in many real-world scenarios and gives recommendations for additional research lastly.

The remaining sections of this research are laid out as follows. The research methodology and AO survey taxonomy are shown in Sect. 2. A basic mathematical explanation of AO, its complexity, and parameters are provided in Sect. 3. The improved AO versions are discussed and analyzed in Sect. 4. Section 5 discusses the AO's applications. AO evaluation has been completed in Sect. 6. Section 7 serves as both the paper's conclusion and a suggestion for potential future directions.

2 Research Methodology and Survey Taxonomy

The research papers are collected using the standard search phrases such as metaheuristics, swarm intelligence, binary variants, multi-objective, optimization, etc. combined with Aquila Optimizer (also with the abbreviation AO) and Aquila optimization algorithm (also with the abbreviation AOA). The search phrases are used to perform a systematic search in the Google Scholar, IEEE Xplore, ScienceDirect, SpringerLink, ACM Digital Library, etc. There are 112 papers are collected through this procedure after removing the duplicate papers. Figure 1 shows the selection process for research papers.

Finally, fresh taxonomies for the study are also recommended based on the collected publications. Based on whether or not the papers are "improved," "binary," or "multi-objective," papers are divided into various classes. The articles are then analyzed and summarized from the perspective of the application fields. Figure 2 shows the number of AO improved variations issued by various publishers according to the survey. It is clearly noticed that "MDPI" is the top most publisher for publishing the AO based articles.

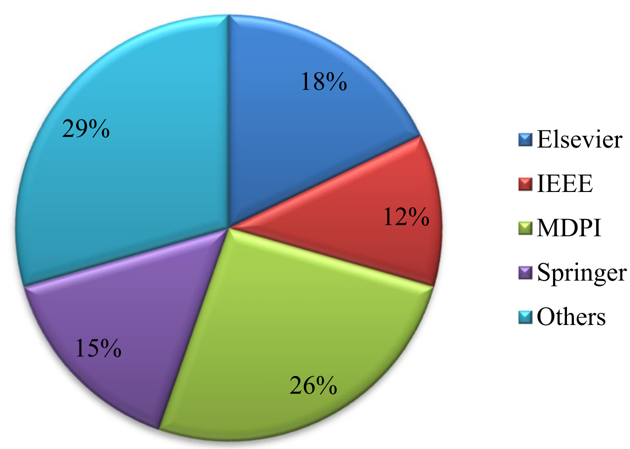


Fig. 2 Number of AO based research papers published by different publishers

Fig. 1 Diagrammatic representation of the paper collection process

After that "Elsevier", "Springer", and "IEEE" are also published significant number of articles. Figure 3 shows the annual number of articles on AO that are published. Figure 3 shows that AO has been successful in attracting researchers since it was founded in 2021. For that, a wonderful number of citations (884 on the date 03.04.2023) have been given.

3 Overview of Aquila Optimizer (AO)

Aquila Optimizer (AO), a population based meta-heuristic, has been proposed by Abualigah et al. [1] in the year 2021. Aquila, the Latin name for the eagle (bird of prey) regarded as one of the smart and skilled hunters with strapping feet and hefty sharpen talons that employs its swiftness, agility to grab hold of its prey. There are four primary hunting strategies used by Aquila, including high soar with a vertical stoop, contour flight with a short glide, low fly with a gradual descent, and walking and grabbing prey. Inspired by these four fundamental hunt mechanisms of Aquila, a nature inspired optimization algorithm is fabricated upon known as Aquila Optimizer (AO) that essentially elucidates the action of every stage of the hunt. The classical Aquila Optimizer (AO) principally focuses on five significant steps namely Initialization, Expanded Exploration, Narrowed Exploration, Expanded Exploitation and Narrowed Exploitation.

The steering of AO algorithm from exploration to exploitation step typically transpires in view of one of the most prominent provisions of the algorithm, i.e., **current iteration** $\leq (2/3)$ **maximum iteration**. If the referred to condition is true exploration step will be keyed up else exploitation step gets executed.

3.1 Step 1: Initialization

In this first phase, the population of solutions has been generated at random and AO's other parameters are initialized.

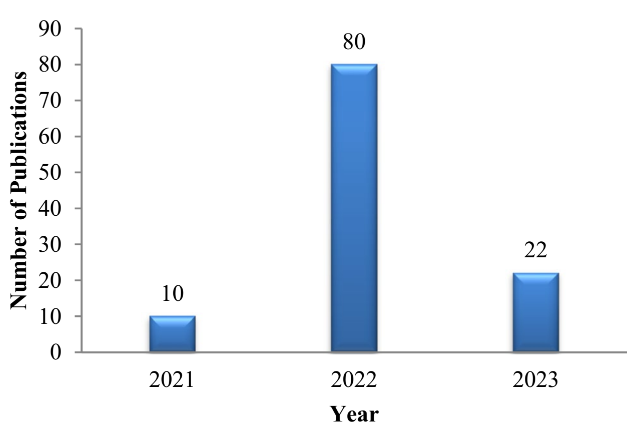


Fig. 3 Year-wise research articles publication

3.2 Step 2: Expanded Exploration

Expanded Exploration basically bring to light the primary hunting technique practiced by Aquila that is the high soar with vertical stoop in which the algorithms aim to find out the search apace from the high ascend and the same is mathematically embodied using Eq. (1).

$$X_1(t+1) = X_{best}(t) \times \left(1 - \frac{t}{T}\right) + \left(X_M(t) - X_{best}(t) * rand\right) \tag{1}$$

where, $X_1(t+1)$ is the solution of the next iteration of t, and it is created by the first search technique X_1. $X_{best}(t)$ is the finest solution found so far. $(1 - 1/T)$ controls the exploration. T and t represent maximum number of iteration and current iteration. X_M (t) is the location mean value of current solution. rand value lies between 0 to 1.X_M is calculated as follows:

$$X_M(t) = \frac{1}{N}\sum_{i=1}^{N} X_i(t), \forall_j = 1, 2, \ldots Dim \tag{2}$$

Here *Dim* is the dimension and *N* is the population size.

3.3 Step 3: Narrowed Exploration

In the third phase, i.e., **Narrowed Exploration,** the second hunting technique carried out by Aquila that is the contour flight with short glide is projected wherein Aquila arranges the land thereby circling around the target prey to attack. Such behavior of Aquila is precisely brought to light using Eq. (3).

$$X_2(t+1) = X_{best}(t) \times Levy(D) + \left(X_R(t) + (y - x) * rand\right) \tag{3}$$

$X_2(t+1)$ denotes solution of the next iteration of t, $X_{best}(t)$ is the best solution, $X_R(t)$ represents random solution belonging to $[1, N]$, *rand* denotes random number belonging to $[0, 1]$, x and y presents the spiral shape in the search and is denoted using Eq. (4).

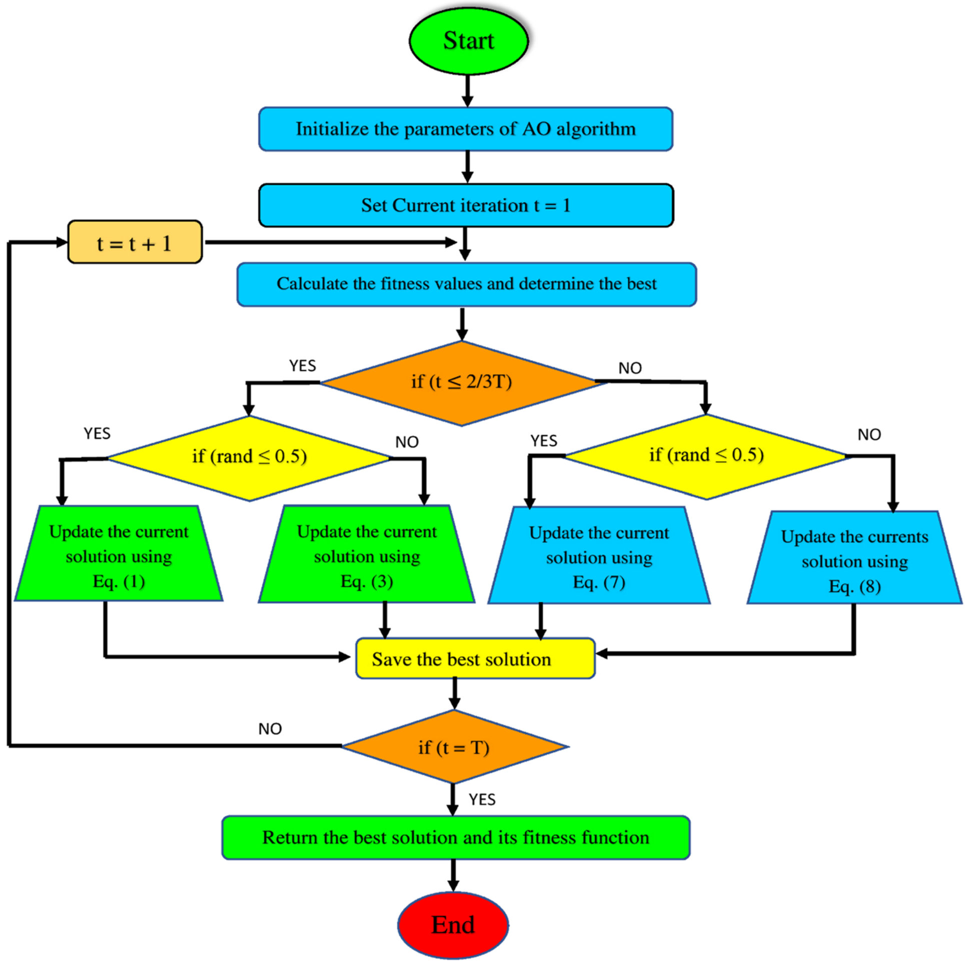
$$x = r \times sin(\theta)$$

$$y = r \times cos(\theta) \tag{4}$$

Further, in the same context, Lévy flight distribution function, Levy (D) used in Eq. (3) is highlighted in Eq. (5) and σ is calculated using Eq. (6) respectively.

$$Levy(D) = s \times \frac{u \times \sigma}{|v|^{\frac{1}{\beta}}} \tag{5}$$

Fig. 4 Flowchart depicting the mechanism of AO



$$\sigma = \frac{\tau(1+\beta) \times sine\left(\frac{\beta \Pi}{2}\right)}{\tau\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}} \tag{6}$$

where, Levy (D) denotes Lévy flight distribution function at dimension space D, s is the constant value assigned as 0.1, u, v are random number within [0,1], β is the constant value [0.5], σ represents one of the parameters in Lévy flight distribution function,

3.4 Step 4: Expanded Exploitation

The fourth phase named as **Expanded Exploitation,** the idea that lies in third hunting mechanism is portrayed basically the low flight with a slow descent attack wherein Aquila gradually plunge in the targeted space and move closer to the prey to attack. This action of Aquila is thus revealed accurately using Eq. (7).

$$\begin{aligned} X_3(t+1) = & \left(X_{best}(t) \times (X_M(t)\right) \times \ \alpha - rand \\ & + ((UB - LB) \times rand + LB) \times \delta \end{aligned} \tag{7}$$

where,T signifies the maximum number of iterations. UB and LB represents Upper and Lower Bound respectively, α and δ denotes fixed exploitation adjustment parameters.

3.5 Step 5: Narrowed Exploitation

Finally in the fifth and the final step, the idea is taken from last attacking method of Aquila widely known as walk and grab attack to frame the last step known as **Narrowed Exploitation** and the same is mathematically revealed using Eq. (8).

$$\begin{aligned} X_4(t+1) = \ & QF \times X_{best}(t) - (G_1 \times X(t) \times rand) \\ & - (G_2 \times Levy(D) + rand \times G_1 \end{aligned} \tag{8}$$

Additionally, function utilized to balance the search method known as QF(t) (Quality Function), various motions

G_1 and flight slope G_2 are depicted using Eqs. (9, 10, and 11) respectively.

$$QF(t) = t^{\frac{2\times rand-1}{(1-T)2}} \tag{9}$$

$$G_1 = 2 \times rand - 1 \tag{10}$$

$$G_2 = 2 \times \left(1 - \frac{t}{T}\right) \tag{11}$$

Based on the above discussion, the procedure of the AO is presented in Algorithm 1 and the flowchart of the AO has been delineated in Fig. 4.

Algorithm 1: Algorithm of Aquila Optimizer.

Step 1: **Population Initialization Phase**
Initialize population randomly and set other parameters.
while (the termination criteria do not met) ***do***
  Calculate the fitness of the solutions
  Find out X_{best} as per fitness function
  for $i = 1 : N$ ***do***
    Update $X_M(t)$
    Update parameters $Levy(D)$, G_1 and G_2
    Update y, x as per **Eq. (4)**
    if $t \le \left(\frac{2}{3}\right) * T$ ***then***
      if $\boldsymbol{rand} \le \mathbf{0.5}$ ***then***
Step 2: **X1 : Expanded Exploration**
        Update the present solution as per **Eq. (1)**
        if $Fitness\ (X_1(t+1)) < Fitness\ (X(t))$ ***then***
          $X(t) = (X_1(t+1))$
          if $Fitness\ (X_1(t+1)) < Fitness\ (X_{best}(t))$ ***then***
            $X_{best}(t) = (X_1(t+1))$
          end ***if***
        end ***if***
      else *Step 3*
Step 3: **X2 : Narrowed Exploration**
        Update the present solution as per **Eq. (3)**
        if $Fitness\ (X_2(t+1)) < Fitness\ (X(t))$ ***then***
          $X(t) = (X_2(t+1))$
          if $Fitness\ (X_2(t+1)) < Fitness\ (X_{best}(t))$ ***then***
            $X_{best}(t) = (X_2(t+1))$
          end ***if***
        end ***if***
      end ***if***
    else
      if $\boldsymbol{rand} \le \mathbf{0.5}$ ***then***
Step 4: **X3 : Expanded Exploitation**
        Update the present solution as per **Eq. (7)**
        if $Fitness\ (X_3(t+1)) < Fitness\ (X(t))$ ***then***
          $X(t) = (X_3(t+1))$
          if $Fitness\ (X_3(t+1)) < Fitness\ (X_{best}(t))$ ***then***
            $X_{best}(t) = (X_3(t+1))$
          end ***if***
        end ***if***
      else *Step 5*
Step5: **X4 : Narrowed Exploitation**
        Update the present solution as per **Eq. (8)**
        if $Fitness\ (X_4(t+1)) < Fitness\ (X(t))$ ***then***
          $X(t) = (X_4(t+1))$
          if $Fitness\ (X_4(t+1)) < Fitness\ (X_{best}(t))$ ***then***
            $X_{best}(t) = (X_4(t+1))$
          end ***if***
        end ***if***
      end ***if***
    end ***if***
  end ***for***
end ***while***
Return X_{best}

3.6 Computational Complexity

The measurement of computational complexity of the AO crucially determined by some factors, namely size of the population (n), initializing the individuals, computing the fitness, and upgrading the individuals, number of iterations (T), problem dimension (D). The complexity for initialize the population is O (n). We do not consider the fitness function's complexity in this case because it depends on the problem. The complexity of upgrading the individuals is $O(T \times n) + O(T \times n \times D)$. Thus, the total time complexity of AO is $O(n \times (T \times (D + 1)))$. Determining the computational complexity of such stochastic strategies, unfortunately, is a highly challenging process because NIOAs' searching process is unique that fails to ensure the finding of the global optimum within a specified time frame. As a result, it has been noted in the literature that temporal complexity may evaluate NIOAs using theoretical CPU time [6].

3.7 Parameter Sensitivity

The main two parameters of AO are α and δ on which the exploitation capability of the AO depends. In the classical AO, experiment has been conducted over mathematical benchmark functions to find the optimal values for α and δ. Experimental results show that AO produces best results in maximum times when α and δ are set to small value 0.1. Nonetheless, it is stated in [8] that AO has significant exploration ability but suffering from exploitation capability due to this fixed values initialization to α and δ. The exploration ability of the AO significantly depends on the parameters of Lévy distribution.

Furthermore, authors measured the influence of population size over AO's performance. The size and the individuals of initial population have a significant influence over the performance of AO. Literature has demonstrated that a constant population size for all issues across multiple dimensions cannot be scientific, it is known as the "curse of population size" [9, 10]. According to the literature, one type of adaptation mechanism to include well-balanced exploration and exploitation into any NIOA is to change the population size. In the source paper, authors tested several population sizes, i.e., 10, 20, 30, 40, and 50 over mathematical benchmark functions and claims that AO is more robust and less overwhelmed by the population size, i.e., the change in the population size does not significantly affect optimal solution finding ability of the ability AO.

4 Related Work on Classical AO and Its Improved Variants

Aquila Optimizer is a novel NIOA that is designed to resolve complex optimization problems. The algorithm aims to find the ideal solution to an optimization problem by mimicking the hunting behavior of the Aquila bird. It has been

Table 1 Classical AO and its improvement strategies

Sl. no	Utilized Strategies	References
1	Classical AO	[11–59]
2	Hybridization (By integrating entire NIOA, NIOA's operators)	[91, 103–131]
3	Opposition based Learning (OBL)	[61–70]
4	Chaotic Sequence	[71, 72]
5	Levy flight-based strategy	[73, 74]
6	Gauss map and crisscross operator	[75]
7	Niche Thought with Dispersed Chaotic Swarm	[76]
8	Random learning mechanism and Nelder-Mead simplex search	[77]
9	Velocity-aided global search mechanism and adaptive OBL	[78]
10	Chaotic sequence, Binary Tournament selection, Roulette wheel selection, Shuffle crossing over, and displacement mutation-based population	[79]
11	Wavelet Mutation	[80]
12	Probabilistic Perturbation strategy and Cauchy operator	[81]
13	Normal distribution-based randomization and Weibull function	[82]
14	Weighted adaptive searching technique	[83]
15	Inertia Weight, Vertical and horizontal crossing strategy, and Cauchy Elite Mutation	[84]
16	Wavelet Mutation operator, Elite opposition based learning and Exploitative manipulation equation	[85]
17	Niche Thought	[86]
18	Heterogeneous strategy	[87]
19	Binary AO	[88–90]
20	Multi-objective AO	[81, 90–93]

demonstrated that the AO is effective in handling a wide variety of optimization issues, including unconstrained and constrained issues, single-objective and multi-objective issues, continuous and discrete issues. Due to its extreme supremacy and efficacy the classical Aquila Optimizer has been utilized in various fields of study, including engineering, finance, image and signal processing, machine learning, renewable energy, and transportation. A brief review on classical AO is shown in Sect. 4.1. Despite its promising results in resolving various optimization problems, AO has some limitations. One of the main limitations is premature convergence, which happens when the algorithm becomes stuck in a local optimum and fails to explore the search space effectively. This can lead to suboptimal solutions and limit the algorithm's capability to find the global optimum. Another limitation is slow convergence, especially for high-dimensional and complex optimization problems. AO could require more time to converge than other optimization algorithms to get the optimal situation. Additionally, the scalability of AO is limited, and it may struggle to solve large-scale optimization problems with a high number of variables and constraints. To address these limitations, researchers have proposed various improved variants of the AO over several optimization fields. These improvements include hybridizing the algorithm with other optimization techniques, introducing new movement strategies, and modifying the parameters of the algorithm. The improved variants of AO are more capable of dealing with a extensive range of complex real-world optimization problems than traditional AO. The classical AO and its improvement methodologies are specified in Table 1. Research articles based on the methodologies are deliberated as follows.

4.1 Classical AO

Gul et al. [11] suggested an AO algorithm-based approach, specifically designed for multi-robot space exploration. The suggested method utilized a novel parallel communication protocol, that facilitates the exploration of space by multiple robots while reducing the computational burden on the system. CME, a deterministic method, was first used to calculate the cost and usefulness of neighboring cells to the robot. The AO method was then applied to improve the precision of the final solution. The result of this study showed that in comparison with CME, the proposed CME-AO improved map exploration in a significantly shorter amount of time with nearly no fail runs.

For early heart disease prediction, Barfungpa et al. [12] developed an intelligent system using hybrid deep dense Aquila network. The primary goal of the suggested architecture is to offer a deep learning (DL) model that is coupled with cutting-edge data mining techniques for formulating sensible decisions and precise disease prediction. The weight of the disease classification model was updated using the AO method. The efficacy of the suggested method is assessed against recently released algorithms. The suggested method has the highest accuracy of all existing methods, up to 99.57%.

An AO-based cooperative path planning method for numerous UAVs was presented by Huang et al. [13] to fulfil the autonomy needs of the Close Air Support (CAS) mission. The cooperative route planning problem is transmuted into an optimization problem that can be utilized to solve the multi-UAV trajectory using the properties of multiple iterative AO approaches. produced UAV trajectory avoids opponent and terrain risks, keeps team members at a safe area from one another, and is crash-proof, all while meeting the stringent stability, security, and environmental versatility standards for UAVs used in CAS missions.

Hemavathi et al. [14] present a new framework for improving QoS in WSN-IoT networks. The AO algorithm was used to pick a cluster head that works well and is reliable. It does this by considering things like the level of the node, its distance from the sink node, its energy level, and so on. By making the QoS-aware relay node the cluster head, the AO algorithm makes the WSN network last longer. Various metrics were used in the simulations and the outcome of this study demonstrated that the proposed method improve the QoS of the WSN-IoT networks comparison to the state-of-the-art methods.

In WSN anchor nodes play a vital role because it helps to determine optimum positioning of other nodes. To achieve suitable coordinate points of the nodes in the network Agarwal et al. [15] suggested a novel IAOAB-NLS algorithm for node localization (NL) in WSN. The suggested model is motivated by Aquila's behavior and has the capacity to achieve accurate coordinate points for the network's nodes. The findings of this research showed that the suggested technique outperformed the other examined algorithms.

Aribowo et al. [16] employed the AO algorithm to find the proportional integral derivative (PID) controller's parameters in order to manage a dc motor's speed. The suggested technique enhanced the PID's undershoot by 0.5% while averaging a 0.023% reduction in PID overshoot. The findings of this research indicate that the AO method for the DC motor speed regulation system produced more effectiveness in comparison to the SOA, MPA, GPC, and ChOA method respectively.

For the Intrusion detection systems (IDS) system, Fatani et al. [17] created novel feature extraction and selection techniques by utilising the benefits of the AO algorithms. An ideal feature subset that accurately reflects the properties of the datasets is chosen using the AO technique. Four well-known public datasets, were utilized to evaluate the effectiveness of the developed IDS technique.

Using a variety of evaluation metrics, including precision, recall, and F1-Measure, the research findings demonstrated that the suggested method outperforms GWO, FFA, MVO, WOA, BAT, TSO, and MFO.

In their study, Mehmood et al. [18] employed the AO technique to estimate the control autoregressive (CAR) model's parameters. To boost the AO's efficiency, raising the population and the number of generations is done at the expense of the computation complexity. While $\beta = 0.1$ and $\mu = 0.1$ exploitation customizable parameters result in the best fitness for many generations and populations. Changing the level of noise reduces the AO's resilience and precision. In terms of performance metrics, the result of this study demonstrated that the presented technique performed well when compared to AOA, SCA, and RSA algorithms.

Hussain et al. [19] proposed a redesigned H-bridge inverter with seven levels, fewer components, and less overall harmonic distortion. Two DC sources and six IGBTs were utilized to provide a 7th level output voltage, and the AO was utilized to manage the output voltage with a functionality to remove the fifth and seventh harmonics. The research finding demonstrated that the AO provides superior results for a variety of modulation indices compared to a number of famous metaheuristic algorithms such as GA and DE.

Xing et al. [20] introduced a novel wind speed prediction system that integrates data pre-processing, benchmark model selection, an AO algorithm for point and interval forecast. This research shows that the weights assigned by this optimizer are Pareto optimum solutions. The outcomes of the study showed that the designed technique accomplished greater accuracy than the tested techniques in every case for point forecasting, and achieved high coverage and low width error in a forecasting interval, which is a vital directive for ensuring the safety and consistency of the power system.

To identify the optimal deep learning method for HSI image classification, Reddy et al. [21] developed a compressed synergic DCNN-AO (CSDCNN-AO) model by utilizing AO algorithm. This combo will decrease the wavelet feature's learning complexity and the AO's loss function. AO strategy can reduce the supreme amount of data characteristics without altering the data's characteristic state, while consuming less processing time as well as memory. The outcome of this study showed that the CSDCNN-AO technique produce finest result than the ALO, WOA, PSO, and GWO and has the greatest accuracy across the four datasets.

A novel TAAO-SDTIM model was presented by Abualkishik et al. [22] to achieve the highest levels of security and information supervision in WSN. The presented technique selects the best possible set of routes to base station using a fitness function with RE, DBS, and TL parameters. In order to choose highly secure nodes for the data transmission process, the trust level of the nodes is included in the route selection procedure. The experimental outcome of this work showed that, in comparison to the CHICD, FABC, and ABC models, the proposed technique achieved maximum PDR of 95.80%.

For the evaluation of wind energy potential using Weibull parameter estimation, El-Ela et al. [23] employed the AO algorithm. The AO achieved the lowest standard deviation and error, making it the most reliable technique. Subsequently it delivers optimum solution through the calculation of the first and second site parameters, respectively, the AO method exhibits the fastest convergence when compared to the others. The unique meta-heuristic AO approach performs best at obtaining the highest R^2, lowest RMSE, and lowest MAE values for adjusting Weibull parameters. The experimental findings showed that AO performs more accurately and consistently when calculating wind energy and Weibull parameters.

A high-speed electric spindle thermal error modelling based on AO and least squares SVM (LSSVM) was proposed by Li et al. [24]. The proposed approach resolves the issue of prediction error brought on by essential LSSVM model parameters that were provided subjectively. The average goodness of fit rises by 0.157, the average absolute error decreases by 2.1985 and the average root mean square error rises by 2.7025 as a result, improving accuracy rate by 7.25 percent on average. This boosts the LSSVM model's reliability and accuracy rate. The AO-LSSVM model is therefore better suited for predicting thermal errors in high-speed motorized spindles compared to PSO-LSSVM.

Rajinikanth et al. [25] proposed a study that intends to enforce a joint thresholding and segmentation model to effectively extricate the Gastric-Polyp (GP). They utilized AO algorithm that reinforced tri-level thresholding with entropy and a between-class-variance method for enhancing the GP region. The experimental evaluation was conducted utilizing four benchmark EI datasets, and the similarity metrics, including accuracy, precision, sensitivity, Jaccard, Dice, and specificity, were calculated to validate the medical relevance. The results showed that employing the AO algorithm greatly improved segmentation performance and archived greater similarity metrics than the other tested algorithms.

To increase the energy balance in clusters between sensor nodes during network communications and to decrease power consumption, Taha et al. [26] offer a novel enhancement technique based on AO. The AO was utilized to identify the best cluster heads and make sure the network is clustering effectively and steadily, which reduces energy use and lengthens the lifespan of the network. AO implementation was two-phased. The AO simulation code was conducted using an ideal sensor distribution to select clustering heads in the first phase. In the second phase, the AO was utilized to enhance method efficiency and network lifespan with low energy consumption. The suggested AO was contrasted with

HHO, LEACH, COY, and GA in order to assess its performance. The results of the experiment showed that the suggested AO outperforms other methods.

Aarthi et al. [27] employed AO, which enhances shaming classification depending on category. This was done in order to enhance the precision of classification. The experimental study was conducted by establishing a number of effectiveness measurements and thereafter contrasting those indicators to comparable works that are already in existence. The results of the performance evaluation indicated that the proposed procedure was far more advantageous than any of the alternatives considered.

In order to track a battery's global maximum power point (GMPPT) under partial shade conditions, Karmouni et al. [28] developed a new MPPT algorithm based on AO that is specific to photovoltaic (PV) energy storage conversion systems. In order to evaluate the effectiveness of the presented MPPT control, numerous experimentations were carried out. The findings of the study demonstrated that the suggested AO had the maximum accuracy, speed, and stability under PSC conditions.

To achieve a more uniform distribution of cluster heads, Hosseinzadeh et al. [29] presented a clustering method utilizing swarm intelligence. The HAOFA technique is presented here for IoT clustering and routing operations by the combination of AO and FA. The AO algorithm is used in the disclosed technique to select and cluster CH nodes. FA then chooses the best possible route. The efficiency of this clustering and routing approach was assessed using the number of hops, remaining energy, mean distances, and node balance. This study's experimental results revealed that the proposed technique improved system energy usage and packet delivery ratio.

Jnr et al. [30] proposed a new hybrid method for predicting wind speed. The hybrid model was created by combining DWT, PSR, AO, and BPNN. The initial decon of the actual wind speed time series data was performed using the DWT. After that, PSR was used to retrieve the input–output vectors for the BPNN from the decomposed time series data. Finally, AO was utilized to fine-tune the implemented BPNN in order to estimate the wind speed. The research findings showed that the suggested method outperformed four other researched hybrid models in terms of various statistical parameters, including MAE, RMSE, MAE, and ELG.

An all-electric car with a hybrid energy system is presented by Narasimhulu et al. [31]. Topologies for combining the energy sources (solar panel, battery, and UC) are analyzed alongside the control method. The suggested combined ANN-AOA technique minimizes high-speed dynamic battery charging and discharging currents while maximizing UC and minimizing the battery discharge current. In comparison to existing Modified Harmony Search and PID based on a GA, the proposed method obtains a greater speed of 91 km/h.

Li et al. [32] developed a method for performing feature selection (FS) from the CT image feature sets utilized for COVID-19 identification in order to increase detection accuracy and speed. AO, a population-based intelligent optimization technique, was employed FS in this study. This FS approach converts continuous values to binary using an S-shaped transfer function. When the modified solution performs poorly, a different mutation technique is suggested to improve convergence. When applied to the two publicly available datasets, the presented technique accomplished a prediction accuracy of 99.67 and 99.28%, respectively.

To avoid the difficulties associated with the calibration and solution processes of a nonlinear underwater structured light vision model, Wang et al. [33] devised a novel checkerboard usage calibration approach based on AO. The process of finding a calibration solution is viewed as a nonlinear optimization problem, with the distance among the feature points interconnected by the checkerboard and the laser serving as the optimization objective. Finally, the proposed method's efficiency is confirmed through experimental calibration and measurements.

To reduce the amount of time needed to categories an ideal output, Vinayaki et al. [34] presented a four-stage method including pre-processing, segmentation, feature extraction, and classification. In this study, retinal images were classified using AO powered by a deep neural network. Furthermore, 516 image sets from the publicly available IDRiD dataset are used in this research to evaluate the efficacy of the suggested method. The experimental results showed that the proposed method has a high efficiency rate in compared to other tested methods.

Chen et al. [35] suggested two prediction models based on an AO algorithm and a back-propagation neural network model for estimating the equilibrium moisture level and precise gravity of thermally modified wood. The efficacy and accuracy of the suggested model were proved by comparison with two other models, as well as an ANN. According to the findings, the root mean square error value was cut by 87 and 97%, respectively, when the original back-propagation model was compared to the AO algorithm. R^2 values of 0.99 and 0.98 were obtained while attempting to determine the equilibrium moisture content and specific gravity, respectively.

A new AOTL-CDA3S method was developed by Duhayyim et al. [36] for environmentally friendly smart cities. The purpose of the described AOTL-CDA3S method was to categorize the various population densities existing in the smart cities' environment. The WAF method was initially employed in the AOTL-CDA3S approach to enhance the superiority of the input frames. After that, the AO was used in conjunction with the SqueezeNet model in the AOTL-CDA3S model to extract features. Finally, the XGBoost classification model is employed to categorize crowd densities. The efficiency of the presented method was assessed using a variety of metrics by testing it on benchmark crowd datasets. The findings of this research showed that presented method perform well compared to other examined methods.

Obayya et al. [37] presented a BUI-based AOBNN-BDNN approach for breast cancer detection and classification. WF-based noise removal and U-Net model are utilized in the pre-processing phases for BUI breast cancer detection and classification. SqueezeNet also generates feature vectors from pre-processed images. Next, BNN classifies input images. Finally, the AO approach fine-tunes BNN algorithm parameters to increase classification efficiency. The efficacy of the presented technique was assessed using the benchmark dataset and the findings of this research showed that the proposed approach outperformed recent methods with 99.72% accuracy.

To significantly guarantee a high-quality connection grid of a PV system, Guo et al. [38] suggested a unique PID parameter tuning technique of PLL with AO. The AO was utilized to decrease power fluctuations and enhance grid connection quality. The rapid and precise phase recognition and lock provided by a PLL regulation strategy helps to reduce power fluctuations and overshoot with minimal complexity and effort. The experimental findings demonstrated that the PV connected grid system achieved desirable outcomes using the PLL regulation approach based on the AO algorithm.

Aribowo et al. [39] developed a droop controlling mechanism based on the AO algorithm to preserve system efficiency and minimize overloads. The effectiveness of AO was evaluated by contrasting its transitory reaction to that of traditional approaches and PSO. The experimental outcomes showed that the AO approach performed well. The AO was able to shorten the settling time by 0.7% comparison to the traditional approach, and its final power is 0.798% greater than that of the traditional method.

A computer vision technique was presented by Nguyen et al. [40] to classify the severity of concrete spalls. Images are sorted into classes using a combination of DCNN and the extreme gradient boosting machine. In addition, the AO algorithm improves XGBoost's ability to make accurate predictions. The ARCS-LBP was used for feature extraction. Results of the experiment with a dataset of real-world concrete surface images and twenty independent model evaluations showed that the AO optimized XGBoost applied with ARCS-LBP has a classification accuracy rate of 99%.

Grace et al. [41] employed AO algorithm and hybrid LSTM-SVM model to predict android malware. AO used as a fitness evaluation based on the mean square error from the cross-validation procedure to choose the most appropriate features during the feature extraction phase. The retrieved features are fed into a LSTM-SVM model for classification. The efficiency of the suggested algorithm was tested using a various metrics, including Accuracy, Precision, Specificity, F1 score, and Recall. The results demonstrated that the presented model achieved greater performance metrics in comparison to the state-of-the-art technique.

Gul et al. [42] proposed a hybrid architecture for multi-robot space exploration by integrating the CME algorithm with the AO method. The architecture begins with a deterministic CME calculation of the cost and utility of the robot's adjacent cells. The AO method is then applied to improve the precision of the final solution. Many experiments were conducted under various scenarios to verify the efficacy of the suggested approach. The presented technique contrasted with traditional CME and CME-WO. The study's findings demonstrated that the proposed technique enhanced map exploration in noisy situations while drastically reducing execution time and computing complexity with almost no failed runs.

To speed up HSI classification, Reddy et al. [21] proposed CSDCNN-AO model. Both the wavelet concept's learning curve and the AO's loss function will be smoothed out by this combination. AO algorithms reduce the size of data sets while preserving their essential structure, while requiring less processing power and storage space. The presented method accomplished the maximum classification accuracy when comparison with other models. The experimental findings showed that the CSDCNN-AO model is the most accurate of the four used datasets. Nevertheless, the presented model does not perform good enough with certain samples.

Guan et al. [43] suggested an improved LOS guiding algorithm that is based on time-varying virtual guidance distance and AO. In order to finalize the solution of the control quantity, the AO was employed. The proposed method achieved superior performance when constructing the path following supervisor of a motorized buoy, through comparison simulation to solve the guidance angle.

An enhanced DV-Hop method based on hop distance correction and AO algorithm was presented by Yang et al. [44]. To increase the typical hop distance while decreasing the range error, a weighted correction factor was implemented. The AO algorithm was employed instead of the least-squares technique to determine the coordinates of the unidentified nodes. The experimental outcomes demonstrated that the positioning accuracy was increased by 42.6% using the proposed method as opposed to the conventional DV-Hop algorithm.

Yousif et al. [45] suggested a generic optimization and learning approach for Parkinson disease (PD) using speech and handwritten data. Eight CNNs were pre-trained on the NewHandPD dataset using transfer learning optimized by an

AO algorithm in order to detect PD from handwriting samples. Basically, AO was used to optimize the hyperparameters of various CNN models to find the optimal structure. The result of this study showed that optimization of hyperparameters using AO algorithm greatly improved the CNN model's performance and the best reported performance was 99.75% by VGG19 model.

A framework for estimating wind speeds was proposed by Khamees et al. [46], using probability distribution functions (PDFs) and the AO algorithm. The AO was used to calculate approximations for the parameters of the proposed original and mixed PDFs. The result of this study demonstrated that the mixture distributions in conjunction with the AO were more successful in simulating the frequency distribution of wind speeds than the original distribution.

Shankar et al. [47] developed an automatic fruit classification model using hyperparameter optimized deep transfer learning. Image quality is improved using a pre-processing step called contrast enhancement. The detection and classification of fruits was accomplished with the aid of a RNN model. The AO was utilized to enhance the RNN model's hyperparameters and boost classification accuracy. The effectiveness of the suggested model was tested on a benchmark dataset, and the findings indicated excellent results compared to the state-of-the-art methods.

An integrated prediction technique was developed by Li et al. [48] that combines the ELM network, the AO approach, and the EN regression technique with the goal of investigating the routes leading to net-zero CO_2 emissions. The AO technique was used to improve the efficacy of ELM. The result of this study clearly revealed that incorporating of these techniques improve the performance of the proposed method.

Li et al. [49] presented a hybrid estimate framework that incorporates the IF algorithm, the SWT technique, AO algorithm, and the LSTM. Initially, anomalous data was identified using the Isolated Forest algorithm. The original power signal from the new wind turbine was then de-noised using the SWT technique. Using the extended short-term memory network algorithm, the wind power prediction framework was constructed. The AO algorithm was utilized to optimize the LSTM parameters in order to eliminate the impact of random parameters on prediction accuracy. The results demonstrated that the presented model accurately predicts the power output of new wind turbines.

An adaptive NFIS (ANFIS) hybridization with an AO algorithm was presented by Jamei et al. [50]. The BRF feature selection approach was created to estimate the salinity of multi-aquifers in Bangladesh's seaside regions. The experimental findings demonstrated that compared to ANFIS-SMA and ANFIS-ACO, the proposed ANFIS-AO approach performed better in terms of the correlation coefficient (R), RMSE, and the KGE.

A unique mechanism based on the AO algorithm was suggested by Samal et al. [51] to optimally integrate distributed generations into the distribution system. This research was conducted with the goal of decreasing active power loss while maintaining bus voltage limitations, branch current restrictions, and active power injection from distributed generations. Total active power loss in connection to AO algorithm parameters was also evaluated. The findings of this study demonstrated that the proposed approach results in less total active power loss than the other approaches.

The new routing method proposed by Sing et al. [52] uses the whale-based tunicate swarm algorithm (WTSA) for CH selection and the AO algorithm for inter-CH routing. Effective CHs in the WSN network were identified using WTSA, and the remaining nodes joined these CHs based on the cost function value. Inter-cluster routing between far-flung CHs was accomplished with the help of AO algorithm. AO was utilized to grow efficient connections between CHs and eliminate inefficient connections using a fitness function. The efficiency of the suggested technique was contrasted with BOA, PSO- WTSA, GSA, BERA, and LEACH in terms of delay, throughput, and average energy consumption. The result showed that, in comparison to WTSA, the suggested method significantly enhanced the performance of WSN by decreasing average energy consumption by 8.3%.

Venkateswarlu et al. [53] proposed an Aquila feedback artificial tree (AFAT) based deep residual network (DRN) for the identification of fake news and its impact. AFAT was used to fine-tune the DRN's weights to their optimal settings and helps to determine how many people shared the fake news. The proposed method performed well in comparisin to other tested techniques in terms of accuracy, sensitivity, and specificity.

For oil production prediction, AlRassas et al. [54] suggested an improved ANFIS built with the help of the AO. The prediction accuracy of ANFIS was greatly improved by employing the AO to optimize its parameters. Five different algorithms (GWO, PSO, SSA, SCA, and SMA) were tested and compared to the proposed AO-ANFIS algorithm. Significant results were obtained by AO-ANFIS, which outperformed the other examined algorithms in terms of RMSE, MAE, and R^2.

An architecture for classifying COVID-19 images using a combination of DL and swarm-based techniques was proposed by Elaziz et al. [55]. As a deep learning model, the MobileNetV3 was utilized to train and extract useful image representations. To enhance classification accuracy while decreasing the number of dimensions necessary to represent images, an AO technique was employed as a feature selector. Experimental results demonstrated that the suggested approach performs well during feature selection and extraction, with respect to both dimensionality reduction and classification accuracy. The AO feature selection method obtains greater accuracy than competing methods based on performance metrics.

Vashishtha et al. [56] presented a technique to identify bearing flaws in a Francis turbine using a sound signal and a minimal entropy deconvolution (MED) filter. The AO algorithm, which takes the autocorrelation energy as a fitness function, has adaptively chosen the ideal filter length. Francis turbines with faulty bearings were used in experiments, and the suggested fault identification technique was compared to existing MED models. The findings of this research demonstrated that the presented technique accomplished well for detecting the faulty signal in the presence of significant noise.

For the purpose of automating the recognition of printed and handwritten Telegu characters inside a single image, Sonthi et al. [57] suggested an AO based deep learning model. To prepare the images for analysis, an adaptive fuzzy filtering method was used and the character segmentation techniques to isolate useful areas. Moreover, feature extraction was achieved by the combination of the EfficientNet and CapsuleNet models. Finally, the AO algorithm equipped with a bidirectional LSTM model was implemented for recognition procedure. The performance of the presented technique was assessed using Telugu character dataset and the results demonstrated that it gives superior performance compare to the other state-of-the-art methods.

In order to maximizing scheduled electricity from wind farms while minimizing total operational expenses, Khamees et al. [58] introduced a stochastic optimum power flow (SCOPF) technique. Since the SCOPF problem is significantly nonconvex and nonlinear, the AO algorithm was employed to solve it. The suggested method's efficacy was evaluated using standard IEEE-30, 57, and 118 bus systems. The findings of this research showed that, comparison to the other algorithms, the AO method produced in the cheapest cost of fuel.

Ma et al. [59] suggested an integrated approach to fault detection that makes use of adaptive chirp mode decomposition, gini index fusion, and a LSTM neural network optimized with the AO algorithm. The AO method guarantees the accuracy and precision of the model, can improved monitor the operating state of rolling bearing, and circumvents the time-consuming and parameter-uncertainty of human parameter modification in optimizing the parameters of LSTM. The study findings demonstrated that the suggested model outperformed the state-of-the-art intelligent diagnosis models for bearing compound faults in terms of accuracy and resilience.

4.2 Hybridization Based AO Variants

Hybridization is one of the most prominent and efficient methods to boost NIOA performance. Hybridization allows algorithms to include domain-specific information, which improves the exploration–exploitation tradeoff. It also increases the efficiency and speed of the optimizer, allowing it to handle larger datasets and more complex optimization problems. Based on the literature review hybrid AO variants can be classified into two categories based on the implementation method. The two classes are (i) Hybridization based on the entire NIOA, (ii) Hybridization based on the NIOA's operator.

Table 2 Synopsis of the hybridized AO variants

Sl. No	Considered NIOA or NIOA's operators	References
1	Spider Monkey Optimization	[103]
2	Sine–Cosine Algorithm	[104]
3	Particle Swarm Optimization	[105, 110]
4	Elephant Herding Optimization	[91]
5	Arithmetic Optimization Algorithm	[106, 107]
6	Whale Optimization Algorithm	[108, 124]
7	Grey Wolf Optimizer	[109]
8	Differential Evolution	[111]
9	Salp Swarm Algorithm	[112, 118]
10	Tangent Search Algorithm	[113]
11	Tabu Search Algorithm	[114]
12	Mayfly Optimization Algorithm	[115]
13	Seagull Optimization Algorithm	[116]
14	African Vulture Optimization	[117]
15	Artificial Rabbits Optimization	[119]
16	Simulated Annealing	[120]
17	Artificial Hummingbird Algorithm	[121]
18	Harris Hawk Optimization	[122, 129]
19	Coyote Optimization	[123]
20	Spotted Hyena Optimizer	[125]
21	Genetic Algorithm	[126]
22	Slime Mould algorithm	[127]
23	Eagle Optimization Algorithm	[128]
24	Grasshopper Optimization Algorithm	[130]
25	Hunger Games Search	[131]

In the hybridization based on the entire NIOA, the two NIOAs under consideration function independently. Nevertheless, they share information amongst their populations during the optimization procedure. It is the simplest form of hybridization that utilizes multiple NIOAs to determine the optimal solution. The main benefit of entire NIOA based hybridization is that it can allow for the strengths of different algorithms to be combined, resulting in improved performance and increased robustness. For example, one algorithm may be particularly effective at exploring the search space and another algorithm may be particularly effective at exploiting promising solutions. By combining these two algorithms, the hybrid algorithm can benefit from the strengths of both approaches.

To establish a deep fusion among NIOAs and to expand the searching capacity, NIOAs use an operator-based hybridization in which operators from one NIOA are integrated into another. Overall, NIOA's operator-based hybridization can be a powerful tool for improving the performance of NIOAs by enhancing their ability to optimize complex systems and solve challenging problems. A brief overview of the hybridized variant of AO is presented in Table 2.

4.3 Opposition Based Learning (OBL)

OBL is utilised to accelerate the convergence of metaheuristic algorithms. The theoretical underpinnings of OBL [60] were first presented in 2005, and it assess the current guess (estimate) and its opposite equal simultaneously to produce an effective solution. When a method's primary goal is to find the best solution to an objective function, considering a guess and its opposite simultaneously will improve the algorithm's performance. In traditional OBL, the opposite solution of $X \in [l, u]$, i.e., $\overline{X}$ is computed as follows.

$$\overline{X} = l + u - X \tag{12}$$

Researchers applied OBL and its variations to AO and observed that it was a decent method for increasing efficiency. For the improvement of the AO, classical OBL [61, 62], quasi based OBL [63, 64], random OBL [65, 66] had been successfully applied. In an IoD environment, Perumalla et al. [61] presented an oppositional AO-based feature selection with ML enabled intrusion detection system (OAOFS-MLIDS) model. With the use of intrusion detection, the suggested model can ensure safe access to the system. The suggested model uses min–max normalization as a preliminary pre-processing step for the networking data to reach this goal. The findings of this study demonstrated that the suggested model outperformed other models in the evaluation using the benchmark dataset. An enhanced ANFIS model was suggested by Al-qaness et al. [62], using AO and OBL methods to predict production of oil. The proposed model's fundamental concept is to improve the AO's search process and use the AOOBL to elevate the ANFIS's functionality. Many performance criteria, such as RMSE, MAE, R^2, Std, and computing time, were utilized to assess the presented model against real-world oil production datasets compiled from several oilfields. The experimental result showed that proposed AOOBL method gives better result for ANFIS model compared to PSO, GWO, SCA, SMA, and GA. Hou et al. [67] suggested a hybrid AO (HAO) method that takes advantage of both OBL and altruism in order to find the best possible settings for a monopole antenna. By using OBL technique, HAO increases algorithm population diversity, accelerates algorithm convergence, and incorporates altruism to boost the algorithm's potential to escape the local optimum. The HAO's performance was measured against that of the unimodal benchmark and the Multimodal benchmark; the results showed that the HAO performed better than other tested algorithms. Kandan et al. [63] suggested a quasi-oppositional AO (QOAO)-based technique to job scheduling in an IoT-enabled cloud environment. To boost the functionality of the conventional AO, the QOAO was developed. The proposed method minimizes the makespan while considering the interdependencies between the various tasks and outperformed the other algorithms in terms of flow time, span, throughput, lateness, and utilization ratio. Ma et al. [64] presented an improved AO by merging a quasi-OBL technique with a wavelet mutation approach. The quasi-OBL increases diversity in the population and promotes the efficacy of the solution, while the wavelet mutation method improved the algorithm's computation accuracy and its capacity to escape the local optimum. The suggested IAO method was evaluated on the CEC2017 test functions and found to have quick convergence rate and greater convergence accuracy compared to existing algorithms. For effective text-based information retrieval, Kumar et al. [65] suggested an effective AO algorithm based on COOT search and random OBL model. Because of its superior efficiency in exploitation search, the COOT algorithm was used. At the exploitation phase, used of a random OBL method improved the algorithm's capacity to escape from local optimums. When compared to existing algorithms, the suggested approach does well on performance metrics such as recall, precision, MRR, MAP, F-measure, and NDCG. Gao et al. [66] suggested an improved AO (IAO) that enhanced the performance of classical AO algorithm in three stages. At first, search control factor (SCF) was incorporate a to enhance the optimization process; the SCF's absolute value decreases with each iteration, which helps the AO's hunting methods. Second, to improve the algorithm's exploitability, the random ROBL technique is incorporated. At last, the Gaussian mutation (GM) technique is used to fine-tune the exploration phase. The proposed IAO's performance was evaluated on the CEC2019 test functions, and the results indicating that the algorithm outperforms traditional AO and other popular methods.

Researchers has also incorporated OBL with other strategies in order to boost the efficiency of classical AO. By employing a OBL and Nelder-Mead simplex search method an improved AO algorithm was proposed in [68]. The OBL helps the AO become more diverse, and the NM technique helps it become more intense. Using benchmark functions from the CEC 2019 test functions, the study findings revealed that the presented approach outperformed the original AO algorithm. Ekinci et al. [69] suggested an innovative and effective technique for vehicle cruise control by enhancing the performance of the AO algorithm through chaotic local search and OBL procedures. A straightforward but effective objective function was utilized to improve the effectiveness of the suggested approach and accurately regulate the real PIDD2 controller's parameters. Through the experiments using well-known unimodal, multimodal benchmark functions, the outcome of this research showed that the proposed technique gives superior performance compared to GWO, PSO, SSA and the original AO. An improved AO technique based on OBL, restart strategy, and chaotic local search was proposed by Yu et al. [70] for global optimization and constrained engineering issues. Where OBL enhances the original algorithm's exploration phase, restart strategy to replace the worse agents with completely random agents, and chaotic local search to offer additional exploitation capabilities to the original algorithm. Five different engineering optimization problems and 29 CEC2017 functions were used to evaluate the efficiency of the presented approach, and the results showed that it performs better than other tested algorithms.

4.4 Chaotic Sequence-Based Strategies

Chaotic sequence-based strategies are a type of optimization strategy used in NIOA to improve their performance. Chaotic sequences are generated using chaotic maps, which are mathematical functions that exhibit chaotic behavior. These sequences can be used in NIOA algorithms to introduce randomness and assortment in the search process, which can help to circumvent getting stuck in local optima and improve the exploration of the search space. There are several ways in which chaotic sequence-based strategies can be used in NIOA algorithms. One approach is to use a chaotic sequence to generate initial population of solutions for the optimization algorithm. This can be done by using the chaotic sequence to initialize the variables of the optimization problem. Another approach is to use the chaotic sequence to perturb the solutions generated by the optimization algorithm during the search process. This can be done by adding a small amount of chaos to the variables of the solutions, which can help to explore the search space more effectively. The chaotic sequence can be used to perturb the solutions generated during the local search process, which can help to explore the search space more effectively and circumvent getting stuck in local optima. Overall, chaotic sequence-based strategies can be an effective tool for improving the performance of NIOA algorithms. By introducing randomness and diversity into the search process, these strategies can help to avoid getting stuck in local optima and improve the exploration of the search space. Duan et al. [71] proposed a short-term, multistep model for predicting solar radiation based on the WRF-Solar framework, by utilizing CNN, and chaotic AO. First, the WRF-Solar model forecasts solar radiation, and spliced data are input into five completely CNNs for prediction. At last, the final solar radiation forecast was generated by integrating the five networks' predictions using a chaotic AO technique. The experimental result showed that incorporation of chaotic AO greatly improves the utilized neural network performance. Kujur et al. [72] presented a chaotic AO in order to solve the demand response program in a residential microgrid that is connected to the main power grid. The primary goal is to reduce total user costs in a building with a variable power rate by optimizing the scheduling pattern of linked devices. The performance of both the Chaotic AO and AO algorithms is evaluated independently. The experimental result showed that comparison to other tested algorithms proposed algorithms performs well and it also revealed that use of chaotic sequence greatly improved the AO performance.

4.5 Lévy Flight-Based Strategies

In recent years, Lévy flight (LF), a search strategy in metaheuristic optimization algorithms, has gained traction as a technique of addressing complex real-world problems. The search procedure is characterized in an LF-based approach as a stochastic walk, with the distribution of the steps performed based on a Lévy probability distribution. Because of the "heavy tails" of this distribution, greater steps can be taken more frequently than in a typical random walk. Exploring other regions of the search space and escaping local optima can both benefit from this trait. It has been found that LF-based algorithms perform better than their non-LF counterparts, therefore they are expected to be more useful in cases where no previous information is available, the goals are difficult to detect, and the distribution of the goals is sparse. It is used in a variety of chemical, physical and biological processes and was created as a replacement for the gaussian distribution (GD) to add randomness to metaheuristic optimization methods. By combining the LF strategy with the conventional AO, Chaudhari et al. [73] suggested a superior route planning approach for UAV networks. The suggested method additionally selects the best possible paths based on a fitness value that can be established with a wide variety of inputs. The study findings demonstrated that the presented techniques perform better than SA, GA, ACO, and GA-PSO. A cluster-based energy-efficient routing protocol for WSNs was proposed by Ramamoorthy et al. [74], using an optimization technique. A modified AO with fuzzy based on the levy flight strategy was developed for optimum and effective cluster formation and cluster head computation. The suggested method's primary purpose was to determine which node has the highest trust level and this node will serve as the cluster's head. The results of this study demonstrated that the suggested technique was efficient in energy consumption, average hop count, throughput, and network lifetime.

4.6 Others Strategies

In order to overcome the shortcomings of AO, Huang et al. [75] proposed a hybrid AO (HAO) based on **gauss map** and **crisscross operator**. To boost the superiority of the initial population, the Gauss map was implemented into the Aquila initialization procedure. The crisscross operator was used to encourage information exchange within the population and preserve the diversity at every iteration, which not only improves the method capabilities to escape the local optimum but also speeds up the global convergence. Through experiments utilizing 21 classical benchmark functions, this study demonstrated that the proposed HAO obtained superior performance compared to classical AO, GWO, WOA, and CSO.

Zhang et al. [76] suggested an adaptive AO in combination with **niche thought** and **dispersed chaotic swarm** algorithm to overcome the limitations of AO. The initial populations were generated using DLCS chaotic mapping to improve the algorithm's exploration state by increasing diversity of populations and uniformity of distribution in the search space. Then, an adaptive adjustment technique of de-searching preferences is provided to enhance the algorithm's search precision. Experiments on 15 benchmark functions, the Wilcoxon rank, and engineering problems demonstrate that the presented approach outperformed other sophisticated algorithms in terms of search efficiency and computational efficacy.

Ekinci et al. [77] presented a new balanced AO for air–fuel ratio control systems that makes use of **random learning** and **Nelder-Mead simplex search** techniques. These approaches were incorporate in AO, in order to enhance exploration and exploitation ability of the original AO. In addition, a new objective function was introduced in the system to recognize the optimal parameters. The experimental results revealed that the suggested method improves air–fuel ratio system control by greater than 64% for overshoot, 9% for rise time, and 7% for settling time.

An improved AO method developed by Wang et al. [78], with the use of a **velocity-aided global search** and **adaptive OBL** strategy. The acceleration and velocity parameters were applied to the AO to assist the search agent in updating its position and to prevent several dominant locations from being ignored through the optimization procedure. The adaptive OBL method was used to get rid out from the local optima. The efficacy of the method was assessed using 27 benchmark functions, the Friedman test, and five engineering problems. The experimental outcomes demonstrated that the presented method outperforms AO, IAO, and other tested algorithms.

For faster convergence, Verma et al. [79] incorporated **chaotic mapping** into the traditional AO. A single-stage **evolutionary algorithm** is also incorporated into AO in order to maintain the equilibrium between its exploration and exploitation capabilities. The suggested framework generates one population using conventional AO and another using a single stage GA based evolutionary strategy that employs **binary tournament selection**, **roulette wheel selection**, **shuffle crossing over**, and **displacement mutation** to build a new population. The efficiency of traditional AO and improved AO was evaluated using well-known unimodal and multimodal benchmark functions. The findings of this research demonstrated that improved AO with chaotic mapping and evolutionary algorithm performs superior then the traditional AO.

In order to improve wireless communication, Alangari et al. [80] presented a **wavelet mutation** AO-based routing algorithm. One goal of the proposed method was to enable energy-aware routing in WSNs. The wavelet mutation process introduces a random mutation into the each individual subsequent to the exploitation and exploration stage. The experimental result showed that compare to other tested algorithms the proposed technique performs better and improve the wireless communication.

To resolve the ORPD, Long et al. [81] presented an improved AO (IAO) algorithm with multiple strategies, including: a **probabilistic perturbation** strategy to better balance global exploitation and local exploration; a **Cauchy operator** to boost the population diversity of Aquila; an **elite group navigation** strategy to further avoid falling into a local optimum; and an analysis of IAO's performance. The suggested method's efficacy was evaluated using standard IEEE-30, 57, 118 bus systems. The result of this study demonstrated that compared to classical AO the presented method performs well to solve the ORPD.

Kun et al. [82] developed an enhance variant of AO in which the original random method was altered with **normal-distribution parameters** and an adaptive function defined by a **Weibull function** was incorporated to the predator's motion law. Using 16 benchmark functions, the enhanced algorithm is compared to numerous newly invented methods. The results demonstrate that the convergence speed and accuracy of the modified algorithm are enhanced, while the original algorithm's benefits are maintained.

The unique weighted adaptive AO developed by Das et al. [83] was offered to address the ORPD's complex and non-linear nature. By incorporating a **weighted adaptive searching** technique mechanism into the original AO, an improved version of AO was created that could optimize any optimization method more quickly and accurately than the original AO. The suggested method's efficacy was evaluated using standard IEEE-30, 57, 118 bus systems. The findings of this study demonstrated that improved AO significantly reduces power consumption compared to the other tested algorithms.

An improved variant of the AO algorithm was presented by Li et al. [84] to overcome the shortcomings of the original AO. Initially, the **inertia weight factor** was incorporated into the location update method during the global expansion stage to modify the optimal individual's level of population influence. Second, the **vertical and horizontal crossing** approach was incorporated into the algorithm's local advancement to facilitate the sharing of individual data across dimensions and enhance the algorithm's local advancement capability; Third, **Cauchy elite mutation** was incorporated in order to avoid the method from falling into local optima. Through the test experiments on 23 benchmark functions, the results of this study showed that the accuracy and stability of the presented method were enhanced in comparison to the original AO.

Turgut et al. [85] presented an improved AO, in order to fix the flaws in the original AO. To improve the standard AO's exploratory capabilities, an ensemble of **wavelet mutation operators** has been incorporated into the algorithm. In addition, a novel local search method was added into the foundational AO to focus on the previously explored attractive locations by capitalizing on the synergies between **elite opposition-based learning** and a straightforward, yet efficient, **exploitative manipulation equation**. Through tests on various benchmark functions, the findings from this research showed that the accuracy and consistency of the proposed method are superior to those of the original AO.

In order to improve cancer diagnosis on medical imaging, Alkhalaf et al. [86] developed an adaptable AO (AAO) that is enabled by explainable AI. The goal of the suggested method was to efficiently classify cancers of the colon and bone. The AAO algorithm employs the sharing model developed by **niche thought**, which is used to evaluate the relative distance between individuals within a specific environment. In order to guarantee that the state of each individual was optimal, a threshold was established to raise the fitness of a person with the maximum fitness. As compared to existing algorithms tested using medical cancer imaging databases, the suggested approach demonstrated superior accuracy.

Zhao et al. [87] developed a multiple updating principle using a **heterogeneous strategy** to optimize the AO algorithm's slow later convergence rates and boost its overall performance. Simulation experimentations were conducted on unimodal and multimodal benchmark functions, and comparisons were made with other algorithms; the majority of the results corroborated the enhanced functionality, demonstrating superior intensification and diversification capabilities, a fast convergence speed, low residual errors, robust scalability, and convincing validation outcomes.

4.7 Binary AO

The study of discrete problems is crucial to the field of optimization. Unlike the continuous distribution of the decision space for continuous problems, the decision space for discrete problems is not uniform, and a minor modification in a single dimension of the solution might have a large impact on the solution in the objective space. Application of AO to discrete issues requires refinement of its algorithmic foundation and encoding approaches. Several AO variations have been created by researchers. The fundamental mathematical principles underlying AO have not changed, however researchers have refined the original AO's encoding methodology by assigning binary values to each dimension of the candidate solution. For instance, the binary encoding binarizes the probable solutions while keeping the algorithmic basis of the original AO and adapts the operator formulas to accommodate binary data operations.

Bas [88] developed a binary AO, namely (BAO) and employed for 0–1 knapsack problem. With the use of eight distinct transfer function tests, the BAO-T algorithm was developed. In order to move from the continuous search space to the binary search space, transfer functions are utilized. Moreover, BAO has been created by incorporating candidate solution step crossover and mutation techniques (BAO-CM). The results demonstrated that BAO-CM outperformed BAO-T and can be recommended as a substitute method to solve binary optimization issues.

Nadimi-Shahraki et al. [89] introduced two binary variants of the AO and used them to determine the effective characteristics of the wrapper method. Two binary algorithms, the S-shaped binary AO (SBAO) and the V-shaped binary AO (VBAO), have been presented for feature selection in medical datasets. S and V-shaped transfer functions are utilized to generate binary position vectors while the search space remains continuous. Using seven standard medical datasets, the proposed techniques are contrasted with six modern binary optimization algorithms. The outcomes of this study showed that SBAO was superior to the examining algorithms in terms of accuracy achieved with the fewest number of features.

To recognize the most informative genes, Pashaei [90] developed a novel wrapper gene (or feature) selection mechanism based on a modified binary AO (MBAO) with a time-varying mirrored S-shaped (TVMS) transfer function. In the first stage of the presented method, top-ranked genes are selected using the filtering method of Minimal Redundancy Maximum Relevance, and in the second stage, the most discriminative genes are identified using the efficient wrapping strategy of MBAO-TVMS. A mutation mechanism is included into binary AO help the method to escape from local optima and increase its global search abilities, while

TVMS is used to convert the continuous variant of AO to a binary one.

The binary AOA still has many shortcomings. To begin, most research does not discretize the AOA formula, but rather translates the continuous values of the possible solutions into discrete binary values using a conversion function. This strategy is straightforward but inefficient since the local area in continuous space can be formed by making slight tweaks to the candidate solution across multiple dimensions, even though continuous and discrete spaces are fundamentally different from one another. In contrast, describing the local area in a discrete space is tough, and in discrete space, a seemingly insignificant modification in one dimension of a potential solution might have a huge impact on the objective value.

4.8 Multi-Objective AO

Multi objective optimization is diverse from traditional single objective optimization in that it requires balancing many goals. Optimal solutions are hard to come by because competing aims make it necessary to sacrifice one for the sake of the other. Instead, they communicate and work together to reach the greatest possible outcome. Hence, optimizing a problem with many objectives is a difficult task in the field of optimization. Because to its excellent performance in maximizing a single objective, AO has been the subject of numerous attempts to improve its ability to tackle multi-objective situations.

In order to save bandwidth and improve load balancing and cloud performance, Mohamed et al. [91] presented a multi-objective optimization based on the suggested AOEHO. Additionally, the proposed method minimizes consumer response time and waiting time. Experimental outcomes showed that the presented AOEHO method performed better than other algorithms when compared to the same benchmarks applied to replicated data.

In order to achieve optimal performance, Wang et al. [92] designed a hybrid system and optimized it using an improved Multi-objective AO. Exergy efficiency and total product cost have both been optimized via this multi-objective optimization process. The result of this study showed that the proposed algorithm gives satisfactory result and enhanced the accuracy and precision.

A multi-objective AO was presented by Xing et al. [20] in order to locate the optimal weights, and a theoretical demonstration suggests that the weights supplied by this optimizer are Pareto optimal solutions. The results of the experiments showed that the designed method achieved greater accuracy than the examined methods in every case for point forecasting, and achieved a forecasting interval with great coverage and minimal error.

Ali et al. [93] developed a multi-objective modified AO in order to find the optimal numerous renewable energy sources in distribution networks. The employment of this multi-objective AO comprises reducing the amount of power loss and total voltage deviation in a distribution system while maintaining the system's operational and security limits. The results of the experiments demonstrated that the suggested algorithm generated competitive results in comparison to conventional AO and PSO, and trader-inspired algorithms, when applied to the IEEE-33 bus.

A multi-objective improved AO (MOIAO) was developed by Long et al. [81] to solve optimal reactive power dispatch. This approach incorporates a probabilistic perturbation method to improve the equilibrium between global exploitation and local exploration. The MOIAO is utilized to achieve optimal performance in terms of the P_{loss} and V_d. The study findings showed that the presented MOIAO outperformed the state-of-the-art methods MOPSO, MOAO, and MOIPSO.

Even though Multi-objective AO produced remarkable results, it is clear that very few works and applications were actually completed. Therefore, more and more researchers are turning to AO to address complex optimization issues with many objectives. In terms of exploration, exploitation, and convergence speed, AO has been shown to perform better than competing methods in experimental settings. As a result, using AO to MOPs could be an interesting area of study. Further studies on multi-objective AO are needed because there is currently a dearth of data on the topic.

5 Application Areas of AO and Its Variants

Aquila Optimizer is a relatively new NIOA algorithm that has shown promising outcomes in resolving various optimization problems. It has been revealed that the AO is efficient in resolving a variety of optimization problems, including unconstrained and constrained optimization problems, single and multi-objective optimization problems, and continuous and discrete optimization problems. The AO has been employed in various fields of study, including engineering, power systems, image and signal processing, machine learning, renewable energy, and transportation. Table 3 provides a brief summary of these application areas.

6 Evaluation of AO

The AO's efficiency is evaluated by using a mathematical benchmark functions CEC 2020 and validated using many statistical criteria and measurements namely: Average Fitness (Mean), Standard Deviation (STD), Best Fitness (Best), and Worst Fitness (Worst) for all solutions obtained by the

Table 3 A succinct synopsis of the application areas of AO variants

Sl. No	Application area	References
1	PID controller	[16, 38, 68, 69]
2	Multi Robot Space Exploration	[11, 42]
3	0–1 knapsack problem	[88]
4	Text, Data clustering and Mathematical benchmark functions	[75, 111]
5	Solar radiation prediction	[71]
6	Demand response management of microgrid	[39, 72]
7	Feature selection	[17, 32, 41, 61, 89, 108, 121, 124]
8	Mathematical benchmark function	[76, 84, 85, 106, 119]
9	Deep Learning model optimization	[12, 21, 27, 34, 36, 37, 45, 47, 49, 53, 55, 57, 59, 86, 103, 104, 123, 125–128]
10	Air fuel ratio control system	[77]
11	Mathematical benchmark function and engineering optimization problems	[66, 70, 78, 79, 87, 107, 109, 113, 117, 122, 129–131]
12	UAV/Robot path planning	[13, 73, 112, 120]
13	Text based information retrieval	[65]
14	Offshore wind farm power collection system	[23, 105]
15	Fog Computing	[91]
16	Wireless sensor network (QoS)	[14, 15, 22, 26, 52, 74, 80, 115]
17	Time Series Forecasting (Parameter optimization of ANFIS model)	[62]
18	Energy optimization	[92, 93]
19	Parameter estimation of the controlled autoregressive	[18]
20	Multilevel inverter	[19]
21	Wind Forecasting	[20, 30, 46, 116]
22	IOT	[21, 29, 63]
23	Parameter optimization of SVM	[24, 124]
24	Flow shop scheduling	[114]
25	MLT based image segmentation	[25]
26	Neuro Fuzzy system optimization	[50, 54, 118]
27	Photovoltaic (PV) energy storage	[28, 31, 119]
28	Under water structured light vision calibration	[33]
29	Parameter optimization of simple neural network	[35, 48, 82]
30	Gene selection	[90]
31	Optimal de-active power dispatch	[81, 83]
32	Parameter optimization of extreme bosting machine	[40]
33	Motorized buoy path following problem	[43]
34	Antenna design	[67]
35	DV-Hop node positioning	[44]
36	Distributed generations in distribution system	[51]
37	Detection of bearing defect in Francis turbine	[56]
38	Parameter optimization of population forecasting model	[64]
39	Stochastic power flow problem	[58]

AO and other tested algorithms. In the CEC 2020, 1-unimodal function (F1), 3-basic functions (F2-F4), 3-hybrid functions (F5-F7), and 3- composite functions (F8-F10) are provided for the evaluation of an algorithm. Moreover, the Friedman test is also used to rank these algorithms. Results are compared with well-known techniques namely: Nutcracker optimization algorithm (NOA) [94], Arithmetic optimization algorithm (AOA) [95], Dragonfly Algorithm (DA) [96], Genetic Algorithm (GA) [97], Smell Agent Optimizer (SAO) [98], COVIDOA [99], Butterfly Optimization Algorithm (BOA) [100], Sand Cat Swarm Optimization (SCSO) [101], and Prairie dog optimization (PDO) [102]. Table 4 gives the configuration parameters for each algorithm. All parameters for all used algorithms are set to the values from their original papers to have a good and fair comparisons. In addition to that all the tested algorithms have been run

Table 4 Parameter Settings of the tested optimization algorithms

Algorithm	Parameter	Value
AO	U	0.00565
	r1	10
	ω	0.005
	α	0.1
	δ	0.1
	G1	∈[− 1, 1]
	G2	∈[2, 0]
NOA	P_{a1}	Decrease from 1 to 0
	P_{a2}	0.2
	σ_1	∈[0, 1]
	σ_2	∈[0, 1]
AOA	α	5
	μ	0.5
BOA	a	0.1
PDO	ρ	0.1
	ε	2.2204e-16
	Δ	0.005
DA	radius	0.2
COVIDOA	Shifting No	1
	Mutation	0.1
	No. of proteins	2
SCSO	r_G	2 up to 0
	R	$-2.r_G$ to $2.r_G$
GA	M	0.2
	C	0.8
SAO	T	3
	SL	2.5
	M	2.4

using population size = 50 and termination condition is the maximum number of iterations which is set to 1000. The simulations were carried out on a computer with a 2.0 GHz processor and 16 GB RAM using MATLAB R2022b that runs on Windows 10.

The results for these experiments are given in Table 5 for each function in CEC 2020 with 30 dimensions. From these results, it can be easily seen that AO has a very good results in all used criteria. For example, AO has been ranked in the first place in seven functions (seven out of ten). These functions are F2, F3, F4, F6, F8, F9, and F10. It is also ranked third in two functions F1 and F5.

Boxplot analysis is utilized to demonstration the data distribution characteristics, which can deliver a finer understanding of the distribution of results, especially for functions related with too many local minima. Figure 5 shows the boxplots of results for every algorithm and function, which represent data distributions into quartiles. The edges of the whiskers signify the lowest and largest data points reached by the algorithm, while the lower quartile and upper quartile are surrounded by the ends of the rectangles. A narrow boxplot indicates high agreement among data. The outcomes of ten functions boxplot for Dimension = 30 are shown in Fig. 5. The boxplots of AO algorithm are generally narrower compared to other algorithm distributions, with lower values.

In this subsection, we examine the convergence behaviour of AO in comparison to other competing algorithms. Figure 6 illustrates the convergence curves for the CEC 2020 functions, where the AO reaches a stable point for all functions. This indicates that the algorithm is capable of converging. Additionally, the proposed algorithm achieves the lowest average of the best solutions at a faster rate than most functions. This rapid convergence to the (near) optimal solution is noteworthy and suggests that the AO algorithm is a viable option for optimizing problems that demand fast computation, including online optimization problems.

Wilcoxon's rank-sum test is a non-parametric statistical test utilized to comparison two samples and determine if they have different medians. In this case, it is being utilized to assess the implication of the performance difference amongst the AO algorithm and the other competing algorithms. Table 6 presents a paired-algorithm comparison of the average of the best-so-far solutions using the Wilcoxon's rank-sum test with a significance level of 5%. The table shows the pair-wise contrast of the average of the best-so-far solutions of two groups for each function, and the p-values are computed. If the p-value is less than or equal to 0.05, it means that the difference between the two groups is statistically significant at the 5% level. The results of the test show that for most functions, the *p*-values are less than or equal to 0.05, indicating that the performance difference between the AO algorithm and the other competing algorithms is statistically significant. This suggests that the AO algorithm is not randomly producing better results, but rather it is performing better due to its optimization rules and strategies.

7 Conclusions and Future Directions

This paper provides a current overview of AO, its various versions, and their uses in various optimization disciplines. The strong research advancement of AO draws researchers to work on it. In this study, we outline the article collection process and provide a review of AO, highlighting its time commitment and parameter robustness. Following that, a discussion of the used enhancement tactics for AO is held. Furthermore, binary AO variations for discrete problem solution are also presented. In multi-objective based optimization areas, AO and its advancements delivered outstanding results. The use of various AO variations to tackle discrete single and multi-objective problems are remarkable and inspiring. We also assess the performance of AO over

Table 5 Aquila Optimizer results compared with other NIOA

Function	Measure	AO	NOA	AOA	DA	GA	SAO	COVIDIA	BOA	SCSO	PDO
F1	Best	3.62E+06	1.20E+10	2.72E+10	8.29E+06	1.85E+10	4.94E+03	1.66E+10	3.99E+03	3.97E+04	1.46E+10
	Worst	2.37E+07	3.02E+10	2.72E+10	9.04E+08	3.89E+10	9.60E+03	2.82E+10	7.92E+03	4.81E+09	3.19E+10
	Mean	8.97E+06	2.47E+10	2.72E+10	3.86E+08	2.90E+10	6.81E+03	2.34E+10	5.33E+03	1.17E+09	2.44E+10
	STD	4.58E+06	3.77E+09	1.16E-05	2.57E+08	5.99E+09	1.20E+03	2.83E+09	7.67E+02	1.25E+09	4.52E+09
	Rank	3	8	9	4	10	2	6	1	5	7
F2	Best	2.18E+03	4.62E+03	3.74E+03	2.32E+03	4.58E+03	4.65E+03	3.91E+03	4.99E+03	2.30E+03	4.11E+03
	Worst	4.30E+03	5.65E+03	4.45E+03	6.12E+03	6.47E+03	6.99E+03	5.65E+03	5.79E+03	4.83E+03	6.16E+03
	Mean	3.15E+03	5.28E+03	3.76E+03	4.20E+03	5.59E+03	5.65E+03	5.23E+03	5.42E+03	3.61E+03	5.27E+03
	STD	5.55E+02	2.39E+02	1.29E+02	7.16E+02	4.16E+02	5.19E+02	2.98E+02	2.30E+02	5.96E+02	5.00E+02
	Rank	1	7	3	4	9	10	5	8	2	6
F3	Best	7.98E+02	1.30E+03	9.63E+02	8.03E+02	9.89E+02	1.02E+03	1.00E+03	9.46E+02	8.14E+02	9.11E+02
	Worst	9.03E+02	1.62E+03	9.86E+02	9.98E+02	1.13E+03	1.17E+03	1.10E+03	1.04E+03	9.70E+02	1.17E+03
	Mean	8.37E+02	1.51E+03	9.85E+02	8.54E+02	1.06E+03	1.10E+03	1.06E+03	9.85E+02	8.83E+02	9.85E+02
	STD	2.23E+01	7.24E+01	4.23E+00	4.43E+01	3.51E+01	3.36E+01	2.28E+01	2.45E+01	4.28E+01	5.90E+01
	Rank	1	10	6	2	8	9	7	5	3	4
F4	Best	1.91E+03	1.76E+04	1.94E+05	1.91E+03	5.65E+05	2.60E+04	3.70E+04	4.60E+04	1.91E+03	5.94E+04
	Worst	1.94E+03	4.22E+05	3.30E+05	2.52E+03	1.55E+07	3.11E+06	5.21E+05	4.68E+05	2.31E+03	1.50E+06
	Mean	1.92E+03	1.84E+05	1.99E+05	1.98E+03	4.33E+06	1.11E+06	1.45E+05	1.97E+05	1.99E+03	3.83E+05
	STD	7.00E+00	1.01E+05	2.48E+04	1.39E+02	3.54E+06	9.31E+05	9.55E+04	1.11E+05	8.98E+01	3.68E+05
	Rank	1	5	7	2	10	9	4	6	3	8
F5	Best	2.10E+04	2.21E+06	2.16E+05	4.24E+04	8.92E+06	4.14E+05	7.77E+05	1.47E+05	4.60E+04	7.15E+05
	Worst	1.51E+06	8.89E+06	2.94E+05	3.39E+06	4.85E+07	2.45E+07	5.05E+06	4.35E+06	1.64E+06	1.45E+07
	Mean	5.55E+05	5.14E+06	2.92E+05	1.05E+06	3.54E+07	5.22E+06	2.74E+06	1.30E+06	4.83E+05	3.59E+06
	STD	3.93E+05	1.83E+06	1.44E+04	9.74E+05	1.29E+07	5.14E+06	9.90E+05	9.58E+05	4.55E+05	2.70E+06
	Rank	3	8	1	4	10	9	6	5	2	7
F6	Best	1.88E+03	2.41E+03	2.58E+03	2.07E+03	2.71E+03	2.99E+03	2.69E+03	2.59E+03	1.82E+03	2.48E+03
	Worst	2.49E+03	3.29E+03	2.67E+03	3.08E+03	4.59E+03	4.26E+03	3.45E+03	3.75E+03	2.68E+03	3.84E+03
	Mean	2.12E+03	3.05E+03	2.58E+03	2.54E+03	3.55E+03	3.52E+03	3.07E+03	3.07E+03	2.23E+03	3.04E+03
	STD	1.73E+02	1.82E+02	1.80E+01	2.85E+02	4.16E+02	3.35E+02	2.01E+02	2.99E+02	2.29E+02	3.52E+02
	Rank	1	6	4	3	10	9	7	8	2	5
F7	Best	2.43E+04	2.32E+05	1.22E+05	1.69E+04	4.31E+06	4.93E+04	1.42E+05	1.05E+05	1.03E+04	2.44E+05
	Worst	1.40E+06	3.90E+06	1.43E+05	2.15E+06	6.11E+07	2.20E+07	3.21E+06	1.00E+06	8.43E+05	1.10E+07
	Mean	4.44E+05	1.46E+06	1.42E+05	5.56E+05	2.51E+07	4.32E+06	7.49E+05	4.33E+05	1.99E+05	2.85E+06
	STD	2.93E+05	8.01E+05	3.90E+03	5.93E+05	1.52E+07	4.90E+06	6.02E+05	2.42E+05	1.71E+05	2.69E+06
	Rank	4	7	1	5	10	9	6	3	2	8

Table 5 (continued)

Function	Measure	AO	NOA	AOA	DA	GA	SAO	COVIDIA	BOA	SCSO	PDO
F8	Best	2.31E+03	4.29E+03	5.67E+03	2.35E+03	4.19E+03	4.19E+03	4.33E+03	2.49E+03	2.31E+03	3.14E+03
	Worst	2.32E+03	6.02E+03	5.78E+03	6.44E+03	7.14E+03	7.63E+03	5.79E+03	3.84E+03	5.01E+03	7.07E+03
	Mean	2.31E+03	5.26E+03	5.67E+03	3.77E+03	5.37E+03	6.60E+03	5.26E+03	2.90E+03	2.53E+03	5.22E+03
	STD	1.14E+00	4.19E+02	1.98E+01	1.59E+03	7.50E+02	9.76E+02	3.00E+02	3.33E+02	4.94E+02	1.02E+03
	Rank	1	6	9	4	8	10	7	3	2	5
F9	Best	2.88E+03	3.10E+03	3.28E+03	2.89E+03	3.11E+03	3.18E+03	3.17E+03	2.72E+03	2.85E+03	3.05E+03
	Worst	3.01E+03	3.27E+03	3.54E+03	3.33E+03	3.52E+03	3.81E+03	3.34E+03	3.72E+03	3.01E+03	3.19E+03
	Mean	2.92E+03	3.20E+03	3.29E+03	3.04E+03	3.31E+03	3.50E+03	3.28E+03	3.26E+03	2.93E+03	3.14E+03
	STD	2.93E+01	4.21E+01	4.74E+01	9.70E+01	1.04E+02	1.39E+02	4.07E+01	2.07E+02	3.86E+01	3.44E+01
	Rank	1	5	8	3	9	10	7	6	2	4
F10	Best	2.92E+03	3.84E+03	5.00E+03	2.92E+03	3.92E+03	4.94E+03	3.56E+03	3.99E+03	2.96E+03	3.62E+03
	Worst	3.04E+03	5.96E+03	6.62E+03	3.11E+03	1.11E+04	9.60E+03	5.13E+03	7.92E+03	3.45E+03	6.41E+03
	Mean	3.00E+03	5.02E+03	6.56E+03	3.01E+03	6.44E+03	6.81E+03	4.41E+03	5.33E+03	3.04E+03	4.79E+03
	STD	2.90E+01	4.62E+02	2.95E+02	4.68E+01	1.96E+03	1.20E+03	3.73E+02	7.67E+02	9.11E+01	7.45E+02
	Rank	1	6	9	2	8	10	4	7	3	5

mathematical benchmark functions (CEC2020). The results revealed AO's superiority over the other nine NIOAs. The main advantages of AO are its versatility, strong performance in situations with high dimensions and complexity, and simple structure. Nonetheless, the following are AO's main restrictions in light of the AOA's framework.

1. AO has two parameters, i.e., α and δ on which the exploitation capability of the AO depends. However, they are fixed so some constant values in most of the times. Thus, experiments must be performed on these parameters to make them adaptive in different problem domains.
2. The slow convergence speed of AO is also reported in literature. Effective population making strategies or population diversity maintain mechanism can be utilized to overcome this.
3. The exploration ability of the AO significantly depends on the Lévy distribution based random walk.

Additionally, the surveyed research reveals the following limitations:

(1) Classical AO is majorly utilized to solve several real-world optimization problems.
(2) The use of AO and its variants to solve discrete and multi-objective problems is limited. AO has primarily been used in power and control engineering, as well as ML-based model optimization.
(3) The mostly used techniques for improving AO efficiency are hybridization and OBL.
(4) Chaotic sequence, Lévy flight, and other efficiency enhancement strategies are used, but not extensively.

The relevant directions for future can be determined in light of the analysis above:

1. Additional research should be conducted on the adaptability of parameters, including the population size of AO, which makes it versatile across a variety of problems and problem dimensions.
2. Future planning should take multi-swarm or multi-population based tactics into consideration. In the near future, it is advisable to implement further cutting-edge and tried-and-true improvement tactics, particularly those based on randomization.
3. The use of AO's versions in multi-objective and discrete situations may be a promising area for future research. Furthermore, ML and computer vision applications must be made.

Qualitative analysis is typically used when examining AO. There are numerous techniques to evaluate AO

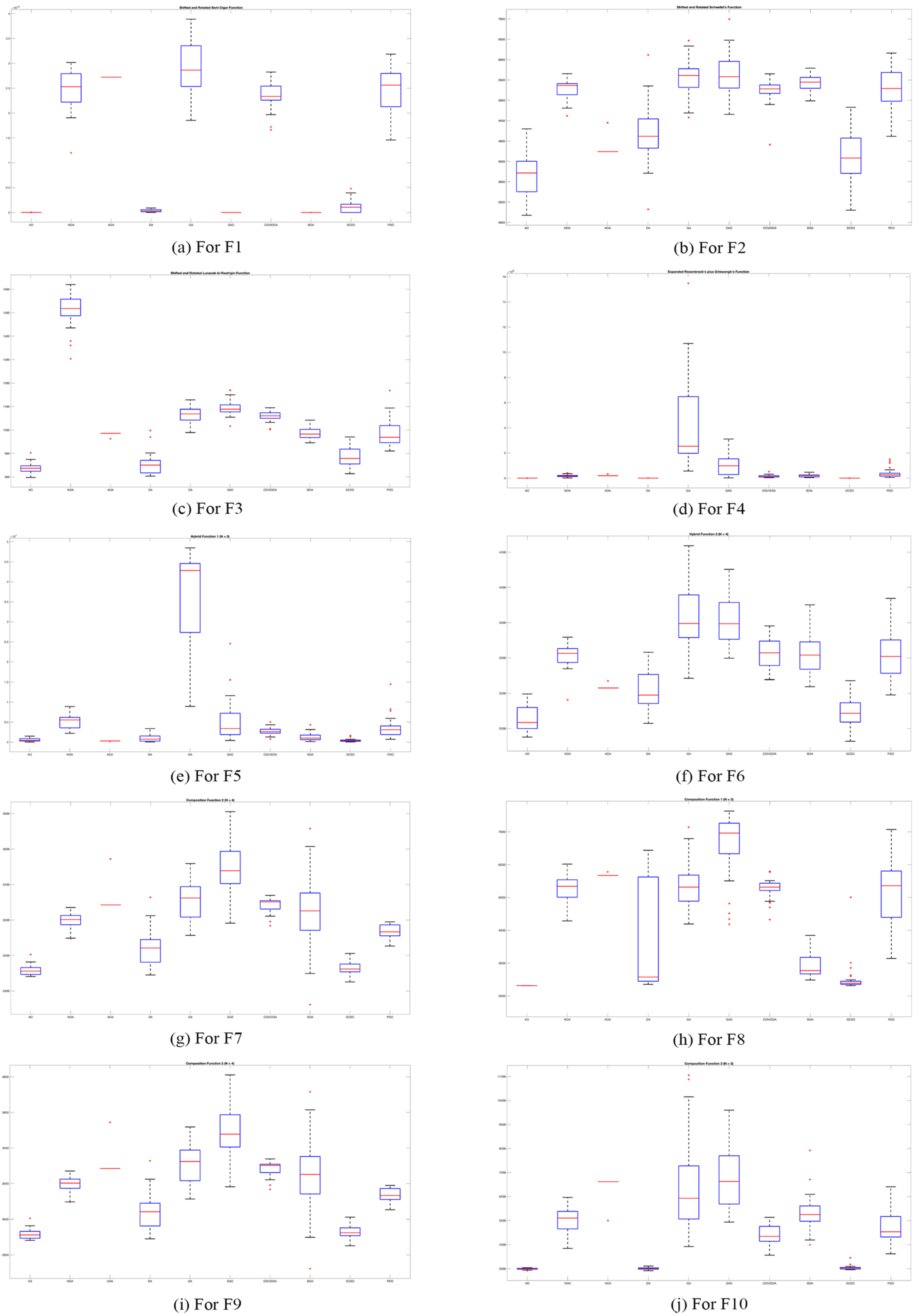


Fig. 5 Box plots for all the tested NIOAs over CEC 2020

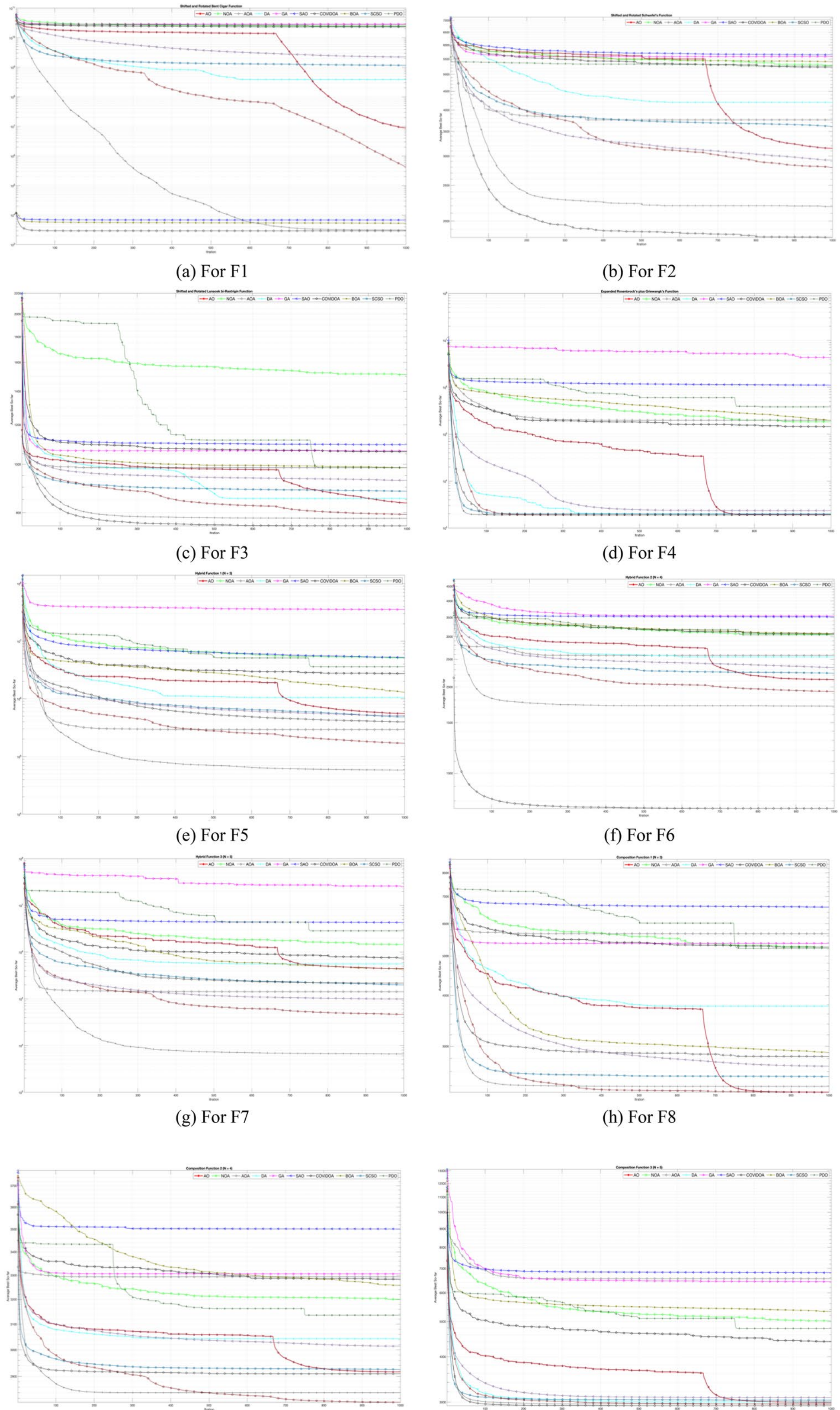


Fig. 6 Convergence curves for all the tested NIOAs over CEC 2020

(a) For F1

(b) For F2

(c) For F3

(d) For F4

(e) For F5

(f) For F6

(g) For F7

(h) For F8

(i) For F9

(j) For F10

Table 6 Wilcoxon Test (p-values) for AO vs other NIOA

Function	NOA	AOA	DA	GA	SAO	COVIDIA	BOA	SCSO	PDO
F1	3.02E-11	1.21E-12	1.09E-10	3.02E-11	3.02E-11	3.02E-11	3.02E-11	0.001953	3.02E-11
F2	3.02E-11	1.27E-08	1.73E-07	3.02E-11	3.02E-11	3.69E-11	3.02E-11	0.006972	3.69E-11
F3	3.02E-11	1.72E-12	0.115362	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.53E-05	3.02E-11
F4	3.02E-11	1.72E-12	0.864994	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.61E-06	3.02E-11
F5	3.02E-11	0.004176	0.08771	3.02E-11	1.86E-09	8.15E-11	0.000301	0.297272	1.21E-10
F6	3.69E-11	1.72E-12	8.35E-08	3.02E-11	3.02E-11	3.02E-11	3.02E-11	0.043584	3.34E-11
F7	6.01E-08	2.59E-06	0.695215	3.02E-11	1.41E-09	0.012212	0.78446	0.000104	2.19E-08
F8	3.02E-11	1.72E-12	3.02E-11	3.02E-11	3.02E-11	3.02E-11	3.02E-11	1.09E-10	3.02E-11
F9	3.02E-11	1.72E-12	2.02E-08	3.02E-11	3.02E-11	3.02E-11	3.82E-09	0.200949	3.02E-11
F10	3.02E-11	1.72E-12	0.332855	3.02E-11	3.02E-11	3.02E-11	3.02E-11	0.011228	3.02E-11

theoretically, but these studies are based on strict assumptions that, in some conditions, are also impracticable. The creation of an appropriate mathematical framework for AO may provide a new area of study.

Funding There is no funding associated with this research.

Data Availability The authors do not have the permission to share the data.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no conflict of interest. The authors declare that they have no conflict of interest.

Ethical Approval This article does not contain any studies with human participants or animals performed by any of the authors.

References

1. Abualigah L, Yousri D, Abd Elaziz M, Ewees AA, Al-Qaness MA, Gandomi AH (2021) Aquila Optimizer: a novel meta-heuristic optimization algorithm. Comput Ind Eng 157:107250
2. Li J, Wei X, Li B, Zeng Z (2022) A survey on firefly algorithms. Neurocomputing 500:662–678
3. Rai R, Das A, Dhal KG (2022) Nature-inspired optimization algorithms and their significance in multi-thresholding image segmentation: an inclusive review. Evol Syst 13(6):889–945
4. Rai R, Das A, Ray S, Dhal KG (2022) Human-inspired optimization algorithms: theoretical foundations, algorithms, open-research issues and application for multi-level thresholding. Arch Comput Methods Eng 29:5313–5352. https://doi.org/10.1007/s11831-022-09766-z
5. Dhal KG, Ray S, Das A, Das S (2019) A survey on nature-inspired optimization algorithms and their application in image enhancement domain. Arch Comput Methods Eng 26:1607–1638
6. Dhal KG, Das A, Gálvez J, Ray S, Das S (2020) An overview on nature-inspired optimization algorithms and their possible application in image processing domain. Pattern Recognit Image Anal 30:614–631
7. Yang XS (2012) Free lunch or no free lunch: that is not just a question? Int J Artif Intell Tools 21(03):1240010
8. Dhal KG, Rai R, Das A, Ray S, Ghosal D, Kanjilal R (2023) Chaotic fitness-dependent quasi-reflected Aquila Optimizer for super pixel based white blood cell segmentation. Neural Comput Appl. https://doi.org/10.1007/s00521-023-08486-0
9. Dhal KG, Das A, Sahoo S, Das R, Das S (2021) Measuring the curse of population size over swarm intelligence based algorithms. Evol Syst 12:779–826
10. Dhal KG, Sahoo S, Das A, Das S (2019) Effect of population size over parameter-less firefly algorithm. Applications of firefly algorithm and its variants: case studies and new developments. Singapore, Springer Singapore, pp 237–266
11. Gul F, Mir I, Mir S (2023) Aquila Optimizer with parallel computation application for efficient environment exploration. In *AIAA SCITECH 2023 Forum* (p. 1655).
12. Barfungpa SP, Sarma HKD, Samantaray L (2023) An intelligent heart disease prediction system using hybrid deep dense Aquila network. Biomed Signal Process Control 84:104742
13. Huang H, Li H, Wang M, Wu Y, He X (2023) Multi-UAV cooperative path planning based on Aquila Optimizer. In Proceedings of 2022 international conference on autonomous unmanned systems (ICAUS 2022). Singapore: Springer Nature Singapore, pp. 2005–2014.
14. Hemavathi S, Latha B (2023) HFLFO: Hybrid fuzzy levy flight optimization for improving QoS in wireless sensor network. Ad Hoc Netw 142:103110
15. Agarwal N, Gokilavani M, Nagarajan S, Saranya S, Alsolai H, Dhahbi S, Abdelaziz AS (2023) Intelligent Aquila optimization algorithm-based node localization scheme for wireless sensor networks. Comput Mater Continua 74(1):141–152
16. Aribowo W, Supari BS, Suprianto B (2022) Optimization of PID parameters for controlling DC motor based on the Aquila Optimizer algorithm. Int J Power Electron Drive Syst (IJPEDS) 13(1):808–2814
17. Fatani A, Dahou A, Al-Qaness MA, Lu S, Abd Elaziz M (2022) Advanced feature extraction and selection approach using deep learning and Aquila Optimizer for IoT intrusion detection system. Sensors 22(1):140
18. Mehmood K, Chaudhary NI, Khan ZA, Raja MAZ, Cheema KM, Milyani AH (2022) Design of Aquila optimization heuristic for identification of control autoregressive systems. Mathematics 10(10):1749
19. Hussan MR, Sarwar MI, Sarwar A, Tariq M, Ahmad S, Shah Noor Mohamed A et al (2022) Aquila optimization based harmonic elimination in a modified H-bridge inverter. Sustainability 14(2):929

20. Xing Q, Wang J, Lu H, Wang S (2022) Research of a novel short-term wind forecasting system based on multi-objective Aquila Optimizer for point and interval forecast. Energy Convers Manage 263:115583
21. Subba Reddy, T., Harikiran, J., Enduri, M. K., Hajarathaiah, K., Almakdi, S., Alshehri, M. et al. (2022). Hyperspectral image classification with optimized compressed synergic deep convolution neural network with Aquila optimization. *Comput Intell Neurosci*, *2022*
22. Abualkishik AZ, Alwan AA (2022) Trust aware Aquila Optimizer based secure data transmission for information management in wireless sensor networks. J Cybersecur Inf Manage 9(1):40–51
23. Abou El-Ela AA, El-Sehiemy RA, Shaheen AM, Shalaby AS (2022) Aquila optimization algorithm for wind energy potential assessment relying on Weibull parameters estimation. Wind 2(4):617–635
24. Li Z, Wang Q, Zhu B, Wang B, Zhu W, Dai Y (2022) Thermal error modeling of high-speed electric spindle based on Aquila Optimizer optimized least squares support vector machine. Case Stud Thermal Eng 39:102432
25. Rajinikanth V, Aslam SM, Kadry S, Thinnukool O (2022) Semi/fully-automated segmentation of gastric-polyp using Aquila-optimization-algorithm enhanced images. Comput Mater Continua 70:4087–4105
26. Taha AA, Abouroumia HO, Mohamed SA, Amar LA (2022) Enhancing the lifetime and energy efficiency of wireless sensor networks using Aquila Optimizer algorithm. Future Internet 14(12):365
27. Aarthi B, Chelliah BJ (2022) Deep recurrent neural network-based Aquila optimization-based online shaming emotion analysis. Concurrency Comput 34(11):e6882
28. . Karmouni H, Chouiekh M, Motahhir S, Dagal I, Oufettoul H, Qjidaa H, Sayyouri M (2022) A Novel MPPT algorithm based on Aquila Optimizer under PSC and implementation using raspberry. In *2022 11th international conference on renewable energy research and application (ICRERA)*. IEEE, pp. 446–451
29. Hosseinzadeh M, Ionescu-Feleaga L, Ionescu BȘ, Sadrishojaei M, Kazemian F, Rahmani AM, Khan F (2022) A hybrid delay aware clustered routing approach using Aquila Optimizer and firefly algorithm in internet of things. Mathematics 10(22):4331
30. Jnr EON, Ziggah YY, Rodrigues MJ, Relvas S (2022) A hybrid chaotic-based discrete wavelet transform and Aquila optimisation tuned-artificial neural network approach for wind speed prediction. Results Eng 14:100399
31. Narasimhulu N, Krishnam Naidu RSR, Falkowski-Gilski P, Divakarachari PB, Roy U (2022) Energy management for PV powered hybrid storage system in electric vehicles using artificial neural network and Aquila Optimizer algorithm. Energies 15(22):8540
32. Li L, Pan JS, Zhuang Z, Chu SC (2022) A novel feature selection algorithm based on Aquila Optimizer for COVID-19 classification. Intelligent information processing XI: 12th IFIP TC 12 international conference, IIP 2022, Qingdao, China, May 27–30, 2022, proceedings. Springer International Publishing, Cham, pp 30–41
33. Wang X, Fan J, Deng S, Wu Z, Zhou C, Ou Y, Han Y (2022) An underwater structured light vision calibration method considering unknown refractive index based on Aquila Optimizer. *IEEE Trans. Instrumentation Measurement*
34. Vinayaki VD, Kalaiselvi R (2022) Aquila Optimizer based deep neural network for Automatic Diabetic Retinopathy detection and classification. In *2022 IEEE international conference on signal processing, informatics, communication and energy systems (SPICES)*, Vol. 1. IEEE, pp. 55–60.
35. Chen Y, Wang W, Li N (2022) Prediction of the equilibrium moisture content and specific gravity of thermally modified wood via an Aquila optimization algorithm back-propagation neural network model. BioResources 17(3):4816–4836
36. Al Duhayyim M, Alabdulkreem E, Tarmissi K, Aljebreen M, El Khier BSIA, Zamani AS et al (2022) Aquila optimization with transfer learning based crowd density analysis for sustainable smart cities. Appl Sci 12(21):11187
37. Obayya M, Hassine SBH, Alazwari S, Nour MK, Mohamed A, Motwakel A, Yaseen I, Zamani AS, Abdelmageed AA, Mohammed GP (2022) Aquila Optimizer with Bayesian neural network for breast cancer detection on ultrasound images. Appl Sci 12(17):8679
38. Guo Z, Yang B, Han Y, He T, He P, Meng X, He X (2022) Optimal PID tuning of PLL for PV inverter based on Aquila Optimizer. Front Energy Res. https://doi.org/10.3389/fenrg.2021.812467
39. Aribowo W, Suryoatmojo H, Pamuji FA (2022) Optimalization droop control based on Aquila Optimizer algorithm for DC Microgrid. In *2022 international seminar on intelligent technology and its applications (ISITIA)*. IEEE, pp. 460–465
40. Nguyen H, Hoang ND (2022) Computer vision-based classification of concrete spall severity using metaheuristic-optimized extreme gradient boosting machine and deep convolutional neural network. Autom Constr 140:104371
41. Grace M, Sughasiny M (2022) Malware detection for android application using Aquila Optimizer and Hybrid LSTM-SVM classifier. EAI Endorsed Trans Scalable Inf Syst. https://doi.org/10.4108/eetsis.v9i4.2565
42. Gul F, Mir A, Mir I, Mir S, Islaam TU, Abualigah L, Forestiero A (2022) A centralized strategy for multi-agent exploration. IEEE Access 10:126871–126884
43. Guan F, Yang Z, Zhang X, Huang J (2022) Motorized buoy path following based on improved LOS algorithm and Aquila Optimizer algorithm. In *2022 IEEE international conference on mechatronics and automation (ICMA)*. IEEE, pp. 376–381
44. Yang, F., Ding, M. (2022). An improved DV-hop algorithm using hop distance correction and Aquila optimization. In *2022 International Conference on Artificial Intelligence and Computer Information Technology (AICIT)*. IEEE, pp. 1–4
45. Yousif NR, Balaha HM, Haikal AY, El-Gendy EM (2022) A generic optimization and learning framework for Parkinson disease via speech and handwritten records. J Ambient Intell Human Comput. https://doi.org/10.1007/s12652-022-04342-6
46. Khamees AK, Abdelaziz AY, Ali ZM, Alharthi MM, Ghoneim SS, Eskaros MR, Attia MA (2022) Mixture probability distribution functions using novel metaheuristic method in wind speed modeling. Ain Shams Eng J 13(3):101613
47. Shankar K, Kumar S, Dutta AK, Alkhayyat A, Jawad AJAM, Abbas AH, Yousif YK (2022) An automated hyperparameter tuning recurrent neural network model for fruit classification. Mathematics 10(13):2358
48. Li W, Zhang S, Lu C (2022) Exploration of China's net CO_2 emissions evolutionary pathways by 2060 in the context of carbon neutrality. Sci Total Environ 831:154909
49. Li Z, Luo X, Liu M, Cao X, Du S, Sun H (2022) Short-term prediction of the power of a new wind turbine based on IAO-LSTM. Energy Rep 8:9025–9037
50. Jamei M, Karbasi M, Malik A, Abualigah L, Islam ARMT, Yaseen ZM (2022) Computational assessment of groundwater salinity distribution within coastal multi-aquifers of Bangladesh. Sci Rep 12(1):11165
51. . Samal P, Panigrahy D (2022) A novel technique based on Aquila optimiser algorithm for optimal integration of distributed generations in the distribution system. *Process Integr Optim Sustain*, 1–10.

52. Singh JP, Gupta AK (2022) An optimized routing technique in wireless sensor network using Aquila Optimizer. Int J Intel Eng Syst 12(4). https://doi.org/10.22266/ijies2022.0831.28
53. Venkateswarlu B, Viswanath Shenoi V, Tumuluru P (2022) Aquila optimized feedback artificial tree for detection of fake news and impact identification. In *Web Intelligence* (No. Preprint). IOS Press, pp. 1–22
54. AlRassas AM, Al-qaness MA, Ewees AA, Ren S, Abd Elaziz M, Damaševičius R, Krilavičius T (2021) Optimized ANFIS model using Aquila Optimizer for oil production forecasting. Processes 9(7):1194
55. Abd Elaziz M, Dahou A, Alsaleh NA, Elsheikh AH, Saba AI, Ahmadein M (2021) Boosting COVID-19 image classification using MobileNetV3 and Aquila Optimizer algorithm. Entropy 23(11):1383
56. Vashishtha G, Kumar R (2021) Autocorrelation energy and Aquila Optimizer for MED filtering of sound signal to detect bearing defect in Francis turbine. Meas Sci Technol 33(1):015006
57. Sonthi VK, Nagarajan S, Krishnaraj N (2021) Automated Telugu printed and handwritten character recognition in single image using Aquila Optimizer based deep learning model. Int J Adv Comput Sci Appl. https://doi.org/10.14569/IJACSA.2021.0121275
58. Khamees AK, Abdelaziz AY, Eskaros MR, El-Shahat A, Attia MA (2021) Optimal power flow solution of wind-integrated power system using novel metaheuristic method. Energies 14(19):6117
59. Ma J, Wang X (2021) Compound fault diagnosis of rolling bearing based on ACMD, Gini index fusion and AO-LSTM. Symmetry 13(12):2386
60. . Tizhoosh HR (2005) Opposition-based learning: a new scheme for machine intelligence. In *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)*, Vol. 1. IEEE, pp. 695–701.
61. Perumalla S, Chatterjee S, Kumar AS (2023) Modelling of oppositional Aquila Optimizer with machine learning enabled secure access control in Internet of drones environment. Theoret Comput Sci 941:39–54
62. Al-qaness MA, Ewees AA, Fan H, AlRassas AM, Abd Elaziz M (2022) Modified Aquila Optimizer for forecasting oil production. Geo-Spatial Inf Sci 25(4):519–535
63. Kandan M, Krishnamurthy A, Selvi SAM, Sikkandar MY, Aboamer MA, Tamilvizhi T (2022) Quasi oppositional Aquila Optimizer-based task scheduling approach in an IoT enabled cloud environment. J Supercomput 78(7):10176–10190
64. Ma L, Li J, Zhao Y (2021) Population forecast of China's rural community based on CFANGBM and improved Aquila Optimizer algorithm. Fractal Fractional 5(4):190
65. Kumar R, Sharma SC (2023) Hybrid optimization and ontology-based semantic model for efficient text-based information retrieval. J Supercomput 79(2):2251–2280
66. Gao B, Shi Y, Xu F, Xu X (2022) An improved Aquila Optimizer based on search control factor and mutations. Processes 10(8):1451
67. Hou X, Wang M, Dong J (2022) Antennas optimal design using opposition-based learning and altruistic Aquila Optimizer algorithm. In *2022 IEEE MTT-S international wireless symposium (IWS)*, Vol. 1. IEEE, pp. 1–3
68. Ekinci S, Izci D, Eker E, Abualigah L (2023) An effective control design approach based on novel enhanced Aquila Optimizer for automatic voltage regulator. Artif Intell Rev 56(2):1731–1762
69. Ekinci S, Izci D, Abualigah L, Zitar RA (2023) A modified oppositional chaotic local search strategy based Aquila Optimizer to design an effective controller for vehicle cruise control system. J Bionic Eng. https://doi.org/10.1007/s42235-023-00336-y
70. Yu H, Jia H, Zhou J, Hussien A (2022) Enhanced Aquila Optimizer algorithm for global optimization and constrained engineering problems. Math Biosci Eng 19(12):14173–14211
71. Duan J, Zuo H, Bai Y, Chang M, Chen X, Wang W et al (2023) A multistep short-term solar radiation forecasting model using fully convolutional neural networks and chaotic Aquila optimization combining WRF-Solar model results. Energy 271:126980
72. Kujur S, Dubey HM, Salkuti SR (2023) Demand response management of a residential microgrid using chaotic Aquila optimization. Sustainability 15(2):1484
73. Chaudhari SV, Dhipa M, Ayoub S, Gayathri B, Siva M, Banupriya V (2022) Modified Aquila optimization based route planning model for unmanned aerial vehicles networks. In *2022 international conference on automation, computing and renewable systems (ICACRS)*. IEEE, pp. 370–375
74. Ramamoorthy R, Ranganathan R, Ramu S (2022) An improved Aquila optimization with fuzzy model based energy efficient cluster routing protocol for wireless sensor networks. Yanbu J Eng Sci 19(1):51–61
75. Huang C, Huang J, Jia Y, Xu J (2023) A hybrid Aquila Optimizer and its K-means clustering optimization. Trans Inst Meas Control 45(3):557–572
76. Zhang Y, Xu X, Zhang N, Zhang K, Dong W, Li X (2023) Adaptive Aquila Optimizer combining niche thought with dispersed chaotic swarm. Sensors 23(2):755
77. Ekinci S, Izci D, Abualigah L (2023) A novel balanced Aquila Optimizer using random learning and Nelder-Mead simplex search mechanisms for air–fuel ratio system control. J Braz Soc Mech Sci Eng 45(1):68
78. Wang Y, Zhang Y, Yan Y, Zhao J, Gao Z (2023) An enhanced Aquila optimization algorithm with velocity-aided global search mechanism and adaptive opposition-based learning. Math Biosci Eng 20(4):6422–6467
79. Verma M, Sreejeth M, Singh M, Babu TS, Alhelou HH (2022) Chaotic mapping based advanced Aquila Optimizer with single stage evolutionary algorithm. IEEE Access 10:89153–89169
80. Alangari S, Obayya M, Gaddah A, Yafoz A, Alsini R, Alghushairy O et al (2022) Wavelet mutation with Aquila optimization-based routing protocol for energy-aware wireless communication. Sensors 22(21):8508
81. Long H, Liu S, Chen T, Tan H, Wei J, Zhang C, Chen W (2022) Optimal reactive power dispatch based on multi-strategy improved Aquila optimization algorithm. *IAENG Int. J. Comput Sci* 49(4).
82. Hu K et al (2022) Magnetic force prediction of hybrid magnet with Halbach array using generalized regression neural network optimized by a modified Aquila Optimizer. Int J Appl Electromagnet Mech 71:1–24
83. Das T, Roy R, Mandal KK (2022) A novel weighted adaptive Aquila Optimizer technique for solving the optimal reactive power dispatch problem. Res Sq. https://doi.org/10.21203/rs.3.rs-2134558/v1
84. Li X, Ma Y, Li Y, Li H, Zuo H, Sun W (2022) Improved Aquila Optimizer optimization algorithm based on multi-strategy fusion. In *2022 2nd international conference on computer science, electronic information engineering and intelligent control technology (CEI)*. IEEE, pp. 757–760.
85. Turgut OE, Turgut MS (2023) Local search enhanced Aquila optimization algorithm ameliorated with an ensemble of wavelet mutation strategies for complex optimization problems. Math Comput Simul 206:302–374

86. Alkhalaf S, Alturise F, Bahaddad AA, Elnaim BME, Shabana S, Abdel-Khalek S, Mansour RF (2023) Adaptive Aquila Optimizer with explainable artificial intelligence-enabled cancer diagnosis on medical imaging. Cancers 15(5):1492
87. Zhao J, Gao ZM (2022) The heterogeneous Aquila optimization algorithm. Math Biosci Eng 19:5867–5904
88. Baş E (2023) Binary Aquila Optimizer for 0–1 knapsack problems. Eng Appl Artif Intell 118:105592
89. Nadimi-Shahraki MH, Taghian S, Mirjalili S, Abualigah L (2022) Binary Aquila Optimizer for selecting effective features from medical data: a covid-19 case study. Mathematics 10(11):1929
90. Pashaei E (2022) Mutation-based binary Aquila Optimizer for gene selection in cancer classification. Comput Biol Chem 101:107767
91. Mohamed AA, Abualigah L, Alburaikan A, Khalifa HAEW (2023) AOEHO: a new hybrid data replication method in fog computing for IoT application. Sensors 23(4):2189
92. Wang S, Ma J, Li W, Khayatnezhad M, Rouyendegh BD (2022) An optimal configuration for hybrid SOFC, gas turbine, and proton exchange membrane electrolyzer using a developed Aquila Optimizer. Int J Hydrogen Energy 47(14):8943–8955
93. Ali MH, Salawudeen AT, Kamel S, Salau HB, Habil M, Shouran M (2022) Single-and multi-objective modified Aquila Optimizer for optimal multiple renewable energy resources in distribution network. Mathematics 10(12):2129
94. Abdel-Basset M, Mohamed R, Jameel M, Abouhawwash M (2023) Nutcracker optimizer: a novel nature-inspired metaheuristic algorithm for global optimization and engineering design problems. Knowl-Based Syst 262:110248
95. Abualigah L, Diabat A, Mirjalili S, Abd Elaziz M, Gandomi AH (2021) The arithmetic optimization algorithm. Comput Methods Appl Mech Eng 376:113609
96. Mirjalili S (2016) Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Comput Appl 27:1053–1073
97. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
98. Salawudeen AT, Mu'azu MB, Yusuf A, Adedokun AE (2021) A novel smell agent optimization (SAO): an extensive CEC study and engineering application. Knowl-Based Syst 232:107486
99. Khalid AM, Hosny KM, Mirjalili S (2022) COVIDOA: a novel evolutionary optimization algorithm based on coronavirus disease replication lifecycle. Neural Comput Appl 34(24):22465–22492
100. Arora S, Singh S (2019) Butterfly optimization algorithm: a novel approach for global optimization. Soft Comput 23:715–734
101. Seyyedabbasi A, Kiani F (2022) Sand Cat swarm optimization: a nature-inspired algorithm to solve global optimization problems. Eng Comput. https://doi.org/10.1007/s00366-022-01604-x
102. Ezugwu AE, Agushaka JO, Abualigah L, Mirjalili S, Gandomi AH (2022) Prairie dog optimization algorithm. Neural Comput Appl 34(22):20017–20065
103. Nirmalapriya G, Agalya V, Regunathan R, Ananth MBJ (2023) Fractional Aquila spider monkey optimization based deep learning network for classification of brain tumor. Biomed Signal Process Control 79:104017
104. Nirmalapriya G, Maram B, Lakshmanan R, Navaneethakrishnan M (2023) ASCA-squeeze net: Aquila sine cosine algorithm enabled hybrid deep learning networks for digital image forgery detection. Comput Secur 128:103155
105. Song D, Yan J, Zeng H, Deng X, Yang J, Qu X et al (2023) Topological optimization of an offshore-wind-farm power collection system based on a hybrid optimization methodology. J Marine Sci Eng 11(2):279
106. Mahajan S, Abualigah L, Pandit AK, Altalhi M (2022) Hybrid Aquila Optimizer with arithmetic optimization algorithm for global optimization tasks. Soft Comput 26(10):4863–4881
107. Zhang YJ, Yan YX, Zhao J, Gao ZM (2022) AOAAO: The hybrid algorithm of arithmetic optimization algorithm with Aquila Optimizer. IEEE Access 10:10907–10933
108. Ewees AA, Algamal ZY, Abualigah L, Al-qaness MA, Yousri D, Ghoniem RM, Abd Elaziz M (2022) A cox proportional-hazards model based on an improved Aquila Optimizer with whale optimization algorithm operators. Mathematics 10(8):1273
109. Ma C, Huang H, Fan Q, Wei J, Du Y, Gao W (2022) Grey wolf optimizer based on Aquila exploration method. Expert Syst Appl 205:117629
110. Abualigah L, Elaziz MA, Khodadadi N, Forestiero A, Jia H, Gandomi AH (2022) Aquila Optimizer based PSO swarm intelligence for IoT task scheduling application in cloud computing. Integrating meta-heuristics and machine learning for real-world optimization problems. Springer International Publishing, Cham, pp 481–497
111. Abualigah L, Almotairi KH (2022) Dynamic evolutionary data and text document clustering approach using improved Aquila Optimizer based arithmetic optimization algorithm and differential evolution. Neural Comput Appl 34(23):20939–20971
112. Yao J, Sha Y, Chen Y, Zhang G, Hu X, Bai G, Liu J (2022) IHSSAO: An improved hybrid Salp swarm algorithm and Aquila Optimizer for UAV path planning in complex terrain. Appl Sci 12(11):5634
113. . Akyol S (2022) A new hybrid method based on Aquila Optimizer and tangent search algorithm for global optimization. *J Ambient Intell Human Comput*, 1–21.
114. Utama DM, Primayesti MD (2022) A novel hybrid Aquila Optimizer for energy-efficient hybrid flow shop scheduling. Results Control Optim 9:100177
115. Natesan G, Konda S, de Prado RP, Wozniak M (2022) A hybrid mayfly-Aquila optimization algorithm based energy-efficient clustering routing protocol for wireless sensor networks. Sensors 22(17):6405
116. Al-qaness MA, Ewees AA, Elaziz MA, Samak AH (2022) Wind power forecasting using optimized dendritic neural model based on seagull optimization algorithm and Aquila Optimizer. Energies 15(24):9261
117. Xiao Y, Guo Y, Cui H, Wang Y, Li J, Zhang Y (2022) IHAOAVOA: An improved hybrid Aquila Optimizer and African vultures optimization algorithm for global optimization problems. Math Biosci Eng 19(11):10963–11017
118. Al-qaness MA, Ewees AA, Thanh HV, AlRassas AM, Abd Elaziz M (2022) An optimized neuro-fuzzy system using advance nature-inspired Aquila and Salp swarm algorithms for smart predictive residual and solubility carbon trapping efficiency in underground storage formations. J Energy Storage 56:106150
119. Wang Y, Xiao Y, Guo Y, Li J (2022) Dynamic chaotic opposition-based learning-driven hybrid Aquila Optimizer and artificial rabbits optimization algorithm: framework and applications. Processes 10(12):2703
120. Ait-Saadi A, Meraihi Y, Soukane A, Ramdane-Cherif A, Gabis AB (2022) A novel hybrid chaotic Aquila optimization algorithm with simulated annealing for unmanned aerial vehicles path planning. Comput Electr Eng 104:108461
121. Elaziz MA, Dahou A, El-Sappagh S, Mabrouk A, Gaber MM (2022) AHA-AO: artificial hummingbird algorithm with Aquila optimization for efficient feature selection in medical image classification. Appl Sci 12(19):9710
122. Zhang YJ, Zhao J, Gao ZM (2022). Hybridized improvement of the chaotic Harris Hawk optimization algorithm and Aquila Optimizer. In *International conference on electronic information*

engineering and computer communication (EIECC 2021), Vol. 12172. SPIE, , pp. 327–332
123. Gupta J, Pathak S, Kumar G (2022) Aquila coyote-tuned deep convolutional neural network for the classification of bare skinned images in websites. Int J Mach Learn Cybern 13(10):3239–3254
124. . Bousmaha, R. (2022). A novel hybrid Aquila Optimizer with Whale optimziation algorithm for global optimization, feature selection and optimizing SVM parameters. In *2022 4th international conference on pattern analysis and intelligent systems (PAIS)*. IEEE, pp. 1–8
125. Gupta J, Pathak S, Kumar G (2022) A hybrid optimization-tuned deep convolutional neural network for bare skinned image classification in websites. Multimedia Tools App 81(18):26283–26305
126. Wang TT, Chu SC, Hu CC, Jia HD, Pan JS (2022) Efficient network architecture search using hybrid optimizer. Entropy 24(5):656
127. Al-Qaness MA, Ewees AA, Thanh HV, AlRassas AM, Dahou A, Elaziz MA (2023) Predicting CO_2 trapping in deep saline aquifers using optimized long short-term memory. Environ Sci Pollut Res 30(12):33780–33794
128. Rathod V, Tiwari A, Kakde OG (2022) Aquila-eagle-based deep convolutional neural network for speech recognition using EEG signals. Int J Swarm Intell Res (IJSIR) 13(1):1–28
129. Wang S, Jia H, Abualigah L, Liu Q, Zheng R (2021) An improved hybrid Aquila Optimizer and Harris Hawks algorithm for solving industrial engineering optimization problems. Processes 9(9):1551
130. . Verma M, Sreejeth M, Singh M (2021) Application of hybrid metaheuristic technique to study influence of core material and core trench on performance of surface inset PMSM. *Arab J Sci Eng*, 1–17
131. Zhang Y, Yan Y, Zhao J, & Gao Z (2021). Chaotic map enabled algorithm hybridizing Hunger Games Search algorithm with Aquila Optimizer. In *ICMLCA 2021; 2nd international conference on machine learning and computer application*. VDE, pp. 1–5