Online-Journals.org

iJOE International Journal of Online and Biomedical Engineering

iJOE | eISSN: 2626-8493 | Vol. 19 No. 9 (2023) | OPEN ACCESS

https://doi.org/10.3991/ijoe.v19i09.39683

REVIEW

Metaheuristics: A Review of Algorithms

Haval Tariq Sadeeq(✉), Adnan Mohsin Abdulazeez

Information Technology Department, Technical College of Informatics-Akre, Duhok Polytechnic University, Duhok, Kurdistan Region of Iraq

haval.tariq@dpu.edu.krd

ABSTRACT

In science and engineering, many optimization tasks are difficult to solve, and the core concern these days is to apply metaheuristic (MH) algorithms to solve them. Metaheuristics have gained significant attention in recent years, with nature serving as the fundamental inspiration where self-organization property led to collective intelligence emerging from the behavior of a swarm of birds or colony of insects or more and more natural behavior. These swarms or colonies, even with extremely low individual competence, have the ability to accomplish many complicated activities that can be considered necessary for their existence. Accordingly, many MH algorithms have been developed based on natural phenomena. In this article, an analysis review of more than one hundred metaheuristics have been made. Further, the main contributions of this article are to give some vital insights about metaheuristics, presenting and proposing the general mathematical framework of MH algorithms and dividing it into a number of tasks with possible progress for each task. While there are still many open issues in this field, it is worth noting that there have been significant advancements in recent years. As a result, new algorithms are continuously being proposed to address these challenges.

KEYWORDS

metaheuristics, evolutionary algorithms, swarm intelligence, exploration, exploitation

1 INTRODUCTION

The term "optimization" refers to the process of determining the best values for various system characteristics in order to complete the system design at the lowest possible cost [1] [2]. In general, real-world artificial intelligence and machine learning applications are unconstrained or discrete [3]. For that reason, finding optimal solutions using conventional (classical) methods is difficult. It is worth noting that conventional optimization methods have a number of drawbacks and restrictions, such as convergence to local optima and an undefined search space [4]. Furthermore, they only have a single-based solution. Accordingly, various stochastic methods have been designed in recent years to address and overcome these flaws as well as improve the performance of various systems and minimize computation costs [5] [6] [7]. Thus, optimization problems can be found in any scientific subject [8] [9].

Sadeeq, H.T., Abdulazeez, A.M. (2023). Metaheuristics: A Review of Algorithms. *International Journal of Online and Biomedical Engineering (iJOE)*, 19(9), pp. 142–164. https://doi.org/10.3991/ijoe.v19i09.39683

Article submitted 2023-03-18. Resubmitted 2023-04-12. Final acceptance 2023-04-13. Final version published as submitted by the authors.

Stochastic algorithms optimize optimization problems randomly. Therefore, they intrinsically benefit from higher local optima avoidance compared to conventional optimization algorithms [10] [11]. Stochastic algorithms can be heuristic or metaheuristic. In general, metaheuristic algorithms outperform heuristic algorithms. According to [12] "A metaheuristic is formally defined as an iterative generation process that guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space. Learning strategies are used to structure information in order to find efficient, near-optimal solutions."

Overall, metaheuristics are problem-independent algorithms used to find approximate optimal solutions to complicated and highly nonlinear optimization problems that no deterministic approach is able to handle in an acceptable amount of time [13].

In general, all of these MH algorithms consist of two main components that share certain characteristics, such as the search process, which is divided into two phases: intensification (exploitation) and diversification (exploration).

Exploitation involves using information obtained from the problem at hand to generate new solutions that are better than existing ones. However, this process tends to be local in nature, and the information used is also local, such as gradients, making it suitable for local search. For instance, hill-climbing employs derivative information to guide the search procedure, with new steps always attempting to ascend the local gradient. Exploitation often leads to high convergence rates, but it may get trapped in a local optimum, as the final solution point heavily relies on the starting point [14].

On the contrary, exploration enables more efficient exploration of the search space, producing diverse solutions that are distant from current ones, thereby facilitating global-scale search. While the primary benefit of exploration is avoiding getting trapped in local modes and increasing the accessibility of global optimality, it has drawbacks such as slow convergence and excessive computational effort, as generating many new solutions that are far from the global optimum can be wasteful [15].

Thus, a proper equilibrium must be reached for an algorithm to perform well. Overemphasis on exploration at the expense of exploitation can cause the search path to wonder, resulting in slow convergence. Conversely, overemphasis on exploitation at the cost of exploration may lead to quicker convergence but lower chances of locating the true global optimum.

To cope with the above issues, some reviews and surveys about MH algorithms in the literature have been selected. Some MH algorithms are shown in [16] along with an analysis of how they relate to self-organization. Some key problems have to be addressed in the future, like the problem of convergence speed. Therefore, to overcome this problem, the optimization algorithms could be hybridized (combined), according to the authors.

Some different examples of SI-based algorithms were highlighted by [17], and after that, their key components and traits were studied. Furthermore, the authors have underlined some significant concerns and provided some possible solutions. The authors expect that a wider range of optimization problems will be solved by advancing the state of research on swarm intelligence and nature-inspired computation.

Authors in [18] further classified the swarm-based algorithms into animal based and insect-based algorithms. Despite the fact that these algorithms have been used in a wide range of different topics, the authors recommend using them in business and social science applications. The authors also discussed several less commonly known algorithms, such as the glow-worm. However, it should be noted that these algorithms have yet to gain widespread adoption in many application-oriented fields.

The authors in [19] had classified the MH algorithms into new-generation algorithms in the authors' terms or old-generation (i.e., classical algorithms). The key elements of the new generation of algorithms have been described in the article, as well. It was observed that the majority of MH algorithms of the new generation had a significant set of parameters.

The authors in [20] analyze the latest literature on the application of several MH algorithms and data envelopment analysis (DEA) to optimization problems. The application of MH algorithms in DEA is discussed, as are their applications, the scope of activity, and the ideal solution obtained from combining these two methods.

In order to thoroughly examine the similarities and differences between multiple MH algorithms, [21] provided an overview of the challenges and future research prospects in the field. This was achieved by formalizing a description of eleven MH algorithms, which were then tested on benchmark optimization functions to assess their accuracy, robustness, and sensitivity to parameters.

Considering the aforementioned overviews of MH algorithms, there is still a need to present the most recently developed metaheuristics, defining their working mechanisms, highlighting their major challenges, and predicting the future of applicable real-world problems.

Contributions of this article are stated as follows:

- We present a comprehensive list of more than one hundred MHAs, and make a preliminary analysis of their basic information, which can provide a panoramic view for MHAs study.
- Showing in detail the methodology of MHAs by defining their general framework and dividing them into a number of tasks.
- We outline the problems and potential solutions facing the entire field of MHAs, which might serve as a reference point for future MHAs research.

The review article consists of six sections. The remaining sections include Section 2 which presents the metaheuristics. Section 3 explains in details the design and the general framework of MH algorithms. Some newly developed MH algorithms are explained in short in section 4. Section 5 presents the advantages, disadvantages, and future directions of MH algorithms. Finally, the discussion of the review is displayed in section 6.

2 METAHEURISTICS

In the literature, MH algorithms can be classed as: population-based versus single point search, nature inspired versus non-nature-inspired, static objective versus dynamic objective function, and memory use opposed to memory less methods, and various neighborhood versus single neighborhood. Consequently, any algorithm can be categorized into one of the following groups depending on its source of inspiration:

2.1 Evolutionary-based algorithms (EAs)

Using the principle of survival of the fittest, this is the most prevalent and oldest sort of metaheuristic, which imitates the evolutionary behavior of organisms in nature. EAs begin with some random solutions that, over time and iterations, improve the fitness value by creating new solutions and discarding the worst ones. Since these algorithms do not rely on any presumptions about the fundamental fitness landscape, they often succeed in locating optimal or near-optimal solutions. Table 1 provides a list of EAs.

Table 1. Evolutionary-based algorithms

Algorithm	Ref.	Inspiration	Year
Genetic Programming (GP)	[22]	Natural selection	1992
Cultural Algorithm (CA)	[23]	Changes in culture over time	1994
Differential Evolution (DE)	[24]	The evolutionary theory proposed by Darwin	1997
Grammatical Evolution (GE)	[25]	Evolutionary process in living organisms	1998
Differential Search Algorithm (DSA)	[26]	Movement of organisms	2011
Backtracking Search Algorithm (BSA)	[27]	Memory-based evolutionary principles	2013
Stochastic Fractal Search (SFS)	[28]	The implementation of fractal concepts	2014
Invasive tumor growth optimization algorithm (ITGO)	[29]	Inspired by invasive tumor growth.	2015
Tree growth algorithm (TGA)	[30]	Trees competition for acquiring light and foods	2018
Wildebeests Herd Optimization (WHO)	[31]	Herding behavior of Wildebeest	2019
Barnacles Mating Optimizer (BMO)	[32]	Mating behavior of barnacles	2020
Learner Performance based Behavior (LPB)	[33]	The procedure of admitting high school graduates into various university disciplines.	2021

2.2 Swarm intelligence-based (SI) algorithms

This is the most prominent class of metaheuristics that models the cooperative, adaptive, cognitive, and concerted gregarious behavior of natural flocks or communities. These communities include schools of fish, flocks of birds, flocks of mammals, colonies of insects like bees, and numerous flocks of other species of organisms. Researchers are becoming more interested in this sort of metaheuristic, which competes heavily with EAs. This category has a wide range of algorithms, with some of the most well-known examples presented in Table 2.

2.3 Natural science-based algorithms (NSAs)

NSAs imitate specific chemical principles or physical processes (e.g., gravity, electrical charges, ion motion, river systems, etc.). Table 3 contains some of the most prominent examples of NSAs.

2.4 Human-based algorithms (HBAs)

Human behavior, including non-physical activities like thinking and associated societal perceptions, falls under this scope. Researchers' interest in this category of algorithms has increased over the past decade and continues to grow. Different methods have been taken in related works; here, we include the most often used ones in Table 4.

Table 2. Swarm-based algorithms

Algorithm	Ref.	Inspiration	Year
Ant Colony Optimization (ACO)	[34]	Colonies of Ants	1991
Particle Swarm Optimization (PSO)	[35]	Bird Flock	1995
Marriage in Honey Bees Optimization Algorithm (MBO)	[36]	Behaviors of honey bees	2001
Artificial Fish Swarm Algorithm (AFSA)	[37]	Swarm of fishes	2003
Artificial Bee Colony (ABC)	[38]	Foraging behaviors of honey bees	2006
Cat Swarm Optimization (CSO)	[39]	Tracing and relaxing behaviors of cats	2006
Firefly Algorithm (FA)	[40]	Flashing techniques of fireflies	2008
Cuckoo Search (CS) algorithm	[41]	Inspired by cuckoo bird reproduction	2009
Bat Algorithm (BA)	[42]	Microbat behaviors	2010
Krill herd (KH)	[43]	Herding behavior of krill	2012
Grey Wolf Optimizer (GWO)	[44]	Hunting preys	2014
Cuttlefish Algorithm (CFA)	[45]	Color-changing cuttlefish behavior	2014
Moth-Flame Optimization (MFO) algorithm	[46]	Transverse orientation of moths	2015
Ant Lion Optimizer (ALO)	[47]	Hunting activities of antlions	2015
Whale Optimization Algorithm (WOA)	[48]	humpback whales' social behavior	2016
Dragonfly Algorithm (DA)	[49]	Dragonfly behaviors of organized swarms	2016
Crow Search Algorithm (CSA)	[50]	Crows' intellectual behavior	2016
Lion Optimization Algorithm (LOA)	[51]	The specialized, unique lifestyle of lions	2016
Salp Swarm Algorithm (SSA)	[52]	Salps foraging in oceans	2017
Grasshopper Optimization Algorithm (GOA)	[53]	Grasshopper swarm behaviors	2017
Spotted Hyena Optimizer (SHO)	[54]	Spotted hyena behaviors	2017
Emperor Penguin Optimizer (EPO)	[55]	Huddling behavior of emperor penguins	2018
Coyote Optimization Algorithm (COA)	[56]	Social organization of the coyotes	2018
Harris Hawks Optimizer (HHO)	[57]	Cooperative hunting behavior of Harris' hawks	2019
Squirrel search algorithm (SSA)	[58]	Gliding and foraging behaviors of squirrels.	2019

(Continued)

Table 2. Swarm-based algorithms *(Continued)*

Algorithm	Ref.	Inspiration	Year
SailFish Optimizer (SFO)	[59]	Hunting sailfish	2019
Sea Lion Optimization (SLnO) algorithm	[60]	The habits of sea lions when they are hunting	2019
Emperor Penguins Colony (EPC)	[61]	The emperor penguin's behavior	2019
Bald Eagle Search (BES) Algorithm	[62]	Behavior and hunting strategies of bald eagles	2020
Red deer algorithm (RDA)	[63]	Mating behavior of Scottish red deer	2020
Sparrow search algorithm (SSA)	[64]	Sparrows' behavior's	2020
Chimp Optimization Algorithm (ChOA)	[65]	Sexual motivation of chimps	2020
Mayfly Algorithm (MA)	[66]	Flight and mating of mayflies	2020
Aquila Optimizer (AO)	[67]	Hunting behavior of aquila	2021
Seagull Optimization Algorithm (SOA)	[68]	Feeding and hunting behavior	2021
Red Fox Optimization Algorithm (RFO)	[69]	Red fox habits	2021
Golden Eagle Optimizer (GEO)	[70]	Intelligence behavior of golden eagles	2021
Beluga whale optimization (BWO)	[71]	Behaviors of beluga whales	2022
Giant Trevally Optimizer (GTO)	[72]	Hunting strategies of giant trevallies	2022
Termite life cycle optimizer (TLCO)	[73]	Life cycle of a termite colony	2023
Nutcracker Optimization Algorithm (NOA)	[74]	Intelligent behaviors of nutcrackers	2023

Table 3. Natural Science-based algorithms

Algorithm	Ref.	Inspiration	Year
Simulated Annealing (SA)	[75]	Annealing procedure	1983
Variable Neighborhood Search (VNS)	[76]	The concept of neighborhood modifications	1995
Big Bang-Big Crunch (BB-BC)	[77]	Big Bang and Big Crunch theory	2005
Central Force Optimization (CFO)	[78]	Metaphor of gravitational kinematics	2007
Intelligent Water Drops (IWD)	[79]	From observing flowing drops of water in a river	2007
Slime Mold Algorithm (SMA)	[80]	The life style of slime mold	2008
Gravitational Search Algorithm (GSA)	[81]	Newtonian gravity theory in physics	2009
Charged System Search (CSS)	[82]	Based on fundamental mechanical and physical concepts	2010
Electro-Magnetism Optimization (EMO)	[83]	Fundamentals of Electro-magnetism	2011
Water Cycle Algorithm (WCA)	[84]	mechanisms of water cycle in nature	2012
Black Hole Algorithm (BHA)	[85]	Gravitational force of the black hole	2012

(Continued)

Table 3. Natural Science-based algorithms *(Continued)*

Algorithm	Ref.	Inspiration	Year
Mine Blast Algorithm (MBA)	[86]	Mine bomb detonation	2013
Colliding Bodies Optimization (CBO)	[87]	Motivated by the two-body collision laws in a single dimension	2014
Lightning Search Algorithm (LSA)	[88]	Lightning occurrence	2015
Multi-Verse Optimizer (MVO)	[89]	Multiple universes interacting	2016
Thermal Exchange Optimization (TEO)	[90]	The concepts of cooling	2017
Find-Fix-Finish-Exploit-Analyze (F3EA)	[91]	Installations or selecting objects for destruction in warfare.	2018
Henry Gas Solubility Optimization (HGSO)	[92]	Based on Henry gas law	2019
Equilibrium Optimizer (EO)	[93]	Physics dynamic mass balance	2020
Turbulent flow of water-based optimization (TFWO)	[94]	Whirlpools made by turbulent flow of water	2020
Archimedes Optimization Algorithm (AOA)	[95]	Archimedes theory	2021
Lichtenberg Algorithm (LA)	[96]	Lightning storms	2021
Special Relativity Search (SRS)	[97]	Particles movement and interaction in an electromagnetic field	2022
Energy Valley Optimizer (EVO)	[98]	Stability and different modes of particle decay	2023
Young's Double-Slit Experiment (YDSE) optimizer	[99]	Derived from Young's double-slit experiment	2023

Table 4. Human-based algorithms

Algorithm	Ref.	Inspiration	Year
Society Civilization Algorithm (SCA)	[100]	Leadership characteristics	2003
Seeker Optimization Algorithm (SOA)	[101]	The way people search for things at random	2006
Imperialist Competitive Algorithm (ICA)	[102]	Human social evolution	2007
Human-Inspired Algorithm (HIA)	[103]	People's knowledge and intelligence	2009
Social Emotional Optimization Algorithm (SEOA)	[104]	Societal norms and routines	2010
Brain Storm Optimization (BSO)	[105]	The process of brainstorming	2011
Teaching-Learning-Based Optimization (TLBO)	[106]	Teaching and learning process	2011
Anarchic Society Optimization (ASO)	[107]	Social and personal traits	2012
Human Mental Search (HMS)	[108]	Exploratory technique for navigating the bid space of online auctions	2017
Political optimizer (PO)	[109]	Complex nature of the political process	2020
Poor and rich optimization algorithm (PRO)	[110]	The attempts of the poor and the rich people to enhance their financial circumstances.	2020

(Continued)

Table 4. Human-based algorithms *(Continued)*

Algorithm	Ref.	Inspiration	Year
Group Teaching Optimization Algorithm (GTOA)	[111]	Human group teaching techniques	2020
Gaining Sharing Knowledge based Algorithm (GSK)	[112]	Knowledge gained and shared through the course of a lifetime	2020
Coronavirus Herd Immunity Optimizer (CHIO)	[113]	Implementing the theory of herd immunity to fight the COVID-19	2021
Ali Baba and the Forty Thieves (AFT)	[114]	The tale of Ali Baba and the forty thieves	2021
War Strategy Optimization Algorithm (WSO)	[115]	The strategic positioning of military forces during wars	2022

It is worth noting that other MHAs have been proposed in recent years that do not fall under the categories listed above such as:

Sports-inspired algorithms: Algorithms inspired by sports imitate the procedures, regulations, and activities of numerous sports, mainly football. League championship algorithm [116], world cup optimization [117], and football game algorithm [118] are sports-inspired MHAs.

Music-based algorithms: algorithms that draw inspiration from music, such as melody search [119], and harmony search [120].

Plant-based algorithms: Plant-based algorithms simulate the intelligence of plants mimics the plant intelligence. Some well-known plant-based algorithms include the path planning algorithm [121], rooted tree optimization algorithm [122], and the flower pollination algorithm [123].

Mathematics-based algorithms: The characteristics of mathematics serve as the basis for mathematically-based algorithms. This class contain some famous metaheuristics such as Sine Cosine Algorithm [124], and Arithmetic optimization algorithm [125].

Water-behavior algorithms: Inspired by the intelligence movements of water, algorithms based on water behavior have been developed. Some examples of this class include, water flow algorithm [126], and circular water wave algorithm [127].

3 METHODOLOGY OF METAHEURISTICS

In order to recognize and explain in depth the design process and the methodology of MHAs, the general framework of the metaheuristic algorithms is presented in this section. While it is undoubtedly a challenging task to explain the primary steps or tasks involved in MH algorithms, we will endeavor to do so by reviewing past and potential future improvements aimed at enhancing their performance. Figure 1 illustrates the general framework of the MH algorithms, showing that almost all MH algorithms use the same general framework despite employing diverse search methodologies. However, the basic tasks involved in MH algorithms are stated below:

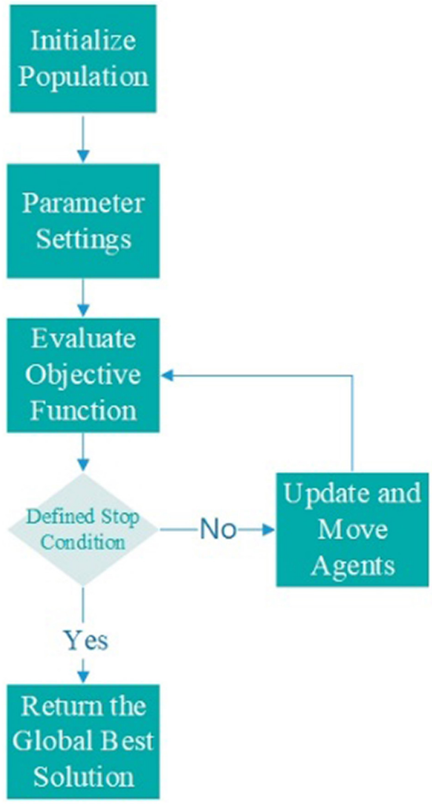

Fig. 1. General framework of the MH algorithms

3.1 Task 1, initialization

This step normally takes place randomly, influencing the diversity and convergence during the operation of the algorithms. As the researchers become aware of the significance of initialization, they look for various methods that can enhance the population diversity, and ultimately, the algorithms' optimal solutions should be unaffected by their starting decisions. In many cases, initialization has been done by employing uniform distributions, with results that are applicable to nearly all metaheuristic optimization techniques. In practice, however, uniform distributions may not be suitable for all applications. There are also some other commonly used initialization techniques, such as chaotic initialization, sequence-based deterministic initialization, opposition-based learning, and Latin hypercube sampling [128].

3.2 Task 2, parameter settings

A considerable impact on the solution quality is caused by the initial parameter values. As most metaheuristic algorithms are parameterized, performance metrics such as processing time and result quality are evaluated based on the optimal parameter set. The process of discovering the best parameter set is known as parameter optimization or tuning, which can be accomplished through trial-and-error or deep learning techniques. It is worth mentioning that task 2 includes the maximum number of iterations, error rate, and population size.

3.3 Task 3, evaluate objective function

In optimization problems, test functions represent the objective function, and metaheuristic algorithms are often evaluated using benchmark test functions in publications. The effectiveness of good algorithms in solving numerical optimization problems is a crucial factor. However, due to the absence of a standard or globally accepted test suite, researchers tend to use different test functions, which makes it challenging to evaluate the overall robustness of a proposed metaheuristic algorithm accurately.

It is worth mentioning that utilizing a set of functions with diverse characteristics, such as unimodal and multimodal functions, along with incorporating sensitivity parameters like increasing the number of dimensions or utilizing some modern CEC functions, can be an effective approach for assessing the performance of newly developed algorithms.

3.4 Task 4, stop conditions

The primary disadvantage of the majority of metaheuristics is their lack of effective termination criteria. The majority of implementations of such algorithms terminate after a specified number of iterations without progress toward the optimal solution value. In some instances, the algorithm may perform an excessive number of iterations that are unnecessary. In other cases, the algorithm may terminate just prior to performing an iteration that might result in a better, or even optimum, output.

Another stopping criteria that is also used is the number of function evaluations (FEs) that can be calculated as follows:

$$EFs = Pop \times Max \tag{1}$$

where *Pop* is the population size and *Max* is the maximum number of iterations.

The best moment to snap the optimal solution in MH algorithms is still an open problem. However, to our knowledge, combining the maximum number of iterations with a specified tolerance or error rate is one of the most effective stopping criteria.

3.5 Task 5, update and move agents

This task can be divided into two subtasks: update and move. As it is clear from the framework, if the predefined stop conditions are not met, then the alternative task will be updated, and the move will be performed. The term "update" does not apply to the limited sense of the word; rather, it describes the meta operation of MH algorithms that enables metaheuristics to reach (iteration I+1) where better results can be found. This subtask is updated each time an iteration fitness assessment is conducted at iteration (I). While most algorithms do the moving subtask randomly so as to expand the range of search space the optimization algorithm explores. It is essential to note that repeating iterations is not really a task for its own sake; rather, it is the repetition of the update and move agent tasks until the set stop criteria are met.

4 SOME RECENT METAHEURISTICS

In this section, to further investigate the working mechanisms of metaheuristics, some newly developed algorithms are presented, and for each main category, one algorithm is chosen. Learner performance-based behavior algorithm is chosen for evolutionary algorithms, the giant trevally optimizer for swarm intelligence, city council evolution for human-based algorithms, and special relativity search for science-based algorithms.

4.1 Lerner performance-based behavior algorithm (LPB)

Learner performance-based behavior algorithm (LPB) [33] influenced by the college admissions process for high school graduates. Transferring high school graduates to universities begins with a class of high school graduates. Some of these students' applications to departments are accepted, while others are rejected based on their grade point averages (GPA). The minimal GPA required by each department is determined by the departments. Students are accepted into a certain department if their GPA is at least as high as the minimum GPA required for that department. On the other hand, students who have a better grade point average tend to be accepted first. In order to randomly split up a certain number of students, the algorithm makes use of the division probability (*dp*) operator. Using equation (2), a subset of the population can be isolated.

$$S = Pop \times dp \tag{2}$$

Where: S represents the number of members isolated from the core population, Pop represents the population size, and dp can take values between [0.1 and 0.9].

Individuals in the separated group will be split into two subpopulations based on their fitness scores, good or bad. Members of a good population have higher fitness levels, as measured by GPA, whereas those in a bad population have lower fitness levels. A good population has members with higher fitness (GPA) and a bad population has members with lower fitness. Then, the fitness will be computed for all members and the members will be divided on groups based on their fitness according to equations 3, 4, and 5.

$$x \in BP,\ if\ f(x) \leq Max\ (BP) \tag{3}$$

$$x \in PP,\ if\ f(x) > Max\ (GP) \tag{4}$$

$$x \in GP,\ if\ f(x) \leq Max\ (GP) \tag{5}$$

Where BP, PP, and GP refers to bad population, perfect population and good population respectively.

It should be noted that the priority is given to the individuals in the perfect population to go through the optimization process first, followed by the individuals in the good population and the bad population. In addition, when students attend college, their studying habits are influenced by the studying habits of their peers. In order to demonstrate this in the algorithm, a crossover operator was used. Furthermore, students with an acceptable level of metacognition are superior to those with an insufficient level. To demonstrate this, the genetic algorithm's mutation operator was used.

4.2 Giant trevally optimizer (GTO)

Giant Trevally Optimizer (GTO) [72] inspired by the strategies of giant trevally when hunting seabirds (sooty terns). The mathematical model of GTO is divided into three main steps. In the first step, which is called "extensive search" the foraging movement patterns of giant trevallies are simulated using equation (6):

$$X(t+1) = Best_p \times R + ((Maximm - Minimum) \times R + Minimum) \times Levy(Dim) \tag{6}$$

where $X(t + 1)$ is the next-iteration giant trevally position vector, $Best_p$ denotes the current search area chosen by giant trevallies based on the best position found during their previous search, R is a random selected number in the range [0, 1]. While $Levy(Dim)$ is the Levy flight.

In the second step, "choosing area" the giant trevallies choose the appropriate area in terms of food where they can hunt for prey. Equation (7) simulates this behavior mathematically.

$$X(t+1) = Best_p \times \mathcal{A} \times R + Mean\_Info - Xi(t) \times R \tag{7}$$

where $\mathcal{A}$ is a position-change-controlling parameter that accept values from 0.3 to 0.4. $Xi(t)$ is the location of the giant trevally i, at time t (current iteration). Meanwhile, *Mean_Info*, which relates to the mean, indicates that these giant trevallies have utilized all accessible data from the earlier points.

Then, the trevally begins to pursue the bird in the final step "Attacking". In order to catch its prey, trevally will jump from the water or even snatches the prey from the surface of the water.

In order to simulate the behavior of a giant trevally during chasing and attacking the prey, it was assumed in GTO that trevallies are affected by visual distortion, which is mainly caused by the refraction of light.

It should be underlined that the giant trevally plays the role of an observer while the bird takes on the role of an object. Consequently, the apparent height of the bird, is always seen to be higher than its true height.

Finally, how giant trevally act when chasing and jumping out of the water is simulated using (8).

$$X(t+1) = \mathcal{L} + \mathcal{V} + \mathcal{H} \tag{8}$$

Where $\mathcal{L}$ is the launch speed and has been used to simulate chasing the bird.

4.3 Special relativity search (SRS)

The behavior of particles in an electromagnetic field serves as the conceptual basis for this algorithm [97]. Particles move along a circular path due to the magnetic force between them, which acts perpendicular to the direction of motion of charged particles and the magnetic field. By combining the effects of length contraction and time dilation, the SRS main step equation can be derived. Particles with charges are part of the initial population that is formed at random, and their charge is established by their fitness.

Using the Euclidean norm, we can calculate the distance between X_i and X_j in the magnetic field.

$$D_{ij}(t) = (X_i(t) - X_j(t)) \tag{9}$$

where D_{ij} is the separate distance between particles X_i and X_j at iteration time t.

Then, using Eq (10) which accounts for the fitness of the particles themselves, we can determine the charge of Q_i and Q_j particles.

$$Q_i(t) = Q_j(t) = Fit_i(t) - Worst(t)/Global(t) - Worst(t) \tag{10}$$

where Fit_i is the fitness value of particles ith at iteration time t, *Global* and *Worst* is the lowest and highest values of the objective function at iteration time t, respectively.

Finally, the velocity and new position of the particle are obtained using Eq. (11).

$$X_{ij}(t+1) = \beta^2 X_j(t) + V_j(t)\sqrt{1-\beta^2} + X_j(t)\sqrt{1-\beta^2} \tag{11}$$

Where β is a random number between [0, 1] that represents the ratio of the particle velocity to the speed of light ($3e + 8m/s$), $\beta = v/c$.

4.4 City councils evolution (CCE)

In light of how domestic governments have developed over time. The motivation for this algorithm, comes from the fact that the city council holds a great deal of power in a city, including the ability to choose the mayor and other executive officers [129]. In essence, CCE uses the competitive nature of council membership as a means to choose its best members and even its CEO.

Councils Tree (CT) is a hierarchical structure similar to a tree that can be used to model the development of city councils. There are three stages to this modeling process. In the root node of the CT (i.e., at level/depth 1) resides the supreme council's boss. In the second stage, the members of the Supreme Council have taken up residence as leaf nodes off the root (i.e., level 2). The higher tiers of the CT have council members from other locations, is the last stage.

To compensate for the influence of council heads, CCE use a modified version of the arithmetic crossover known as improve to boost members productivity.

Algorithm 1 is then used to convert the current population.

Algorithm 1. CCE make new population steps

Input: *N*: the size of the population, *crN*: the number of PICs, *C*: a two-dimensional array with *N* rows and *crN* columns, *fit*: an array with size *N*, *d*: the member number of each council.
Output: a Councils Tree (CT).

```
for j = ⌊(N+1)/d⌋ downto 1 do
  while d*j+1 <= N do
    k = findMaxIndex (C, fit, d*j+2-d, d*j+1);
    if fit[j] > fit[k] then break;
    swap (C[j], C[k]); swap (fit[j], fit[k]);
    j = k;
  end while
end for
return C;
```

5 DISCUSSIONS

MHAs have been extensively addressed in the field of stochastic optimization in recent years. As a result, numerous related methods have emerged. Along with the promising performances metaheuristics has realized, the research literature has also indicated some disadvantages, as well as inherent trends. These are described below.

5.1 Advantages

- Metaheuristics are easy to be to implemented and the optimization problem can be solved without prior knowledge or ground truth.
- MH algorithms can be applied in a wide variety of problems in different fields.
- One commonly recognized advantage of metaheuristic algorithms is their ability to solve complex problems within reasonable timeframes, whilst exact algorithms may fail due to time constraints.
- The most fundamental aspect of MH algorithms is their flexible population size. Therefore, as long as the population is neither too small nor too large, their control mechanisms are not overly dependent on it.
- Global behavior can emerge from the interactions of some relatively simple individuals functioning independently at the local level.

5.2 Disadvantages

- In solving large-scale optimization problems, the evaluation of fitness function can become a significant computational bottleneck if its complexity is high.
- Without any sort of centralized control, MH algorithms run the risk of being stuck or prematurely converging to a local optimum. Hence, it is necessary to develop adaptive mechanisms to ensure continuous exploration and exploitation of the search space.
- MH algorithms tend to be time-consuming techniques that are influenced by factors such as the number of dimensions and the required number of iterations.
- There are no uniform structures or templates for the known algorithms such as PSO, FA, and GWA in order for the new introduced algorithms to be compared with.

5.3 Future research directions

- Metaheuristic parameters must be fine-tuned for each optimization problem in order to produce good results in a reasonable amount of time. However, to overcome this limitation, research has progressed. Several metaheuristic algorithms, such as GWA, SSO, SOS, and TLBO, aim to minimize the number of input parameters.
- Hardware advancements in the application domain.
- The emerging technology in this area is hybrid metaheuristic algorithms. Hybrid/hyper-heuristic algorithms are typically designed to achieve higher solution quality than conventional metaheuristic algorithms.

- In this context, it is also important to note that incorporating strategies such as chaotic maps and Levy flights can enhance the overall performance of the algorithm.
- Almost all modern metaheuristics are mathematically simulated in a way to perform exploration step then exploitation step. It is always recommended to keep a good tradeoff between both of them. However, this procedure does not necessarily prevent the occurrence of local optima stagnation due to the stochastic nature of metaheuristics. To mitigate this issue, researchers may consider incorporating an "explore again" step in addition to the exploration and exploitation steps. This extra step can help prevent the algorithm from becoming trapped in local optima, particularly when dealing with real-world optimization problems.

6 CONCLUSIONS

In recent decades, MH algorithms have been recognized as effective global optimization techniques. There have been numerous MH optimization algorithms presented and successfully applied to various types of problems. While these algorithms were first introduced in pure science and engineering (e.g., computer science and engineering), now these kinds of algorithms can be used to optimize different issues with different real parameters.

This article presents a comprehensive survey of MH algorithms, which reduces the gap in this research field. One of the objectives of this article is to give some insights and thoughts about MH algorithms, and this has been done by providing a detailed explanation of the general framework of MH algorithms and dividing the flowchart into a number of tasks. Furthermore, possible enhancements and suggestions are made for each task, such as initialization, objective function, and stop conditions.

The success and popularity of the MH algorithms indicate that the large number of studies devoted to the development of new MH algorithms will continue to rise in the future. These attempts will continue until some broad guidelines and general rules are established in this field, at which point flaws can be recognized and metaheuristic evaluations can be conducted more accurately.

According to the no-free lunch theory, none of the metaheuristic algorithms can be the ideal algorithm for all problems, unless they are developed specifically for that problem. This fact can motivate and prompt researchers in a variety of fields to examine the performance of new algorithms. After all, it was concluded that there is one concrete fact: due to the lack of the most efficient method for all types of search and optimization problems, novel metaheuristic algorithms are still being proposed.

7 REFERENCES

[1] H. Sadeeq and A. M. Abdulazeez, "Hardware implementation of firefly optimization algorithm using FPGAS," *ICOASE 2018 – Int. Conf. Adv. Sci. Eng.*, pp. 30–35, 2018, https://doi.org/10.1109/ICOASE.2018.8548822

[2] S. M. Almufti, R. B. Marqas, P. S. Othman, and A. B. Sallow, "Single-based and population-based metaheuristics for solving np-hard problems," *Iraqi J. Sci.*, vol. 62, no. 5, pp. 1710–1720, 2021, https://doi.org/10.24996/10.24996/ijs.2021.62.5.34

[3] H. S. Yahia and A. S. Mohammed, "Path planning optimization in unmanned aerial vehicles using meta-heuristic algorithms: A systematic review," *Environ. Monit. Assess.*, vol. 195, no. 1, 2023, https://doi.org/10.1007/s10661-022-10590-y

[4] A. S. Issa, Y. H. Ali, and T. A. Rashid, "An efficient hybrid classification approach for COVID-19 based on Harris Hawks optimization and Salp Swarm optimization," *Int. J. online Biomed. Eng.*, vol. 18, no. 13, pp. 113–130, 2022, https://doi.org/10.3991/ijoe.v18i13.33195

[5] H. T. Sadeeq and A. M. Abdulazeez, "Improved Northern Goshawk Optimization Algorithm for Global Optimization," in *2022 4th International Conference on Advanced Science and Engineering (ICOASE)*, 2022, pp. 89–94, https://doi.org/10.1109/ICOASE56293.2022.10075576

[6] H. Sadeeq, A. Abdulazeez, N. Kako, and A. Abrahim, "A novel hybrid bird mating optimizer with differential evolution for engineering design optimization problems," *Lect. Notes Data Eng. Commun. Technol.*, vol. 5, pp. 522–534, 2018, https://doi.org/10.1007/978-3-319-59427-9_55

[7] H. T. Sadeeq, A. M. Abdulazeez, N. A. Kako, D. A. Zebari, and D. Q. Zeebaree, "A New Hybrid Method for Global Optimization Based on the Bird Mating Optimizer and the Differential Evolution," *Proc. 7th Int. Eng. Conf. "Research Innov. Amid Glob. Pandemic", IEC 2021*, pp. 54–60, 2021, https://doi.org/10.1109/IEC52205.2021.9476147

[8] B. M. Abed and W. M. Jasim, "Multi objective optimization algorithms for mobile robot path planning: A survey," *Int. J. online Biomed. Eng.*, vol. 18, no. 15, pp. 160–177, 2022, https://doi.org/10.3991/ijoe.v18i15.34397

[9] S. Kochar and R. Garg, "Spectrum sensing for cognitive radio using genetic algorithm," *Int. J. Online Eng.*, vol. 14, no. 9, pp. 190–199, 2018, https://doi.org/10.3991/ijoe.v14i09.9064

[10] A. Sabry Eesa, A. Mohsin Abdulazeez, and Z. Orman, "Cuttlefish algorithm – A novel bio-inspired," *Int. J. Sci. Eng. Res.*, vol. 4, no. 9, pp. 1978–1986, 2013.

[11] A. S. Eesa, A. Mohsin, A. Brifcani, and Z. Orman, "A new tool for global optimization problems- Cuttlefish algorithm," 2014.

[12] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: Overview and conceptual comparison," *ACM Comput. Surv.*, vol. 35, no. 3, pp. 268–308, 2003, https://doi.org/10.1145/937503.937505

[13] X.-S. Yang, *Nature-inspired metaheuristic algorithms*. 2010.

[14] J. Xu and J. Zhang, "Exploration-exploitation tradeoffs in metaheuristics: Survey and analysis," *Proc. 33rd Chinese Control Conf. CCC 2014*, pp. 8633–8638, 2014, https://doi.org/10.1109/ChiCC.2014.6896450

[15] K. Hussain, M. N. M. Salleh, S. Cheng, and Y. Shi, "On the exploration and exploitation in popular swarm-based metaheuristic algorithms," *Neural Comput. Appl.*, vol. 31, no. 11, pp. 7665–7683, 2019, https://doi.org/10.1007/s00521-018-3592-0

[16] A. Darwish, "Bio-inspired computing: Algorithms review, deep analysis, and the scope of applications," *Futur. Comput. Informatics J.*, vol. 3, no. 2, pp. 231–246, 2018, https://doi.org/10.1016/j.fcij.2018.06.001

[17] X. S. Yang, S. Deb, Y. X. Zhao, S. Fong, and X. He, "Swarm intelligence: Past, present and future," *Soft Comput.*, vol. 22, no. 18, pp. 5923–5933, 2018, https://doi.org/10.1007/s00500-017-2810-5

[18] A. Chakraborty and A. K. Kar, "Swarm intelligence: A review of algorithms," *Model. Optim. Sci. Technol.*, vol. 10, pp. 475–494, 2017, https://doi.org/10.1007/978-3-319-50920-4_19

[19] T. Dokeroglu, E. Sevinc, T. Kucukyilmaz, and A. Cosar, "A survey on new generation metaheuristic algorithms," *Comput. Ind. Eng.*, vol. 137, 2019, https://doi.org/10.1016/j.cie.2019.106040

[20] M. Vaez-Ghasemi, Z. Moghaddas, H. Askari, and F. Valizadeh, "A review of meta-heuristic algorithms and data envelopment analysis," *Iran. J. Optim.*, vol. 12, no. 1, pp. 115–137, 2020.

[21] Z. Wang, C. Qin, B. Wan, and W. W. Song, "A comparative study of common nature-inspired algorithms for continuous function optimization," *Entropy*, vol. 23, no. 7, pp. 1–40, 2021, https://doi.org/10.3390/e23070874

[22] J. R. Koza, Genetic programming, on the programming of computers by means of natural selection. A Bradford Book, *MIT Press*, 1992.

[23] R. Reynolds, *An introduction to cultural algorithms*. 1994, https://doi.org/10.1016/B978-0-08-042392-0.50005-X

[24] R. Storn and K. Price, "Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces," *J. Glob. Optim.*, vol. 11, pp. 341–359, 1997, https://doi.org/10.1023/A:1008202821328

[25] M. O'Neill and C. Ryan, "Grammatical evolution," *IEEE Trans. Evol. Comput.*, vol. 5, no. 4, pp. 349–358, 2001, https://doi.org/10.1109/4235.942529

[26] P. Civicioglu, "Transforming geocentric cartesian coordinates to geodetic coordinates by using differential search algorithm," *Comput. Geosci.*, vol. 46, pp. 229–247, 2012, https://doi.org/10.1016/j.cageo.2011.12.011

[27] P. Civicioglu, "Backtracking search optimization algorithm for numerical optimization problems," *Appl. Math. Comput.*, vol. 219, no. 15, pp. 8121–8144, 2013, https://doi.org/10.1016/j.amc.2013.02.017

[28] H. Salimi, "Stochastic fractal search: A powerful metaheuristic algorithm," *Knowledge-Based Syst.*, vol. 75, pp. 1–18, 2015, https://doi.org/10.1016/j.knosys.2014.07.025

[29] D. Tang, S. Dong, Y. Jiang, H. Li, and Y. Huang, "ITGO: Invasive tumor growth optimization algorithm," *Appl. Soft Comput. J.*, vol. 36, pp. 670–698, 2015, https://doi.org/10.1016/j.asoc.2015.07.045

[30] A. Cheraghalipour, M. Hajiaghaei-Keshteli, and M. M. Paydar, "Tree growth algorithm (TGA): A novel approach for solving optimization problems," *Eng. Appl. Artif. Intell.*, vol. 72, pp. 393–414, 2018, https://doi.org/10.1016/j.engappai.2018.04.021

[31] M. Motevali, A. Shanghooshabad, R. Aram, and H. Keshavarz, "WHO: A new evolutionary algorithm bio-Inspired of wildebeests with a case study on bank customer segmentation," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 33, Sep. 2018, https://doi.org/10.1142/S0218001419590171

[32] M. H. Sulaiman, Z. Mustaffa, M. M. Saari, and H. Daniyal, "Barnacles mating optimizer: A new bio-inspired algorithm for solving engineering optimization problems," *Eng. Appl. Artif. Intell.*, vol. 87, no. 2019, p. 103330, 2020, https://doi.org/10.1016/j.engappai.2019.103330

[33] C. M. Rahman and T. A. Rashid, "A new evolutionary algorithm: Learner performance based behavior algorithm," *Egypt. Informatics J.*, vol. 22, no. 2, pp. 213–223, 2021, https://doi.org/10.1016/j.eij.2020.08.003

[34] A. Colorni, M. Dorigo, and V. Maniezzo, "Distributed optimization by ant colonies," *Proc. First Eur. Conf. Artif. Life*, pp. 134–142, 1991.

[35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 – International Conference on Neural Networks*, 1995, vol. 4, pp. 1942–1948, https://doi.org/10.1109/ICNN.1995.488968

[36] H. A. Abbass, "MBO: Marriage in honey bees optimization a haplometrosis polygynous swarming approach," *Proc. IEEE Conf. Evol. Comput. ICEC*, vol. 1, pp. 207–214, 2001, https://doi.org/10.1109/CEC.2001.934391

[37] X. Li, "An optimizing method based on autonomous animats: Fish-swarm algorithm," *Syst. Eng. Pract.*, vol. 22, no. 11, pp. 32–38, 2002.

[38] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Glob. Optim.*, vol. 39, no. 3, pp. 459–471, 2007, https://doi.org/10.1007/s10898-007-9149-x
[39] S. A. Chu, P. W. Tsai, and J. S. Pan, "Cat swarm optimization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4099 LNAI, pp. 854–858, 2006, https://doi.org/10.1007/11801603_94
[40] X. S. Yang, "Firefly algorithms for multimodal optimization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5792 LNCS, pp. 169–178, 2009, https://doi.org/10.1007/978-3-642-04944-6_14
[41] X. S. Yang and S. Deb, "Cuckoo search via Lévy flights," *2009 World Congr. Nat. Biol. Inspired Comput. NABIC 2009 – Proc.*, pp. 210–214, 2009, https://doi.org/10.1109/NABIC.2009.5393690
[42] X. S. Yang, "A new metaheuristic bat-inspired algorithm," *Stud. Comput. Intell.*, vol. 284, pp. 65–74, 2010, https://doi.org/10.1007/978-3-642-12538-6_6
[43] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 12, pp. 4831–4845, 2012, https://doi.org/10.1016/j.cnsns.2012.05.010
[44] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, https://doi.org/10.1016/j.advengsoft.2013.12.007
[45] A. S. Eesa, Z. Orman, and A. M. A. Brifcani, "A novel feature-selection approach based on the cuttlefish optimization algorithm for intrusion detection systems," *Expert Syst. Appl.*, vol. 42, no. 5, pp. 2670–2679, 2015, https://doi.org/10.1016/j.eswa.2014.11.009
[46] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-Based Syst.*, vol. 89, pp. 228–249, 2015, https://doi.org/10.1016/j.knosys.2015.07.006
[47] S. Mirjalili, "The ant lion optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, 2015, https://doi.org/10.1016/j.advengsoft.2015.01.010
[48] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016, https://doi.org/10.1016/j.advengsoft.2016.01.008
[49] S. Mirjalili, "Dragonfly algorithm: A new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems," *Neural Comput. Appl.*, vol. 27, no. 4, pp. 1053–1073, 2016, https://doi.org/10.1007/s00521-015-1920-1
[50] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," *Comput. Struct.*, vol. 169, pp. 1–12, 2016, https://doi.org/10.1016/j.compstruc.2016.03.001
[51] M. Yazdani and F. Jolai, "Lion optimization algorithm (LOA): A nature-inspired metaheuristic algorithm," *J. Comput. Des. Eng.*, vol. 3, no. 1, pp. 24–36, 2016, https://doi.org/10.1016/j.jcde.2015.06.003
[52] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. 114, pp. 163–191, 2017, https://doi.org/10.1016/j.advengsoft.2017.07.002
[53] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: Theory and application," *Adv. Eng. Softw.*, vol. 105, pp. 30–47, 2017, https://doi.org/10.1016/j.advengsoft.2017.01.004
[54] G. Dhiman and V. Kumar, "Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications," *Adv. Eng. Softw.*, vol. 114, pp. 48–70, 2017, https://doi.org/10.1016/j.advengsoft.2017.05.014
[55] G. Dhiman and V. Kumar, "Emperor penguin optimizer: A bio-inspired algorithm for engineering problems," *Knowledge-Based Syst.*, vol. 159, pp. 20–50, 2018, https://doi.org/10.1016/j.knosys.2018.06.001

[56] J. Pierezan and L. Dos Santos Coelho, "Coyote optimization algorithm: A new metaheuristic for global optimization problems," *2018 IEEE Congr. Evol. Comput. CEC 2018 – Proc.*, pp. 1–8, 2018, https://doi.org/10.1109/CEC.2018.8477769
[57] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, and H. Chen, "Harris hawks optimization: Algorithm and applications," *Futur. Gener. Comput. Syst.*, vol. 97, pp. 849–872, 2019, https://doi.org/10.1016/j.future.2019.02.028
[58] M. Jain, V. Singh, and A. Rani, "A novel nature-inspired algorithm for optimization: Squirrel search algorithm," *Swarm Evol. Comput.*, vol. 44, pp. 148–175, 2019, https://doi.org/10.1016/j.swevo.2018.02.013
[59] S. Shadravan, H. R. Naji, and V. K. Bardsiri, "The sailfish optimizer: A novel nature-inspired metaheuristic algorithm for solving constrained engineering optimization problems," *Eng. Appl. Artif. Intell.*, vol. 80, pp. 20–34, 2019, https://doi.org/10.1016/j.engappai.2019.01.001
[60] R. Masadeh, B. A. Mahafzah, and A. Sharieh, "Sea lion optimization algorithm," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 5, pp. 388–395, 2019, https://doi.org/10.14569/IJACSA.2019.0100548
[61] S. Harifi, M. Khalilian, J. Mohammadzadeh, and S. Ebrahimnejad, "Emperor penguins colony: A new metaheuristic algorithm for optimization," *Evol. Intell.*, vol. 12, no. 2, pp. 211–226, 2019, https://doi.org/10.1007/s12065-019-00212-x
[62] H. A. Alsattar, A. A. Zaidan, and B. B. Zaidan, "Novel meta-heuristic bald eagle search optimisation algorithm," *Artif. Intell. Rev.*, vol. 53, no. 3, pp. 2237–2264, 2020, https://doi.org/10.1007/s10462-019-09732-5
[63] A. M. Fathollahi-Fard, M. Hajiaghaei-Keshteli, and R. Tavakkoli-Moghaddam, "Red deer algorithm (RDA): A new nature-inspired meta-heuristic," *Soft Comput.*, vol. 24, no. 19, pp. 14637–14665, 2020, https://doi.org/10.1007/s00500-020-04812-z
[64] J. Xue and B. Shen, "A novel swarm intelligence optimization approach: Sparrow search algorithm," *Syst. Sci. Control Eng.*, vol. 8, no. 1, pp. 22–34, 2020, https://doi.org/10.1080/21642583.2019.1708830
[65] M. Khishe and M. R. Mosavi, "Chimp optimization algorithm," *Expert Syst. Appl.*, vol. 149, p. 113338, 2020, https://doi.org/10.1016/j.eswa.2020.113338
[66] K. Zervoudakis and S. Tsafarakis, *A mayfly optimization algorithm*, vol. 145. Elsevier Ltd, 2020. https://doi.org/10.1016/j.cie.2020.106559
[67] L. Abualigah, D. Yousri, M. Abd Elaziz, A. A. Ewees, M. A. A. Al-qaness, and A. H. Gandomi, "Aquila optimizer: A novel meta-heuristic optimization algorithm," *Comput. Ind. Eng.*, vol. 157, p. 107250, 2021, https://doi.org/10.1016/j.cie.2021.107250
[68] G. Dhiman and V. Kumar, "Seagull optimization algorithm: Theory and its applications for large-scale industrial engineering problems," *Knowledge-Based Syst.*, vol. 165, pp. 169–196, 2019, https://doi.org/10.1016/j.knosys.2018.11.024
[69] D. Połap and M. Woźniak, "Red fox optimization algorithm," *Expert Syst. Appl.*, vol. 166, 2021, https://doi.org/10.1016/j.eswa.2020.114107
[70] A. Mohammadi-Balani, M. Dehghan Nayeri, A. Azar, and M. Taghizadeh-Yazdi, "Golden eagle optimizer: A nature-inspired metaheuristic algorithm," *Comput. Ind. Eng.*, vol. 152, pp. 1–45, 2021, https://doi.org/10.1016/j.cie.2020.107050
[71] C. Zhong, G. Li, and Z. Meng, "Beluga whale optimization: A novel nature-inspired metaheuristic algorithm," *Knowledge-Based Syst.*, vol. 251, p. 109215, 2022, https://doi.org/10.1016/j.knosys.2022.109215
[72] H. T. Sadeeq and A. M. Abdulazeez, "Giant trevally optimizer (GTO): A novel metaheuristic algorithm for global optimization and challenging engineering problems," *IEEE Access*, pp. 121615–121640, 2022, https://doi.org/10.1109/ACCESS.2022.3223388

[73] H. Le Minh, T. Sang-To, G. Theraulaz, M. Abdel Wahab, and T. Cuong-Le, "Termite life cycle optimizer," *Expert Syst. Appl.*, vol. 213, p. 119211, 2023, https://doi.org/10.1016/j.eswa.2022.119211

[74] M. Abdel-Basset, R. Mohamed, M. Jameel, and M. Abouhawwash, "Nutcracker optimizer: A novel nature-inspired metaheuristic algorithm for global optimization and engineering design problems," *Knowledge-Based Syst.*, vol. 262, p. 110248, 2023, https://doi.org/10.1016/j.knosys.2022.110248

[75] S. Kirkpatrick, C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983, https://doi.org/10.1126/science.220.4598.671

[76] N. Mladenović and P. Hansen, "Variable neighborhood search," *Comput. Oper. Res.*, vol. 24, no. 11, pp. 1097–1100, 1997, https://doi.org/10.1016/S0305-0548(97)00031-2

[77] O. K. Erol and I. Eksin, "A new optimization method: Big bang-big crunch," *Adv. Eng. Softw.*, vol. 37, no. 2, pp. 106–111, 2006, https://doi.org/10.1016/j.advengsoft.2005.04.005

[78] R. A. Formato, "Central force optimization: A new metaheuristic with applications in applied electromagnetics," *Prog. Electromagn. Res.*, vol. 77, pp. 425–491, 2007, https://doi.org/10.2528/PIER07082403

[79] H. Shah-Hosseini, "Problem solving by intelligent water drops," *2007 IEEE Congr. Evol. Comput. CEC 2007*, pp. 3226–3231, 2007, https://doi.org/10.1109/CEC.2007.4424885

[80] D. R. Monismith and B. E. Mayfield, "Slime mold as a model for numerical optimization," *2008 IEEE Swarm Intell. Symp. SIS 2008*, 2008, https://doi.org/10.1109/SIS.2008.4668295

[81] E. Rashedi, H. Nezamabadi-pour, and S. Saryazdi, "GSA: A gravitational search algorithm," *Inf. Sci. (Ny).*, vol. 179, no. 13, pp. 2232–2248, 2009, https://doi.org/10.1016/j.ins.2009.03.004

[82] A. Kaveh and S. Talatahari, "A novel heuristic optimization method: Charged system search," *Acta Mech.*, vol. 213, no. 3–4, pp. 267–289, 2010, https://doi.org/10.1007/s00707-009-0270-4

[83] E. Cuevas, D. Oliva, D. Zaldivar, M. Pérez-Cisneros, and H. Sossa, "Circle detection using electro-magnetism optimization," *Inf. Sci. (Ny).*, vol. 182, no. 1, pp. 40–55, 2012, https://doi.org/10.1016/j.ins.2010.12.024

[84] H. Eskandar, A. Sadollah, A. Bahreininejad, and M. Hamdi, "Water cycle algorithm – A novel metaheuristic optimization method for solving constrained engineering optimization problems," *Comput. Struct.*, vol. 110–111, pp. 151–166, 2012, https://doi.org/10.1016/j.compstruc.2012.07.010

[85] A. Hatamlou, "Black hole: A new heuristic optimization approach for data clustering," *Inf. Sci. (Ny).*, vol. 222, pp. 175–184, 2013, https://doi.org/10.1016/j.ins.2012.08.023

[86] A. Sadollah, A. Bahreininejad, H. Eskandar, and M. Hamdi, "Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems," *Appl. Soft Comput. J.*, vol. 13, no. 5, pp. 2592–2612, 2013, https://doi.org/10.1016/j.asoc.2012.11.026

[87] A. Kaveh and V. R. Mahdavi, "Colliding bodies optimization: A novel meta-heuristic method," *Comput. Struct.*, vol. 139, pp. 18–27, 2014, https://doi.org/10.1016/j.compstruc.2014.04.005

[88] H. Shareef, A. A. Ibrahim, and A. H. Mutlag, "Lightning search algorithm," *Appl. Soft Comput. J.*, vol. 36, pp. 315–333, 2015, https://doi.org/10.1016/j.asoc.2015.07.028

[89] S. Mirjalili, S. M. Mirjalili, and A. Hatamlou, "Multi-verse optimizer: A nature-inspired algorithm for global optimization," *Neural Comput. Appl.*, vol. 27, no. 2, pp. 495–513, 2016, https://doi.org/10.1007/s00521-015-1870-7

[90] A. Kaveh and A. Dadras, "A novel meta-heuristic optimization algorithm: Thermal exchange optimization," *Adv. Eng. Softw.*, vol. 110, pp. 69–84, 2017, https://doi.org/10.1016/j.advengsoft.2017.03.014

[91] A. Husseinzadeh Kashan, R. Tavakkoli-Moghaddam, and M. Gen, "Find-fix-finish-exploit-analyze (F3EA) meta-heuristic algorithm: An effective algorithm with new evolutionary operators for global optimization," *Comput. Ind. Eng.*, vol. 128, no. 2018, pp. 192–218, 2019, https://doi.org/10.1016/j.cie.2018.12.033

[92] F. A. Hashim, E. H. Houssein, M. S. Mabrouk, W. Al-Atabany, and S. Mirjalili, "Henry gas solubility optimization: A novel physics-based algorithm," *Futur. Gener. Comput. Syst.*, vol. 101, pp. 646–667, 2019, https://doi.org/10.1016/j.future.2019.07.015

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

[94] M. Ghasemi, I. F. Davoudkhani, E. Akbari, A. Rahimnejad, S. Ghavidel, and L. Li, "A novel and effective optimization algorithm for global optimization and its engineering applications: Turbulent flow of water-based optimization (TFWO)," *Eng. Appl. Artif. Intell.*, vol. 92, 2020, https://doi.org/10.1016/j.engappai.2020.103666

[95] F. A. Hashim, K. Hussain, E. Houssein, M. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: A new metaheuristic algorithm for solving optimization problems," *Appl. Intell.*, vol. 51, pp. 1–21, 2021, https://doi.org/10.1007/s10489-020-01893-z

[96] J. L. J. Pereira, M. B. Francisco, C. A. Diniz, G. Antônio Oliver, S. S. Cunha, and G. F. Gomes, "Lichtenberg algorithm: A novel hybrid physics-based meta-heuristic for global optimization," *Expert Syst. Appl.*, vol. 170, 2021, https://doi.org/10.1016/j.eswa.2020.114522

[97] V. Goodarzimehr, S. Shojaee, S. Hamzehei-Javaran, and S. Talatahari, "special relativity search: A novel metaheuristic method based on special relativity physics," *Knowledge-Based Syst.*, vol. 257, p. 109484, 2022, https://doi.org/10.1016/j.knosys.2022.109484

[98] M. Azizi, U. Aickelin, H. A Khorshidi, and M. Baghalzadeh Shishehgarkhaneh, "Energy valley optimizer: A novel metaheuristic algorithm for global and engineering optimization," *Sci. Rep.*, vol. 13, no. 1, p. 226, 2023, https://doi.org/10.1038/s41598-022-27344-y

[99] M. Abdel-Basset, D. El-Shahat, M. Jameel, and M. Abouhawwash, "Young's double-slit experiment optimizer: A novel metaheuristic optimization algorithm for global and constraint optimization problems," *Comput. Methods Appl. Mech. Eng.*, vol. 403, p. 115652, 2023, https://doi.org/10.1016/j.cma.2022.115652

[100] T. Ray and K. M. Liew, "Society and civilization: An optimization algorithm based on the simulation of social behavior," *IEEE Trans. Evol. Comput.*, vol. 7, no. 4, pp. 386–396, 2003, https://doi.org/10.1109/TEVC.2003.814902

[101] C. Dai, Y. Zhu, and W. Chen, "Seeker optimization algorithm BT – computational intelligence and security," pp. 167–176, 2007, https://doi.org/10.1007/978-3-540-74377-4_18

[102] E. Atashpaz-Gargari and C. Lucas, "Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition," *2007 IEEE Congr. Evol. Comput. CEC 2007*, pp. 4661–4667, 2007, https://doi.org/10.1109/CEC.2007.4425083

[103] L. Zhang, C. Dahlmann, and Y. Zhang, "Human-inspired algorithms for continuous function optimization," Nov. 2009, https://doi.org/10.1109/ICICISYS.2009.5357838

[104] Y. Xu, Z. Cui, and J. Zeng, "Social emotional optimization algorithm for nonlinear constrained optimization problems," In *Swarm, Evolutionary, and Memetic Computing. SEMCCO 2010. Lecture Notes in Computer Science*, B. K. Panigrahi, S. Das, P. N. Suganthan, and S. S. Dash, (Eds.), vol. 6466, Springer, Berlin, Heidelberg, 2010, https://doi.org/10.1007/978-3-642-17563-3_68

[105] Y. Shi, "Brain storm optimization algorithm," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6728 LNCS, no. PART 1, pp. 303–309, 2011, https://doi.org/10.1007/978-3-642-21515-5_36

[106] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: An optimization method for continuous non-linear large scale problems," *Inf. Sci. (Ny).*, vol. 183, no. 1, pp. 1–15, 2012, https://doi.org/10.1016/j.ins.2011.08.006

[107] H. Shayeghi and J. Dadashpour, "Anarchic society optimization based PID control of an automatic voltage regulator (AVR) system," *Electr. Electron. Eng.*, vol. 2, no. 4, pp. 199–207, 2012, https://doi.org/10.5923/j.eee.20120204.05

[108] S. J. Mousavirad and H. Ebrahimpour-Komleh, "Human mental search: A new population-based metaheuristic optimization algorithm," *Appl. Intell.*, vol. 47, 2017, https://doi.org/10.1007/s10489-017-0903-6

[109] Q. Askari, I. Younas, and M. Saeed, "Political optimizer: A novel socio-inspired meta-heuristic for global optimization," *Knowledge-Based Syst.*, vol. 195, p. 105709, 2020, https://doi.org/10.1016/j.knosys.2020.105709

[110] S. H. Samareh Moosavi and V. K. Bardsiri, "Poor and rich optimization algorithm: A new human-based and multi populations algorithm," *Eng. Appl. Artif. Intell.*, vol. 86, no. 2018, pp. 165–181, 2019, https://doi.org/10.1016/j.engappai.2019.08.025

[111] Y. Zhang and Z. Jin, "Group teaching optimization algorithm: A novel metaheuristic method for solving global optimization problems," *Expert Syst. Appl.*, vol. 148, p. 113246, 2020, https://doi.org/10.1016/j.eswa.2020.113246

[112] A. W. Mohamed, A. A. Hadi, and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: A novel nature-inspired algorithm," *Int. J. Mach. Learn. Cybern.*, vol. 11, no. 7, pp. 1501–1529, 2020, https://doi.org/10.1007/s13042-019-01053-x

[113] M. A. Al-Betar, Z. A. A. Alyasseri, and M. A. Awadallah, et al., "Coronavirus herd immunity optimizer (CHIO)," *Neural Comput & Applic*, vol. 33, pp. 5011–5042, 2021. https://doi.org/10.1007/s00521-020-05296-6

[114] M. Braik, M. H. Ryalat, and H. Al-Zoubi, A novel meta-heuristic algorithm for solving numerical optimization problems: Ali Baba and the forty thieves. *Neural Comput & Applic*, vol. 34, pp. 409–455, 2022. https://doi.org/10.1007/s00521-021-06392-x

[115] T. S. L. V. Ayyarao *et al.*, "War strategy optimization algorithm: A new effective meta-heuristic algorithm for global optimization," *IEEE Access*, vol. 10, pp. 25073–25105, 2022, https://doi.org/10.1109/ACCESS.2022.3153493

[116] A. Husseinzadeh Kashan, "League championship algorithm (LCA): An algorithm for global optimization inspired by sport championships," *Appl. Soft Comput. J.*, vol. 16, pp. 171–200, 2014, https://doi.org/10.1016/j.asoc.2013.12.005

[117] N. Razmjooy, M. Khalilpour, and M. Ramezani, "A new meta-heuristic optimization algorithm inspired by FIFA World Cup Competitions: Theory and its application in PID designing for AVR system," *J. Control. Autom. Electr. Syst.*, vol. 27, no. 4, pp. 419–440, 2016, https://doi.org/10.1007/s40313-016-0242-6

[118] E. Fadakar and M. Ebrahimi, "A new metaheuristic football game inspired algorithm," in *2016 1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC)*, 2016, pp. 6–11, https://doi.org/10.1109/CSIEC.2016.7482120

[119] S. M. Ashrafi and A. B. Dariane, "Performance evaluation of an improved harmony search algorithm for numerical optimization: Melody search (MS)," *Eng. Appl. Artif. Intell.*, vol. 26, no. 4, pp. 1301–1321, 2013, https://doi.org/10.1016/j.engappai.2012.08.005

[120] A. Rezoug and D. Boughaci, "A self-adaptive harmony search combined with a stochastic local search for the 0–1 multidimensional knapsack problem," *Int. J. Bio-Inspired Comput.*, vol. 8, pp. 234–239, 2016, https://doi.org/10.1504/IJBIC.2016.078641

[121] Y. Zhou, Y. Wang, X. Chen, L. Zhang, and K. Wu, "A novel path planning algorithm based on plant growth mechanism," *Soft Comput.*, vol. 21, 2017, https://doi.org/10.1007/s00500-016-2045-x

[122] Y. Bekakra, Y. Labbi, D. Ben Attous, and O. P. Malik, "Rooted tree optimization algorithm to improve DTC response of DFIM," *J. Electr. Eng. Technol.*, vol. 16, 2021, https://doi.org/10.1007/s42835-021-00796-4

[123] X. S. Yang, "Flower pollination algorithm for global optimization," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 7445 LNCS, pp. 240–249, 2012, https://doi.org/10.1007/978-3-642-32894-7_27

[124] S. Mirjalili, "SCA: A sine cosine algorithm for solving optimization problems," *Knowledge-Based Syst.*, vol. 96, pp. 120–133, 2016, https://doi.org/10.1016/j.knosys.2015.12.022

[125] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," *Comput. Methods Appl. Mech. Eng.*, vol. 376, p. 113609, 2021, https://doi.org/10.1016/j.cma.2020.113609

[126] A. A. Kamarudin, Z. A. Othman, and H. M. Sarim, "Water flow algorithm decision support tool for travelling salesman problem," *AIP Conf. Proc.*, vol. 1761, no. 1, p. 20054, Aug. 2016, https://doi.org/10.1063/1.4960894

[127] M. E. Colak and A. Varol, "A novel intelligent optimization algorithm inspired from circular water waves," *Elektron. ir Elektrotechnika*, vol. 21, no. 5, pp. 3–6, 2015, https://doi.org/10.5755/j01.eee.21.5.13316

[128] Q. Li, Y. Bai, and W. Gao, "Improved initialization method for metaheuristic algorithms: A novel search space view," *IEEE Access*, vol. 9, pp. 121366–121384, 2021, https://doi.org/10.1109/ACCESS.2021.3073480

[129] E. Pira, *City councils evolution: a socio-inspired metaheuristic optimization algorithm*, no. 0123456789. Springer Berlin Heidelberg, 2022. https://doi.org/10.1007/s12652-022-03765-5

8 AUTHORS

Haval Tariq Sadeeq is a lecturer in the Information Technology Department at Duhok Polytechnic University in Duhok, Kurdistan Region of Iraq. His research interests include computational intelligence, metaheuristics, swarm intelligence, and engineering optimization. E-mail address: haval.tariq@dpu.edu.krd

Adnan Mohsin Abdulazeez is a professor in the Information Technology Department at Duhok Polytechnic University in Duhok, Kurdistan Region of Iraq. His research interests are intelligent systems, soft computing, multimedia, and network security. E-mail address: adnan.mohsin@dpu.edu.krd