engineering proceedings

MDPI

Proceeding Paper

Metaheuristic Algorithms for Optimization: A Brief Review [†]

Vinita Tomar [1,*], Mamta Bansal [1] and Pooja Singh [2]

[1] Department of Computer Science, Shobhit Institute of Engineering & Technology (Deemed-to-Be University), Meerut 250110, India; mamta.bansal@shobhituniversity.ac.in

[2] Department of Computer Science, Maharaja Surajmal Institute, Janakpuri, New Delhi 110058, India; pooja.asm@gmail.com

[*] Correspondence: tomar.vinita@gmail.com

[†] Presented at the International Conference on Recent Advances in Science and Engineering, Dubai, United Arab Emirates, 4–5 October 2023.

Abstract: In the area of optimization, metaheuristic algorithms have attracted a lot of interest. For many centuries, human beings have utilized metaheuristic algorithms as a problem-solving approach. The application of these methods to combinatorial optimization problems has rapidly become a growing area of research, incorporating principles of natural selection, evolution, and problem-solving strategies. While conventional software engineering methods may not always be effective in resolving software issues, mathematical optimization using metaheuristics can offer a solution. As a result, metaheuristics have become an increasingly important part of modern optimization, with a large number of algorithms emerging over the last two decades. The purpose of this study is to present a quick overview of these algorithms so that researchers may choose and use the best metaheuristic method for their optimization issues. The key components and concepts of each type of algorithm have been discussed, highlighting their benefits and limitations. This paper aims to provide a comprehensive review of these algorithms, including evolution-based methods, swarm intelligence-based, physics-based, human-related, and hybrid metaheuristics by highlighting their key components and concepts and comparing and contrasting their similarities and differences. This work also addressed some of the difficulties associated with metaheuristic algorithms. Some practical uses of these metaheuristic algorithms were addressed.

Keywords: metaheuristic algorithms; optimization; evolution-based; physics-based techniques; swarm intelligence; hybrid

check for updates

Citation: Tomar, V.; Bansal, M.; Singh, P. Metaheuristic Algorithms for Optimization: A Brief Review. *Eng. Proc.* **2023**, *59*, 238. https://doi.org/10.3390/engproc2023059238

Academic Editors: Nithesh Naik, Rajiv Selvam, Pavan Hiremath, Suhas Kowshik CS and Ritesh Ramakrishna Bhat

Published: 13 March 2024

1. Introduction

Metaheuristic algorithms are optimization techniques that are designed to find an adequate solution for a broad range of optimization problems. These algorithms stand out from other optimization techniques in several ways. Firstly, they are derivative-free, meaning that they do not require any sort of calculation of derivatives in the search space, as opposed to gradient-based search techniques. This makes metaheuristic algorithms much simpler, more flexible, and more capable of avoiding local optima, making them highly effective for handling challenging optimization tasks. The stochastic nature is another characteristic of metaheuristic algorithms, which implies they begin the optimization process by generating random results. This makes it more likely that the algorithms will be able to avoid premature convergence and quickly and effectively examine the search space. Metaheuristics balance between exploration and profit to accomplish this. During the discovery step, the algorithms thoroughly examine the search space's interesting regions, and then, in the analysis phase, they carry out local searches in these regions to perceive the most gilt-edge resolution. The specific and primary key advantages of metaheuristic algorithms are their versatility and flexibility. They can be modified easily to fit the specific requirements of a particular problem, making them an ideal solution for a broad range of optimization problems across various fields of engineering and science. For example, metaheuristics have been successfully applied in electrical engineering for power generation optimization, in industrial scheduling and transportation, in civil engineering for bridge and building design, in communication for radar design and networking, and in data mining for classification, prediction, clustering, and system modeling. Metaheuristics are a powerful and widely used framework for solving optimization problems. They provide a set of guidelines and strategies that can be used to develop efficient heuristic optimization algorithms. Metaheuristics are sophisticated methods or heuristics that are intended to locate, induce, or select a heuristic that can offer the best possible solution to an optimization issue, even in the absence of enough data or when computational resources are limited. They are employed in both mathematical optimization and computer science. Metaheuristics enable the efficient exploration of a large search space by testing a subdivision containing elucidations that could ordinarily be excessively sizeable to be wholly recapitulated or explored. Metaheuristics can be used in different types of problems as they showcase a class of generic search algorithms. What inspires them are ideas from different areas which help them in finding a way to solve optimization problems. As examples, we can look at an artificial electric field optimizer, which is a physics-based algorithm, or an evolutionary strategy, which is an evolution-based algorithm. In optimization problems, mathematical theorems are used to make decisions that help find the best possible solution to a problem, which is far better than going through every possible solution.

A few of the most commonly used classes of metaheuristics, described below, are capable of solving problems for which even the most powerful classical computers cannot be programmed. Contingently as per their behavior, metaheuristic breakthroughs could be classified within four distinct classifications: human-related, physics-rooted, evolution-based, and based on swarm intellects. The field of nature-inspired intelligent algorithms has a rich history, stretching back to its early development years. These algorithms, often referred to as NII algorithms, are intelligent metaheuristic optimization techniques known for their ability to refine candidate solution populations using information acquired during the algorithm's execution. The birth of this field can be traced back to the introduction of the first genetic algorithm by Holland in 1975, which ignited a spark for the development of NII algorithms. Although genetic algorithms are not typically categorized as NII methods, they paved the way for scientists to examine other natural concepts that could be modeled for high-performance optimization. The first such algorithm, known as "Simulated Annealing", was put forth in 1983 by Kirkpatrick et al. [1] This algorithm was modeled after the annealing process in metallurgy and has since become one of the most recognized optimization methods.

Another well-known NII algorithm is the Stochastic Diffusion Search, introduced in 1989 by Bishop and later referred to as such by Bishop and Torr in 1992. With this approach, agents seek out more effective solutions and collect close to locally optimal solutions as they explore the solution space. M. Dorigo first suggested ant colony optimization, or ACO, in his doctoral dissertation in 1992. This approach utilizes a utility-based model and focuses on specific solutions while avoiding low-quality ones, using pheromones as a smart operator. Each representative reconditions the pheromone's immensity for every escape trail they discover, forming powerful pheromone strings. The concluding optimal remedy is often composed of elements of these trials, as they are marked with more pheromones due to having been followed by a greater number of agents. In 1995, Particle Swarm Optimization was proposed by Eberhart and Kennedy. The first time an NII algorithm was concluded upon the cumulative intelligence of various representatives as inimical towards the development of a specific answer such as simulated annealing, was with this population-based approach which takes inspiration from the collective intelligence of animal swarms and flocks. In 1997, Storn and Price instigated differential evolution drawing inspiration from Holland's work on genetic algorithms. Despite being classified as a metaheuristic, the authors claimed that their approach was more of a heuristic method. In recent years, the number of NII algorithms being published has only continued to grow, leading researchers to question the necessity of so many algorithms in literature and their crucial role in solving different problems. The research by Fister et al. [2] which is dominantly mortified based on population further instigates NII algorithms and fabricated a few compelling observations, further inspiring the authors to explore deeper into this problem.

2. Optimization Problems and Metaheuristics

Metaheuristics are a class of optimization algorithms that can handle complex, nonlinear problems and find a good solution without necessarily finding the global optimum. Unlike traditional optimization techniques that linearize the objective function or use derivatives and gradients, metaheuristics employ advanced strategies to search for a solution They are extensively deployed in several industries and professions, including administration, planning, architecture, engineering, healthcare, and logistics. The efficiency of metaheuristics in solving difficult optimization problems has made them a popular choice in many applications. A group of optimization techniques known as metaheuristics directs the search process to provide elevated outcomes. They are particularly useful in situations where an explicit equation-based model cannot be developed. In comparison to conventional optimization techniques, the capacity to thoroughly explore the problem search space results in a larger probability of obtaining the optimal solutions. Over the years, several metaheuristic algorithms have emerged, including evolution-based, nature-inspired, physics-based, and stochastic algorithms. Many of these algorithms are population-based, meaning that they maintain and manipulate an abundance of remedies to perceive the optimal escape. Metaheuristic optimization leverages these algorithms to resolve an extensive scope of optimization complications in profuse domains including engineering design, economics, holiday planning, and internet routing. With limited resources and time, it is essential to optimize the utilization of these resources to achieve the best results. The optimization of real-world problems is often characterized by its complexity and non-linearity, along with multiple conflicting objectives and various challenging constraints. Finding the optimal solution for such problems can be an arduous task, as optimal solutions may not even exist in some cases. The goal of this article is to give a general overview of metaheuristic optimization, including some of the most popular metaheuristic algorithms and their underlying ideas.

The task of determining the least or maximum value of a given function can be viewed as an optimization problem. For instance, if we consider a function $f(a) = a^2$, we can determine that its minimum value, *fmin = 0*, occurs at a = *0* in the entire domain of -infinity < a < infinity. However, for simpler functions, we can determine the potential solution by setting the first derivative, $f'(a) = 0$, to zero. In addition, we can verify if the answer is minimal or maximal by using the second derivative, $f''(a)$. But, in certain cases, the functions may have discontinuities, making it difficult to obtain derivative information.

Optimization

In the domain of optimization, a task that involves minimization or maximization can be expressed as a problem.

$$\text{minimize} f1(a), \ldots, fi(a), \ldots, fI(a), a = (a1, \ldots, ad) \tag{1}$$

subject to

$$pk(a) = 0, \ (j = 1,\ 2, \ldots, J)$$

$$st(a) \leq 0, (k = 1, 2, \ldots, K)$$

where, *pk* and *st* are the equality and inequality constraints, respectively, and $f1\ldots, fI$ is the set of objectives. When I = 1, this problem is mentioned as a particular-equitable optimization complication, and further $I \geq 2$, it is cited as an assorted-equitable optimization complication.

It is worth noting that the functions *fi*, *pk*, and *st* in this optimization problem can be nonlinear. If they are linear, the problem is simplified to a linear programming problem that can be answered using Dantzig's simplex method, which was initially put forth in 1963. For nonlinear optimization problems, metaheuristics are often used as a solution strategy, as they can handle the complexities and uncertainties inherent in these types of problems. In addition, the inequality constraints *st* may be flipped by substituting *st* with $-st$, and the minimization problem can be changed into a maximization issue by simply substituting fi with $-fi$. This highlights the versatility of mathematical optimization and the various forms it can take to address diverse real-world problems.

At its core, the most basic form of optimization is known as unconstrained function optimization. Ackley's function, which has a global minimum of 0 at the point (0,0), is a frequent test function used to verify and test this kind of optimization. In mathematics, optimization problems entail selecting the optimal option among a range of viable options. These problems are typically defined as having an objective function with one or more variables and a set of constraints, which can either be discrete or continuous in nature depending on the variables involved.

The number of variables taken into account in the objective function has a significant impact on how complex an optimization issue is. The term "NP" (non-deterministic polynomial time) problem refers to a class of optimization problems that can be solved in polynomial time by non-deterministic algorithms. This class includes many real-world optimization problems. Figure 1 illustrates the NP problem.

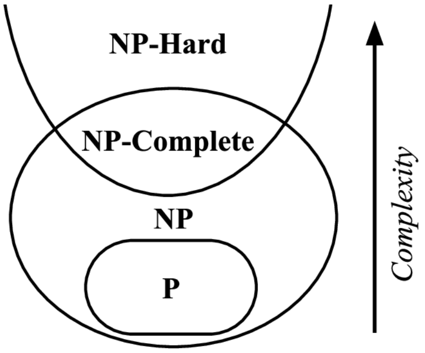

Figure 1. NP Problem.

Many common problems like the traveling salesman and graph coloring fall into this category. This is where a metaheuristic can help us. As a higher-level heuristic or procedure, a metaheuristic provides a sufficient solution to an optimization problem that is sufficiently good enough to solve. Most of the time, they work by sampling a subset of solutions that is too large to enumerate in full. In addition, they can also work with incomplete or imperfect data, which is crucial to their effectiveness. A metaheuristic cannot ensure that it will discover the globally optimal solution, in contrast to numerical optimization techniques. It can produce satisfactory results much faster and with significantly less processing effort.

3. Framing the Metaheuristic

A metaheuristic seeks to maximize efficiency by exploring the search space to find near-optimal solutions. They are based on a strategy to drive the search process. The strategy can take inspiration from any natural or artificial system under observation. This can come from as diverse sources as the metallurgical process of annealing to the foraging behavior of ants. Defining a metaheuristic around a search strategy requires us to pursue scientific and engineering goals. The scientific goal is to model the mechanism behind an inspiration like a swarm of ants. The engineering goal is to design systems that can solve practical problems. While it is impractical to define a generic framework, we can discuss some defining characteristics. Finding the ideal balance between exploration and exploitation is a crucial aspect of any metaheuristic strategy. Exploration consists of exploring the entire feasible region as much as possible to evade suboptimal solutions. Exploitation involves exploring the surrounding area of a promising region to find the ideal solution. Figure 2 illustrates the exploitation and exploration flowchart.

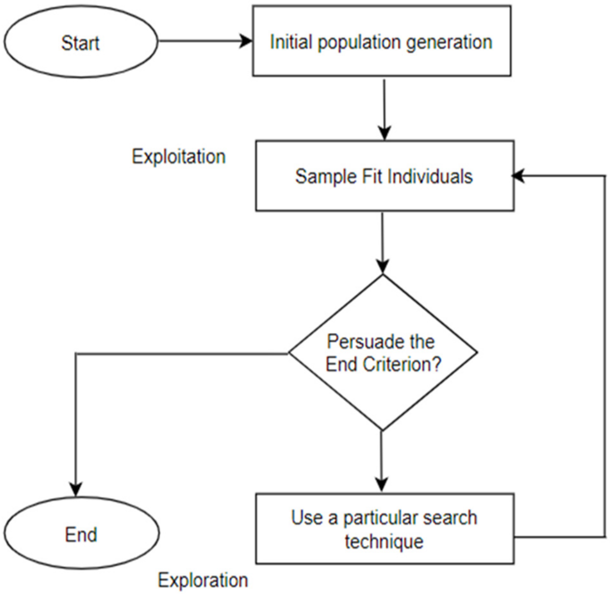
Start
Initial population generation
Exploitation
Sample Fit Individuals
Persuade the End Criterion?
End
Use a particular search technique
Exploration

Figure 2. Exploitation and Exploration flowchart.

Almost in all such metaheuristics, we tend to employ a fitness function to evaluate the candidate solutions. This is to sample the best solutions so far to focus on exploitation. Further, we use certain aspects of the search strategy to bring randomness and emphasize exploration. This is unique to every search strategy and hence quite difficult to represent using a general formulation. We can use these metaheuristics to solve multi-dimensional real-value functions without relying on their gradient. This is a crucial point, because it implies that these algorithms can solve optimization problems that are non-continuous, noisy, and change over time as opposed to several algorithms that employ gradient descent, such as linear regression.

4. Categories of Metaheuristics

The classification of nature-inspired algorithms is shown in Figure 3 below.

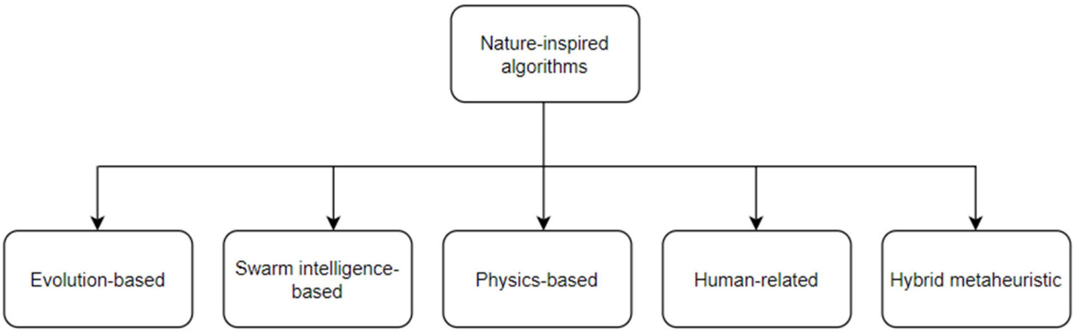

Figure 3. Classification of nature-inspired algorithms.

4.1. Evolution-Based Algorithms

Evolutionary algorithms (EA) are a class of algorithms inspired by Darwin's evolutionary theory. His theory asserts that variation occurs randomly among members of a species. Evolutionary algorithms take inspiration from this theory to identify near-optimal solutions in the search space. Each iteration in such an algorithm is known as a generation and is composed of parent selection, recombination (crossover), mutation, and survivor selection. While crossover and mutation are responsible for the exploration, parent and survivor selection brings out the exploitation. The optimization techniques inspired by natural evolution are referred to as evolutionary algorithms and include the popular genetic algorithms (GA) and differential evolution (DE) algorithms. These methods initiate their procedure with arbitrarily generated potential solutions and refine the population by recombining the best solutions to create new individuals through processes, for instance, crossover, and mutation.

The genetic algorithm (GA), which is further contented upon the Darwinian progression, is the most extensively utilized of the numerous evolutionary algorithms. The strategy of evolution escalates the Genetic Programming; Tabu hunting and differential expansion are additional prominent algorithms in this domain. A useful tool in the field of image processing is the ground-breaking chaotic differential search method developed by Gan and Duan [3]. This algorithm is unique in its combination of lateral inhibition for extracting edges and enhancing images. In conclusion, evolution-based algorithms have proven to be a valuable tool in various fields ranging from image processing to disease diagnosis, wind speed forecasting, and even cancer symptom identification.

4.2. Swarm Intelligence-Based Algorithms

The second category of metaheuristic algorithms, called swarm intelligence, is modeled after how social animals in a herd communicate knowledge about each other during the optimization process. The concept of swarm algorithms (SA) originates from the way animals and insects behave in groups. The group behavior of ants or bees in the natural world serves as the model for these algorithms. The key point in such algorithms is the information shared within the swarm, which can directly influence the movement of each agent. By controlling the information sharing between agents in a swarm, we can achieve the equilibrium between the investigation as well as the manipulation of the forage expanse. Instances of representative metaheuristics in this domain include the BAT (Bio-Inspired) algorithm, a metaheuristic algorithm inspired by bat echolocation. It explores the search space and optimizes solutions by altering the frequency and loudness of outgoing signals using echolocation and adaptive frequency tuning methods. The CS (Cuckoo Search) algorithm has been extensively employed to fathom the diversity of real-world issues. It was inspired by the breeding behavior of cuckoo birds. To deal with binary optimization problems, several binary adaptations of the CS algorithm have been developed. The life of a grasshopper and how its behavior evolves serve as the basis for GOA (Grasshopper Optimization Algorithm). It replicates grasshopper interactions and movements to achieve optimal solutions by balancing exploration and exploitation through location updates based on attraction and repulsion processes. The FA (Firefly Algorithm), based on the behavior of fireflies communicating through light flashes, has become a popular approach for feature selection problems. It stimulates the attraction and movement of fireflies to address optimization issues by updating locations based on brightness and distance estimates, facilitating convergence toward optimal solutions in the search space. The DA (Dragonfly Algorithm) is a metaheuristic optimization approach that is influenced by the behavior of dragonflies in nature. The approach has gained widespread acceptance and was successfully applied to resolve a diversity of optimization issues. The computational technique known as the GWO (Grey Wolf Optimizer) is based on how wolves hunt as a group. It replicates the leadership hierarchy and cooperative hunting of wolves to optimize solutions by altering locations and exploring a multi-dimensional search space. The Flower Pollination Algorithm (FPA) is a metaheuristic algorithm that was inspired by flower pollination. It emulates pollination behavior by sharing and recombining information among candidate solutions, enabling exploration and exploitation in the search space. A widely used method called ALO (Ant Lion Optimizer) was influenced by ant lion and ant hunting. It can be used to identify optimal (or nearly optimal) solutions to a range of real-time situations. The WOA (Whale Optimization Algorithm) is rooted in the hunting tactics of humpback whales. It is influenced by humpback whales' bubble-net hunting behavior. It searches for optimum solutions by using the ideas of exploration, exploitation, and encircling, replicating the behavior of whales.

4.3. Physics-Based Algorithms

The third type of metaheuristic algorithm includes physics-based techniques, replicating physical rules during optimization to discover the best. These techniques are motivated by the physical principles of nature. There are several popular algorithms, including simulated annealing (SA) is a metaheuristic algorithm that draws inspiration from the metallurgical annealing procedure. It solves optimization challenges by mimicking a material's cooling and crystallization. It is especially useful for issues involving rocky or multi-modal environments, in which there may be several local optima. The Lightning Search Algorithm (LSA) is a metaheuristic algorithm influenced by the natural factors of lightning strikes. It uses the unpredictable and strong nature of lightning to explore the search space and identify optimal solutions. It blends random search, local search, and global search algorithms to equalize exploration and exploitation for efficient optimization. The Gravitational Search Algorithm (GSA) is a metaheuristic algorithm influenced by gravity and motion principles. It simulates the interaction of celestial bodies in order to address optimization difficulties. It employs gravitational forces to attract candidate solutions to better portions of the search space and changes the placements based on mass and acceleration estimations. Electromagnetic Field Optimization (EFO) is a metaheuristic method based on electromagnetism principles. To tackle optimization issues, it simulates the behavior of charged particles and magnetic fields. EFO uses particle attraction and repulsion to direct the search process and converge on optimal solutions in the search space. Multiple optimization algorithms have been created that follow the principles of physics. Examples of these algorithms include the multi-verse optimizer, the sine–cosine algorithm, and the gravitational search algorithm. These algorithms have been designed to identify the best set of features among various datasets.

4.4. Human-Related Algorithms

These human-based metaheuristic algorithms are driven by social interaction or behavioral patterns in people. We present an overview of humanly rooted algorithms for resolving characteristic optimization situations. An overview of three algorithms is: The BSO (Brainstorm Optimization) algorithm functions like how people generate ideas, and it was also utilized for data classification. It solves optimization issues by iteratively creating, assessing, and refining potential solutions using a collaborative search process. Teaching-based learning optimization (TBLO), the teacher's influence over the class's students is the foundation of this algorithm. It integrates teacher and student concepts in order to explore the search space and identify optimal answers. To develop candidate solutions iteratively, it employs instructional tactics such as exploration, exploitation, and knowledge exchange. The Gaining Sharing Knowledge-Based Algorithm (GSKA) is a metaheuristic algorithm that uses knowledge sharing and acquisition among humans to solve optimization challenges. It encourages cooperation and information exchange to improve the search process, allowing the algorithm to successfully explore the search space and settle on ideal solutions. It is founded on the idea of people learning from one another and passing on their knowledge.

4.5. Hybrid Metaheuristic Algorithms

Hybrid algorithms have gained popularity recently for handling optimization issues. Many hybrid metaheuristic algorithms have been developed, specifically for the issue of feature selection to extract the pertinent and ideal subset of features from the original dataset. It is created by fusing the most effective operators from other metaheuristic algorithms. The enhanced technique helps remove local optimization trapping to avoid premature convergence, efficiently and effectively explore the search space, and achieving better usage.

Additionally, the upgraded algorithms achieve ideal or nearly optimal outcomes, striking superior balances between algorithmic search and utilization features. The best features of various algorithms are combined, to create new algorithms. Hybrid metaheuristics can provide greater convergence, solution quality, and efficiency by combining diverse methods.

A comparison of various categories of metaheuristic algorithms is shown in Table 1 below.

Table 1. Comparison of various categories of metaheuristic algorithms.

Algorithm Type	Classification	Fundamental Ideas	Applicability
Evolution-based	Genetic Algorithm (GA)	Genetic operators, population evolution	A wide range of optimization challenges
Swarm intelligence-based	Firefly Algorithm (FA)	Attraction and movement based on brightness	Problems in dynamic or evolving environments
Physics-based	Gravitational Search Algorithm (GSA)	Gravity, mass, acceleration, attraction	Problems in which physical analogies can be used
Human-related	Teaching-based Learning Optimization (TBLO)	Teaching strategies, collaboration, knowledge sharing	Domain-specific knowledge or constraint issues
Hybrid	Hybrid Metaheuristic Algorithms	Combination of multiple algorithms or techniques	Complex optimization issues with a wide range of characteristics

5. Related Research

A study by Negahbani et al. [4] utilized the differential search algorithm in conjunction with fuzzy c-means to diagnose coronary artery disease and achieved promising results in terms of accuracy and sensitivity. The binary-operating backtracking algorithm designed by Zhang et al. [5] leveraged the power of extreme learning machines for wind speed forecasting. This algorithm converts continuous variables into binary variables using a sigmoidal function. Dhal et al. [6] evaluated the speculative fractal forage algorithm to optimize the recognition of leukemia cancer symptoms and compared it to classical methods, with impressive results. Galaxy color images were recognized using extreme machine learning, and a bipartite debatable fractal forage was additionally evolved. These examples illustrate the potential and versatility of evolution-based algorithms in solving complex optimization problems.

In this field of study, the work of Nakamura et al. [7] involved the development of a binary version of the BAT algorithm. A sigmoid function is used to convert the BAT locations to binary variables. The Optimum Path Forest classifier was applied to assess the algorithm's accuracy over five datasets. To achieve a finer equilibrium between the algorithm research and ill-treatment, Sayed et al. [8] developed the Chaotic Whale Optimization Algorithm (CWOA), which used 10 disordered plans in place of random parameters. Rodrigues et al. [9] proposed the Binary Cuckoo Search (BCS) algorithm, by employing a function that turns continuous variables into binary form. This was tested on two datasets of power system theft detection using the Optimum Path Forest classifier and proved to be the fastest and most suitable method for component-based nomination about commercial datasets. Pandey et al. [10] inaugurated the Binary Binomial Cuckoo Search algorithm to identify the finest performed functions and further appertained it to more than 10 various data sets of utmost criticalities from the UCI repository. There have been numerous implementations consisting of machine learning that lately have been resolved by progressing various categories of the CS algorithm. Huang et al. [11] suggested a hybrid approach called HGOA by combining GOA with an AI-based bee colony algorithm (ABC) to decode the feature selection problems. The fitness function was based on the classification accuracy with the KNN classifier. The proposed approach was gauged on the criterion data sets belonging to various UCI repositories. They introduced a hybrid GOA with a differential evolution algorithm (DGOA) to discover the capital attribute subset for classification problems. Moreover, the proposed method was evaluated regarding five standardized datasets conveyed by the UCI repository and showed vast improvement in results when juxtaposed with various algorithms. Emary et al. [12] were the first to implement a binary version of the FFA, utilizing a threshold value to achieve efficient exploration quality and fast solution discovery when applied to UCI benchmark datasets with a KNN classifier. To enhance performance, Kanimozhi and Latha [13] utilized the FFA and SVM classifier for optimal feature selection in image retrieval, testing the technique on Corel Caltech and Pascal database images. In the medical field, Subha and Murugan [14] employed the FFA with SVM on cardiotocography data to predict diseases. Medjahed et al. [15] leveraged the binary dragonfly (BDF) algorithm along with a support vector machine (SVM) to develop a comprehensive cancer diagnosis procedure. The SVM-recursive feature elimination (SVM-RFE) method was utilized to extract relevant genes coming out of the data set, and BDF was introduced to increase the execution and production of the SVM-RFE. Further, the suggested approach demonstrated exceptional accuracy results when evaluated on six microarray datasets. Mafarja et al. [16] suggested a duplex description of the dragonfly algorithm (BDA) that utilized a transfer function to solve feature selection problems. To strike an equilibrium between exploration and exploitation, the researchers created a binary version of the approach that used time-varying transfer functions. These techniques were used on datasets from the UCI repository and compared against pioneering metaheuristic optimizations. Sharma et al. [17] developed a variant of the GWO for classifying the signs of Parkinson's disease. Another iteration of the GWO, known as the levy flight GWO, was proposed by Pathak et al. [18] This version of the algorithm was used to excerpt pertinent functions from datasets, and the arbitrary Holt classifier was petitioned to the Bossbasever 1.01 dataset for image steganalysis. The results obtained from this version showed exceptional performance in terms of convergence. The ABGWO (Advanced GWO) algorithm was developed by Hu et al. [19] utilizing emerging convey tasks and an improved method for changing the GWO's parameters. Twelve datasets from the UCI repository were used to test this modified version, which produced superior outcomes than existing algorithms. Rodrigues et al. [20] suggested a binary-constrained version of FPA, referred to as BFPA, that utilizes a local pollination generation to produce a binary solution. The BFPA was tested using the Optimum Path Forest classifier to determine its accuracy and was found to perform as well as other well-established metaheuristic algorithms such as PSO, HS, and FA. To enhance the presentation of BFPA, Zawbaa and Emary [21] utilized a KNN probabilistic with a modern bipartite alternative of FPA. It required the transformation of dependent variables into binary strings using a threshold. The results from this version showed superior performance compared to other algorithms like PSO, GA, and BA. ABFPA, an adapted version of BFPA, was proposed by utilizing different values of the λ parameter to deepen its adaptation scheme. Using continuous variable thresholds, Zawbaa et al. [22] suggested a binary version of the ALO technique. They tested the suggested approach BALO with K-NN classifiers on 18 distinct datasets and compared the outcomes to those of more well-known metaheuristic algorithms namely genetic algorithms and Particle Swarm Optimization. They calculated performance using various metrics such as accuracy in average classification, numerable features spotted on an average and a Fisher score (F-score) mean. The optimization approach provided by Emary et al. [23] comes in a variety of iterations where each element moves by the intersection operator across two binary choices. These are obtained by applying transfer functions (like S-shaped and V-shaped) or implying the fundamental operator. Furthermore, three initialization methods were employed to properly explore the forage expanse, and it was finalized that the introductory procedure affects the exploration standard and algorithm presentation. Hussien et al. [24] utilized transfer functions of S and V shapes in their standard WOA to address the binary optimization issue, and in 2017, they applied it to the feature selection problem using 11 UCI datasets. To assess the relevance of the selected features, the study used a KNN classifier. The WBA approach demonstrated its ability to achieve both the greatest correctness and the least amount of selected attributes. Tubishat et al. [25] applied an improved WOA to the sentiment analysis of an Arabic dataset. IWOA integrated developmental machinists such as crosswalk, alteration, selection, and differentiation evolution and was evaluated on four openly obtainable sets of data in comparison with distinct approaches.

Papa et al. [26] combined binary BSO with numerous S- and V-shaped transfer functions. This approach was tested on several datasets from Arizona State University and then subjected to the Optimum Path Forest Classifier. They utilized a fuzzy min-max neural network learning model with a binary BSO method for real-world datasets. A fuzzy ARTMAP model utilizing the BSO method was also introduced by them. For medical categorization, Tuba et al. [27] utilized their BSO algorithm with their SVM classifier. Furthermore, the BSO algorithm has been used to enhance the SVM parameters. Due to its higher search quality, Oliva and Elaziz [28] suggested a new iteration of the BSO algorithm. The solution was generated using a chaos map and an inverse learning algorithm. To update the initial population, the disruptor operator was employed. Eight datasets from the UCI repository were taken into consideration for the best characterization, and a revised version was adopted for classification. The optimal features were chosen using the TLBO technique and SVM classifier on the image retrieval dataset by Jain and Bhadauria [29]. A better iteration of his TLBO technique with a wavelet transform function was put out by Krishna and Vishwakarma to recognize fingerprints. An approach for multi-objective TLBO feature selection in binary classification tasks was proposed by Balakrishnan [30]. The software was evaluated using the known UCI dataset using three supervised learning techniques. TLBO-based logistic regression produced the best results across the majority of datasets among the three categorization models. A binary TLBO (BTLBO) was created by Allam and Nandhini [31] with a threshold to limit variables to binary form. To categorize datasets related to breast cancer, they have employed various classifiers. With fewer features, the suggested method demonstrated great accuracy. On the dataset for chronic renal illness, a better iteration of the TLBO method was used. The Chebyshev distance formula was used to evaluate fitness functions, and the results were obtained. By including binary junior and senior extraction and division stages, Agrawal et al. [32] presented the first binary variant of his GSK method for feature selection problems (FS-NBGSK). Using the KNN classifier, 23 benchmark datasets from the UCI repository were used to test the FS-NBGSK algorithm. This technique surpassed the others in terms of accuracy and the least number of characteristics used.

The MAKHA method was introduced by Hafez et al. [33], in which the krill swarm algorithm's evolutionary operators (mutation and crossover) are combined with the monkey algorithm's leaping process to discover the best solution rapidly. The algorithm's classification accuracy was tested using ANN classifiers on 18 UCI data sets. The most well-liked and promising method in the physics-based category is simulated annealing (SA). Mafarja and Mirjalili [34] added SA to their WOA to enhance the performance of the whale optimization algorithm. By enhancing the top choice made after each iteration, they enhanced the adoption of WOA. Using ANN classifiers, 18 data sets were used to examine the performance of the WOA–SA hybrid method. To achieve a fair equilibrium between exploration and exploitation, Arora et al. [35] used the location update quality of the crow search algorithm in the gray wolf optimizer. They hybridized the algorithm to apply GWOCSA on the 21 entries from their UCI repository that were known to exist. The GWOCSA algorithm constrains the binary search space using a sigmoidal transfer function. In comparison to other cutting-edge metaheuristic algorithms, the accuracy of the ANN classifier under consideration was assessed. Abd Elaziz et al. [36] suggested a hybrid approach using the local search method of the differential evolution algorithm to get rid of local optima in the sine/cosine algorithm. Eight data sets from UCI were used to evaluate an enhanced sine/cosine algorithm, which performed better in terms of statistical analysis and power measurements. The feature selection problem in binary space was solved by Tawhid and Dsouza [37] using a hybrid algorithm that combined an enhanced Bat-His algorithm with the PSO method. They employed an S-shaped transfer function to acquire the binary positions of the particles in his PSO method and a V-shaped transfer function to change the position of the bat in binary space. The best characteristics of the 20 common data sets are preserved by the hybrid algorithm, which combines the BAT algorithm's efficient search with the PSO method's convergence characteristics. In comparison to other algorithms, the acquired findings demonstrated the ability to have great accuracy. To enhance EPO's performance, Balarsingh added a social engineering optimizer. In the suggested hybrid strategy, the SVM classifier is altered using the meme algorithm and applied to the medical dataset. When compared against other well-known metaheuristic algorithms, the suggested hybrid approach excels them all. The Cultural Face Recognition Algorithm proposes yet another hybrid method for EPO. The proposed approach improves the performance of existing approaches and is applied to their SVM classifier for face recognition and showed the best results. Shukla et al. [38] integrate the SA approach with supervised learning-based optimization to find the best genes from gene expression data. The TLBO algorithm's solution quality was enhanced by the SA algorithm, which also assisted in the discovery of genes related to cancer detection. Additionally, a brand-new transfer function with a V shape is suggested to change the variables into binary variables. On ten sets of microarray datasets, classification accuracy was assessed using the SVM classifier. To address various applications of feature selection problems, numerous combinations of various metaheuristic algorithms have been created. In gene selection, the Jaya algorithm is coupled with forest optimization methods by adjusting the two parameters of the forest optimization technique, extended JA is used. On microarray data sets, this hybrid strategy performed better than other optimizers. Text feature selection was carried out using the gray wolf optimizer and his locust optimization technique and industrial form injection was conducted using the PSO and gravity search algorithms. For the feature selection, a hybrid of the locust and cat herd optimization method and the grey wolf and probabilistic fractal search algorithm is employed. Table 2 summarizes the algorithms used, applications, and outcomes of numerous investigations conducted by the authors.

Table 2. Literature Survey.

Author	Year	Algorithm Used	Application	Outcome
Nakamura et al. [7]	2012	Binary BAT Algorithm (BBA)	Feature Selection	Enhanced feature selection.
Rodrigues et al. [9]	2013	Binary Cuckoo Search (BCS)	Power System Theft Detection	Fastest and most appropriate for commercial datasets.
Negahbani et al. [4]	2015	Differential Search Algorithm	Diagnosing Coronary Artery Disease	Improved disease diagnosis precision.
Kanimozhi and Latha [13]	2015	Firefly Algorithm (FFA) and Support Vector Machine (SVM)	Region-Based Image Retrieval	Image recovery with optimal feature selection.
Rodrigues et al. [20]	2015	Binary Flower Pollination Algorithm (BFPA)	Feature Selection Problems	improved performance in feature selection.
Zawbaa et al. [22]	2015	Binary Artificial Life Optimization (BALO)	Various Datasets	In contrast to GA and PSO, continuous variable thresholds.
Hafez et al. [33]	2015	Hybrid Monkey Algorithm with Krill Herd Algorithm Optimization	Classification Accuracy	Improved solution finding by combining krill swarm and monkey algorithms.
Subha and Murugan [14]	2016	Firefly Algorithm (FFA)	Cardiotocography Data	Disease prognosis.
Emary et al. [23]	2016	Binary Ant Lion Approaches	Feature Selection	Enhanced performance in feature selection.
Jain and Bhadauria [29]	2016	Teacher Learning-Based Optimization (TLBO)	Enhanced Content-Based Image Retrieval	TLBO and SVM classifiers are used to optimize feature selection.

Table 2. *Cont.*

Author	Year	Algorithm Used	Application	Outcome
Zhang et al. [5]	2017	Hybrid Backtracking Search Algorithm	Wind Speed Forecasting	Extreme learning machines were used, and good results were obtained.
Medjahed et al. [15]	2017	Binary Dragonfly Algorithm (BDF)	Cancer Diagnosis	Increased execution and production of SVM-RFE.
Abd Elaziz et al. [36]	2017	Hybrid Sine/Cosine Algorithm with Differential Evolution (DE)	Feature Selection Problem	Local optima were eliminated, and statistical analysis and power measurements were improved.
Mafarja et al. [16]	2018	Binary Dragonfly Algorithm (BDA)	Feature Selection Problems	When compared to metaheuristic optimization, time-varying transfer functions.
Zawbaa and Emary [21]	2018	Flower Pollination Algorithm	Feature Selection, Knapsack Problems	Superior than other algorithms in performance.
Sayed et al. [8]	2018	Chaotic Whale Optimization Algorithm (CWOA)	Feature Selection	increased harmony between algorithm research and exploitation.
Papa et al. [26]	2018	Binary Brain Storm Optimization (BBSO)	Real-world datasets	Improved categorization using the fuzzy min-max neural network learning model.
Sharma et al. [17]	2019	Gray Wolf Optimization (GWO)	Diagnosis of Parkinson's Disease	Improved diagnostic performance with a Parkinson's disease classification variant.
Pathak et al. [18]	2019	Levy Flight-Based Gray Wolf Optimization	Image Steganalysis	Outstanding convergence performance.
Hussien et al. [24]	2019	Whale Optimization Algorithm (WOA)	Feature Selection Problem	Achieved excellent accuracy while reducing characteristics.
Tubishat et al. [25]	2019	Improved Whale Optimization Algorithm (WOA)	Sentiment Analysis	Improved sentiment analysis using the Arabic dataset
Tuba et al. [27]	2019	Brain Storm Optimization Algorithm and Support Vector Machine (SVM)	Medical Categorization	SVM classifier integration, improved SVM parameters.
Mafarja and Mirjalili [34]	2019	Hybrid Binary Ant Lion Optimizer with Rough Set and Approximate Entropy Reducts	Performance Enhancement	The hybrid algorithm improves adoption and performance.
Arora et al. [35]	2019	Hybrid Gray Wolf Optimizer with Crow Search Algorithm (GWOCSA)	Unconstrained Function Optimization, Classification Accuracy	The hybrid algorithm increased accuracy.
Tawhid and Dsouza [37]	2019	Bat-His Algorithm with Particle Swarm Optimization (PSO)	Feature Selection Problem	A hybrid algorithm with efficient search and convergence features.
Shukla et al. [38]	2019	Hybrid Wrapper TLBO and SA with SVM Approach	Gene Expression Data	Enhanced cancer diagnosis with enhanced TLBO solution quality.
Dhal et al. [6]	2020	Speculative Fractal Forage	Optimizing Leukemia Cancer Symptom Recognition	When compared to traditional approaches, the outcomes are remarkable.
Pandey et al. [10]	2020	Binary Binomial Cuckoo Search	Various Data Sets	Best-performing functions have been identified.

Table 2. *Cont.*

Author	Year	Algorithm Used	Application	Outcome
Hu et al. [19]	2020	Advanced GWO (ABGWO)	Various UCI Datasets	Better results than existing algorithms.
Oliva and Elaziz [28]	2020	Improved BSO Algorithm	Eight Datasets from UCI Repository	increased categorization and enhanced search quality.
Balakrishnan [30]	2020	Multi-objective TLBO Feature Selection	Binary Classification Tasks	Models such as logistic regression, SVM, and ELM performed better.
Allam and Nandhini [31]	2022	Binary TLBO (BTLBO)	Breast Cancer Dataset	Excellent precision with fewer characteristics.

6. Research Gaps

The field of algorithms based on physical principles, natural evolution and human behavior remains largely underexplored. A significant gap exists in the development of binary versions of algorithms that take into account natural evolution and human activities. A binary variant of swarm-based algorithms like the Egyptian vulture optimization, paddy field algorithm, eagle strategy, bird mating optimizer, hierarchical swarm optimization, Japanese tree frogs calling algorithm, great salmon run algorithm, shark smell optimization, spotted hyena optimizer, and emperor penguin's colony has not yet been proposed. Similarly, in the realm of physics-based algorithms, there is a lack of research on binary versions of galaxy-based search algorithms, curved space optimization, ray optimization, lightning search, thermal exchange optimization, and find-fix finish exploit analysis. Furthermore, human-related algorithms, such as the league championship algorithm and human-inspired algorithm, as well as social-emotional optimization, have yet to be adapted to solve feature selection problems.

In addition to exploring the possibility of developing binary variants of metaheuristic algorithms, researchers can also examine the potential of using new and innovative S- and V-shaped transfer functions. The area of application of these algorithms remains underutilized, with only a limited number of researchers exploring the potential of metaheuristics in stock market prediction, short-term load forecasting, weather prediction, spam detection, and Parkinson's disease. Furthermore, the existing literature primarily focuses on two objectives in feature selection, namely, maximizing accuracy and minimizing the number of selected features. However, it may be worthwhile for researchers to consider other goals, such as computational time, complexity, stability, and scalability, in multi-objective feature selection.

7. Practical Applications

As we have seen earlier, the reason behind a surge of interest in metaheuristics is to solve real-world optimization problems that are otherwise difficult to solve. We often come across optimization problems in engineering and other domains that present a vast and difficult search space. To find a helpful solution in such cases, using traditional approaches proves to be inefficient. Metaheuristics have been effectively used to tackle well-known combinatorial issues such as the traveling salesman problem since its inception. We have also seen applications of these algorithms in a wide range of domains, like education, robotics, medical diagnosis, sentiment analysis, finance, and fraud detection to name a few. Metaheuristic articles published in different domains are illustrated in Figure 4 below.

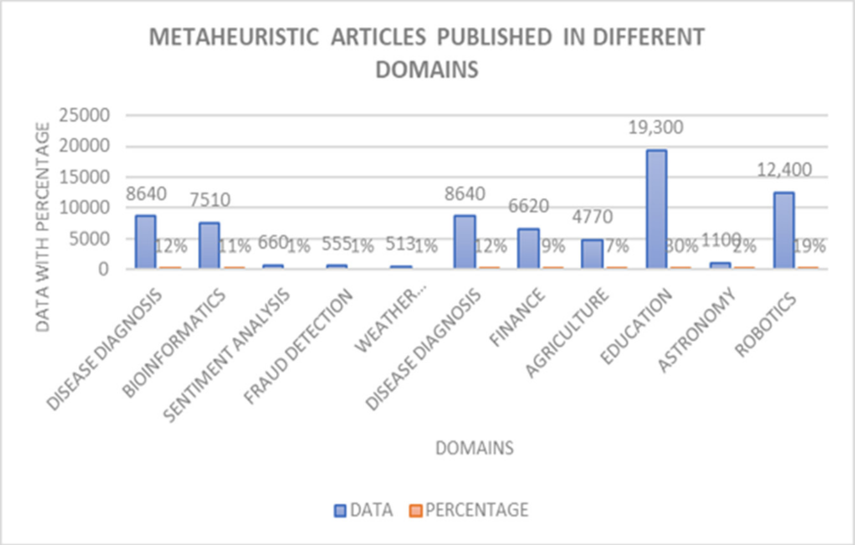

Figure 4. Metaheuristic articles published in different domains.

It is important to note that a metaheuristic takes very few assumptions about optimization problems. Hence, they apply to a vast variety of problems. But, at the same time, it does not guarantee the same level of performance for all these problems. Hence, we must make specific alterations in the algorithm to make it more suitable for particular problems. This has resulted in numerous variations in the common nature-inspired metaheuristics that we have seen in this tutorial. It is much beyond the scope of this tutorial to even name all of them! Further, a lot of research goes into fine-tuning the parameters of each of these algorithms that can make them suitable for a specific problem domain. Finally, it is important to note that while we have developed a lot of intuition behind these algorithms, they largely work like black boxes. So, it is challenging to predict which algorithms in some specific form can work better for an optimization problem. As we keep discovering new problems and demand better performance for existing ones, we have to keep investing in research.

8. Challenges in Metaheuristics

Metaheuristic algorithms have been successful in resolving several real-world issues, as we have learned from this review. However, several difficult issues with metaheuristics must be addressed. Yan noted that the theoretical study of these algorithms currently lacks a coherent framework and has numerous unanswered difficulties. For example, how do algorithm-dependent parameters affect algorithm performance? For metaheuristic algorithms to operate as effectively as possible, what is the ideal ratio between exploration and exploitation? What benefits may an algorithm gain from using algorithmic memory? Since metaheuristic applications are growing quickly before mathematical analysis, the gap between theory and practice is another significant issue. However, the majority of applications involve modest issues. Large-scale applications and research should be prioritized in the future. Contrarily, there are a lot of new algorithms, but having more algorithms makes it more challenging to comprehend how metaheuristics operate in general. To comprehend all metaheuristics more thoroughly, we might require a uniform method for algorithm analysis, preferably for the classification of these algorithms. These challenges also provide timely and hot research opportunities for researchers to make significant progress shortly.

9. Conclusions and Future Scope

Metaheuristic algorithms are capable of solving complicated optimization issues in a wide range of fields. While much high-quality research has been undertaken in this area, most literature remains largely experimental. Although the literature claims novelty and practical efficacy, they may not prove to be practical for real-world engineering problems. It is for us to complete a rigorous exercise to understand their value. Nevertheless, we should continue to invest and improve in metaheuristics. There is a lot of cross-over between the areas of study that inspires a metaheuristic and hence it is bound to be quite complex. In this paper, we have discussed the basics of nature-inspired metaheuristics and why we even need them. Although the spectrum of these algorithms is quite wide, we focused on some of the well-known algorithms in the category of evolutionary algorithms and swarm algorithms. The goal of this study is to learn about the most recent breakthroughs in metaheuristic algorithms, with a particular emphasis on research on the global state from 2012 to 2022. The writers endeavored to grasp the algorithms, applications, and outcomes of studies. This paper also discussed some of the challenges of metaheuristic algorithms. Finally, we discussed some of the practical applications of these metaheuristic algorithms. The purpose of this review is to present a comparative and comprehensive list of all the algorithms in the literature, to inspire further vital research.

Author Contributions: Conceptualization, V.T. and M.B.; methodology, V.T. and P.S.; validation, V.T. and P.S.; formal analysis, M.B., V.T. and P.S.; investigation, V.T.; resources, V.T.; writing—original draft preparation, V.T.; writing—review and editing, M.B. and P.S.; supervision, M.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All the data used are made available in the present work.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Kirkpatrick, S.; Gelatt, C.D., Jr.; Vecchi, M.P. Optimization by simulated annealing. *Science* **1983**, *220*, 671–680. [CrossRef] [PubMed]
2. Fister Jr, I.; Mlakar, U.; Brest, J.; Fister, I. A new population-based nature-inspired algorithm every month: Is the current era coming to the end? In Proceedings of the 3rd Student Computer Science Research Conference, Ljubljana, Slovenia, 12 October 2016; pp. 33–37.
3. Gan, L.; Duan, H. Biological image processing via chaotic differential search and lateral inhibition. *Optik* **2014**, *125*, 2070–2075. [CrossRef]
4. Negahbani, M.; Joulazadeh, S.; Marateb, H.; Mansourian, M. Coronary artery disease diagnosis using supervised fuzzy c-means with differential search algorithm-based generalized Minkowski metrics. *Peertechz J. Biomed. Eng.* **2015**, *1*, 6–14. [CrossRef]
5. Zhang, C.; Zhou, J.; Li, C.; Fu, W.; Peng, T. A compound structure of ELM based on feature selection and parameter optimization using hybrid backtracking search algorithm for wind speed forecasting. *Energy Convers. Manag.* **2017**, *143*, 360–376. [CrossRef]
6. Dhal, K.G.; Gálvez, J.; Ray, S.; Das, A.; Das, S. Acute lymphoblastic leukemia image segmentation driven by stochastic fractal search. *Multimedia Tools Appl.* **2020**, *79*, 12227–12255. [CrossRef]
7. Nakamura, R.Y.; Pereira, L.A.; Costa, K.A.; Rodrigues, D.; Papa, J.P.; Yang, X.S. BBA: A binary bat algorithm for feature selection. In Proceedings of the 2012 25th SIBGRAPI Conference on Graphics, Patterns and Images, Ouro Preto, Brazil, 22–25 August 2012; pp. 291–297. [CrossRef]
8. Sayed, G.I.; Darwish, A.; Hassanien, A.E. A new chaotic whale optimization algorithm for features selection. *J. Classif.* **2018**, *35*, 300–344. [CrossRef]
9. Rodrigues, D.; Pereira, L.A.; Almeida, T.N.S.; Papa, J.P.; Souza, A.N.; Ramos, C.C.; Yang, X.S. BCS: A binary cuckoo search algorithm for feature selection. In Proceedings of the 2013 IEEE International Symposium on Circuits and Systems (ISCAS), Beijing, China, 19–23 May 2013; pp. 465–468. [CrossRef]
10. Pandey, A.C.; Rajpoot, D.S.; Saraswat, M. Feature selection method based on hybrid data transformation and binary binomial cuckoo search. *J. Ambient. Intell. Humaniz. Comput.* **2020**, *11*, 719–738. [CrossRef]
11. Huang, J.; Li, C.; Cui, Z.; Zhang, L.; Dai, W. An improved grasshopper optimization algorithm for optimizing hybrid active power filters' parameters. *IEEE Access* **2020**, *8*, 137004–137018. [CrossRef]
12. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary ant lion approaches for feature selection. *Neurocomputing* **2016**, *213*, 54–65. [CrossRef]

13. Kanimozhi, T.; Latha, K. An integrated approach to region based image retrieval using firefly algorithm and support vector machine. *Neurocomputing* **2015**, *151*, 1099–1111. [CrossRef]
14. Subha, V.; Murugan, D. Opposition based firefly algorithm optimized feature subset selection approach for fetal risk anticipation. *Mach. Learn. Appl. Int. J.* **2016**, *3*, 55–64. [CrossRef]
15. Medjahed, S.A.; Saadi, T.A.; Benyettou, A.; Ouali, M. Kernel-based learning and feature selection analysis for cancer diagnosis. *Appl. Soft Comput.* **2017**, *51*, 39–48. [CrossRef]
16. Mafarja, M.; Aljarah, I.; Heidari, A.A.; Faris, H.; Fournier-Viger, P.; Li, X.; Mirjalili, S. Binary dragonfly optimization for feature selection using time-varying transfer functions. *Knowl. Based Syst.* **2018**, *161*, 185–204. [CrossRef]
17. Sharma, P.; Sundaram, S.; Sharma, M.; Sharma, A.; Gupta, D. Diagnosis of Parkinson's disease using modified grey wolf optimization. *Cogn. Syst. Res.* **2019**, *54*, 100–115. [CrossRef]
18. Pathak, Y.; Arya, K.V.; Tiwari, S. Feature selection for image steganalysis using levy flight-based grey wolf optimization. *Multimed. Tools Appl.* **2019**, *78*, 1473–1494. [CrossRef]
19. Hu, P.; Pan, J.S.; Chu, S.C. Improved binary grey wolf optimizer and its application for feature selection. *Knowl. -Based Syst.* **2020**, *195*, 105746. [CrossRef]
20. Rodrigues, D.; Yang, X.S.; De Souza, A.N.; Papa, J.P. Binary flower pollination algorithm and its application to feature selection. In *Recent Advances in Swarm Intelligence and Evolutionary Computation*; Springer: Cham, Switzerland, 2015; pp. 85–100. [CrossRef]
21. Zawbaa, H.M.; Emary, E. Applications of flower pollination algorithm in feature selection and knapsack problems. In *Nature-Inspired Algorithms and Applied Optimization*; Springer: Cham, Switzerland, 2018; pp. 217–243. [CrossRef]
22. Zawbaa, H.M.; Emary, E.; Parv, B. Feature selection based on antlion optimization algorithm. In Proceedings of the 2015 Third World Conference on Complex Systems (WCCS), Marrakech, Morocco, 23–25 November 2015; pp. 1–7. [CrossRef]
23. Emary, E.; Zawbaa, H.M.; Hassanien, A.E. Binary grey wolf optimization approaches for feature selection. *Neurocomputing* **2016**, *172*, 371–381. [CrossRef]
24. Hussien, A.G.; Hassanien, A.E.; Houssein, E.H.; Bhattacharyya, S.; Amin, M. S-shaped binary whale optimization algorithm for feature selection. In *Recent Trends in Signal and Image Processing: ISSIP 2017*; Springer: Singapore, 2017; pp. 79–87. [CrossRef]
25. Tubishat, M.; Abushariah, M.A.; Idris, N.; Aljarah, I. Improved whale optimization algorithm for feature selection in Arabic sentiment analysis. *Appl. Intell.* **2019**, *49*, 1688–1707. [CrossRef]
26. Papa, J.P.; Rosa, G.H.; de Souza, A.N.; Afonso, L.C. Feature selection through binary brain storm optimization. *Comput. Electr. Eng.* **2018**, *72*, 468–481. [CrossRef]
27. Tuba, E.; Strumberger, I.; Bezdan, T.; Bacanin, N.; Tuba, M. Classification and feature selection method for medical datasets by brain storm optimization algorithm and support vector machine. *Procedia Comput. Sci.* **2019**, *162*, 307–315. [CrossRef]
28. Oliva, D.; Elaziz, M.A. An improved brainstorm optimization using chaotic opposite-based learning with disruption operator for global optimization and feature selection. *Soft Comput.* **2020**, *24*, 14051–14072. [CrossRef]
29. Jain, K.; Bhadauria, S.S. Enhanced content-based image retrieval using feature selection using teacher learning based optimization. *Int. J. Comput. Sci. Inf. Secur. (IJCSIS)* **2016**, *14*, 1052–1057.
30. Balakrishnan, S. Feature selection using improved teaching learning based algorithm on chronic kidney disease dataset. *Procedia Comput. Sci.* **2020**, *171*, 1660–1669. [CrossRef]
31. Allam, M.; Nandhini, M. Optimal feature selection using binary teaching learning-based optimization algorithm. *J. King Saud Univ. Comput. Inf. Sci.* **2022**, *34*, 329–341. [CrossRef]
32. Agrawal, P.; Abutarboush, H.F.; Ganesh, T.; Mohamed, A.W. Metaheuristic algorithms on feature selection: A survey of one decade of research (2009–2019). *IEEE Access* **2021**, *9*, 26766–26791. [CrossRef]
33. Hafez, A.I.; Hassanien, A.E.; Zawbaa, H.M.; Emary, E. Hybrid monkey algorithm with krill herd algorithm optimization for feature selection. In Proceedings of the 2015 11th International Computer Engineering Conference (ICENCO), Cairo, Egypt, 29–30 December 2015; pp. 273–277. [CrossRef]
34. Mafarja, M.M.; Mirjalili, S. Hybrid binary ant lion optimizer with rough set and approximate entropy reducts for feature selection. *Soft Comput.* **2019**, *23*, 6249–6265. [CrossRef]
35. Arora, S.; Singh, H.; Sharma, M.; Sharma, S.; Anand, P. A new hybrid algorithm based on grey wolf optimization and crow search algorithm for unconstrained function optimization and feature selection. *IEEE Access* **2019**, *7*, 26343–26361. [CrossRef]
36. Abd Elaziz, M.E.; Ewees, A.A.; Oliva, D.; Duan, P.; Xiong, S. A hybrid method of sine cosine algorithm and differential evolution for feature selection. In *Neural Information Processing: 24th International Conference, ICONIP 2017, Guangzhou, China, 14–18 November 2017*; Proceedings, Part V 24; Springer International Publishing: Cham, Switzerland, 2017; pp. 145–155. [CrossRef]
37. Tawhid, M.A.; Dsouza, K.B. Solving feature selection problem by hybrid binary genetic enhanced particle swarm optimization algorithm. *Int. J. Hybrid Intell. Syst.* **2019**, *15*, 207–219. [CrossRef]
38. Shukla, A.K.; Singh, P.; Vardhan, M. A new hybrid wrapper TLBO and SA with SVM approach for gene expression data. *Inf. Sci.* **2019**, *503*, 238–254. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.