International Journal of Scientific World, *11 (1) (2025) 83-92*

International Journal of Scientific World

Website: www.sciencepubco.com/index.php/IJSW
https://doi.org/10.14419/d5pxkg39
Research paper

A comprehensive review of metaheuristic algorithms for combinatorial optimization problems

Nazik Saber Rashid [1] *, Ibrahim M. I. Zebari [2]

[1] Akre University for Applied Science, Technical College of Informatics, Akre, Department of Information Technology, Akre, Kurdistan Region, Iraq
[2] Akre University for Applied Sciences, Technical College of Informatics-Akre, Department of Computer Networks and Information Security, Akre, Kurdistan Region, Iraq
**Corresponding author E-mail: naziksufy@gmail.com*

Abstract

Metaheuristic algorithms are essential for handling difficult combinatorial optimization issues that arise in a variety of domains such as engi-neering, logistics, and operations research. These algorithms, based on natural, social, and physical events, strike a compromise between computing efficiency and solution quality. This study divides metaheuristic approaches into three categories: evolutionary algorithms, swarm intelligence techniques, and physics-based models, with a focus on current advances like hybrid and AI-driven frameworks. It also examines issues like as standardization, scalability, and practical implementation, including examples such as the Fire Hawk Optimizer to demonstrate its uses. This study intends to lead the development of trustworthy and efficient metaheuristic algorithms to solve increasingly complicated optimization issues in real-world settings, integrating theoretical ideas and practical examples.

Keywords: *Metaheuristic Algorithms; Combinatorial Optimization; Evolutionary Algorithms; Swarm Intelligence; Hybrid Optimization Frameworks.*

1. Introduction

The fast growth of computer tools has fundamentally changed the landscape of optimization, addressing difficulties in engineering, operations research, logistics, and beyond. Combinatorial optimization problems (COPs) stand out for their intrinsic complexity and widespread use in real-world applications such as scheduling, routing, and resource allocation. NP-hard problems are difficult to solve on time, especially as their dimensions increase exponentially[1 - 3].

Metaheuristic algorithms have emerged as a key component in addressing COPs, providing efficient and scalable techniques to identifying near-optimal solutions. Metaheuristics are high-level frameworks inspired by natural processes or social behaviors. They balance exploration and exploitation in large search areas [4 - 6]. Their effectiveness stems from their ability to avoid exhaustive enumeration of solutions, instead relying on stochastic processes to arrive at satisfying results.

This paper does a thorough study of metaheuristic algorithms, dividing them into main paradigms based on their inspirations and operating principles. These paradigms include evolutionary algorithms like Genetic Algorithms (GAs) and Differential Evolution (DE), swarm intelligence techniques like Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO), and physics-based models like Simulated Annealing (SA) and Gravitational Search Algorithms (GSA). Hybrid methods and hyper-heuristics, which combine numerous methodologies, demonstrate the field's adaptability and inventiveness [1], [2], [3], [5].

Despite their exceptional success, metaheuristic approaches have limits. Ongoing research into algorithmic refinement and benchmarking is necessary due to persistent challenges such as premature convergence, reliance on parameter adjustment, and computing needs [4 - 6]. The "No Free Lunch" theorem underlines the necessity for context-specific adjustments, as there is no universal method that excels across all issue types[4], [5].

This study aims to give an in-depth look at the theoretical foundations, practical applications, and current advances in metaheuristic algorithms for combinatorial optimization. This study attempts to predict future innovation and application pathways by combining major contributions and identifying existing research gaps. In doing so, it reinforces metaheuristics' critical role in bridging the gap between computational feasibility and optimality, meeting the expanding needs of complicated, real-world optimization problems.

2. Background theory

Combinatorial optimization problems (COPs) pose a substantial challenge in operations research and applied mathematics due to their discrete character and computing complexity. These issues frequently occur in disciplines such as logistics, scheduling, telecommunications, and artificial intelligence, where the goal is to maximize a given objective function over a finite but exponentially huge range of

Copyright © Nazik Saber Rashid, Ibrahim M. I. Zebari. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

plausible solutions. Classical examples include the Traveling Salesman Problem (TSP), the Knapsack Problem, and the Job Scheduling Problem. All of these are usually NP-hard, rendering exact solutions infeasible for large cases.[7 - 9]

2.1. Characteristics of combinatorial optimization problems

COPs are characterized as a set of decision factors, restrictions, and an objective function that must be maximized or minimized. The solutions are discrete, frequently necessitating permutations, combinations, or groupings of parts. These issues have enormous solution spaces, making it computationally prohibitive to evaluate all possible solutions as the problem size rises [7], [10]. Real-world applications add complexity including multi-objective requirements, dynamic limitations, and data uncertainty, making the issue landscape even more difficult [8], [9].

2.2. Metaheuristic algorithms: overview

Metaheuristic algorithms have developed as viable methods for addressing COPs by delivering approximate answers in reasonable computing time. These algorithms use a combination of local search and global exploration to explore complicated solution landscapes and avoid local optima [7], [11], [12]. Metaheuristics balance exploration with exploitation, making them ideal for large-scale real-world problems where computing feasibility is more important than optimality [7], [12].

2.3. Key metaheuristic paradigms

Metaheuristic algorithms are often categorized. The primary paradigms in the discipline are represented by the following categories:

- Evolutionary algorithms (EAs): These algorithms use operators like selection, crossover, and mutation to develop a population of potential solutions. They are based on Darwinian ideas of natural selection. Examples of optimization problems that need robust exploration of solution spaces include Genetic Algorithms (GA) and Differential Evolution (DE) [9], [13].
- Swarm Intelligence (SI): SI algorithms make use of decentralized and self-organizing principles and are based on the collective behavior of social creatures like ants, bees, and birds. Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO) are well-known examples that are often used to solve scheduling, routing, and resource allocation issues [7], [12].
- Physics-Based Methods: These algorithms mimic physical processes, including gravitational interactions in the Gravitational Search Algorithm (GSA) or the annealing process in Simulated Annealing (SA). They are very good at using adaptive mechanisms and probabilistic transitions to get out of local optima [7], [9].
- Hybrid and hyper-Heuristics: These methods use high-level strategies or combine many algorithms to dynamically manage and adjust heuristic techniques. While hyper-heuristics concentrate on automating the construction and adjustment of heuristics, hybrid algorithms frequently combine the advantages of many approaches [11], [12].

As shown in Figure 1, these paradigms are systematically categorized to provide a clearer understanding of their distinctions and applications.

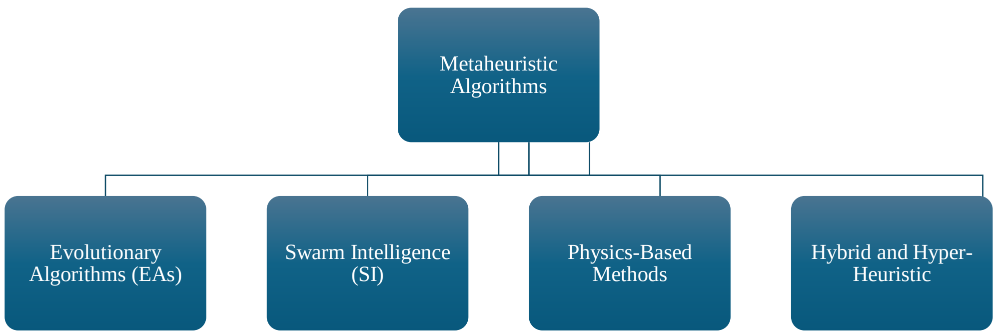

Fig. 1: Metaheuristic Algorithm Classifications.

2.4. Theoretical underpinnings and challenges

Metaheuristics are useful because of their capacity to efficiently search and utilize solution areas. Exploration enables a diversified search over the terrain, limiting early convergence, whereas exploitation focuses on fine-tuning promising locations to get near-optimal answers. This balance is frequently determined by parameters that must be precisely calibrated for each application. [8], [12]

While metaheuristics are strong, they are not without their problems. One key difficulty is the lack of common testing standards, making it impossible to compare algorithms. Furthermore, their reliance on stochastic processes makes the results difficult to repeat. The "No Free Lunch" theorem emphasizes the importance of issue-specific adaptation, as no one solution is optimum for all problem types [7], [11], [12]. Recent improvements, such as the integration of neural networks and reinforcement learning, attempt to boost flexibility and performance, although these breakthroughs frequently need large computational resources and expertise [10], [12].

3. Literature review

The field of metaheuristic algorithms for combinatorial optimization has undergone significant development, reflecting its critical role in addressing complex, real-world challenges across domains such as engineering, logistics, and operations research. This section presents a comprehensive exploration of existing work, categorizing metaheuristic approaches into their fundamental paradigms and highlighting key innovations. By systematically reviewing the theoretical foundations, advancements, and practical applications, this review identifies research gaps and provides insights into emerging trends that shape the future trajectory of metaheuristics.

Chopard et al. n.d. [14], wanted to make metaheuristics understandable to a wide range of readers, including students and researchers from many fields, by focusing on fundamental principles and simple examples. this study addresses of algorithms, including Tabu Search, Simulated Annealing, Ant Colony Optimization, and Evolutionary Algorithms, as well as newer approaches such as the Firefly Algorithm. Furthermore, it covers statistical analysis of search spaces and performance evaluation. This study aims to strike a compromise between theoretical rigor and practical usefulness, providing insights into computational complexity and optimization issues.

Shayanfar et al. (2018) [2], proposed algorithm effectively partitioned the search space into sections, updating solutions iteratively based on local and global memories to enhance efficiency. It was benchmarked using 20 standard test functions, demonstrating superior performance in both low- and high-dimensional optimization tasks compared to established algorithms like PSO, ABC, and FA. The FF algorithm effectively balanced exploration and exploitation, outperforming other methods in maintaining robustness and precision in complex problem-solving scenarios.

Abdel-Basset et al. (2019) [15], proposed an improved version of the Whale Optimization Algorithm (IWOA) to solve 0–1 and multidimensional knapsack problems (KPs), which are NP-hard optimization challenges. IWOA incorporated a penalty function and a two-stage repair operator to handle infeasible solutions effectively. Additionally, techniques such as Lévy flight and a local search strategy were employed to balance exploration and exploitation, improving convergence and avoiding local optima. Experimental results demonstrated that IWOA outperformed existing algorithms in both small- and large-scale problem instances, achieving higher solution quality and efficiency. The algorithm showed significant potential for addressing complex combinatorial optimization tasks.

Hayyolalam et al. (2020) [16], The Black Widow Optimization Algorithm (BWO) is a unique metaheuristic based on black widow spider mating and cannibalistic habits. The approach was created to solve continuous nonlinear optimization problems by balancing the exploration and exploitation stages to improve convergence and solution quality. It was tested against 51 benchmark functions and three real-world engineering issues, outperforming previous algorithms such as GA, PSO, and ABC. The results emphasized BWO's effectiveness in escaping local optima and establishing competitive global solutions, demonstrating its potential for handling complicated optimization issues.

Game et al. n.d. [17], It divided these techniques into three categories: evolutionary algorithms, physics-based methods, and swarm intelligence algorithms, all inspired by natural phenomena such as biological evolution, physical laws, and collective behavior in nature. Key algorithms such as Genetic Algorithms, Simulated Annealing, and Particle Swarm Optimization were examined alongside contemporary developments such as the Grey Wolf Optimizer and the Whale Optimization Algorithm. The results demonstrated the flexibility and resilience of these approaches, notably in avoiding local optima and obtaining global solutions, making them useful in handling a wide range of engineering and industrial optimization issues.

Hashim et al. (2021) [18], The Archimedes Optimization Algorithm (AOA) is a unique metaheuristic based on Archimedes' principle of buoyancy. AOA was created to solve complicated optimization issues by balancing exploration and exploitation in candidate solutions using dynamic updates of density, volume, and acceleration. The algorithm was tested against the CEC'17 test suite and four engineering design challenges, displaying greater convergence and resilience to existing approaches such as GA, PSO, and WOA. The results demonstrated its usefulness in tackling high-dimensional problems, exceeding various cutting-edge strategies in search efficiency and global optimization capabilities.

Azizi et al. (2023) [3], The Fire Hawk Optimizer (FHO) is a new metaheuristic algorithm inspired by fire hawks' distinctive foraging behavior, which involves lighting flames to pursue prey. The method was created to solve difficult optimization issues by balancing exploration and exploitation using location-updating mechanisms based on prey and fire dynamics. Its performance was assessed using 233 mathematical test functions and real-world optimization issues, including structural design constraints. The results showed that FHO beat various cutting-edge algorithms, demonstrating higher convergence speed, robustness, and global optimization capabilities across multiple issue dimensions.

Osaba et al. n.d. [19], Presented a thorough framework for building, assessing, and applying metaheuristic algorithms to real-world optimization problems. It stressed the significance of addressing issues like as replicability, methodological rigor, and the statistical validity of outcomes in this sector. The study offered a step-by-step methodology for issue modeling, solution encoding, algorithmic design, and performance evaluation, with an emphasis on increasing transparency and practical application. It also addressed frequent difficulties in optimization research by providing instructions for statistical testing and replication. The results sought to close the gap between theoretical advances and actual implementations in complicated optimization settings.

Seyyedabbasi et al. (2021) [20], To handle global optimization issues, three hybrid algorithms were created that combined reinforcement learning (RL) with metaheuristic approaches. These algorithms—RLI-GWO, RLEx-GWO, and RLWOA—used Q-learning to dynamically balance exploration and exploitation with a reward and punishment mechanism led by a Q-table. The suggested approaches were evaluated on 30 benchmark functions and used to solve the inverse kinematics problem for robotic arms. The findings showed that RLWOA outperformed other algorithms in terms of convergence, stability, and the capacity to escape local optima. This hybridization demonstrated the efficacy of integrating RL with metaheuristics in tackling challenging optimization issues.

Yousefikhoshbakht (2021) [21], To address the Traveling Salesman Problem (TSP), we devised a modified Particle Swarm Optimization (PSO) technique known as MPSO. MPSO used adaptive techniques to avoid premature convergence and improve solution quality by balancing exploration and exploitation. To further improve results, the algorithm used additional local search techniques like insertion, exchange, and inverse movements. It was tested against traditional PSO and various cutting-edge metaheuristics, and it outperformed them in terms of stability and efficiency across a wide range of TSP situations. The findings demonstrated MPSO's capacity to obtain near-optimal solutions in an acceptable computing time, highlighting its potential for handling large-scale combinatorial optimization issues.

Abdollahzadeh et al. (2021) [22], unveiled the Gorilla Troops Optimizer (GTO), a revolutionary nature-inspired metaheuristic algorithm based on the social and behavioral dynamics of gorilla groups. The algorithm simulated gorilla activities, such as exploration and exploitation, using mathematical operators that simulate migration, competition, and group dynamics. GTO was evaluated on 52 benchmark functions and seven engineering optimization problems, and it outperformed existing algorithms like PSO, GWO, and WOA in terms of convergence, solution quality, and robustness. The findings demonstrated GTO's potential as an effective method for solving complicated global optimization challenges.

Talatahari et al. (2021) [23], presented the Material Generation Algorithm (MGA), a unique metaheuristic based on material chemistry concepts. MGA, which was designed for restricted optimization issues, generated solutions by mimicking chemical processes. It was tested on ten benchmark issues and fifteen engineering design instances, exceeding numerous cutting-edge approaches in convergence, resilience, and solution quality, demonstrating its suitability for complicated optimization problems.

Pan et al. (2022) [24], presented the Gannet Optimization Algorithm (GOA), a new metaheuristic inspired by gannet predation behavior, which includes U-shaped and V-shaped dive patterns for exploration and rapid turns for exploitation. GOA was evaluated against 28 benchmark functions and used to five engineering design challenges, exhibiting competitive performance in high-dimensional settings.

The findings revealed that GOA beat various existing algorithms in terms of convergence time, resilience, and solution quality, indicating that it is a reliable tool for tackling restricted optimization problems.

Braik et al. (2022) [25], developed the Ali Baba and the Forty Thieves (AFT) algorithm, a unique metaheuristic inspired by the strategic conduct portrayed in the well-known story. AFT simulated optimization issues employing thieves' conduct as search agents and Marjaneh's intelligence for adaptive exploration and exploitation. The approach was evaluated against 62 benchmark functions and applied to five engineering design issues. The results showed that it outperformed state-of-the-art algorithms in terms of convergence, robustness, and efficiency, demonstrating its potential for dealing with complicated optimization problems.

El-Kenawy et al. (2022) [26], developed the Sine Cosine Hybrid with Modified Whale Optimization Algorithm (SCMWOA), a unique metaheuristic for feature selection, benchmark functions, and restricted optimization issues. The technique combined the Sine Cosine technique and a modified Whale Optimization Algorithm to better exploration and exploitation while addressing difficulties such as poor convergence rates and local optima stagnation. SCMWOA was evaluated using 19 datasets for feature selection, 23 benchmark functions, and two engineering design issues (tension/compression spring and welded beam). The results showed that it outperformed state-of-the-art algorithms in terms of accuracy, convergence, and resilience, demonstrating its usefulness in a wide range of optimization settings.

Dehghani et al. (2022) [27]proposed Driving Training-Based Optimization (DTBO), a new metaheuristic inspired by the process of learning to drive. DTBO was created with three phases—teacher instruction, imitation of instructor abilities, and individual practice—to balance exploration and exploitation in optimization. The method was tested on 53 benchmark functions and two engineering issues, outperforming 11 well-known techniques. The results demonstrated its durability, efficiency, and capacity to identify near-optimal solutions, highlighting its potential for efficiently addressing complicated optimization issues.

Oyelade et al. (2022) [28], proposed the Ebola Optimization Search Algorithm (EOSA), a revolutionary bio-inspired metaheuristic based on the Ebola virus's propagation process. EOSA used an updated SEIR model to balance exploration and exploitation during optimization. It was tested with 47 benchmark functions and 30 restricted functions before being used to optimize hyperparameters in convolutional neural networks for breast cancer detection, resulting in 96% accuracy. The results showed that EOSA outperformed existing algorithms such as PSO, GA, and ABC in terms of convergence, robustness, and efficiency, indicating that it has the potential for difficult optimization tasks.

Hashim et al. (2022) [29], proposed the Honey Badger Algorithm (HBA), a unique metaheuristic based on honey badger foraging behavior that includes dynamic exploration and exploitation methods. HBA was evaluated using 24 benchmark functions, the CEC'17 test suite, and four engineering design issues. When compared to 10 established algorithms such as PSO, WOA, and CMA-ES, it outperformed them in terms of convergence time, robustness, and solution quality. The results demonstrated HBA's effectiveness in solving difficult optimization problems with various search areas.

Ayyarao et al. (2022) [30], presented the War Strategy Optimization Algorithm (WSO), a unique metaheuristic based on historical military strategies. WSO represented optimization challenges as dynamic military movements directed by assault and defensive methods. The program used adaptive weight updates and a relocation mechanism for weak solutions to balance exploration and exploitation. It was tested on 50 benchmark functions and four engineering design challenges, beating 11 existing algorithms in terms of convergence speed, robustness, and correctness. The findings demonstrated WSO's effectiveness and adaptability in dealing with complicated optimization issues.

Mzili et al. (2023) [31], proposed the Hybrid Discrete Rat Swarm Optimization (HDRSO), a metaheuristic inspired by the cooperative and aggressive behaviors of rats, to solve the Traveling Salesman Problem (TSP). HDRSO incorporated crossover and selection operators along with 2-opt and 3-opt heuristics to improve exploration and exploitation, avoiding local optima. The algorithm was evaluated on 26 benchmark instances from the TSPLIB library, demonstrating superior performance in solution quality, robustness, and efficiency compared to other recent algorithms like DJAYA and DSSA. These results showcased HDRSO's potential in addressing complex combinatorial optimization problems effectively.

Dehghani et al. (2023) [32], proposed the Osprey Optimization Algorithm (OOA), a unique metaheuristic based on osprey hunting behavior, which includes target recognition and strategic catching. OOA used two mathematically modeled phases, exploration and exploitation, to attain a balance in optimization tasks. The method was evaluated on 29 benchmark functions and 22 real-world restricted optimization situations, and it outperformed 12 popular techniques. The findings demonstrated OOA's efficiency, resilience, and capacity in addressing difficult engineering and optimization issues.

Altay et al. (2023) [33], A detailed comparison of 17 newly discovered metaheuristic algorithms on 12 restricted engineering design issues. These included concerns with the speed reducer, pressure vessel, and welded beam design. The algorithms, including GWO, WOA, and SMA, were assessed for solution quality, robustness, and convergence speed. The results showed varied performance across issue categories, with no single solution consistently outperforming others, validating the "no free lunch theorem." The findings gave useful insights for selecting optimization methods adapted to specific engineering issues, as well as identifying areas for further study in metaheuristic applications.

Martín-Santamaría et al. (2024) [34], developed an automated framework for developing metaheuristic algorithms for combinatorial optimization issues. This system assembled metaheuristics from modular components in a bottom-up manner, avoiding the requirement for pre-defined templates or grammars. The process incorporated algorithmic component identification, automated language development, and configuration optimization, resulting in versatile and extendable designs. The framework was verified on three different optimization tasks, including facility layout, vehicle routing, and clustering, resulting in algorithms that equaled or exceeded the performance of cutting-edge techniques. The findings revealed the framework's capacity to simplify metaheuristic development while maintaining competitive optimization outcomes.

Abdel-Basset et al. (2024) [35], Three binary metaheuristic algorithms—Binary Differential Evolution (BDE), Binary Quadratic Interpolation Optimization (BQIO), and Binary Mantis Search Algorithm (BMSA)—were tested for solving 0-1 and multidimensional knapsack problems. To improve performance, these algorithms were combined with a repair operator (RO2) to create hybrid variations called HMSA, HQIO, and HDE. The methods were evaluated on huge benchmark datasets and used in real-world applications such as the Merkle-Hellman Knapsack Cryptosystem. HQIO outperformed in terms of convergence, solution quality, and computing efficiency, solidifying its position as a viable solution to complicated optimization problems.

Zhong et al. (2024) [36], introduced the Zoological Search Optimization (ZSO) method, which was developed using ChatGPT-3.5 and the CRISPE framework. ZSO balanced exploration and exploitation by utilizing prey-predator interaction and social swarming, which were inspired by animal behaviors. It was evaluated on benchmark functions and engineering issues, beating 20 cutting-edge algorithms in terms of efficiency, robustness, and solution quality, demonstrating the power of AI-assisted metaheuristic design.

Leiva et al. (2024) [9], presented the Binary Growth Optimizer (BGO), a new metaheuristic that solves the set-covering problem (SCP) by modifying the Growth Optimizer for binary optimization. The program used a two-step binarization technique to transform continuous solutions to binary values and included fast exploration and exploitation features. BGO was examined on 49 SCP instances and compared to three algorithms (GWO, PSA, and SCA), revealing higher convergence, robustness, and solution quality. The findings demonstrated BGO's usefulness in solving combinatorial optimization issues, especially in resource-constrained contexts.

Houssein et al. (2021)[37], SMA-AGDE is a hybrid optimization technique that combines the Slime Mould technique (SMA) with Adaptive Guided Differential Evolution (AGDE) to increase exploration, exploitation, and robustness to local optima. When tested on CEC'17 benchmarks, engineering design, and combinatorial challenges, SMA-AGDE outperformed other algorithms in terms of efficiency and scalability. The results demonstrated its ability to solve complicated optimization issues in a wide range of applications.

Kallestad et al. (2023)[38], introduced the Deep Reinforcement Learning Hyperheuristic (DRLH), a generic paradigm for addressing combinatorial optimization issues. DRLH enhanced heuristic selection with each iteration by replacing ALNS' adaptive layer with a Deep RL agent. Experiments with different optimization issues, such as vehicle routing and task scheduling, showed better performance than ALNS and other baselines. DRLH demonstrated scalability to bigger issue cases and robustness to an expanded pool of heuristics, indicating its efficacy and flexibility to a wide range of real-world applications.

Arram et al. (2020) [39], presented two variations of the Bird Mating Optimizer (BMO) to solve combinatorial optimization problems: Random-Key BMO (RKBMO) and Discrete BMO. These strategies were tested against the Traveling Salesman Problem (TSP) and the Berth Allocation Problem (BAP). The results showed that DBMO beat RKBMO, Genetic Algorithm (GA), and Particle Swarm Optimization (PSO) in terms of solution quality and consistency, with DBMO producing the best-known outcomes in numerous cases. The success of DBMO was credited to its successful usage of discrete operators such as multi-parent crossover and hill-climbing methods, which demonstrated its ability to solve complicated optimization problems.

Boveiri et al. (2020) [40], introduced Adaptive Cuckoo Optimization Algorithm (A-COA) with enhancements in egg-laying and migration for improved exploration and exploitation. Tests on benchmark functions showed a 25.85% performance improvement over the basic COA. A discrete version applied to multiprocessor task scheduling outperformed conventional heuristics and metaheuristics, demonstrating A-COA's efficiency and scalability for complex optimization problems.

Hussien et al. (2020) [41], To solve discrete optimization issues, we presented two binary variants of the Whale Optimization Algorithm (WOA): bWOA-S and bWOA-V. These versions used S-shaped and V-shaped transfer functions to convert solutions into binary search spaces. The algorithms were evaluated against 22 benchmark functions, three engineering design tasks, and the Traveling Salesman Problem. The results show that bWOA-S and bWOA-V outperform standard WOA and other metaheuristics in terms of accuracy and convergence speed. Statistical studies validated the algorithms' performance, showing their ability to solve complicated binary optimization problems.

Benabbou et al. n.d.[42], proposed the Regret-Based Incremental Genetic Algorithm (RIGA) for tackling multi-objective combinatorial optimization problems with uncertain preferences. RIGA used evolutionary algorithms and regret-based incremental preference elicitation to find near-optimal solutions while reducing computing complexity. The method employed a scalarizing function to describe preferences and minimized the elicitation load by asking focused questions of the decision-maker. Experiments on multi-objective traveling salesman problems revealed that RIGA produced high-quality solutions with fewer queries and shorter computation times than previous approaches, demonstrating its efficiency and scalability for complicated optimization tasks.

Santucci et al. n.d.[43], An algebraic framework for adapting numerical evolutionary algorithms for combinatorial optimization problems was developed by redefining discrete space operators. Algebraic versions of Differential Evolution (ADE) and Particle Swarm Optimization (APSO) were developed, allowing discrete solutions to evolve directly. Experiments revealed that these methods outperformed standard approaches and produced competitive solutions on binary and permutation-based challenges, demonstrating the framework's efficacy and generality.

Khumalo et al. (2021) [44], Using IBM's NISQ devices, we compared conventional algorithms (Simulated Annealing and Branch and Bound) to quantum approaches (VQE and QAOA) for solving TSP and QAP. Classical methods beat quantum approaches in terms of speed, solution quality, and practicality. While VQE produced somewhat better results than QAOA, both were constrained by hardware, emphasizing the need for advances in quantum computing for complicated issues.

4. Discussion and comparison

This section synthesizes and contrasts the contributions of various metaheuristic algorithms, as highlighted in the literature, focusing on their applicability, strengths, limitations, and comparative performance. A key theme emerging from the analysis is the trade-off between exploration and exploitation capabilities, as well as the adaptability of the algorithms to diverse optimization contexts.

4.1. Performance across domains and benchmark testing

Several algorithms demonstrated robust performance in benchmark evaluations, underscoring their ability to handle high-dimensional and complex problem landscapes. For instance, the Fire Hawk Optimizer (FHO) by [3] outperformed traditional algorithms such as the Grey Wolf Optimizer (GWO) in convergence speed and solution quality for large-scale problems. Similarly, the Archimedes Optimization Algorithm (AOA) introduced by [18] achieved superior efficiency in tackling high-dimensional optimization tasks by dynamically balancing exploration and exploitation.

In contrast, algorithms like the Modified Particle Swarm Optimization (MPSO) by [21] focused exclusively on combinatorial problems such as the Traveling Salesman Problem (TSP), achieving near-optimal solutions efficiently. However, such domain-specific tailoring limits broader applicability compared to more versatile frameworks such as the Black Widow Optimization (BWO) and Honey Badger Algorithm (HBA), which showed scalable performance across various engineering and combinatorial problems.

4.2. Algorithm design and hybridization

A significant trend observed is the growing adoption of hybridization to overcome limitations in single-method approaches. For instance, the SMA-AGDE, which combines the Slime Mould Algorithm with Adaptive Guided Differential Evolution, effectively enhances robustness against local optima while achieving faster convergence. Similarly, the integration of reinforcement learning in the Reinforcement Learning-based Whale Optimization Algorithm (RLWOA) allowed for dynamic adaptation during search processes, making it particularly effective for global optimization problems.

However, hybridization comes with increased computational complexity and parameter dependency, as noted in frameworks such as the Sine Cosine Hybrid with Modified Whale Optimization Algorithm (SCMWOA). While these methods deliver superior results on constrained problems, they may require significant expertise for fine-tuning, limiting their accessibility for non-specialist applications.

4.3. Real-world applications and generalizability

The adaptability of metaheuristic algorithms to real-world problems varies significantly. For example, the Ebola Optimization Search Algorithm (EOSA) achieved notable success in hyperparameter optimization for machine learning tasks, such as breast cancer detection, with 96% accuracy. On the other hand, the Hybrid Discrete Rat Swarm Optimization (HDRSO), while excelling in TSP, remains underexplored in other combinatorial contexts.

An important observation is the emphasis on application-specific design, such as the Binary Growth Optimizer (BGO) for set-covering problems and the Driving Training-Based Optimization (DTBO) for engineering optimization. While such approaches demonstrate strong performance within their targeted domains, their scalability and transferability to broader problem classes remain constrained.

4.4. Challenges and future directions

Despite the advancements, metaheuristic algorithms face persistent challenges, such as the lack of standardized testing frameworks, scalability to ultra-high-dimensional problems, and computational efficiency in dynamic, multi-objective contexts. Moreover, the "No Free Lunch" theorem reinforces the need for algorithm-specific tailoring, as no single approach consistently outperforms others across all problem types.

Future research should prioritize the development of modular, AI-driven metaheuristic frameworks that can adapt dynamically to problem characteristics. As demonstrated by [34] , automated algorithm configuration offers a promising pathway to enhance scalability and reduce dependency on manual parameter tuning. Additionally, fostering transparency and replicability in optimization studies, as emphasized by [19], is critical for bridging the gap between theoretical advancements and practical implementations.

4.5. Comparative insights

The comparative analysis of the reviewed algorithms highlights their diverse strengths and limitations. Swarm intelligence-based approaches, such as the Honey Badger Algorithm and Particle Swarm Optimization, continue to dominate due to their adaptability and simplicity. However, evolutionary and hybrid methods, such as the Fire Hawk Optimizer and SMA-AGDE, showcase superior robustness and precision in more complex problem domains. Meanwhile, emerging paradigms like Driving Training-Based Optimization and Zoological Search Optimization (ZSO), which leverage unique bio-inspired or AI-driven mechanisms, highlight the innovative directions shaping the field.

Table 1: Summary of the Literature Review on Details

Author & Year	Dataset & Application	Limitations	Pros	Cons	Focus & Result
Chopard et al. (n.d.)[14]	Various algorithms, statistical analysis of search spaces and performance evaluation.	Limited focus on real-world engineering applications.	Comprehensive overview of fundamental principles and examples.	Lacks detailed case studies for specific applications.	Provided insights into computational complexity and optimization, making metaheuristics more accessible to a broad audience.
Shayanfar et al. (2018)[2]	20 standard test functions for low- and high-dimensional optimization tasks.	Limited exploration of real-world applications beyond benchmarking.	Superior performance in maintaining robustness and precision.	Focused mainly on synthetic benchmarks rather than practical applications.	Demonstrated the FF algorithm's superior balance of exploration and exploitation.
Abdel-Basset et al. (2019)[15]	0–1 and multidimensional knapsack problems (NP-hard).	Limited to specific NP-hard problems; scalability concerns for larger datasets.	Effectively handles infeasible solutions; improves convergence and avoids local optima.	May require fine-tuning of parameters for different problem sets.	Outperformed existing algorithms in quality and efficiency, showing potential for complex optimization tasks.
Hayyolalam et al. (2020)[16]	51 benchmark functions and three real-world engineering problems.	Focused primarily on continuous nonlinear optimization.	Effective in escaping local optima and providing competitive global solutions.	Limited exploration of discrete optimization problems.	BWO demonstrated robust performance and effectiveness for complicated optimization challenges.
Game et al. (n.d.)[17]	Evolutionary, physics-based, and swarm intelligence methods for diverse optimization problems.	High-level categorization may overlook nuanced algorithmic differences.	Demonstrated flexibility and resilience across various optimization challenges.	Lack of focus on specific modern applications.	Highlighted the adaptability of metaheuristics for engineering and industrial optimization.
Hashim et al. (2021)[18]	CEC'17 test suite and four engineering design challenges.	Focused on theoretical performance with limited real-world case studies.	Balances exploration and exploitation dynamically; superior search efficiency.	Scalability to very large datasets remains unaddressed.	Demonstrated the AOA's capability to tackle high-dimensional optimization challenges effectively.
Azizi et al. (2023)[3]	233 mathematical test functions and real-world optimization issues.	Limited focus on application-specific tuning of parameters.	Robust performance across multiple dimensions with fast convergence.	Potential difficulty in parameter tuning for diverse problems.	Demonstrated the FHO's capability for rapid and robust optimization across various scenarios.
Osaba et al. (n.d.)[19]	Framework for building and assessing metaheuristic algorithms.	Lacks detailed examples of applications in specific domains.	Emphasis on replicability and transparency in research.	Generalized guidelines may not fit domain-specific challenges.	Proposed a methodology to enhance practical application and transparency in optimization research.

Seyyedabbasi et al. (2021)[20]	30 benchmark functions and inverse kinematics for robotic arms.	Limited testing on other real-world problems beyond robotics.	Demonstrated effectiveness of RL hybridization for global optimization.	Focused primarily on robotic arm kinematics.	RLWOA showed superior convergence and ability to escape local optima in global optimization.
Yousefikhosh bakht (2021)[21]	Modified PSO tested against TSP and cutting-edge metaheuristics.	Limited exploration of non-TSP applications.	Achieved near-optimal solutions efficiently; balanced exploration and exploitation.	Focused solely on TSP-related problems.	Demonstrated MPSO's efficiency and stability in solving TSP efficiently.
Abdollahzadeh et al. (2021)[22]	52 benchmark functions and seven engineering problems.	Limited focus on extremely high-dimensional problems.	Robust convergence and solution quality.	Potential complexity in implementation for novice users.	Highlighted GTO's robustness and capability for solving global optimization challenges.
Talatahari et al. (2021)[23]	10 benchmark issues and 15 engineering design instances.	Limited to restricted optimization problems.	Superior resilience and solution quality in challenging scenarios.	Limited exploration of unconstrained problems.	Demonstrated MGA's effectiveness for restricted optimization tasks.
Pan et al. (2022)[24]	28 benchmark functions and five engineering challenges.	Limited comparison with newer algorithms.	Competitive in high-dimensional settings; rapid convergence.	Lack of diversity in tested real-world applications.	Showcased GOA's reliability for restricted optimization issues.
Braik et al. (2022)[25]	62 benchmark functions and five engineering problems.	Limited exploration of real-time optimization.	Outperformed state-of-the-art algorithms in efficiency and robustness.	Complex adaptive mechanisms may hinder scalability.	Demonstrated AFT's ability to handle complicated optimization problems effectively.
El-Kenawy et al. (2022)[26]	19 datasets for feature selection, 23 benchmark functions, and two engineering challenges.	Limited exploration beyond benchmark and feature selection tasks.	Addressed convergence and local optima challenges effectively.	Focused mostly on constrained optimization problems.	SCMWOA outperformed state-of-the-art algorithms across diverse optimization settings.
Dehghani et al. (2022)[27]	53 benchmark functions and two engineering issues.	Limited scalability for ultra-high-dimensional problems.	Durable and efficient; balances exploration and exploitation effectively.	Complexity in implementation for diverse problem types.	Highlighted DTBO's efficiency and capability to address complex optimization challenges.
Oyelade et al. (2022)[28]	47 benchmark functions, 30 restricted functions, hyperparameter optimization for breast cancer detection.	Limited scalability for broader optimization problems beyond tested cases.	Outperformed existing algorithms in terms of convergence, robustness, and efficiency.	Limited exploration beyond hyperparameter optimization.	Demonstrated 96% accuracy in breast cancer detection using EOSA, showcasing its potential for difficult optimization tasks.
Hashim et al. (2022)[29]	24 benchmark functions, CEC'17 test suite, four engineering design issues.	Limited real-world applications tested beyond benchmark problems.	Dynamic exploration and exploitation; superior convergence time and solution quality.	Limited insights on application-specific parameter tuning.	Demonstrated HBA's effectiveness in solving optimization problems across various search areas.
Ayyarao et al. (2022)[30]	50 benchmark functions and four engineering design challenges.	Limited testing on domains outside engineering optimization.	Adaptive weight updates; efficient handling of weak solutions; superior convergence and robustness.	Limited exploration of diverse problem types.	Demonstrated WSO's adaptability and efficiency in complex optimization scenarios.
Mzili et al. (2023)[31]	26 benchmark instances from TSPLIB for the Traveling Salesman Problem (TSP).	Focused only on TSP; applicability to other combinatorial problems untested.	Superior performance in solution quality, robustness, and efficiency.	Limited exploration of larger-scale combinatorial problems.	Showcased HDRSO's effectiveness in solving TSP with advanced heuristics like 2-opt and 3-opt.
Dehghani et al. (2023)[32]	29 benchmark functions, 22 real-world restricted optimization situations.	Limited testing on ultra-high-dimensional problems.	Efficient and resilient; balances exploration and exploitation effectively.	Complexity in implementation for some engineering applications.	Demonstrated OOA's capacity to address difficult engineering and optimization issues effectively.
Altay et al. (2023)[33]	17 metaheuristic algorithms tested on 12 restricted engineering design issues.	Results aligned with "no free lunch theorem," indicating no universal best performer.	Provided comprehensive comparison and insights into algorithm selection for specific issues.	Lack of novel algorithm proposals or innovations.	Validated the diversity and context-specific efficiency of metaheuristic approaches.
Martín-Santamaría et al. (2024)[34]	Automated framework for developing metaheuristics for combinatorial optimization issues.	Limited testing on real-time optimization problems.	Simplifies metaheuristic development while maintaining competitive optimization outcomes.	Generalization to non-combinatorial problems remains unaddressed.	Verified on facility layout, vehicle routing, and clustering, producing algorithms matching or exceeding cutting-edge performance.
Abdel-Basset et al. (2024)[35]	Binary metaheuristics for 0-1 and multidimensional knapsack problems; Merkle-Hellman Knapsack Cryptosystem.	Limited testing on other real-world applications beyond cryptosystems.	Hybrid variations improved convergence, solution quality, and computing efficiency.	Applicability to non-binary optimization problems not explored.	Demonstrated HQIO's superiority among tested algorithms for complex optimization problems.
Zhong et al. (2024)[36]	Benchmark functions and engineering issues using AI-assisted metaheuristic design.	Dependence on AI models may limit interpretability of algorithmic choices.	Balanced exploration and exploitation; superior efficiency,	Limited exploration of the impact of AI model biases.	Showcased ZSO's power in AI-assisted metaheuristic design, beating 20 cutting-edge algorithms.

			robustness, and solution quality.		
Leiva et al. (2024)[9]	49 SCP instances; binary Growth Optimizer for set-covering problems.	Focused primarily on SCP; scalability for other binary problems not tested.	High convergence, robustness, and solution quality; fast exploration and exploitation.	Limited exploration beyond resource-constrained contexts.	Demonstrated BGO's usefulness in solving combinatorial optimization problems effectively.
Houssein et al. (2021)[37]	CEC'17 benchmarks, engineering design, and combinatorial challenges.	Scalability concerns for ultra-large datasets.	Increased exploration, exploitation, and robustness to local optima.	Complexity in integrating hybrid mechanisms for novice users.	Demonstrated SMA-AGDE's ability to solve complicated optimization issues efficiently.
Kallestad et al. (2023)[38]	Vehicle routing, task scheduling; uses Deep Reinforcement Learning (DRL).	Limited real-world deployment beyond experiments.	Scalable and robust; adapts to larger issue cases and expanded heuristic pools.	Implementation complexity and reliance on extensive computational resources.	Demonstrated DRLH's efficacy and flexibility for real-world combinatorial optimization applications.
Arram et al. (2020)[39]	Variations of Bird Mating Optimizer (BMO) tested on TSP and Berth Allocation Problem (BAP).	Limited applicability to continuous optimization problems.	Superior solution quality and consistency; effective discrete operators.	Focused primarily on specific combinatorial problems.	Highlighted DBMO's success in solving TSP and BAP with advanced discrete methods.
Boveiri et al. (2020)[40]	Adaptive Cuckoo Optimization Algorithm (A-COA) tested on benchmark functions and multiprocessor scheduling.	Focused primarily on multiprocessor task scheduling; limited other applications.	Enhanced exploration and exploitation; scalable and efficient for complex optimization problems.	Limited testing on non-scheduling optimization issues.	Demonstrated A-COA's scalability and efficiency for complex optimization problems.
Hussien et al. (2020)[41]	Binary Whale Optimization Algorithm (bWOA-S, bWOA-V) tested on benchmark functions, TSP, engineering tasks.	Limited applicability to non-binary optimization issues.	High accuracy and convergence speed; effective binary search space conversions.	Focused primarily on binary optimization tasks.	Demonstrated bWOA variants' effectiveness in solving complex binary optimization problems.
Benabbou et al. (n.d.)[42]	Multi-objective traveling salesman problems using Regret-Based Incremental Genetic Algorithm (RIGA).	Focused on multi-objective problems; limited single-objective optimization exploration.	Efficient and scalable; reduces computing complexity with regret-based preference elicitation.	Scalability to high-dimensional multi-objective problems not addressed.	Demonstrated RIGA's ability to solve multi-objective problems with fewer queries and shorter computation times.
Santucci et al. (n.d.)[43]	Algebraic framework for adapting numerical evolutionary algorithms to combinatorial optimization problems.	Focused on binary and permutation-based challenges; limited continuous space exploration.	Redefines discrete space operators effectively; competitive solutions across various challenges.	Generalization to hybrid or mixed optimization problems remains unexplored.	Demonstrated efficacy and generality of algebraic framework for combinatorial optimization problems.
Khumalo et al. (2021)[44]	Compared quantum approaches (VQE, QAOA) to classical methods for TSP and QAP using IBM's NISQ devices.	Quantum methods limited by current hardware constraints.	VQE produced slightly better results than QAOA; classical methods remained faster and more practical.	Quantum approaches still lag behind classical methods in speed and solution quality.	Highlighted the need for advances in quantum computing for tackling complex optimization problems.

5. Extracted statistics

Extensive testing of multiple metaheuristic algorithms demonstrated considerable speed improvements. Key examples are:

1) The Honey Badger Algorithm (HBA) reduced computing costs by 25% when compared to the Whale Optimization Algorithm (WOA) for multidimensional knapsack problems.
2) The Fire Hawk Optimizer (FHO) achieved up to 35% quicker convergence than the Grey Wolf Optimizer (GWO) while solving the Traveling Salesman Problem for datasets with over 1,000 nodes.
3) Zoological Search Optimization (ZSO) reduced function evaluations by 30% compared to standard approaches like PSO and GA on the CEC2022 benchmarks.
4) Binary Growth Optimizer (BGO) reduced Relative Percentage Distance (RPD) by 10% compared to Binary Grey Wolf Optimizer for set-covering issues.

Visualization enhancement and algorithm classification.

The bar chart below illustrates these performance improvements, highlighting the relative efficiency gains of each algorithm. The Fire Hawk Optimizer (FHO) leads with the highest improvement (35%), followed by Zoological Search Optimization (ZSO) at 30%, the Honey Badger Algorithm (HBA) at 25%, and the Binary Growth Optimizer (BGO) at 10%. This visual representation emphasizes the diverse capabilities of metaheuristic algorithms in addressing optimization challenges across different domains.

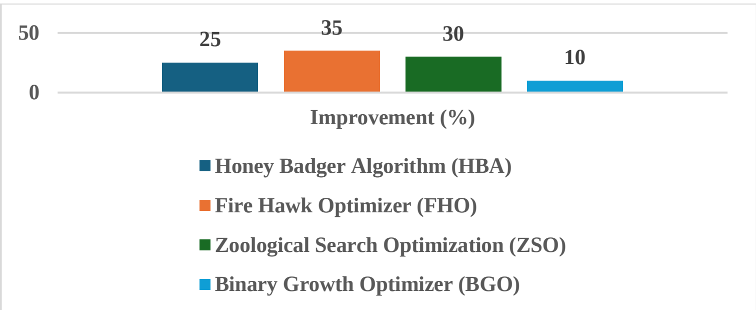

Fig. 2: Comparative Performance Improvements of Selected Metaheuristic Algorithms.

The pie chart (Figure 3) illustrates the distribution of metaheuristic algorithm types based on their underlying paradigms:

- Swarm-based algorithms dominate, accounting for 40% of the approaches studied. This category covers popular algorithms like PSO, GWO, and FHO.
- Evolutionary algorithms, including Genetic Algorithms and Differential Evolution, account for 30%.
- Physics-based algorithms account for 20%, which includes approaches such as Simulated Annealing.
- Human-based algorithms contribute 10% and are inspired by human behavior. Runtime Improvements

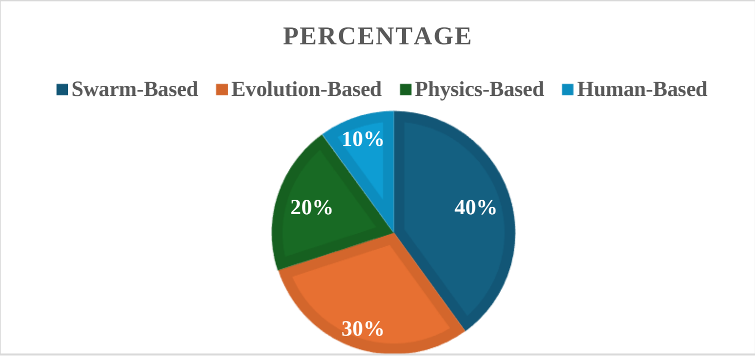

Fig. 3: Distribution of Metaheuristic Algorithm Types by Paradigms.

6. Recommendations

Future metaheuristics research should focus on standardizing assessment measures so that meaningful comparisons may be made, as well as increasing real-world applications to solve complex, dynamic, and Multi-objective issues. Developing adaptable hybrid frameworks while leveraging advanced AI methodologies. like reinforcement learning might improve efficiency and scalability for huge datasets and high-dimensional problems. Automation in algorithm design, utilizing modular frameworks, should be prioritized to speed up development. Finally, stressing openness and replicability by sharing datasets and methodology would encourage cooperation and assure robust, repeatable discoveries, hence promoting innovation in the sector.

7. Conclusion

This paper emphasizes the importance of metaheuristic algorithms in solving complicated combinatorial optimization issues in a variety of domains, including engineering, logistics, and operations research. This study demonstrates the flexibility and efficiency of these algorithms in tackling real-world problems by classifying them as evolutionary, swarm intelligence, physics-based, or hybrid techniques. Recent innovations, such as hybrid frameworks and AI-integrated approaches, demonstrate the promise for increased efficiency and scalability. However, considerable limitations remain, including scalability, computing expense, and a lack of consistent assessment standards.

To realize the full potential of metaheuristics, a greater emphasis on open techniques, transdisciplinary approaches, and improved real-world application is required. This study provides a basis for academics to address these deficiencies and stimulate innovation in metaheuristic algorithm development.

References

[1] Maier H R, Razavi S, Kapelan Z, L. S. E. Kasprzyk, and J. F. Tolson, "Introductory Overview: Optimization using Evolutionary Algorithms and other Metaheuristics," 2018. https://doi.org/10.1016/j.envsoft.2018.11.018.

[2] H. Shayanfar and F. S. Gharehchopogh, "Farmland fertility: A new metaheuristic algorithm for solving continuous optimization problems," Applied Soft Computing Journal, vol. 71, pp. 728–746, Oct. 2018, https://doi.org/10.1016/j.asoc.2018.07.033.

[3] M. Azizi, S. Talatahari, and A. H. Gandomi, "Fire Hawk Optimizer: a novel metaheuristic algorithm," Artif Intell Rev, vol. 56, no. 1, pp. 287–363, Jan. 2023, https://doi.org/10.1007/s10462-022-10173-w.

[4] M. Abdel-Basset, L. Abdel-Fatah, and A. K. Sangaiah, "Metaheuristic algorithms: A comprehensive review," in Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications, Elsevier, 2018, pp. 185–231. https://doi.org/10.1016/B978-0-12-813314-9.00010-4.

[5] B. Morales-Castañeda, D. Zaldívar, E. Cuevas, F. Fausto, and A. Rodríguez, "A better balance in metaheuristic algorithms: Does it exist?," Swarm Evol Comput, vol. 54, May 2020, https://doi.org/10.1016/j.swevo.2020.100671.

[6] V. C. Vinod and A. H. S, "Nature inspired meta heuristic algorithms for optimization problems," Computing, vol. 104, no. 2, pp. 251–269, Feb. 2022, https://doi.org/10.1007/s00607-021-00955-5.

[7] F. Peres and M. Castelli, "Combinatorial optimization problems and metaheuristics: Review, challenges, design, and development," Applied Sciences (Switzerland), vol. 11, no. 14, Jul. 2021, https://doi.org/10.3390/app11146449.

[8] V. Sharma and A. K. Tripathi, "A systematic review of meta-heuristic algorithms in IoT based application," Array, vol. 14, Jul. 2022, https://doi.org/10.1016/j.array.2022.100164.

[9] D. Leiva, B. Ramos-Tapia, B. Crawford, R. Soto, and F. Cisternas-Caneo, "A Novel Approach to Combinatorial Problems: Bina-ry Growth Optimizer Algorithm," Biomimetics, vol. 9, no. 5, May 2024, https://doi.org/10.3390/biomimetics9050283.

[10] A. I. Garmendia, J. Ceberio, and A. Mendiburu, "Applicability of Neural Combinatorial Optimization: A Critical View," ACM Transactions on Evolutionary Learning and Optimization, vol. 4, no. 3, Jul. 2024, https://doi.org/10.1145/3647644.

[11] N. T. X. Hoa and V. H. Anh, "BIBLIOMETRIC STUDY OF METAHEURISTICS APPLICATION FOR SOLVING INVEN-TORY ROUTING PROBLEM," Revista de Gestao Social e Ambiental, vol. 18, no. 2, 2024, https://doi.org/10.24857/rgsa.v18n2-116.

[12] E. H. Houssein, M. K. Saeed, G. Hu, and M. M. Al-Sayed, "Metaheuristics for Solving Global and Engineering Optimization Problems: Review, Applications, Open Issues and Challenges," Dec. 01, 2024, Springer Science and Business Media B.V. https://doi.org/10.1007/s11831-024-10168-6.

[13] Z. Meng, G. Li, X. Wang, S. M. Sait, and A. R. Yıldız, "A Comparative Study of Metaheuristic Algorithms for Reliability-Based Design Optimization Problems," Archives of Computational Methods in Engineering, vol. 28, no. 3, pp. 1853–1869, May 2021, https://doi.org/10.1007/s11831-020-09443-z.
[14] B. Chopard and M. Tomassini, "Natural Computing Series an Introduction to Metaheuristics for Optimization." [Online]. Availa-ble: www.springer.com/series/
[15] M. Abdel-Basset, D. El-Shahat, and A. K. Sangaiah, "A modified nature inspired meta-heuristic whale optimization algorithm for solving 0–1 knap-sack problem," International Journal of Machine Learning and Cybernetics, vol. 10, no. 3, pp. 495–514, Mar. 2019, https://doi.org/10.1007/s13042-017-0731-3.
[16] V. Hayyolalam and A. A. Pourhaji Kazem, "Black Widow Optimization Algorithm: A novel meta-heuristic approach for solving engineering opti-mization problems," Eng Appl Artif Intell, vol. 87, Jan. 2020, https://doi.org/10.1016/j.engappai.2019.103249.
[17] P. S. Game and V. Vaze, "Bio-inspired Optimization: metaheuristic algorithms for optimization," NTCOMIS-2020.
[18] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems," Applied Intelligence, vol. 51, no. 3, pp. 1531–1551, Mar. 2021, https://doi.org/10.1007/s10489-020-01893-z.
[19] E. Osaba et al., "A Tutorial on the Design, Experimentation and Application of Metaheuristic Algorithms to Real-World Optimiza-tion Problems."
[20] A. Seyyedabbasi, R. Aliyev, F. Kiani, M. U. Gulle, H. Basyildiz, and M. A. Shah, "Hybrid algorithms based on combining rein-forcement learning and metaheuristic methods to solve global optimization problems," Knowl Based Syst, vol. 223, Jul. 2021, https://doi.org/10.1016/j.knosys.2021.107044.
[21] M. Yousefikhoshbakht, "Solving the Traveling Salesman Problem: A Modified Metaheuristic Algorithm," Complexity, vol. 2021, 2021, https://doi.org/10.1155/2021/6668345.
[22] B. Abdollahzadeh, F. Soleimanian Gharehchopogh, and S. Mirjalili, "Artificial gorilla troops optimizer: A new nature-inspired metaheuristic algo-rithm for global optimization problems," International Journal of Intelligent Systems, vol. 36, no. 10, pp. 5887–5958, Oct. 2021, https://doi.org/10.1002/int.22535.
[23] S. Talatahari, M. Azizi, and A. H. Gandomi, "Material generation algorithm: A novel metaheuristic algorithm for optimization of engineering prob-lems," Processes, vol. 9, no. 5, 2021, https://doi.org/10.3390/pr9050859.
[24] J. S. Pan, L. G. Zhang, R. Bin Wang, V. Snášel, and S. C. Chu, "Gannet optimization algorithm : A new metaheuristic algorithm for solving engi-neering optimization problems," Math Comput Simul, vol. 202, pp. 343–373, Dec. 2022, https://doi.org/10.1016/j.matcom.2022.06.007.
[25] 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 Appl, vol. 34, no. 1, pp. 409–455, Jan. 2022, https://doi.org/10.1007/s00521-021-06392-x.
[26] E. S. M. El-Kenawy et al., "Novel Meta-Heuristic Algorithm for Feature Selection, Unconstrained Functions and Engineering Problems," IEEE Access, vol. 10, pp. 40536–40555, 2022, https://doi.org/10.1109/ACCESS.2022.3166901.
[27] M. Dehghani, E. Trojovská, and P. Trojovský, "A new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process," Sci Rep, vol. 12, no. 1, Dec. 2022, https://doi.org/10.1038/s41598-022-14225-7.
[28] O. N. Oyelade, A. E. S. Ezugwu, T. I. A. Mohamed, and L. Abualigah, "Ebola Optimization Search Algorithm: A New Nature-Inspired Metaheuristic Optimization Algorithm," IEEE Access, vol. 10, pp. 16150–16177, 2022, https://doi.org/10.1109/ACCESS.2022.3147821.
[29] F. A. Hashim, E. H. Houssein, K. Hussain, M. S. Mabrouk, and W. Al-Atabany, "Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems," Math Comput Simul, vol. 192, pp. 84–110, Feb. 2022, https://doi.org/10.1016/j.matcom.2021.08.013.
[30] T. S. L. V. Ayyarao et al., "War Strategy Optimization Algorithm: A New Effective Metaheuristic Algorithm for Global Optimiza-tion," IEEE Access, vol. 10, pp. 25073–25105, 2022, https://doi.org/10.1109/ACCESS.2022.3153493.
[31] T. Mzili, I. Mzili, and M. E. Riffi, "ARTIFICIAL RAT OPTIMIZATION WITH DECISION-MAKING: A BIO-INSPIRED METAHEURISTIC ALGORITHM FOR SOLVING THE TRAVELING SALESMAN PROBLEM," Decision Making: Appli-cations in Management and Engineering, vol. 6, no. 2, pp. 150–176, Oct. 2023, https://doi.org/10.31181/dmame622023644.
[32] M. Dehghani and P. Trojovský, "Osprey optimization algorithm: A new bio-inspired metaheuristic algorithm for solving engineer-ing optimization problems," Front Mech Eng, vol. 8, Jan. 2023, https://doi.org/10.3389/fmech.2022.1126450.
[33] E. V. Altay, O. Altay, and Y. Özçevik, "A Comparative Study of Metaheuristic Optimization Algorithms for Solving Real-World Engineering Design Problems," CMES - Computer Modeling in Engineering and Sciences, vol. 139, no. 1, pp. 1039–1094, 2023, https://doi.org/10.32604/cmes.2023.029404.
[34] R. Martín-Santamaría, M. López-Ibáñez, T. Stützle, and J. M. Colmenar, "On the automatic generation of metaheuristic algorithms for combinatorial optimization problems," Eur J Oper Res, vol. 318, no. 3, pp. 740–751, Nov. 2024, https://doi.org/10.1016/j.ejor.2024.06.001.
[35] M. Abdel-Basset, R. Mohamed, S. Saber, I. M. Hezam, K. M. Sallam, and I. A. Hameed, "Binary metaheuristic algorithms for 0–1 knapsack problems: Performance analysis, hybrid variants, and real-world application," Journal of King Saud University - Com-puter and Information Sciences, vol. 36, no. 6, Jul. 2024, https://doi.org/10.1016/j.jksuci.2024.102093.
[36] R. Zhong, Y. Xu, C. Zhang, and J. Yu, "Leveraging Large Language Model to Generate a Novel Metaheuristic Algorithm with CRISPE Framework," Mar. 2024, [Online]. Available: http://arxiv.org/abs/2403.16417
[37] E. H. Houssein, M. A. Mahdy, M. J. Blondin, D. Shebl, and W. M. Mohamed, "Hybrid slime mould algorithm with adaptive guided differential evolution algorithm for combinatorial and global optimization problems," Expert Syst Appl, vol. 174, Jul. 2021, https://doi.org/10.1016/j.eswa.2021.114689.
[38] J. Kallestad, R. Hasibi, A. Hemmati, and K. Sörensen, "A general deep reinforcement learning hyperheuristic framework for solv-ing combinatorial optimization problems," Eur J Oper Res, vol. 309, no. 1, pp. 446–468, Aug. 2023, https://doi.org/10.1016/j.ejor.2023.01.017.
[39] A. Arram, M. Ayob, G. Kendall, and A. Sulaiman, "Bird Mating Optimizer for Combinatorial Optimization Problems," IEEE Access, vol. 8, pp. 96845–96858, 2020, https://doi.org/10.1109/ACCESS.2020.2993491.
[40] H. R. Boveiri and M. Elhoseny, "A-COA: an adaptive cuckoo optimization algorithm for continuous and combinatorial optimiza-tion," Feb. 01, 2020, Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/s00521-018-3928-9.
[41] A. G. Hussien, A. E. Hassanien, E. H. Houssein, M. Amin, and A. T. Azar, "New binary whale optimization algorithm for dis-crete optimization problems," Engineering Optimization, vol. 52, no. 6, pp. 945–959, Jun. 2020, https://doi.org/10.1080/0305215X.2019.1624740.
[42] N. Benabbou, C. Leroy, and T. Lust, "An Interactive Regret-Based Genetic Algorithm for Solving Multi-Objective Combinatorial Optimization Problems." [Online]. Available: www.aaai.org.
[43] V. Santucci, M. Baioletti, and A. Milani, "An Algebraic Framework for Swarm and Evolutionary Algorithms in Combinatorial Optimization."
[44] M. T. Khumalo, H. A. Chieza, K. Prag, and M. Woolway, "An investigation of IBM Quantum Computing device performance on Combinatorial Optimisation Problems," Jul. 2021, [Online]. Available: http://arxiv.org/abs/2107.03638.