American Journal of Computer Science and Technology
2024, Vol. 7, No. 4, pp. 195-217
https://doi.org/10.11648/j.ajcst.20240704.17

Research Article

Comprehensive Study of Population Based Algorithms

Yam Krishna Poudel[1, *], Jeewan Phuyal[2], Rajiv Kumar[1]

[1]Department of Computer Science Engineering, RIMT University, Punjab, India
[2]Department of Electrical and Electronics, Nepal Engineering College, Pokhara University, Pokhara, Nepal

Abstract

The exponential growth of industrial enterprise has highly increased the demand for effective and efficient optimization solutions. Which is resulting to the broad use of meta heuristic algorithms. This study explores eminent bio-inspired population based optimization techniques, including Particle Swarm Optimization (PSO), Spider Monkey Optimization (SMO), Grey Wolf Optimization (GWO), Cuckoo Search Optimization (CSO), Grasshopper Optimization Algorithm (GOA), and Ant Colony Optimization (ACO). These methods which are inspired by natural and biological phenomena, offer revolutionary problems solving abilities with rapid convergence rates and high fitness scores. The investigation examines each algorithm's unique features, optimization properties, and operational paradigms, conducting broad comparative analyses against conventional methods, such as search history, fitness functions and to express their superiority. The study also assesses their relevance, arithmetic andlogical efficiency, applications, innovation, robustness, andlimitations. The findings show the transformative potential of these algorithms and offering valuable wisdom for future research to enhance and broaden upon these methodologies. This finding assists as a guiding for researchers to enable inventive solutions based in natural algorithms and advancing the field of optimization.

Keywords

Meta Heuristic Algorithms, Particle Swarm Optimization, Spider Monkey Optimization, Grey Wolf Optimization, Cuckoo Search Optimization, Grasshopper Optimization Algorithm, Ant Colony Optimization

1. Introduction

Population based meta-heuristic optimization algorithms have regularly illustrated outstanding performance in addressing a wide range of real-world optimization challenges. These algorithms are widely used in robotics, wireless networks, power systems, job shop scheduling, and artificial neural network classification and training [1, 2]. Although they have widespread utility, to achieve a global optimal solution often requires a prominent number of fitness evaluations which poseslimitations for high-complexity problems; such as computational fluid dynamics simulations and structural optimization. In such cases, assessing candidate solutions often requires computationally in-depth numerical methods. This method can demand substantial CPU time, ranging from several minutes to days [3, 4].

The researchers have highly focused on Swarm Intelligence (SI) techniques, to address these challenges. SI algorithm is a subset of meta-heuristic methods, which emulates the collective behavior of natural agents to achieve coherent global

[*]Corresponding author: yampd01@gmail.com (Yam Krishna Poudel)

Received: 20 November 2024; **Accepted:** 7 December 2024; **Published:** 23 December 2024

patterns throughlocal interactions, such as fish schooling, bird flocking, and ant foraging [5]. Evolutionary Algorithms rely on mutation and selection mechanisms but SI techniques utilize self-organizing behaviors to strike a balance between exploration and exploitation. One of the widely adopted methods for scheduling, power system optimization and neural network training is Particle Swarm Optimization (PSO); due to its fast convergence and solution accuracy [6]. Similarly, Grey Wolf Optimizer (GWO) has proven effective in multi-objective optimization problems, including IoT network resource allocation and dynamic trajectory optimization [7, 8].

Grasshopper Optimization Algorithm (GOA) and Spider Monkey Optimization (SMO) have shown significant application in handling engineering design problems and healthcare applications. GOA has been successfully utilized for energy management in micro-grids and structural optimization [9, 10] and SMO has been applied to medical image feature extraction and network intrusion detection [11, 12]. The hybrid use of these algorithms improves dynamic optimization quality and helps to further expand their applications mitigating individual algorithmic weakness [13, 14].

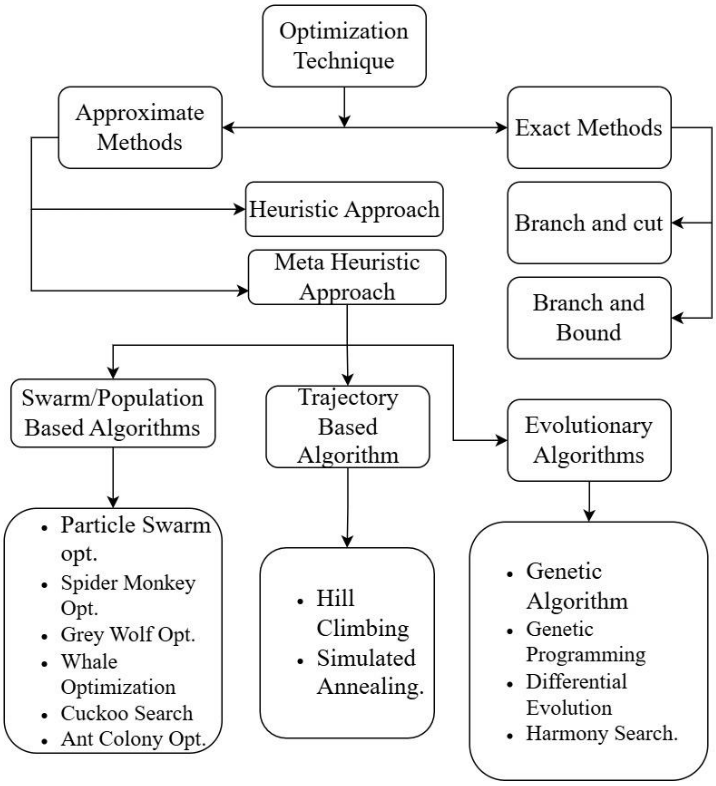

Figure 1. *Classification of Optimization Technique.*

Innovationslike adaptive parameter tuning, dynamic population updates, and self-learning mechanisms have enhanced the efficiency and robustness of these algorithms, in addition to hybridization [15, 16]. Adaptive mechanisms accelerate convergence rates and improving overall performance which helps maintain a balance between exploration of the search space and exploitation of the identified solutions. These innovations help algorithmslike WOA and CSO for optimizing renewable energy systems, feature selection in big data analytics, andlarge-scale scheduling tasks [17, 18].

The computational intensity of meta-heuristic algorithms remains a significant concern, despite their advantages. In case of high-complexity problems, such as those encountered in CFD and structural design: it required embed problem-specific knowledge of hybrid model to improve computational efficiency. For example, integrating GOA with chaos theory has shown success in energy management applications, while fuzzy-enhanced ACO has demonstrated success inlogistical optimization [19, 20]. These hybrid approaches mitigate thelimitations of individual algorithms and open doors for innovative applications, especially in dynamic and high-dimensional problem domains.

The versatility and adaptability of meta-heuristic optimization algorithms make them valuable tools in addressing modern challenges. Their impact is evident across diverse fields, from enhancing renewable energy systems to optimizing machinelearning models and solvinglarge-scale scheduling problems. To ensure these algorithms remain relevant and efficient for emerging optimization challenges, future research should continue to explore hybrid models, adaptive mechanisms, and domain-specific enhancements.

This paper aims to demonstrate the working principles, strengths, andlimitations of these algorithms in depth. It also highlights their relevance and assess the recent innovations and applications, with a particular focus on their hybrid forms. The study aims to provide a broad understanding of how these powerful algorithms are shaping optimization practices and the advancements thatlie ahead in their development.

1.1. Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO)was introduced by Kennedy and Eberhart. It is a stochastic and swarm-based algorithm influenced by the collective behavior of animalslike fish in schools or birds in flocks. A particle moving through the problem space with a certain velocity is represented as the each possible solution in the PSO. These particles mimicking the social dynamics of a group for adjust their movements based on their own best experiences and the successes of their neighbors. This iterative process guides the swarm toward the optimal solution, muchlike a flock collectively searching for food [3, 21].

PSO is broadly popular due to its simplicity and fewer parameters to adjust. It has been applied effectively across various fields and is known for its potential to be hybridized or specialized for specific needs. The algorithm faces challenges in high-dimensional or complex problem spaces. It often converges slowly and may struggle to escapelocal optima, it resulting in suboptimal performance. Particles can become confined tolimited regions of the search space. Which reducing thelikelihood of finding the global best solution in problems with numerous dimensions. Despite theselimitations, PSO remains a powerful and adaptable tool in the field of optimization [22, 23].

This study provides a detailed taxonomy of PSO applications across various domains, including healthcare, environment, industry, commerce, smart cities, and general optimization challenges. Specific issues where found in each domain, such as economic emission dispatch, PV parameter identification, pollution forecasting, water quality monitoring, and food control in environmental applications. The taxonomy addresses these issues by classifying and reviewing key research contributions. Moreover, general concerns in PSO implementations are shown, proposing conceptual approaches to enhance adaptability across diverse applications [21, 24]. Comparative analyses of studies are also provided, focusing on their goals, case studies, strengths, limitations, and results, fostering the development of more efficient PSO-based solutions.

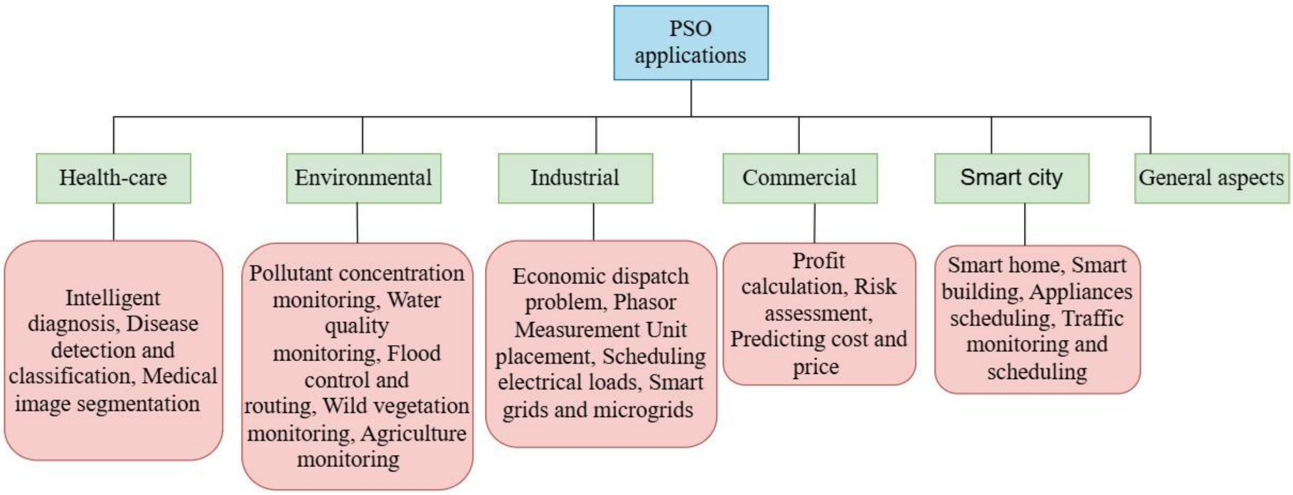

Figure 2. *Applications of PSO.*

Modified PSO by the Inertia Constant

This model is referred to as the standard PSO throughout this paper. In this model, a swarm of particles flies in a d-dimensional search space searching an optimal solution. Each particle i has a current velocity vector $V_i = [v_{i1}, v_{i2}, \dots, v_{in}]$ and a current position vector $X_i = [x_{i1}, x_{i2}, \dots, x_{in}]$, where n is the number of dimensions. The PSO process starts by randomly initializing $X_i and\ V_i$.Then, the best position that has been found by particle $i, Pbest_i = [Pbest_{i1}, Pbest_{i2}, \dots, Pbest_{in}]$ and the best position that has been found by the whole swarm $Gbest = [Gbest_1, Gbest_2, \dots, Gbest\_n]$ lead particle i to update its velocity and position by equations (1) and (2) in each iteration:

$$V_i(t+1) = W * V_i(\text{t}) + c_1 * r_1 [Pbest_i - X_i(\text{t})] + c_2 * r_2 [Gbest - X_i(\text{t})] \quad (1)$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (2)$$

Here, V_i(t) and X_i(t) are the velocity and position of particle i at time t, W is the inertia weight, $c_1$and c_2 are acceleration coefficients and r_1 and $r_2$are random numbers between 0 and 1.

PSO's outcomes strongly relies on three key parameters. They are inertia weight (w), cognitive component (c_1), and social component (c_2). These parameters are crucial for achieving high performance by tuning optimally. Various research results have introduced advanced procedures for parameter tuning, including dynamic, adaptive, and self-tuning approaches. These processes aim to balance exploration and exploitation efficiently, enhancing convergence rates and result quality.latest advancements have focused on hybrid strategies and machine-learning-based techniques to refine parameter settings, showcasing significant improvements in PSO's efficiency and applicability.

Algorithm 1. PSO

1: Initialization;

2: Define the swarm size S and the number of dimensions n;

3: for each particle i ∈ [1..S]

4: Randomly induce Xi and $Vi,$ and assess the fitness of Xi indicating it as $f\ (Xi)$;

5: Set $Pbest_i\ =\ Xi$ and $f\ (Pbest_i)\ =\ f\ (X_i)$;

6: ending of for

7: Set $Gbest\ =\ Pbest_1$ and $(Gbest)\ =\ f\ (Pbest_1)$;

8: for each particle i ∈ [1..S]

9: if $f\ (Pbest_i)\ <\ f\ (Gbest)$ then

10: $f\ (Gbest)\ =\ f\ (Pbest_i)$;

11: ending of if

12: ending of for

13: while t < maximum iterations number

14: for each particle i ∈ [1..S] ;

15: Evaluate its velocity $v_{in}\ (t\ +\ 1)$;

16: Update the position $x_{in}\ (t\ +\ 1)$ of the particle ;

17: if $f\ (x_i\ (t\ +\ 1))\ <\ f\ (Pbest_i)$ then

18: $Pbest_i\ =\ x_i\ (t\ +\ 1)$

19: $f\ (Pbest_i)\ =\ f\ (x_i\ (t\ +\ 1)$

20: ending of if

21: if $f\ (Pbest_i)\ <\ f\ (Gbest)$ then

22: $Gbest\ =\ Pbest_i$

23: $f\ (Gbest)\ =\ f\ (Pbest_i)$

24: ending of if

25: ending of for

26: $t\ =\ t\ +\ 1$

27: end of while

28: return the $Gbest$

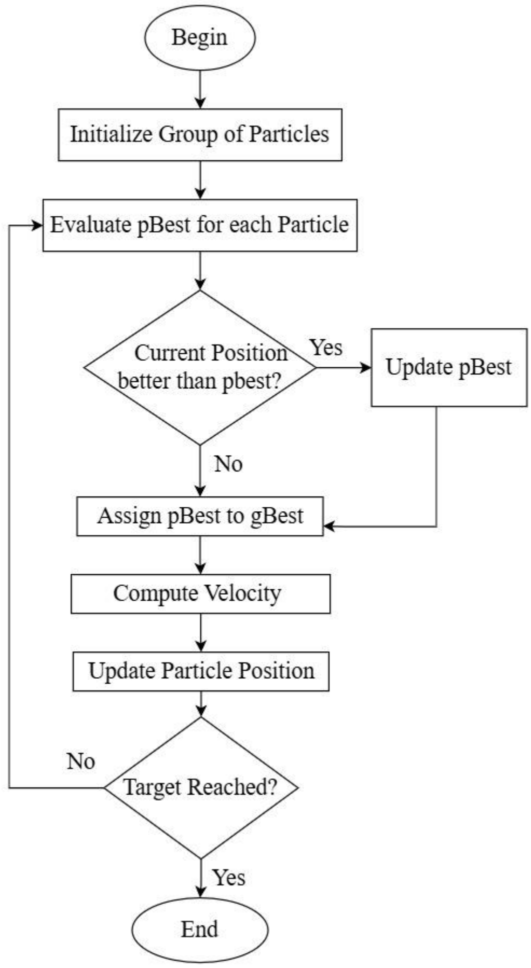

Figure 3. Flowchart of PSO.

Table 1. Applications and Related Research of PSO in various fields.

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Smart Homes	Optimized energy management using PSO	Achieved reduced costs and efficient energy use in residential buildings.	[1]
Traffic Management	PSO for urban traffic signal optimization	Reduced congestion and improved traffic flow.	[21]
Power Grid Optimization	PSO forload flow optimization	Enhanced grid reliability and reducedlosses.	[3]
Building Design	Heatingload prediction using PSO	Optimized energy consumption forlarge-scale buildings.	[25]
Business Centerlocation	Location optimization using PSO	Improved accessibility and cost-effectiveness of business center placement.	[2]

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Cost Prediction in Engineering	Transmissionline cost optimization using hybrid PSO	Reduced costs with better estimation accuracy.	[22]
Wireless Networks	Energy-efficient routing with PSO	Prolonged networklife and enhanced data delivery in ad hoc networks.	[25]
Image Processing	Hybrid PSO for image restoration and clustering	Improved image quality and segmentation accuracy.	[23]
Electrical Systems	PSO for power flow optimization	Improved system reliability and security under varyingload conditions.	[26]
Robotics Path Planning	Trajectory optimization in autonomous robots	Achieved smooth, collision-free motion in complex environments.	[6]
Renewable Energy Systems	Maximum power point tracking for solar systems using PSO	Increased energy efficiency under variable shading conditions.	[27]
Healthcare Systems	Disease prediction and diagnostics using PSO	Improved diagnostic accuracy for cardiovascular and diabetic conditions.	[9]
IoT Optimization	Resource allocation and energy optimization for IoT networks	Extended batterylife and improved throughput in IoT devices.	[24]
Manufacturing Optimization	PSO for scheduling in productionlines	Reduced processing time and optimized resource utilization.	[28]
Civil Infrastructure Optimization	Truss design optimization with PSO	Enhancedload distribution and minimized material usage.	[29]

1.2. Spider Monkey Optimization (SMO)

Spider Monkey Optimization (SMO) is a swarm intelligence algorithm influenced by the social organization and foraging behavior (fission-fusion dynamics) of spider monkeys. Monkeys collaboratively optimize their search for resources, by sharing information based on their positions and postures. SMO includes six key phases. They are initialization, localleader phase, globalleader phase, globalleaderlearning, localleaderlearning, and decision phase [39]. These steps help the algorithm to find out a balanced convergence between exploration and exploitation, enhancing processing speed and optimizing performance with fewer iterations [41]. SMO has been efficiently used in solving complex optimization problems across various domains, involving engineering design, machinelearning, and scheduling tasks [36].

Research shows its strengths in balancing search diversity and precision, making it appropriate for multi-objective optimization and real-world problems requiring high efficiency and robust results [39, 43].

a. Initialization Equation

The N number of spider monkeys are initialized with the upper andlower bound values.

$$\mathrm{K_{ij}} = \mathrm{K_{minj}} + \mathrm{U}(0,1) \times (\mathrm{K_{maxj}} - \mathrm{K_{minj}}) \tag{3}$$

where, Ki denotes the spider monkey, $\mathrm{K_{minj}}$ and $\mathrm{K_{maxj}}$ are thelower and upper bounds of the searching space, and U(0,1) denotes the regularly distributed function which ranges from 0 to 1.

b. Position Update (Localleader Phase)

$$\mathrm{K_{newij}} = \mathrm{K_{ij}} + \mathrm{U}(0,1) \times (\mathrm{loc_{tj}} - \mathrm{K_{ij}}) + \mathrm{U}(-1,1) \times (\mathrm{K_{rj}} - \mathrm{K_{ij}}) \tag{4}$$

where, K_{ij} denotes the $\mathrm{j^{th}}$ position of spider monkey i, Loc_{tj} is thelocalleader of $\mathrm{t^{th}}$ group, and U(−1,1) indicates the regularly distributed random number.

c. Fitness Function

$$FF = \begin{cases} \frac{1}{1+o_j} \; if \; o_i \geq 0 \\ 1 + abs(o_j) \; if \, o_i < 0 \end{cases} \tag{5}$$

The fitness FF calculates how good a monkey's position is based on o_i, which is the value of the objective function at that position and Positive or negative values are dealt with separately to normalize the fitness score.

d. Selection probability

$$Pb_i = \frac{f_i}{\sum_{i=1}^{N} f\_i} \tag{6}$$

This calculates the probability of selecting a monkey as aleader.

f_i is the fitness value of monkey I and $\sum_{i=1}^{N} f_i$ is the total fitness of all monkeys.

e. Globalleader Update

$$K_{newij} = K_{ij} + U(0,1) \times (Gloc_j - K_{ij}) + U(-1,1) \times (K_{rj} - K_{ij}) \quad (7)$$

f. Decision Phase (Final Position Update)

$$K_{newij} = K_{ij} + U(0,1) \times (Glo_{ij} - K_{ij}) + U(0,1) \times (K_{rj} - loc_{tj}) \quad (8)$$

Algorithm2. Spider Monkey Optimization (SMO)

1: Initialize the set of parameters as population, localleaderlimitloc, globalleaderlimit Glo, and perturbation rate;

2: thelocal and globalleaders are identified;

3: Update the position oflocalleader;

For each member $K_{ij} \in t^{th}$ group do

For each $j \in \{1,2,\ldots\ldots,D\}$ do

If $U(0,1) \geq per_r$ then

$$K_{newij} = K_{ij} + U(0,1) \times (loc_{tj} - K_{ij}) + U(-1,1) \times (K_{rj} - K_{ij}) ;$$

Else

$K_{newij} = K_{ij}$;

Ending of if

Ending of for

4: Update the position of globalleader;

Initialize count $Cnt = 0$;

While $Cnt <$ size of group do

If $U(0,1) > Pb_1$

$Cnt = Cnt + 1$;

Select the integer randomly from

$j \in \{1,2,\ldots\ldots,D\}$;

Select the monkey $k_r \in$ group

$$K_{newij} = K_{ij} + U(0,1) \times (Glo_{ij} - K_{ij}) + U(0,1) \times (K_{rj} - loc_{tj});$$

Ending of if

Ending of for

Ending of while

5: Performlearning throughlocal and globalleaders;

6: The positions oflocalleader and global are updated in the decision phase;

//localleader

IflLCnt $> LLL$ then

Initializelocallimiter countlLCnt $= 0$;

For each $j \in \{1,2,\ldots\ldots,D\}$ group do

If $U(0,1) > Pb_i$ then

$Knew_{ij} = K_{minj} + U(0,1) \times (K_{maxj} - K_{minj})$;

Else

$$K_{newij} = K_{ij} + U(0,1) \times (loc_{tj} - K_{ij}) + U(-1,1) \times (K_{rj} - K_{ij});$$

Ending of if

Ending of for

Ending of if

//Globalleader

If $GLCnt > GLL$ then

Initialize globallimiter count $GLCnt = 0$;

If No of groups $<$ max group then

The swarms are split into groups;

Else

Single group can be created by integrating all the groups;

Ending of if

Update the position of alllocalleaders;

7: Based on the decision of globalleader, the decision of fusion-fission is obtained;

8: If (termination) is satisfied

Stop;

Else

The globalleader position is updated in the optimal solution;

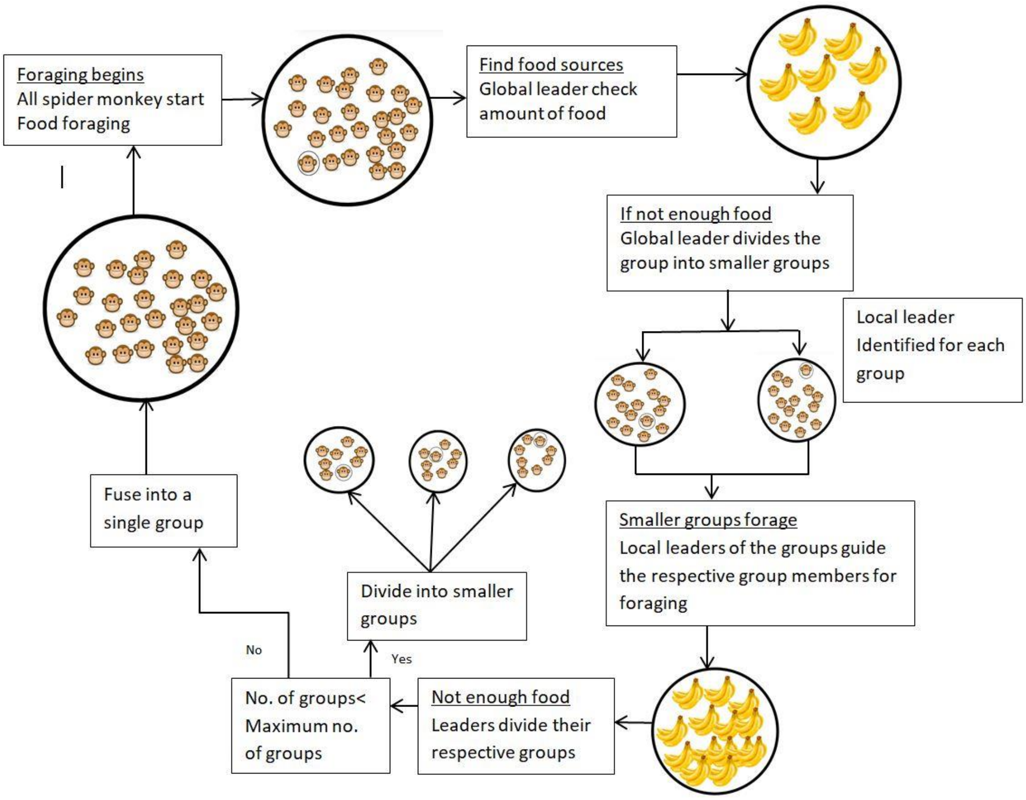

Figure 4. Spider Monkey foraging behavior analysis [30].

Table 2. Applications and Related Research of SMO in various fields.

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Android Malware Detection	SMO-based Bi-LSTM for malware classification	Achieved high accuracy in detecting Android malware for cybersecurity applications.	[31]
Electric Vehicle Power Systems	Control for interleaved parallel bidirectional DC-DC converters	Enhanced grid integration and energy efficiency in electric vehicles.	[32]
Load Flow Optimization	SMO combined with swarm intelligence forload flow in power grids	Improved efficiency and convergence in-large-scale power networks.	[33]
Wireless Networks	Smart SMO for energy-efficient wireless communication	Achieved reduced energy consumption and en-hanced nodelifetime.	[34]
Network Intrusion De-tection	Hybrid SMO with hierarchical swarm intel-ligence for feature selection	Enhanced detection accuracy for intrusion pre-vention in network security systems.	[35]
Chemical Engineering	SMO for optimization in chemical data processing	Improved hyperparameter tuning for chemical data models, increasing processing accuracy.	[12]
Structural Engineering	SMO for bridgeload optimization	Improved structural safety and cost efficiency in bridge designs.	[36]
Healthcare Applications	SMO for medical image feature extraction and segmentation	Enhanced accuracy in disease diagnosis through optimized image processing.	[37]
Renewable Energy Sys-tems	SMO for wind farm placement optimization	Achieved higher energy efficiency and reduced setup costs in renewable energy projects.	[38]
Machinelearning Opti-mization	SMO for optimizing deeplearning hyperpa-rameters	Increased model accuracy with efficient hyperpa-rameter tuning.	[39]
Robotics and Path Plan-	SMO for robot trajectory optimization in	Improved obstacle avoidance and energy effi-	[40]

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
ning	dynamic environments	ciency in robotic movements.	
IoT Network Management	SMO for bandwidth and energy optimization in IoT networks	Enhanced network utilization and extended devicelife in IoT applications.	[41]
Bioinformatics	SMO for gene selection in protein structure analysis	Achieved higher predictive accuracy in bioinformatics applications.	[42]
Civil Infrastructure Optimization	SMO for optimizing truss designs	Enhancedload distribution and material utilization inlarge-scale truss structures.	[36]
Transportation Optimization	SMO for vehicle routing in urbanlogistics	Improved delivery efficiency and reduced transportation costs.	[43]

1.3. Grey Wolf Optimization (GWO)

The Grey Wolf Optimization (GWO) algorithm refines exploration and exploitation. Due to its simplicity, few control parameters, and high convergence accuracy, it reduces the risk oflocal optima and gaining popularity [44, 45]. Researchers have been enhancing GWO by integrating it with other meta-heuristics and using advanced strategies to address various optimization challenges. For example, hybrid approacheslike GWO-SCA, WOAGWO, and GWO-PSO have enhanced performance in engineering applications such as PV model parameter extraction, pressure vessel design, and reactive power scheduling [46, 47]. Strategy-based enhancements such as K-means clustering, stochasticlearning, and nonlinear convergence factors, have also reinforced GWO's exploration and convergence capabilities [48, 49].

Applications of GWO span various fields, including PV parameter estimation, emotion recognition, structural optimization, and investment predictions [50]. GWO contends with highly confined problems or those including varinouslocal extrema [51, 52]. This study presents an improved GWO algorithm integrating reverselearning, nonlinear convergence strategies, and concepts from Tunicate Swarm and Particle Swarm algorithms to address theselimitations. Benchmark tests and real-world engineering problems validate its enhanced accuracy, robustness, and applicability [53].

A. Basic Background

Gray wolveslive in packs with a well-structured social hierarchy. At the top of this, is the α-wolf, wholeads the pack by making important decisions about hunting strategies, food distribution, and choosing resting places. β-wolves support theleader and are ranked second. They assist the α-wolf in decision-making processes. δ-wolves follows them, who hold tertiary roleslike scouting, patrolling, and acting as guards. Atlast there are ω-wolves, whose primary role is to create harmony within the group and maintain social dynamics. This hierarchical structure and the predation behavior of gray wolves are important to their pack dynamics and have contributed optimization models., let the solution space be represented in d dimensions and the population size by N, in the context of the GWO algorithm for optimization challenges. Thelocation of the i-th wolf within this pack indicates one probable solution in the search space expressed as:

$$X_i = \{X_i^1, X_i^2, \ldots\ldots, X_i^d\}, i = 1, 2, \ldots\ldots, N \tag{9}$$

α, β, and δ, respectively denotes the optimal, sub-optimal, and third optimal solutions in the gray wolf population, and the rest of the solutions are indicated as ω. ω constantly updates the position based on the positions of α, β, and δ in order to search the best solution or the optimal position. The positions of the gray wolves are calculated as:

$$D = |C \cdot X_p(t) - X(t)| \tag{10}$$

$$X(t + 1) = X_p(t) - A \cdot D \tag{11}$$

Where i is pack size, D is the distance between current wolf position and the best solution, $X(t)$ is the

position of wolf iteration t, $X_p(t)$ denotes the position of prey at iteration t, t denotes the current

position, $X(t + 1)$ denotes the updated position of the wolf, A and C are coefficient vectors.

$$A = 2a \cdot rand() - a \tag{12}$$

$$C = 2 \cdot rand() \tag{13}$$

This is obtained by minimizing the value of a in the Eq. (12). Note that the oscillation range of A islikewise decreased by a. A is a random number in the interval [−a, a] where a is decreased from 2 to 0 throughout iterations.

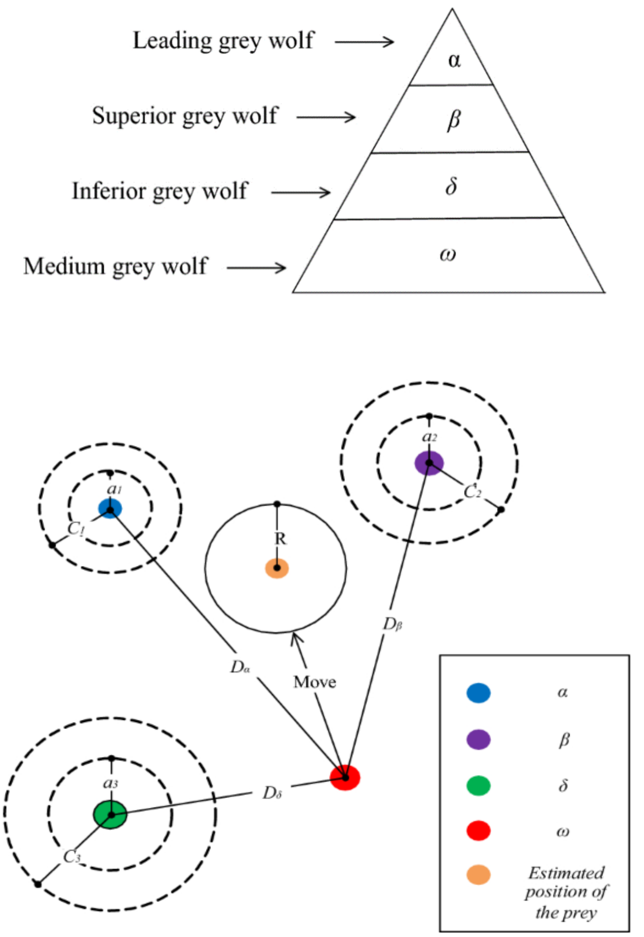

Figure 5. *Schematic diagram of gray wolf population hierarchy and predation processes [54].*

It is assumed that the alpha wolf (representing the best solution), along with the beta and delta wolves, possess superior knowledge regarding the possiblelocation of the prey, to mathematically model the hunting behavior of gray wolves. As a result, the top three best solutions identified so far are held. The remaining search agents update their positions based on the guidance provided by theseleading search agents including the ω - wolves.

$$D = \begin{cases} D_\alpha = |C_1 \cdot X_\alpha - X| \\ D_\beta = |C_1 \cdot X_\beta - X| \\ D_\delta = |C_1 \cdot X_\delta - X| \end{cases} \quad (14)$$

$$X = \begin{cases} X_1 = X_\alpha - A_1 \cdot (D_\alpha) \\ X_2 = X_\beta - A_2 \cdot (D_\beta) \\ X_3 = X_\delta - A_3 \cdot (D_\delta) \end{cases} \quad (15)$$

$$X(t+1) = \frac{X_1 + X_2 + X_3}{3} \quad (16)$$

Algorithm3: Grey Wolf Optimization Algorithm
1: Initialize the grey wolf population X_i $(i = 1, 2, \ldots, n)$;
2: Initialize the coefficient vectors a, A and C ;
3: Calculate the fitness of each search agent (wolf);
4: X_α = the best search agent (wolf);
5: X_β = the second best search agent (wolf);
6: X_δ = the third best search agent (wolf);
7: while iteration < maximum Iteration, do
8: for each wolf X_i,do
9: Update the position of current wolf X_i ;
10: Update the position of wolf X_i,if exceed boundaries ;
11: ending of for
12: Update the coefficient vectors a, A and C ;
13: Calculate the fitness of all search agents (wolfs);
14: Update the value of X_α, X_β and X_δ;
15: ending of while
16: return X_α;

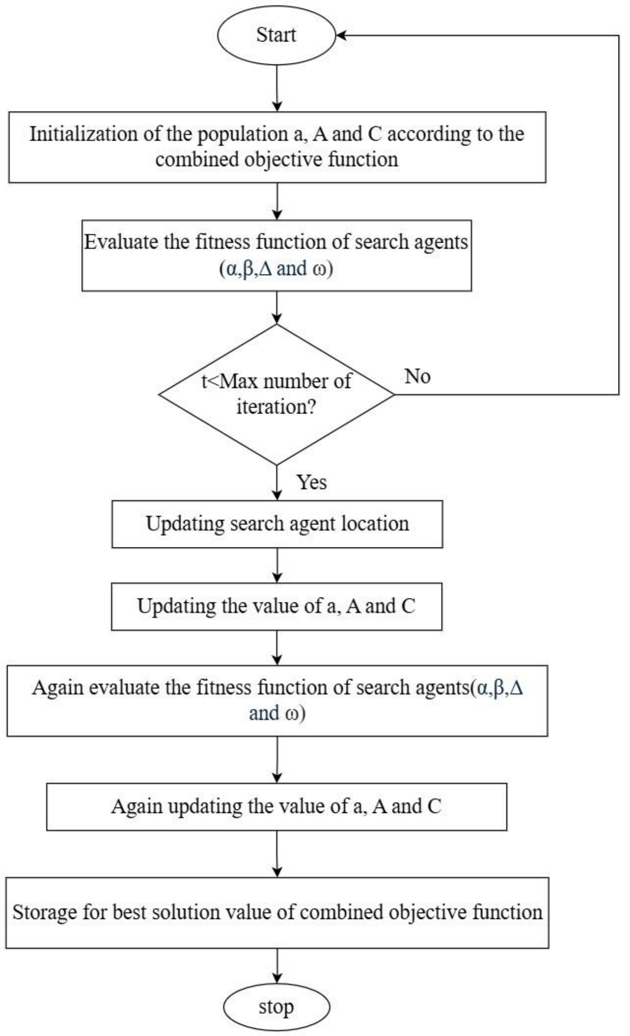

Figure 6. *Flowchart of Grey Wolf Optimization.*

Table 3. *Applications and Related Research of GWO in various fields.*

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Structural Engineering	Enhanced GWO for structural optimization	Achieved improved efficiency and stability in building designs.	[44]
Renewable Energy Systems	GWO for optimizing hybrid renewable energy systems	Improved energy efficiency and power balancing in solar-wind hybrid systems.	[45]
Control Systems	Adaptive GWO for PID controller design	Enhanced performance in industrial control systems with optimal parameter tuning.	[51]
Electromagnetic Systems	GWO-based antenna array optimization	Achieved better directional performance with reduced design costs.	[46]
Healthcare Applications	GWO for feature selection in disease diagnostics	Improved classification accuracy in cancer and diabetes detection.	[47]
Machinelearning	Integration of GWO with deeplearning models	Optimized hyperparameter tuning for improved model performance.	[48]
IoT and Network Systems	GWO for resource allocation in IoT networks	Enhanced bandwidth utilization and reducedlatency inlarge-scale IoT networks.	[50]
Robotics and Path Planning	GWO for robot trajectory optimization	Improved obstacle avoidance and energy efficiency in dynamic environments.	[55]
Power Systems	GWO forload frequency control in power grids	Achieved better frequency regulation and stability in smart grids.	[52]
Environmental Monitoring	GWO for optimizing sensor deployment	Improved coverage and reduced costs in environmental monitoring systems.	[56]
Bioinformatics	GWO for protein structure prediction	Enhanced accuracy in determining stable protein conformations.	[57]
Transportation andlogistics	GWO for vehicle routing problem	Improved delivery efficiency and reduced transportation costs.	[58]
Financial Applications	GWO for stock market prediction	Optimized trading strategies with higher predictive accuracy.	[53]
Energy Optimization	GWO for maximum power point tracking in solar panels	Enhanced energy harvesting under partial shading conditions.	[49]
Civil Engineering	GWO for optimizing truss structures	Improvedload-bearing efficiency and material usage in bridge designs.	[59]

1.4. Grasshopper Optimization Algorithm (GOA)

The Grasshopper Optimization Algorithm (GOA) is a bio-inspired meta-heuristic optimization method that follows the swarming behavior of grasshoppers during theirlife cycle. Grasshoppers show specific movement patterns in their nymph and adult stages, which are effectively modeled in GOA to balance exploration and exploitation in optimization tasks. Grasshoppers formlarge groups with slow, coordinated movements, resemblinglocal exploration to refine solutions in nearby areas during the nymph phase [60]. On the other hand adult grasshoppers show sudden, wide-ranging movements during aerial migrations, assisting global exploration of the search space [61]. The dual-phase behavior indicates the need to discover diverse regions of a problem space while adjusting favorable solutions.

To simulate the swarm's collective behavior, GOA utilizes social interaction mechanismslike attraction, repulsion, and alignment. These mechanismslead candidate solutions (agents) towards best results by balancing wide-range exploration and precise exploitation [62, 63]. GOA is particularly valued for its simplicity and adaptability in solving complex multi-modal optimization problems thus it is broadly used in engineering, robotics, energy systems, and machinelearning, [64]. Its ability to emulate natural processes makes it an efficient tool for facing real-world problems.

$$X_i^d = c\left(\sum_{j=1}^{N} c\frac{ub_d - lb_d}{2} s\left(\left|X_j^d - X_i^d\right|\right)\frac{X_j - X_i}{d_{ij}}\right) + T_d, J \neq i \quad (17)$$

Equation (18) is used for the update the position of the grasshopper.

Common term and parameters are used in the mathematical models:

n is the population size of the grasshoppers, d indicates the population dimension, X_i indicates the position of the ith grasshopper, X_i^d indicates the updated position of the ith grasshopper, t denotes as the current iteration, t_{max} indicates the maximum iteration, $s(r)$ is the social component, l is the attraction force, f is the repulsion force and T_d indicates the optimal solution so far.

Where ub_d indicates the upper bound in the d^{th} dimension, lb_d indicateslower bound in the d^{th} dimension, T_d indicates optimal solution found so far, and c is a decreasing coefficient tolessen the comfort zone, repulsion zone, and attraction zone.

$$c = c_{max} - \left(\frac{c_{max} - c_{min}}{t_{max}}\right) \quad (18)$$

The social component $s(r)$ is defined as:

$$s(r) = fe^{\frac{-r}{l}} - e^{-r} \quad (19)$$

Where $= \left|X_j^d - X_i^d\right|$, here, l denotes as the attraction force and f denotes as the repulsion force.

Algorithm4. Grasshopper Optimization Algorithm (GOA)

1: Initialize the grasshopper population X_i $(i = 1, 2, \dots, n)$;

2: Calculate the fitness of each search agent (grasshopper);

3: T = the best search agent (grasshopper)

4: while iteration no < maximum iteration no, do

5: Update the value of c;

6: for each grasshopper X_i, do;

7: Normalize the distance between the grasshoppers;

8: Update the position of current grasshopper X_i;

9: Update the position X_i; if exceed boundaries

10: ending of for

11: Update Td if there is a better solution;

12: ending of while

13: return T_d;

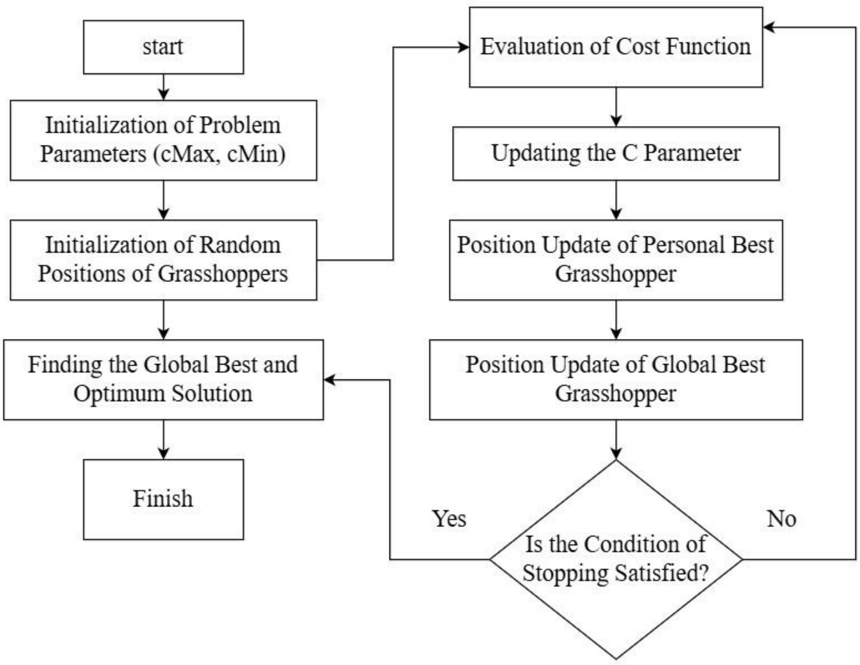

Figure 7. *Flowchart of Grasshopper Optimization Algorithm (GOA).*

Table 4. *Applications and Related Research od GOA in various fields.*

Application Area/Field	**Proposed Method/Approach**	**Strengths/Contribution**	**Reference**
Power Systems Control	Standard GOA for controlling power systems	Efficient balancing of control parameters and optimization in complex systems.	[62]
Routing in FANETs	Hybrid GOA with Invasive Weed Optimization	Enhanced routing efficiency and reduced computational overhead for network optimization.	[65]

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Structural Analysis	Finite Element Method (FEM) plugin in Grasshopper	Improved structural modeling and optimization using a parametric environment.	[66]
Lung Cancer Classification	Binary GOA combined with Artificial Bee Colony	Effective feature selection and classification in medical applications using deeplearning.	[63]
Pavement Crack Detection	GOA integrated with U-Net framework	Accurate crack detection and condition scoring for pavement maintenance.	[67]
Machinelearning Optimization	GOA for optimizing machinelearning models	Achieved better refinement and performance of predictive models in healthcare applications.	[64]
Wind Farm Power Systems	GOA forload Frequency Control (LFC)	Improved dynamic stability in power systems incorporating renewable energy.	[60]
Tunnel Design	GOA for iterative and dynamic tunnel modeling	Enhanced geometric adjustments and airflow optimization in tunnel design.	[68]
Interior Design	Parametric modeling with Grasshopper optimization	Improved customization and innovation in furniture and architectural design.	[69]
Conceptual Design Process	Computational design methods based on Grasshopper	Facilitated brainstorming and creative solutions in the early stages of design.	[70]
Concrete Dam Optimization	GOA for gravity dam design	Achieved better structural stability and resource efficiency.	[10]
Photovoltaic Systems	Improved GOA for global maximum power tracking	Enhanced energy efficiency in solar panels under varying conditions.	[61]
Fuzzy Neural Networks	GOA integrated with Recurrent Fuzzy Neural Networks	Accurate predictions of surface ozonelevels using hybrid optimization methods.	[71]
Energy Management in Micro-Grids	Modified Chaos GOA for optimizing renewable energy output	Achieved higher efficiency in hybrid renewable energy systems.	[8]
Heart Disease Prediction	GOA-optimized Convolutional Neural Network (CNN)	Improved prediction accuracy and computational performance for medical diagnostics.	[72]

1.5. Cuckoo Search Optimization (CSO)

The Cuckoo Search (CS) algorithm is a population-based meta-heuristic optimization technique. It is known for its simplicity, minimal parameter requirements, and effective global search capabilities [73]. It uses thelévy flight strategy to produce new solutions, allowing intense exploration of the solution space while maintaining diversity. However it canlead to reducedlocal exploitation and slower convergence, particularly in complex optimization problems as the algorithm relies on highly random movements [16].

Improvements to the algorithm have focused on parameter control and hybridization to address these problems. Adaptive schemes have been broadly applied, including strategieslike dynamic updates of discovery probability, Cauchy distribution, andlehmer mean to enhance convergence performance in parameter control. Other approaches have involved dynamically fine-tuning step size and probability parameters, showing improved results in different benchmarks and constrained optimization works [74-77].

Hybridization with other algorithms has also proven efficient in improving CS performance. Combining CS with optimization techniques such as Grey Wolf Optimization, Quantum-Behaved Particle Swarm Optimization, and Bat Algorithm to refine solution quality, improve exploration, and balance population diversity are some of the hybridization in practice. In field such as engineering design, medical diagnostics, data clustering, and predictive modeling, variants such as Dynamic CS, Quantum-Inspired CS, and Adaptive CS have been successfully applied. These improvements address key challenges in optimization, exhibiting the adaptability and effectiveness of the CS algorithm in solving complex, real-world challenges.

Algorithm5. Cuckoo Search Optimization Algorithm (CSO)

1: Start;

2: Objective function $(x), x = (x_1, \ldots, x_d)^T$;

3: Initial a population of n host nests x_i $(i = 1,2, \ldots, n)$;

4: while (t < Maximum Generation) or (stop criterion)

5: Get a cuckoo (say i) randomly and generate a new solution bylévy flights;

6: Evaluate its quality/fitness; Fi Choose a nest among n (say) randomly;

7: if ($F_i > F_j$), Replace j by the new solution

8: ending of if

9: Abandon a fraction (Pa) of worse nests [and build new ones at newlocations vial évy flights];

10: Keep the best solutions (or nests with quality solutions);

11: Rank the solutions and find the current best;

12: ending of while

13: Post process results and visualization;

14: ending

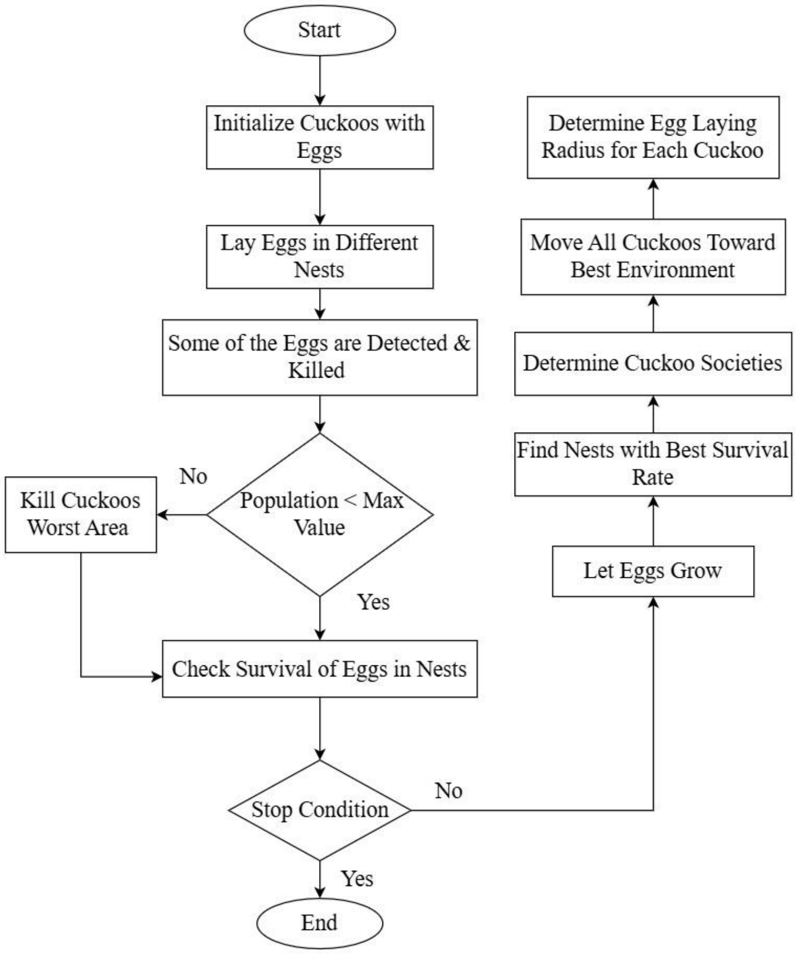

Figure 8. *Flowchart of Cuckoo Search Optimization.*

Table 5. *Application and related research of Cuckoo Search Optimization in various fields.*

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Power Systems Control	Reactive power compensation using CS for grid system optimization	Improved stability and efficiency in grid systems with FACTS devices.	[16]
Network Telemetry	Structure-aware CS for real-time traffic monitoring	Enhanced data tracking in modern computer networks.	[78]
Renewable Energy Systems	CS for photovoltaic power forecasting	Increased accuracy in solar irradiance and photovoltaic power predictions.	[79]
Fault Diagnosis	Hybrid CS for gas turbine engine fault identification	Enhanced diagnostic accuracy for gas turbine systems with constrained nonlinear optimization.	[80]
Data Mining Optimization	Dynamic CS combined with neutrosophic cognitive mapping	Improved feature selection and clustering efficiency inlarge datasets.	[74]

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Wind Power Prediction	CS for wind power installed capacity forecasting	Accurate predictions for energy capacity planning in renewable systems.	[75]
Vehicular Networks	CS for resource allocation in vehicular networks	Efficient caching and offloading in resource-constrained vehicular systems.	[76]
Groundwater Contamination	CS for identifying contamination sources	Improved environmental monitoring with kernel extremelearning machines.	[77]
Structural Engineering	CS-based optimization for concrete dam design	Achieved better structural stability and resource utilization.	[81]
Healthcare Systems	CS for disease classification and prediction	Enhanced diagnostic accuracy for medical imaging and classification tasks.	[82]
Image Segmentation	Triple hybrid CS with Type II fuzzy sets	Improved multi-level image segmentation with adaptive mechanisms.	[83]
Engineering Design Problems	Multi-algorithm CS with adaptive mutation mechanism	Enhanced constraint handling and optimization efficiency.	[84]
Mechanical Design	Enhanced CS withlevy flight and GANs	Optimized mechanical manufacturing processes with generative models.	[85]
Advanced Machining	CS for optimization of machining parameters	Improved accuracy in machining tasks with reduced waste.	[86]
Bearing Fault Diagnosis	Adaptive CS for noise-resistant fault diagnosis	Achieved effective fault identification under strong noise conditions.	[87]

1.6. Ant Colony Optimization Algorithm

Ant Colony Optimization (ACO), first introduced by in 1991, is a population-based meta-heuristic algorithm influenced by the foraging behavior of ants. Ants guide their movements toward food sources by communicating indirectly through pheromone trails. This biological behavior has been applied into optimization algorithms to solve complex problems. Recent studies have exhibited the flexibility of ACO in solving real-world challengeslikelogistics, routing, and scheduling [89, 95]. ACO has evolved with hybrid approaches and parameter optimization techniques, improving its performance and extending its applicability over the years. ACO has been successfully used across diverse fields. ACOleverages to optimize tourist itineraries and to achieve reduced travel times. ACO is applied to improve the accuracy oflaser drilling processes, reducing waste and improving efficiency in industrial settings [88]. likewise, ACO'sleveraged effectiveness in optimizing vehicular and power network routing [96]. These applications exhibits ACO's adaptability and efficacies in solving complex optimization challenges. The basic advantages of ACO include its adaptability to dynamic environments and ability in solving multi-objective optimization problems [89, 96]. Due to its parallel search capabilities, ACO is particularly efficient inlarge-scale problems. Thelimitations of ACO involves suffering from slow convergence in complex scenarios and is prone to stagnation inlocal optima, as the computational cost of ACO increases significantly as the problem size grows [91]. These problems needs further research to improve the algorithm's efficiency and scalability.

The food of ants can be represented by the destinations. All ants are randomly positioned at either one or any of the nodes of the transport network at the beginning of the evolution. The probability of transition of any ant to an adjacent node from time t to $t+1$ is found by using following equation:

$$f(x) = \begin{cases} \frac{\tau_{ij.pq}^{\alpha}(t).\eta_{ij.pq}^{\beta}(t)}{\sum_{u=1}^{m_p}\tau_{ij.pu}^{\alpha}(t).\eta_{ij.pu}^{\beta}(t)}, \ q = 1,2,\ldots..,m_p \\ 0, otherwise \end{cases} \quad (20)$$

Where $\tau_{ij.pq}^{\alpha}$ indicates the intensity of trial on edge (e_{ij}, e_{pq}) at time, $\tau_{ij.pq}^{\alpha} \in (\tau_{min,}\tau_{max})$.

$\eta_{ij.pq}$ is the visibility of edge (e_{ij}, e_{pq}), α is the relative importance of trial, $\alpha \geq 0$, β is the relative importance of the visibility, $\beta \geq 0$.

When α =0, the jobs with the shortest processing times are morelikely to be chosen. Whichleading to a classical stochastic algorithm. Only pheromone amplification is at work which willlead to the pre-mature convergence of the method to strongly sub-optimal solution if on the contrary $\beta \geq 0$. Therefore, the transition probability demonstrate a compromise between visibility i.e, the shorter the processing time the higher the probability to choose it and trial intensity (the higher the traffic on the arc $(e_{ij}, e_{pq}$), the higher its attractiveness).

Ants select an adjacent node using Eq. (21), and this continues until all ants move to a neighboring node, completing what is called an iteration or cycle. After this, the trial intensity is updated using Eq. (21).

$$\tau_{ij.pq}(t+n) = (1-\rho)\tau_{ij.pq}(t) + \Delta\tau_{ij.pq}(t) \quad (21)$$

$$\Delta\tau_{ij.pq}(t) = \sum_{k=1}^{m} \Delta\tau_{ij.pq}{}^{k}(t) \quad (22)$$

Where ρ is the coefficient that represents the evaporation of trial between time t and $+n, \rho \subset [0,1)$. $\Delta\tau_{ij.pq}(t)$ is the total intensity of node (e_{ij}, e_{pq}) during an iteration of ants, m is the total number of ants, $\Delta\tau_{ij.pq}{}^{k}(t)$ is the amount of pheromone ant k deposits on the areas it has visited. This usually amounts to the value:

$$\Delta\tau_{ij.pq}{}^{k} = \frac{Q}{Z(C_k)} \quad (23)$$

Where $Z(C_k)$ is thelength of the tour and Q is a positive constant.

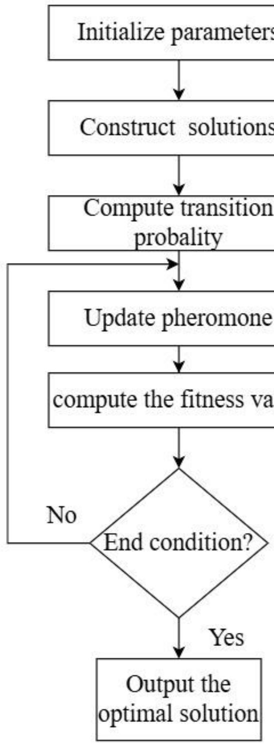

Figure 9. *Flowchart of the Ant Colony Optimization.*

Table 6. *Application and Related research of ACO in various fields.*

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
Agriculture	Boosting agriculture and water efficiency with advanced ACO	Enhanced accuracy and efficiency in predictive modeling.	[89]
Tourist Route Optimization	ACO-based recommendation system	Reduced travel times and optimized itineraries.	[90]
Human Resource Management	ACO for job candidate optimization	Improved recruitment and resource management.	[91]
Robotic Swarm Cleaning	Bee-inspired ACO for robotic cleaners	Improved navigation and task completion in industrial setups.	[92]
Load Balancing in Computing	Dynamic ACO for serverload balancing	Reduced downtime and improved computation efficiency.	[93]
Energy Monitoring	ACO with neural networks for harmonic distortion monitoring	Enhanced detection and prediction in energy systems.	[94]
Microenterprise Vulnerability	ACO for fuzzy geodemographic clustering	Improved business vulnerability analysis.	[95]
Logistics and Routing	Improved ACO for integratedlogistics optimization	Enhanced delivery efficiency and cost reduction.	[96]
Network Optimization	Multi-ACO for vehicular routing problem	Optimized traffic flow and resource utilization.	[97]
Humanitarian Aid Distribution	ACO forlocation routing problem	Improved speed and efficiency in critical resource distribution.	[98]
Laser Drilling Optimization	ACO with gradient descent for precisionlaser drilling	Achieved higher accuracy and reduced waste.	[88]

Application Area/Field	Proposed Method/Approach	Strengths/Contribution	Reference
3D Containerloading	Hybrid ACO for multi-objective optimization	Improved packing efficiency and resource utilization.	[99]
Carbon Emissions Modeling	ACO with Cobb-Douglas models for emission analysis	Enhanced understanding of emission patterns and impacts.	[100]
Power and Transportation Networks	Collaborative ACO for urban transportation and power systems	Improved integration and efficiency.	[101]
Edge-Cloud Resource Allocation	Bi-directionallSTM and ACO for adaptive resource scheduling	Enhanced cloud computing efficiency.	[102]

2. Comparative Analysis

Table 7. Comparison of Population-Based Algorithms.

Algorithm	Inspiration	Strength	Limitations	Scalability	Computational Complexity	Flexibility	Convergence Rate
PSO	Swarm behavior of birds and fish	Simple implementation, effective in dynamic systems	Prone to premature convergence	Moderate Scalability in medium sized problems	Lower computational cost compared to others	High adaptability to dynamic environments	Fast convergence but risks local optima
SMO	Social behavior of spider monkeys	Effective in multi objective optimizations tasks	Slower convergence in complex scenarios	Moderate Scalability with hybrid enhancements	Higher computational cost in large scale problems	Good adaptability in structured problems	Moderate convergence requires parameter tuning
GWO	Social hierarchy and hunting behavior of grey wolves	Simplicity, low parameter dependency,	Balancing global and local search is challenging	High Scalability in large problem spaces	Moderate computational cost	Flexible for different optimizations problems	Efficient convergence in static environments
GOA	Swarming behavior of grasshoppers	Strong exploration abilities	High sensitivity to parameter settings	Moderate scalability in medium complexity tasks	Higher computational complexity	Good flexibility in multimedia tasks	Fast convergence in structured environments
CSO	Brood parasitism of cuckoos	Excellent global exploration capabilities	Poor local search requires hybridization	Limited scalability for very large data sets	Higher computations demands	Moderate flexibility in constrained problems	Effective in global optimization tasks
ACO	Pheromone laying behavior of ants	Best for combinational problems	High computational cost for large problems	Moderate scalability with hybrid approaches	Significant computational complexity	Highly flexible for discrete problems	Slower convergence in dynamic scenarios

3. Usage and Performance Analysis of Bio-Inspired Optimization Algorithms

The pie chart below shows the proportion of research studies using each of the six bio-inspired optimization algorithms over the past decade. The algorithms based on benchmark such as convergence speed, solution accuracy, and robustness across various applications are shown below as the bar graph which compares the average performance efficiency of each algorithm.

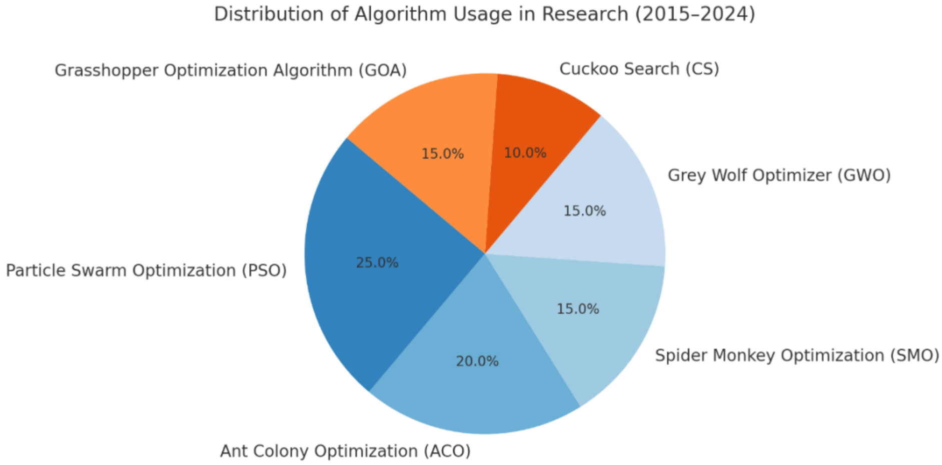

Figure 10. *Pie Chart of Distribution of Algorithm Usage in Research.*

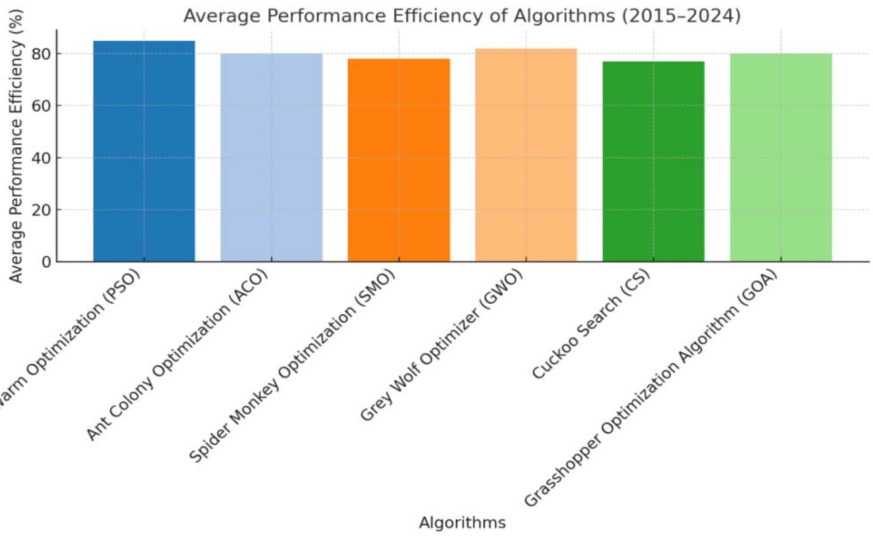

Figure 11. *Bar graph of the Average Performance Efficiency of Algorithms.*

4. Discussion

This section provides an extensive review of the bio-inspired optimization algorithms which include Particle Swarm Optimization (PSO), Spider Monkey Optimization (SMO), Grey Wolf Optimization (GWO), Grasshopper Optimization Algorithm (GOA), Cuckoo Search Optimization (CSO), and Ant Colony Optimization (ACO). Such algorithms use natural processes and behaviors and are efficient in solving complicated optimization issues. These algorithms keep the diversity of the population and do not converge early by simulating such natural processes as group behavior, hierarchy, and distribution of resources. However, they also show efficiency in terms of computational complexity and dependency on parameters that start the algorithm, which affects their scalability and performance mainly with problems of high dimensionality.

Metaheuristic algorithms such PSO and GWO are some of the bio inspired algorithms that have proven to be efficient in numerous engineering fields. For example, PSO has been used in energy management of smart homes and smart grid applications to prove its applicability in dynamic environments.likewise, GWO has been applied to resource allocation for IoT networks andload frequency control for power systems. These algorithms have evaluation-exploitation phases in which they can come close to the finest solutions efficiently. However, they have some difficulties, for example, premature convergence in multimodal search spaces, which needs high escape methods to avoidlocal optima.

Due to the fact that SMO and GOA are particularly suitable for multi-objective optimization problems. SMO, based on the social interactions of the spider monkeys, performs well in the problems that require pyramid decision making and categorization. Recent research work has demonstrated its use inlow power wireless communication and in feature extraction of medical images, thus making it evident that it is a very general purpose filter. On the other hand, GOA copies the swarming behavior of grasshoppers and has been used in energy control and in the optimization of machinelearning model. However, both algorithms heavily depend on the parameter tuning and, therefore, may not be very efficient forlarge data sets or complex environments.

CSO and ACO is based on the exploration of the optimization mechanisms. CSO, based on the brood parasitism behavior of cuckoos, employsl évy flights to traverse the solution space efficiently. New trendslike the hybrid CSO models for fault detection of gas turbines and photovoltaic power prediction have enhanced its usage in the renewable energy and mechanical systems. ACO which simulates the foraging behavior of ants is well known for its performance in solving routing and scheduling issues. Its usage in routing of vehicular network and in the collaborative optimization of urban transport confirms its ability to solve reallifelogistics problems. However, the computational complexity of ACO is directly proportional to the problem size, and thus, the problem size must be reduced or hybrid with other algorithms or adaptive methods to improve scalability.

Despite the complexity of the calculations, bio-inspired algorithms are flexible and can be adapted easily. They give reliable results in different problem contexts, which is why they are indispensable in such areas as medicine and transportation, engineering and geophysical modeling. However, several issues are still worth discussing. Generalization of the procedure is also a problem due to high computational costs and sensitivity to the parameter initialization, especially in the high-dimensional cases. To overcome these drawbacks, next studies could concentrate on the application of parallel computing methods and varying step size control strategies which vary during the optimization process to perform a fine balance between exploration and exploitation.

Furthermore, the integration of these algorithms, for example, CSO-ACO or incorporating machinelearning into the algorithms, provides potential for improvement. For instance, the integration of global searching capability of ACO with-local optimizing strategies of PSO can enhance the convergence speed and yet at the same time, would decrease the computational burden. In addition to that, it is possible to integrate machinelearning models that can rapidly adjust to time-sensitive conditions such as hyper parameter tuning in order to enhance the results of the optimization techniques. Such developments can bring together the theoretical aspects of optimization and real-world problem-solving scenarios that canlead to the creation of new and better solutions in a constantly expanding range of problem areas.

5. Conclusion

This paper gives the broad analysis of bio-inspired meta-heuristic population based optimization methods. Which demonstrated the efficacy and flexibility of these algorithm, in solving complex optimization problems across various fields. The key patterns and strengths of these challenges were identified by assessing the usage trends, applications and performance efficiencies of algorithms such as Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), Grasshopper Optimization Algorithm (GOA), Spider Monkey Optimization (SMO), Grey Wolf Optimizer (GWO), and Cuckoo Search (CS).

The PSO has a superior convergence speed and solution accuracy which emerges as the most broadly used and effective algorithm. Where as GOA and SMO explored growing relevance and adaptability. Algorithmslike Cuckoo Search also proved effective for niche applications, although theirlower frequency of usage.

Thus bio-inspired population based optimization techniques represent a strong class of problem-solving methodologies, providing intelligent, adaptable, and efficient solutions to real-world and theoretical problems. This paper direct to provide fellow researchers with clear insights for choosing an appropriate method tailored to their specific needs in the field of path planning. Future work can focus on hybridizing these algorithms toleverage their individual strengths by further enhancing their applicability in dynamic and high-dimensional problem spaces

Abbreviations

PSO	Particle Swarm Optimization
SMO	Spider Monkey Optimization
GWO	Grey Wolf Optimization
GOA	Grasshopper Optimization Algorithm
CS	Cuckoo Search
ACO	Ant Colony Optimization

Author Contributions

Yam Krishna Poudel: Methodology, Validation, Writing – original draft, Writing – review & editing

Jeewan Phuyal: Data curation, Methodology, Project administration, Software, Visualization

Rajiv Kumar: Supervision, Validation

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] A. Ahmad *et al.*, "An Optimized Home Energy Management System with Integrated R enewable Energy and Storage Resources," *Energies*, vol. 10, no. 4, p. 549, Apr. 2017, https://doi.org/10.3390/en10040549

[2] J. U *et al.*, "Particle Swarm Optimization based Spatiallocation Allocation of Urban Parks," *2014 Third Int. Conf. Agro-Geoinformatics*, no. March, pp. 1–6, 2014.

[3] W. Al-Saedi, S. W. lachowicz, D. Habibi, and O. Bass, "Power flow control in grid-connected microgrid operation using Particle Swarm Optimization under variableload conditions," *Int. J. Electr. Power Energy Syst.*, vol. 49, pp. 76–85, Jul. 2013, https://doi.org/10.1016/j.ijepes.2012.12.017

[4] S. M. Haakonsen, S. H. Dyvik, M. luczkowski, and A. Rønnquist, "A Grasshopper Plugin for Finite Element Analysis with Solid Elements and Its Application on Gridshell Nodes," *Appl. Sci.*, vol. 12, no. 12, 2022, https://doi.org/10.3390/app12126037

[5] S. R. Kamel and R. Yaghoubzadeh, "Feature selection using grasshopper optimization algorithm in diagnosis of diabetes disease," *Informatics Med. Unlocked*, vol. 26, p. 100707, 2021, https://doi.org/10.1016/j.imu.2021.100707

[6] J. J. Kim and J. J. lee, "Trajectory optimization with particle swarm optimization for manipulator motion planning," *IEEE Trans. Ind. Informatics*, vol. 11, no. 3, pp. 620–631, 2015, https://doi.org/10.1109/TII.2015.2416435

[7] P. Hou, W. Hu, M. Soltani, and Z. Chen, "Optimized Placement of Wind Turbines inlarge-Scale Offshore Wind Farm Using Particle Swarm Optimization Algorithm," *IEEE Trans. Sustain. Energy*, vol. 6, no. 4, pp. 1272–1282, Oct. 2015, https://doi.org/10.1109/TSTE.2015.2429912

[8] Z. Yan, Y. li, and M. Eslami, "Maximizing micro-grid energy output with modified chaos grasshopper algorithms," *Heliyon*, vol. 10, no. 1, p. e23980, Jan. 2024, https://doi.org/10.1016/j.heliyon.2024.e23980

[9] H. H. Inbarani, A. T. Azar, and G. Jothi, "Supervised hybrid feature selection based on PSO and rough sets for medical diagnosis," *Comput. Methods Programs Biomed.*, vol. 113, no. 1, pp. 175–185, Jan. 2014, https://doi.org/10.1016/j.cmpb.2013.10.007

[10] S. Abbasi *et al.*, "Design optimization of concrete gravity dams using grasshopper optimization algorithm," *Innov. Infrastruct. Solut.*, vol. 9, no. 12, p. 453, Dec. 2024, https://doi.org/10.1007/s41062-024-01741-w

[11] M. Prabakaran, M. K. Bhole, V. Kalpana, S. Dixit, K. Divya, and D. A. Chauhan, "Enhancing Disease Prediction in Healthcare: A Comparative Analysis of PSO and Extremelearning Approach," in *2023 3rd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA)*, IEEE, Dec. 2023, pp. 1092–1097. https://doi.org/10.1109/ICIMIA60377.2023.10426121

[12] D. Kumari, A. Sinha, S. Dutta, and P. Pranav, "Optimizing neural networks using spider monkey optimization algorithm for intrusion detection system," *Sci. Rep.*, vol. 14, no. 1, p. 17196, Jul. 2024, https://doi.org/10.1038/s41598-024-68342-6

[13] T. Hosseinalizadeh, S. M. Salamati, S. A. Salamati, and G. B. Gharehpetian, "Improvement of Identification Procedure Using Hybrid Cuckoo Search Algorithm for Turbine-Governor and Excitation System," *IEEE Trans. Energy Convers.*, vol. 34, no. 2, pp. 585–593, Jun. 2019, https://doi.org/10.1109/TEC.2018.2868747

[14] A. Gálvez, I. Fister, S. Deb, I. Fister, and A. Iglesias, "Hybrid GA-PSO method withlocal search and image clustering for automatic IFS image reconstruction of fractal colored images," *Neural Comput. Appl.*, Nov. 2023, https://doi.org/10.1007/s00521-023-08954-7

[15] S. Biswas, K. Mandal, D. Pramanik, N. Roy, R. Biswas, and A.. Kuar, "Prediction and optimization of Nd: YAGlaser transmission micro-channelling on PMMA employing an artificial neural network model," *Infrared Phys. Technol.*, vol. 137, p. 105121, Mar. 2024, https://doi.org/10.1016/j.infrared.2024.105121

[16] A. I. Omar, Z. M. Ali, S. H. E. Abdel Aleem, E. E. A. El-Zahab, and A. M. Sharaf, "A dynamic switched compensation scheme for grid-connected wind energy systems using cuckoo search algorithm," *Int. J. Energy Convers.*, vol. 7, no. 2, pp. 64–74, 2019, https://doi.org/10.15866/irecon.v7i2.16895

[17] R. Sridhar, C. Subramani, and S. Pathy, "A grasshopper optimization algorithm aided maximum power point tracking for partially shaded photovoltaic systems," *Comput. Electr. Eng.*, vol. 92, p. 107124, Jun. 2021, https://doi.org/10.1016/j.compeleceng.2021.107124

[18] S. R. Salkuti, "Optimal Reactive Power Scheduling Using Cuckoo Search Algorithm," *Int. J. Electr. Comput. Eng.*, vol. 7, no. 5, p. 2349, Oct. 2017, https://doi.org/10.11591/ijece.v7i5.pp2349-2356

[19] A. A. Ghavifekr, A. Mohammadzadeh, and J. F. Ardashir, "Optimal Placement and Sizing of Energy-related Devices in Microgrids Using Grasshopper Optimization Algorithm," in *2021 12th Power Electronics, Drive Systems, and Technologies Conference (PEDSTC)*, IEEE, Feb. 2021, pp. 1–4. https://doi.org/10.1109/PEDSTC52094.2021.9405951

[20] Li Yancang, Cheng Fangmeng, and J. Suo, "Improved ACO inspired bylogistics and distribution problem," in *2010 2nd International Conference on Advanced Computer Control*, IEEE, 2010, pp. 369–371. https://doi.org/10.1109/ICACC.2010.5487086

[21] R. DIAF, C. TOLBA, and A. Nait Sidi Moh, “Traffic Urban Control Using an Intelligent PSO Algorithm Based on Integrated Approach,” *Alger. J. Signals Syst.*, vol. 5, no. 1, pp. 1–9, Mar. 2020, https://doi.org/10.51485/ajss.v5i1.89

[22] C. H. Ram Jethmalani, S. P. Simon, K. Sundareswaran, P. S. R. Nayak, and N. P. Padhy, “Auxiliary Hybrid PSO-BPNN-Based Transmission Systemloss Estimation in Generation Scheduling,” *IEEE Trans. Ind. Informatics*, vol. 13, no. 4, pp. 1692–1703, Aug. 2017, https://doi.org/10.1109/TII.2016.2614659

[23] P. C. S. Rao, P. K. Jana, and H. Banka, “A particle swarm optimization based energy efficient cluster head selection algorithm for wireless sensor networks,” *Wirel. Networks*, vol. 23, no. 7, pp. 2005–2020, Oct. 2017, https://doi.org/10.1007/s11276-016-1270-7

[24] A. K. Sangaiah, A. A. R. Hosseinabadi, M. B. Shareh, S. Y. B. Rad, A. Zolfagharian, and N. Chilamkurti, “IoT resource allocation and optimization based on heuristic algorithm,” *Sensors (Switzerland)*, vol. 20, no. 2, 2020, https://doi.org/10.3390/s20020539

[25] L. Xudong, l. Shuo, and F. Qingwu, “Prediction of Building Heating and Coolingload Based on IPSO-LSTM Neural Network,” in *2020 Chinese Automation Congress (CAC)*, IEEE, Nov. 2020, pp. 1085–1090. https://doi.org/10.1109/CAC51589.2020.9327849

[26] H. E. Mostafa, M. A. El-Sharkawy, A. A. Emary, and K. Yassin, “Design and allocation of power system stabilizers using the particle swarm optimization technique for an interconnected power system,” *Int. J. Electr. Power Energy Syst.*, vol. 34, no. 1, pp. 57–65, Jan. 2012, https://doi.org/10.1016/j.ijepes.2011.09.005

[27] M. H. Ibrahim, S. P. Ang, M. N. Dani, M. I. Rahman, R. Petra, and S. M. Sulthan, “Optimizing Step-Size of Perturb & Observe and Incremental Conductance MPPT Techniques Using PSO for Grid-Tied PV System,” *IEEE Access*, vol. 11, pp. 13079–13090, 2023, https://doi.org/10.1109/ACCESS.2023.3242979

[28] Q. S. Khalid *et al.*, “Hybrid Particle Swarm Algorithm for Products’ Scheduling Problem in Cellular Manufacturing System,” *Symmetry (Basel).*, vol. 11, no. 6, p. 729, May 2019, https://doi.org/10.3390/sym11060729

[29] G.-C. luh and C.-Y. lin, “Optimal design of truss-structures using particle swarm optimization,” *Comput. Struct.*, vol. 89, no. 23–24, pp. 2221–2232, Dec. 2011, https://doi.org/10.1016/j.compstruc.2011.08.013

[30] A. Agrawal, D. Garg, D. Popli, A. Banerjee, A. Raj, and I. Dikshit, “A review of spider monkey optimization: modification and its biomedical application,” *Int. J. Interact. Des. Manuf.*, Dec. 2023, https://doi.org/10.1007/s12008-023-01671-4

[31] B. Isong and O. Kgote, “Insights into Modern Intrusion Detection Strategies for Internet of Things Ecosystems,” 2024.

[32] Y. lan, Q. Chen, l. Zhang, and R. long, “Model Predictive Control Based On Spider monkey optimization Algorithm of Interleaved Parallel Bidirectional DC-DC Converter,” pp. 50–55, 2020.

[33] D. Tripathy, B. K. Sahu, N. B. D. Choudhury, and S. Dawn, “Spider monkey optimization based cascade controller forlFC of a hybrid power system.,” pp. 747–753, 2018.

[34] R. K. Sanapala, “An Optimized Energy Efficient Routing for Wireless Sensor Network using Improved Spider Monkey Optimization Algorithm,” vol. 15, no. 1, pp. 188–197, 2022, https://doi.org/10.22266/ijies2022.0228.18

[35] R. Alkanhel, A. A. Abdelhamid, A. Ibrahim, M. A. Alohali, M. Abotaleb, and D. S. Khafaga, “Metaheuristic Optimization,” 2023, https://doi.org/10.32604/cmc.2023.033273

[36] X. Zhang, Y. M. Xie, and S. Zhou, “A nodal-based evolutionary optimization algorithm for frame structures,” *Comput. Civ. Infrastruct. Eng.*, vol. 38, no. 3, pp. 288–306, 2023, https://doi.org/10.1111/mice.12834

[37] G. Nirmalapriya, V. Agalya, R. Regunathan, and M. Belsam Jeba Ananth, “Fractional Aquila spider monkey optimization based deeplearning network for classification of brain tumor,” *Biomed. Signal Process. Control*, vol. 79, p. 104017, Jan. 2023, https://doi.org/10.1016/j.bspc.2022.104017

[38] W. Sultana and S. D. S. Jebaseelan, “Optimal allocation of solar PV and wind energy power for radial distribution system using spider monkey optimization,” *Sustain. Comput. Informatics Syst.*, vol. 42, p. 100986, Apr. 2024, https://doi.org/10.1016/j.suscom.2024.100986

[39] N. Khare *et al.*, “SMO-DNN: Spider Monkey Optimization and Deep Neural Network Hybrid Classifier Model for Intrusion Detection,” *Electronics*, vol. 9, no. 4, p. 692, Apr. 2020, https://doi.org/10.3390/electronics9040692

[40] A. Kumar Kashyap and D. R. Parhi, “Multi-objective trajectory planning of humanoid robot using hybrid controller for multi-target problem in complex terrain,” *Expert Syst. Appl.*, vol. 179, p. 115110, Oct. 2021, https://doi.org/10.1016/j.eswa.2021.115110

[41] P. Vijayalakshmi and D. Karthika, “Hybrid dual-channel convolution neural network (DCCNN) with spider monkey optimization (SMO) for cyber security threats detection in internet of things,” *Meas. Sensors*, vol. 27, p. 100783, Jun. 2023, https://doi.org/10.1016/j.measen.2023.100783

[42] B. Sahu, A. Panigrahi, B. Dash, P. K. Sharma, and A. Pati, “A hybrid wrapper spider monkey optimization-simulated annealing model for optimal feature selection,” *Int. J. Reconfigurable Embed. Syst.*, vol. 12, no. 3, pp. 360–375, 2023, https://doi.org/10.11591/ijres.v12.i3.pp360-375

[43] M. Montalvo-Martel, A. Ochoa-Zezzatti, E. Carrum, and D. Barzaga, “Design of an Urban Transport Network for the Optimallocation of Bus Stops in a Smart City Based on a Big Data Model and Spider Monkey Optimization Algorithm,” 2021, pp. 167–201. https://doi.org/10.1007/978-3-030-68655-0_9

[44] S. Mirjalili, S. M. Mirjalili, and A. lewis, “Grey Wolf Optimizer,” *Adv. Eng. Softw.*, vol. 69, pp. 46–61, 2014, https://doi.org/10.1016/j.advengsoft.2013.12.007

[45] A. Yahiaoui, F. Fodhil, K. Benmansour, M. Tadjine, and N. Cheggaga, "Grey wolf optimizer for optimal design of hybrid renewable energy system PV-Diesel Generator-Battery: Application to the case of Djanet city of Algeria," *Sol. Energy*, vol. 158, pp. 941–951, Dec. 2017, https://doi.org/10.1016/j.solener.2017.10.040

[46] X. li and Y.-X. Guo, "The Grey Wolf Optimizer for Antenna Optimization Designs: Continuous, binary, single-objective, and multiobjective implementations," *IEEE Antennas Propag. Mag.*, vol. 64, no. 6, pp. 29–40, Dec. 2022, https://doi.org/10.1109/MAP.2021.3127798

[47] Q. Al-Tashi *et al.*, "Binary Multi-Objective Grey Wolf Optimizer for Feature Selection in Classification," *IEEE Access*, vol. 8, pp. 106247–106263, 2020, https://doi.org/10.1109/ACCESS.2020.3000040

[48] S. KILIÇARSLAN, "PSO + GWO: a hybrid particle swarm optimization and Grey Wolf optimization based Algorithm for fine-tuning hyper-parameters of convolutional neural networks for Cardiovascular Disease Detection," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 1, pp. 87–97, Jan. 2023, https://doi.org/10.1007/s12652-022-04433-4

[49] R. Debbarma* and D. C. Nandi*, "Maximum Power Point Tracking using Grey Wolf Technique Under Fast-Changing Irradiance," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 12, pp. 365–371, Sep. 2020, https://doi.org/10.35940/ijitee.K7838.0991120

[50] Z. Wang, Z. Jin, Z. Yang, W. Zhao, and M. Trik, "Increasing efficiency for routing in internet of things using Binary Gray Wolf Optimization and fuzzylogic," *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 35, no. 9, p. 101732, Oct. 2023, https://doi.org/10.1016/j.jksuci.2023.101732

[51] S. Yadav, S. K. Nagar, and A. Mishra, "Tuning of parameters of PID controller using Grey Wolf Optimizer," *SSRN Electron. J.*, 2020, https://doi.org/10.2139/ssrn.3575432

[52] I. I. Novendra, I. M. Wirawan, A. Kusumawardana, and A. K. latt, "Optimization ofload frequency control using grey wolf optimizer in micro hydro power plants," *J. Mechatronics, Electr. Power, Veh. Technol.*, vol. 14, no. 2, pp. 166–176, Dec. 2023, https://doi.org/10.14203/j.mev.2023.v14.166-176

[53] S. Kumar Chandar, "RETRACTED ARTICLE: Grey Wolf optimization-Elman neural network model for stock price prediction," *Soft Comput.*, vol. 25, no. 1, pp. 649–658, Jan. 2021, https://doi.org/10.1007/s00500-020-05174-2

[54] Y. Qiu, X. Yang, and S. Chen, "An improved gray wolf optimization algorithm solving to functional optimization and engineering design problems," *Sci. Rep.*, vol. 14, no. 1, p. 14190, Jun. 2024, https://doi.org/10.1038/s41598-024-64526-2

[55] A. Pawlowski, S. Romaniuk, Z. Kulesza, and M. Petrovic, "Trajectory optimization usinglearning from demonstration with meta-heuristic grey wolf algorithm," *IAES Int. J. Robot. Autom.*, vol. 11, no. 4, p. 263, Dec. 2022, https://doi.org/10.11591/ijra.v11i4.pp263-277

[56] K. Jaiswal and V. Anand, "A Grey-Wolf based Optimized Clustering approach to improve QoS in wireless sensor networks for IoT applications," *Peer-to-Peer Netw. Appl.*, vol. 14, no. 4, pp. 1943–1962, Jul. 2021, https://doi.org/10.1007/s12083-021-01099-1

[57] J. Y. An, Z. H. You, Y. Zhou, and D. F. Wang, "Sequence-based Prediction of Protein-Protein Interactions Using Gray Wolf Optimizer–Based Relevance Vector Machine," *Evol. Bioinforma.*, vol. 15, 2019, https://doi.org/10.1177/1176934319844522

[58] S. li and F. Wang, "Research on optimization of improved gray wolf optimization-extremelearning machine algorithm in vehicle route planning," *Discret. Dyn. Nat. Soc.*, vol. 2020, 2020, https://doi.org/10.1155/2020/8647820

[59] H. Alkhraisat, l. M. Dalbah, M. A. Al-Betar, M. A. Awadallah, K. Assaleh, and M. Deriche, "Size Optimization of Truss Structures Using Improved Grey Wolf Optimizer," *IEEE Access*, vol. 11, no. February, pp. 13383–13397, 2023, https://doi.org/10.1109/ACCESS.2023.3243164

[60] A. S. Mohammed and A. Dodo, "Load Frequency Control of One and Two Areas Power System Using Grasshopper Optimization Based Fractional Order PID Controller," *Control Syst. Optim. lett.*, vol. 1, no. 1, pp. 32–40, Apr. 2023, https://doi.org/10.59247/csol.v1i1.12

[61] T. Tamilarasan and M. V. Suganyadevi, "An improvement of Global Maximum Power Point Tracking Using a Novel Grasshopper Optimisation Algorithm of Photovoltaic System," *Iran. J. Sci. Technol. Trans. Electr. Eng.*, vol. 48, no. 2, pp. 929–943, Jun. 2024, https://doi.org/10.1007/s40998-024-00709-x

[62] A. Abdulrahman,.. Z. M., A. M. Zaki, F. H. H. Rizk, M. M. Eid, and E.-S. M. EL EL-Kenawy, "Exploring Optimization Algorithms: A Review of Methods and Applications," *J. Artif. Intell. Metaheuristics*, vol. 7, no. 2, pp. 08–17, 2024, https://doi.org/10.54216/JAIM.070201

[63] S. Kolli and B. R. Parvathala, "A Novel Assessment oflung Cancer Classification System Using Binary Grasshopper with Artificial Bee Optimisation Algorithm with Double Deep Neural Network Classifier," *J. Inst. Eng. Ser. B*, vol. 105, no. 5, pp. 1129–1143, Oct. 2024, https://doi.org/10.1007/s40031-024-01027-w

[64] J. Xia *et al.*, "Performance optimization of support vector machine with oppositional grasshopper optimization for acute appendicitis diagnosis," *Comput. Biol. Med.*, vol. 143, p. 105206, Apr. 2022, https://doi.org/10.1016/j.compbiomed.2021.105206

[65] X. Yue, H. Zhang, and H. Yu, "A Hybrid Grasshopper Optimization Algorithm With Invasive Weed for Global Optimization," *IEEE Access*, vol. 8, pp. 5928–5960, 2020, https://doi.org/10.1109/ACCESS.2019.2963679

[66] S. M. Haakonsen, S. H. Dyvik, M. luczkowski, and A. Rønnquist, "A Grasshopper Plugin for Finite Element Analysis with Solid Elements and Its Application on Gridshell Nodes," *Appl. Sci.*, vol. 12, no. 12, p. 6037, Jun. 2022, https://doi.org/10.3390/app12126037

[67] M. Sabouri and A. Sepidbar, “U-Net-based integrated framework for pavement crack detection and zone-based scoring,” *Int. J. Pavement Eng.*, vol. 25, no. 1, Dec. 2024, https://doi.org/10.1080/10298436.2024.2308183

[68] C. Waibel, l. Bystricky, A. Kubilay, R. Evins, and J. Carmeliet, “Validation of Grasshopper-based Fast Fluid Dynamics for Air Flow around Buildings in Early Design Stage,” Aug. 2017. https://doi.org/10.26868/25222708.2017.582

[69] T. Wortmann, “Model-based Optimization for Architectural Design: Optimizing Daylight and Glare in Grasshopper,” *Technol. + Des.*, vol. 1, no. 2, pp. 176–185, Nov. 2017, https://doi.org/10.1080/24751448.2017.1354615

[70] A. Maksoud, H. B. Al-Beer, A. A. Hussien, S. Dirar, E. Mushtaha, and M. W. Yahia, “Computational Design for Futuristic Environmentally Adaptive Building Forms and Structures,” *Archit. Eng.*, vol. 8, no. 1, pp. 13–24, 2023, https://doi.org/10.23968/2500-0055-2023-8-1-13-24

[71] M. Braik *et al.*, “Predicting Surface Ozonelevels in Eastern Croatia: leveraging Recurrent Fuzzy Neural Networks with Grasshopper Optimization Algorithm,” *Water, Air, Soil Pollut.*, vol. 235, no. 10, p. 655, Oct. 2024, https://doi.org/10.1007/s11270-024-07378-w

[72] S. Polepaka *et al.*, “Optimized convolutional neural network using grasshopper optimization technique for enhanced heart disease prediction,” *Cogent Eng.*, vol. 11, no. 1, p., 2024, https://doi.org/10.1080/23311916.2024.2423847

[73] S. Selvarajan, *A comprehensive study on modern optimization techniques for engineering applications*, vol. 57, no. 8. Springer Netherlands, 2024. https://doi.org/10.1007/s10462-024-10829-9

[74] B. Chithra and R. Nedunchezhian, “Dynamic neutrosophic cognitive map with improved cuckoo search algorithm (DNCM-ICSA) and ensemble classifier for rheumatoid arthritis (RA) disease,” *J. King Saud Univ. - Comput. Inf. Sci.*, vol. 34, no. 6, pp. 3236–3246, Jun. 2022, https://doi.org/10.1016/j.jksuci.2020.06.011

[75] S. Afanasyeva, J. Saari, O. Pyrhönen, and J. Partanen, “Cuckoo search for wind farm optimization with auxiliary infrastructure,” *Wind Energy*, vol. 21, no. 10, pp. 855–875, Oct. 2018, https://doi.org/10.1002/we.2199

[76] R. Zhang, X. Jiang, and R. li, “Improved decomposition-based multi-objective cuckoo search algorithm for spectrum allocation in cognitive vehicular network,” *Phys. Commun.*, vol. 34, pp. 301–309, Jun. 2019, https://doi.org/10.1016/j.phycom.2018.06.003

[77] S. Haghdoost, M. H. Niksokhan, M. G. Zamani, and M. R. Nikoo, “Optimal wasteload allocation in river systems based on a new multi-objective cuckoo optimization algorithm,” *Environ. Sci. Pollut. Res.*, vol. 30, no. 60, pp. 126116–126131, Nov. 2023, https://doi.org/10.1007/s11356-023-31058-7

[78] H. Xue, “Adaptive Cultural Algorithm-Based Cuckoo Search for Time-Dependent Vehicle Routing Problem with Stochastic Customers Using Adaptive Fractional Kalman Speed Prediction,” *Math. Probl. Eng.*, vol. 2020, 2020, https://doi.org/10.1155/2020/7258780

[79] S. Sengar and X. liu, “Optimal electricalload forecasting for hybrid renewable resources through a hybrid memetic cuckoo search approach,” *Soft Comput.*, vol. 24, no. 17, pp. 13099–13114, Sep. 2020, https://doi.org/10.1007/s00500-020-04727-9

[80] A. Zadeh Shirazi, M. Hatami, M. Yaghoobi, and S. J. Seyyed Mahdavi Chabok, “An Intelligent Approach to Predict Vibration Rate in a Real Gas Turbine,” *Intell. Ind. Syst.*, vol. 2, no. 3, pp. 253–267, 2016, https://doi.org/10.1007/s40903-016-0057-6

[81] F. Ahmadkhanlou and H. Adeli, “Optimum cost design of reinforced concrete slabs using neural dynamics model,” *Eng. Appl. Artif. Intell.*, vol. 18, no. 1, pp. 65–72, 2005, https://doi.org/10.1016/j.engappai.2004.08.025

[82] A. Kumar, S. S. Satyanarayana Reddy, G. B. Mahommad, B. Khan, and R. Sharma, “Smart Healthcare: Disease Prediction Using the Cuckoo-Enabled Deep Classifier in IoT Framework,” *Sci. Program.*, vol. 2022, pp. 1–11, May 2022, https://doi.org/10.1155/2022/2090681

[83] R. Salgotra, N. Mittal, A. S. Almazyad, and A. W. Mohamed, “RGN: A Triple Hybrid Algorithm for Multi-level Image Segmentation with Type II Fuzzy Sets,” *Ain Shams Eng. J.*, vol. 15, no. 11, p. 102997, Nov. 2024, https://doi.org/10.1016/j.asej.2024.102997

[84] R. Salgotra and S. Mirjalili, “Multi-algorithm based evolutionary strategy with Adaptive Mutation Mechanism for Constraint Engineering Design Problems,” *Expert Syst. Appl.*, vol. 258, p. 125055, Dec. 2024, https://doi.org/10.1016/j.eswa.2024.125055

[85] M. Guerrero, O. Castillo, and M. García, “Cuckoo Search vialévy Flights and a Comparison with Genetic Algorithms,” 2015, pp. 91–103. https://doi.org/10.1007/978-3-319-10960-2_6

[86] A. R. Yildiz, “Cuckoo search algorithm for the selection of optimal machining parameters in milling operations,” *Int. J. Adv. Manuf. Technol.*, vol. 64, no. 1–4, pp. 55–61, Jan. 2013, https://doi.org/10.1007/s00170-012-4013-7

[87] R. Salgotra, U. Singh, S. Saha, and A. H. Gandomi, “Self adaptive cuckoo search: Analysis and experimentation,” *Swarm Evol. Comput.*, vol. 60, p. 100751, Feb. 2021, https://doi.org/10.1016/j.swevo.2020.100751

[88] H. Tang, X. li, l. Meng, Z. Zhang, and S. Chen, “Process modeling and optimization inlaser drilling of bulk metallic glasses based on GABPNN and machine vision,” *Opt.laser Technol.*, vol. 172, p. 110502, May 2024, https://doi.org/10.1016/j.optlastec.2023.110502

[89] B. S. Anukeerthana, D. S. lavanya, V. Gurucharran, and R. Madhumathi, “Improving Agricultural Productivity and Water Usage Through Advanced ACO Technique,” in *2024 10th International Conference on Communication and Signal Processing (ICCSP)*, IEEE, Apr. 2024, pp. 93–97. https://doi.org/10.1109/ICCSP60870.2024.10544083

[90] S. Ç. Öztürk and E. Ö. A. Aktan, "A Cultural Route Recommendation Based on Optimization Techniques in Urban Spaces," *Int. J. Sustain. Dev. Plan.*, vol. 19, no. 9, pp. 3417–3430, 2024, https://doi.org/10.18280/ijsdp.190912

[91] S. Kashef and H. Nezamabadi-pour, "An advanced ACO algorithm for feature subset selection," *Neurocomputing*, vol. 147, pp. 271–279, Jan. 2015 https://doi.org/10.1016/j.neucom.2014.06.067

[92] T. Islam, M. E. Islam, and M. R. Ruhin, "An Analysis of Foraging and Echolocation Behavior of Swarm Intelligence Algorithms in Optimization: ACO, BCO and BA," *Int. J. Intell. Sci.*, vol. 08, no. 01, pp. 1–27, 2018, https://doi.org/10.4236/ijis.2018.81001

[93] K. Jairam Naik, "A Dynamic ACO-Based Elasticload Balancer for Cloud Computing (D-ACOELB)," 2020, pp. 11–20. https://doi.org/10.1007/978-981-15-1097-7_2

[94] J. Wu and H. Zou, "Harmonic detection technology based on ant colony optimization BP neural network," *J. Phys. Conf. Ser.*, vol. 2221, no. 1, p. 012058, May 2022, https://doi.org/10.1088/1742-6596/2221/1/012058

[95] A. W. Wijayanto, S. Mariyah, and A. Purwarianti, "Enhancing clustering quality of fuzzy geographically weighted clustering using Ant Colony optimization," *Proc. 2017 Int. Conf. Data Softw. Eng. ICoDSE 2017*, vol. 2018-January, pp. 1–6, 2017, https://doi.org/10.1109/ICODSE.2017.8285858

[96] Z. Wan, Y. Guo, J. Yang, X. Wang, and J. li, "Logistics Routing Intelligence based on Improved Ant Colony Algorithm and Dijkstra Algorithm," *Front. Sci. Eng.*, vol. 4, no. 8, pp. 130–142, Aug. 2024, https://doi.org/10.54691/669kn656

[97] Y. Gajpal and P. l. Abad, "Multi-ant colony system (MACS) for a vehicle routing problem with backhauls," *Eur. J. Oper. Res.*, vol. 196, no. 1, pp. 102–117, Jul. 2009, https://doi.org/10.1016/j.ejor.2008.02.025

[98] B. Saeidian, M. S. Mesgari, B. Pradhan, and M. Ghodousi, "Optimizedlocation-Allocation of Earthquake Relief Centers Using PSO and ACO, Complemented by GIS, Clustering, and TOPSIS," *ISPRS Int. J. Geo-Information*, vol. 7, no. 8, p. 292, Jul. 2018, https://doi.org/10.3390/ijgi7080292

[99] Y. Gao, J. Wang, and C. li, "Escape afterlove: Philoponella prominens optimizer and its application to 3D path planning," *Cluster Comput.*, vol. 28, no. 2, p. 81, Apr. 2025, https://doi.org/10.1007/s10586-024-04761-4

[100] Sukono *et al.*, "The effect of gross domestic product and population growth on CO2 emissions in Indonesia: An application of the ant colony optimisation algorithm and cobb-douglas model," *Int. J. Energy Econ. Policy*, vol. 9, no. 4, pp. 313–319, 2019, https://doi.org/10.32479/ijeep.8011

[101] M. R. Jabbarpour, H. Malakooti, R. M. Noor, N. B. Anuar, and N. Khamis, "Ant colony optimisation for vehicle traffic systems: applications and challenges," *Int. J. Bio-Inspired Comput.*, vol. 6, no. 1, p. 32, 2014, https://doi.org/10.1504/IJBIC.2014.059970

[102] A. Tahir *et al.*, "Hybrid HP-BOA: An Optimized Framework for Reliable Storage of Cloud Data Using Hybrid Meta-Heuristic Algorithm," *Appl. Sci.*, vol. 13, no. 9, p. 5346, Apr. 2023, https://doi.org/10.3390/app13095346