



A comprehensive survey on the chicken swarm optimization algorithm and its applications: state-of-the-art and research challenges

Binhe Chen¹ · Li Cao¹ · Changzu Chen¹ · Yaodan Chen¹ · Yinggao Yue^{1,2}

Accepted: 5 May 2024 / Published online: 11 June 2024
© The Author(s) 2024

Abstract

The application of optimization theory and the algorithms that are generated from it has increased along with science and technology's continued advancement. Numerous issues in daily life can be categorized as combinatorial optimization issues. Swarm intelligence optimization algorithms have been successful in machine learning, process control, and engineering prediction throughout the years and have been shown to be efficient in handling combinatorial optimization issues. An intelligent optimization system called the chicken swarm optimization algorithm (CSO) mimics the organic behavior of flocks of chickens. In the benchmark problem's optimization process as the objective function, it outperforms several popular intelligent optimization methods like PSO. The concept and advancement of the flock optimization algorithm, the comparison with other meta-heuristic algorithms, and the development trend are reviewed in order to further enhance the search performance of the algorithm and quicken the research and application process of the algorithm. The fundamental algorithm model is first described, and the enhanced chicken swarm optimization algorithm based on algorithm parameters, chaos and quantum optimization, learning strategy, and population diversity is then categorized and summarized using both domestic and international literature. The use of group optimization algorithms in the areas of feature extraction, image processing, robotic engineering, wireless sensor networks, and power. Second, it is evaluated in terms of benefits, drawbacks, and application in comparison to other meta-heuristic algorithms. Finally, the direction of flock optimization algorithm research and development is anticipated.

Keywords Swarm intelligence · Evolutionary computation · Bio-inspired algorithm · Chicken swarm optimization algorithm

Binhe Chen and Li Cao are contributed equally to this work and considered as co-first authors.

✉ Yinggao Yue
yueyinggao2006@163.com

¹ School of Intelligent Manufacturing and Electronic Engineering, Wenzhou University of Technology, Wenzhou 325035, China

² Intelligent Information Systems Institute, Wenzhou University, Wenzhou 325035, China

1 Introduction

The majority of real-world engineering applications frequently center on optimization challenges in the domains of computer, electrical, Internet of Things, robotics, and other engineering (Li and Zhou 2023). The term "optimization problem" refers to the design of an optimization algorithm to determine the combination of schemes or parameters under the given restriction conditions that causes the performance index to reach the highest value among all of the schemes or parameters, using the specific performance index of the entire project as the optimization object (Singh and Kumar 2023; Jiang et al. 2024). Traditional optimization algorithms such as the gradient descent method (Rostami et al. 2021), conjugate gradient method, Newton method, Lagrange multiplier method, etc. must traverse the entire parameter space while being processed (Cao et al. 2023). As a result, they frequently take a long time to find due to the difficulty of the traversal and the size of the parameter space. The failure of the solution might even be caused by the ideal solution (Cao et al. 2022). Therefore, finding accurate and effective optimization algorithms has become one of the primary study themes of associated disciplines due to the increasingly complicated practical engineering application challenges (Bai et al. 2023).

Numerous academics have suggested and begun research on intelligent optimization algorithms for the optimization of complicated engineering applications, drawing inspiration from human intelligence, biological group dynamics, and natural laws (Tang et al. 2024; Wang et al. 2022a). By modeling certain natural ecosystem systems, intelligent optimization algorithms may handle difficult optimization issues (Chen et al. 2024). According to their functional properties, these algorithms are classified as general optimization algorithms, evolutionary algorithms, and swarm intelligence optimization algorithms (Yue et al. 2022). The general optimization algorithm is based on the greedy algorithm, which always searches for the optimal solution in the local range of the current position when solving the problem (Yue et al. 2021), and the typical representative is the hill-climbing algorithm. However, the algorithm is easy to fall into the local optimal solution in the process of solving, so it cannot obtain the global optimal solution. There are two fixes for this issue. Setting a probability parameter comes first. It will hop out of the present place with a fixed probability and solve it again once the entire solution space has reached a local optimum solution. Simulated annealing algorithm is the model algorithm. The second is to emulate human memory, create a taboo list to record the local optimum solution, avoid the local optimal solution in the list in subsequent searches, and alter the search range to be global. The taboo search algorithm (Wang et al. 2024; Yue et al. 2023) is a good example of an algorithm.

The evolutionary algorithm simulates a number of biological evolution processes, such as genetics, selection, and mutation, as a parameter replacement process. It then calculates the value of the corresponding solution after each replacement to arrive at the best possible solution (Bai et al. 2022). Evolutionary algorithms can be categorized into genetic algorithms, cooperative evolutionary algorithms (CCEAS) (Cai et al. 2021a), immune evolutionary algorithms (IA) (Li et al. 2021a), differential evolutionary algorithms (DE) (Deng et al. 2021), multi-objective evolutionary algorithms (MOP) (Falcón-Cardona et al. 2021), and others based on the various replacement processes. Through the use of bionics, the swarm intelligence optimization method resolves challenging optimization issues (Tawhid and Ibrahim 2023). The fundamental idea behind this kind of algorithm is to treat the optimization problem's parameters as individuals within a group, update each individual parameter sequentially by simulating animal group behavior, set the fitness value to assess

the updated parameters, and then output all parameters that correspond to the optimal solution in the solution (Jaafari et al. 2022). Representatives of swarm intelligence optimization algorithms include the Ant Colony Optimization Algorithm (ACO) (Maheshwari et al. 2021) that simulates the foraging behavior of ants, the Particle Swarm Optimization Algorithm (PSO) (Yue et al. 2024) that simulates the foraging behavior of birds, and the Shuffled Frog Leaping Algorithm (SFLA) (Maarroof et al. 2022) that simulates the foraging process of frogs, artificial bee colony optimization algorithm (ABC) (Kaya et al. 2022) for simulating the foraging behavior of bee colonies, wolf search algorithm (WSA) (Devika et al. 2021) for simulating the predation behavior of wolves, dragonfly optimization algorithm (DA) (Aghelpour et al. 2021) for simulating the predation behavior of dragonflies, etc. The swarm intelligence optimization algorithms listed are shown in Table 1.

The self-organization capability of swarm intelligence algorithms is the primary cause of the new bionic swarm intelligence algorithms' recent rapid growth trend, as can be seen from the table (Jaafari et al. 2022). Self-organization capability indicates that each parameter in the algorithm operates independently and is only changed and altered in accordance with a predetermined set of rules (Rosado-Olivieri and Brivanlou 2021). The intelligent optimization of the whole algorithm population finally reflects this "decentralized" parameter modification process.

Meng (46), who first introduced the Chicken Swarm Optimization algorithm (CSO), describes CSO as an intelligent optimization algorithm that plans the parameter change process in accordance with the behavior of the flock. This algorithm simulates the population behavior of chickens, divides the parameter variables into three groups of hens, roosters, and chicks, sets the parameter self-organization transformation process in accordance with the rigid foraging hierarchy of the chicken flock (Ishikawa et al. 2020), integrates the optimization process of the optimization problem, etc. The result can be compared to how flocks of chickens forage. Since it was first developed, the chicken swarm optimization technique has received a lot of interest, and several academics have investigated it from various perspectives. From the viewpoint of algorithm convergence, In the literature (Nagarajan 2023), the author demonstrated the Markov chain theory-based chicken swarm optimization algorithm's global convergence. Researchers, including the algorithm's original creator, evaluate and compare the benefits and drawbacks of the chicken swarm optimization algorithm and other intelligent optimization algorithms using a number of benchmark functions for optimization problems. They then conduct simulation experiments to determine the algorithm's solution speed and accuracy. It outperforms the differential evolution algorithm, bat algorithm, and particle swarm algorithm in terms of resilience and robustness (Saif et al. 2023). Some academics have enhanced the chicken swarm optimization technique and used it to solve real-world engineering challenges in addition to providing evidence and analysis. By examining the drawbacks of mindlessly following the hen's parameters in the chicken flock and readily slipping into local optimal solutions, the author of the literature (Wang et al. 2023) developed a better CSO method based on Gaussian migration strategy. In the literature (Nuvvula et al. 2022), the author used the differential improved CSO algorithm for the smart grid's energy management and solved the algorithm's optimal solution to find a way to reduce power consumption and cost. In the literature (Yu et al. 2022), the author utilized the CSO algorithm to extract features, and via comparison studies with various optimization algorithms, confirmed that the flock optimization approach has the advantages of rapid speed and high accuracy in feature extraction application situations. The demands for the accuracy and speed of optimization problem solutions have been increasing in recent years due to engineering applications such as data mining (Gordan et al. 2022), wireless sensor networks (Majid et al. 2022), robotics

Table 1 Swarm intelligence optimization algorithm

Algorithms	Acronym	Bio-Objects	Algorithms	Acronym	Bio-Objects
Ant Colony Optimization	ASO	Ant	Particle Swarm Optimization	PSO	Bird
Bacterial Foraging Optimization (Guo et al. 2021)	BFO	Bacterium	Fish-School Search (Demidova and Gorchakov 2021)	FFS	Fish-School
Artificial Fish Swarms Algorithm (Pourpanah et al. 2023)	AFSA	Fish	Biogeography-based optimization (Li and Wang 2022)	BBO	Biogeography
Shuffled Frog Leaping Algorithm	SFLA	Frog	Group Search Optimization (Abualigah 2021)	GSO	Biological search
Firefly Algorithm (Kumar and Kumar 2021)	FA	Firefly	Cockroach Swarm Optimization (El-dosuky and Shams 2022)	CSO-II	Cockroach
Cat Swarm Optimization (Seyyedabbasi 2023)	CSO-I	Catamount	Cuckoo Search Optimization (Cuong-Le et al. 2021)	CSO-III	Cuckoo
Invasive Weed Optimization (Garlapati et al. 2022)	IWO	Weed	Bat Algorithm	BA	Bat
Artificial BeeColony Algorithm	ABC	Bee	Termite colony optimization (Joseph and Asaletha 2023)	TCO	Termite
Fruit Fly Optimization Algorithm (Sutradhar et al. 2023)	FOA	Fruit Fly	Spotted Hyena Optimizer (Gafar et al. 2024)	SHO	Spotted Hyena
Wolf Search Algorithm	WSA	Wolf	Polar Bear Optimization (Tian et al. 2022)	PBO	Polar bear
Swallow Swarm Optimization (Slezkin et al. 2021)	SSO-I	Swallow	Grasshopper Optimization Algorithm (Meraihi et al. 2021)	GOA	Grasshopper
Chicken Swarm Optimization (Meng et al. 2014)	CSO	Chicken	Mouth Brooding Fish Algorithm (Babazadeh et al. 2023)	MBFA	Mouth Brooding Fish
Grey Wolf Optimizer (Aguila-Leon et al. 2023)	GWO	Grey Wolf	Eagle Perching Optimizer (Khan et al. 2023)	EPO	Eagle
Spider Monkey Optimization (Kumar et al. 2023)	SMO	Spider Monkey	Squirrel Search Algorithm (Maden et al. 2023)	SSA-II	Squirrel
Moth-Flame Optimization (Shehab et al. 2020)	MFO	Moth	Harris Hawks Optimization (Gharehchopogh 2023)	HHO	Harris Hawks
Dragonfly Algorithm	DA	Dragonfly	Seagull Optimization Algorithm (Nabavi et al. 2023)	SOA	Seagull
Shark Smell Optimization (Ahamad et al. 2022)	SSO-II	Shark	Sooty Tern Optimization Algorithm (Singh et al. 2022)	STOA	Sooty Tern
Whale Optimization Algorithm (Chakraborty et al. 2021)	WOA	Whale	Deer Hunting Optimization (Duan et al. 2023)	DHOA	Hunting
Crow Search Algorithm (Hussien et al. 2020)	CSA	Crow	Mayfly Optimization Algorithm (Zervoudakis and Tsafarakis 2020)	MOA	Mayfly

Table 1 (continued)

Algorithms	Algorithms	Acronym	Bio-Objects	Algorithms	Acronym	Bio-Objects
Salp Swarm Algorithm (Abualigah et al. 2020)		SSA-I	Salp	Black Widow Optimization Algorithm (Hayyolalam and Kazem 2020)	BWOA	Black Widow
Rat Swarm Optimization (Dhiman et al. 2021)		RSO	Rat	Honey Badger Algorithm (Hashim et al. 2022)	HBA	Honey Badger
Capuchin Search Algorithm (Braik et al. 2021)		CapSA	Capuchin	Sand Cat Swarm Optimization (Seyyedabbasi 2023)	SCSO	Sand Cat
Tuna Swarm Optimization (Xie et al. 2021)		TSO	Tuna	Pelican Optimization Algorithm (Trojovský and Delghani 2022)	POA	Pelican
Wild Horse Optimizer (Naruei and Keynia 2022)		WHO	Wild Horse	White Shark Optimizer (Braik et al. 2022)	WSO	White Shark

engineering (Macenski et al. 2022), electric power (Silvestre et al. 2021), feature extraction (Zhang et al. 2022), etc. Numerous academics have presented a number of fresh use cases and implementation strategies for chicken swarm optimization algorithms through constant innovation. Therefore, the review research in this paper will concentrate on the use of the most recent chicken swarm optimization algorithms in particular domains and look ahead to the direction in which optimization algorithms will develop in the future.

The organizational structure of this paper is as follows: Section 2 introduces the biological characteristics of the CSO algorithm, the CSO algorithm and its advantages and disadvantages. Section 3 introduces an improved algorithm for the CSO algorithm. Section 4 introduces the application of CSO algorithm in different fields. Section 5 discusses the above and looks forward to the future research of CSO algorithm. Section 6 Conclusions the paper.

2 Chicken swarm optimization

2.1 Chicken characteristics

The gregarious nature of chickens makes them a particular breed of poultry animal, and they frequently coordinate their food-finding efforts in groups (Yan et al. 2021). In chicken flocks, there are three different categories of people: hens, chicks, and roosters. There is a clear foraging hierarchy in the group, according to varied foraging capacities (Carvalho et al. 2022). Due to their inferior foraging abilities compared to roosters, hens in this hierarchy forage after them, while chicks will do the same because they have worse foraging abilities (Basha et al. 2023). Figure 1 depicts the population structure of chickens, showing that the rooster occupies the population's center, the hens are arranged around the rooster, and the chicks are positioned around the hen. As a result, there is a natural pattern of mutual learning and rivalry between individuals of the same species, such as roosters and roosters, hens and hens, or between members of different species, such as hens and chicks, throughout the process of foraging (Jonsson and Vahlne 2023). For example, hen groups H_1 , H_2 will forage around rooster R_2 and learn the foraging patterns of rooster R_2 ,

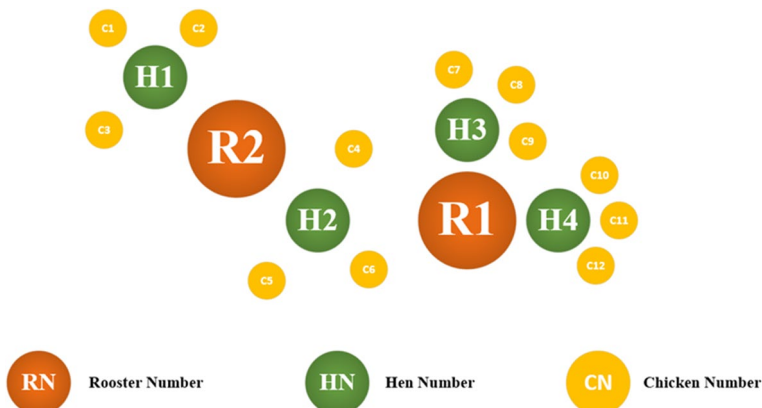


Fig. 1 Schematic diagram of chicken population structure

which will determine the foraging trajectory of hens H_1, H_2 . At the same time, as hen H_2 is close to rooster R_1 , the foraging pattern of rooster R_1 will also influence hen H_2 to some extent. Chicks C_4, C_5 and C_6 will forage around hen H_2 , learning her foraging pattern, and hen H_2 will determine the foraging trajectory of chicks C_4, C_5 and C_6 . The developers of the method were motivated by the fact that this coexistence of learning and competition is consistent with the self-organizing growth of group intelligence and so proposed chicken swarm optimization (CSO).

2.2 Algorithm model

The objective function that demands an optimal solution is the optimization object of the intelligent optimization algorithm, and its independent variable parameters can be made up of n j -dimensional space vectors X , where n denotes the number and j denotes the dimensionality, and n is any positive integer. By the quantity of vectors X , the Chicken Optimization Algorithm is classified into three categories. The fitness value f of each individual is what distinguishes the rooster, hen, and chick flocks. In Eq. (1), the rooster group R_i is assigned to the first RN individuals with the lowest fitness value; in Eq. (2), the chick group C_i is assigned to the CN individuals with the highest fitness value; and in Eq. (3), the remaining HN individuals are assigned to the hen group H_i . Consequently, RN, HN , and CN represent the corresponding numbers of individuals in each group within the colony: RN represents the rooster group, HN represents the hen group, and CN represents the chick group.

$$R_i = \{R_1, R_2, \dots, R_{RN}\} \tag{1}$$

$$C_i = \{C_1, C_2, \dots, C_{CN}\} \tag{2}$$

$$H_i = \{H_1, H_2, \dots, H_{HN}\} \tag{3}$$

Every hen in a flock has a matching individual dominant male, and every chick has a corresponding individual mother hen. The following formulas are used to update the foraging location of roosters, hens, and individual chicks (Wang et al. 2023b):

1) Calculation formula for Location Succession of rooster groups

$$R_{i,j}^{t+1} = R_{i,j}^t [1 + randn(0, \delta^2)] \tag{4}$$

$$\delta^2 = \begin{cases} 1, & f_i \leq f_s \\ e^{\frac{f_s - f_i}{|f_i| + \epsilon}}, & f_i > f_s \\ s \in [1, n], & s \neq i \end{cases} \tag{5}$$

wherein, in Eq. (4), $R_{i,j}^t$ is the position of the i -th rooster in the j -th dimension after t iterations, $0, \delta^2$ is a Gaussian distributed random number obeying a mean of 0 and a variance of δ^2 . Wherein, in Eq. (5), The individual's fitness value is represented by f , and the random rooster index, or S , is a little but important integer that keeps the denominator from being zero.

2) Calculation formula for Location Succession of the hen groups

$$H_{ij}^{t+1} = H_{ij}^t + k_1 * rand * (R_{Hi}^t - M_{ij}^t) + k_2 * rand * (RH^t - H_{ij}^t) \tag{6}$$

$$k_1 = e^{\frac{f_{RH} - f_{rHi}}{|V_{Hi}| + \epsilon}} \tag{7}$$

$$k_2 = e^{f_{rHi} - f_{RH}} \tag{8}$$

wherein, in Eq. (6), H_{ij}^t is the position of the i -th hen in the j -th dimension after t iterations. $rand$ is a random number between [0,1]. R_{Hi}^t is the position of the leader rooster of the i -th hen after t iterations; RH^t is the position after t iterations of individuals randomly selected among the other roosters and hens except the leader cock and hen itself; The rooster’s influence factor is represented by k_1 , and the random individual effect factor is shown by k_2 . where the fitness value of the i -th hen is represented by f_{Hi} in Eq. (7). The fitness value of the rooster that leads the hen is f_{rHi} , where f_{RH} , the fitness value of a random person, appears in Eq. (8).

3) Calculation formula for Location Succession of the hen groups

$$C_{ij}^{t+1} = C_{ij}^t + F * (H_{ij}^t - C_{ij}^t) \tag{9}$$

wherein, in Eq. (9), C_{ij}^t is the position of the i -th chick in the j -th dimension after t iterations; H_{ij}^t is the position of the corresponding hen followed by the i -th chick after t iterations; F is a random number of (0,2).

2.3 Algorithm steps

Assume that the objective function of the optimization problem is $F(\vec{X})$, where \vec{X} is composed of n m -dimensional space vectors, n represents the number, m represents the dimension. The particular implementation phases of the chicken swarm optimization algorithm are detailed below (Ayvaz 2022) based on the composition of the algorithm operators already defined:

- 1) Set the chicken flock to contain N individuals, corresponding to \vec{X} in the objective function. Assign the numbers CN, HN, RN, MN to the hens, roosters, chicks, and mother chicks, accordingly. Establish the number of position dimensions (j), the number of rearrangement indicators (G), the maximum number of iterations (Max_G), and the initial number of iterations ($t=0$) for the method. Go to step 2).
- 2) Check to determine if $t < Max_G$ is true; if so, proceed to step 3; if not, break out of the loop and report the best answer.
- 3) Calculate the fitness value f_i of each individual in the chicken population, go to step 4).
- 4) Determine whether Eq. (10) is true, if true, go to step 5), if not, go to step 6).

$$t \bmod G == 0 \tag{10}$$

- 5) Establish the corresponding connection and define individuals in accordance with the order of fitness values, where the rooster has the lowest fitness value, the chick has the highest, and the hen is in the middle. Go to step 6).

- 6) All chicken groups start foraging, update the position coordinates, and calculate the objective function value F , and record the latest optimal solution once a better function value F_{best} appears. Execute $t = t + 1$ after updating, go to step 2).

The flow chart of chicken flock optimization algorithm is shown in Fig. 2.

2.4 Algorithm parameter setting

The Chicken Swarm optimization technique uses the following parameters: N , RN , HN , CN , MN , Max_G , t , j , and G . Max_G , N , and j are the optimization problem's general settings, while N , j and Max_G stand for the parameters' number and dimension, respectively. These three parameters are often chosen by the optimization problem's function to be solved. Problems demanding high solution accuracy typically specify a bigger Max_G , while complicated solution functions typically have larger N and j . The letters RN , HN , CN , and MN , respectively, stand for the number of roosters, hens, chicks, and mother hens (Kumar and Pandey 2022). Among them, it has been demonstrated that HN represents more hens than RN , and the algorithm's performance is good since HN will produce more chicks. The explanation is that hens are better at stabilizing the parameters of chicks. The chicken group optimization algorithm's parameter G is the population reorganization reference index. The population will increase when the requirements of algorithm step 4 are

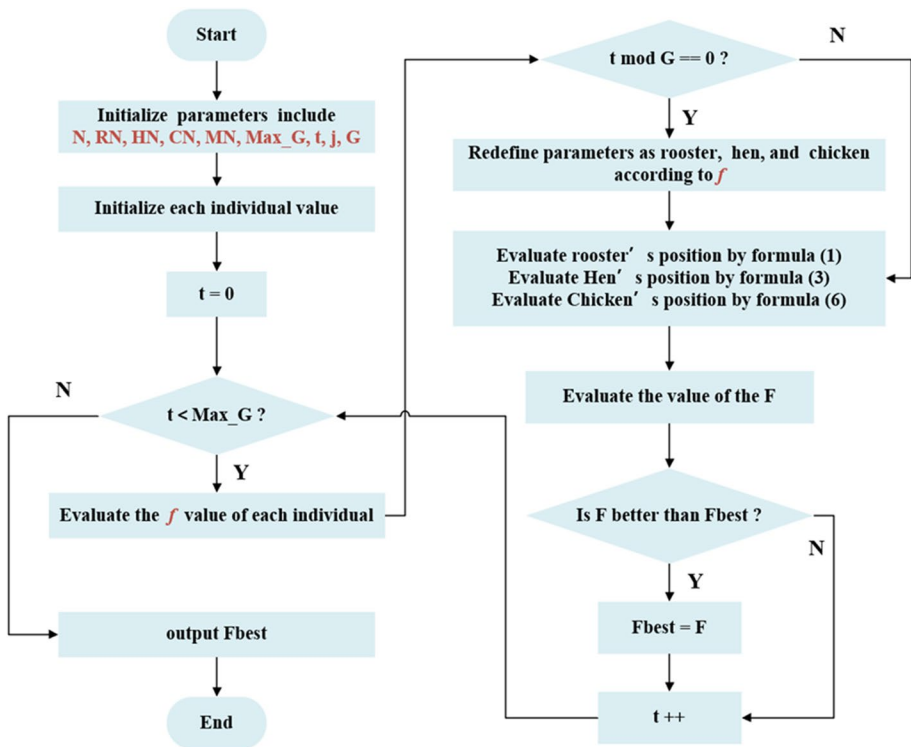


Fig. 2 Flowchart of flock optimization algorithm

satisfied, and a special performance is that certain chicks' fitness value will decline as they mature into roosters or hens. Some roosters and hens get more fit, which causes them to pass away and be classified as a new chick. The frequency of the changing process depends on the magnitude of the parameter G . The frequency of modifications will be high and the algorithm's effectiveness will decline if the parameter G is set to a value that is too great. The changes won't occur frequently enough if the parameter G is set too low, and the algorithm will quickly settle into a local optimum solution (Gu et al. 2022). After experimental validation, it is advisable to restrict the range of parameter G to Singh and Kumar (2023); Tawhid and Ibrahim 2023).

The parameter setting of CSO algorithm can also reflect that the algorithm has a certain degree of portability (Deng et al. 2022). Set the CSO parameters RN and CN to 0, and the CSO algorithm can be transformed into the DE algorithm. Set the RN and CN of the CSO algorithm parameters to 0, set the k_1 and k_2 parameters to the C_1 and C_2 parameters in the PSO, and the chicken group optimization algorithm can be converted into the PSO algorithm.

2.5 Analysis of CSO Advantages

The accuracy and computing efficiency of the swarm intelligence optimization method determine its benefits and drawbacks (Nadikattu 2021). The computational effectiveness of the method depends on its complexity, and the accuracy of the algorithm depends on its traversal range, provided that the number of iteration cycles Max_G is fixed (Sabale and Mini 2021). The CSO method has a straightforward procedure and excellent computing efficiency, and because it splits the parameters into three different population types, the traversal range of the algorithm will also expand as a result of the various search ranges of various populations (Ding et al. 2020). As a result, the CSO method offers a lot of advantages in terms of computing speed and accuracy (Mansouri et al. 2021). Many academics analyze the benefits and drawbacks of the CSO algorithm and other varieties of swarm intelligence optimization algorithms using simulated exercises.

Meng (Meng et al. 2014) conducted performance comparison analyses of the CSO, Particle Swarm Optimization (PSO), Bat Algorithm (BA), and Differential Evolution (DE) algorithms using 12 typical benchmark functions for algorithm testing. The comparison of simulation experiment results reveals that CSO offers greater benefits in terms of accuracy, algorithm efficiency, and resilience. Griewank et al. conducted a performance comparison analysis of the CSO, BA, and Bat Algorithm based on Differential Evolution (DEBA) in the literature (Fu et al. 2019), where the author employed three test functions. The simulation findings demonstrate that CSO performs significantly better. The author conducted a comparative performance analysis of CSO, PSO, and BA using the literature (Kadhuim and Al-Janabi 2023). According to the simulation findings, CSO greatly outperforms PSO and BA in terms of optimizing results. There is a premature occurrence in optimization issues that is simple to fit into a local optimum solution. Table 2 displays the comparative outcomes of several methods. It is clear from the comparative analysis findings in the table that the CSO algorithm performs better and is more stable during the process of finding the best solution than other swarm intelligence algorithms. and increased flexibility (Deb et al. 2020). The CSO algorithm also has the issue that the calculation accuracy is insufficient and that it is simple to slip into a local optimum solution in the latter stage of convergence (Chen et al. 2023).

Table 2 Comparison of CSO and other algorithms

Algorithm	Advantage	Disadvantage	Application
PSO	fast algorithm convergence	low convergence accuracy	multi-objective optimization problem
BA	Computational efficiency	Premature convergence when dealing with high-dimensional problems	Nonlinear Optimization Problems
DE	High precision and stable results	low diversity	Multi-objective function optimal solution
DEBA	Strong global search ability	Individuals are prone to local optima	Problems with Nonlinear Equations
CSO	Stable convergence results and high accuracy	A local optimal solution appears when the period is long	High Dimensional Optimization Problems

2.6 Analysis of CSO Disadvantages

The CSO algorithm's drawback is that it is simple to settle on a local optimum solution. The operator entering a local optimum and the incorrect parameter setting are to blame (Qi et al. 2022). When parameters are specified incorrectly, the algorithm cannot forecast the ideal setting in advance, leading to the selection of subpar parameters, which ultimately produces subpar output results. The operator is a local optimum solution candidate for the following reasons. The individual hen of CSO will always follow the rooster to perform parameter replacement because the rooster plays a crucial role in updating the position of the hen. As a result, once the rooster operator falls into the local optimal solution, the hen operator will also quickly fall into a local optimal solution, resulting in rapid convergence of the algorithm (Ren and Long 2021). The hen operator will fall into the local optimal solution because the parameter replacement of the chicken operator only depends on the hen operator, and the chicken operator will quickly fall into the local optimal solution as a result, accelerating the convergence of the algorithm based on the previous, creating a vicious circle.

3 Improvement of chicken swarm optimization algorithms

When compared to conventional swarm intelligence algorithms, the CSO has shown improved efficiency and resilience. This is explained by its streamlined algorithmic architecture, built-in stability, and exceptional portability (Singh et al. 2023). However, crucial factors like the mother hen population and recombination coefficient substantially influence the convergence of CSO (Wu 2021). A poor design of these parameters can slow convergence and diminish the algorithm's search accuracy by reducing population diversity, premature convergence, and a tendency to become stuck in local optima. The CSO and other swarm intelligence algorithms are prone to premature convergence, which cannot totally be avoided (Yang et al. 2023).

Various enhancement solutions have been presented by numerous scholars to solve the aforementioned inherent restriction. These methods may be generally divided into three groups: conventional CSO upgrades, hybrid CSO enhancements that combine the algorithm with other methods, and additional cutting-edge improvement methods. Notably, compared to the original CSO method, these modified algorithms have consistently shown greater convergence efficiency and global search optimization capabilities (Goldanloo and Gharehchopogh 2022).

3.1 Improved chicken swarm optimization algorithms

The modified CSO method is often designed to accomplish optimization by changing the algorithm's operators, making implementation relatively simple (Gharehchopogh et al. 2020). The m-CSO algorithm will be used as an illustration of this kind of algorithm in this section. Additionally, we will pay particular attention to introducing the SA-CSO algorithm and the Integrated CSO (ICSO) algorithm collection. In the fourth section, we'll talk about how these algorithms are used in practice.

3.1.1 Modified chicken swarm optimization (m-CSO)

The author put forward m-CSO in the literature (Fu et al. 2019). The literature examined the composition and organization of a chicken swarm and made the case that the mother hen model has a direct impact on how well the chicken swarm algorithm performs. The accuracy and stability of the m-CSO algorithm are found to be better than the CSO algorithm, as confirmed by experiments using the Rosenbrock test function, when the equation for updating the position of the mother hen is changed from formula (3) to formula (11).

$$M_{ij}^{t+1} = M_{ij}^t + k_1 * rand * (R_{mi}^t - M_{ij}^t) + k_2 * rand * (M_{ij}^t - RM^t) \tag{11}$$

3.1.2 Improved chicken swarm optimization (ICSO)

By refining the CSO method and renaming it the improved CSO (ICSO) algorithm, researchers have successfully solved the issue of local optima. ICSO, which stands for improved CSO, is a comprehensive collection of algorithms developed by academics.

In the literature (Wu et al. 2015), the author proposed ICSO-I. The location of the chick operator in the CSO algorithm is exclusively reliant on the position of the hen and is unaffected by the position of the rooster, according to research into the positional dynamics of chicks. As a result, the chick operator is unable to acquire crucial data produced from the rooster’s whereabouts. This restriction indicates that the chick operator is as vulnerable to becoming trapped in a local optima that is difficult to leave when the hen operator does.

The current study advises changing Eq. (9) to Eq. (12) in the chick’s location equation to allay this worry. Using a set of eight benchmark issue test functions, a comparison is made between ICSO-I and other well-known algorithms, including PSO, BA, CSO, and ICSO-I. The experimental results support the significant accuracy improvements made by ICSO-I, which are especially noticeable in high-dimensional problem settings and outperform the three alternative methods under study.

$$C_{ij}^{t+1} = \omega * C_{ij}^t + F * (H_j^t - C_{ij}^t) + C * (R_j^t - C_{ij}^t) \tag{12}$$

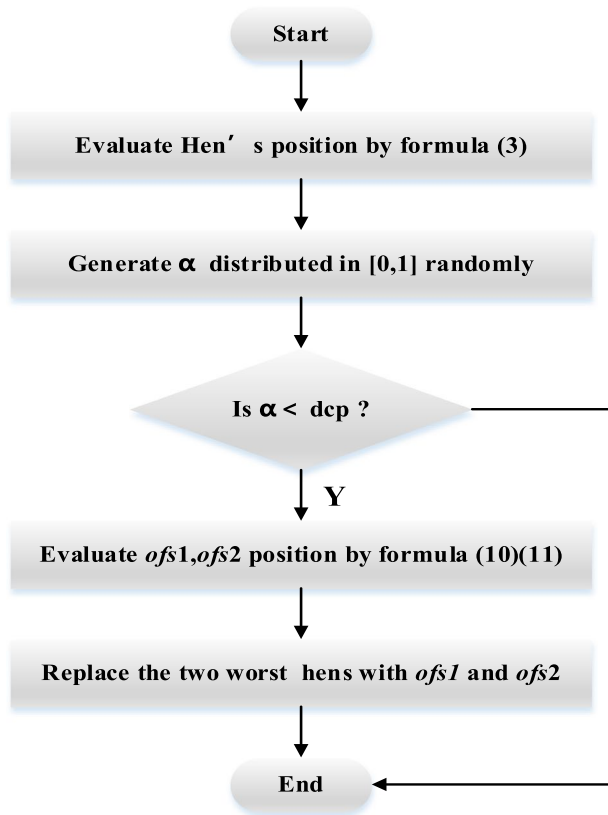
wherein, in Eq. (12), C is a learning parameter representing the extent to which the chicks are influenced by the rooster. R_j^t represents the dominant rooster associated with the i -th chick after t iterations. The self-learning coefficient for the chick is denoted by ω , with its value ranging from 0.4 to 0.9. The calculation formula for ω is outlined in Eq. (13).

$$\omega = \omega_{fin} * (\omega_{ini} / \omega_{fin})^{[1 / (1 + 10 * t / \omega_{max})]} \tag{13}$$

wherein, in Eq. (13), ω_{fin} is the final value of the iteration, ω_{ini} the initial value of the iteration, t is the current iteration count, and ω_{max} is the maximum value of the iteration. Ω is calculated based on the given initial value and iteration count.

The ICSO-II approach, which addresses the issue that the hen operator in CSO is prone to falling into the local optimum solution, is suggested by the author in the literature (Wu et al. 2018a) and offers a measure to increase the crossover probability. Figure 3 depicts the precise use of the crossover probability enhancement. A default crossover probability (dcp) parameter is set while the algorithm is running. Equation (3) is used to update the hen operator, and after that, a random number is produced from a uniform distribution in the $[0, 1]$ range. If $\alpha < dcp$,

Fig. 3 The flow chart crossover probability improvement



the two hens with the best fitness values among all hen operators are selected for the crossover calculation using Eqs. (14) and (15). This produces offspring operators ofs_1 and ofs_2 . Finally, ofs_1 and ofs_2 replace the two hen operators with the poorest fitness values. By replacing hen operators with subpar fitness values, the crossover probability enhancement has the advantage of lowering the likelihood that the algorithm would enter a local optimum. Furthermore, there is a chance that in the same iteration, the fitness values of the progeny hen operators will be higher than those of the rooster operators. The progeny hen operators will be identified as rooster operators in accordance with the hierarchy concept of CSO, increasing the algorithm's accuracy. By comparing the performance of ICSO-II with CSO, ABC, PSO, and ACO using the cost function of minimal aerodynamic heating rate, the improved algorithm is assessed. According to experimental simulations, ICSO-II outperforms other intelligent algorithms in terms of convergence efficiency and algorithm correctness.

$$ofs1 = p * hen_1^t + (1 - p) * hen_2^t \quad (14)$$

$$ofs2 = (1 - p) * hen_1^t + p * hen_2^t \quad (15)$$

wherein, in Eqs. (14) and (15), hen_1^t and hen_2^t represent the two hen operators with the best fitness values after t iterations. p is a percentage coefficient that represents the proportion of inheritance from hen_1^t and hen_2^t to the offspring ofs_1 and ofs_2 .

In the literature (Liang et al. 2020a), the author proposed ICSO-III. The hen and chick operators in CSO benefit from the advances brought about by this algorithm. A modified search technique based on "Levy flight" is suggested to overcome the issue of the hen operator in CSO slipping into local optima. The comparatively high number of hen parameters among all the factors is taken into consideration by this technique. It modifies the hen's search behavior such that it combines less long-range searches with close-range, in-depth searches. Equation (16) illustrates the enhanced "Levy flight" search technique. The method also advises switching the search pattern of the chick operator to the chick search Eq. (12) from ICSO-I in order to avoid the problem of the chick operator in CSO being affected by the hen and falling into local optima. Finally, it is determined from studies employing eight test functions that the ICSO-III algorithm greatly outperforms the CSO and PSO algorithms in terms of accuracy and efficiency.

$$H_{i,j}^{t+1} = H_{i,j}^t + k_1 * rand * (R_{Hi}^t - M_{i,j}^t) + k_2 * rand * L_{evy}(\lambda) \otimes (RH^t - H_{i,j}^t) \quad (16)$$

wherein, in Eq. (16), $L_{evy}(\lambda)$ represents a random search jump function that follows a Levy distribution. λ is a random number distributed between 1 and 3. \otimes is the dot product operator, which represents the inner product of vectors.

A solution to the problem of the chick operator in CSO being vulnerable to local optima controlled by the hen operator was provided in the literature (Wang et al. 2020) by the author as ICSO-IV. Equation (6) should be changed to Eq. (17), according to the algorithm. The crucial change keeps the hen operator's hold over the chick operator while adding the rooster operator. In addition, as shown in Eq. (18), a dynamic cosine inertia weight coefficient is included to control the effect of the rooster. Equation (18) indicates that the coefficient range lies within $[w_{min} - w_{max}, w_{max} + w_{min}]$. The degree of the rooster's effect over the chick is determined by the coefficient's absolute value. The number of iterations has an impact on the coefficient's size. This improvement creates a more even distribution by adding a rooster impact component to the chick calculation. This should lessen the chance that the chick operator may enter a local optimum. In reference (Ishikawa et al. 2020), function simulations are used to compare ICSO-IV with GA and PSO algorithms. The results show that ICSO-IV greatly increases algorithm accuracy when compared to GA and PSO algorithms.

$$C_{i,j}^{t+1} = C_{i,j}^t + F * (H_{i,j}^t - C_{i,j}^t) + w * (R_{i,j}^t - C_{i,j}^t) \quad (17)$$

$$w = \frac{w_{max} - w_{min}}{2} \cos \frac{\pi}{t_{max}} t + \frac{w_{max} + w_{min}}{2} \quad (18)$$

wherein, in Eq. (17), w is the dynamic cosine inertia weight coefficient, and $R_{i,j}^t$ is the corresponding dominance of the rooster over the i -h chick after t iterations. Wherein, in Eq. (18), w_{max} is the maximum inertia coefficient set. w_{min} is the minimum inertia coefficient set. t_{max} is the maximum number of iterations. t represents the current iteration count.

ICSO-V was suggested by the author in the literature (Liu et al. 2020a). Both the hen and the chick are susceptible to falling into a local optimum when the rooster operator achieves a local optimum solution since the rooster operator is crucial to the CSO algorithm. The approach incorporates the cosine inertia weight Cip to improve the rooster's capacity to perform local searches in order to resolve this issue. The inertia factor, Cip , is represented in Eq. (19), which is the modified version of Eq. (4). Equation (20) provides the definition of Cip . The rooster operator may switch between

global and local searches because to this inertia component, which lessens the chance of entering a local optimum. Additionally, ICSO-V addresses the issue of the hen operator controlling the chick operator solely. The learning part technique, which changes Eq. (9) to Eq. (20), is the improvement it suggests. Equation (20) introduces the global best operator into the chick operator equation and sets a learning coefficient to control its impact. The purpose of this technique is to lessen the likelihood that the chick operator may become stuck in local optima. In comparison to CSO, PSO, and WOA algorithms, simulations in the literature (Liu et al. 2020a) employing six distinct test functions show that ICSO-V produces more accurate computational results, especially for high-dimensional issues.

$$R_{ij}^{t+1} = Cip * R_{ij}^t + R_{ij}^t * randn(0, \delta^2) \quad (19)$$

$$Cip = Cip_{min} + (Cip_{max} - Cip_{min}) * \cos\left(\pi * \frac{t}{T}\right) \quad (20)$$

wherein, in Eq. (19), Cip is the cosine inertia factor, which is used to enhance the search capability of the rooster. Wherein, in Eq. (20), Cip_{max} is the maximum value of the inertia factor, set to 0.8, Cip_{min} is the minimum value of the inertia factor, set to 0.3, T is the maximum number of iterations, and t represents the current iteration count.

By altering the equations in the CSO algorithm, the aforementioned group of ICSO algorithms seeks to solve the unique flaws of roosters, hens, and chicks. This technique to improvement has the advantage of being straightforward and simple to apply, which has certain algorithmic optimization implications. An extensive range of enhanced algorithms is also included in the ICSO algorithm library. In the literature (Bharanidharan and Rajaguru 2020), the author presented ICSO-VI, which improves the CSO algorithm through four enhancement measures: controlled randomness optimization (CRO), cross-over operator (CO), rooster selection (RSEL), and. Figure 4 shows the flowchart for this thorough upgrade.

CPT involves optimizing the control parameters during initialization. In this measure, the algorithm seeks values for the parameters k_1 , k_2 , and F that yield high-precision results for the optimization function. Experimental testing determined that using $k_1=0.7$, $k_2=0.5$, and $F=0.9$ leads to highly accurate results for most optimization functions. Recombination strategy for roosters is called RSEL. The placements of the rooster individuals are changed using the roulette wheel selection technique after the algorithm calculates the fitness values and obtains groups of hens, roosters, and chicks. People in the rooster species that score lower on the fitness scale are more likely to switch roles. Position considering rate (PCR) and position adjusting rate (PAR), two fixed parameters that must be established in order to use CRO as an improvement metric. Following the computation of Eqs. (4), (7), and (9), this measurement is made. Following CRO, CO is an improvement metric. According to Eqs. (14) and (15), the crossover procedures on the hen individuals are the main focus of this measurement. The two patients with the lowest fitness scores are changed after the procedure. The CSO algorithm's parameters, rooster equations, hen equations, and overall fitness are all the focus of ICSO-VI's four improvement measures. It has been discovered through simulation testing that the ICSO-VI algorithm greatly increases accuracy when compared to the CSO algorithms.

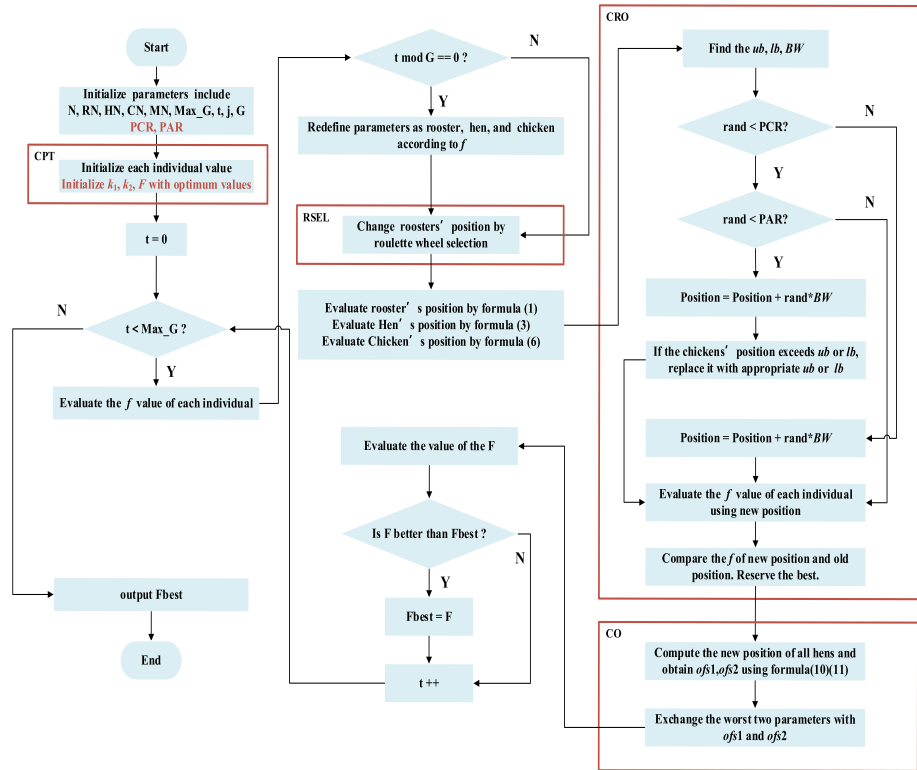


Fig. 4 ICSO-VI algorithm flow chart

3.1.3 Self-adaptive chicken swarm optimization (SA-CSO)

In the literature (Kumari et al. 2022a), the author proposed SA-CSO algorithm. The problem of the CSO method becoming stuck in local optima when handling high-dimensional problems was addressed in the literature. After updating Eqs. (1), (3), and (6) in the CSO method, it includes a self-adaptive improvement strategy that modifies the placements of the flock by introducing an adaptive parameter (Gheibi et al. 2021). The flock is aided in escaping local optima by this change. Equation (21) defines the adaptively parameter as follows, while Eq. (22), which focuses on the global random flock operator, describes how the flock is adjusted. SA-CSO is straightforward to use and makes progress in avoiding local optimum. It has been discovered through experimental simulations that SA-CSO surpasses other algorithms in terms of precision and accuracy, including CSO, PSO, WOA, and multi-verse optimizer algorithm (MVO) (Abualigah 2020).

$$\alpha(i) = \frac{F_i(ts - 1) - F_i(ts)}{F_i(ts - 1)} \tag{21}$$

$$C_{ij}^{ts}(i, po) = C_{ij}^{ts}(i, po) \times \alpha(i) \tag{22}$$

wherein, in Eq. (21), (i) is the adjustment parameter for the selected individual i . $F_i(ts)$ is the fitness value of individual i in the ts iteration. Wherein, in Eq. (22), $C_{i,j}^{ts}(i, po)$ is the adjusted position of the selected individual i after the fitness parameter adjustment in the ts iteration. Here, l is the ranking position of the individual among all individuals, j represents the dimension, and po is a random number indicating the likelihood of being selected for fitness adjustment.

3.2 Hybrid improved chicken swarm optimization

Standalone methods fall short of properly resolving high-dimensional uncertain optimization issues in the actual world (Radaideh and Shirvan 2021). In order to enhance, hybrid improved chicken swarm algorithms combine the chicken swarm algorithm with additional clever optimization techniques (Chen et al. 2022). The Bat-Chicken Swarm Optimization (CSO-BA), Chicken Swarm plus Deer Hunting Optimization Algorithm (CSO-DH), and Hybrid Ant Lion Chicken Swarm Optimization Algorithm (CSO-ALO) are the three hybrid enhanced algorithms that will be covered in this part. In the fourth section, we'll talk about how these algorithms are used in practice.

3.2.1 Bat-Chicken swarm optimization (CSO-BA)

In the literature (Kong et al. 2020), the author proposed CSO-BA algorithm. The Bat Algorithm is a search algorithm that excels at short-range and high-precision search. It controls the direction of algorithm advancement through two factors: the loudness of sound waves A_i and the frequency r_i (Bangyal et al. 2022). With the help of this technique, the algorithm is guaranteed to advance toward greater precision while simultaneously having a chance of successfully escaping from local optima. Based on the Bat Algorithm's recommendations for improvement, CSO-BA incorporates the Bat Algorithm's control parameters into the Rooster operator of CSO. The Rooster Eq. (4) in CSO is modified to Eq. (23), where A_i and r_i are defined as Eq. (24) shows. This improvement effectively avoids the Rooster operator from getting stuck in local optima and enhances the computational precision.

$$R_{i,j}^{t+1} = \begin{cases} x_{best} + 0.01 * randn(1, N) & Rand \geq r_i^t \\ R_{i,j}^t * randn(0, \sigma^2) + \epsilon * A_i^t & Rand < r_i^t \end{cases} \quad (23)$$

$$A_i^{t+1} = \alpha A_i^t, \quad r_i^{t+1} = r_i^t(1 - \exp(-\lambda t)) \quad (24)$$

wherein, in Eq. (23), x_{best} is the global best solution among the N individuals in the chicken swarm, and ϵ is a random number distributed in the range $[-1,1]$. Wherein, in Eq. (24), α and λ are limiting parameters, which are often set to a value of 0.9 in most cases.

The mother chicken and chick operators are also improved by CSO-BA. To improve the unpredictability of the mother chicken's location and lessen the likelihood that it would become caught in local optima, the parameter F of the chicks is included into the mother chicken Eq. (6), resulting in Eq. (25). Equation (9) is changed to Eq. (26) to include the rooster's position, which increases the unpredictability of the chick's position and lessens the possibility that it will become stuck in local optima.

$$H_{i,j}^{t+1} = F * H_{i,j}^t + k_1 * rand * (R_{Hi}^t - M_{i,j}^t) + k_2 * rand * (RH^t - H_{i,j}^t) \quad (25)$$

$$C_{ij}^{t+1} = C_{ij}^t + F * (Hf_j^t - C_{ij}^t) + rand * (Rf_j^t - C_{ij}^t) \tag{26}$$

Based on the improved operators, the flowchart of the CSO-BA algorithm is shown in Fig. 5, and the specific steps are as follows:

- Step 1: Initialize all chicken swarm individuals. The initial positions of the chicken swarm are defined by Eq. (27).
- Step 2: Apply the CSO algorithm to classify people into roosters, mother hens, and chicks according to their fitness scores. Equations (23), (25), and (26) should be used to update people's locations.
- Step 3: Calculate the updated fitness values and compare them with the previous values. Keep the individuals with better fitness values.
- Step 4: Update the parameters of sound loudness and frequency.
- Step 5: Check if the maximum number of iterations is reached. If so, end the loop and output the maximum value.

$$x_{ij} = Lb + (Ub - Lb) * rand \tag{27}$$

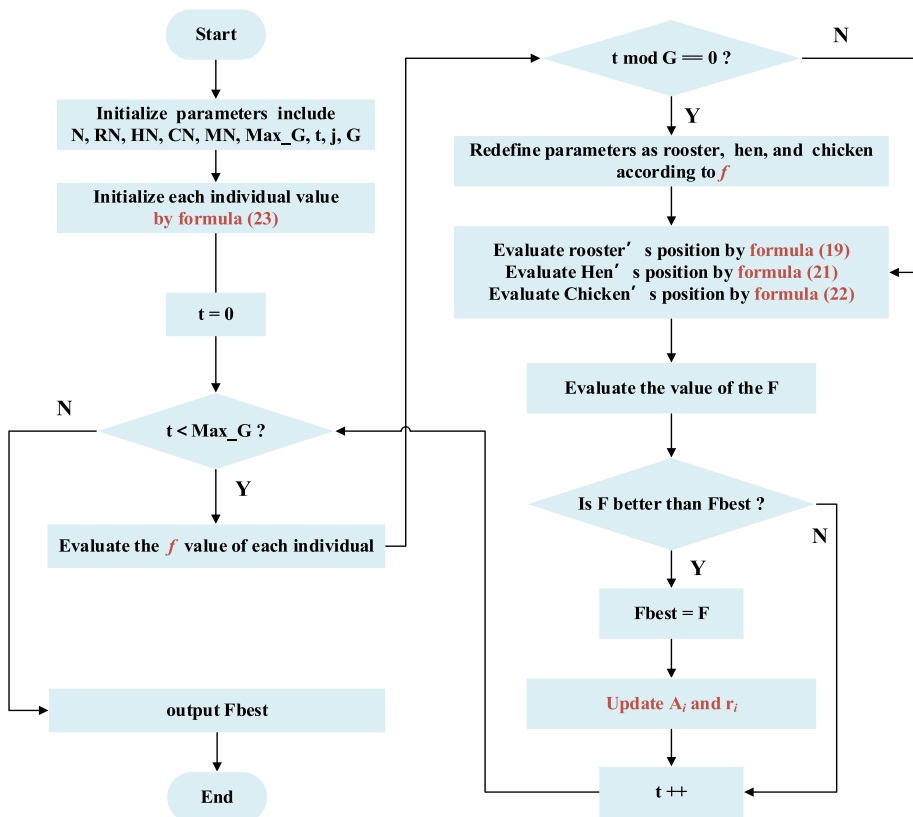


Fig. 5 CSO-BA algorithm flow chart

wherein, in Eq. (27), $x_{i,j}$ is a random individual. Ub is the maximum value of the entire search space. Lb is the minimum value of the entire search space, and $rand$ is a random number distributed between 0 and 1.

3.2.2 Chicken swarm-plus deer hunting optimization algorithm (CSO-DH)

The CSO-DH method was suggested by the author in the literature (Gawali and Gawali 2021). It is an intelligent optimization system that was motivated by how hunters hunt deer. To arrive at optimal position solutions, this algorithm modifies the hunter’s location depending on several hunting strategies (Kanna et al. 2021). The data may be represented as illustrated in Eq. (28), supposing a collection of hunter positions designated as P . The wind angle and the deer angle are two identified hunting effect elements. Equations (29) and (30) yield the definitions of a and b , respectively.

$$P = \{P_1, P_2, \dots, P_j, \dots, P_n\}; 1 \leq j \leq n \tag{28}$$

$$\theta_k = 2\pi r \tag{29}$$

$$a_k = \frac{\pi}{8} \times r \tag{30}$$

wherein, in Eq. (28), j is the currently selected position, and n is the total number of hunters. Wherein, in Eqs. (29) and (30), r is a random number between 0 and 1, and k represents the current iteration number.

According to Deer Hunting Optimization, there are three determining elements that affect the hunting position: the optimal hunting position P^{lead} , the wind angle θ and the deer angle a that affect the hunting, and the successful hunting position $P^{success}$. Based on these factors, the strategy updates in the deer hunting algorithm are defined as Eqs. (31), (32), and (33). Equation (31) represents the strategy for the optimal position, Eq. (32) represents the angle strategy, and Eq. (33) represents the strategy for successful hunting (Prabhakar and Veena 2023).

$$P_{k+1} = P^{lead} - M * c * \left| L \times P^{lead} - P_k \right| \tag{31}$$

$$P_{k+1} = P^{lead} - c * \left| \cos(v) \times P^{lead} - P_k \right| \tag{32}$$

$$P_{k+1} = P^{success} - M * c * \left| L \times P^{success} - P_k \right| \tag{33}$$

wherein, in Eq. (31), M and L are coefficient vectors, defined as shown in Eqs. (30) and (31) respectively. c is a parameter influenced by the angle factor, and its range is between 0 and 2. Wherein, in Eq. (32), v is an influencing parameter defined as $v = \Phi_{i+1}$, where $\Phi_{i+1} = \Phi_i + d_i$, $\Phi_i = \theta_i + \pi$, and $d_i = \theta_i - a$.

$$M = \frac{1}{4} \log \left(k + \frac{1}{k_{max}} \right) * b \tag{34}$$

$$L = 2 * r \tag{35}$$

wherein, in Eq. (34), k_{max} represents the maximum number of iterations, and b is a random number defined in the range $[-1, 1]$. Wherein, in Eq. (35), r is a random number defined in the range $[0, 1]$.

The strategy selection method of Deer Hunting Optimization is as follows: when $c > 1$, Eq. (32) is used to update the hunting position; when $c < 1$ & $L < 1$, Eq. (33) is used to update the hunting position; when $c < 1$ & $L > 1$, Eq. (31) is used to update the hunting position.

The CSO-DH method suggests substituting the iterative calculation process of the rooster Eq. (4) with the hunting position calculation process of Deer Hunting Optimization in order to address the problem of the rooster operator in the CSO algorithm being prone to local optima (Batra et al. 2021). The replacement is specifically carried out by determining if $b < 0$ is true. If so, the deer hunting technique is utilized to update the rooster operator; otherwise, the old rooster calculation equation is applied. In Fig. 6, the CSO-DH flowchart is displayed.

According to comparisons of test function results, the CSO-DH algorithm solves high-dimensional engineering problems more accurately and with less error loss than the RL algorithm, DHOA-RL algorithm, and CSO-RL algorithm.

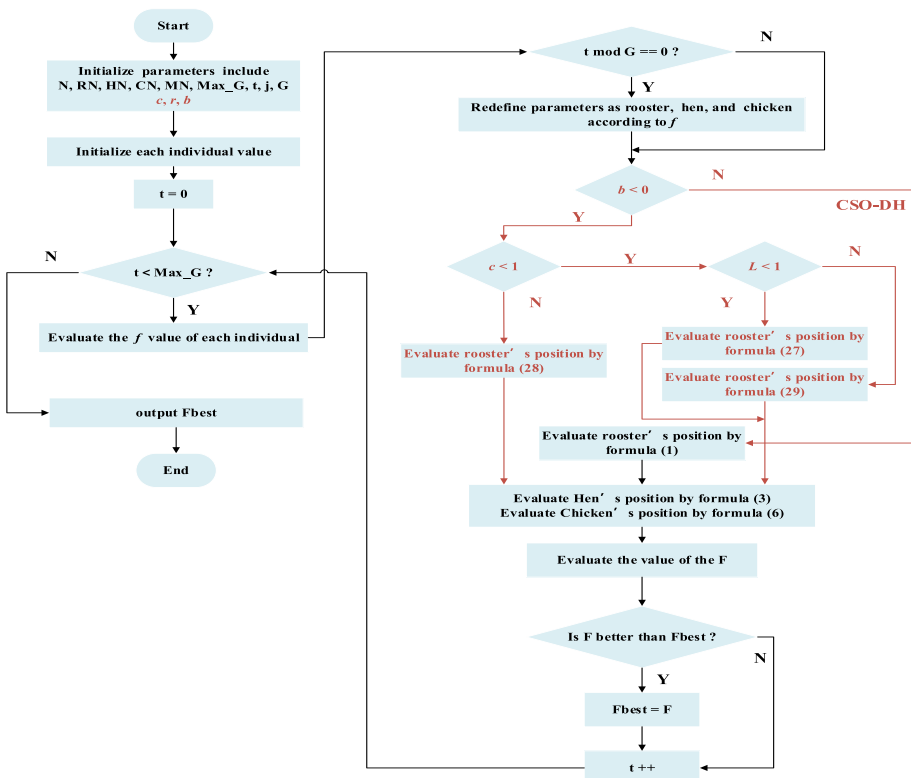


Fig. 6 CSO-DH algorithm flow chart

3.2.3 Hybrid ant lion chicken swarm optimization algorithm (CSO-ALO)

In the literature (Deb and Gao 2021), the author proposed CSO-ALO algorithm. The Ant Lion Optimization (ALO) algorithm’s main principle is to replicate the way that ant lions pursue their prey in order to accomplish global optimization (Abualigah et al. 2021). The ant lion digs a funnel-shaped trap in sandy ground before hunting, hiding at the bottom of the trap and watching for prey to come. A wandering ant that accidentally enters the trap is quickly caught and eaten by the ant lion, who then fixes the trap in preparation for the next hunt (Niu et al. 2022). The ALO algorithm is a population-diverse, highly optimized, parameter-adjustment-friendly, and readily implementable search technique (Mani et al. 2018). It integrates random walks, roulette wheel selection, and elite tactics.

Equation (36) depicts how the ALO algorithm initially specifies the ant’s random walk procedure. Since the ant’s range of movement is limited, utilizing Eq. (32) to directly simulate the ant’s journey might result in an out-of-bounds result. Therefore, depending on the established limits, Eq. (37) is used to normalize the ant’s movement.

$$A(t) = [0, \text{sum}(2r(t_1) - 1), \dots, \text{sum}(2r(t_n) - 1)] \tag{36}$$

wherein, in Eq. (36), $A(t)$ represents the set of steps taken by the ant during random walks. sum is the summation operation. t is the number of steps or iteration count of the ant’s movement. $r(t)$ is a random integer taken from the interval $[0, 1]$.

$$A_i^t = \frac{(A_i^t - a_i) * (d_i^t - c_i^t)}{(b_i - a_i)} + c_i^t \tag{37}$$

wherein, in Eq. (37), a_i is the minimum value of the i -th dimension variable. b_i is the maximum value of the i -th dimension variable. c_i^t is the minimum value of the i -th dimension variable during the t -th random walk. d_i^t is the maximum value of the i -th dimension variable during the t -th random walk.

The movement of ants is influenced by the traps created by antlions. Therefore, a mathematical model is assumed for the antlion traps, as shown in Eq. (38).

$$\begin{cases} c_i^t = AL_i^t + C^t \\ d_i^t = AL_j^t - d^t \end{cases} \tag{38}$$

wherein, in Eq. (38), C^t is the minimum value of all variables at the t -th iteration. d^t is the maximum value of all variables at the t -th iteration. AL_j^t is the position of the selected j -th antlion at the t -th iteration.

Only one antlion at a time may affect an ant colony’s movement. To identify which ant gets preyed upon by which antlion, the computer employs a roulette wheel selection technique (Rani and Garg 2021). According to this method, an antlion’s fitness directly relates to the likelihood that it will catch an ant. Equation (39) is used by the program to replicate this adaptive process.

$$c^t = \frac{c^t}{I}, d^t = \frac{d^t}{I} \tag{39}$$

wherein, in Eq. (35), I represent the scaling factor, which is defined as shown in Eq. (40).

$$I = \begin{cases} 1, & t \leq 0.1T \\ 10^v * \frac{t}{T}, & t > 0.1T \end{cases} \tag{40}$$

wherein, in Eq. (40), T is the maximum number of iterations, and v is a variable that changes as the number of iterations increases.

During the iteration process, if an ant's fitness value is lower than that of its corresponding antlion, the antlion is said to have captured the ant (Li et al. 2022a). Equation (41) is used to update the antlion's location. The elite antlion is chosen based on its fitness rating, and it uses Eq. (42) to decide where the ants will go to in the future.

$$AL_j^t = A_i^t, \text{ if } f(A_i^t) \leq f(AL_j^t) \tag{41}$$

$$A_i^{t+1} = \frac{R_A^t(L) + R_E^t(L)}{2} \tag{42}$$

wherein, in Eq. (41), f is the fitness function. Wherein, in Eq. (42), $R_A^t(L)$ is the value generated by the ant during the L -th step of walking with the corresponding antlion after t iterations. $R_E^t(L)$ is the value generated by the ant during the L th step of random walking around the elite antlion after t iterations. L represents any arbitrary value during the random walking of the ant.

Based on the defined ALO operators, the algorithm follows the specific steps below:

- Step 1: Data initialization. Determine the number and dimensions of antlions and ants, and initialize their positions within the feasible domain. Proceed to Step 2.
- Step 2: Calculate the fitness values for all individuals and determine the elite antlion. Proceed to Step 3.
- Step 3: Update the Eq. (34's) parameters after choosing your antlions using the roulette wheel selection technique. Based on Eqs. (36) and (37), the ants change their locations randomly around the elite antlion. Apply Eq. (42) to the ant locations to update them. Proceed to Step 4.
- Step 4: Increase the iteration count by 1. Check if the maximum iteration count has been reached. If yes, end the iteration; if no, go back to Step 2. The ALO algorithm flowchart is shown in Fig. 7.

A mixture of the CSO and ALO algorithms is known as the CSO-ALO algorithm. A hybrid optimization solution is produced by first using the ALO algorithm to locate the position of the elite antlion and then using the CSO algorithm to further optimize the elite antlion (Asna et al. 2022). Figure 8 depicts the precise flowchart of the hybrid optimization technique. The hybrid optimization technique adds more layers of optimization than typical optimization algorithms, which has benefits for accuracy. In literature (Deb and Gao 2021), it was effectively proved that the CSO-ALO hybrid algorithm considerably enhances the accuracy of the method when compared to the ALO and CSO algorithms in optimizing different load scheduling issues.

3.3 Other improved chicken swarm optimization algorithms

Other categories of enhanced CSO algorithms will be covered in this section. Score-based improvements, genetic evolution-based improvements, chaos theory-based improvements,

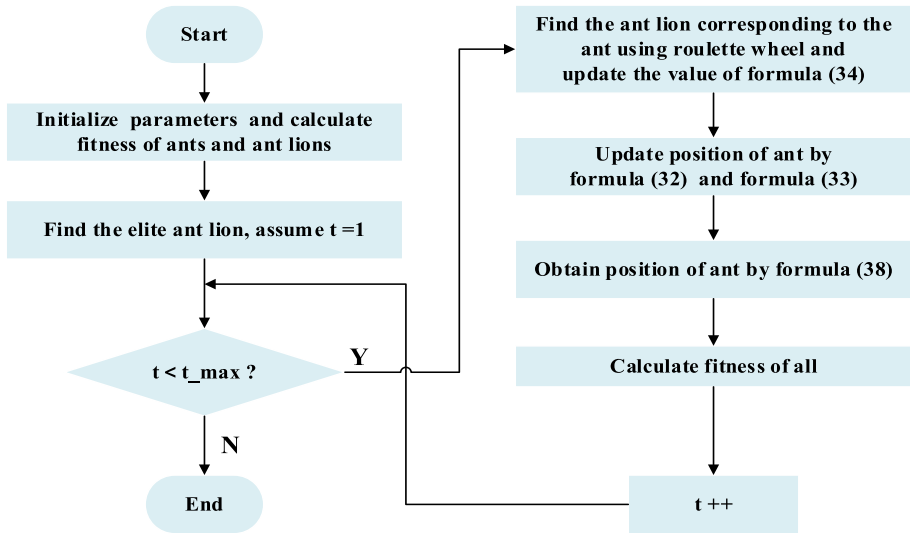


Fig. 7 The flow chart of ALO algorithm

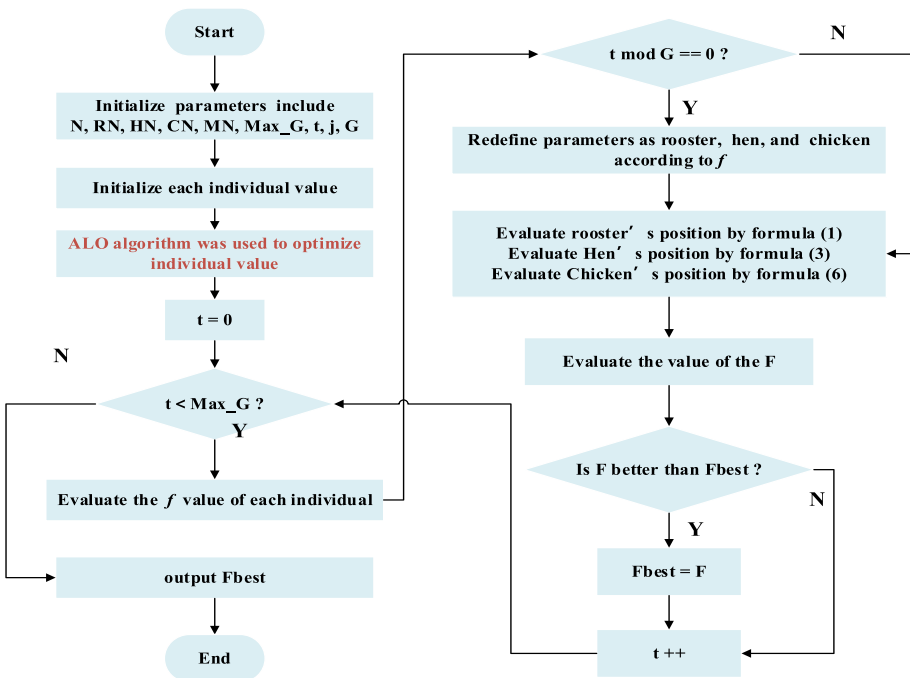


Fig. 8 The flow chart of CSO-ALO algorithm

neural network-based improvements, and improvements for multi-objective optimization are some of the several types of improvements.

3.3.1 Fractional-chicken swarm optimization (Fractional-CSO)

In the literature (Cristin et al. 2021), the author proposed the Fractional-CSO algorithm. With this method, the fractional calculus is combined with the ideas of hierarchy and flock behavior. The Rooster operator in Eq. (4) is changed to Eq. (43) in order to solve the problem of the Rooster operator easily becoming caught in local optima. The modified Rooster calculation process is influenced by different stages of iteration, thereby enhancing the global randomness of the chicken flock.

$$R_{x,y}^{t+1} = R_{x,y}^t (\beta + \delta(0, \kappa^2)) + \frac{1}{2} \beta R_{x,y}^{t-1} + \frac{1}{6} (1 - \beta) R_{x,y}^{t-2} + \frac{1}{24} \beta (1 - \beta) (2 - \beta) R_{x,y}^{t-3} \quad (43)$$

wherein, in Eq. (43), β is the fractional order, which represents the intermediary difference between the current Rooster term and the calculated term. κ^2 is defined as shown in Eq. (44).

$$\kappa^2 = \begin{cases} 1, & \text{if } f_x \leq f_m; m \in [1, A], m \neq x \\ \exp\left(\frac{(f_m - f_x)}{|f_x| + y}\right), & \text{otherwise} \end{cases} \quad (44)$$

wherein, in Eq. (44), f represents the fitness value of the Rooster individual, m is a random individual index, and y is a very small number.

3.3.2 CSO based clustering algorithm with genetic algorithm (CSOCA-GA)

In the literature (Osamy et al. 2020), the author proposed the CSOCA-GA algorithm. The goal of this approach is to solve the discrete data clustering issue. The algorithm achieves parameter clustering by first introducing Eq. (45) to the binary of the chicken flock's individuals and employing discrete individuals as the output for the program's optimal solution. The algorithm incorporates the crossover operator of the genetic algorithm into the computation of each iteration, providing a certain probability to escape from local optima and increase the accuracy of the algorithm. This prevents the algorithm from getting stuck in subpar local optima and producing inferior clustering results. Figure 9 depicts the genetic algorithm's flowchart.

$$b_x = \begin{cases} 1, & \text{if } \frac{1}{1+e^{-x}} > 0.5 \\ 0, & \text{otherwise} \end{cases} \quad (45)$$

wherein, in Eq. (45), x is the fitness value of the corresponding parameter.

3.3.3 Chaotic chicken swarm optimization (CCSO)

The initialization of parameters and operator iterations in the CSO algorithm both require procedures that produce random numbers. To deal with unpredictable processes, the CCSO algorithm applies chaos theory. According to experimental findings, adopting chaotic sequences instead of pseudo-random numbers for population initialization, selection,

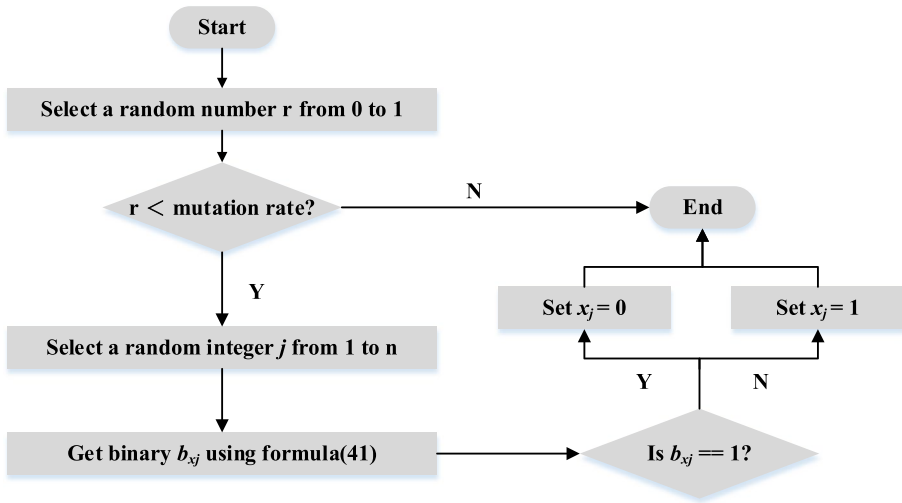


Fig. 9 The flowchart of genetic algorithm

crossover, and mutation operations might significantly alter the entire algorithmic process and produce superior outcomes.

According to the literature (Li et al. 2019), the rooster, hen, and chick operators in the CSO algorithm are processed at random using the chaos theory’s tend map and logistics map. This raises the likelihood that the algorithm will elude local optima. To demonstrate that CCSO outperforms CSO and PSO algorithms in a variety of data processing areas, five test datasets were employed.

The tend map is shown in Eq. (46), and the logistics map is shown in Eq. (47).

$$\begin{aligned}
 P_{i+1} &= \frac{P_i}{Const1} \\
 P_{i+1} &= Const2 * (1 - P_i)
 \end{aligned}
 \tag{46}$$

$$P_{i+1} = Const3 * (1 - P_i)
 \tag{47}$$

wherein, in Eq. (46), P_{i+1} is the new position of the individual, and P_i is the current position. $Const1$ is set to 0.7, $Const2$ is set to 3.33. Wherein, in Eq. (47), $Const3$ is set to 4.

The SCCSO method, which updates the location of the individual with the worst fitness value in each iteration using chaos theory, was put out by the author in the literature. Equation (48) illustrates the new position’s computation.

$$X_w = \begin{cases} X_{Lr}, & \text{if } Rand2 \geq 1 - \frac{t}{t_{max}} \\ X_{Lr} + Rand1 * (2C_k - 1), & \text{otherwise} \end{cases}
 \tag{48}$$

wherein, in Eq. (48), X_w is the updated position after the worst position is updated. X_{Lr} is the position of the individual with the worst fitness value. $Rand1$ and $Rand2$ are random numbers distributed between [0,1]. t is the current iteration count. t_{max} is the maximum number of iterations, and C_k is a parameter for the chaotic sequence defined in Eq. (49).

$$C_{k+1} = 4 * C_k * (1 - C_k) \tag{49}$$

3.3.4 Adaptive chicken swarm optimization (ACSO)

Adaptive Chicken Swarm Optimization(ACSO) is suggested as a viable solution, targeting the issues of low convergence accuracy and easy fall into local optimal solutions for the complicated high-dimensional problems treated by flocking. By enhancing the iterative process, the ACSO may determine the parameter selection adaptively during the algorithm, thereby guaranteeing the flocking algorithm’s ability to do global depth searches (Wang et al. 2021a).

The author suggested the ACSO algorithm in the literature. The choice of constants in the tend map and logistics map was a topic of discussion in the literature since it affected algorithm speed and accuracy. Equation (50) illustrates how the tend map is updated by ACSO, and initialization values for the parameter in Eq. (51) are established in accordance with the modified tend map.

$$P_{i+1} = \begin{cases} \frac{10}{7}P_i, & P_i < 0.7 \\ \frac{10}{3}P_i(1 - P_i), & otherwise \end{cases} \tag{50}$$

$$X_{ini} = X_{min} + P_{i+1}(X_{max} - X_{min}) \tag{51}$$

wherein, in Eq. (51), X_{ini} is the parameter initialization in the ACSO algorithm. X_{max} is the upper bound of the parameter, and X_{min} is the lower bound of the parameter.

To prevent being stuck in local optima, the ACSO algorithm combines the ideas of search rate and search range. In order to do this, the ACSO algorithm incorporates the hen’s influence component w , which is made up of the search rate influence factor h and the search range influence factor s . Equation (52), instead of Eq. (3), modifies the hen operator. The ACSO method has been found to have superior accuracy, stability, and computing efficiency when compared to the CSO algorithm through simulation studies utilizing seven distinct test functions.

$$H_{ij}^{t+1} = wH_{ij}^t + k_1 * rand * (R_{Hi}^t - M_{ij}^t) + k_2 * rand * (RH^t - H_{ij}^t) \tag{52}$$

wherein, in Eq. (52), w is the hen’s influence factor, which is defined in Eq. (53).

$$w = a_0 + a_1 e^{-a_2 \frac{h}{s}} \tag{53}$$

wherein, in Eq. (53), a_0, a_1, a_2 are random numbers distributed in the range (0,1]. h is the search rate influence factor ranging between (0,1], where a larger h corresponds to a slower search rate, and the algorithm converges when h equals 1. s is the search range influence factor ranging between (0,1], where a smaller s corresponds to a larger search range, and all individuals in the algorithm have the same fitness value when s equals 1.

3.3.5 Algorithm complexity analysis

Assuming that the number of populations of the algorithm is N , the dimension of the search space is D , and the maximum number of searches is T_{max} , the complexity of PSO-CSO

Table 3 General improvements of CSO

Algorithm	Methods for improvement	Targets for improved	Author
m-CSO	POO	Hen’s formula	Chen (Fu et al. 2019)
ICSO-I	POO	Chicken’s formula	Wu D (Wu et al. 2015)
ICSO-II	POO	Hen’s formula	Wu Y (Wu et al. 2018a)
ICSO-III	FOO	Hen’s formula	Liang X (Liang et al. 2020a)
ICSO-IV	POO	Chicken’s formula	Wang J (Wang et al. 2020)
ICSO-V	POO	Rooster’s formula	Liu (Liu et al. 2020a)
ICSO-VI	POO,FOO	Chicken swarm	Bharanidharan (Bharanidharan and Rajaguru 2020)
SA-CSO	FOO	Chicken swarm	Kumari (Kumari et al. 2022a)

Table 4 Hybrid improvements of CSO

Algorithm	Methods for improvement	Targets for improved	Author
CSO-BA	POO	Chicken swarm Initial Settings	Liang S (Kong et al. 2020)
CSO-DH	POO,FOO	Rooster’s formula	Gawali (Gawali and Gawali 2021)
CSO-ALO	POO,FOO	Initial Settings	Deb S (Deb and Gao 2021)

includes: the initialization complexity of the population $O(ND)$, the fitness value calculation complexity $O(ND)$, the position update complexity of the global and local search is $O(N^2 \log D)$, the fitness value sorting complexity of the algorithm is $O(N^2)$ and the control parameter update complexity of the algorithm is $O(ND)$. Then the complexity of the PSO-CSO algorithm is shown in Eq. (54):

$$O(PSO - CSO) = O(ND) + O(T_{max})O(ND + N^2 \log N + N^2 + ND) \tag{54}$$

The algorithm time complexity of CSO is shown in Eq. (55):

$$O(CSO) = O(ND) + O(T_{max})O(N^2 \log N + N + ND) \tag{55}$$

3.4 Analysis of different improved algorithms

The primary directions for CSO algorithm improvement may be split into two categories. First, the field of optimization (FOO) needs to be improved to make it appropriate for certain optimization goal functions. Examples that come to mind include CSOCA-GA. The second step is to accelerate convergence and increase precision in the optimization process (POO). Examples that come to mind are ICSO-VI and CSO-ALO.

Through MATLAB simulations, the performance of all the aforementioned enhanced algorithms has been shown, with varied degrees of improvement over the CSO algorithm. However, there is presently little information on how these enhanced algorithms compare, which can be examined in upcoming reviews and more in-depth studies. Tables 3, 4 and 5

Table 5 Other improvements of CSO

Algorithm	Methods for improvement	Targets for improved	Author
Fractional-CSO	POO	Rooster's formula	Cristin (Cristin et al. 2021)
CSOCA-GA	POO,FOO	Chicken swarm	Osamy (Osamy et al. 2020)
CCSO	POO	Random setting	Ahmed (Li et al. 2021b)
SCCSO	POO	Chicken swarm	Li M (Mishra et al. 2022)
ACSO	POO	Initial Settings Hen's formula	Xing Y (Wang et al. 2021a)

display a compilation of the general improvements' algorithms, hybrid improvements algorithms, and additional improvements algorithms discussed in this section.

4 Function testing and performance analysis

In order to study the effectiveness of the chicken swarm optimization algorithm, we studied particle swarm optimization algorithm (PSO) (Nayak et al. 2023), chicken swarm optimization algorithm (CSO) (Li et al. 2021b), sine-cosine optimization algorithm (SCA) (Rizk-Allah and Hassanien 2023), multiverse optimization algorithm (MVO) (Mishra et al. 2022), tree species optimization algorithm (TSA) (Carreon-Ortiz and Valdez 2022), wind driven algorithm (WDO) (Ibrahim et al. 2020) and other 6 algorithms were tested. This paper conducts comparative experiments using 9 standard benchmark functions, which simulate the different difficulties of actual search spaces. See Table 6 for function details. To be more convincing, 30 independent experiments were performed for each test function in all cases, with a maximum number of iterations of 500 and a population size set to 50.

The integrated development environment for all experiments is Matlab_R2020b. The operating system is a 64-bit Windows 11 system. In order to verify the stability and convergence of the seven algorithms, the convergence comparison chart of each algorithm on the test function is listed, as shown in Fig. 10.

The horizontal axis in Fig. 10 represents the number of iterations of the algorithm, and the vertical axis represents the optimal fitness function value. From the convergence curve and the final data table, it can be seen that for the F1 function, CSO has the fastest convergence speed, and the convergence curves of CSO are all below the convergence curves of other algorithms. From the final results, the WDO algorithm has the best result. Judging from the final results, the average responsiveness value obtained using CSO optimization is 18.35, which is closer to the theoretical optimal value. Among them, PSO stops converging prematurely and has the worst performance. It shows that CSO's optimization ability for the F1 function is very stable, and its repetition accuracy is higher than other algorithms. For function F2, SCA has the best performance and the fastest convergence speed. For the F3 function, WDO results in better performance and better convergence speed than other algorithms. For function F4, WDO results are the best, followed by CSO. For function F5, the performance of WDO algorithm and CSO algorithm is better. For function F6, the CSO algorithm has the fastest convergence speed and the best results. For function F7, the six algorithms are still in the process of calculating the optimal value and have not reached convergence. For function F8, the CSO algorithm has the fastest convergence speed and the best results. For function F9, the CSO algorithm has the fastest convergence speed and the best results.

Table 6 Test functions

Function	Equation	Dimension	Bounds	Optimum
F1	$10d + \sum_{i=1}^d [x_i^2 - 10 \cos(2\pi x_i)]$	30	[-100,100]	0
F2	$-20 \exp \left(-0.2 \sqrt{\frac{1}{d} \sum_{i=1}^d x_i^2} \right) - \exp \left(\frac{1}{d} \sum_{i=1}^d \cos(2\pi x_i) \right) + 20 + \exp(1)$	30	[-5.12,5.12]	0
F3	$\frac{1}{4000} \sum_{i=1}^d x_i^2 - \prod_{i=1}^d \cos \frac{x_i}{\sqrt{i}} + 1$	30	[-32,32]	0
F4	$F12 = \frac{\pi}{d} \left\{ 10 \sin(\pi y_1) + \sum_{i=1}^{d-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_n - 1)^2 \right\} + \sum_{i=1}^d u(x_i, 10, 100, 4)$ $y_i = 1 + \frac{x_i + 1}{4} u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$	30	[-600,600]	0
F5	$0.1 \left\{ \sin^2(3\pi x_1) + \sum_{i=1}^{d-1} (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2 [1 + \sin^2(2\pi x_n)] \right\} + \sum_{i=1}^d u(x_i, 5, 100, 4)$	30	[-50,50]	0
F6	$\sum_{i=1}^d \left[a_i - \frac{x_i(b_i^2 + b_i x_i)}{b_i^2 + b_i x_i + x_i} \right]^2$	30	[-5,5]	0.0003
F7	$4x_1^2 - 2.1x_1^4 + \frac{1}{5}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4$	30	[-5,-5]	-1.0316

Table 6 (continued)

Function	Equation	Dimension	Bounds	Optimum
F8	$\left(x_2 - \frac{5.1}{4\pi^2}x_1^2 + \frac{5}{\pi}x_1 - 6\right)^2 + 10\left(1 - \frac{1}{8\pi} \cos x_1\right) + 10$	30	[-5,5]	0.398
F9	$\times [30 + (2x_1 - 3x_2)^2 \times (18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2)]$	30	[-2,2]	3

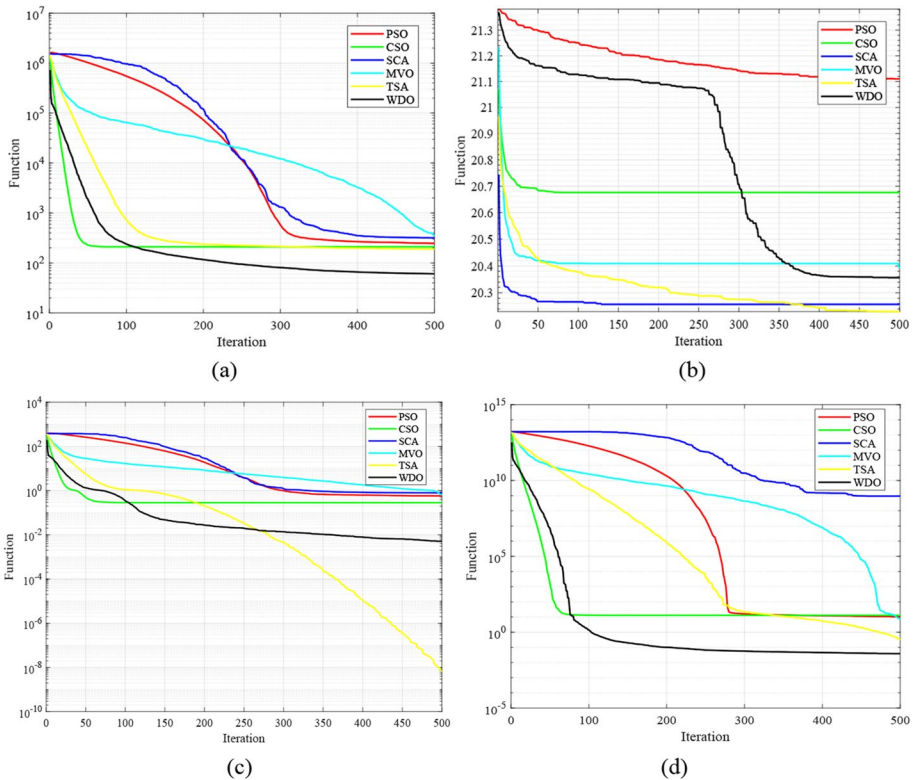


Fig. 10 Comparison of six algorithms to find the optimal solution

Judging from different test results, for different optimization applications, the performance of the algorithm needs to be analyzed in detail to determine which algorithm is better for a certain application. Generally speaking, the chicken flock algorithm is the fastest and has the best results in solving the minimum value of most functions.

Under the same standard test function, the mean value represents the convergence accuracy of the algorithm, and the standard deviation represents the stability of the algorithm. Obviously, the smaller the mean value and standard deviation, the stronger the ability of the algorithm to avoid local solutions and determine the global optimal solution. It can be seen from Table 7 that the CSO algorithm and the WDO algorithm can successfully find an excellent solution on 9 test functions, among which the global optimal values are obtained on 5 functions, while the PSO algorithm and the SCA algorithm, MVO algorithm and TSA algorithm successfully find an excellent solution (global optimal value).

5 Application of chicken swarm optimization

The CSO algorithm and its improved algorithm continue to be used by academics in the fields of data mining, wireless sensor networks of the Internet of Things, robot engineering, electric power, feature extraction, and image processing to solve the optimal problem

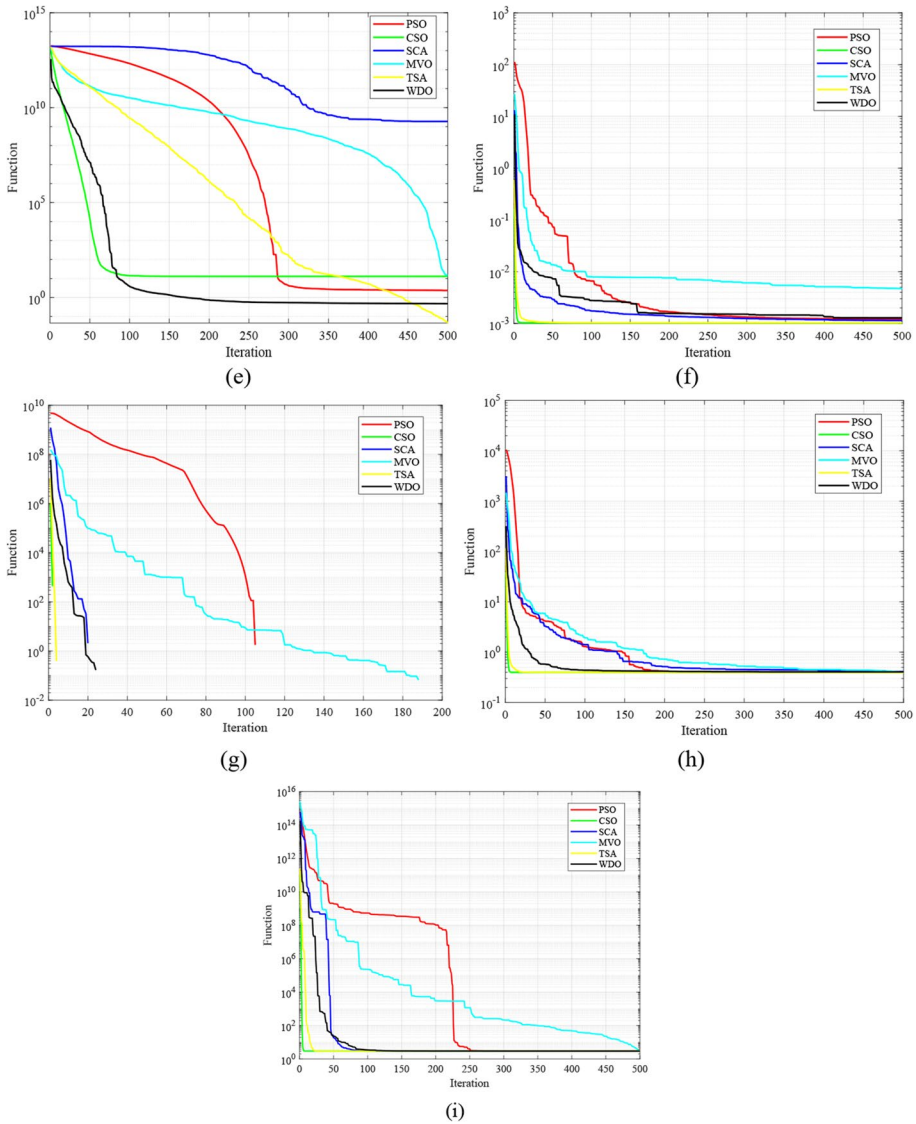


Fig. 10 (continued)

(Mirbabaie et al. 2021). This chapter will concentrate on the application, which is grouped in Tables 8, 9, 10, 11 and 12.

5.1 Data mining

Data mining is the process of using clever algorithms to extract hidden, undiscovered, and valuable information from a particular set of data (Liang et al. 2020b). Many academics are

Table 7 Comparison results of 9 function data

Function	Algorithm	Mean	Std	Best	Worst
F1	PSO	246.9311	17.5945	210.992	278.9645
	CSO	208.0782	18.3451	152.6312	248.3145
	SCA	304.9406	214.2728	58.6584	985.4387
	MVO	339.1023	48.2526	264.9664	453.3886
	TSA	183.6598	20.5057	125.5998	212.3145
	WDO	67.5927	35.3709	25.1827	164.2908
F2	PSO	21.1105	0.0504	21.0233	21.247
	CSO	20.3419	0.1149	20.329	20.2572
	SCA	20.2572	0.0674	20.1555	20.4232
	MVO	20.3941	0.0731	20.2760	20.5961
	TSA	20.2299	0.041	20.1129	20.3014
	WDO	20.3562	3.845	0.0003	21.1192
F3	PSO	0.5763	0.0553	0.496	0.6727
	CSO	0.2525	0.0497	0.2684	0.5473
	SCA	0.793	0.2235	0.2985	1.1984
	MVO	0.648	0.0831	0.4769	0.7757
	TSA	0.0067	0.0237	0	0.1253
	WDO	0.0051	0.0167	0	0.0785
F4	PSO	10.3703	5.6965	0.5267	25.8821
	CSO	11.0547	9.1334	0.1172	32.142
	SCA	0.935	2.93	0	10.633
	MVO	7.4444	2.8099	1.1123	14.3218
	TSA	0.3421	0.404	0.0043	1.5017
	WDO	0.0386	0.1093	0.0008	0.452
F5	PSO	2.3663	0.2674	1.901	3.1219
	CSO	11.0743	35.7516	0.5267	198.5413
	SCA	1.89	4.398	0	16.403
	MVO	15.1328	14.2393	2.482	62.1819
	TSA	0.0458	0.0782	0.0006	0.3453
	WDO	0.4779	0.8667	0.019	3.1323
F6	PSO	0.0012	0.0002	0.001	0.0018
	CSO	0.001	0	0.001	0.001
	SCA	0.0011	0.0002	0.001	0.0018
	MVO	0.0047	0.0091	0.001	0.0305
	TSA	0.001	0	0.001	0.001
	WDO	0.0013	0.0005	0.001	0.0027
F7	PSO	-1.0312	0.0004	-1.0316	-1.0302
	CSO	-1.0316	0	-1.0316	-1.0315
	SCA	-1.0316	0	-1.0316	-1.0316
	MVO	-1.0302	0.0015	-1.0316	-1.0259
	TSA	-1.0316	0	-1.0316	-1.0316
	WDO	-1.0316	0.0001	-1.0316	-1.0314

Table 7 (continued)

Function	Algorithm	Mean	Std	Best	Worst
F8	PSO	0.3984	0.0008	0.3979	0.4017
	CSO	0.3915	0	0.3987	0.398
	SCA	0.4099	0.025	0.398	0.5211
	MVO	0.3999	0.002	0.3979	0.4038
	TSA	0.3979	0	0.3979	0.3979
	WDO	0.406	0.0161	0.3979	0.4622
F9	PSO	3.025	0.0238	3.0018	3.0936
	CSO	3.0013	0.0028	3	3.0119
	SCA	3.0001	0.0001	3	3.0007
	MVO	3.1084	0.0852	3.0025	3.3420
	TSA	3	0	3	3
	WDO	3.0043	0.0082	3.0003	3.0464

Nine benchmark functions including F1 to F9 were selected for testing. Their average, standard deviation, optimal value and worst value test results are shown in Table 6

investigating the use of chicken swarm optimization (CSO) in the field of data mining since the quick and high-precision search requirements match the traits of CSO algorithms.

5.1.1 Data of classification

For the categorization of violent films, the author of a paper (Abdullahi et al. 2020) suggested combining the CSO method and a deep neural network (DNN). A DNN is a neural network made up of input, hidden, and output layers. It has numerous hidden layers (Gordan et al. 2022). Figure 11 depicts the neural network's structure. The input layer, which in this application instance corresponds to the input data of digital movies, is represented by the layer on the left. The model's characteristics have more expressive potential thanks to the multi-layer structure, which is buried between the intermediate levels. The categorization outcomes for violent videos are shown in this application's output layer. The expressive capability of the hidden layers determines how well classification results using DNN are produced. However, in real-world circumstances, depending simply on conventional techniques, such as different activation functions, frequently yields low accuracy and sluggish computing performance. As a result, in accordance with the literature (Abdullahi et al. 2020), the input data is optimized using the CSO algorithm at the input layer with mean squared error (MSE) serving as the fitness function. Prior to collecting features from the hidden layers, this method permits getting video data with improved precision.

In the literature (Kumari et al. 2022a), the author proposed the combination of SA-CSO with K-nearest neighbors (KNN) and Fuzzy-Convolutional neural networks (FCNN) for post-harvest grading of mangoes (Liu et al. 2022). Extracting fundamental information from fruit photos, such as the Gray-level co-occurrence matrix (GLCM), local binary pattern (LBP), discrete fourier transform (DFT), and shape characteristics, is the key to intelligent fruit grading. The data that was collected from the fruit comprises normal and aberrant segmentation characteristics. In the study, the feature vectors were optimized using the SA-CSO method, and KNN and CNN were used to model the categorization of mango

Table 8 Application of CSO in data mining

Author	Real Model	Algorithm	Application
Har (Abdullahi et al. 2020)	Classification of violent videos	CSO	Classification
Kumari (Kumari et al. 2022a)	Automated fruit grading	SA-CSO	Classification
Irsalinda (Harshavardhan et al. 2023)	Clustering framework	CSO	Clustering
Ahmed (Yanto et al. 2020)	Community detection	ACSO	Clustering
Wei (Rabani and Soleimani 2019)	Multi-objective optimization with deep learning	CSO	Clustering
C. Dhanusha (Wei et al. 2021)	Adaptive clustering for Alzheimer disease detection	CCSO	Clustering
Khan (Dhanusha et al. 2022)	Crude oil price prediction	CSO	Prediction
AbuKhalil (Khan et al. 2019)	Cervical cancer prediction	CSO	Prediction
Ayush (AbuKhalil et al. 2022)	Detection of cancer and virtual reality	CSO	Prediction
Xing (Rizk-Allah and Hassamien 2023)	Prediction interval estimation of landslide displacement	ACSO	Prediction

Table 9 Application of CSO in wireless sensor network

Author	Real Model	Algorithm	Application
Shayokh (Tripathi et al. 2020)	Node precision and computation time	CSO	Positioning
Yu (Al Shayokh and Shin 2017)	Deep mine monitoring	CSO	Positioning
Sandeli (Yu et al. 2019)	Radio Signal Strength	ICSO-I	Positioning
Gambhir (Sandeli et al. 2021)	Clustering routing protocol	CSO	Energy Management
Osamy (Osamy et al. 2020)	Problem of cluster head selection	CSOCA-GA	Energy Management
Ajmi (Gambhir et al. 2020)	Process of communication in the network	MWCSGA	Energy Management
Liang (Kong et al. 2020)	Collaborative beamforming	CSO-BA	Energy Management

Table 10 Application of CSO in wireless sensor network

Author	Real Model	Algorithm	Application
Mu (Ajmi et al. 2021)	Robotic manipulators	CSO	Trajectory
Li (Mu et al. 2016)	Ascent trajectory optimization problem	CSO	Trajectory
Wu (Wu et al. 2018a)	Reentry Trajectory Optimization	ICSO-II	Trajectory
Liang (Liang et al. 2020a)	Robot path planning	ICSO-III	Trajectory
Gawali (Gawali and Gawali 2021)	Movement of the robotic arm	CSO-DH	Machine Learning

defects and levels of ripeness. Testing was done using the Kesar Mango dataset from the Mendeley repository. The findings demonstrated that, in comparison to the employment of PSO, WOA, CSO, and other algorithms, the combination of SA-CSO with KNN and FCNN increased the accuracy of optimal feature extraction.

5.1.2 Data clustering

In the literature (Harshavardhan et al. 2023), the author proposed the utilization of the CSO algorithm in data clustering. The classification of comparable items into homogeneous groups is the aim of data clustering (Khorshid and Abdulazeez 2021). The Euclidean distance, which is defined as stated in Eq. (56), determines how similar two things are. The similarity metric was used in the study as the CSO algorithm’s optimization objective function. The Iris Dataset, Ecoli Dataset, Ionosphere Dataset, and Cancer Dataset were the four datasets used to assess the algorithm’s performance. The findings demonstrated that the CSO algorithm exhibited enhanced accuracy in data clustering compared to the GA, CS, and PSO algorithms.

$$ist(o_i, o_j) = \left(\sum_{p=1}^m |o_{ip} - o_{jp}|^{\frac{1}{2}} \right)^2 \tag{56}$$

wherein, in Eq. (56), m is the number of target attributes, and O_{ip} is the target value corresponding to the p -th attribute.

Table 11 Application of CSO in electrical engineering

Author	Real Model	Algorithm	Application
Awaiz (Li et al. 2017)	Minimization of power consumption	CSO	Smart Grids
Wang (Wang et al. 2020)	Interruptible load scheduling model	ICSO-IV	Smart Grids
Biswal (Awaiz et al. 2017)	The optimal allocation of capacitor banks	CSO	Smart Grids
Liu (Liu et al. 2020a)	Short-term photovoltaic power prediction	ICSO-V	New Energy Technology
Othman (Biswal and Shankar 2021)	PID controllers and required parameters of AVICs	CSO	New Energy Technology
Li (Mishra et al. 2022)	Identify the parameters of Photovoltaic models	SCCSO	New Energy Technology
Wu (Othman and El-Fergany 2021)	Maximum power point tracking in photovoltaic system	ICSO-IV	New Energy Technology
Deb (Deb and Gao 2021)	Charger placement problem	CSO-ALO	New-Energy Vehicle
Priyadarshi (Wu et al. 2018b)	Maximum power point trackers	CSO	New-Energy Vehicle
Sachan (Priyadarshi et al. 2021)	The planning and operation of charging stations	CSO	New-Energy Vehicle

Table 12 Application of CSO in feature extraction

Author	Real Model	Algorithm	Application
Verma (Sachan et al. 2021)	Feature Selection	CSO	Feature Extraction
Ahmed (Li et al. 2021b)	Benchmark models	CCSO	Feature Extraction
Liang (Verma et al. 2023)	Synthetic aperture radar image segmentation	CSO	Image Processing
Bharanidharan (Bharanidharan and Rajaguru 2020)	Categorize the magnetic resonance imaging (MRI) images	ICSO-VI	Image Processing
Cristin (Cristin et al. 2021)	Classification of Brain Tumor based on MRI Images	Fractional-CSO	Image Processing
Dhanusha (Wei et al. 2021)	Clustering for Alzheimer Disease Detection	CCSO	Image Processing
Sivanantham (Liang et al. 2018)	Brain Tumor Classification	CSO	Image Processing
Vamsidhar (Sivanantham et al. 2022)	Image enhancement	CSO	Image Processing
Kumari (Vamsidhar et al. 2022)	Two-dimensional image fast segmentation	CSO	Image Processing
Liang (Kumari et al. 2022b)	Image segmentation method	ICSO	Image Processing

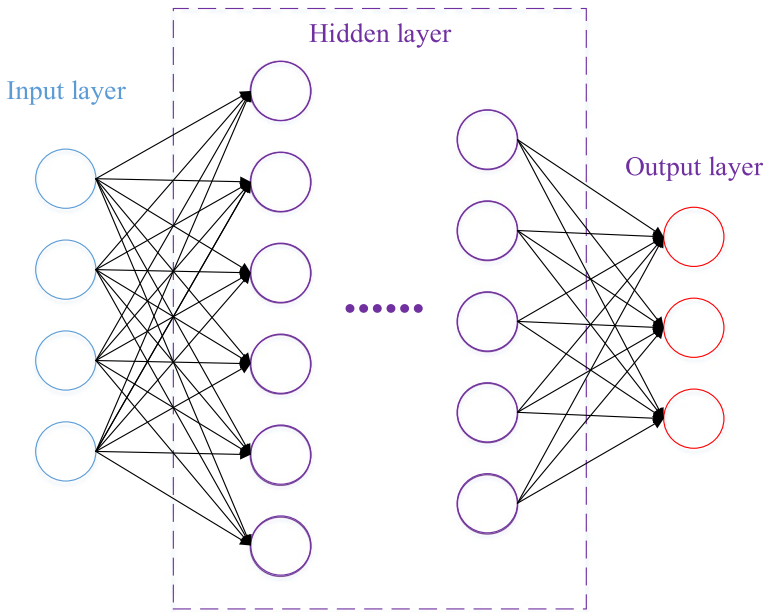


Fig. 11 Schematic diagram of DNN network

The author used the enhanced ACSO algorithm to tackle the challenge of community discovery in intricate social networks in the literature (Yanto et al. 2020). The main goal of community discovery is to redefine the topological structure of complex networks by grouping groups with similar characteristics (Guo et al. 2022). The ACSO method was modified in the literature (Yanto et al. 2020) to become a discrete swarm intelligence algorithm appropriate for clustering issues. As objective functions, we employ the Modularity, NMI, and Group truth of each category. Testing on data sets like the Zachary Karate Club data set and the American college football data set demonstrates that ACSO considerably increases clustering accuracy. The research also emphasizes that the CCSO method can be further enhanced in the future to accommodate application scenarios with a wider scope.

A multi-objective chicken swarm optimization method (MOCSO), which addresses the multi-objective issues faced by various neural network algorithms (Su et al. 2024), was suggested by the author in the literature (Rabani and Soleimani 2019). It is used with four different kinds of tri-objective functions and five different kinds of bi-objective functions. When compared to multi-objective tasks using algorithms like PSO, GA, and EA, MOCSO enhances convergence and optimum bounds according to the performance indicators of Generational distance (GD), Spacing, and Maximum Spread (MS).

In the literature (Wei et al. 2021), the author focused on the problem of the dataset in unsupervised clustering of Alzheimer's disease tending to fall into local optima in deep learning models (Garud et al. 2021). To fine-tune the parameters in deep learning clustering, it is suggested to employ the search concept of the CCSO method. It is determined by contrasting the FCM and IFCM clustering algorithms that initializing the parameters with CCSO may greatly increase the accuracy of the outcomes.

5.1.3 Prediction

The CSO method is used to initialize the weight values of a neural network algorithm in the prediction model of crude oil prices, according to the literature (Dhanusha et al. 2022). The neural network model is then given the optimum weights. It is tested to see if it is less than the halting condition using the mean square error (MSE) as a criterion (Alashwal et al. 2019). If so, the anticipated outcomes are outputted. Otherwise, the weights are continuing to be optimized using the CSO method. It is determined that CSO algorithm optimization delivers greater accuracy and lower MSE by contrasting it with ABCNN (Artificial Bee Colony Neural Network) and ABCBP (Artificial Bee Colony Back-Propagation).

The CSO algorithm is used in the feature extraction stage of predicting cervical cancer (Hodson 2022) in the literature (Khan et al. 2019) and (AbuKhalil et al. 2022). The CSO algorithm’s chickens include the feature values, which are then calculated iteratively to produce the chicken values. Following that, these values are compared to random integers between 0 and 1. The related characteristic is recorded as 1 if a value exceeds the random number; otherwise, it is marked as 0. CSO is discovered through simulation trials to increase the findings’ accuracy when compared to prediction algorithms like KNN, MLP, SVM, CART, and CNN.

In the literature (Rizk-Allah and Hassani 2023), the author addressed the issue of traditional point-based prediction algorithm, which is inability to accurately obtain landslide displacement (Akteer et al. 2021). Two support vector machines (SVM) are used in a proposed approach to produce upper and lower bounds for the displacement. By establishing a projected interval of landslide displacement, this optimization increases the point prediction algorithm’s accuracy of prediction. Given a dataset, SVM is examined using the PICP and PINAW parameters (Eqs. 58 and 60), which result in the comprehensive index CWC (Eq. 61), which is used to calculate the prediction interval (Eq. 57). Reference 82 also suggests using Eq. 61 as the ACSO algorithm’s goal function to enhance the prediction parameters. Using the ACSO method, as opposed to the CSO, SMO, and GWO algorithms, increases forecast accuracy, according to simulation trials.

$$\hat{I}^{(\alpha)}(x_i) = [\hat{L}^{(\alpha)}(x_i), \hat{U}^{(\alpha)}(x_i)] \tag{57}$$

wherein, in Eq. (57), $\hat{L}^{(\alpha)}(x_i)$ is the lower limit of the prediction, and $\hat{U}^{(\alpha)}(x_i)$ is the upper limit of the prediction.

$$PICP = \frac{1}{N} \sum_{i=1}^N \delta_i \tag{58}$$

wherein, in Eq. (58), δ is defined as shown in Eq. (59).

$$\delta_i = \begin{cases} 1, & \text{if } \hat{L}^{(\alpha)}(x_i) \leq y_i \leq \hat{U}^{(\alpha)}(x_i) \\ 0, & \text{otherwise} \end{cases} \tag{59}$$

$$PINAW = \frac{1}{NR} \sum_{i=1}^N (\hat{U}^{(\alpha)}(x_i) - \hat{L}^{(\alpha)}(x_i)) \tag{60}$$

wherein, in Eq. (60), R represents the normalized average width of $PICK$.

$$CWC = PINAW(1 + \gamma(PICP)e^{-\tau(PICP-\kappa)}) \quad (61)$$

wherein, in Eq. (61), τ is the control parameter of the Coverage Width Criterion (CWC). γ is defined as shown in Eq. (62).

$$\gamma(PICP) = \begin{cases} 1, & \text{if } PICP < \kappa \\ 0, & \text{otherwise} \end{cases} \quad (62)$$

5.2 Wireless sensor network (WSN)

In the spatial domain, sensors are distributed and form a network to perceive the surrounding environmental data in a unified and autonomous manner. Wireless Sensor Networks (WSN) (Liu et al. 2020b) are the name given to this technology. WSN has been widely used in various fields such as national defense (Majid et al. 2022), military (Wang and Zhu 2021), industrial monitoring (Pragadeswaran et al. 2021), environmental detection (Aponte-Luis et al. 2018), and smart living (Safaldin et al. 2021), due to its low cost, high scalability, and stability (Li et al. 2022b). However, due to the wide coverage of WSN and the limited energy of nodes, it is difficult to provide precise positioning devices similar to GPS for each node or replace the batteries of each node regularly (Elsmany et al. 2019). Numerous academics have used intelligent optimization algorithms on WSN to solve these problems, with positive outcomes.

5.2.1 Positioning

A small number of GPS sensors (anchor nodes) are used in real-world situations to get over GPS's limitations, while other sensors rely on localization algorithms that take use of the precise location supplied by GPS nodes (Han et al. 2022). According to the literature (Tripathi et al. 2020), the major focus for developing this technology should be on natural-inspired algorithms, with the decrease of computing time and localization mistakes as the primary goals. Equation (63), which accounts for the influence of the environment on wireless sensor networks, determines the separation between other nodes and anchor nodes:

$$\hat{d} = d_i + n_{noise} \quad (63)$$

wherein, in Eq. (63), d represents the distance between a node and the i -th anchor node. n_{noise} is the Gaussian noise affected by the environment.

The CSO optimization procedure, whose objective function is represented by Eq. (58), is first described in the literature (Tripathi et al. 2020) to determine the ideal distance.

$$f = \frac{1}{M} \sum_{i=1}^M (d_i - \hat{d}_i)^2 \quad (64)$$

wherein, in Eq. (64), M represents the number of anchor nodes within the specified area.

EL is set as the optimized error metric to evaluate the results of CSO optimization, and its definition is given in Eq. (65).

$$EL = \frac{\sum_{i=M+1}^N \sqrt{(x - x_i)^2 + (y - y_i)^2}}{NL} \tag{65}$$

wherein, in Eq. (65), NL is the number of unknown nodes. N is the total number of nodes. (x_i, y_i) is the final optimized positions of the target nodes, and (x, y) is the optimal position obtained by the optimization algorithm after each iteration.

The optimization results achieved using the CSO method enhanced the accuracy by 55% compared to the PSO and BPSO optimization algorithms, according to experiments and simulations reported in the literature (Tripathi et al. 2020). In terms of speed, the CSO algorithm decreased the computing time by 50%. The CSO optimization algorithm outperformed the BPSOA optimization algorithm by 10% in terms of accuracy while speeding up calculation by 30%.

The author of the literature (Al Shayokh and Shin 2017) largely concentrated on the usage of WSN in the field of deep mining. An extensive wireless sensor network is set up in deep mine shafts to monitor the environment (Gupta and Mahaur 2021). The received signal strength indication (RSSI) parameter (Muduli et al. 2018) determines the theoretical computed distance for the placement of each node in the network. The real placement distance is less than the estimated distance because of the complicated mining environment and the low connection distance between nodes. Within each node’s communication radius, there must be a minimum of three other nodes. Clustering the nodes in a WSN is a frequent practice to increase accuracy and reduce energy usage during transmission. After clustering, an optimization technique must be created to improve overall communication by setting anchor nodes inside each cluster (Nagah Amr et al. 2021).

The literature (Al Shayokh and Shin 2017) made use of the chicken swarm optimization’s unique properties, treating anchor nodes like individual birds and the rest m nodes like food in a 2D space. Equation (66) is used to determine the ideal food that the anchor nodes should seek. The Eqs. (67) and (68) define the objective function.

$$X = [(x_1, y_1), \dots, (x_m, y_m)] = [x_1, x_2, x_3, x_4, \dots, x_{2m-1}, x_{2m}] \tag{66}$$

$$f(x) = f(x_1, x_2, \dots, x_{2m}) = \frac{\sum_{k=1}^m \sum_{l=1}^m \left(\frac{\| (x_{2k-1}, x_{2k}) - (x_{2l-1}, x_{2l}) \| - D(k,l)}{D(k,l)} \right)}{m^2/E(k,l)} \tag{67}$$

$$E(k, l) = \begin{cases} 0, & D(k, l) = 0 \\ 1, & D(k, l) \neq 0 \end{cases} \tag{68}$$

wherein, in Eq. (67), D is a two-dimensional matrix containing distance data for all nodes.

The CSO algorithm is used to compute the objective function in order to produce the optimum parameters. The final optimal node locations are then produced using the Wheel graph technique utilizing these adjusted settings. The CSO-W localization optimization algorithm is the name given to this coupled method in the literature (Al Shayokh and Shin 2017). By comparing the accuracy performance of the CSO-W algorithm to that of the D3D-MAP and CSO algorithms, it is found that the CSO-W algorithm produces greater localization accuracy.

The objective function (64) is changed to Eq. (69) in the literature (Yu et al. 2019), and the error target EL is changed to Eq. (70). Additionally, ICSO-I, a more advanced CSO algorithm, is shown. Through simulation tests, it was discovered that the ICSO-I method outperforms the original CSO localization optimization and PSO algorithms in terms of localization accuracy.

$$f(x, y) = \frac{1}{M} \sum_{i=1}^M \left(\sqrt{(x - x_i)^2 + (y - y_i)^2} - d_i \right)^2 \quad (69)$$

$$E_L = \frac{1}{N_L} \sum_{i=1}^L ((x_i - \hat{x}_i)^2 + (y_i - \hat{y}_i)^2) \quad (70)$$

wherein, in Eqs. (69) and (70), (x, y) is the coordinates of anchor nodes, (x_i, y_i) is the coordinates of unknown nodes being optimized. L denotes the number of optimized nodes, and \hat{x}_i, \hat{y}_i is the optimal position obtained after each iteration of the node.

5.2.2 Energy management

The lifespan of a wireless sensor network (WSN) depends on the energy consumption of nodes in the system (Karim et al. 2021). In terms of application scenarios, the node with the longest lifespan that results in system failure determines the overall lifespan. This node could be a long-lifespan node that is infrequently used or a short-lifespan node that is often used (Abdulzahra and Al-Qurabat 2022). Optimization methods should be developed to reduce the usage frequency of short-lifespan nodes in order to prolong the lifespan of the WSN (Dalal et al. 2022). With a view to minimizing energy consumption and extending the lifespan of sensor nodes, LEACH is a wireless sensor network routing protocol that tackles the crucial problem of equally spreading network load across sensor nodes. A round of data, which consists of cluster formation and steady data transmission phases, is how LEACH divides the entire sensor network into periodic clusters (Angurala et al. 2022). A threshold random number determines how LEACH clusters develop, and this random formation has downsides because it causes node clustering and higher energy usage.

The CSO method is described in the literature (Sandeli et al. 2021) as a replacement for the LEACH routing protocol's random cluster generation procedure. The fitness function is the energy state of the nodes, and after a predetermined amount of iterations, the CSO solution is used to decide the cluster node composition. Through simulation studies, it is shown that the CSO algorithm outperforms the LEACH routing protocol in terms of fewer nodes experiencing energy exhaustion every round, extending the lifespan of the WSN system as a whole.

In the literature (Osamy et al. 2020), a WSN system with n nodes is constructed based on LEACH. Each node has the same initial energy and remains stationary. The energy consumption of nodes is defined as the transmission energy E_{Tx} and reception energy E_{Rx} , as specified in Eqs. (71) and (72):

$$E_{Tx}(b, d) = E_{elec} \times b + v \times b \times d^p \quad (71)$$

$$E_{Rx}(b, d) = E_{elec} \times b \quad (72)$$

wherein, in Eqs. (71) and (72), E_{elec} is the electronic energy. b is the packet size. d is the transmission distance. ν is the energy amplification factor due to distance, and p is a loss parameter ranging from 2.0 to 4.0.

Based on the transmission and reception energy, within a cluster, assuming there are m nodes, the energy loss functions for the cluster head node (CH) and other nodes (CM) can be defined as in Eqs. (73) and (74). The overall energy loss of the cluster, E , is given by Eq. (75). The optimization objective of the chicken swarm algorithm is to minimize energy loss, which is defined as the fitness function in Eq. (76):

$$e_{CH} = (m - 1)E_{Rx}(b) + mbE_{DA} + E_{Tx}(b, d_{ioBS}) \tag{73}$$

$$e_{CM} = E_{Tx}(b, d_{ioCH}) \tag{74}$$

$$E = e_{CH} + e_{CM} \tag{75}$$

$$F = \frac{\sum_{i=1}^k E(i)}{a + \sum_{i=1}^k E(i)} + \left(\frac{\beta}{a + \beta} \right) \tag{76}$$

wherein, in Eqs. (73), (74), and (75), d_{ioBS} and d_{ioCH} are the distances from nodes to the base station and to the CH node, respectively. k is the total number of CH nodes, β is the total number of selected CH nodes, and a is a positive constant.

The CSOCA-GA algorithm is suggested for usage for optimization based on the fitness function in the literature (Osamy et al. 2020), and simulation tests are carried out to compare it with CSO, EDOC, GCDC, and LEACHPSO (LEACH optimized using PSO).The results show that CSOCA-GA reduces energy loss per round compared to the other algorithms.

In the literature (Gambhir et al. 2020), a multi-objective optimization algorithm called MWCSGA (Multi-Weight Chicken Swarm-based Genetic Algorithm) is proposed, building upon the CSOCA-GA algorithm, to account for the surrounding environment of the sensor network. The weight parameter ω is introduced in the objective function F , as defined in Eq. (77). Through experimental simulations, MWCSGA is shown to improve energy efficiency and reduce loss compared to CSOCA-GA and the optimized LEACH algorithm.

$$F = \sum_i (\omega_i * f_i) \tag{77}$$

wherein, in Eq. (77), f_i is the objective function based on energy loss, and ω_i is the weight parameter of each node, which is related to the distance of the node.

In the literature (Kong et al. 2020), CSO-BA algorithm is introduced to reduce the lower peak side lobe level (PSL) in antenna arrays of sensor networks. The study also comprehensively compares the optimization results of the CSO-BA algorithm with CSO, BA, PSO, and BBO algorithms. It concludes that CSO-BA improves the convergence speed and accuracy of the algorithm compared to other optimization algorithms, showing promising prospects for practical applications.

5.3 Robotics engineering (RE)

Robotics engineering is a multidisciplinary field that focuses on intelligent manufacturing and intelligent robots (Daanoun et al. 2021). It integrates various cutting-edge disciplines such as computer science, optoelectronics (Wang et al. 2021b), automatic control (Pham et al. 2022), sensors, and bionics (Berberich et al. 2022). With the in-depth research on optimization algorithms, corresponding research achievements have been made in the areas of robot path planning and artificial intelligence machine learning.

5.3.1 Trajectory

In the literature (Ajmi et al. 2021), the author applied the chicken swarm optimization algorithm to the optimization of robot motion trajectories. The objective function of the method is chosen to be the least trip time, and a third-order B-spline curve is used to build the motion trajectory. The trials demonstrated that the chicken swarm optimization algorithm successfully decreased the trip time when utilizing a six-degree-of-freedom robotic arm to polish metal workpieces.

In the literature (Mu et al. 2016), the author studied the optimization problem of the ascent trajectory of a generic hypersonic vehicle (GHV). This analysis included limitations including dynamic pressure, load factor, and aerodynamic heating to guarantee the flight safety and structural integrity of the GHV. The GHV's fuel consumption was selected as the goal function, and the optimization process used the chicken swarm technique. The optimization impacts of the CSO algorithm and the PSO algorithm were compared through experimental simulations. The outcomes demonstrated that CSO considerably increased the algorithm's convergence speed and accuracy.

In the literature (Wei et al. 2021), the author addressed the shortcomings of the CSO algorithm in handling high-dimensional problems and proposed the ICSO-II algorithm based on the crossover operator. It was applied to the reentry trajectory problem of the GHV, which has high-dimensional optimization requirements. The least aerodynamic heating rate and the shortest flight time were chosen as the optimization process's objective functions. Through simulation studies, the optimization performance of five algorithms—ICSO-II, CSO, PSO, ABC, and ACO—was examined and compared. The outcomes showed that ICSO-II outperformed the other algorithms by a wide margin.

In the literature (Liang et al. 2018), the author addressed the issue of the CSO algorithm easily converging to local optima and proposed the ICSO-III algorithm for robot path planning. The goal function used by the optimization process, as given in Eq. (78) is the length of the robot's journey. Finally, it was determined through a comparison of the ICSO-III, CSO, and PSO results that, in comparison to the other two algorithms, the ICSO-III method improved algorithm convergence accuracy and stability.

$$f = \sum_{i=1}^{n-1} \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (78)$$

wherein, in Eq. (78), n is the number of nodes in the path, and (x_i, y_i) is the horizontal and vertical coordinates of the robot.

5.3.2 Machine learning

In the literature (Gawali and Gawali 2021), the author focused on the popular machine learning problem in recent years, within the context of reinforcement learning, the CSO-DH optimization algorithm is proposed to optimize the robot's arm motion trajectory as the objective function. The results of simulation experiments show that the CSO-DH optimization-based reinforcement learning algorithm is more accurate than traditional reinforcement learning algorithms, WOA-based reinforcement learning algorithms, and DHOA-based reinforcement learning algorithms, further bridging the gap between humans and robots.

5.4 Electrical engineering (EE)

Electrical engineering plays a crucial role in the energy field today. In recent years, with the introduction of intelligent optimization algorithms, significant progress has been made in smart grids (Tan et al. 2022), new energy technologies (Lamnatou et al. 2022), and new energy vehicles (Pan and Dong 2023).

5.4.1 Smart grids

In the literature (Li et al. 2017), the author aimed to reduce electricity costs. CSO algorithm is utilized for optimization based on the peak-shaving and valley-filling technology in smart grids. The electrical load and Critical Peak Pricing (CPP) are selected as the objective functions. Experimental models lead to the conclusion that CSO successfully lowers its electricity costs.

In the literature (Sivanantham et al. 2022), to achieve balanced electricity consumption during peak periods, a method for regulating interruptible loads to reduce peak loads is proposed. A load shedding scheduling model is designed considering the users' subsidy rate. The model aims to minimize the overall power consumption, defined as the objective function in Eq. (79). Finally, an improved algorithm called ICSO-IV is designed to optimize the mixed nonlinear problem. Through simulation experiments, it is observed that ICSO-IV outperforms the GA and PSO optimization algorithms in terms of convergence speed and accuracy of the optimization model.

$$\min P = P_1 + P_2 + P_3 \quad (79)$$

wherein, in Eq. (79), P is the overall power consumption. P_1 is the power consumption of interruptible loads. P_2 is the power consumption of compensating loads. and P_3 is the power consumption of smart grid operation and maintenance.

In the literature (Awais et al. 2017), parallel capacitor banks are introduced into radial distribution systems. The system's bus voltage and power factor are both improved by the capacitor banks' optimized arrangement. The CSO algorithm is created to optimize the power factor on the 85th and 118th bus systems with the goal of reaching the ideal arrangement and size of the parallel capacitor banks, hoping to lower the system's power consumption. The experimental findings show that the

CSO-achieved optimized capacitor bank arrangement successfully lowers costs while providing dependable power factor performance.

5.4.2 New energy technology

In the literature (Liu et al. 2020a), the author proposed that photovoltaic (PV) power generation is influenced by weather conditions, and its impact on the power grid depends on the short-term power prediction accuracy under different weather conditions. The paper enhances the CSO algorithm first, then suggests the ICSO-V method to increase prediction accuracy. The ICSO-V technique is then used in conjunction with an extreme learning machine model to forecast PV power under various weather scenarios. The test results show that the ICSO-V algorithm and the extreme learning machine model together minimize mean square error and percentage error.

In the literature (Biswal and Shankar 2021), the author aimed to enhance the frequency stability of interconnected renewable energy power systems. It proposes an optimization control scheme that combines CSO with Adaptive Virtual Inertia Control (AVIC). The scheme utilizes CSO to find the optimal values of the gain of the adaptive PID controller and the parameters required by AVICs. Experimental simulations are conducted to compare the superiority of the CSO algorithm with differential evolution and PSO algorithms. The results show that the control scheme based on CSO-AVIC significantly improves the dynamic performance of the system.

In the literature (Mishra et al. 2022), the author addressed the efficiency of converting solar energy to electrical energy in PV systems, which is influenced by the parameters of the PV model. The PV model's parameter identification problem is seen as an optimization problem. To minimize the difference between experimental data and simulated data, error functions Error for single diode and double diode are defined. The objective function RMSE, as shown in Eq. (80), is then defined based on the errors. Finally, function optimization using the Spiral-based Chaotic Chicken Swarm Optimization (SCCSO) technique is used to determine the PV model's ideal parameters. Experimental simulations indicate that the SCCSO algorithm improves the convergence accuracy and stability compared to the CSO and PSO algorithms.

$$RMSE(X) = \sqrt{\frac{1}{N_M} \sum_{i=1}^N F_{error}^i(V_L, I_L, X)^2} \quad (80)$$

wherein, in Eq. (80), X represents the PV parameter solution vector to be solved, N_m is the number of I - V data obtained from measurements, V_L is the output voltage at the port, and I_L is the output current at the port.

In the literature (Othman and El-Fergany 2021), the author addressed the issue of long optimization time in solving high-dimensional optimization problems using the CSO algorithm. It proposes a modified version of the CSO algorithm called ICSO-IV, which incorporates initial sorting based on chaotic sequences and introduces adaptive weight optimization. The algorithm is then applied to the maximum power point optimization tracking problem in photovoltaic systems. Through simulation experiments, it is demonstrated that this optimization algorithm improves the convergence speed and accuracy compared to the CSO, PSO, and BA algorithms.

5.4.3 New-energy vehicle

In the literature (Deb and Gao 2021), the author focused on the optimal placement problem of electric vehicles (EVs) and considers it as a complex high-dimensional optimization problem. The study confirms the CSO algorithm's limits in tackling high-dimensional optimization issues and suggests CSO-ALO, a hybrid enhanced optimization method that combines the CSO algorithm and the ALO algorithm. Through benchmark functions and tests on EV charger placement problems, it is demonstrated that CSO-ALO outperforms CSO, ALO, TLBO, and other optimization algorithms, exhibiting higher convergence accuracy in solving placement problems.

In the literature (Wu et al. 2018b), the maximum power point tracking (MPPT) problem for fuel cells in electric vehicles is addressed. Through simulation studies, it is demonstrated that a CSO-based MPPT algorithm may more accurately reach the ideal fuel cell power.

In the literature (Priyadarshi et al. 2021), the author addressed the need for intelligent energy management algorithms to meet the new requirements for the number of charging stations in parking lots and the capacity of the power grid with the large-scale production of new energy vehicles. The study proposes a charging station planning and operation solution based on the CSO algorithm. The objective functions include reliability, voltage stability characteristics, and power consumption. According to experimental simulations, the CSO-based approach is superior than coordinated charging, non-coordinated charging, and vehicle-to-grid (V2G) solutions in the aforementioned metrics. Future issues with new energy car charging are well addressed by this.

5.5 Feature extraction

With the development of artificial intelligence technology, feature extraction from given datasets and analysis and processing of given graphics are now common research topics.

5.5.1 Feature extraction

In the literature (Sachan et al. 2021), a CSO-based data packet pattern feature extraction method is proposed. This method aims to find the optimal point of the feature fitness function in a multidimensional feature space. It minimizes the selected number of features while ensuring optimal classification performance. Finally, practical simulations show that the CSO-based feature extraction method enhances the accuracy of fitness function optimization compared to methods based on PSO and GA utilizing 18 datasets from the UCI repository as test data. It achieves superior fitness function values.

In the literature (Li et al. 2021b), the author addressed the problem of the CSO algorithm easily getting trapped in local optima when performing dimensionality reduction in feature extraction. It proposes an improved clustering algorithm based on chaos called CCSO. The effectiveness of CCSO is compared to CSO, PSO, DOA, and BAT in feature dimensionality reduction using five datasets (spambase, wdbc, ionosphere, lung, sonar). Through experimental simulations, it is concluded that the CCSO algorithm significantly improves the accuracy of feature extraction.

5.5.2 Image processing

In the literature (Verma et al. 2023), a fast segmentation method for Synthetic Aperture Radar (SAR) images is proposed to address the issues of poor segmentation performance and slow segmentation speed in traditional SAR image processing (Cai et al. 2021b). The method first narrows down the search space of the swarm based on the characteristics of SAR images. Then, it swiftly locates the ideal solution by using the gray entropy model as the algorithm's fitness function. The improved search space chicken swarm optimization technique, as compared to genetic algorithms and artificial fish swarm algorithms, enhances the accuracy and speed of picture segmentation, according to experimental simulations.

In the literature (Bharanidharan and Rajaguru 2020), the author focused on the problem of low convergence accuracy and slow convergence speed when applying the CSO algorithm to medical image processing (Mondini et al. 2021). It proposes a controlled CSO algorithm called ICSO-VI, which improves the algorithm by setting control parameters. The classification of MRI images into DEM type and ND type is the focus of the research, and both statistical features and no statistical features are used in all trials. Finally, it is discovered that the ICSO-VI algorithm greatly increases the accuracy of image classification compared to the CSO method based on the examination of 65 ND and 52 DEM-type real brain MRI images.

In the literature (Cristin et al. 2021), the author combined the behavior patterns of the chicken swarm with derivative factors to enhance the accuracy of the swarm's hierarchical classification. The fitness function's ideal solution is discovered by repeatedly comparing the positions of roosters; this enhanced method is known as fractional-CSO. Finally, it is determined that Fractional-CSO greatly enhances accuracy, specificity, and sensitivity in image processing compared to the CSO algorithm by preprocessing brain pictures and utilizing the accuracy, specificity, and sensitivity of diagnosing cancer as experimental indicators.

In the literature (Wu et al. 2018a), the author proposed that predicting models for Alzheimer's disease often require preprocessing and important feature extraction. However, using preprocessed models for training often performs poorly in clinical scenarios. Therefore, in reference (Radaideh and Shirvan 2021), an unsupervised deep learning model is introduced, and a chaos theory-based improved chicken swarm optimization algorithm (CCSO) is employed to optimize the parameters of deep clustering in unsupervised learning (Li et al. 2023). This overcomes the problem of unsupervised clustering parameters easily getting trapped in local optima. Finally, it is determined through experimental simulations that the CCSO enhanced clustering model, when compared to two other clustering models, FCM and IFCM, enhances the accuracy of the method.

In the literature (Liang et al. 2020a), the author applied the CSO algorithm to brain tumor image classification and proposes a hybrid classification algorithm combining neural networks and CSO, which is also used in feature selection and dimensionality reduction. The introduction of the CSO algorithm helps obtain more realistic classification results. The hybrid classification method of neural networks and CSO is found to increase the accuracy of classification results when compared to classic neural network algorithms, decision tree algorithms, and SVM algorithms through validation testing with current brain tumor picture data.

In the literature (Wang et al. 2020), the author addressed the problem of reduced image processing quality in image enhancement techniques due to data loss. It proposes

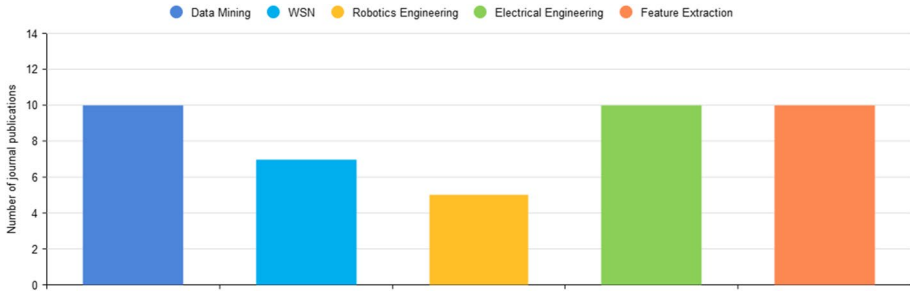


Fig. 12 Core application of CSO algorithm

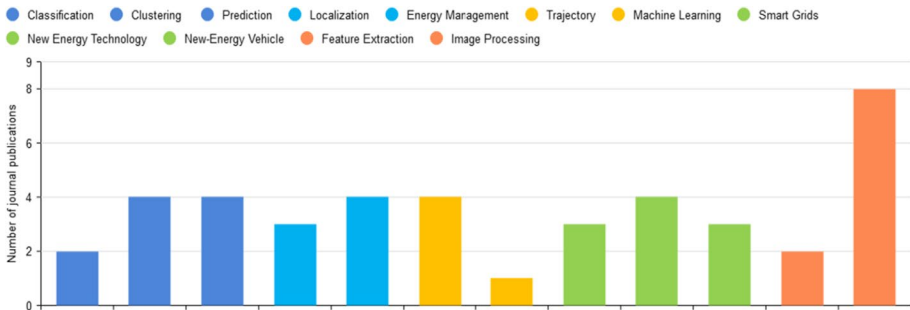


Fig. 13 Specific application of CSO algorithm

a CSO-based large-scale optimization algorithm for image enhancement, where the algorithm optimizes the entropy and peak signal-to-noise ratio of the enhanced image. Finally, through comparative experimental simulations, it is concluded that the CSO-based image enhancement algorithm improves the entropy value and peak signal-to-noise ratio of the enhanced image compared to the traditional histogram equalization (HE) method.

In the literature (Vamsidhar et al. 2022), the author addressed the problem of slow computation and large computational complexity in two-dimensional maximum entropy segmentation of images (Schmarje et al. 2021). It proposes an improved two-dimensional maximum entropy segmentation algorithm based on CSO optimization. In this algorithm, the maximum entropy of the two-dimensional image is used as the fitness function of the optimization algorithm. The chicken swarm algorithm quickly determines the ideal threshold of the objective function to get the ideal segmentation solution through the optimization process. The segmentation method based on CSO optimization, as opposed to particle swarm optimization (PSO) and artificial fish swarm algorithm, enhances the convergence performance and computing speed of image segmentation, according to experimental simulations.

In the literature (Kumari et al. 2022b), the author focused on the parameter setting problem of pulse-coupled neural networks (PCNN) in the field of image segmentation (Wunava et al. 2022). It proposes an improved image segmentation algorithm called ICSO-ISPCNN, which combines an improved chicken swarm optimization algorithm with an improved PCNN model. In this algorithm, the threshold function in the PCNN model is

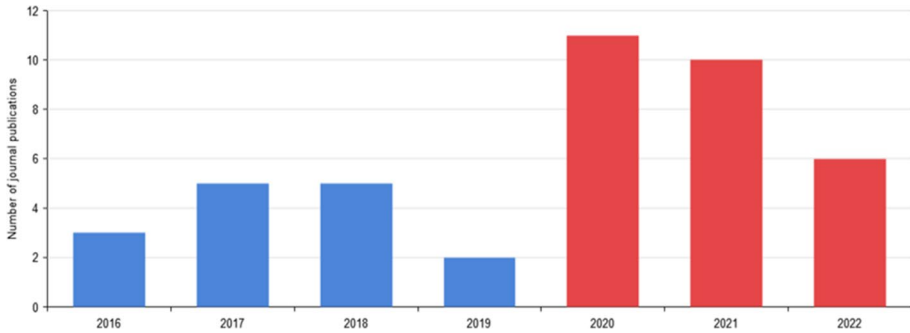


Fig. 14 CSO algorithm research progress

modified, and the CSO algorithm is enhanced with a tournament selection mechanism. The pulse-coupled neural network approach for automated image segmentation iteratively optimizes the parameter values using the integrated cross-entropy as the fitness function. Through experimental simulations, it is discovered that ICSO-ISPCNN outperforms genetic algorithms and ant colony optimization algorithms in terms of convergence performance and segmentation accuracy.

6 Discussion and the key problems

This study systematically explores the CSO algorithm's past research findings in light of the history of optimization algorithms and its guiding principles. Figures 12 and 13 indicate the number of magazines in the five core subjects and the number of publications in the twelve distinct study topics. The CSO algorithm has been incorporated into a number of sectors and has proven effective in realizing the best answer to the relevant engineering challenges. Its convergence accuracy and speed have significantly improved as compared to conventional approaches. The number of scholarly journals in various disciplines from 2016 to 2022 is depicted in Fig. 14. It can be concluded that since 2020, the application of CSO algorithm has increased significantly, and its main research content is concentrated in the fields of new energy and medical image processing. The CSO algorithm has demonstrated beneficial universal applicability when taken into account with the application situation since the algorithm was introduced, and it may be used in new application sectors that are offered in the future when taken into account with societal demands. Researchers in robotics don't have any clear ideas for combining the CSO algorithm with this field based on the statistical data. Technological progress has led to the emergence of industrial robots, medical robots, and autonomous driving robots, which will soon dominate research in their respective domains. The application of this study subject has not incorporated the CSO algorithm.

Compared with most intelligent optimization algorithms, CSO algorithm has been proved to have the advantages of high convergence accuracy and fast convergence speed in dealing with engineering optimization problems. Many enhanced CSO algorithms have been developed to increase the accuracy of the algorithm since, in the later stages of processing the issue, the algorithm has a tendency to settle on a local optimum solution. These include enhancing the hen, chick, and rooster operators' trajectories as well

as the hybrid algorithm. There is no universal improvement method, and the improved CSO algorithm cannot be applied to all application scenarios from the perspective of optimization effect, and the applicable conditions and scope after the improvement still need to be further explored by simulation experiments.

Since the CSO algorithm was proposed, it has achieved results for different expected goals. The combination of different improved algorithm results will make the improved algorithm more targeted and the applicable objects more accurate. The combination of algorithm results in different application fields makes the algorithm collide with sparks in multiple cross fields, obtain better engineering optimization results, and at the same time promote the birth and development of new fields after cross fusion. The relevant work of the future CSO is as follows:

- **More theoretical calculation comparison:** The CSO algorithm has good performance results in solving theoretical optimization problems with the benchmark function as the objective function. However, the birth of the new bionic algorithm is bound to be in contrast with the CSO algorithm. For example, Agushaka proposed the Gazelle Optimization Algorithm (GOA) in 2023 (Minaee et al. 2021). This algorithm simulates the survivability of antelopes under the rule of predators, and divides the algorithm into two stages: development and exploration. The problem-solving ability and superior competitiveness of the GOA algorithm are proved by the benchmark function and a number of engineering design problems. CSO and its improved algorithm can be compared with GOA and other algorithms in theoretical calculation under the same objective function and target task conditions (Agushaka et al. 2023). So as to obtain their respective application advantages and applicable problem ranges.
- **More cross-blending improved contrast:** The optimization algorithm after cross-blending tends to get higher convergence accuracy. A large number of improved algorithms can be obtained by cross-mixing the CSO algorithm and the new optimization algorithm (Zhang et al. 2021). This section of the study material can be completed as a research review or with the creation of a brand-new hybrid algorithm. The circumstances under which the hybrid algorithm and the newly developed hybrid algorithm are applicable are determined by contrasting the benefits and drawbacks of the algorithms in various application scenarios.
- **Explore more application fields:** This article describes the application of the CSO algorithm in the five major fields of Data Mining, WSN, Robotics Engineering, Electrical Engineering, and Feature Extraction. The future research fields of CSO may include brain-computer interface (Tang et al. 2023), integrated circuit low-power design (Ye et al. 2022), artificial intelligence chip (Choi et al. 2022), driverless. The research content can be aimed at the breakthrough of a single problem, such as power consumption, signal acquisition integrity, path tracking strategy, or comprehensive research on multiple engineering problems in the corresponding field.

Chicken swarm optimization algorithm is developing rapidly, and most of its research is on the improvement of the application level, mainly focusing on improving the population coding method, combining with other strategies and ideas to form a hybrid algorithm, and applying it to specific problems. But up to now, the theoretical basis of chicken swarm optimization algorithm is still not perfect. Theoretical analysis plays an important role in our in-depth understanding of the mechanism of the algorithm, and the convergence and stability analysis of the algorithm is helpful for its further development. Moreover, it is

more convincing and academically valuable to evaluate the performance of the algorithm from the perspective of theoretical analysis.

7 Conclusions and future work

While the chicken swarm optimization method has several benefits over other bionic intelligence algorithms in terms of addressing difficult optimization issues, it also reveals numerous drawbacks when it comes to actually solving problems. Therefore, the following characteristics can be used to conduct further study on the chicken flock optimization method.

- (1) The performance of the chicken swarm optimization technique is easily influenced by the choice of parameter values. How to reduce the unsatisfactory parameter value setting and how to impact the performance of the wolf pack algorithm by setting the size of each parameter value appropriately or adaptively according to different challenges.
- (2) The late convergence speed decreased significantly, which was attributed to the low-efficiency search behavior of the flock. The search intensity reflects the local search ability of the flock. How to balance or strengthen the global search ability and local search ability of the flock algorithm to further improve the flock The search efficiency of the group optimization algorithm will be one of the focuses of future research.
- (3) As a bionic intelligent algorithm, the chicken swarm optimization algorithm has clear biological and social traits and only a flimsy mathematical foundation, necessitating a thorough theoretical investigation and mathematical justification.
- (4) Chicken swarm optimization algorithm is an algorithm produced by multidisciplinary fusion and crossover. In future research, new improved algorithms can be designed in combination with theories in other disciplines.
- (5) It is essential to improve search speed while reducing time complexity while building the chicken swarm optimization method in the future to ensure that it is applicable to all problems.

As a relatively new nature-inspired swarm intelligence optimization algorithm, chicken group optimization algorithm has the advantages of strong global and local search ability, high population diversity and strong robustness, so it is widely used in engineering Practice and solve real life problems. First, a thorough explanation of the flock optimization algorithm's fundamental principles is provided. Next, a summary of the algorithm's benefits and drawbacks, as well as its improvement strategies, is provided. Finally, a prediction is made regarding the flock optimization algorithm's future research. The flock optimization approach has only been suggested for a little over eight years. The algorithm itself has been continuously improved and developed. Its theoretical research and engineering application have made great progress, and the theoretical research is gradually becoming mature.

Practical applications: (1) At present, the performance verification of most improved algorithms is based on the benchmark function proof. This verification method is too simple, and the performance of the algorithm can be verified in specific engineering problems in the future. (2) The flock optimization algorithm is widely used in combinatorial optimization problems, and its application in other fields needs to be expanded urgently, such as nonlinear, discrete and large-scale integration problems.

Acknowledgements Binhe Chen and Li Cao contributed equally to this work and should be considered as co-first authors.

Authors contribution Binhe Chen and Li Cao analyzed the data and wrote the paper. Changzu Chen and Yaodan Chen revised the manuscript. Yinggao Yue designed the study.

Funding This work was supported in part by the Natural Science Foundation of Zhejiang Province under Grant LY23F010002, the Industrial Science and Technology Project of Yueqing City under Grant 2022G007, and Wenzhou Association for Science and Technology under Grant kjfw36, the school-level scientific research projects of Wenzhou University of Technology under Grants ky202201 and ky202209, the general scientific research projects of the Provincial Department of Education under Grant Y202250103, and the Wenzhou intelligent image processing and analysis key laboratory construction project under Grant 2021HZSY007105.

Declarations

Competing interests The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

- Abdullahi M, Ngadi MA, Dishing SI et al (2020) A survey of symbiotic organisms search algorithms and applications. *Neural Comput Appl* 32:547–566
- Abdulzahra AMK, Al-Qurabat AKM (2022) A clustering approach based on fuzzy C-means in wireless sensor networks for IoT applications. *Karbala Int J Mod Sci* 8(4):579–595
- Abualigah L (2020) Multi-verse optimizer algorithm: a comprehensive survey of its results, variants, and applications. *Neural Comput Appl* 32(16):12381–12401
- Abualigah L (2021) Group search optimizer: a nature-inspired meta-heuristic optimization algorithm with its results, variants, and applications. *Neural Comput Appl* 33(7):2949–2972
- Abualigah L, Shehab M, Alshinwan M et al (2020) Salp swarm algorithm: a comprehensive survey. *Neural Comput Appl* 32:11195–11215
- Abualigah L, Shehab M, Alshinwan M et al (2021) Ant lion optimizer: a comprehensive survey of its variants and applications. *Arch Comput Methods Eng* 28:1397–1416
- AbuKhalil T, Alqaralleh BAY, Al-Omari AH (2022) Optimal Deep Learning Based Inception Model for Cervical Cancer Diagnosis. *Comput Mater Contin* 72:57–71
- Aghelpour P, Mohammadi B, Mehdizadeh S et al (2021) A novel hybrid dragonfly optimization algorithm for agricultural drought prediction. *Stoch Env Res Risk Assess* 35(12):2459–2477
- Aguila-Leon J, Vargas-Salgado C, Chiñas-Palacios C et al (2023) Solar photovoltaic Maximum Power Point Tracking controller optimization using Grey Wolf Optimizer: A performance comparison between bio-inspired and traditional algorithms. *Expert Syst Appl* 211:118700
- Agushaka JO, Ezugwu AE, Abualigah L (2023) Gazelle Optimization Algorithm: A novel nature-inspired metaheuristic optimizer. *Neural Comput Appl* 35(5):4099–4131
- Ahamad D, Hameed SA, Akhtar M (2022) A multi-objective privacy preservation model for cloud security using hybrid Jaya-based shark smell optimization. *J King Saud Univ-Comput Inf Sci* 34(6):2343–2358
- Ajmi N, Helali A, Lorenz P et al (2021) MWCSGA—multi weight chicken swarm based genetic algorithm for energy efficient clustered wireless sensor network. *Sensors* 21(3):791
- Akter L, Islam MM, Al-Rakhani MS et al (2021) Prediction of cervical cancer from behavior risk using machine learning techniques. *SN Comput Sci* 2:1–10

- Al Shayokh M, Shin SY (2017) Bio inspired distributed WSN localization based on chicken swarm optimization. *Wireless Pers Commun* 97(4):5691–5706
- Alashwal H, El Halaby M, Crouse JJ et al (2019) The application of unsupervised clustering methods to Alzheimer's disease. *Front Comput Neurosci* 13:31
- Angurala M, Singh H, Anupriya, Grover A, Singh M (2022) Testing Solar-MAODV energy efficient model on various modulation techniques in wireless sensor and optical networks. *Wirel Netw* 28(1):413–425
- Aponte-Luis J, Gómez-Galán JA, Gómez-Bravo F et al (2018) An efficient wireless sensor network for industrial monitoring and control. *Sensors* 18(1):182
- Asna M, Shareef H, Muhammad MA et al (2022) Multi-objective quantum atom search optimization algorithm for electric vehicle charging station planning. *Int J Energy Res* 46(12):17308–17331
- Awais M, Abadeen ZU, Bilal T, Faiz Z, Junaid M, Javaid N (2018) Home energy management using enhanced differential evolution and chicken swarm optimization techniques. In *Advances in Intelligent Networking and Collaborative Systems: The 9th International Conference on Intelligent Networking and Collaborative Systems (INCoS-2017)*. Springer International Publishing, pp 468–478
- Ayvaz A (2022) An improved chicken swarm optimization algorithm for extracting the optimal parameters of proton exchange membrane fuel cells. *Int J Energy Res* 46(11):15081–15098
- Babazadeh M, Rezaifar O, Jahani E (2023) Interval reliability sensitivity analysis using Monte Carlo simulation and mouth brooding fish algorithm (MBF). *Appl Soft Comput* 142:110316
- Bai Y, Cao L, Chen B, Chen Y, Yue Y (2023) A Novel Topology Optimization Protocol Based on an Improved Crow Search Algorithm for the Perception Layer of the Internet of Things. *Biomimetics* 8(2):165
- Bai Y, Cao L, Wang S, Ding H, Yue Y (2022) Data collection strategy based on OSELM and gray wolf optimization algorithm for wireless sensor networks. *Computational Intelligence and Neuroscience* 2022
- Bangyal WH, Hameed A, Ahmad J et al (2022) New modified controlled bat algorithm for numerical optimization problem. *Comput Mater Contin* 70(2):2241–2259
- Basha AJ, Aswini S, Aarthini S, Nam Y, Abouhawwash M (2023) Genetic-chicken swarm algorithm for minimizing energy in wireless sensor network. *Computer Systems Science & Engineering* 44(2)
- Batra J, Jain R, Tikkiwal VA et al (2021) A comprehensive study of spam detection in e-mails using bio-inspired optimization techniques. *Int J Inf Manag Data Insights* 1(1):100006
- Berberich J, Scherer CW, Allgöwer F (2022) Combining prior knowledge and data for robust controller design. *IEEE Transactions on Automatic Control*
- Bharanidharan N, Rajaguru H (2020) Improved chicken swarm optimization to classify dementia MRI images using a novel controlled randomness optimization algorithm. *Int J Imaging Syst Technol* 30(3):605–620
- Biswal SR, Shankar G (2021) Optimal capacitor placement in radial distribution system using chicken swarm optimization algorithm. In *Advances in Smart Grid Automation and Industry 4.0: Select Proceedings of ICETSGAI4*. 0. Springer Singapore, pp 113–120
- Braik M, Sheta A, Al-Hiary H (2021) A novel meta-heuristic search algorithm for solving optimization problems: capuchin search algorithm. *Neural Comput Appl* 33:2515–2547
- Braik M, Hammouri A, Atwan J et al (2022) White Shark Optimizer: A novel bio-inspired meta-heuristic algorithm for global optimization problems. *Knowl-Based Syst* 243:108457
- Cai X, Zhao H, Shang S et al (2021a) An improved quantum-inspired cooperative co-evolution algorithm with multi-strategy and its application. *Expert Syst Appl* 171:114629
- Cai W, Wu X, Zhou M et al (2021b) Review and development of electric motor systems and electric powertrains for new energy vehicles. *Automot Innov* 4:3–22
- Cao L, Chen H, Chen Y, Yue Y, Zhang X (2023) Bio-Inspired Swarm Intelligence Optimization Algorithm-Aided Hybrid TDOA/AOA-Based Localization. *Biomimetics* 8(2):186
- Cao L, Wang Z, Yue Y (2022) Analysis and prospect of the application of wireless sensor networks in ubiquitous power internet of things. *Computational Intelligence and Neuroscience* 2022:9004942
- Carreon-Ortiz H, Valdez F (2022) A new mycorrhized tree optimization nature-inspired algorithm. *Soft Comput* 26(10):4797–4817
- Carvalho C, de Oliveira C, Miotto Galli G, de Oliveira Telesca Camargo N, Pereira M, Stefanello T, Melchior R, Andretta I (2022) Behavior of domestic chickens –insights from a narrative review. *Journal of Agroveterinary Sciences* 21:360–369
- Chakraborty S, Saha AK, Chakraborty R et al (2021) An enhanced whale optimization algorithm for large scale optimization problems. *Knowl-Based Syst* 233:107543
- Chen X, Liu L, Du J et al (2022) Intelligent optimization based on a virtual marine diesel engine using GA-ICSO hybrid algorithm. *Machines* 10(4):227
- Chen Y, Xi J, Wang H, Liu X (2023) Grey wolf optimization algorithm based on dynamically adjusting inertial weight and levy flight strategy. *Evol Intel* 16(3):917–927

- Choi C, Kim H, Kang JH et al (2022) Reconfigurable heterogeneous integration using stackable chips with embedded artificial intelligence. *Nat Electron* 5(6):386–393
- Cristin DR, Kumar DKS, Anbazzhagan DP (2021) Severity Level Classification of Brain Tumor based on MRI Images using Fractional-Chicken Swarm Optimization Algorithm. *Comput J* 64(10):1514–1530
- Cuong-Le T, Minh HL, Khatir S et al (2021) A novel version of Cuckoo search algorithm for solving optimization problems. *Expert Syst Appl* 186:115669
- Daanouni I, Abdennaceur B, Ballouk A (2021) A comprehensive survey on LEACH-based clustering routing protocols in Wireless Sensor Networks. *Ad Hoc Netw* 114:102409
- Dalal S, Seth B, Jaglan V et al (2022) An adaptive traffic routing approach toward load balancing and congestion control in Cloud-MANET ad hoc networks. *Soft Comput* 26(11):5377–5388
- Deb S, Tammi K, Gao XZ et al (2020) A hybrid multi-objective chicken swarm optimization and teaching learning based algorithm for charging station placement problem. *IEEE Access* 8:92573–92590
- Deb S, Gao XZ (2021) A hybrid ant lion optimization chicken swarm optimization algorithm for charger placement problem. *Complex & Intelligent Systems* 1–18
- Demidova LA, Gorchakov AV (2021) Application of chaotic Fish School Search optimization algorithm with exponential step decay in neural network loss function optimization. *Procedia Comput Sci* 186:352–359
- Deng W, Shang S, Cai X et al (2021) An improved differential evolution algorithm and its application in optimization problem. *Soft Comput* 25:5277–5298
- Deng J, Jiang H, Chen Q (2022) Characteristic wavelengths optimization improved the predictive performance of near-infrared spectroscopy models for determination of Aflatoxin B1 in maize. *J Cereal Sci* 105:103474
- Devika G, Ramesh D, Karegowda AG (2021) Energy optimized hybrid PSO and wolf search based LEACH. *Int J Inf Technol* 13:721–732
- Dhanusha C, Kumar AS, Musirin IB, Abdullah HMA (2022) Chaotic chicken swarm optimization-based deep adaptive clustering for alzheimer disease detection. In *Pervasive Computing and Social Networking: Proceedings of ICPCSN 2021*. Springer Singapore, pp 709–719
- Dhiman G, Garg M, Nagar A et al (2021) A novel algorithm for global optimization: rat swarm optimizer. *J Ambient Intell Humaniz Comput* 12:8457–8482
- Di Silvestre ML, Ippolito MG, Sanseverino ER et al (2021) Energy self-consumers and renewable energy communities in Italy: New actors of the electric power systems. *Renew Sustain Energy Rev* 151:111565
- Ding Y, Zhou K, Bi W (2020) Feature selection based on hybridization of genetic algorithm and competitive swarm optimizer. *Soft Comput* 24:11663–11672
- Chen C, Cao L, Chen Y, Chen B, Yue Y (2024) A comprehensive survey of convergence analysis of beetle antennae search algorithm and its applications. *Artif Intell Rev* 57(6):141
- Duan F, Chen C, Song F et al (2023) An optimal parameters estimation for the proton exchange membrane fuel cells based on amended deer hunting optimization algorithm. *Sustain Energy Technol Assess* 58:103364
- El-dosuky MA, Shams M (2022) A deep learning based cockroach swarm optimization approach for segmenting brain MRI images. *Medical informatics and Bioimaging using artificial intelligence: Challenges, Issues, Innovations and Recent Developments*, pp 3–13
- Elsmany EFA, Omar MA, Wan TC et al (2019) EESRA: Energy efficient scalable routing algorithm for wireless sensor networks. *IEEE Access* 7:96974–96983
- Falcón-Cardona JG, Gómez RH, Coello CAC et al (2021) Parallel multi-objective evolutionary algorithms: A comprehensive survey. *Swarm Evol Comput* 67:100960
- Fu W, Wang B, Li X et al (2019) Ascent trajectory optimization for hypersonic vehicle based on improved chicken swarm optimization. *IEEE Access* 7:151836–151850
- Gafar M, El-Sehiemy RA, Hasanien HM, Abaza A (2024) Optimal parameter estimation of three solar cell models using modified spotted hyena optimization. *J Ambient Intell Humaniz Comput* 15(1):361–372
- Gambhir A, Payal A, Arya R (2020) Chicken Swarm Optimization Algorithm Perspective on Energy Constraints in WSN. In *2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*. IEEE, pp 1–5
- Garlapati VK, Parashar SK, Klykov S et al (2022) Invasive weed optimization coupled biomass and product dynamics of tuning soybean husk towards lipolytic enzyme. *Biores Technol* 344:126254
- Garud KS, Jayaraj S, Lee MY (2021) A review on modeling of solar photovoltaic systems using artificial neural networks, fuzzy logic, genetic algorithm and hybrid models. *Int J Energy Res* 45(1):6–35
- Gawali MB, Gawali SS (2021) Optimized skill knowledge transfer model using hybrid Chicken Swarm plus Deer Hunting Optimization for human to robot interaction. *Knowl-Based Syst* 220:106945

- Gharehchopogh FS (2023) An improved Harris Hawks optimization algorithm with multi-strategy for community detection in social network. *J Bionic Eng* 20(3):1175–1197
- Gharehchopogh FS, Shayanfar H, Gholizadeh H (2020) A comprehensive survey on symbiotic organisms search algorithms. *Artif Intell Rev* 53:2265–2312
- Gheibi O, Weyns D, Quin F (2021) Applying machine learning in self-adaptive systems: A systematic literature review. *ACM Trans Auton Adapt Syst (TAAS)* 15(3):1–37
- Goldanloo MJ, Gharehchopogh FS (2022) A hybrid OBL-based firefly algorithm with symbiotic organisms search algorithm for solving continuous optimization problems. *J Supercomput* 78(3):3998–4031
- Gordan M, Sabbagh-Yazdi SR, Ismail Z, Ghaedi K, Carroll P, McCrum D, Samali B (2022) State-of-the-art review on advancements of data mining in structural health monitoring. *Measurement* 193:110939
- Gu Y, Lu H, Xiang L et al (2022) Adaptive Simplified Chicken Swarm Optimization Based on Inverted S-Shaped Inertia Weight. *Chin J Electron* 31(2):367–386
- Guo C, Tang H, Niu B et al (2021) A survey of bacterial foraging optimization. *Neurocomputing* 452:728–746
- Guo T, Yu K, Aloqaily M et al (2022) Constructing a prior-dependent graph for data clustering and dimension reduction in the edge of AIoT. *Futur Gener Comput Syst* 128:381–394
- Gupta A, Mahaur B (2021) An improved DV-maxHop localization algorithm for wireless sensor networks. *Wireless Pers Commun* 117:2341–2357
- Han B, Ran F, Li J et al (2022) A novel adaptive cluster based routing protocol for energy-harvesting wireless sensor networks. *Sensors* 22(4):1564
- Jiang S, Yue Y, Chen C, Chen Y, Cao L (2024) A multi-objective optimization problem solving method based on improved golden jackal optimization algorithm and its application. *Biomimetics* 9(5):270
- Harshavardhan A, Cheerla S, Parkavi A et al (2023) Deep learning modified neural networks with chicken swarm optimization-based lungs disease detection and severity classification. *J Electron Imaging* 32(6):062603–062603
- Hashim FA, Houssein EH, Hussain K et al (2022) Honey Badger Algorithm: New metaheuristic algorithm for solving optimization problems. *Math Comput Simul* 192:84–110
- Hayyolalam V, Kazem AAP (2020) Black widow optimization algorithm: a novel meta-heuristic approach for solving engineering optimization problems. *Eng Appl Artif Intell* 87:103249
- Hodson TO (2022) Root-mean-square error (RMSE) or mean absolute error (MAE): when to use them or not. *Geosci Model Dev* 15(14):5481–5487
- Hussien AG, Amin M, Wang M et al (2020) Crow search algorithm: theory, recent advances, and applications. *IEEE Access* 8:173548–173565
- Ibrahim IA, Hossain MJ, Duck BC et al (2020) An improved wind driven optimization algorithm for parameters identification of a triple-diode photovoltaic cell model. *Energy Convers Manage* 213:112872
- Ishikawa A, Sakaguchi M, Nagano AJ et al (2020) Genetic architecture of innate fear behavior in chickens. *Behav Genet* 50:411–422
- Jaafari A, Panahi M, Mafi-Gholami D et al (2022) Swarm intelligence optimization of the group method of data handling using the cuckoo search and whale optimization algorithms to model and predict landslides. *Appl Soft Comput* 116:108254
- Jonsson A, Vahlne JE (2023) Complexity offering opportunity: Mutual learning between Zhejiang Geely Holding Group and Volvo Cars in the post-acquisition process. *Global Strategy Journal* 13(3):700–731
- Joseph AJ, Asaletha R (2023) Pareto multi-objective termite colony optimization based EDT clustering for wireless chemical sensor network. *Wirel Pers Commun* 130(4):2329–2343
- Kadhuim ZA, Al-Janabi S (2023) Codon-mRNA prediction using deep optimal neurocomputing technique (DLSTM-DSN-WOA) and multivariate analysis. *Results Eng* 17:100847
- Kanna SKR, Sivakumar K, Lingaraj N (2021) Development of deer hunting linked earthworm optimization algorithm for solving large scale traveling salesman problem. *Knowl-Based Syst* 227:107199
- Karim S, Shaikh FK, Aurangzeb K et al (2021) Anchor nodes assisted cluster-based routing protocol for reliable data transfer in underwater wireless sensor networks. *IEEE Access* 9:36730–36747
- Kaya E, Gorkemli B, Akay B et al (2022) A review on the studies employing artificial bee colony algorithm to solve combinatorial optimization problems. *Eng Appl Artif Intell* 115:105311
- Khan AT, Li S, Zhang Y et al (2023) Eagle perching optimizer for the online solution of constrained optimization. *Mem-Mater Devices Circ Syst* 4:100021
- Khan A, Shah R, Bukhari J, Akhter N, Attaullah Idrees M, Ahmad H (2019) A novel chicken swarm neural network model for crude oil price prediction. *Advances on Computational Intelligence in Energy: The Applications of Nature-Inspired Metaheuristic Algorithms in Energy*, pp 39–58
- Khorshid SF, Abdulazeez AM (2021) Breast cancer diagnosis based on k-nearest neighbors: a review. *PalArch's J Archaeol Egypt/egyptol* 18(4):1927–1951

- Kong J, Dagefu FT, Sadler BM (2020) Distributed beamforming in the presence of adversaries. *IEEE Trans Veh Technol* 69(9):9682–9696
- Kumar V, Kumar D (2021) A systematic review on firefly algorithm: past, present, and future. *Arch Comput Meth Eng* 28:3269–3291
- Kumar M, Dubey K, Singh S et al (2023) Experimental performance analysis of cloud resource allocation framework using spider monkey optimization algorithm. *Concurr Comput: Pract Experience* 35(2):e7469
- Kumar D, Pandey M (2022) An optimal load balancing strategy for P2P network using chicken swarm optimization. *Peer-to-Peer Networking and Applications*, pp 1–23
- Kumari N, Dwivedi RK, Bhatt AK, Belwal R (2022a) Automated fruit grading using optimal feature selection and hybrid classification by self-adaptive chicken swarm optimization: grading of mango. *Neural Comput & Applic* 34(2):1285–1306
- Kumari N, Dwivedi R K, Bhatt A K, et al. (2022b) Automated fruit grading using optimal feature selection and hybrid classification by self-adaptive chicken swarm optimization: grading of mango. *Neural Comput Appl* 1–22
- Lamnatou C, Chemisana D, Cristofari C (2022) Smart grids and smart technologies in relation to photovoltaics, storage systems, buildings and the environment. *Renew Energy* 185:1376–1391
- Li W, Wang GG (2022) Elephant herding optimization using dynamic topology and biogeography-based optimization based on learning for numerical optimization. *Eng Comput* 38(Suppl 2):1585–1613
- Li J, Zhou T (2023) Evolutionary Multi Agent Deep Meta Reinforcement Learning Method for Swarm Intelligence Energy Management of Isolated Multi Area Microgrid with Internet of Things. *IEEE Internet Things J* 10(14):12923–12937
- Li Y, Wu Y, Qu X (2017) Chicken swarm–based method for ascent trajectory optimization of hypersonic vehicles. *J Aerosp Eng* 30(5):04017043
- Li Y, Wang S, Han M (2019) Truss structure optimization based on improved chicken swarm optimization algorithm. *Adv Civil Eng* 2019:1–16
- Li L, Lin Q, Li K et al (2021a) Vertical distance-based clonal selection mechanism for the multiobjective immune algorithm. *Swarm Evol Comput* 63:100886
- Li M, Li C, Huang Z et al (2021b) Spiral-based chaotic chicken swarm optimization algorithm for parameters identification of photovoltaic models. *Soft Comput* 25(20):12875–12898
- Li Q, Li D, Zhao K et al (2022a) State of health estimation of lithium-ion battery based on improved ant lion optimization and support vector regression. *J Energy Storage* 50:104215
- Li X, Dong H, Yu X (2022b) A parameter optimization method in predicting algorithms for smart living. *Comput Commun* 191:315–326
- Li Y, Lu Y, Li D et al (2023) Trajectory optimization of high-speed robotic positioning with suppressed motion jerk via improved chicken swarm algorithm. *Appl Sci* 13(7):4439
- Liang J, Wang L, Ma M et al (2018) A fast SAR image segmentation method based on improved chicken swarm optimization algorithm. *Multimedia Tools Appl* 77(24):31787–31805
- Liang X, Kou D, Wen L (2020a) An improved chicken swarm optimization algorithm and its application in robot path planning. *IEEE Access* 8:49543–49550
- Liang J, Wang L, Ma M (2020b) A new image segmentation method based on the ICSO-ISPCNN model. *Multimedia Tools Appl* 79(37):28131–28154
- Liu ZF, Li LL, Tseng ML et al (2020a) Prediction short-term photovoltaic power using improved chicken swarm optimizer-extreme learning machine model. *J Clean Prod* 248:119272
- Liu XF, Zhou YR, Yu X (2020b) Cooperative particle swarm optimization with reference-point-based prediction strategy for dynamic multiobjective optimization. *Appl Soft Comput* 87:105988
- Liu YZ, Ren SF, Zhao PF (2022) Application of the deep neural network to predict dynamic responses of stiffened plates subjected to near-field underwater explosion. *Ocean Eng* 247:110537
- Maarouf BB, Rashid TA, Abdulla JM et al (2022) Current studies and applications of shuffled frog leaping algorithm: a review. *Arch Comput Methods Eng* 29(5):3459–3474
- Macenski S, Foote T, Gerkey B et al (2022) *Robot Operating System 2: Design, architecture, and uses in the wild*. *Sci Robot* 7(66):eabm6074
- Maden D, Çelik E, Houssein EH et al (2023) Squirrel search algorithm applied to effective estimation of solar PV model parameters: a real-world practice. *Neural Comput Appl* 35(18):13529–13546
- Maheshwari P, Sharma AK, Verma K (2021) Energy efficient cluster based routing protocol for WSN using butterfly optimization algorithm and ant colony optimization. *Ad Hoc Netw* 110:102317
- Majid M, Habib S, Javed AR et al (2022) Applications of wireless sensor networks and internet of things frameworks in the industry revolution 4.0: A systematic literature review. *Sensors* 22(6):2087
- Mani M, Bozorg-Haddad O, Chu X (2018) Ant lion optimizer (ALO) algorithm. *Advanced optimization by nature-inspired algorithms*, pp 105–116

- Mansouri N, Javidi MM, Mohammad HasaniZade B (2021) A CSO-based approach for secure data replication in cloud computing environment. *J Supercomput* 77:5882–5933
- Meng X, Liu Y, Gao X, Zhang H (2014) A new bio-inspired algorithm: chicken swarm optimization. In *Advances in Swarm Intelligence: 5th International Conference, ICSI 2014, Hefei, China, October 17–20, 2014, Proceedings, Part I 5*. Springer International Publishing, pp 86–94
- Meraih Y, Gabis AB, Mirjalili S et al (2021) Grasshopper optimization algorithm: theory, variants, and applications. *IEEE Access* 9:50001–50024
- Minaee S, Boykov Y, Porikli F et al (2021) Image segmentation using deep learning: A survey. *IEEE Trans Pattern Anal Mach Intell* 44(7):3523–3542
- Mirbabaie M, Stieglitz S, Frick NRJ (2021) Artificial intelligence in disease diagnostics: A critical review and classification on the current state of research guiding future direction. *Heal Technol* 11(4):693–731
- Mishra S, Nayak PC, Prusty RC et al (2022) Modified multiverse optimizer technique-based two degree of freedom fuzzy PID controller for frequency control of microgrid systems with hydrogen aqua electrolyzer fuel cell unit. *Neural Comput Appl* 34(21):18805–18821
- Mondini AC, Guzzetti F, Chang KT et al (2021) Landslide failures detection and mapping using Synthetic Aperture Radar: Past, present and future. *Earth Sci Rev* 216:103574
- Mu Y, Zhang L, Chen X, Gao X (2016) Optimal trajectory planning for robotic manipulators using chicken swarm optimization. In *2016 8th International conference on intelligent human-machine systems and cybernetics (IHMSC)*, vol 3. IEEE, pp 369–373
- Muduli L, Mishra DP, Jana PK (2018) Application of wireless sensor network for environmental monitoring in underground coal mines: A systematic review. *J Netw Comput Appl* 106:48–67
- Nabavi S, Wen L, Gill SS et al (2023) Seagull optimization algorithm based multi-objective VM placement in edge-cloud data centers. *Int Things Cyber-Phys Syst* 3:28–36
- Nadikattu AKR (2021) Influence of artificial intelligence on robotics industry. *International Journal of Creative Research Thoughts (IJCRT)*. ISSN, pp 2320–2882
- Nagah Amr M, ELAttar HM, Abd El Azeem MH et al (2021) An enhanced indoor positioning technique based on a novel received signal strength indicator distance prediction and correction model. *Sensors* 21(3):719
- Nagarajan B, SVN SK (2023) A poisson hidden markov model and fuzzy based chicken swarm optimization algorithm for efficient fault node detection in wireless sensor network. 1:1–12
- Naruei I, Keynia F (2022) Wild horse optimizer: A new meta-heuristic algorithm for solving engineering optimization problems. *Eng Comput* 38(Suppl 4):3025–3056
- Nayak J, Swapnarekha H, Naik B et al (2023) 25 years of particle swarm optimization: Flourishing voyage of two decades. *Arch Comput Methods Eng* 30(3):1663–1725
- Tang W, Cao L, Chen Y, Chen B, Yue Y (2024) Solving Engineering Optimization Problems Based on Multi-Strategy Particle Swarm Optimization Hybrid Dandelion Optimization Algorithm. *Biomimetics* 9(5):298
- Niu G, Li X, Wan X et al (2022) Dynamic optimization of wastewater treatment process based on novel multi-objective ant lion optimization and deep learning algorithm. *J Clean Prod* 345:131140
- Nuvvula RSS, Devaraj E, Elavarasan RM et al (2022) Multi-objective mutation-enabled adaptive local attractor quantum behaved particle swarm optimisation based optimal sizing of hybrid renewable energy system for smart cities in India. *Sustain Energy Technol Assess* 49:101689
- Osamy W, El-Sawy AA, Salim A (2020) CSOCA: Chicken swarm optimization based clustering algorithm for wireless sensor networks. *IEEE Access* 8:60676–60688
- Othman AM, El-Fergany AA (2021) Adaptive virtual-inertia control and chicken swarm optimizer for frequency stability in power-grids penetrated by renewable energy sources. *Neural Comput Appl* 33(7):2905–2918
- Pan Y, Dong F (2023) Factor substitution and development path of the new energy market in the BRICS countries under carbon neutrality: Inspirations from developed European countries. *Appl Energy* 331:120442
- Pham PV, Bodepudi SC, Shehzad K et al (2022) 2D heterostructures for ubiquitous electronics and optoelectronics: principles, opportunities, and challenges. *Chem Rev* 122(6):6514–6613
- Pourpanah F, Wang R, Lim CP et al (2023) A review of artificial fish swarm algorithms: Recent advances and applications. *Artif Intell Rev* 56(3):1867–1903
- Prabhakar TS, Veena MN (2023) Efficient anomaly detection using deer hunting optimization algorithm via adaptive deep belief neural network in mobile network. *J Ambient Intell Humaniz Comput* 14(12):16409–16425
- Pragadeswaran S, Madhumitha S, Gopinath S (2021) Certain investigation on military applications of wireless sensor network. *Int J Adv Res Sci Commun Technol* 3(1):14–19

- Priyadarshi N, Azam F, Solanki SS et al (2021) A bio-inspired chicken swarm optimization-based fuel cell system for electric vehicle applications[M]//Bio-inspired neurocomputing. Springer, Singapore, pp 297–308
- Qi S, Zou J, Yang S et al (2022) A self-exploratory competitive swarm optimization algorithm for large-scale multiobjective optimization. *Inf Sci* 609:1601–1620
- Rabani H, Soleimani GF (2019) An optimized firefly algorithm based on cellular learning automata for community detection in social networks. *J Adv Comput Res* 10(3):13–30
- Radaideh MI, Shirvan K (2021) Rule-based reinforcement learning methodology to inform evolutionary algorithms for constrained optimization of engineering applications. *Knowl-Based Syst* 217:106836
- Rani R, Garg R (2021) Pareto based ant lion optimizer for energy efficient scheduling in cloud environment. *Appl Soft Comput* 113:107943
- Ren F, Long D (2021) Carbon emission forecasting and scenario analysis in Guangdong Province based on optimized Fast Learning Network. *J Clean Prod* 317:128408
- Rizk-Allah RM, Hassanien AE (2023) A comprehensive survey on the sine-cosine optimization algorithm. *Artif Intell Rev* 56(6):4801–4858
- Rosado-Olivieri EA, Brivanlou AH (2021) Synthetic by design: exploiting tissue self-organization to explore early human embryology. *Dev Biol* 474:16–21
- Rostami M, Berahmand K, Nasiri E et al (2021) Review of swarm intelligence-based feature selection methods. *Eng Appl Artif Intell* 100:104210
- Sabale K, Mini S (2021) Localization in wireless sensor networks with mobile anchor node path planning mechanism. *Inf Sci* 579:648–666
- Sachan S, Deb S, Singh SN et al (2021) Planning and operation of EV charging stations by chicken swarm optimization driven heuristics. *Energy Convers Econ* 2(2):91–99
- Safaldin M, Otair M, Abualigah L (2021) Improved binary gray wolf optimizer and SVM for intrusion detection system in wireless sensor networks. *J Ambient Intell Humaniz Comput* 12:1559–1576
- Saif MAN, Niranjan SK, Murshed BAH et al (2023) CSO-ILB: chicken swarm optimized inter-cloud load balancer for elastic containerized multi-cloud environment. *J Supercomput* 79(1):1111–1155
- Sandeli M, Bouanaka MA, Kitouni I (2021) An efficient localization approach in wireless sensor networks using chicken swarm optimization. In 2021 International Conference on Information Systems and Advanced Technologies (ICISAT). IEEE, pp 1–6
- Schmarje L, Santarossa M, Schröder SM et al (2021) A survey on semi-, self-and unsupervised learning for image classification. *IEEE Access* 9:82146–82168
- Seyyedabbasi A, Kiani F (2023) Sand cat swarm optimization: a nature-inspired algorithm to solve global optimization problems. *Engineering with Computers* 39(4):2627–2651
- Shehab M, Abualigah L, Al Hamad H et al (2020) Moth-flame optimization algorithm: variants and applications. *Neural Comput Appl* 32:9859–9884
- Singh A, Sharma A, Rajput S, Mondal AK, Bose A, Ram M (2022) Parameter extraction of solar module using the sooty tern optimization algorithm. *Electronics* 11(4):564
- Singh H, Rai V, Kumar N et al (2023) An enhanced whale optimization algorithm for clustering. *Multi-media Tools Appl* 82(3):4599–4618
- Singh JP, Kumar M (2023) Conditional autoregressive-tunicate swarm algorithm based generative adversarial network for violent crowd behavior recognition. *Artif Intell Rev* 56(Suppl 2):2099–2123
- Sivanantham K, Kalaiarasi I, Leena B (2022) Brain Tumor Classification Using Hybrid Artificial Neural Network with Chicken Swarm Optimization Algorithm in Digital Image Processing Application[M]//Advance Concepts of Image Processing and Pattern Recognition. Springer, Singapore, pp 91–108
- Slezkin AO, Hodashinsky IA, Shelupanov AA (2021) Binarization of the Swallow swarm optimization for feature selection. *Program Comput Softw* 47:374–388
- Su X, Xue S, Liu F, Wu J, Yang J, Zhou C, Hu W, Paris C, Nepal S, Jin D, Sheng QZ, Yu PS (2024) A comprehensive survey on community detection with deep learning. *IEEE transactions on neural networks and learning systems* 35:4682–4702
- Sutradhar S, Karforma S, Bose R et al (2023) A Dynamic Step-wise Tiny Encryption Algorithm with Fruit Fly Optimization for Quality of Service improvement in healthcare. *Healthc Anal* 3:100177
- Tan Y, Liu X, Tang W et al (2022) Flexible pressure sensors based on bionic microstructures: from plants to animals. *Adv Mater Interfaces* 9(5):2101312
- Tang X, Shen H, Zhao S et al (2023) Flexible brain-computer interfaces. *Nat Electron* 6(2):109–118
- Tawhid MA, Ibrahim AM (2023) An efficient hybrid swarm intelligence optimization algorithm for solving nonlinear systems and clustering problems. *Soft Comput* 27(13):8867–8895

- Tian Y, Liu X, Zhang L et al (2022) Prediction of thermophysical properties of chlorine eutectic salts via artificial neural network combined with polar bear optimization. *J Energy Storage* 55:105658
- Tripathi AK, Garg P, Tripathy A, Vats N, Gupta D, Khanna A (2020) Application of chicken swarm optimization in detection of cancer and virtual reality. *Advanced Computational Intelligence techniques for virtual reality in Healthcare*, pp 165–192
- Trojovský P, Dehghani M (2022) Pelican optimization algorithm: a novel nature-inspired algorithm for engineering applications. *Sensors* 22(3):855
- Vamsidhar A, Surya Kavitha T, Ramesh Babu G (2022) Image enhancement using chicken swarm optimization. In *Proceedings of the international conference on computational intelligence and sustainable technologies: ICoCIST 2021*. Singapore: Springer Nature Singapore, pp 555–565
- Verma S, Sahu SP, Sahu TP (2023) Two-stage hybrid feature selection approach using levy's flight based chicken swarm optimization for stock market forecasting. *Computational Economics* 1–32
- Wang Z, Zhu D (2021) Sports monitoring method of national sports events based on wireless sensor network. *Wirel Commun Mob Comput* 2021:1–13
- Wang J, Zhang F, Liu H et al (2020) Interruptible load scheduling model based on an improved chicken swarm optimization algorithm. *CSEE J Power Energy Syst* 7(2):232–240
- Wang S, Cao L, Chen Y, Chen C, Yue Y, Zhu W (2024) Gorilla optimization algorithm combining sine cosine and cauchy variations and its engineering applications. *Scientific Reports* 14(1):1–20
- Wang Z, Qin C, Wan B, Song WW, Yang G (2021a) An Adaptive Fuzzy Chicken Swarm Optimization Algorithm. *Math Probl Eng* 2021:1–7
- Wang B, Tao F, Fang X et al (2021b) Smart manufacturing and intelligent manufacturing: A comparative review. *Engineering* 7(6):738–757
- Wang Z, Yue Y, Cao L (2022a) Mobile Sink-Based Path Optimization Strategy in Heterogeneous WSNs for IoT Using Pigeon-Inspired Optimization Algorithm. *Wirel Commun Mob Comput* 2022:1–18
- Wang Z, Zhang W, Guo Y et al (2023) A multi-objective chicken swarm optimization algorithm based on dual external archive with various elites. *Appl Soft Comput* 133:109920
- Wang T, Huang H, Li X, Guo X, Liu M, Lei H (2023b) Optimal estimation of proton exchange membrane fuel cell model parameters based on an improved chicken swarm optimization algorithm. *International Journal of Green Energy* 20(9):946–965
- Wei Q, Huang D, Zhang Y (2021) Artificial chicken swarm algorithm for multi-objective optimization with deep learning. *J Supercomput* 77(11):13069–13089
- Wu Y (2021) A survey on population-based meta-heuristic algorithms for motion planning of aircraft. *Swarm Evol Comput* 62:100844
- Wu Y, Yan B, Qu X (2018a) Improved chicken swarm optimization method for reentry trajectory optimization. *Math Probl Eng* 2018:1–13
- Wu Z, Yu D, Kang X (2018b) Application of improved chicken swarm optimization for MPPT in photovoltaic system. *Optim Control Appl Methods* 39(2):1029–1042
- Wu D, Kong F, Gao W, Shen Y, Ji Z (2015) Improved chicken swarm optimization. In *2015 IEEE international conference on cyber technology in automation, control, and intelligent systems (CYBER)*. IEEE, pp 681–686
- Wunnava A, Naik MK, Panda R et al (2022) A differential evolutionary adaptive Harris hawks optimization for two dimensional practical Masi entropy-based multilevel image thresholding. *J King Saud Univ-Comput Inf Sci* 34(6):3011–3024
- Xie L, Han T, Zhou H et al (2021) Tuna swarm optimization: a novel swarm-based metaheuristic algorithm for global optimization. *Comput Intell Neurosci* 2021:1–22
- Yan C, Xiao J, Chen D et al (2021) Feed restriction induced changes in behavior, corticosterone, and microbial programming in slow-and fast-growing chicken breeds. *Animals* 11(1):141
- Yang Q, Song GW, Gao XD, Lu ZY, Jeon SW, Zhang J (2023) A random elite ensemble learning swarm optimizer for high-dimensional optimization. *Complex & Intelligent Systems* 9(5):5467–5500
- Yanto ITR, Setiyowati R, Irsalinda N et al (2020) Laying Chicken Algorithm (LCA) Based For Clustering. *JOIV: Int J Inform Vis* 4(4):208–212
- Ye C, Wang A, Breakwell C et al (2022) Development of efficient aqueous organic redox flow batteries using ion-sieving sulfonated polymer membranes. *Nat Commun* 13(1):3184
- Yu X, Zhou L, Li X (2019) A novel hybrid localization scheme for deep mine based on wheel graph and chicken swarm optimization. *Comput Netw* 154:73–78
- Yu Y, Rashidi M, Samali B et al (2022) Crack detection of concrete structures using deep convolutional neural networks optimized by enhanced chicken swarm algorithm. *Struct Health Monit* 21(5):2244–2263
- Yue Y, You H, Wang S et al (2021) Improved whale optimization algorithm and its application in heterogeneous wireless sensor networks. *Int J Distrib Sensor Networks* 17(5):15501477211018140

- Yue Y, Cao L, Lu D et al (2023) Review and empirical analysis of sparrow search algorithm. *Artif Intell Rev* 3:1–53
- Yue Y, Lu D, Zhang Y, Xu M, Hu Z, Li B, ..., Ding H (2022) A data collection method for mobile wireless sensor networks based on improved dragonfly algorithm. *Computational Intelligence and Neuroscience* 2022
- Yue Y, Cao L, Zhang Y (2024) Novel WSN Coverage Optimization Strategy Via Monarch Butterfly Algorithm and Particle Swarm Optimization. *Wirel Pers Commun* 135:2255–2280
- Zervoudakis K, Tsafarakis S (2020) A mayfly optimization algorithm. *Comput Ind Eng* 145:106559
- Zhang Y, Zhao J, Wang L et al (2021) An improved OIF Elman neural network based on CSO algorithm and its applications. *Comput Commun* 171:148–156
- Zhang C, Mousavi AA, Masri SF et al (2022) Vibration feature extraction using signal processing techniques for structural health monitoring: A review. *Mech Syst Signal Process* 177:109175

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.