A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends

Jun Tang, Gang Liu, and Qingtao Pan

***Abstract*—Swarm intelligence algorithms are a subset of the artificial intelligence (AI) field, which is increasing popularity in resolving different optimization problems and has been widely utilized in various applications. In the past decades, numerous swarm intelligence algorithms have been developed, including ant colony optimization (ACO), particle swarm optimization (PSO), artificial fish swarm (AFS), bacterial foraging optimization (BFO), and artificial bee colony (ABC). This review tries to review the most representative swarm intelligence algorithms in chronological order by highlighting the functions and strengths from 127 research literatures. It provides an overview of the various swarm intelligence algorithms and their advanced developments, and briefly provides the description of their successful applications in optimization problems of engineering fields. Finally, opinions and perspectives on the trends and prospects in this relatively new research domain are represented to support future developments.**

***Index Terms*—Ant colony optimization (ACO), artificial bee colony (ABC), artificial fish swarm (AFS), bacterial foraging optimization (BFO), optimization, particle swarm optimization (PSO), swarm intelligence.**

I. Introduction

In nature, groups of thousands, millions, or trillions of individual elementary entities can self-organize into multifarious forms to fit a functional objective, purely through local and ordinary interactions. Examples at different biological levels include the self-assembly of crystals of bacterial flagella at the molecular scale, regular development of multicellular organisms at the cellular scale, and interconnected food searching or risk avoidance (for energy-saving in ants) at the colony scale [1]. Research on biological behavioral intelligence has been conducted for a long time; a study regarding Cape bees and " animal intelligence" was published in 1883 [2].

Throughout nature, different organisms often profit from acting in swarms. Models of the simple collective behavior of individuals indicate that uncomplicated partial interactions are sufficient to produce and present group morphologies. Individuals constitute groups that can collectively process information and provide corporate decisions. By shared information, the group could make better decisions than a single individual; this phenomenon is called as "collective intelligence" [3]. For example, bees appear to rely on decentralized decision-making for the critical choice of a new nest site. Although no individual evaluates the complete set of available information or directly compares the available options, the swarm efficiently integrates the resulting flow of information into a high-quality final decision, without central control.

By studying the characteristics of individuals and their relationships with groups, algorithms for the corresponding mechanism, known as " swarm intelligence" have been formalized. These essentially comprise biologically inspired computations and have been deemed an emerging field and an integral part of artificial intelligence (AI) [4]. Swarm intelligence algorithms can be organic combination of evolutionary computing, artificial neural networks (ANNs), and fuzzy systems in a series of computational intelligence (CI) approaches [4]. They are known as a group of nature-inspired methods that are used to deal with complex optimization problems for which mathematical or traditional approaches are invalid. This ineffectiveness is mainly caused by, e.g., the overwhelming complexity of the procedure for mathematical reasoning, possible uncertainties, or a procedure that is stochastic by nature. Among the heuristic algorithms inspired by nature, in addition to the swarm-based intelligent optimization algorithms mainly introduced in this paper, some are also inspired by natural evolution, physical rules, and human behavior. The hierarchical distribution of these algorithms is shown in Fig. 1 . Specifically, we divide the heuristic algorithms inspired by nature into the above four categories. In each category, we have listed five classic algorithms. The first is the method of swarm intelligence inspired by the intelligent behavior between biological

Manuscript received March 31, 2021; revised April 25, 2021; accepted May 17, 2021. This work was supported in part by the National Natural Science Foundation of China (62073330), in part by the Natural Science Foundation of Hunan Province (2019JJ20021, 2020JJ4339) and in part by the Scientific Research Fund of Hunan Province Education Department (20B272). Recommended by Associate Editor Shangce Gao. *(J. Tang and G. Liu contributed equally to this work. Corresponding author: Jun Tang.)*

Citation: J. Tang, G. Liu, and Q. T. Pan, " A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends," *IEEE/CAA J. Autom. Sinica*, vol. 8, no. 10, pp. 1627–1643, Oct. 2021.

J. Tang and Q. T. Pan are with the College of Systems Engineering, National University of Defense Technology, Changsha 410000, China (e-mail: tangjun06@nudt.edu.cn; panqingtao@nudt.edu.cn).

G. Liu is with the School of Information Science and Engineering, Hunan Institute of Science and Technology, Yueyang 414000, China (e-mail: liugang@hnist.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JAS.2021.1004129

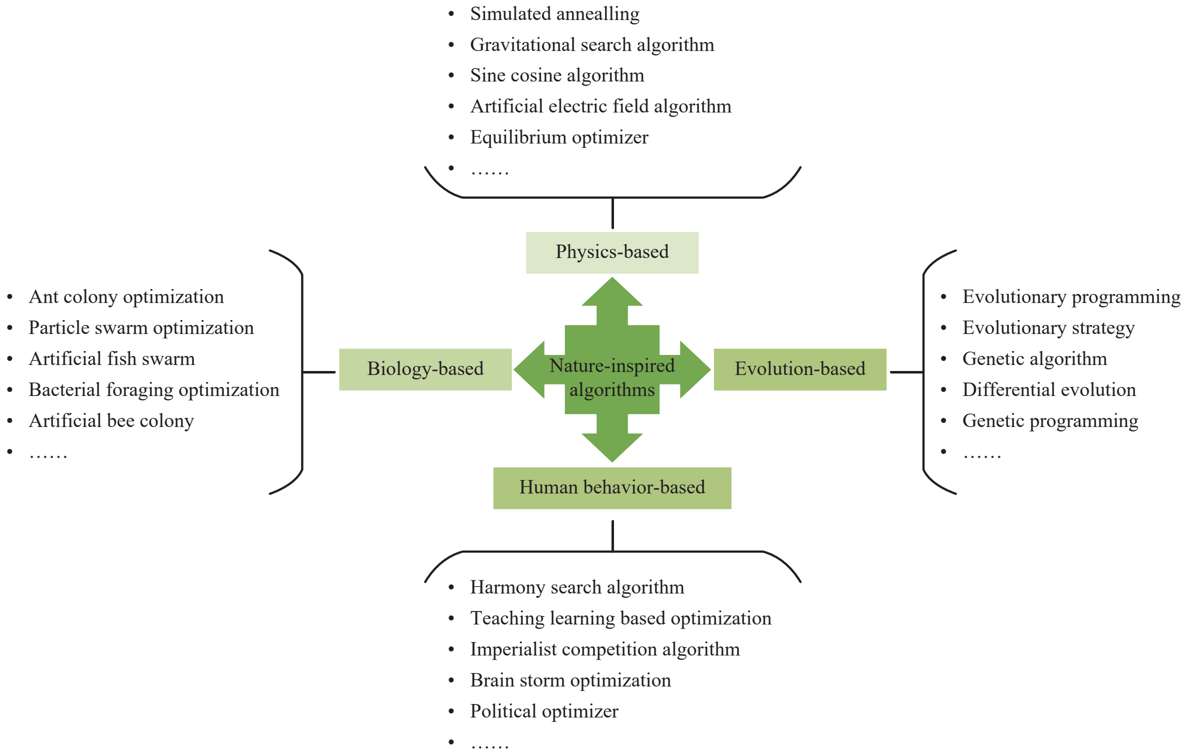
• Simulated annealling
• Gravitational search algorithm
• Sine cosine algorithm
• Artificial electric field algorithm
• Equilibrium optimizer
• ……
Physics-based
• Ant colony optimization
• Particle swarm optimization
• Artificial fish swarm
• Bacterial foraging optimization
• Artificial bee colony
• ……
Biology-based
Nature-inspired algorithms
Evolution-based
• Evolutionary programming
• Evolutionary strategy
• Genetic algorithm
• Differential evolution
• Genetic programming
• ……
Human behavior-based
• Harmony search algorithm
• Teaching learning based optimization
• Imperialist competition algorithm
• Brain storm optimization
• Political optimizer
• ……

Fig. 1. The hierarchical distribution of nature-inspired algorithms.

swarms, which will be discussed in this review. Secondly, algorithms based on the laws of cosmic physics are also an important branch, including the well-known simulated annealing algorithm, gravity search algorithm, sine cosine algorithm, artificial electric field algorithm, etc. Then, evolutionary algorithms based on evolutionary theory are produced by using the laws and processes of biological heredity and population evolution in nature. The most popular is the genetic algorithm that simulates Darwin's evolution. In addition, classic algorithms include differential evolution, evolution strategy and evolution programming. Finally, there are algorithms based on human intelligent behavior, for example, the harmony search algorithm that simulates music creation by musicians, the teaching learning-based optimization algorithm imitating the influence of teachers on students, the political optimization algorithm that imitates the multi-stage process of politics, etc.

The term "swarm intelligence" was formally put forward by Beni and Wang in the background of studying a cellular robotics system [5]. Currently, it is considered as a subdiscipline of CI, aiming to solve problems by modelling populations of agents that can self-organize and interact with each other. In view of well-organized principles, we have selected many of the most representative swarm intelligence algorithms and divided them into six categories, based on the number of algorithms used and the basic biology they are derived from: ant colony optimization (ACO), particle swarm optimization (PSO), artificial fish swarm (AFS), bacterial foraging optimization (BFO), artificial bee colony (ABC) and other swarm intelligence algorithms. As swarm intelligence algorithms possess features of self-organization, parallel operations, distributive operations, flexibility, and robustness, they have been gradually very widespread and are utilized in many events. For example, swarm intelligence algorithms are used for scheduling problems, robots, power systems, parameter optimization, system identification, image processing, signal processing, and other optimization problems of engineering fields. Hence, studying the swarm intelligence algorithms possesses extremely significant academic and practical value.

From the number of publications based on Google Scholar data since 2000, we can obtain that abundant research has been conducted on swarm intelligence algorithms in the past two decades, as shown in Fig. 2. Overall, from 2000 to 2005, the development of the various algorithms for swarm intelligence was relatively slow; this period can be regarded as the initial stage. From 2005 to 2012, the publication of various algorithms in Google Scholar contributed to a rapid growth trend, meaning that the swarm intelligence algorithms developed rapidly. After 2012, the related research was still increasing, but the trend was relatively flat. Specifically, among the five swarm intelligence algorithms, PSO has been the most researched; it developed particularly rapidly from 2005 to 2012. However, the growth rate slowed down after 2012, and particularly, the publications have continued to slightly decline since 2016. It is likely that the AI experts and scholars have gradually paid attention to machine learning, tree searching, reinforcement learning, and deep neural networks in recent years [6]. The research on ACO is second only to that on PSO, and the overall research shows a steady growth trend. Unlike the other algorithms, whose growth rates slowed down after 2012, the ABC algorithm has become more pronounced since 2012, owing to the popular research topic of

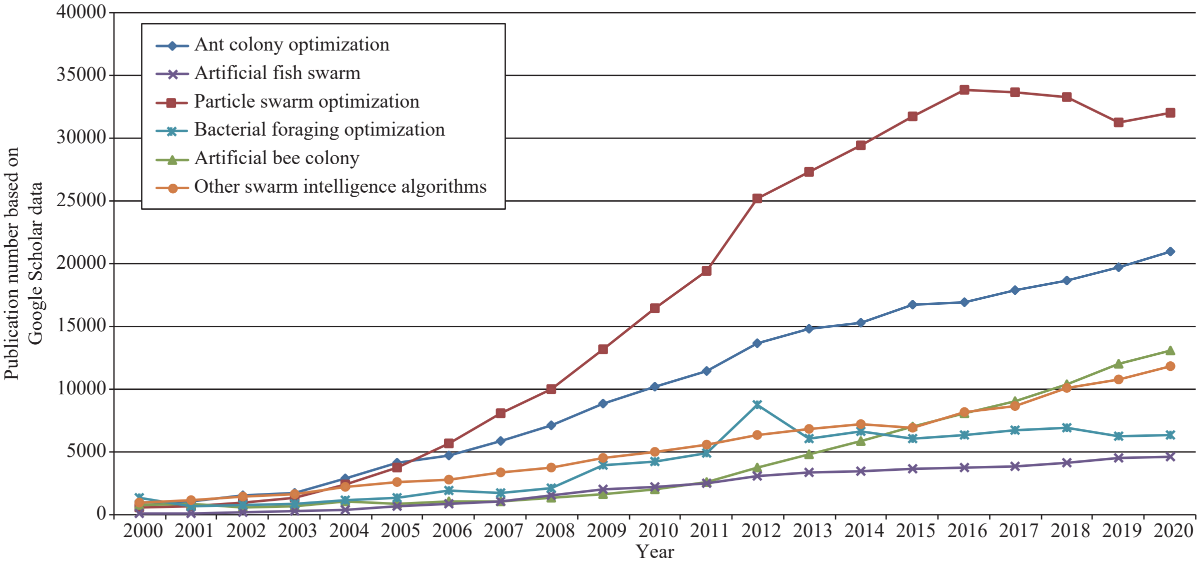

Fig. 2. Google trends indicator of swarm intelligence algorithms from 2000 to 2020.

robot swarms. The AFS algorithm has had relatively fewer academic achievements over the years, but is the most stable. Academic publications regarding the BFO algorithm were particularly prominent in 2012, as more academic conferences were held on related topics. In addition to the above five types of algorithms, there are many other swarm intelligence algorithms, and the corresponding research is also increasing each year.

This review is intended to have reference significance for professional and technical researchers interested in the state-of-the-art in swarm intelligence algorithms and their trends in related fields. The remainder of this paper is organized as follows. Section II divides the swarm intelligence algorithms into six categories. Section III describes the achievements of the swarm intelligence schemes in various applications of related fields; several results from these applications are provided and are compared with the pipelines of commonly utilized approaches. Section IV summarizes major strengths and limitations in swarm intelligence algorithms, along with main trends that would be developed in the future. Section V concludes this review.

II. Swarm Intelligence Algorithms

From flocks of birds to schools of fish to swarms of insects, biological aggregations displaying collective behavior emerge throughout nature. This self-organized, natural phenomenon has been the subject of intensive modelling for decades, as it is highly commonly employed in nature to efficiently solve problems and owing to its potential ability as a biomimetic control strategy for engineering systems [7]. In recent years, swarm intelligence has been universally studied in various areas; consequently, many related approaches to collective behavior have emerged. A categorization of swarm intelligence algorithms with several representative works, are illustrated in Fig. 3 . In this section, we will briefly review each of these swarm intelligence algorithms, along with their most recent developments and applications.

Essentially, swarm intelligence algorithms are iteratively based on stochastic search algorithms, where heuristic information is shared to conduct the search for the following iterations. Fig. 4 shows a general framework for swarm intelligence algorithms [8]. Before the initialization phase, the parameter values are required to be defined. The evolutionary process then begins with initialization and its corresponding strategies. Then, the termination condition used to stop the algorithm execution is set; it is either one condition or comprised of two separate conditions. The fitness function that is in charge of the evaluation of the search agents is evaluated; the fitness function can be a basic metric, or a composed one. The algorithm updates agents until the previous termination condition is achieved. Finally, the best search result is generated. For a specific swarm intelligence algorithm, the sequence of each process may be different, and some processes could be executed several times in a single iteration.

A. Ant Colony Optimization

In computer science and operational research, ACO is a probabilistic method for resolving computational problems which could be reduced to searching the optimal paths through graphs. It was first proposed by Marco Dorigo in 1992 based on his Ph.D. research [9], and the inspiration for this basic algorithm came from the behavior of ants finding paths in the procedure of food searching or risk avoidance. The ACO algorithm has characteristics of distributed computing, positive information feedback, and a heuristic search [10]. Its basic idea is to imitate an ant's dependence on pheromones, and to guide each ant's action by using positive feedback among the ants. This algorithm can be utilized to deal with most optimization problems or can be transformed into optimization problems. Ants can leave a type of material called a pheromone in the paths they pass. They can sense the intensity of this material and can thereby guide their own direction of action in the process of foraging [11]. The basic searching process of ACO is as follows [12].

At first, m ants are randomly placed in the location. The

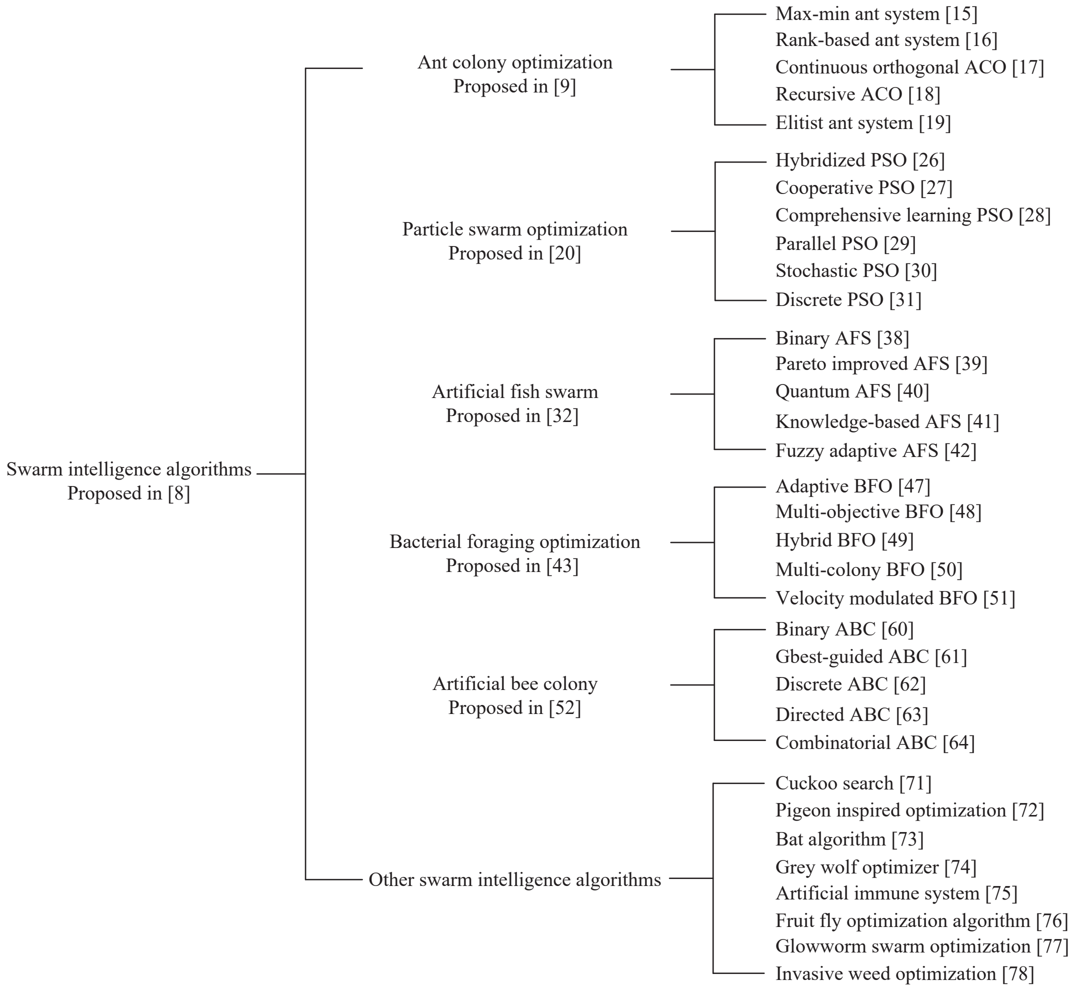

Fig. 3. A categorization of the swarm intelligence algorithms and their representative works.

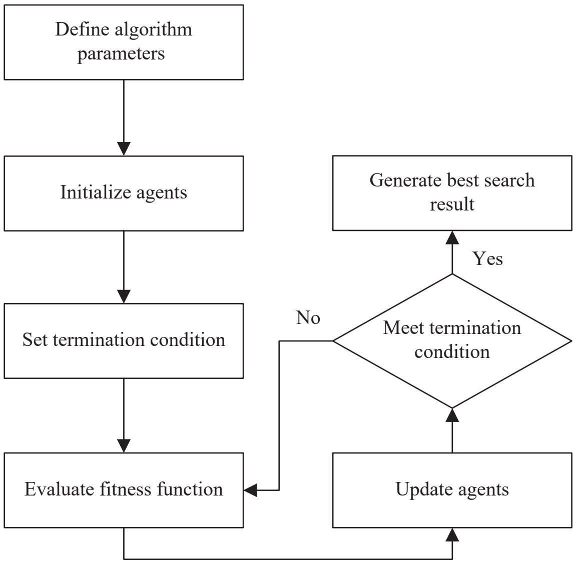
Define algorithm parameters
Initialize agents
Set termination condition
Evaluate fitness function
Update agents
Meet termination condition
No
Yes
Generate best search result

Fig. 4. General framework of the swarm intelligence algorithms.

initial values of the pheromones on each path are equal, that is $\tau_{ij}(0)=\tau_0$. Then, the k-th ($k=1,2,\ldots,m$) ant chooses the next location to be transferred to on account of the random proportion rule, and its selection probability is shown as follows [13]:

$$p_{ij}^{k}(t)=\begin{cases}\dfrac{\left[\tau_{ij}(t)\right]^{\alpha}\left[\eta_{ij}(t)\right]^{\beta}}{\sum\limits_{s\in allowed_k}[\tau_{is}(t)]^{\alpha}[\eta_{is}(t)]^{\beta}}, & j\in allowed_k\\ 0, & \text{Otherwise.}\end{cases} \tag{1}$$

In the above, τ_{ij} is the pheromone of edge; $\eta_{ij}=1/d_{ij}$ is the heuristic factor of transferring from location i to location j, where d_{ij} is the distance; and $allowed_k$ is the location set that the k-th ant is allowed to visit next.

After the t-moment, all ants have accomplished a tour. The path length of each ant is calculated, and the shortest path length is saved. At the same time, the pheromones on each edges are updated. The pheromone update includes two parts [14]: i) the volatilization of pheromones, as in (2); and ii) the release of pheromones by ants on the edge they pass by (3)

$$\tau_{ij}=(1-\rho)\tau_{ij} \tag{2}$$

$$\tau_{ij}=\tau_{ij}+\sum_{k=1}^{m}\Delta\tau_{ij}^{k}. \tag{3}$$

In the above, ρ is the volatilization coefficient of the pheromone, $\Delta\tau_{ij}^{k}$ is the pheromone released by the k-th ant passing the edge (i,j). After the ants complete a cycle, they return to the original location and prepare for the next tour.

The original ACO concept has been diversified to manage a wider group of numerical problems. However, several new problems have emerged, and additional studies have been required to improve the ACO performance, depending on different aspects of the ant behaviors. The *MAX–MIN* ant system [15] is a typical representative method for directly improving an ant system. This algorithm modifies the approach to updating pheromones. After each iteration, only one ant can update the pheromone to obtain a better solution. To avoid the search stagnation, the pheromone on the path is limited to $[MIN, MAX]$. In addition, the initial value of the pheromone takes its upper limit, helping to promote the algorithm's search ability. Bullnheimer *et al.* [16] provided a novel rank-based version of an ant system. After each iteration, the paths of the ants are arranged from small to large. The algorithm gives different weights on account of the path length; the shorter the path length, the greater the weight. Hu *et al.* [17] proposed a new ACO denoted "continuous orthogonal ACO" for solving continuous problems. The pheromone deposition mechanism of this algorithm enables ants to efficiently search for solutions. Gupta *et al.* [18] introduced the concept of depth for a recursive ACO. The depth determined the number of recursions, and each depth was based on an ordinary ant colony algorithm. Gao *et al.* [19] introduced a K-means clustering algorithm to combine with ACO algorithm with three immigrant schemes for addressing the dynamic location routing problem. Hemmatian *et al.* [20] applied an elitist ACO algorithm to the multi-objective optimization of hybrid laminates, to obtain the minimum cost in the calculation process.

B. Particle Swarm Optimization

In groups of birds acting in a cooperative energy-saving manner, individuals in the group continuously optimize the search mode, in view of the experience of group members. Based on a study of swarm behavior, Kennedy and Eberhart put forward the PSO algorithm in 1995 [21]. In PSO, a flock of particles retain motion in a constrained parameter space, interact with each other, and update their velocities and positions on account of their own and their neighbors' information, finding the global optimum [22], [23]. The initial state of the algorithm is a group of random particles comprising a particle swarm; it also represents a random solution space. Each particle owns only two properties: velocity and position. Each particle searches for the optimal solution results in the corresponding space, and continuously updates its individual optimal value. Particle swarm members share their individual optimal values. The global optimal solution is the best of the current individual optimal values [24], [25]. Based on the current individual optimal value and the current global optimal solution, the particles continuously adjust the velocities and positions. The above search process can be formulated as (4) and (5) [26], as follows:

$$v_{t+1} = c_1 v_t + c_2 r_1 (p_{i,t} - x_t) + c_3 r_2 (p_{g,t} - x_t) \tag{4}$$

$$x_{t+1} = x_t + v_{t+1}. \tag{5}$$

In the above, v_t and x_t are the velocity and position of particle i at time t, respectively. $p_{i,t}$ is the individual optimal value of particle i found before time t, and $p_{g,t}$ represents the global optimal solution of the particle swarm found before time t. c_1, c_2, c_3 are the respective cognition coefficients, and r_1, r_2 are the random parameters within $[0, 1]$.

The first portion of (4) is the "memory term" and indicates the particle's velocity vector in the current state. The second portion of (4) is a " self-cognition term", which is a vector from the current value to the individual optimal value. It expresses the influence of the particle's own experience. The third portion of (4) is a " global-cognition term", i.e., the vector from the particle's current value to the current global optimal solution. It reflects the cooperation and information sharing between particles.

Because of its simplicity, effectiveness, and low computational cost, PSO has obtained significant popularity, and many improvements have been proposed. Most researches of PSO improvement fall into categories which modify the model coefficients, consider the population structures, and alter the interaction modes [27]. By combining PSO and a genetic algorithm, Ghamisi and Benediktsson [28] provided a new feature selection method. This method also adopts a support vector machine (SVM) as a fitness function, which improves the classification accuracy. To address the challenge of premature convergence to a local optimum in global optimization problems, Li *et al.* [29] introduced a competitive and cooperative PSO with a data-sharing mechanism. It allows particles to share each other's individual optimal values, instead of simply reflecting the cooperation in the current global optimal solution. Furthermore, the comprehensive learning PSO proposed in [30] allows members in the swarm to share the historical individual optimal values of all other particles. This algorithm is particularly outstanding in solving multimodal problems. In [31], a parallel PSO algorithm with a special operator was proposed, and it has proven effective in solving the problem of PSO convergence at a suboptimal location. Since the PSO algorithm was proposed, its convergence has been an important aspect of the research. Via the depth analysis of basic PSO algorithm, Zeng and Cui proposed the stochastic PSO algorithm [32], which can ensure the probability of converging to be globally optimal. In contrast to the traditional PSO, which focused on solving continuous problems, Clerc [33] studied the reasonable adoption of a discrete PSO to the classic traveling salesman problem. The results represent that the discrete PSO algorithm could be easily combined with other algorithms to efficiently solve certain discrete problems. In [34], a PSO-based neural network identifier was constructed to model the unknown system dynamics, and proved to be more effective in resolving the Hamilton-Jacobi-Bellman equation by utilizing the estimated system states. Roy *et al.* [35] proposed a PSO-based artificial neural network for enhanced forecasting of the reliability of software, considering the fault generation phenomenon with the fault complexity of different levels. Lv *et al.* [36] combined a surrogate-assisted PSO with Pareto active learning to solve the multi-objective optimization problem with high computational cost, and in this research

PSO was used as a sampler to generate candidate solutions.

C. Artificial Fish Swarm

In nature, fish could discover more nutritious areas by individual searching or following other fish, and areas with more individuals are normally more nutritious. The stochastic population-based AFS optimization algorithm was initially proposed by Li *et al.* in 2002 [37]. Its main idea is to imitate fish behaviors such as preying, swarming, and following, with a local search of individuals for generating a global optimum [38]. The AFS algorithm possesses attractive features similar to those of a genetic algorithm, but it can reach a faster convergence speed and requires the adjustment of fewer parameters. The AFS algorithm does not include crossover and mutation procedures, so it can be executed more easily. It is also an optimizer in view of population; the system is initialized with a group of randomly generated potential solutions, and then iteratively executes a search for the optimum one [39]. The AFS algorithm acts as a parallel and random search optimization method, with the advantages of a strong global search capacity and an excellent convergence speed. The fish group is represented by a set of points or solutions, with each agent representing a candidate solution. The feasible solution space constitutes the "waters" where the artificial fish moves and searches for the optimum. The survival, competition, and coordination mechanisms are introduced into the basic operation [40]. The main challenge in the algorithm implementation is determining how to use simple and effective ways to construct and realize these behaviors. In addition, the PSO algorithm and the AFS algorithm are often used for comparison in research: the PSO algorithm is inspired by a flock of birds, and it uses some particles flying in the search space to find the best solution, that is, these particles continuously adjust themselves based on the original path according to the position of the optimal solution in the particle swarm; the AFS algorithm is mainly based on the foraging behavior of the fish school in the real environment with a bottom-up design idea, and it mainly uses the three operators of artificial fish school for foraging, clustering and tail-finding to construct the underlying behavior of the individual.

The description of basic AFS processing in a global optimization problem for minimizing $f(x) x \in \Omega$ is as follows [41], [42]. There exist a population of artificial fish including n agents or points $x_i, i = 1,2,\ldots,n$ and x_i is a floating-point encoding with the lower and upper bounds l_i and u_i. The search space of x_i is: $\Omega = \{x \in \mathbb{R}^n : l_i \le x_i \le u_i, i = 1,2,\ldots,n\}$. Each point x_i has a closed neighborhood with a radius v, which is called the " visual scope", and $v = \delta \max_{j \in \{1,2,\ldots,n\}}$ $(u_j - l_j)$. In addition, $\delta \in (0,1)$ is a positive visual parameter which can be decreased by the iterative process. The "visual scope" of point x_i is np^i number of points. The crowding factor $C_f = np^i/N$ indicates the degree of crowding. If $np^i = 0$, the " visual scope" is empty, and point x_i will move stochastically and look for better areas; if $C_f \le \theta$, the "visual scope" is not crowded, and point x_i can chase or swarm towards the best or central point of the " visual scope". If $C_f \ge \theta$, the "visual scope" is congested, and the point x_i has a problem insofar as chasing. It will randomly choose and move toward another point. In the AFS algorithm, the swarm behavior is featured by the motion to the central point of the "visual scope".

During the use of the AFS algorithm, it has been developed and improved by numerous researchers, generating a series of novel variants. In [43], a binary version of the AFS is proposed for resolving 0–1 multidimensional knapsack problems. In this method, a binary string of 0/1 bits represents a point. Through copying the relevant bit from itself or from another specified point with identical probability, each bite of a trial point is produced. Zhang *et al.* [44] proposed a Pareto-improved AFS algorithm for resolving a multi-objective fuzzy disassembly line balancing problem, and it was useful in decreasing the existing uncertainty. To heighten the global search ability and the convergence speed, Zhu and Jiang [45] proposed a novel quantum AFS algorithm. The algorithm is on account of principles and concepts of quantum computing, e.g., the quantum bit and quantum gate. As regular artificial fish can get stuck in blindness hunting, Gao *et al*. [46] provided a knowledge-based AFS algorithm with crossover. In addition, the conventional regular AFS could only perform local or global searching by continuously initializing the step and visualization parameters. To remedy this, Yazdani *et al.* [47] proposed two novel adaptive methods for utilizing fuzzy systems to adjust the step and visualization parameters during execution. The purpose is to adaptively adjust the capability for global and local searching. In the first one, the visualization and step parameters of all fish are adjusted uniformly, whereas in the second, a special fuzzy controller controls the step and visualization parameters for each artificial fish.

D. Bacterial Foraging Optimization

The BFO algorithm was first proposed by Passino in 2002 [48]. It models the foraging behavior of Escherichia coli in humans, is relatively new to the family of nature-inspired optimization algorithms. It optimizes through the competition and cooperation among bacterial populations and is utilized as a global random search algorithm. It has characteristics of simplicity and fast convergence, and there is no need to optimize the gradient information of the object during the optimization process [49]. The BFO simulation of the bacterial population contains three steps: chemotaxis, reproduction, and elimination/dispersal, corresponding to the following three processes [50]: in chemotaxis, a bacterium moves by carrying on small steps while searching for nutrients, and a core concept of BFO is simulating the chemotactic movement of the virtual bacteria in the corresponding search space; then, in the reproduction step, the remaining bacteria multiply to keep the population size; finally, the local area where the individual bacteria live may suddenly change, which may lead to elimination/dispersal, i.e., the collective death of the bacterial population staying in this local area or the collective migration to another local area [51]. The new individual randomly created by the migration operation may be nearer to the global optimal solution, which is more valuable to jumping out of a local optimal solution.

The main procedures in basic BFO are summarized as follows [48], [49].

The size of the bacterial population is S, and the location of the bacteria indicates a candidate solution to this problem. The information of a bacterium i is labelled with a D-dimensional vector $\theta_i = \left[\theta^i_1, \theta^i_2, \theta^i_3, \ldots, \theta^i_D\right]$. A chemotactic step is deemed as a tumble followed by a tumble, or a tumble followed by a run. In addition, $\theta_i(j+1,k,l)$ indicates the position of the bacterium i after the j-th tropism operation, k-th replication operation, and l-migration operation. Moreover, $C(i) > 0$ represents the step size of forward swimming, and $\Phi(j)$ represents the unit direction vector (randomly selected after rotation). If the fitness at $\theta_i(j+1,k,l)$ is better than that at $\theta_i(j,k,l)$, Φ is kept unchanged, and the bacteria keep to swim in this direction until a position with the best fitness is discovered or the set number of tendencies is reached; otherwise, a new Φ is generated, and the next rotation is performed.

$$\theta_i(j+1,k,l) = \theta_i(j,k,l) + C(i)\Phi(j). \tag{6}$$

When searching for food in their own way, each bacterium also receives attractive signals from other individuals in the population, and receives repulsive signals from nearby individuals for maintaining a safe distance. $P(j,k,l)$ is the position of each bacterium in the population S after the j-th tropism operation, k-th replication operation, and l-migration operation. $J_{cc}(\theta, P(i,k,l))$ indicates that the synthesized effects of attraction and repulsion among the bacteria are considered simultaneously.

$$J_{cc}(\theta, P(i,k,l)) = \sum_{i=1}^{S} J^i_{c\varepsilon}(\theta, \theta^i(j,k,l)). \tag{7}$$

In the BFO algorithm, its population size does not change after a copy operation. The number of bacteria eliminated is usually set as $S_r = S/2$. First, the bacteria are sorted according to their pros and cons; then, the S_r bacteria that are ranked lower are eliminated. The remaining S_r bacteria will self-replicate. A newly generated individual has the same position as the original individual. The migration operation appears with a certain probability. If a bacterial individual meets the probability of migration, the bacterial individual would die, and a new individual would be randomly created at any position in the solution space. Destroyed individuals may own different positions.

The BFO algorithms also have some extensions for subsequent research. Majhi *et al.* [52] proposed a high-performance forecasting model for the prediction of various stock indices, and they minimized the mean square error to optimize the connecting weights of the adaptive linear combiner. In [53], Niu *et al.* described a multi-objective BFO and introduced two different performance metrics, diversity and generational distance, to evaluate multi-objective optimization problems. Chen *et al.* [54] developed a multi-colony BFO algorithm in which a single population bacterial foraging method was applied to the interacting multiple colony model (which links the chemotaxis behavior of individual bacterial cells to the intercellular communications of bacterial communities). Several improvements have been made by combining bacterial foraging optimization algorithms with other algorithms. Gollapudi *et al.* [55] developed a velocity-modulated BFO technique by combining the BFO algorithm and a PSO technique. Kim *et al.* [56] presents a hybrid technique involving genetic algorithms and BFO algorithms, focusing on mutations, crossovers, step changes, chemotaxis steps, and bacterial lifespans.

E. Artificial Bee Colony

A group of honey bees, called as a swarm, could complete different tasks through social collaboration. Karaboga proposed the ABC algorithm in 2005 to solve a multivariate function optimization problem, in view of the intelligent foraging behavior of a honey bee swarm [57]. Bees form groups of insects, and although the behavior of a single insect is extremely simple, a group of single individuals represent very complex behavior. Real bee populations can collect nectar from food sources (flowers) with high efficiency in any environment; they can also simultaneously adapt to environmental changes [58]. In the original ABC algorithm, it contains three kinds of individuals: employed bees, onlooker bees, and scout bees. Each employed bee corresponds to a certain food source (solution vector), and the field of the food source is iteratively searched [59]. More formally, employed bees will search for a richer food source (d_m) near the original food source (x_m). They will evaluate the fitness and compare it with that of original food source. They would find a new food source by using a process corresponding to (8) [60], as follows:

$$d_{mi} = x_{mi} + \omega_{mi}(x_{mi} - x_{ki}). \tag{8}$$

Each food source x_m is a solution vector to a problem with n components ($x_{mi}, i = 1,2,\ldots,n$); and ω_{mi} is a random number within a certain range. The bees choose the food source based on the fitness value, using a greedy selection. The fitness can be calculated as follows [61]:

$$fit_m(\vec{x}_m) = \begin{cases} \dfrac{1}{1+f_m(x_m)} & \text{if } f_m(x_m) \geq 0 \\ 1 + abs(f_m(x_m)) & \text{if } f_m(x_m) < 0\,. \end{cases} \tag{9}$$

In the above, $f_m(\cdot)$ is the objective function for the optimization problem.

In the onlooker bee phase, an onlooker bee selects a food resource, based on the probability values computed by the fitness values of the employed bees' food resources. A roulette method based on the probability value is employed to observe and collect food with the onlooker bees. The probability value indicates how likely a food resource is to be selected by an onlooker bee, and it can be computed by using (10) [62], as follows:

$$p_m = \frac{fit_m(x_m)}{\sum\limits_{m=1}^{M} fit_m(x_m)}. \tag{10}$$

Here, M indicates the population size of the food resources.

During the scout bee phase, if the food source is not updated multiple times, the food source is abandoned, and the employed bee turns to a scout bee to randomly search for another food source [63]. To prevent this method from falling into a local optimum, when the food source iteration limit is not improved, the food source is recorded in a taboo table. Simultaneously, the employed bee corresponding to the honey source is transformed into a scout bee, to randomly produce a new position for finding a new food resource [64].

Based on observing the operations and structure, there are also drawbacks to the basic ABC algorithm. For example, the artificial bee could only move straight to one of the nectar sources, and the characteristic may decrease the set of explored zones. Hence, several novel ABC algorithms have been proposed to address some discovered disadvantages. The searching mechanism can be improved by using an efficient genetic selection method [65]. This new discrete binary ABC algorithm can solve the dynamic clustering problem, and the number of clusters is determined automatically. In [66], a Gbest-guided ABC and global bee colony search are improved by taking the basis of the maximum fitness values instead of the maximum cycle numbers, and the results are superior. Pan *et al*. [67] improved the method for generating food resources by using a self-adaptive strategy to ensure that the discrete ABC algorithm could be adopted to discrete spaces. The improvement in the local search approach is also shown to be effective in enhancing the local intensification capability. Its convergence speed can be enhanced by applying a control parameter, as discussed in [68]. Experiments with this method have proven the solution quality and the convergence to global optimum. In [69], Karaboga and Gorkemli provided a combinatorial ABC algorithm that showed good performance in solving complex optimization problems regarding finding a Hamiltonian path with the lowest cost. Ji *et al*. [70] proposed a scale-free ABC algorithm that applied an innovative method for a hybrid ABC algorithm with a scale-free network to enhance the optimization performance without sacrificing efficiency.

F. Other Swarm Intelligence Algorithms

Other swarm intelligence algorithms are still mainly derived from the simulation of natural biomes, especially those based on insects and animals. They are optimized using artificial algorithm simulations of the processes of finding food or exchanging information in biomes, and most of them are directional iterative methods based on a probability search [71]. With the development of swarm intelligence, the optimization algorithms are not limited to introducing biological population characteristics into the algorithm. Some studies have introduced human biological characteristics into swarm intelligence algorithms, such as the human immune mechanism [72]. The new swarm intelligence algorithms and improvements to classic algorithms proposed in recent years mainly focus on reducing parameters, simplifying processes, increasing computation speeds, and improving search capabilities, and are aimed at high-dimensional and multi-objective optimization problems [73]–[76]. The fields of application for swarm intelligence algorithms are becoming increasingly extensive, and the expansion of algorithm applications has also guided the development directions.

In addition to the several classic swarm intelligence algorithms, there are many other extended algorithms that have been widely discussed and employed. In view of the obligate brood parasitic behavior of several cuckoo species in combination with the Lévy flight behavior of several birds and fruit flies, a meta-heuristic algorithm defined as the "cuckoo search" was formulated to deal with optimization problems [77]. Pigeon-inspired optimization is a swarm intelligence optimizer utilized to resolve air-robot path planning problems. It presents a map and compass operator model on account of a magnetic field and the sun, whereas a landmark operator model is constructed on account of landmarks [78]. The bat algorithm is an animal group/herd-based learning algorithm which utilizes the echolocation behavior of bats to generate solutions for single- and multi-objective domains within a continuous solution space [79]. The grey wolf optimizer is a swarm intelligence algorithm inspired by grey wolves, and simulates their leadership hierarchy and hunting mechanisms in nature [80]. An artificial immune system (ARTIS) contains several typical properties of natural immune systems, such as diversity, distributed computation, error tolerance, dynamic learning and adaptation, and self-monitoring. ARTIS acts as a complete framework for a distributed adaptive system and could, in principle, be utilized in different domains [81]. The fruit fly optimization algorithm (FOA) is a method of seeking global optimization based on the foraging behavior of fruit flies. Fruit flies can rely on their outstanding sense of smell and vision to find better food sources and locations where their companions gather [82]. Glow-worm swarm optimization is developed by incorporating the behavior of glow-worms into AI systems. It can be used for the simultaneous calculation of multiple optima of multimodal functions [83]. Inspired by colonizing weeds, invasive weed optimization is a numerical stochastic optimization algorithm for mimicking the robustness, adaptation, and randomness of colonizing weeds, using an uncomplicated yet effective optimization algorithm [84].

III. Applications and Results

The appearance of swarm intelligence algorithms has provided fast and reliable methods for obtaining solutions to different complex optimization problems. These swarm intelligence algorithms, including ACO, PSO, AFS, BFO, ABC, and other algorithms, reflect their ability in resolving nonlinear design problems in real-world applications thinking over nearly all areas of science, engineering, and industry. These applications include, e.g., those concerning transport problems (such as those concerning unmanned aerial vehicles (UAVs)), network routing, route planning, robot systems for scheduling problems, power systems, fault diagnosis, parameter optimization, system identification, cluster analysis, data mining, image processing, layout optimization, and signal processing. The purpose of this section is to present some typical and relevant professional applications in which the use of swarm-based optimization algorithms has successfully been made in the literature (but not to present all possible applications).

Fig. 5 depicts the various above-mentioned applications for swarm intelligence algorithms in different fields, based on the

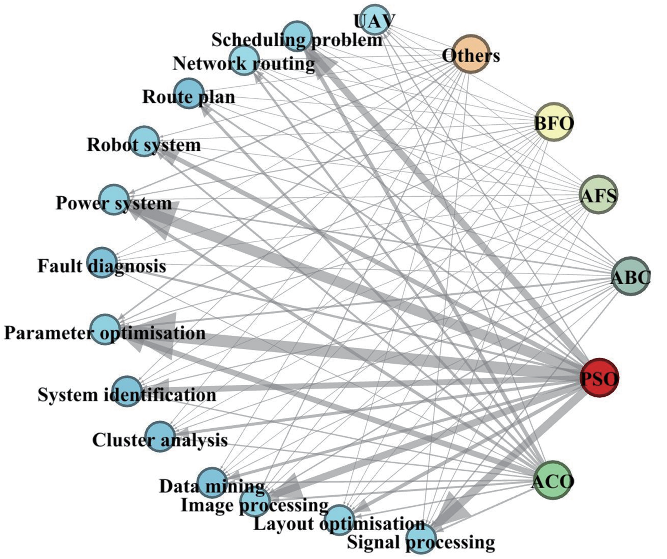

Fig. 5. Various applications of swarm intelligence algorithms in different fields.

published studies explored in Google Scholar. If one paper satisfies applications of swarm intelligence algorithms, it is selected in this review for further introduction. Some high-frequency application domain keywords are used to refine literature classification. Based on this kind of search strategy and criteria, appropriate publications are recorded and reviewed. Among them, the direction of the arrow indicates the application of that type of swarm intelligence to resolve the corresponding problem; the line thickness is based on the number of studies found. The information shown in Fig. 5 is basically consistent with that in Fig. 1, e.g., PSO and ACO are the two most proverbially used algorithms. In addition, we can find that scheduling problem, robot system, power system, parameter optimization, system identification, image processing, and signal processing utilize the swarm intelligence algorithms the most; these seven applications are explained in further detail below. To ensure the quality of review, it should be emphasized that the principle of adopting the corresponding swarm intelligence algorithm literatures in each type of application is that the citation rate is relatively high. In addition, the fan diagrams in Fig. 6 are used to illustrate the proportion of representative applications of different swarm intelligence algorithms, and this format refers to [85].

A. Scheduling Problem

Scheduling acts as the procedure of arranging, controlling, and optimizing task, resource, workloads, etc. in a production procedure, manufacturing module, or mathematical computation [86]. Most research on scheduling problems aims to maximize the efficiency of operations and to reduce costs. For example, in manufacturing, the objective of scheduling is to minimize the production time and costs by instructing a facility when to produce a product, with which staff, and on which equipment.

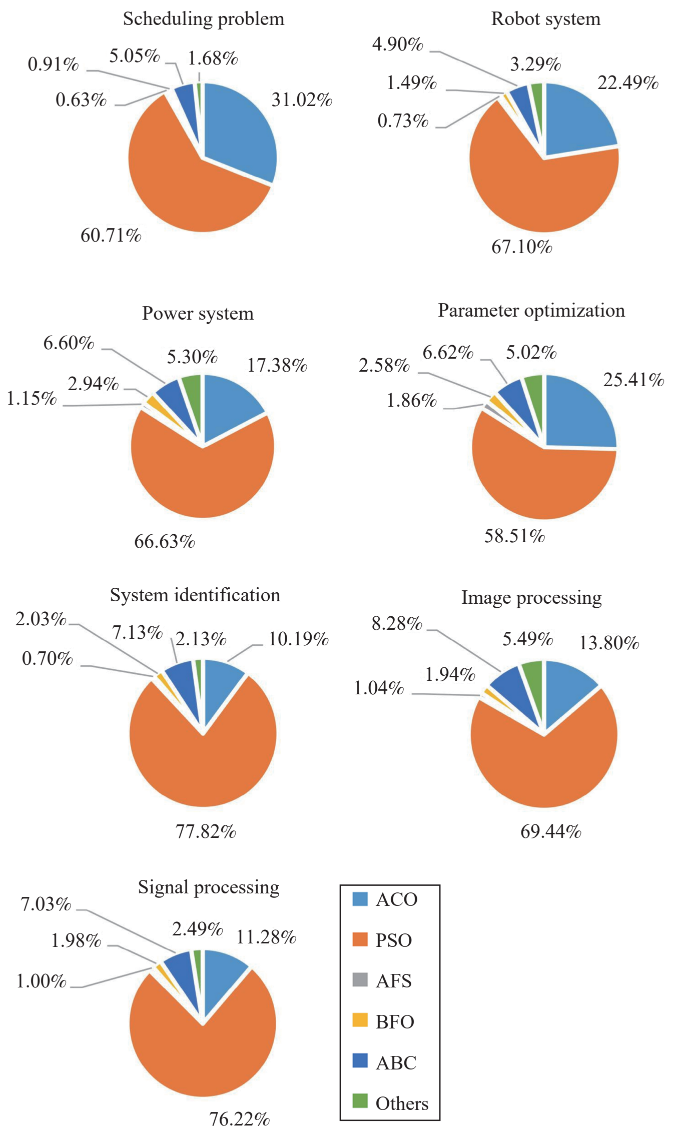

Fig. 6. Statistical diagrams to show the proportion of applications of swarm intelligence algorithms.

The resource-constrained project scheduling problem (RCPSP) acts as one of the most investigated kinds of scheduling problems; its objective is to generate the resource-to-task assignments that make a finite project plan the cheapest or shortest. Myszkowski *et al.* [87] studied the usability and robustness of the ACO and its hybrids with priority rules in resolving the RCPSP, through renewing pheromone values in view of both the best and worst solutions preserved by the ants. Based on PSO, García-Nieto *et al.* [88] determined successful cycle programs for traffic lights and applied the optimization technique for two extensive traffic networks placed in the metropolitan centers of Spanish cities. Pan [89] presented a new cooperative co-evolutionary ABC algorithm for addressing a novel steelmaking continuous casting (SCC) scheduling problem in the context of iron and steel production processing. Its effectiveness was confirmed using instances from real-world SCC program. Kumar *et al.* [90] used the AFS algorithm as a stochastic-based search algorithm for serviceably solving the optimal scheduling problem of energy generation among available renewable energy sources. Bamini and Enoch [91] applied a classification and regression tree as a workflow to increase an allocation of resources, and a modified BFO algorithm was implemented to efficiently schedule the resources among the submitted jobs and to minimize the overall time of execution.

B. Robot System

Robot research in swarm intelligence focuses on the robot design, the physical body group size, and the control behaviors. Generally, this research studies the coordination of multiple robots, as a system consisting of many simple physical robots [92]. It is supposed that the constant feedback for the desired collective behaviors appears from the interactions between these robots, and their interactions with the environment. As compared to ordinary distributed robotic systems, swarm robotics emphasizes large numbers of robots and heightens scalability.

Globally optimal path planning is a common problem in the navigation of autonomous mobile robots, and the navigation must be collision-free. Tan *et al*. [93] utilized the ACO algorithm to resolve this problem, and verified that it provided better performance in convergence speed, solution variation, dynamic convergence behavior, and calculative efficiency than the general path planning technique, on account of a real-coded genetic algorithm with an elitist model. Contreras-Cruz *et al*. [94] combined the ABC algorithm and an evolutionary programming algorithm to improve a feasible path generated by a series of local optimization procedures. Experiments demonstrated the statistical significance of the promotion obtained by their proposed method. Yao *et al*. [95] structured a hybrid adaptive AFS algorithm based on adaptive enhanced prey behavior and the segmented adaptive strategy for an artificial fish's view and steps, and the algorithm was verified based on path planning for a coal mine rescue robot. In addition, for other issues in robotics research, Pugh *et al*. [96] explored the performance of PSO in noisy environments while paying attention to unsupervised robotic learning and suggested an augmentation of the original method for achieving maximization of performance. Aghajarian *et al*. [97] presented an innovative application of the BFO algorithm to generate a fuzzy controller for tracking control of a robot manipulator. It was used efficiently to constitute a rule base and membership functions. The simulation results represented the superiority of the BFO algorithm (as compared to the PSO algorithm) for this application.

C. Power System

Owing to the network connections of the power system and further innovations in the electricity market, the power system is in fact developing to be a large-scale nonlinear dynamic system, and several complex engineering calculations are required to be resolved [98]. Note that swarm intelligence does not require any preconditions for centralized control and a global model, it is really appropriate for solving large-scale nonlinear optimization problems that are difficult to solve using traditional methods (and for which it is hard to establish effective formalized models). Recently, there have been several studies on the successful practical application of swarm intelligence in power systems.

Economic dispatch is of great value for electrical power system operations. To solve this complex problem, Zhou *et al*. [99] described a multi-objective multi-population ACO algorithm for a continuous domain. It proved valuable in minimizing a total fuel cost subject to various physical and operational constraints, via appropriately allocating power demands to generator modules. Abido [100] employed a PSO algorithm to find the optimal settings of power-system stabilizer parameters. An eigenvalue analysis and the nonlinear simulation results demonstrated its effectiveness in damping out the local and inter-area modes of oscillations, and it performed availably over a wide range of loading conditions and system configurations. Gozde *et al*. [101] provided the usage of the ABC algorithm as an AI-based optimization technique for optimizing an automatic generation control system, and provided a comprehensive analysis of its tuning performance and contributions to robustness. The market clearing price (MCP) acts as one of the main factors for load control and market monitoring in a power system. Li and Wang [102] applied the AFS algorithm to improve the ability to determine the MCP, and it provided advantages insofar as fast training, the overcoming of local extrema, swift access to a global optimum, and so on. Ali and Abd-Elazim [103] proposed BFO-based load frequency control for the suppression of oscillations in a power system. It was adopted to obtain optimal controller parameters by minimizing a time domain objective function.

D. Parameter Optimization

Parameter optimization considers a field of problems that seek optimal control parameters [104]. Traditionally, the most elaborate systematic methods for parameter optimization are on account of generalization error estimations and gradient descents. In recent years, the development of parameter optimizations for various operating systems has been supported by AI techniques and evolutionary strategies.

One of the most important research problems in an SVM is the generation of optimal parameters to develop an efficient SVM, so as to achieve the desired output with an acceptable level of accuracy. Zhang *et al*. [105] applied the ACO algorithm to establish a new SVM model to resolve this problem. It only explored a finite subset of the possible values, to obtain the parameters that minimized the generalization error. A flux-cored arc welding (FCAW) procedure is a fusion welding procedure in which the welding electrode is a tubular wire that is continuously fed to the weld area. The welding input parameters are truly critical in determining the quality of a weld joint. Panda *et al*. [106] provided a model of the weld bead geometry in the FCAW course based on an ANN, and optimization of the process parameters utilizing the PSO algorithm. Focusing on the improvement of algorithm performance, Akay and Karaboga [107] introduced some improvements on the original ABC algorithm, and the performance of the improved ABC algorithm was investigated for real-parameter optimization in both basic and composite functions. Proportional integral derivative (PID) control, as a classical control strategy, has good robustness, and is widely adopted in process and motion control. Cheng and Hong [108] utilized the AFS algorithm for parameter optimization, which is the core of PID controller design. In addition, the decrease of product development cycle time is a core concern for industries intended to keep competitive in the marketplace; therefore, rapid fabrication techniques are increasingly important. Panda *et al.* [109] applied the BFO algorithm to represent a theoretical combination of parameter settings for simultaneously achieving great strength in all procedure responses.

E. System Identification

System identification is the phrase adopted in the automatic control field to denote the optimal estimation of dynamic black-box models for systems, in view of measurements of the input and output signals of those systems [110]. It aims to start from measurements of the system behavior and external influences, and attempts to confirm a mathematical relation between them, without investigating the details of what is actually occurring inside the system.

Fuzzy logic has been used in the analysis of physical and engineering problems. In a fuzzy identification problem, several unknown parameters are required to be found, and the interval for each parameter is uncertain. Thus, Tsai and Chen [111] provided a novel ACO-based fuzzy identification algorithm for finding silted initial consequent parameters and an initial membership function matrix. In [112], Alfi and Modares proposed a method to find optimal system parameters and optimal control parameters utilizing a new adaptive PSO algorithm. Its main advantages were in achieving a faster convergence speed and better solution accuracy with the minimum increment in the computational burden. To distinguish unknown fractional-order chaotic systems from the view of optimization, Hu *et al.* [113] presented an innovative parameter estimation scheme on account of a hybrid ABC algorithm; it was proven to be a feasible, effective, and promising method. Han *et al.* [114] addressed parameter identification for a photovoltaic module in view of an enhanced AFS algorithm. Its feasibility was approved by identifying the parameters of one commercial photovoltaic submodule with a single diode model under various operating conditions. Majhi and Panda [115] developed a valuable identification scheme for nonlinear dynamic systems adopting combinatorial swarm intelligence algorithms, e.g., BFO and PSO. Compared to functional link ANN-based methods, the combinatorial technique is faster, more accurate, and involves less computation.

F. Image Processing

Image processing is the use of a computer to process different images using algorithms [116]. The main influencing factor in the development of image processing is the increasing demand for a comprehensive range of applications, e.g., in medicine, the military, agriculture, and industry. Currently, increasing research has been conducted on nature-inspired methods for image processing, such as swarm intelligence algorithms.

An edge is one of the simplest and the most important features of an image, and it is widely adopted in image recognition, segmentation, enhancement, and compression. Tian *et al.* [117] successfully applied an ACO algorithm to image edge detection by combining gradients and the relative differences of statistical means. Based on PSO, Omran *et al.* [118] provided a novel dynamic clustering method for image segmentation. It could automatically determine the optimum cluster number and simultaneously cluster the data set, with minimal user interference. Image enhancement acts as the core phase in an image processing system and tries to improve both the visual and informational qualities of distorted images. Draa and Bouaziz [119] proposed an innovative ABC algorithm for image contrast enhancement, and its superiority was proven by comparing the obtained results with those of a genetic algorithm. For efficient color image quantization, El-Said [120] exploited the optimization capability of an improved AFS algorithm to overcome the deficiencies of a fuzzy C-means algorithm. The algorithm could also be used to obtain visually improved images after the quantization. In addition, to enhance color images using a fuzzy logic technique, Hanmandlu *et al.* [121] used a bacterial foraging algorithm to generate the parameters for optimized entropy and the visual factors which are involved in the objective function.

G. Signal Processing

Signal processing plays an extremely key role in extracting valuable information from different detectors, and acts as an electrical engineering subfield which includes analyzing, modifying, and synthesizing signals such as sounds, images, and biological measurements [122]. Relevant techniques are applied to promote the transmission, storage efficiency, and subjective quality of the signals, and also to emphasize or find components of interest in a measured signal.

Signal processing in regard to brain activity has become a challenge for various researchers. Huang *et al.* [123] proposed an ACO algorithm to generate the distributions of the intensity of a targeted image, so as to obtain the spatiotemporal characteristics of brain activities. Shadmand and Mashoufi [124] used an ANN to classify electrocardiogram heartbeats of a patient into five typical types, and the network structure and weights were effectively optimized adopting a classic PSO algorithm. Koza and Karaboga [125] utilized an ABC algorithm for the construction of a quadrature mirror filter bank with a mitral valve Doppler signal, and used it in various subjects, such as image coding and signal processing. Jiang and Yuan [126] combined an AFS algorithm with a conventional wavelet threshold algorithm, and enhanced the effects of signal processing using optimization. Sahu *et al.* [127] resolved the problem of electric power systems being polluted by unwanted fluctuations in both the current and voltage signals, based on an improved BFO algorithm. The algorithm could also raise the classification accuracy of various power quality events.

IV. Discussions and Trends

Recently, swarm intelligence has been extensively discussed in the field of multi-objective optimization. Consequently, many related approaches have emerged. Along with the promising performances swarm intelligence has realized, the research literature has also indicated some important challenges, as well as inherent trends. These are summarized in Table I, and are described below.

TABLE I
ANALYSIS OF SWARM INTELLIGENCE ALGORITHMS

Strengths	Relatively higher scalability Excellent exploration and exploitation capability Simple individuals and collective intelligence Good robustness
Limitations	Temporal complexity Stagnation situation/Local optimum Possible slower feedback
Trends	Novel extension of swarm algorithms Characteristic refinement of individual agent Further improvement of application transformation Comprehensive consideration of negative factors Normative systematization of refined datasets

A. Strengths

Swarm intelligence algorithms present advantages as compared to traditional methods, and even as compared to other AI algorithms. Based on the relevant literature, some representative advantages are summarized as follows.

1) The most basic feature of the swarm intelligence algorithms is the flexible number of individuals, facilitating higher scalability. Thus, their control mechanisms are not excessively based on swarm size, as long as it is not too small or too large.

2) They present the capability to realise a relatively large-scale search, and to refine the solutions in a single search process. In the phase of generating the optimal solution, they usually present excellent exploration and exploitation capabilities.

3) Several simple individuals interacting locally among themselves can eventually lead to a sophisticated global behavior. Low-level interactions between simple individuals have been represented to make the percolation of information known by only a few individuals throughout an entire aggregate, and to promote the emergence of collective intelligence. This mechanism is particularly suitable for some frontiers in special fields, such as targeted medical therapies, bee colony drones, and geographic surveillance.

4) Basically, the swarm operates without central control, and no specific individual is indispensable for the swarm to keep operation. This means that a malfunction in any part of this system would not increase the risk of a complete failure of the entire swarm system, providing good robustness.

B. Limitations

As concluded in the majority of the papers, swarm intelligence algorithms have proven to be valuable in dealing with various tasks and challenges. However, there are also several limitations worth mentioning, and the representative limitations are summarized as follows.

1) Swarm intelligence algorithms are generally time-consuming processes which are affected by factors such as the size of population, frequency of iterations, and pattern of iterations. Thus, these factors directly affect the efficiency of swarm intelligence algorithms relative to the size of relevant applications, and these algorithms may be worthless when the factors exceed certain sizes.

2) Basic swarm intelligence algorithms can suffer from stagnation or a premature convergence to a local optimum, owing to the lack of central coordination. Therefore, they need to be improved to realise adaptive mechanisms for continuously exploiting and exploring the search space, and for balancing the searching speed.

3) Blocking coordination mechanisms increase the calculation time. In particular, an agent has to stay for other agents to be evaluated before it could continue to another position and start exploring the search domain. For example, the basic PSO algorithm is computationally costly, owing to the waiting time between evaluation of the first particle and that of the rest of the swarm before the state could finally be updated.

C. Future Research Directions

The research in swarm intelligence algorithms has shown a positive evolution; in addition, it provides several future research directions that may contribute to further growth in the field of swarm intelligence, as discussed below.

1) Novel extensions of swarm algorithms. The existing swarm intelligence algorithms form an older category of optimization techniques, i.e., immense research has been introduced, and potential application areas have been expanded. These techniques are represented by an adequate number of references in the previous research. Deeper investigation into the normal, possibly universal, features of the collective behaviors of animals, bacteria, cells, molecular motors, and driven granular matter are necessary for extension of the swarm algorithm family. Appropriate and novel application areas are required to be sought out for other advanced and/or simplified, time-effective algorithms, as these are still unexplored and comparatively newer approaches to resolving complicated computation problems.

2) Characteristic refinement of individual agent. Future research should also pay attention to the characteristic refinement of each individual agent, to promote the overall optimization performance of the swarm intelligence algorithms. For example, there are two proverbial kinds of interaction patterns: hierarchical, and egalitarian. Some research has indicated that an individual will switch interaction patterns in different situations. This switching mechanism is prospective for potential industrial applications in multi-robot system coordination, unmanned vehicle formation control, and other areas. Within these individual agent-based techniques, there is also immense potential for fine-tuning, i.e., to further optimize outcomes based on context-specific requirements. Furthermore, the interplay of

social and environmental information and their influences on population performance could be further investigated.

3) Further improvement of application transformation. In nature, thousands (or even millions) of individuals could easily form a great range of patterns such as bacteria colonies, bird flocks, and insect swarms, purely based on local communications. In view of the designed methods and wireless communication, robotic systems could simulate several complex biological swarm structures. Using patterns and rules, these elements could dramatically amend the swarm shape based on the environment. However, creating a swarming robotic system with a large number of individual members at a micro-scale that embodies functional collective behaviors is also a challenge. Therefore, while expanding the algorithm, future research should also focus on hardware developments and algorithm transformations in the application area.

4) Comprehensive consideration of negative factors. It is well-known that social interactions benefit swarm members in diverse ways, such as by increasing the foraging ability or reducing the predation risk. However, the social benefits should be weighed against possible adverse effects, such as de-valued individual information, and reduced sensitivity to changing environments. Individuals must weigh environmental information gathered from their senses along with social information or face a larger risk of isolation and predation. These possible negative factors from group members should be investigated, to make it easier to consider and simulate negative downside elements in an actual system when applying swarm intelligence algorithms.

5) Normative systematization of refined datasets. Swarm intelligence is achieving popularity for resolving different optimization problems and is attracting the attention of an incremental number of researchers. Researchers should pay attention to comparing different swarm intelligence algorithms and their variants and providing in-depth empirical and analytic studies. All the innovative algorithms should be tested, and a normative systematization of refined datasets should be required and/or recommended. The refined datasets should be classified based on the different algorithms. In addition, a taxonomy of uniformly recognized evaluation methods (e.g., for testing the performance, advantages, and drawbacks of specific algorithms) are necessary for improving comparisons of the results, and could greatly promote the application process.

V. Conclusion

Swarm intelligence algorithms mimic the swarm intelligence behavior of biological elements in nature, and have been gradually popular cross-discipline and/or research field in recent years. This review provides a comprehensive survey of swarm intelligence and represents a categorization scheme for analyzing the existing swarm intelligence studies with applications. It is evident from the study that the representative algorithms discussed here (ACO, PSO, AFS and BFO and ABC) are very popular and well-explored, with several literatures relating to each subject. Authors have also expressed details of lesser-known algorithms, among which the cuckoo search, pigeon-inspired optimization, bat, grey wolf optimizer, artificial immune system, and fruit fly optimization algorithms have formed an important and growing category. These algorithms have been utilized in numerous diverse domains for solving optimization problems, e.g., scheduling problems, robots, power systems, parameter optimization, system identification, image processing, and signal processing. Despite the promising results reported so far, there is significant space for further advances. The strengths and limitations are discussed and future research trends are summarized, in an attempt to provide a standard for the evaluation and application of the algorithms and to display potential valuable directions for subsequent research. Swarm intelligence is still active, and its potential is far from being exhausted; research and applications will continue to increase in the future.

Abbreviations

In Table II, we summarize the abbreviations and notations used throughout the review in alphabetical order.

TABLE II
Abbreviations Used in This Overview

Abbreviation	Term
ABC	artificial bee colony
ACO	ant colony optimization
AFS	artificial fish swarm
AI	artificial intelligence
ANN	artificial neural networks
ARTIS	artificial immune system
BFO	bacterial foraging optimization
CI	computational intelligence
FCAW	flux cored arc welding
FOA	fruit fly optimization algorithm
MCP	market clearing price
PID	proportional integral derivative
PSO	particle swarm optimization
RCPSP	resource-constrained project scheduling problem
SCC	steelmaking-continuous casting
SVM	support vector machine
UAV	unmanned aerial vehicle

References

[1] M. Rubenstein, A. Cornejo, and R. Nagpal, " Programmable self-assembly in a thousand-robot swarm," *Science* , vol. 345, no. 6198, pp. 795–799, Aug. 2014.

[2] J. H. De Villiers, " Cape Bees and " animal intelligence"," *Nature*, vol. 28, no. 705, pp. 5–6, May 1883.

[3] T. Sasaki and D. Biro, "Cumulative culture can emerge from collective intelligence in animal groups," *Nat. Commun.*, vol. 8, no. 1, Article No. 15049, Apr. 2017.

[4] A. Chakraborty and A. K. Kar, " Swarm intelligence: A review of algorithms," in *Nature-Inspired Computing and Optimization: Theory*

and Applications, S. Patnaik, X. S. Yang, and K. Nakamatsu, Eds. Cham, Germany: Springer, 2017, pp. 475−494.

[5] G. Beni and J. Wang, "Swarm intelligence in cellular robotic systems," in *Robots and Biological Systems: Towards a New Bionics?*, P. Dario, G. Sandini, and P. Aebischer, Eds. Berlin, Heidelberg, Germany: Springer, 1993, pp. 703−712.

[6] J. X. Chen, " The evolution of computing: AlphaGo," *Comput. Sci. Eng.*, vol. 18, no. 4, pp. 4–7, Jul.–Aug. 2016.

[7] J. G. Puckett, D. H. Kelley, and N. T. Ouellette, " Searching for effective forces in laboratory insect swarms," *Sci. Rep.*, vol. 4, p. 4766, Apr. 2014.

[8] Y. Tan and K. Ding, " A survey on GPU-based implementation of swarm intelligence algorithms," *IEEE Trans. Cybern.* , vol. 46, no. 9, pp. 2028–2041, Sept. 2016.

[9] M. Dorigo, " Optimization, learning and natural algorithms," Ph.D. dissertation, Politecnico di Milano, Milano, Italy, 1992.

[10] C. Blum and M. Dorigo, "The hyper-cube framework for ant colony optimization," *IEEE Trans. Syst.*, *Man*, *Cybern.*, *Part B* (*Cybern.*), vol. 34, no. 2, pp. 1161–1172, Apr. 2004.

[11] M. Dorigo and C. Blum, "Ant colony optimization theory: A survey," *Theor. Comput. Sci.*, vol. 344, no. 2–3, pp. 243–278, Nov. 2015.

[12] K. M. Sim and W. H. Sun, "Ant colony optimization for routing and load-balancing: Survey and new directions," *IEEE Trans. Syst.*, *Man*, *Cybern.-Part A*: *Syst. Hum.*, vol. 33, no. 5, pp. 560–572, Sept. 2003.

[13] M. Dorigo and G. Di Caro, "Ant colony optimization: A new meta-heuristic," in *Proc. Congr. Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, Washington, USA, 1999, pp. 1470−1477.

[14] H. M. Botee and E. Bonabeau, " Evolving ant colony optimization," *Adv. Complex Syst.*, vol. 1, no. 2–3, pp. 149–159, Jun.–Sept. 1998.

[15] T. Stützle and H. H. Hoos, "MAX–MIN ant system," *Future Generat. Comput. Syst.*, vol. 16, no. 8, pp. 889–914, Jun. 2000.

[16] B. Bullnheimer, R. F. Hartl, and C. Strauss, "A new rank based version of the ant system–A computational study," *Cen. Eur. J. Oper. Res.*, vol. 7, no. 1, pp. 25–38, Jan. 1999.

[17] X. M. Hu, J. Zhang, and Y. Li, "Orthogonal methods based ant colony search for solving continuous optimization problems," *J. Comput. Sci. Technol.*, vol. 23, no. 1, pp. 2–18, Jan. 2008.

[18] D. K. Gupta, Y. Arora, U. K. Singh, and J. P. Gupta, "Recursive ant colony optimization for estimation of parameters of a function," in *Proc. 1st Int. Conf. Recent Advances in Information Technology*, Dhanbad, India, 2012, pp. 448−454.

[19] S. C. Gao, Y. R. Wang, J. J. Cheng, Y. Inazumi, and Z. Tang, "Ant colony optimization with clustering for solving the dynamic location routing problem," *Appl. Math. Comput.*, vol. 285, pp. 149–173, Jul. 2016.

[20] H. Hemmatian, A. Fereidoon, A. Sadollah, and A. Bahreininejad, "Optimization of laminate stacking sequence for minimizing weight and cost using elitist ant system optimization," *Adv. Eng. Softw.*, vol. 57, pp. 8–18, Mar. 2013.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. ICNN'95–Int. Conf. Neural Networks*, Perth, Australia, 1995, pp. 1942−1948.

[22] D. B. Fogel and H. G. Beyer, "A note on the empirical evaluation of intermediate recombination," *Evol. Comput.*, vol. 3, no. 4, pp. 491–495, Dec. 1995.

[23] Y. H. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. 7th Int. Conf. Evolutionary Programming*, San Diego, California, USA, 1998, pp. 591−600.

[24] M. Clerc, *Particle Swarm Optimization*. Hoboken, USA: John Wiley & Sons, 2010.

[25] Z. G. Ren, T. H. Chen, and Z. Z. Wu, "Optimal matching control of a low energy charged particle beam in particle accelerators," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 2, pp. 460–470, Mar. 2019.

[26] R. Poli, J. Kennedy, and T. Blackwell, "Particle swarm optimization," *Swarm Intell.*, vol. 1, no. 1, pp. 33–57, Aug. 2007.

[27] Y. Gao, W. B. Du, and G. Yan, "Selectively-informed particle swarm optimization," *Sci. Rep.*, vol. 5, p. 9295, Mar. 2015.

[28] P. Ghamisi and J. A. Benediktsson, " Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geosci. Remote Sens. Lett.*, vol. 12, no. 2, pp. 309–313, Feb. 2015.

[29] Y. H. Li, Z. H. Zhan, S. J. Lin, J. Zhang, and X. N. Luo, "Competitive and cooperative particle swarm optimization with information sharing mechanism for global optimization problems," *Inf. Sci.* , vol. 293, pp. 370–382, Feb. 2015.

[30] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. Evol. Comput.*, vol. 10, no. 3, pp. 281–295, Jun. 2006.

[31] S. Cui and D. S. Weile, " Application of a parallel particle swarm optimization scheme to the design of electromagnetic absorbers," *IEEE Trans. Antennas Propag.*, vol. 53, no. 11, pp. 3616–3624, Nov. 2005.

[32] J. C. Zeng and Z. H. Cui, "A guaranteed global convergence particle swarm optimizer," *J. Comput. Res. Dev.*, vol. 41, no. 8, pp. 1333–1338, Aug. 2004.

[33] M. Clerc, " Discrete particle swarm optimization, illustrated by the traveling salesman problem," in *New Optimization Techniques in Engineering*, G. C. Onwubolu and B. V. Babu, Eds. Berlin, Heidelberg, Germany: Springer, 2004, pp. 219−239.

[34] H. W. Lin, B. Zhao, D. R. Liu, and C. Alippi, " Data-based fault tolerant control for affine nonlinear systems through particle swarm optimized neural networks," *IEEE/CAA J. Autom. Sinica*, vol. 7, no. 4, pp. 954–964, Jul. 2020.

[35] P. Roy, G. S. Mahapatra, and K. N. Dey, " Forecasting of software reliability using neighborhood fuzzy particle swarm optimization based novel neural network," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 6, pp. 1365–1383, Nov. 2019.

[36] Z. M. Lv, L. Q. Wang, Z. Y. Han, J. Zhao, and W. Wang, "Surrogate-assisted particle swarm optimization algorithm with Pareto active learning for expensive multi-objective optimization," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 3, pp. 838–849, May 2019.

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

[38] W. Shen, X. P. Guo, C. Wu, and D. S. Wu, "Forecasting stock indices using radial basis function neural networks optimized by artificial fish swarm algorithm," *Knowl.-Based Syst.* , vol. 24, no. 3, pp. 378–385, Apr. 2011.

[39] M. Neshat, G. Sepidnam, M. Sargolzaei, and A. N. Toosi, "Artificial fish swarm algorithm: A survey of the state-of-the-art, hybridization, combinatorial and indicative applications," *Artif. Intell. Rev.* , vol. 42, no. 4, pp. 965–997, May 2014.

[40] X. L. Li and J. X. Qian, "Studies on artificial fish swarm optimization algorithm based on decomposition and coordination techniques," *J. Circuits Syst.*, vol. 8, no. 1, pp. 1–6, Feb. 2002.

[41] C. R. Wang, C. L. Zhou, and J. W. Ma, "An improved artificial fish-swarm algorithm and its application in feed-forward neural networks," in *Proc. Int. Conf. Machine Learning and Cybernetics*, Guangzhou, China, 2005, pp. 2890−2894.

[42] X. W. Wang, N. Gao, S. X. Cai, and M. Huang, " An artificial fish swarm algorithm based and ABC supported QoS unicast routing scheme in NGI," in *Proc. Int. Conf. Frontiers of High Performance Computing and Networking*, Sorrento, Italy, 2006, pp. 205−214.

[43] A. K. Azad, A. M. A. Rocha, and E. M. G. P. Fernandes, "Improved

binary artificial fish swarm algorithm for the 0–1 multidimensional knapsack problems," *Swarm Evol. Comput.* , vol. 14, pp. 66–75, Feb. 2014.

[44] Z. Q. Zhang, K. P. Wang, L. X. Zhu, and Y. Wang, " A Pareto improved artificial fish swarm algorithm for solving a multi-objective fuzzy disassembly line balancing problem," *Expert Syst. Appl.*, vol. 86, pp. 165–176, Nov. 2017.

[45] K. C. Zhu and M. Y. Jiang, "Quantum artificial fish swarm algorithm," in *Proc. 8th World Congr. on Intelligent Control and Automation*, Jinan, China, 2010, pp. 1−5.

[46] X. Z. Gao, Y. Wu, K. Zenger, and X. L. Huang, "A knowledge-based artificial fish-swarm algorithm," in *Proc. 13th IEEE Int. Conf. Computational Science and Engineering*, Hong Kong, China, 2010, pp. 327−332.

[47] D. Yazdani, A. N. Toosi, and M. R. Meybodi, " Fuzzy adaptive artificial fish swarm algorithm," in *Proc. 23rd Australasian Joint Conf. Artificial Intelligence*, Adelaide, Australia, 2010, pp. 334–343.

[48] K. M. Passino, " Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Syst. Mag.* , vol. 22, no. 3, pp. 52–67, Jun. 2002.

[49] S. Das, A. Biswas, S. Dasgupta, and A. Abraham, "Bacterial foraging optimization algorithm: Theoretical foundations, analysis, and applications," in *Foundations of Computational Intelligence Volume 3*, A. Abraham, A. E. Hassanien, P. Siarry, and A. Engelbrecht, Eds. Berlin, Heidelberg, Germany: Springer, 2009, pp. 23−55.

[50] E. S. Ali and S. M. Abd-Elazim, " Bacteria foraging optimization algorithm based load frequency controller for interconnected power system," *Int. J. Electr. Power Energy Syst.*, vol. 33, pp. 633–638, 2011.

[51] K. R. Devabalaji and K. Ravi, "Optimal size and siting of multiple DG and DSTATCOM in radial distribution system using bacterial foraging optimization algorithm," *Ain Shams Eng. J.*, vol. 7, no. 3, pp. 959–971, Sept. 2016.

[52] R. Majhi, G. Panda, B. Majhi, and G. Sahoo, "Efficient prediction of stock market indices using adaptive bacterial foraging optimization (ABFO) and BFO based techniques," *Expert Syst. Appl.*, vol. 36, no. 6, pp. 10097–10104, Aug. 2009.

[53] B. Niu, H. Wang, J. W. Wang, and L. J. Tan, "Multi-objective bacterial foraging optimization," *Neurocomputing* , vol. 116, pp. 336–345, Sept. 2013.

[54] H. N. Chen, Y. L. Zhu, and K. Y. Hu, "Multi-colony bacteria foraging optimization with cell-to-cell communication for RFID network planning," *Appl. Soft Comput.*, vol. 10, no. 2, pp. 539–547, Mar. 2010.

[55] S. V. R. S. Gollapudi, S. S. Pattnaik, O. P. Bajpai, S. Devi, and K. M. Bakwad, " Velocity modulated bacterial foraging optimization technique (VMBFO)," *Appl. Soft Comput.*, vol. 11, no. 1, pp. 154–165, Jan. 2011.

[56] D. H. Kim, A. Abraham, and J. H. Cho, "A hybrid genetic algorithm and bacterial foraging approach for global optimization," *Inf. Sci.*, vol. 177, no. 18, pp. 3918–3937, Sept. 2007.

[57] D. Karaboga, " An idea based on honey bee swarm for numerical optimization," Erciyes Univ., Kayseri, Turkey, Tech. Rep. TR06, 2005.

[58] D. Karaboga and B. Akay, " A comparative study of artificial bee colony algorithm," *Appl. Math. Comput.*, vol. 214, no. 1, pp. 108–132, Aug. 2009.

[59] J. C. Bansal, H. Sharma, and S. S. Jadon, " Artificial bee colony algorithm: A survey," *Int. J. Adv. Intell. Paradig.*, vol. 5, no. 1–2, pp. 123–159, 2013.

[60] B. Akay and D. Karaboga, "A survey on the applications of artificial bee colony in signal, image, and video processing," *Signal*, *Image Video Process.*, vol. 9, no. 4, pp. 967–990, Mar. 2015.

[61] A. Kaur and S. Goyal, " A survey on the applications of bee colony optimization techniques," *Int. J. Comput. Sci. Eng.*, vol. 3, no. 8, pp. 3037–3046, Aug. 2011.

[62] W. F. Gao and S. Y. Liu, "A modified artificial bee colony algorithm," *Comput. Oper. Res.*, vol. 39, no. 3, pp. 687–697, Mar. 2012.

[63] A. Banharnsakun, T. Achalakul, and B. Sirinaovakul, "The best-so-far selection in artificial bee colony algorithm," *Appl. Soft Comput.*, vol. 11, no. 2, pp. 2888–2901, Mar. 2011.

[64] P. W. Tsai, J. S. Pan, B. Y. Liao, and S. C. Chu, "Enhanced artificial bee colony optimization," *Int. J. Innovative Comput.*, *Inf. Control*, vol. 5, no. 12, pp. 5081–5092, Dec. 2009.

[65] C. Ozturk, E. Hancer, and D. Karaboga, " Dynamic clustering with improved binary artificial bee colony algorithm," *Appl. Soft Comput.*, vol. 28, pp. 69–80, Mar. 2015.

[66] G. P. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Appl. Math. Comput.*, vol. 217, no. 7, pp. 3166–3173, Dec. 2010.

[67] Q. K. Pan, M. F. Tasgetiren, and P. N. Suganthan, and T. J. Chua, "A discrete artificial bee colony algorithm for the lot-streaming flow shop scheduling problem," *Inf. Sci.* , vol. 181, no. 12, pp. 2455–2468, Jun. 2011.

[68] M. S. Kımathran and O. Fımathndımathk, " A directed artificial bee colony algorithm," *Appl. Soft Comput.*, vol. 26, pp. 454–462, Jan. 2015.

[69] D. Karaboga and B. Gorkemli, "A combinatorial artificial bee colony algorithm for traveling salesman problem," in *Proc. Int. Symp. Innovations in Intelligent Systems and Applications*, Istanbul, Turkey, 2011, pp. 50−53.

[70] J. K. Ji, S. B. Song, C. Tang, S. C. Gao, Z. Tang, and Y. Todo, "An artificial bee colony algorithm search guided by scale-free networks," *Inf. Sci.*, vol. 473, pp. 142–165, Jan. 2019.

[71] D. D. Zhang, G. M. Xie, J. Z. Yu, and L. Wang, " Adaptive task assignment for multiple mobile robots via swarm intelligence approach," *Rob. Auton. Syst.*, vol. 55, no. 7, pp. 572–588, Jul. 2007.

[72] J. Timmis, P. Andrews, and E. Hart, " On artificial immune systems and swarm intelligence," *Swarm Intell.*, vol. 4, no. 4, pp. 247–273, Sept. 2010.

[73] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, " Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Adv. Eng. Softw.*, vol. 114, pp. 163–191, Dec. 2017.

[74] K. Z. Gao, Z. G. Cao, L. Zhang, Z. H. Chen, Y. Y. Han, and Q. K. Pan, "A review on swarm intelligence and evolutionary algorithms for solving flexible job shop scheduling problems," *IEEE/CAA J. Autom. Sinica*, vol. 6, no. 4, pp. 904–916, Jul. 2019.

[75] X. S. Yang, S. Deb, Y. X. Zhao, S. Fong, and X. S. He, "Swarm intelligence: Past, present and future," *Soft Comput.* , vol. 22, no. 18, pp. 5923–5933, Sept. 2018.

[76] A. Slowik and H. Kwasnicka, " Nature inspired methods and their industry applications—swarm intelligence algorithms," *IEEE Trans. Ind. Inform.*, vol. 14, no. 3, pp. 1004–1015, Mar. 2018.

[77] X. S. Yang and S. Deb, "Cuckoo search via Láevy flights," in *Proc. World Congr. Nature & Biologically Inspired Computing*, Coimbatore, India, 2009, pp. 210−214.

[78] H. B. Duan and P. X. Qiao, " Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning," *Int. J. Intell. Comput. Cybern.*, vol. 7, no. 1, pp. 24–37, Mar. 2014.

[79] X. S. Yang and A. H. Gandomi, "Bat algorithm: A novel approach for global engineering optimization," *Eng. Comput.* , vol. 29, no. 5, pp. 464–483, Jul. 2012.

[80] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[81] S. A. Hofmeyr and S. Forrest, "Architecture for an artificial immune system," *Evol. Comput.*, vol. 8, no. 4, pp. 443–473, Dec. 2000.

[82] W. T. Pan, " A new Fruit Fly Optimization Algorithm: Taking the

financial distress model as an example," *Knowl.-Based Syst.*, vol. 26, pp. 69–74, Feb. 2012.

[83] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimization for simultaneous capture of multiple local optima of multimodal functions," *Swarm Intell.*, vol. 3, no. 2, pp. 87–124, Jun. 2009.

[84] A. R. Mehrabian and C. Lucas, " A novel numerical optimization algorithm inspired from weed colonization," *Ecol. Inform.*, vol. 1, no. 4, pp. 355–366, Dec. 2006.

[85] Y. R. Wang, Y. Yu, S. Y. Cao, X. Y. Zhang, and S. C. Gao, "A review of applications of artificial intelligent algorithms in wind farms," *Artif. Intell. Rev.*, vol. 53, no. 5, pp. 3447–3500, Jun. 2020.

[86] M. L. Pinedo, *Scheduling*. New York, USA: Springer, 2012.

[87] P. B. Myszkowski, M. E. Skowroński, L. P. Olech, and K. Oślizło, "Hybrid ant colony optimization in solving multi-skill resource-constrained project scheduling problem," *Soft Comput.*, vol. 19, no. 12, pp. 3599–3619, Dec. 2015.

[88] J. García-Nieto, E. Alba, and A. C. Olivera, "Swarm intelligence for traffic light scheduling: Application to real urban areas," *Eng. Appl. Artif. Intell.*, vol. 25, no. 2, pp. 274–283, Mar. 2012.

[89] Q. K. Pan, " An effective co-evolutionary artificial bee colony algorithm for steelmaking–continuous casting scheduling," *Eur. J. Oper. Res.*, vol. 250, no. 3, pp. 702–714, May 2016.

[90] K. P. Kumar, B. Saravanan, and K. S. Swarup, " Optimization of renewable energy sources in a microgrid using artificial fish swarm algorithm," *Energy Proced.*, vol. 90, pp. 107–113, Dec. 2016.

[91] A. B. Bamini and S. Enoch, " Optimized resource scheduling using classification and regression tree and modified bacterial foraging optimization algorithm," *Int. J. Appl. Eng. Res.*, vol. 10, no. 16, pp. 37170–37175, Sept. 2015.

[92] J. Kennedy and R. Eberhart, " Swarm intelligence in cellular robotic systems," in *Proc. IEEE Int. Conf. Neural Networks*, Piscataway, USA, 1995, pp. 1942−1948.

[93] G. Z. Tan, H. He, and A. Sloman, "Ant colony system algorithm for real-time globally optimal path planning of mobile robots," *Acta Autom. Sinica*, vol. 33, no. 3, pp. 279–285, Mar. 2007.

[94] M. A. Contreras-Cruz, V. Ayala-Ramirez, and U. H. Hernandez-Belmonte, "Mobile robot path planning using artificial bee colony and evolutionary programming," *Appl. Soft Comput.*, vol. 30, pp. 319–328, May 2015.

[95] Z. H. Yao and Z. H. Ren, "Path planning for coalmine rescue robot based on hybrid adaptive artificial fish swarm algorithm," *Int. J. Control Autom.*, vol. 7, no. 8, pp. 1–12, Aug. 2014.

[96] J. Pugh, A. Martinoli, and Y. Zhang, "Particle swarm optimization for unsupervised robotic learning," in *Proc. IEEE Swarm Intelligence Symp.*, Pasadena, USA, 2005, pp. 92−99.

[97] M. Aghajarian, K. Kiani, and M. M. Fateh, "Design of fuzzy controller for robot manipulators using bacterial foraging optimization algorithm," *J. Intell. Learn. Syst. Appl.*, vol. 4, no. 1, pp. 53–58, Feb. 2012.

[98] H. Bai and B. Zhao, "A survey on application of swarm intelligence computation to electric power system," in *Proc. 6th World Congr. Intelligent Control and Automation*, Dalian, China, 2006, pp. 7587−7591.

[99] J. Z. Zhou, C. Wang, Y. Z. Li, P. Wang, C. L. Li, P. Lu, and L. Mo, "A multi-objective multi-population ant colony optimization for economic emission dispatch considering power system security," *Appl. Math. Model.*, vol. 45, pp. 684–704, May 2017.

[100] M. A. Abido, " Optimal design of power-system stabilizers using particle swarm optimization," *IEEE Trans. Energy Conver.*, vol. 17, no. 3, pp. 406–413, Sept. 2002.

[101] H. Gozde, M. C. Taplamacioglu, and I. Kocaarslan, "Comparative performance analysis of artificial bee colony algorithm in automatic generation control for interconnected reheat thermal power system," *Int. J. Electr. Power Energy Syst.*, vol. 42, pp. 167–178, 2012.

[102] C. Li and S. L. Wang, " Next-day power market clearing price forecasting using artificial fish-swarm based neural network," in *Proc. 3rd Int. Symp. Neural Networks*, Chengdu, China, 2006, pp. 1290−1295.

[103] E. S. Ali and S. M. Abd-Elazim, " BFOA based design of PID controller for two area load frequency control with nonlinearities," *Int. J. Electr. Power Energy Syst.*, vol. 51, pp. 224–231, Oct. 2013.

[104] H. B. Duan, " Aerial robot formation control via pigeon-inspired optimization," in *Robotic Systems: Concepts, Methodologies, Tools, and Applications*, Information Resources Management Association, Ed. Hershey, USA: IGI Global, 2020, pp. 1143−1180.

[105] X. L. Zhang, X. F. Chen, and Z. J. He, "An ACO-based algorithm for parameter optimization of support vector machines," *Expert Syst. Appl.*, vol. 37, no. 9, pp. 6618–6628, Sept. 2010.

[106] S. K. Panda, S. Padhee, A. K. Sood, and S. S. Mahapatra, "Simulation and parameter optimization of flux cored ARC welding using artificial neural network and particle swarm optimization algorithm," *J. Intell. Manuf.*, vol. 25, no. 1, pp. 67–76, Feb. 2014.

[107] B. Akay and D. Karaboga, "A modified artificial bee colony algorithm for real-parameter optimization," *Inf. Sci.*, vol. 192, pp. 120–142, Jun. 2012.

[108] Z. Cheng and X. Hong, "PID controller parameters optimization based on artificial fish swarm algorithm," in *Proc. 5th Int. Conf. Intelligent Computation Technology and Automation*, Zhangjiajie, China, 2012, pp. 265−268.

[109] S. K. Panda, S. Padhee, and A. K. Sood, S. S. Mahapatra, "Optimization of fused deposition modelling (FDM) process parameters using bacterial foraging technique," *Intell. Inf. Manage.*, vol. 1, no. 2, pp. 89–97, Nov. 2009.

[110] L. Ljung, *System Identification: Theory for the User*. Upper Saddle River, USA: PTR Prentice Hall, 1999, pp. 1−14.

[111] S. H. Tsai and Y. W. Chen, " A novel fuzzy identification method based on ant colony optimization algorithm," *IEEE Access*, vol. 4, pp. 3747–3756, Jun. 2016.

[112] A. Alfi and H. Modares, " System identification and control using adaptive particle swarm optimization," *Appl. Math. Model.*, vol. 35, no. 3, pp. 1210–1221, Mar. 2011.

[113] W. Hu, Y. G. Yu, and S. Zhang, " A hybrid artificial bee colony algorithm for parameter identification of uncertain fractional-order chaotic systems," *Nonlinear Dyn.*, vol. 82, no. 3, pp. 1441–1456, Jul. 2015.

[114] W. Han, H. H. Wang, and L. Chen, " Parameters identification for photovoltaic module based on an improved artificial fish swarm algorithm," *Sci. World J.*, vol. 2014, pp. 1–12, Aug. 2014.

[115] B. Majhi and G. Panda, " Development of efficient identification scheme for nonlinear dynamic systems using swarm intelligence techniques," *Expert Syst. Appl.*, vol. 37, no. 1, pp. 556–566, Jan. 2010.

[116] M. Sonka, V. Havac, and R. Boyle, *Image Processing, Analysis, and Machine Vision*, Boston, USA: Cengage Learning, 2014.

[117] J. Tian, W. Y. Yu, and S. L. Xie, " An ant colony optimization algorithm for image edge detection," in *Proc. IEEE Congr. Evolutionary Computation*, Hong Kong, China, 2008, pp. 751−756.

[118] M. G. H. Omran, A. Salman, and A. P. Engelbrecht, "Dynamic clustering using particle swarm optimization with application in image segmentation," *Pattern Anal. Appl.*, vol. 8, no. 4, Article No. 332, Feb. 2006.

[119] A. Draa and A. Bouaziz, "An artificial bee colony algorithm for image contrast enhancement," *Swarm Evol. Comput.*, vol. 16, pp. 69–84, Jun. 2014.

[120] S. A. El-Said, " Image quantization using improved artificial fish

swarm algorithm," *Soft Comput.*, vol. 19, no. 9, pp. 2667–2679, Sept. 2015.

[121] M. Hanmandlu, O. P. Verma, N. K. Kumar, and M. Kulkarni, "A novel optimal fuzzy system for color image enhancement using bacterial foraging," *IEEE Trans. Instrum. Meas.*, vol. 58, no. 8, pp. 2867–2879, Aug. 2009.

[122] D. Merkle and M. Middendorf, " Swarm intelligence and signal processing [DSP Exploratory]," *IEEE Signal Process. Mag.*, vol. 25, no. 6, pp. 152–158, Nov. 2008.

[123] X. Huang, R. Fernandez-Rojas, K. L. Ou, and A. C. Madoc, "Novel signal processing of brain activity based on ant colony optimization and wavelet analysis with near infrared spectroscopy," in *Proc. 8th IEEE Int. Conf. Communication Software and Networks*, Beijing, China, 2016, pp. 81−85.

[124] S. Shadmand and B. Mashoufi, " A new personalized ECG signal classification algorithm using block-based neural network and particle swarm optimization," *Biomed. Signal Process. Control*, vol. 25, pp. 12–23, Mar. 2016.

[125] T. Koza and N. Karaboga, " Quadrature mirror filter bank design for mitral valve doppler signal using artificial bee colony algorithm," *Elektron. Elektrotechn.*, vol. 23, no. 1, pp. 57–62, Dec. 2017.

[126] M. Y. Jiang and D. F. Yuan, " Wavelet threshold optimization with artificial fish swarm algorithm," in *Proc. Int. Conf. Neural Networks and Brain*, Beijing, China, 2005, pp. 569−572.

[127] G. Sahu, B. Biswal, and A. Choubey, "Non - stationary signal classification via modified fuzzy C - means algorithm and improved bacterial foraging algorithm," *Int. J. Nume. Model.: Electron. Networks, Devices Fields*, vol. 30, no. 2, Mar.–Apr. 2017.

Jun Tang is currently an Associate Professor at the Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology. He is the Winner of William Sweet Smith Prize in 2015. He was dedicated to the Ph.D. researches in the Technical Innovation Cluster on Aeronautical Management, Universitat Autònoma de Barcelona. His research interests include logistic systems, causal modelling, state space, air traffic management and discrete event simulation (DES). He has been very active in the simulation community, organizing as General Co-Chair for several international conferences.

Gang Liu received the B.Sc. and master degrees from Naval Arms Command Institute in 2005 and 2008, respectively, and the Ph.D. degree from National University of Defense Technology in 2013. His research interests include intelligent information processing, optimization and path planning.

Qingtao Pan received the B.Sc. degree at the College of Mathematical Sciences, Ocean University of China, in 2020. He is currently working toward the M.S. degree at the College of Systems Engineering, National University of Defense Technology, majoring in control science and engineering. His research interests include intelligent optimization algorithm, system simulation and swarm intelligence.