Nature Inspired Methods and Their Industry Applications—Swarm Intelligence Algorithms

Adam Slowik, *Senior Member, IEEE*, and Halina Kwasnicka

***Abstract*—In this paper, we present the swarm intelligence (SI) concept and mention some metaheuristics belonging to the SI. We present the particle swarm optimization (PSO) algorithm and the ant colony optimization (ACO) method as the representatives of the SI approach. In recent years, researchers are eager to develop and apply a variety of these two methods, despite the development of many other newer methods as Bat or FireFly algorithms. Presenting the PSO and ACO we put their pseudocode, their properties, and intuition lying behind them. Next, we focus on their real-life applications, indicating many papers presented varieties of basic algorithms and the areas of their applications.**

***Index Terms*—Ant colony optimizations (ACOs), metaheuristics, particle swarm optimization (PSO), real-life applications, swarm intelligence (SI).**

I. Introduction

OPTIMIZATION tasks are very popular with real-life problems, therefore developing skill and efficient optimization methods is still an interesting and intensively developing research area. Fig. 1 presents some approaches in optimization techniques. One of the important categories of optimization methods based on observations of nature is so-called swarm intelligence (SI). Different taxonomy can be found in the literature on a different level of generality. Fister *et al.* [1] present that *SI-based* $\subset$ *bio-inspired* $\subset$ *nature-inspired methods*. We show bioinspired methods in more detail. We list a few examples of SI methods; from the remaining bioinspired methods, we have separated a very popular group of methods based on the biological evolution paradigm. We also show examples of other methods based on nature. Nature-inspired methods are shown against the context of general optimization methods (formal and heuristics).

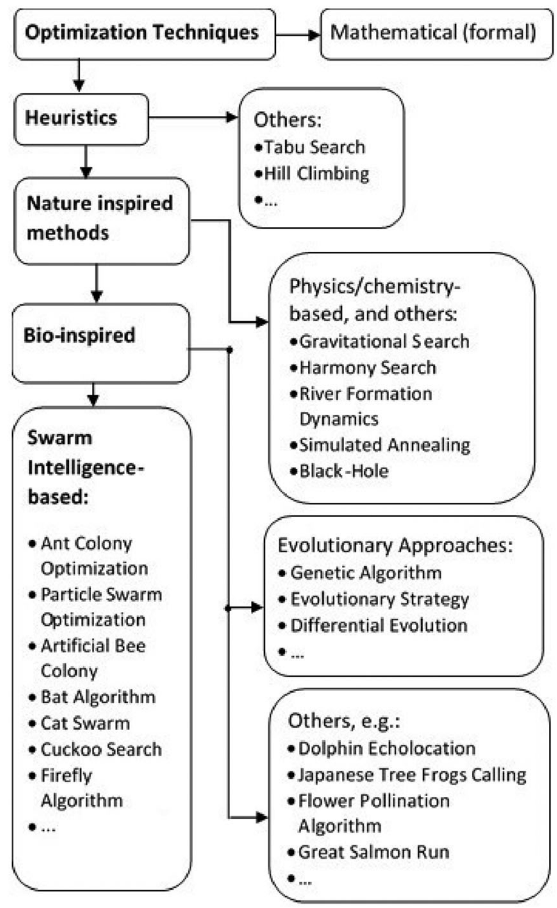

Fig. 1. Taxonomy of nature-inspired methods.

SI can be perceived as a kind of the wider concept—collective intelligence (CI). The good definition of CI gave Malone (conversation given in 2012, [2]): "It is important to realize that intelligence is not just something that happens inside individual brains. It also arises with groups of individuals. In fact, I would define Collective Intelligence as groups of individuals acting collectively in ways that seem intelligent."

A system in which coherent functional global patterns emerge from interacted (locally) agents in their environment is called SI [3]. A simple behavior of particular agents and self-organizing interaction among them are observed in nature, e.g., ant colonies and bird flocking. Such behaviors were the inspiration for developing "artificial colonies of agents" able to solve difficult optimization problems. *Self-organization* and *division of labor* are the two fundamental concepts necessary to obtain swarm intelligent behavior. According to [3], the term *swarm intelligent* was first time used by Beni in [4], where the system

Manuscript received October 3, 2017; revised November 27, 2017; accepted December 17, 2017. Date of publication December 25, 2017; date of current version March 1, 2018. Paper no. TII-17-2337. *(Corresponding author: Adam Slowik.)*

A. Slowik is with the Department of Electronics and Computer Science, Koszalin University of Technology, Koszalin 75-453, Poland (e-mail: aslowik@ie.tu.koszalin.pl).

H. Kwasnicka is with the Department of Computational Intelligence, Wroclaw University of Science and Technology, Wroclaw 51-370, Poland (e-mail: halina.kwasnicka@pwr.edu.pl).

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/TII.2017.2786782

composed of autonomous robots cooperated to fulfill the task is studied.

Many heuristics have been developed that imitate the behavior of different species, but particle swarm optimization (PSO) [5]–[8] and ant colony optimization (ACO) [9], [10] are perceived as the mainstream algorithms belonging to the SI group [3]; therefore, in this paper we focus on them.

The main goal of this paper is to present the possible applications of the considered metaheuristics in different areas of industry and, in general, in an economy. In the next sections, we present the idea of PSO, ACO, and their modifications. Also, some other algorithms belonging to the SI group are mentioned to show the variety of approaches. Section III contains the description of the various uses of the considered methods in selected areas. Recent advances and the current trends of these metaheuristics are described at the end of this paper.

II. Brief Presentation of the Selected SI Methods

In this section, we focus on the most popular approaches—PSO and ACO. Also, some words about other methods are included to show the variety of approaches.

A. PSO Algorithm

PSO was proposed in 1995 by Kennedy and Eberhart [5]. It is a stochastic optimization method, which simulates the social behavior of animals. The important role plays *cooperation*—each member changes its search pattern according to its own and other members experiences. The PSO algorithm, took the root from evolutionary algorithms [17], [18], namely use a swarm mode (population based) of searching the large solution space. The second "seed" of PSO is artificial life [19]—the use of artificial systems with life characteristics, such as sensing the quality change in the environment and respond, changing the behavior mode only when it is worthy, etc. [20]. The algorithm consists of a swarm of particles. Each single particle is a solution in the solution space; the solution is determined by the fitness function. The PSO starts with random particles. Next, each particle moves in the direction dependent on its current movement, the personal best, and the global best values. New best values get updated after each iteration. The general schema of PSO is similar to the genetic algorithm (GA). Over the years, researchers have proposed the variety of its modifications. Algorithm 1 presents the pseudocode of the original PSO. Each particle is a D-dimensional vector, where D is a dimension (a number of variables) of the task being optimized. The PSO algorithm is searching for the optimal solution by creating subsequent generations of solutions. The particle's velocity and position are given by the following equations:

$$V(t) = V(t-1) + c_1 \cdot r_1 \cdot (P_{\text{best}} - X(t-1)) + c_2 \cdot r_2 \cdot (G_{\text{best}} - X(t-1)) \quad (1)$$

$$X(t) = X(t-1) + V(t) \quad (2)$$

where $V(t)$ and $X(t)$ are the velocity and the position of the particle, respectively; P_{best} and G_{best} are the personal best and the global best values up to the current iteration; r_1 and r_2 are

Algorithm 1: Pseudo code of the original version of PSO.

initialize the swarm, it could be randomly
evaluate particles in the swarm – calculate their fitness
select the best particle G_{best} – the best fitness value in the swarm
while the stop criterion is not meet **do**
  for each particle P_i **do**
    evaluate particle P_i – calculate its fitness
    if fitness value of P_i is better than the personal best in history (P_{best}) **then** update P_{best} with the use of P_i
    end if
    if fitness value of P_i is better than the fitness value of G_{best} **then** update G_{best} with the use of P_i
    end if
  end for
  for each particle P_i **do**
    calculate particle velocity according to equation (1)
    update particle position according to equation (2)
  end for
end while

random from the range [0, 1) with uniform distribution, these parameters introduce randomness to the method; and c_1 and c_2 are the coefficients of the particle acceleration (learning factors); values are chosen experimentally from the range [0, 2].

The first component in (1) can be perceived as inertia, momentum, or habit. It reflects the particles tendency to continue moving in the same direction. The second component reflects the linear tendency to follow the best position found in the past by the particle. The last component is similar to the previous, but it takes into account the best position found in the past by all the particles. It can be referred as "cooperation," "social knowledge," "group knowledge," or "shared information."

There exist two versions of PSO algorithms: *local* (LPSO) and *global* (GPSO) algorithm [7], [22]. The LPSO algorithm is especially needed in the optimization problems where more than one global solution exist. Due to the LPSO algorithm, the localization of all global optimum is possible in one run of the algorithm. The GPSO algorithm always searches only one—the best global solution [8]. The improved version of the PSO is presented in [21] where the randomly selected neighbors learning strategy was introduced in the early stage of PSO.

The primary version of PSO is dedicated only for continuous optimization problems without constraints. Currently, many modifications have been developed [22], [23]. Due to these modifications, we can use PSO algorithm for very wide group of optimization problems, e.g., for constraints optimization [24], [25], combinatorial optimization (CO) [26], [27], and multiobjective optimization [28], [29]. Also, many modifications of the PSO algorithm, improving its convergence, have been elaborated. Among these modifications, we can mention velocity clamping, inertia weight, and constriction coefficient. Velocity clamping is introduced to assure that the new positions of particles will not be located outside of the acceptable space for a given search space. The inertia weight is introduced to better control the particle swarm ability of exploration (searching over the whole solution space) and exploitation (searching in the neighborhood of "good" solutions). The main task of constriction coefficient is balancing of the PSO algorithm properties between global and local searching of the solution space. Similarly as in the evolutionary algorithms, many open research problems are present in the PSO approaches, such as self-adaptation of parameters in standard and multiobjective PSO, increasing the efficiency of the PSO algorithm, the study on a convergence of multiobjective PSO, and applications of PSO in dynamic optimization problems. We believe that these problems are very promising paths for future research in the area of PSO paradigm.

B. Ant Colony Optimization

Numerous real-life problems belong to the CO. In a CO problem, $P = (A, f)$, we have given a set of objects S (also called the search space) and the objective function $f : A \rightarrow R^+$ that assigns a cost value to each object $s \in S$. The goal is to minimize the cost value [10]. ACO is one of the most recent techniques for CO problems; it was introduced in the early 1990s by Dorigo [9], [73], [74]. ACO is a search technique inspired by the SI of ants colonies using pheromone as a mean of chemical messenger. When ants search for food, they initially explore randomly the area surrounding their nest. Moving ants leave a pheromone trail on the ground, this pheromone evaporates in time. Next, ants choose their paths with probability depended on the pheromone concentrations on the explored paths. Such indirect communication enables ants to find the shortest paths between the nest and source of food.

The general pseudocode of the basic ACO is presented in Algorithm 2.

Often, in ACO algorithms, the transition probabilities (step 6 in the pseudocode) are defined by the following equation:

$$p(c_i|s) = \frac{[\tau_i]^\alpha \cdot [\eta(c_i)]^\beta}{\sum_{c_j \in N(s)} [\tau_j]^\alpha \cdot [\eta(c_j)]^\beta} \quad \forall c_i \in N(s) \tag{3}$$

where η is an optional weighting function, which assigns a heuristic value $\eta(c_j)$ to each feasible solution component $c_j \in N(s)$; τ_i is the pheromone level on the corresponding path; and α and β are the positive parameters, which determine the relation between pheromone information and heuristic information. The way of pheromone updating differs ACO variants. Pheromone evaporation uniformly decreases all the pheromone values. Equation (4) gives a frequently used formula for increasing the values of pheromone trail on solution components that are part of these solutions

$$\tau_i \leftarrow (1 - \rho) \cdot \tau_i + \rho \cdot \sum_{s \in S_{\text{upd}} | c_i \in s} w_s \cdot F(s) \tag{4}$$

where S_{upd} is the set of solutions used for the update, $\rho \in (0, 1]$ is an *evaporation rate* parameter, $F : S \rightarrow R^+$ is a quality function, and $w_s \in R^+$ is the weight of a solution s.

Algorithm 2: Pseudo code of the ACO.

1: for the CO problem to be solved, derive the finite set $C = \{c_1, c_2, \ldots, c_n\}$ of solution components (they are used to assembly solution to the CO)
2: define the pheromone values T (the pheromone model, it is a parameterized probabilistic model), the pheromone values $\tau_i \in T$ are linked with solution components
3: **while** termination conditions not met **do**
4:     $s = []$ – an empty sequence of solution components (ant based solution construction starts)
5:     **while** $N(s) \neq \emptyset$ **do**
6:         $c \leftarrow$ choose from $(N(s))$ – add a feasible solution component at each construction step with respect to the pheromone model (equation 3 can be used)
7:         determine $N(s)$, $N(s) \subseteq C \setminus s$, the specification of $N(s)$ depends on the solution construction mechanism
8:     **end while**
9:     pheromone update
10: **end while**

The pseudocode of ACO strongly depends on the considered problem. The two main steps in ACO approach are: construct candidate solutions using the pheromone model (the parameterized probability distribution over the solution space) and modify the pheromone values using candidate solutions. The amount of pheromone left on the path traveled and the rate of its evaporation are responsible for the balance between exploitation and exploration. This problem is present in all the nature inspired methods. Therefore, such approaches require being adapted and tuned to the problem solved. Strong preference of the better paths causes premature convergence to the possibly local optimum. Choice of the proper values of parameter for evaporation and/or adding the pheromone decides about the quality of ACO. Parameter values suited for one problem usually are not well for other, even similar problems, e.g., for a problem of the same type but of a greater dimension. For decades, a variety of ACO approaches have been developed, e.g., for constraint-satisfaction problems [75], for continuous functions [76], [77], for multiobjective problem [78], hybrid ACO with local search [79], and many others.

C. Few Words About Other SI Approaches

Some other algorithms belonging to the SI group are listed in Table I. Few of them are mentioned here to show the variety of approaches.

The artificial bee colony (ABC) [11], [12] was developed by Karaboga in 2005. It uses a colony of bees divided into three groups. Bees belonging to each group play different role: *employed* (*forager*) *bees*—they are associated with a particular food source that they are currently exploiting, and they carry with them information about this particular source (as its distance and direction from the nest, the profitability) and share their information with a probability proportional to the

TABLE I
IN ALL, 38 ALGORITHMS SELECTED FROM THE WHOLE FAMILY OF THE SWARM ALGORITHMS

Algorithm, Ref.	Author	Inspiration	Year
ACO, [97]	Dorigo	Real ants colonies using pheromone as a mean of chemical messenger	1992
PSO, [5]	Kennedy *et al.*	Social behavior of bird flocking or fish schooling	1995
Bee system, [100]	Lucic *et al.*	Foraging behavior of bee colonies	2001
Bacterial foraging, [13]	Passino	Social foraging behavior of Escherichia coli	2002
Fish-swarm algorithm, [108]	Li *et al.*	Fish behaviors such as praying and swarming	2002
Beehive, [101]	Wedde *et al.*	Communicative and evaluative methods and procedures of honey bees	2004
Bacterial colony chemotaxis, [127]	Li *et al.*	Bacteriums reaction to chemoattractants	2005
BCO, [99]	Teodorovic *et al.*	Bee colonies in nature	2005
Bees swarm optimization, [102]	Drias *et al.*	Behavior of real bees in nature	2005
Virtual bees, [111]	Yang	Swarm of bees and interactions between them when they find nectar	2005
Cat swarm, [104]	Chu *et al.*	Behaviors of cats and their skills such as tracing and seeking	2006
ABC, [12]	Karaboga *et al.*	Natural foraging behavior of real honey bees	2007
Fast bacterial swarming, [107]	Chu *et al.*	Foraging mechanism of Escherichia coli and the swarming pattern of birds	2008
Bumblebees, [103]	Comellas *et al.*	Collective behavior of social insects	2009
Cuckoo search, [106]	Yang *et al.*	Brood parasitic behavior of some cuckoo species	2009
FA, [15]	Yang	Behavior of fireflies and their flashing light (process of bioluminescence)	2009
Glowworm swarm optimization, [109]	Krishnanand *et al.*	Luciferin induced glow of a glowworm which is used to attract mates/prey	2009
Artificial fish school algorithm, [126]	Hu *et al.*	Fish behaviors such as praying, swarming, following	2010
Bat algorithm, [98]	Yang	Echolocation characteristics of microbats	2010
Cockroach swarm optimization, [114]	Chen *et al.*	Social behavior of cockroaches	2010
Hunting search, [119]	Oftadeh *et al.*	Group hunting of animals such as lions, wolves, and dolphins	2010
Bacterial colony optimization, [128]	Niu *et al.*	Five basic behaviors of Escherichia coli bacteria in their whole lifecycle	2012
Blind-naked mole-rats, [125]	Taherdangkoo *et al.*	Social behavior of the blind naked mole-rats colony	2012
Krill herd, [110]	Gandomi *et al.*	Herding behavior of krill individuals	2012
Lion's algorithm, [123]	Rajakumar	Lion's social behavior that aids to keep the mammal be strong in the world	2012
Wolf search, [112]	Tang *et al.*	Wolves search for food and survive by avoiding their enemies	2012
Fruit fly optimization, [118]	Xing *et al.*	Behavior of fruit flies	2013
Social spider optimization, [116]	Cuevas *et al.*	Cooperative behavior of social-spiders that interact to each other	2013
Swarm dolphin algorithm, [138]	Chen	Social behaviors of dolphin	2013
Artificial wolf pack algorithm, [139]	Chen	Social behaviors of the wolf pack in: scouting, calling, and besieging	2013
Elephant herding, [115]	Wang *et al.*	Herding behavior of the elephant groups	2015
Monarch butterfly optimization, [120]	Wang *et al.*	Migration of monarch butterflies	2015
Crow search, [122]	Askarzadeh	Intelligent behavior of crows	2016
Dolphin swarm algorithm, [113]	Wu *et al.*	Dolphins echolocation, information exchanges, cooperation	2016
Dynamic virtual Bats algorithm, [121]	Topal *et al.*	Bat's ability to manipulate frequency/wavelength of the emitted sound waves	2016
Whale optimization algorithm, [117]	Mirjalili *et al.*	Social behavior of humpback whales—the bubble-net hunting strategy	2016
Grasshopper optimization, [124]	Saremi *et al.*	Behavior of grasshopper swarms in nature	2017
Spotted hyena optimizer, [105]	Dhiman *et al.*	Social relationship between spotted hyenas and their collaborative behavior	2017

profitability of the food source; *onlooker* (*observer*) *bees*—they wait in the nest and establish a food source through the information shared by employed foragers; and *scouts*—these bees continually look out for a food source to exploit. Communication among bees about the quality of food sources takes place in the dancing area. More information is circulated about the more profitable sources, therefore a probability of onlookers choosing more profitable sources is greater. Randomization is carried out by scout and employed bees, mainly by mutation.

The bacterial foraging optimization [13] approach is based on the assumption that animals search for and obtain nutrients in such a way that maximizes their energy intake per unit time spent foraging. Foraging strategies are the methods for locating, handling, and ingesting food. A foraging animal takes actions under constraints presented by its own physiology (e.g., sensing and cognitive capabilities) and environment (e.g., density of prey, physical characteristics of the search area, risks from predators). This idea leaded scientists to use this approach as an optimization method.

The FireFly algorithm (FA) [14], [15] is based on the flashing patterns and behavior of tropical fireflies. The FA algorithm is simple, flexible, and easy to implement. It uses the following three idealized rules:

1) all fireflies are unisex (a FireFly will be attracted to other fireflies regardless of their sex);
2) the attractiveness is proportional to their brightness, and they both decrease as their distance increases (if for some particular FireFly there is no brighter one, it moves randomly); and
3) the brightness of a FireFly is affected or determined by the objective function.

The fish swarm optimization (FSO) [16] is based on the behavior of fish swarm in search for food. The following types of behavior can be considered: *random behavior*—when fish looks at random for food and other companion; *searching behavior*—when the fish discovers a region with more food; *swarming behavior*—fish swarms naturally in order to avoid danger; *chasing behavior*—when a fish discovers food, the others will find the food dangling after it; and *leaping behavior*—it is when fish stagnates in a region and it is required a move to look for food in other regions.

The above-mentioned algorithms and other, not mentioned here algorithms (e.g., the Bat algorithm, the cuckoo search that is one of the latest nature-inspired metaheuristic algorithms) are not presented here due to their less popularity and similarity to the two main approaches.

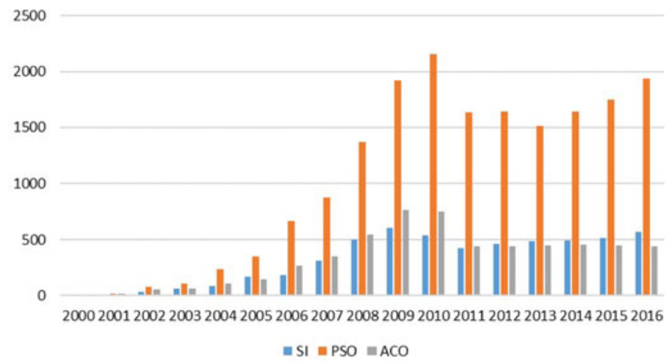

Fig. 2. Number of publications in the IEEE base (years 2000–2016).

D. SI—Problems and Challenges

Artificial colonies of agents able to solve hard optimization problems become more and more popular. Thanks to the self-organization and division of labor, such colonies are capable of achieving intelligent behavior. Swarm algorithms have the advantages of simplicity, flexibility, and ergodicity [81]. Some SI methods are suitable for dynamic applications, i.e., ACO can adapt to changes in the distances. SI can be used when a formal algorithm is unknown or too complicated computationally. SI also reveals some disadvantages. The critical weakness seems to be a lack of theoretical analysis; it is studied mainly experimentally. Although many swarm methods have proven to solve well practical problems, we do not have a theoretical explanation of why these algorithms are so good. The method working very well on a given problem can work poorly for other problem, even similar in some sense. The weak theoretical foundation, difficulty in assessing their stability (repeatability of the results), and convergence diminish the usefulness of these methods. The efficient use of such approaches requires from a user experience and intuition not only in parameter tuning but also in defining a solved problem regarding used method. Ensure the balance between exploration and exploitation is more art than the science. Despite these weaknesses, we observe the growing popularity of SI (see Fig. 2 and Table II). A major reason is the growing demand for smart optimization methods in many business and engineering activities. Nature-inspired methods are suitable mainly for optimization, scheduling, planning, design, and management problems. These kinds of problems are everywhere: in investments, production, distribution, etc. Table II gives some idea of potential uses of SI methods in industry.

As the significant and future use of SI in modern IT technologies, we should mention applications shown on professional websites. On the https://www.whichcar.com.au/car-advice/swarm-intelligence-for-autonomous-cars-explained/. Ash Westerman explains the use of swarm intelligence for autonomous cars, the *car-to-x*—communication between cars and their surroundings. Continental has developed the vehicle-to-vehicle *eHorizon* system for automobiles; next, this system is adapted for two-wheelers. The system uses SI, it allows share information from cloud http://blog.motorcycle.com/2016/12/06/products/continental-bringing-ehorizon-swarm-intelligence-system-motorcycles/.

At https://newatlas.com/uav-swarm-technology/19581/ Darren Quick writes that Boeing use swarm technology for reconnaissance missions using unmanned aerial vehicles. This technology can be used in search-and-rescue missions and identifying enemy threats ahead of ground patrols. John Reed writes at https://www.defensetech.org/2011/08/19/boeing-testing-drone-swarming-techniques/ that Boeing is testing the concept of drone warfare using swarming technology. According to Carl Engelking, the Defense Advanced Research Agency is planning to overwhelm enemies with swarming drones http://blogs.discovermagazine.com/drone360/2015/04/06/darpas-swarming-drones/. Observation of the SI development indicates that these methods are developing and will develop in the future to meet existing and future needs.

III. Swarm Methods in Real-Life Problems

In this section, we present some applications of the above-mentioned methods to solve real-life problems.

A. PSO in Real-Life Problems

The PSO algorithms are applied to many real-world problems. del Valle *et al.* [30] point out three advantages of PSO over other metaheuristics, from which PSO is more commonly used in real-life applications; they are as follows:

1) ease of implementation and fewer parameters for tuning;
2) more efficient memory capabilities due to remembering the own previous best value and the global best value; and
3) efficient in maintaining the diversity of the swarm—each particle can improve itself using the information related to the best particle.

The *electric power* industry is one of the most widely studied areas of the PSO applications. The interesting summary of the PSO technique applied to power systems is given in [30]. The most popular areas in this summary are "reactive power and voltage control," "generation expansion problem," and "system identification and intelligent control." Such significant problems as "short-term load forecasting" and "generator contributions to transmission system" were also considered. PSO is used in electrical engineering in the generation, transmission, state estimation, unit commitment, fault detection and recovery, economical load dispatch, control applications, in optimal use of electrical motor, structuring and restructuring of network, and renewable energy systems (RES) [31]. One can expect that in RES, nature-inspired techniques as PSO will increase their popularity; they can be useful in planning, designing, and control of RES. The renewable energy sources introduce variability into the stand-alone hybrid power systems. It is caused by the fluctuating availability of solar and wind. Properly sizing the battery of the system is important for the reliability and low cost of the system. The paper [32] deals with the system consisting of solar photovoltaic (PV) panels, wind turbines, and traditional diesel generators. Incorporation of the renewable resources into the system can be evaluated by penetration level. It is the installed capacity of the renewable generator over the peak load demand. The authors adjust the PSO algorithm (DM-PSO, the

TABLE II
POPULARITY OF SOME SI METHODS IN THE SELECTED SCIENTIFIC DATABASES—A NUMBER OF TITLES IN GENERAL, AND IN SELECTED AREAS: SI—SWARM INTELLIGENCE, PSO—PARTICLE SWARM OPTIMIZATION, AND ACO—ANT COLONY OPTIMIZATION

Method	Google Scholar	Springer	IEEE Explore	ACM	Scientific	Science Direct	Sage	Taylor	WoS	Total
SI	336 000	24 796	5720	1340	1353	9764	1713	4845	5766	391 297
PSO	211 000	19 399	18 741	360	2861	20 210	1044	3174	34478	311 267
ACO	555 000	32185	5956	653	1546	34 028	1598	5432	10084	646 482
SI—Application	203 000	21305	1506	1300	1156	8495	1433	3821	1355	243 371
PSO—Application	202 000	17 474	4604	310	2294	17 124	1017	3105	6599	254 527
ACO—Application	315 000	27373	1582	582	1255	24 570	1441	4880	2017	378 700
SI—Data mining	49 100	7963	288	811	143	1822	217	748	177	61 269
PSO—Data mining	40 900	5752	511	162	236	2479	111	451	505	51 107
ACO—Data mining	35 500	8405	289	213	123	2717	287	703	284	48 521
SI—Scheduling	47 900	5863	268	506	299	2911	302	913	272	59 234
PSO—Scheduling	56 600	5186	1199	147	623	5729	226	939	2490	73 139
ACO—Scheduling	78 600	8800	717	297	507	7456	511	1534	1341	99 763
SI—Railway	23 200	611	11	20	32	256	141	528	2	24 801
PSO—Railway	14 400	234	73	13	50	318	20	88	63	15 259
ACO—Railway	10 200	599	40	16	41	582	66	220	57	11 821
SI—Energy	98 800	8134	500	551	410	4364	757	2056	479	116 051
PSO—Energy	99 700	6955	2834	162	956	10 539	538	1489	3514	126 687
ACO—Energy	106 000	11 746	672	205	406	15 090	781	2035	788	137 723
SI—Water	63 600	4382	6	102	167	2204	555	1875	126	73 017
PSO—Water	50 300	3125	278	36	348	5380	186	742	877	61 272
ACO—Water	63 400	7933	75	58	186	12 290	580	1733	296	86 551
SI—Transport	40 900	3479	44	423	93	1471	472	1366	65	48 313
PSO—Transport	38 200	2188	135	56	176	3363	188	776	276	45 358
ACO—Transport	45 300	6641	101	110	180	8502	601	1695	171	63 301
SI—Urban	32 600	2059	64	431	75	762	366	890	58	37 305
PSO—Urban	26 700	1024	176	26	112	1321	78	223	196	29 856
ACO—Urban	32 000	2426	120	43	81	2841	384	643	137	38 675
SI—Management	108 000	10124	667	837	365	3622	809	2550	339	127 313
PSO—Management	87 200	7835	2279	260	738	6207	296	1611	1914	108 340
ACO—Management	135 000	13570	1038	384	536	10 209	870	2768	656	165 031

distributed mutated PSO) to obtain the optimal size of battery for systems with different penetration levels of renewables. As the objective of the optimal control, they used dispatch cost and the levelized cost of electricity. The method gave the optimal dispatch for the considered case; for benchmarking data, the results were also good. PV energy becomes more popular source of nonconventional energy every year. Such power depends strongly on atmospheric conditions, especially temperature and irradiation. Solar panels can be made of different materials; however, at the output from boost converter they give constant output. In [33], the PSO is developed to find out duty cycle to the boost converter irrespective of the volume of the power produced. The authors conclude the paper stating that the PSO method gives good results even under partially shaded and varying atmospheric conditions. In [34], a global maximum power point tracking (MPPT) for PV is developed. It is based on leader PSO. Exclusive mutation strategies were proposed to achieve the global best leader. The LPSO with mutation managed to excel also in hardware experimentation; so it is suitable for MPPT applications. The similar subject is considered in [35] where the PSO technique is used to calculate the required duty cycle. The method can track the maximum power accurately; the oscillations in the steady state are almost zero. The use of PSO for designing unified power flow controller based stabilizers to improve power system transient stability is a subject of [36]. The authors search for the optimum stabilizer parameter settings with a given objective function. Also, to illustrate the potential of PSO, they explored the usability of the approach to robust controller design using simultaneous stabilization. The authors conclude their work that the eigenvalue-based objective function can be used to design individual and coordinated stabilizers. The authors also note that use of the time-domain-based goal function has the advantage of designing several controllers with different objectives in a coordinated manner. Ma *et al.* [37] solve the problem of heating system planning by combining the integral mathematical model and the improved particle swarm optimization. The life cycle cost depends mainly on the distance between the source of heating and the installation of the heat-consuming installation, and the number of heating sources. The above was the main premise for modifying the PSO basic version. Performed in the paper, case study allowed to check the method feasibility in practical use. The optimal power flow (OPF) problem and minimization of losses in distribution systems have gained increasing attention because they are hard optimization problems and the cost of electric energy is high. The function optimized in OPF undergoes inequality constraints concerning the upper and the lower limits of the control and some state variables. In the same time, the power flow equations are the equality constraints. The paper [38] presents the above problem and proposes the PSO algorithm for solving the optimal distribution system reconfiguration problem for power loss minimization by installing the shunt compensator. The optimal setting of OPF is searched based on loss minimization function. In the authors' opinion, the result on standard IEEE 14-, IEEE 30-, and IEEE 118-bus test systems confirms the potential and the effectiveness of the developed method. The oil demand in Iran using the PSO approach, based on socioeconomic indicators, is studied in [39]. The authors consider two metaheuristics: GA and PSO as tools for forecasting the demand up to 2030. As indicators, they used such data as the population, gross domestic product, and import and export data. The authors conclude that their results provide helpful insight into energy system modeling and are useful for scholars and policy makers.

Efficient *design* is an another group of problems solved with the use of nature-inspired approach. The social behavior of fish colonies are the base of the FSO algorithm used to four different design problems [50]: welded beam design problem, tension/compression string design problem, pressure vessel design problem, and binary distillation column design problem. The authors refer promising results. Constraint structural optimization tasks are considered by Perez and Behdinan [51]. They used the PSO algorithm; obtained solutions were better than those produced by other optimization methods. The two applications with PSO algorithm are presented in [52]: a structural problem—identification of structural parameters; and a material problem, which focuses on a resolution of contradictory multiobjective functions. The general problem is the indirect inverse analysis, where the goal is to minimize the error between the real data and predicted data obtained through a mechanical model. In both problems, the advantage of PSO was observed. The paper [53] presents the discrete binary PSO combined with finite-different time-domain method used for designing a broadband microstrip antenna. The bandwidth of the optimized antenna was significantly expanded comparing to the original antenna. Huang *et al.* [54] state that the optimization of parameters influences strongly on the "final quality of product in modern industrial manufacturing process." The authors proposed an interesting hybrid method for parameter optimization problems in structure design and machining process: teaching-learning-based cuckoo search (TLCS). TLCS produced better solutions compared to others, taken from the literature. The adapting routing problem is suitable for SI approach, numerous algorithms have been proposed for different types of networks. The paper [55] presents an overview of the research on this area, including wired networks and wireless ad hoc networks. The authors describe different types of proposed algorithms and different networks considered. They conclude that the implementation of the proposed algorithms to real-life problems is still too limited. The route from simulations to real devices is not a trivial task. Second problem refers to the very dynamic nature of modern networks, which requires new and effective algorithms able to learn about the current network, user context, adapt decision policies to it, and others. This is a challenge for researchers developing SI techniques.

An exhaustive review of applications of PSO algorithms in the railway domain can be find in [56]. The number of applications of a variety of PSO approaches in this area increases continuously. The largest areas within railway domain are scheduling, active controls, and network layout planning. The authors identified the optimization of vehicle mechanical systems dynamics as the prospective area and a challenge for the PSO. El-Abd *et al.* [57] considered the placement problem in field programmable gate arrays. They propose a discrete version of PSO (DPSO) to find the optimum logic blocks and IO pins location to minimize the total wire-length. They also developed and studied a cooperative version of the DPSO. The methods are better for the small- and medium-sized problems and similar to the state of the art for bigger problems. The PSO is used to design stable IIR digital filters with nonstandard amplitude characteristics [58]. The obtained filter is stable and fulfill all the prescribed design assumptions.

Data mining, or widely knowledge discovery from data, becomes a very important research and application subject. In this area, the nature-inspired techniques have found an important role. The paper [64] discusses three properties of big data analytics namely, the high dimensionality of data, the dynamical change of data, and the multiobjective of problems. The two real-life problems are mentioned as potential applications of SI, intelligent transportation system, and wireless sensor networks. Sun *et al.* [65] faced the problem of sparse representation, which causes difficulties in analysis of such data. The improved PSO version called the cluster guide PSO is proposed and studied as a method for underdetermined blind source separation. The big data still are an open research area. The data dimension reduction and, in particular, selection of the best possible set of features for a given task is very significant research area in machine learning domain. The paper [66] presents the fitness proportionate selection binary particle swarm optimization applied to the future selection in sentiment classification of documents.

In [67], the PSO algorithm is used for parameters optimization of wavelet kernel local Fisher discriminant analysis (WKLFDA)—the useful tool for feature extraction and dimensionality reduction. The developed method is named PSO-WKLFDA. Due to problems with multiclass classifications, the author proposed the individual PSO-WKLFDA method for extracting proper features of each binary class. The sentiment analysis is one of the very significant tasks within text mining; it is very useful, e.g., in social networks analysis. The thesis [68] considers an application of PSO to pattern recognition problems. As the clustering, the image segmentation problems are investigated. Also, the PSO-based approaches to tackle the color image quantization and spectral unmixing problems are proposed and studied. SI methods are used for images segmentation also in [69] and [70]. The paper [71] presents an interesting and novel local outlier detection approach (LOMA), the method mining local outliers in high-dimensional data. LOMA reduces the size of data analyzing irrelevance attributes and objects in the data. The essence of LOMA is searching sparse subspace, which implements the PSO in reduced datasets. LOMA achieves good mining efficiency and accuracy. Forsati *et al.* proposed the modified version of bee colony optimization (BCO) [72]. They introduced cloning and fairness concepts into the BCO making it more efficient for data clustering.

B. ACO in Real-Life Problems

ACO is an intensively studied metaheuristic; it has found applications in many areas. Save *energy* and deal with cloud computing is a potential problem for nature-inspired applications. Duan *et al.* [80] proposed a hybrid method consisting of a prediction model based on fractal mathematics and a scheduler—an improved ACO. The fractal-based prediction model indicates the load trend; the improved ACO is responsible for minimizing energy consumption considering the quality of service. The method exhibits excellent energy efficiency and resource utilization.

Future selection for *machine learning* tasks, especially for remote sensing images with high-dimensional features, is a serious problem. Wang *et al.* [82] decided to tune the parameters of support vector machine (SVM) classifier and to select the features of classified objects together. Both problems are affected by each other. They proposed the hybrid method consisting of a modified coded ACO and GA to optimize the parameters of SVM and feature selection simultaneously. The method produced better results when compared to other metaheuristics. The similar studies are presented in [83], the GA is used as initial pheromone information. The paper [84] presents some review of ACO in future selection tasks. Multimedia big data are the subject of [85]. The authors proposed an architecture that contains three layers: service layer, platform layer, and infrastructure layer. The system allows building large-scale multimedia big data analytic applications using Hadoop platform. The ACO algorithm is applied for resource allocation in infrastructure layer. Presented simulation results proofed that the ACO can allocate virtual machine optimally minimizing the response time.

Interesting research in data mining area is presented in [92]. The main goal was to discover rule for asphyxia prediction during delivery. For this purpose, the information retrieval had to be done for complementing cardiotocography signals with other data collected in the form of data basis. Different classifiers were tested. The ant-inspired ACO-DTree algorithms significantly outperformed all others. In [93], the ant search is used to find the optimum design that satisfied the required mechanism performance. Four cases were tested, the mechanism was optimized with different criteria: maximization of the amplification ratio, minimization of stresses, and maximization of the output displacement. The proposed use of ant system differs from the previous approaches. The results present the potential of proposed method, especially for multidisciplinary optimization problems not limited to designs with physical path. The paper [94] deals with soil cation exchange capacity (CEC), which is a significant property represented soil fertility status. CEC is difficult to measure, researchers try to predict it. The paper presents a hybrid method based on ACO and network-based fuzzy inference system. The authors' goals were to identify the most determinant parameters of agricultural soils CEC in a given region.

The paper [86] discusses different method applied in *water* resources *planning*, *engineering*, and *management*. The author writes about the nature-inspired methods as modern optimization methods and underline their usefulness in practice. He reports some applications of ACO in water engineering area. In 2001, ACO was used for estimating unsaturated soil hydraulic parameters. Other application is to find optimal design of water distribution systems. An improved ACO was used to optimize a single water-reservoir operation. Other tasks solved by ACO: optimal control of pumps in water distribution networks; and designing optimal irrigation system. In [87], we can find following applications of ACO: reservoir operation and surface water management; water distribution system; urban drainage and sewer system; groundwater managements; and environmental and watershed management. The paper [88] presents simultaneous optimization of initial design and rehabilitation scheduling of water distribution networks. They proposed multiobjective ACO engine combined with a pressure-dependent analysis model and a pipe break prediction model. Two networks were used for testing the proposed method. The results showed that the dynamic design, proposed by the authors, produces more reliable and lower cost networks.

In nuclear power plant, the physical protection system (PPS) is essential to safeguard the nuclear materials. The paper [89] presents the developed heuristic approach for the evaluation of physical protection system effectiveness (HAPPS) method for PPS analyzing and evaluating in which the ACO is one of the three components. HAPPS works as search algorithm; it seeks the vulnerable adversary intrusion and escape path under certain conditions. In the paper [90], the ACO algorithm was used for optimization of process planning to reduce the total cost for machining process. The process is converted to a constraint-based traveling salesmen problem. Next, the mathematical model is constructed. The ACO algorithm is used to solve the proposed mathematical model. The dynamic travelling salesmen problem (DTSP) is more difficult because the weights between nodes may change. Mavrovouniotis *et al.* [79] proposed a memetic ACO integrated with a local search operator, as a tool for symmetric and asymmetric DTSPs. The generalized traveling salesman problem (GTSP) is the subject of [91], where the authors proposed an improved ACO, to solve the GTSP. They design a novel optimized implementation of ACO to reduce the processing costs.

C. Examples of Other Nature-Inspired Methods in Applications

It is worth mentioning that other methods belonging to the group of SI as well as *hybrid methods* are widely used in the energy industry area. Hybrid approaches lie on to join two or more methods to eliminate weaknesses and to strengthen the advantages of combined methods. We mention here few such studies. Rao *et al.* use heuristic optimization big-bang and big-crunch and FireFly optimization to obtain the solution of the OPF with continuous and discrete control variables for quadratic generator output cost functions [40]. The constraints (load bus voltage magnitudes, line flows, reactive power capabilities) are included as quadratic penalties. The FireFly worked better. Pattern search (PS) joined with FA (hFA-PS) method is proposed in [41]. Automatic generation control of multiarea power systems with the consideration of generation rate constraint is studied in this paper. The PS is employed for tuning the best solution provided by FA. The method can handle nonlinearity and physical constraints in the system model. The similar hybrid approach, proposed in [42], is used for a static synchronous series compensator based power oscillation damping controller design. The method outperforms some recently proposed approaches.

Enhanced cuckoo search algorithm, used for contingency constrained economic load dispatch to relieve transmission line overloading, is presented in [43]. The security enhancement of the power systems concerns the necessary control action against overloads under contingency scenario. The proposed method obtains the solution that outperforms the other state-of-the-art metaheuristics. The cuckoo search algorithm in combination with truncation and rounding theory is applied to solve power network planning [44]. The efficiency and feasibility of the developed, improved CS were verified in experiments with 77-bus system. Paper [45] presents the FA applied to optimization load frequency control in the interconnected reheat thermal power system. The PID controller optimized by FA works better than the conventional one. In [46], the FA is used for optimizing the total operating cost. The use of FA for similar problems is considered in [47] and [48]. The Bat algorithm was tested as a tool for the OPF problem [49]; the fuel cost is minimized.

Analysis of complex service systems in manufacturing, transportation networks, computer systems, and communications needs methods supplied by the theory of queuing. The queuing system can be optimized by the nature inspired methods. Paper [59] presents the use of FA. A number of papers are dedicated to using swarm algorithms for scheduling problems. In [60], the multiobjective problem of workflow scheduling in clouds is solved. The Bat algorithm is used to minimize the execution time and maximize the reliability by keeping the budget within the specified limits. The similar problem, minimizing the overall cost of the workflow in the cloud is considered in [61]; the authors introduced Binary bat Algorithm. The ABC is used in [62] to minimize the makespan of the job scheduling process and a number of machines. Another metaheuristic, FireFly is used to develop a minimal makespan in [63].

IV. Summary and Future Trends

The concept of SI is described in this paper, and concise descriptions of two representatives of SI are provided, namely PSO and ACO. Pseudocodes of the discussed methods are included in this paper as well as some of their intuitions and properties. Nature-inspired methods cause a problem with parameters tuning—to assure a balance between exploration and exploitation abilities. The proper defining of the problem to be solved, in terms of used metaheuristic, can be problematic for researchers inexperienced in the use of these methods. As it is shown in the paper, the SI methods are very popular and very useful in practice. Table II presents the answers given by nine scientific databases for following questions: "SI"; 'PSO"; and "ACO." Next we concatenated these questions with the words "application," "data mining," "scheduling," "railway," "energy," "water," "transport," "urban," and "management," respectively. The Google Scholar returns an approximate result; therefore, at least two last digits in the second columns of Table II are zeros. It is worth mentioning that still the new nature-inspired approaches are developed; an example of such new technique is [95]. Lastly, there is more and more research on the development of hybrid methods. Fig. 2 presents the number of publications returned by the IEEE Explore base in the years from 2000 to 2016 (concerning SI, PSO, and ACO). While the performance of SI algorithms and understanding of their work significantly increased, in some areas much more research is to be done.

After the literature analysis, we can say that the ACO and PSO development goes in three main directions. First, is the efficient adaptation to the specificity of the problems that are being solved. These areas are dynamic and stochastic problems, multiple objective problems, etc. [3], [79], [88]. The second is the parallelization of the algorithms for the computing acceleration [129]–[133]. The increasing availability of high-performance computing platforms such as graphics processing units (GPUs) has led to growing interest in their potential as a platform for parallel ACO or PSO algorithms. GPUs typically order high-computational throughput at relatively low financial cost and with low energy consumption [130]. However, applications of ACO or PSO algorithms in GPU require a high degree of parallelism to exploit the full performance of the hardware [130]. Therefore, the efficient parallelization of ACO and PSO algorithms are the challenging cases. The third direction is the hybridization with other techniques such as the local search methods (local search algorithms can better locally explore a neighborhood in the search space). It has been shown that the integration of local search operators can significantly improve the performance of ACO and PSO algorithms [79], [135]. The fuzzy controller allows regulating values of heuristic coefficients of ACO and PSO dynamically. The results received show the high effectiveness of fuzzy logic controllers used in ACO and PSO [136], [137]. The nature-inspired optimization methods can be joined with others, such as GA (the hybridized algorithms can improve the solutions of each other; thus more diversity and better quality solutions can be achieved in the population) [134]. The interesting example of such hybridization is in [96], where Yeh proposed hybrid swarm optimization, which combines concepts borrowed from simplified SI, PSO, simulated annealing, and network reliability methods. The considered problem to solve is generalized redundancy allocation problem.

The mathematical analysis of the SI methods remains unsolved; such analysis is difficult, and the interaction of components of the methods is complex, nonlinear, and stochastic. Snooping and imitation of nature, which is the unsurpassed model of an effective, adaptive system, is and will be a long-term challenge for researchers.

Acknowledgment

The authors would like to thank the anonymous reviewers for their valuable comments, which greatly helped to improve the content of this paper.

References

[1] I. Fister, Jr., X.-S. Yang, I. Fister, J. Brest, and D. Fister, "A brief review of nature-inspired algorithms for optimization," *Elektrotehniski Vestnik*, vol. 80, no. 3, pp. 1–7, 2013.

[2] T. W. Malone, "Conversations at edge collective intelligence: A conversation with Thomas W. Malone," 2012. [Online]. Available: https://www.edge.org/conversation/thomas_w__malone-collective-intelligence. Accessed on: May 24, 2017.

[3] M. Mavrovouniotis, C. Li, and S. Yang, "A survey of swarm intelligence for dynamic optimization—Algorithms and applications," *Swarm Evol. Comput.*, vol. 33, pp. 1–17, 2017.

[4] G. Beni, “The concept of cellular robotic system” in *Proc. IEEE Int. Symp. Intell. Control*, 1988, pp. 57–62.

[5] J. Kennedy and R. C. Eberhart, “Particle swarm optimization” in *Proc. IEEE Int. Conf. Neural Netw.*, 1995, pp. 1942–1948.

[6] R. C. Eberhart and J. Kennedy, “A new optimizer using particle swarm theory,” in *Proc. 6th Int. Symp. Micromach. Hum. Sci.*, 1995, pp. 39–43.

[7] A. Slowik, “Particle swarm optimization” in *The Industrial Electronics Handbook—Intelligent Systems*, 2nd ed., B. M. Wilamowski and J. D. Irwin, Eds. Boca Raton, FL, USA: CRC Press, 2011, pp. 281–289.

[8] J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence*. San Mateo, CA, USA: Morgan Kaufmann, 2001.

[9] A. Colorni, M. Dorigo, and V. Maniezzo, “Distributed optimization by ant colonies,” in *Proc. Eur. Conf. Artif. Life*, 1991, pp. 134–142.

[10] C. Blum, “Ant colony optimization: Introduction and recent trends,” *Phys. Life Rev.*, vol. 2, pp. 353–373, 2005.

[11] D. T. Pham, A. Ghanbarzadeh, E. Ko, S. Otri, S. Rahim, and M. Zaidi, “The bees algorithm—A novel tool for complex optimization problems,” in *Proc. Intell. Prod. Mach. Syst.*, 2006, pp. 454–459.

[12] D. Karaboga and B. Basturk, “A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm,” *J. Global Optim.*, vol. 39, no. 3, pp. 459–471, 2007.

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

[14] X.-S. Yang, “Firefly algorithm,” in *Nature-Inspired Metaheuristic Algorithms*, X.-S. Yang, Ed. Hoboken, NJ, USA: Wiley, 2008, p. 7990.

[15] X.-S. Yang, “Firefly algorithms for multimodal optimization,” in *Proc. Int. Symp. Stoch. Algorithms*, 2009, vol. 5792, pp. 169–178.

[16] L. Li, Z. Shao, and J. Qian, “An optimizing method based on autonomous animals: Fishswarm algorithm,” *Syst. Eng. Theory Pract.*, vol. 22, no. 11, pp. 32–38, 2002.

[17] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA, USA: Addison-Wesley, 1989.

[18] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*. New York, NY, USA: Springer, 1998.

[19] M. Bedau, “Artificial life: Organization, adaptation and complexity from the bottom up,” *Trends Cogn. Sci.*, vol. 7, no. 11, pp. 505–512, 2003.

[20] F. van den Bergh, “An analysis of particle swarm optimizers,” Ph.D. dissertation, Univ. Pretoria, Pretoria, South Africa, 2001.

[21] W. Sun, A. Lin, H. Yu, Q. Liang, and G. Wu, “All-dimension neighborhood based particle swarm optimization with randomly selected neighbors,” *Inf. Sci.*, vol. 405, pp. 141–156, 2017.

[22] M. Imran, R. Hashima, and N. E. A. Khalidb, “An overview of particle swarm optimization variants,” *Procedia Eng.*, vol. 53, pp. 491–496, 2013.

[23] S. D. Chavan and N. P. Adgokar, “An overview on particle swarm optimization: Basic concepts and modified variants,” *Int. J. Sci. Res*, vol. 4, no. 5, pp. 255–260, 2015.

[24] E. Zahara and Y.-T. Kao, “Hybrid Nelder-Mead simplex search and particle swarm optimization for constrained engineering design problems,” *Expert Syst. Appl.*, vol. 36, no. 2, pt. 2, pp. 3880–3886, 2009.

[25] C.-L. Sun, J.-C. Zeng, and J.-S. Pan, “An improved vector particle swarm optimization for constrained optimization problems,” *Inf. Sci.*, vol. 181, no. 6, pp. 1153–1163, 2011.

[26] H. Shayeghi, M. Mahdavi, and A. Bagheri, “An improved DPSO with mutation based on similarity algorithm for optimization of transmission lines loading,” *Energy Convers. Manage.*, vol. 51, no. 12, pp. 2715–2723, 2010.

[27] B. Jarboui, M. Cheikh, P. Siarry, and A. Rebai, “Combinatorial particle swarm optimization (CPSO) for partitional clustering problem,” *Appl. Math. Comput.*, vol. 192, no. 2, pp. 337–345, 2007.

[28] J. Yang, J. Zhou, L. Liu, and Y. Li, “A novel strategy of Pareto-optimal solution searching in multi-objective particle swarm optimization (MOPSO),” *Comput. Math. Appl.*, vol. 57, no. 11–12, pp. 1995–2000, 2009.

[29] M. Rabbani, M. A. Bajestani, and G. B. Khoshkhou, “A multi-objective particle swarm optimization for project selection problem,” *Expert Syst. Appl.*, vol. 37, no. 1, pp. 315–321, 2010.

[30] Y. del Valle, G. K. Venayagamoorthy, S. Mohagheghi, J.-C. Hernandez, and R. G. Harley, “Particle swarm optimization: Basic concepts, variants and applications in power systems,” *IEEE Trans. Evol. Comput.*, vol. 12, no. 2, pp. 171–195, Apr. 2008.

[31] K. Anula and R. Saroj, “A review of particle swarm optimization and its applications in solar photovoltaic system,” *Appl. Soft Comput.*, vol. 13, no. 5, pp. 2997–3006, 2013.

[32] C. Shang, D. Srinivasan, and T. Reindl, “An improved particle swarm optimisation algorithm applied to battery sizing for stand-alone hybrid power systems,” *Int. J. Elect. Power Energy Syst.*, vol. 74, pp. 104–117, 2016.

[33] M. Balamurugan, S. Narendiran, S. K. Sahoo, R. Das, and A. K. Sahoo, “Application of particle swarm optimization for maximum power point tracking in PV system” in *Proc. 3rd Int. Conf. Elect. Energy Syst.*, 2016, pp. 35–38.

[34] J. P. Ram and N. Rajasekar, “A new robust, mutated and fast tracking LPSO method for solar PV maximum power point tracking under partial shaded conditions,” *Appl. Energy*, vol. 201, pp. 45–59, 2017.

[35] R. Venugopalan, N. Krishnakumar, R. Sarjila, and N. Rajasekar, “Application of particle swarm optimization technique for the design of maximum power point tracking,” *Adv. Mater. Res.*, vol. 768, pp. 47–56, 2013.

[36] A. T. Al-Awami, M. A. Abido, and Y. L. Abdel-Magid, “Application of PSO to design UPFC-based stabilizers,” in *Swarm Intelligence: Focus on Ant and Particle Swarm Optimization*, F. T. S. Chan and M. K. Tiwari, Eds. Vienna, Austria: InTech Educ. Publ., 2007, pp. 235–262.

[37] R.-J. Ma, N.-Y. Yu, and J.-Y. Hu, “Application of particle swarm optimization algorithm in the heating system planning problem,” *Sci. World J.*, vol. 2013, 2013, Art. no. 718345. [Online]. Available: http://dx.doi.org/10.1155/2013/718345

[38] A. A. A. Esmin and G. Lambert-Torres, “Application of particle swarm optimization to optimal power systems,” *Int. J. Innov. Comput., Inf. Control*, vol. 8, no. 3(A), pp. 1705–1716, 2012.

[39] E. Assareh, M. A. Behrang, M. R. Assari, and A. Ghanbarzadeh, “Application of PSO (particle swarm optimization) and GA (genetic algorithm) techniques on demand estimation of oil in Iran,” *Energy*, vol. 35, pp. 5223–5229, 2010.

[40] C. V. Gopala, K. Rao, and G. Yesuratnam, “Big-bang and big-crunch (BB-BC) and firefly optimization (FFO): Application and comparison to optimal power flow with continuous and discrete control variables,” *Int. J. Elect. Eng. Informat.*, vol. 4, no. 4, pp. 575–583, 2012.

[41] R. K. Sahu, S. Panda, and S. Padhan, “A hybrid firefly algorithm and pattern search technique for automatic generation control of multi area power systems,” *Elect. Power Energy Syst.*, vol. 64, pp. 9–23, 2015.

[42] S. Mahapatra, S. Panda, and S. C. Swain, “A hybrid firefly algorithm and pattern search technique for SSSC based power oscillation damping controller design,” *Ain Shams Eng. J.*, vol. 5, pp. 1177–1188, 2014.

[43] P. Sekhar and S. Mohanty, “An enhanced cuckoo search algorithm based contingency constrained economic load dispatch for security enhancement,” *Elect. Power Energy Syst.*, vol. 75, pp. 303–310, 2016.

[44] S. Tian *et al.*, “Application of cuckoo search algorithm in power network planning,” in *Proc. 5th Int. Conf. Elect. Utility Deregulation Restruct. Power Technol*, 2015, pp. 604–608.

[45] K. Naidu, H. Mokhlis, and A. H. A. Bakar, “Application of firefly algorithm (FA) based optimization in load frequency control for interconnected reheat thermal power system,” in *Proc. IEEE Jordan Conf. Appl. Elect. Eng. Comput. Technol.*, 2013, pp. 1–5.

[46] D. P. Reddy and J. N. C. Sekhar, “Application of firefly algorithm for combined economic load and emission dispatch,” *Int. J. Recent Innov. Trends Comput. Commun.*, vol. 2, no. 8, pp. 2448–2452, 2014.

[47] T. Apostolopoulos and A. Vlachos, “Application of the firefly algorithm for solving the economic emissions load dispatch problem,” *Int. J. Combinatorics*, vol. 2011, 2011, Art. no. 523806. [Online]. Available: http://dx.doi.org/10.1155/2011/523806

[48] D. K. Mohanty, “Application of firefly algorithm for design optimization of a shell and tube heat exchanger from economic point of view,” *Int. J. Therm. Sci.*, vol. 102, pp. 228–238, 2016.

[49] H. D. Abatari, M. S. S. Abad, and H. Seifi, “Application of bat optimization algorithm in optimal power flow,” in *Proc. 24th Iranian Conf. Elect. Eng*, 2016, pp. 793–798.

[50] F. S. Lobato and V. Steffen, Jr., “Fish swarm optimization algorithm applied to engineering system design,” *Latin Amer. J. Solid Struct.*, vol. 11, pp. 143–156, 2014.

[51] R. E. Perez and K. Behdinan, “Particle swarm approach for structural design optimization,” *Comput. Struct.*, vol. 85, pp. 1579–1588, 2007.

[52] M. Fontan, A. Ndiaye, D. Breysse, and P. Castra, “Inverse analysis in civil engineering: Applications to identification of parameters and design of structural material using mono or multi-objective particle swarm optimization,” in *Theory and New Applications of Swarm Intelligence*, R. Parpinelli, Ed. Vienna, Austria: InTech, 2012, pp. 87–114.

[53] Y.-M. Zhao and J.-D. Xu, “Application of particle swarm optimization for the design of a broadband microstrip antenna” in *Proc. Int. Conf. Comput. Appl. Syst. Model*, 2010, pp. 406–408.
[54] J. Huang, L. Gao, and X. Li, “An effective teaching-learning-based cuckoo search algorithm for parameter optimization problems in structure designing and machining processes,” *Appl. Soft Comput.*, vol. 36, pp. 349–356, 2015.
[55] F. Ducatelle, G. A. Di Caro, and L. M. Gambardella, “Principles and applications of swarm intelligence for adaptive routing in telecommunications networks,” *Swarm Intell.*, vol. 4, no. 3, pp. 173–198, 2010.
[56] Q. Wu, C. Cole, and T. McSweeney, “Applications of particle swarm optimization in the railway domain,” *Int. J. Rail Transp.*, vol. 4, no. 3, pp. 167–190, 2016.
[57] M. El-Abd, H. Hassan, M. Anis, M. S. Kamel, and M. Elmasry, “Discrete cooperative particle swarm optimization for FPGA placement,” *Appl. Soft Comput.*, vol. 10, no. 1, pp. 284–295, 2010.
[58] A. Slowik and M. Bialko, “Design and optimization of IIR digital filters with non-standard characteristics using particle swarm optimization,” in *Proc. 14th IEEE Int. Conf. Electron., Circuits Syst.*, 2007, pp. 162–165.
[59] J. Kwiecien and B. Filipowicz, “Firefly algorithm in optimization of queueing systems,” *Bull. Polish Acad. Sci., Tech. Sci.*, vol. 60, no. 2, pp. 363–368, 2012.
[60] N. Kaur and S. Singh, “A budget-constrained time and reliability optimization bat algorithm for scheduling workflow applications in clouds,” *Procedia Comput. Sci.*, vol. 98, pp. 199–204, 2016.
[61] S. Raghavan, P. Sarwesh, C. Marimuthu, and K. Chandrasekaran, “Bat algorithm for scheduling workflow applications in cloud,” in *Proc. Int. Conf. Electron. Des., Comput. Netw. Autom. Verification*, 2015, pp. 139–144.
[62] A. Muthiah and R. Rajkumar, “A comparison of artificial bee colony algorithm and genetic algorithm to minimize the makespan for job shop scheduling,” *Procedia Eng.*, vol. 97, pp. 1745–1754, 2014.
[63] K. C. Udaiyakumar and M. Chandrasekaran, “Application of firefly algorithm in job shop scheduling problem for minimization of makespan,” *Procedia Eng.*, vol. 97, pp. 1798–1807, 2014.
[64] S. Cheng, Y. Shi, Q. Qin, and R. Bai, “Swarm intelligence in big data analytics,” in *Proc. Intell. Data Eng. Autom. Learn.*, 2013, vol. 8206, pp. 417–426.
[65] T.-Y. Sun, C.-C. Liu, S.-J. Tsai, S.-T. Hsieh, and K.-Y. Li, “Cluster guide particle swarm optimization (CGPSO) for underdetermined blind source separation with advanced conditions,” *IEEE Trans. Evol. Comput.*, vol. 15, no. 6, pp. 798–811, Dec. 2011.
[66] L. Shang, Z. Zhou, and X. Liu, “Particle swarm optimization-based feature selection in sentiment classification,” *Soft Comput.*, vol. 20, no. 10, pp. 3821–3824, 2016.
[67] M. Van and H.-J. Kang, “Bearing defect classification based on individual wavelet local fisher discriminant analysis with particle swarm optimization,” *IEEE Trans. Ind. Informat.*, vol. 12, no. 1, pp. 124–135, Feb. 2016.
[68] M. G. H. Omran, “Particle swarm optimization methods for pattern recognition and image processing,” Ph.D. dissertation, Univ. Pretoria, Pretoria, South Africa, 2004.
[69] H.-J. Sun, “Image segmentation algorithm based on swarm intelligence technology,” in *Proc. Int. Conf. Intell. Comput. Internet Things*, 2015, pp. 68–71.
[70] S. Saatchi and C.-C. Hung, “Swarm intelligence and image segmentation” in *Swarm Intelligence, Focus on Ant and Particle Swarm Optimization*, F. T. S. Chan and M. K. Tiwari, Eds. Vienna, Austria: InTech, 2007, pp. 164–178.
[71] X. Zhao, J. Zhang, and X. Qin, “LOMA: A local outlier mining algorithm based on attribute relevance analysis,” *Expert Syst. Appl.*, vol. 84, pp. 272–280, 2017.
[72] R. Forsati, A. Keikha, and M. Shamsfard, “An improved bee colony optimization algorithm with an application to document clustering,” *Neurocomputing*, vol. 159, pp. 9–26, 2015.
[73] M. Dorigo, V. Maniezzo, and A. Colorni, “Ant system: Optimization by a colony of cooperating agents,” *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 26, no. 1, pp. 29–41, Feb. 1996.
[74] K. Socha and M. Dorigo, “Ant colony optimization for continous domains,” *Eur. J. Oper. Res.*, vol. 185, no. 3, pp. 1155–1173, 2008.
[75] K. Ye, C. Zhang, J. Ning, and X. Liu, “Ant-colony algorithm with a strengthened negative-feedback mechanism for constraint-satisfaction problems,” *Inf. Sci.*, vol. 406–407, pp. 29–41, 2017.
[76] Z. Chen, S. Zhou, and J. Luo, “A robust ant colony optimization for continuous functions,” *Expert Syst. Appl.*, vol. 81, pp. 309–320, 2017.
[77] P. S. Shelokar, P. Siarry, V. K. Jayaraman, and B. D. Kulkarni, “Particle swarm and ant colony algorithms hybridized for improved continuous optimization,” *Appl. Math. Comput.*, vol. 188, no. 1, pp. 129–142, 2007.
[78] M. Zangari, A. Pozo, R. Santana, and A. Mendiburu, “A decomposition-based binary ACO algorithm for the multiobjective UBQP,” *Neurocomputing*, vol. 246, pp. 58–68, 2017.
[79] M. Mavrovouniotis, F. M. Muller, and S. Yang, “Ant colony optimization with local search for dynamic traveling salesman problems,” *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1743–1756, Jul. 2017.
[80] H. Duan, C. Chen, G. Min, and Y. Wu, “Energy-aware scheduling of virtual machines in heterogeneous cloud computing systems,” *Future Gener. Comput. Syst.*, vol. 74, pp. 142–150, 2017.
[81] X.-S. Yang, S. Koziel, and L. Leifsso, “Computational optimization, modelling and simulation: Recent trends and challenges,” *Procedia Comput. Sci.*, vol. 18, pp. 855–860, 2013.
[82] M. Wang, Y. Wan, Z. Ye, and X. Lai, “Remote sensing image classification based on the optimal support vector machine and modified binary coded ant colony optimization algorithm,” *Inf. Sci.*, vol. 402, pp. 50–68, 2017.
[83] Y. Wan, M. Wang, Z. Ye, and X. Lai, “A feature selection method based on modified binary coded ant colony optimization algorithm,” *Appl. Soft Comput.*, vol. 49, pp. 248–258, 2016.
[84] M. Kabir, M. Shahjahan, and K. Murase, “Ant colony optimization toward feature selection” in *Ant Colony Optimization—Techniques and Applications*, H. J. C. Barbosa, Ed. Vienna, Austria: InTech, 2013, pp. 3–44.
[85] K. P. N. Jayasena, L. Li, and Q. Xie, “Multi-modal multimedia big data analyzing architecture and resource allocation on cloud platform,” *Neurocomputing*, vol. 253, pp. 135–143, 2017.
[86] G. Tayfur, “Modern optimization methods in water resources planning, engineering and management,” *Water Resour. Manage.*, vol. 31, no. 10, pp. 3205–3233, 2017.
[87] A. Afshar, F. Massoumi, A. Afshar, and M. A. Marino, “State of the art review of ant colony optimization applications in water resource management,” *Water Resour. Manage.*, vol. 29, no. 11, pp. 3891–3904, 2015.
[88] A. Shirzad, M. Tabesh, and B. Atayikia, “Multiobjective optimization of pressure dependent dynamic design for water distribution networks,” *Water Resour. Manage.*, vol. 31, no. 9, pp. 2561–2578, 2017.
[89] B. Zou, M. Yang, J. Guo, E.-R. Benjamin, and W. Wu, “A heuristic approach for the evaluation of physical protection system effectiveness,” *Ann. Nucl. Energy*, vol. 105, pp. 302–310, 2017.
[90] X.-J. Liu, H. Yi, and Z.-H. Ni, “Application of ant colony optimization algorithm in process planning optimization,” *J. Intell. Manuf.*, vol. 24, no. 1, pp. 1–13, 2013.
[91] K. Jun-man and Z. Yi, “Application of an improved ant colony optimization on generalized traveling salesman problem,” *Energy Procedia*, vol. 17, pp. 319–325, 2012.
[92] M. Bursa and L. Lhotska, “Ant-inspired algorithms in health information system data mining, classification and visualization,” in *Proc. XIVth Mediterranean Conf. Med. Biol. Eng. Comput.*, 2016, pp. 868–873.
[93] N. Diab and A. Smaili, “An ants-search based method for optimum synthesis of compliant mechanisms under various design criteria,” *Mech. Mach. Theory*, vol. 114, pp. 85–97, 2017.
[94] H. Shekofteh, F. Ramazani, and H. Shiran, “Optimal feature selection for predicting soil CEC: Comparing the hybrid of ant colony organization algorithm and adaptive network-based fuzzy system with multiple linear regression,” *Geoderma*, vol. 298, pp. 27–34, 2017.
[95] M. Shabani, S. A. Mirroshandel, and H. Asheri, “Selective refining harmony search: A new optimization algorithm,” *Expert Syst. Appl.*, vol. 81, pp. 423–443, 2017.
[96] W.-C. Yeh, “A new exact solution algorithm for a novel generalized redundancy allocation problem,” *Inf. Sci.*, vol. 408, pp. 182–197, 2017.
[97] M. Dorigo, “Optimization, learning and natural algorithms,” Ph.D. dissertation, , Politecnico di Milano, Milan, Italy, 1992.
[98] X.-S. Yang, “A new metaheuristic bat-inspired algorithm” in *Proc. Nature Inspired Cooperative Strategies Optim.* , 2010, pp. 65–74.
[99] D. Teodorovic and M. DellOrco, “Bee colony optimization—A cooperative learning approach to complex transportation problems,” in *Proc. 16th Mini-EURO Conf. 10th Meeting EWGT*, 2005, pp. 51–60.
[100] P. Lucic and D. Teodorovic, “Bee system: Modeling combinatorial optimization transportation engineering problems by swarm intelligence,” in *Proc. TRISTAN IVth Triennial Symp. Transp. Anal.*, 2001, pp. 441–445.
[101] H. F. Wedde, M. Farooq, and Y. Zhang, “Beehive: An efficient fault-tolerant routing algorithm inspired by honey bee behavior,” in *Proc. Int. Workshop Ant Colony Optim. Swarm Intell.*, 2004, vol. 3172, pp. 83–94.

[102] H. Drias, S. Sadeg, and S. Yahi, "Cooperative bees swarm for solving the maximum weighted satisfiability problem," in *Proc. 8th Int. Conf. Artif. Neural Netw., Comput. Intell. Bioinspired Syst.*, 2005, pp. 318–325.
[103] F. Comellas and J. M. Navarro, "Bumblebees: A multiagent combinatorial optimization algorithm inspired by social insect behaviour," in *Proc. 1st ACM/SIGEVO Summit Genetic Evol. Comput.*, 2009, pp. 811–814.
[104] S.-A. Chu, P.-W. Tsai, and J.-S. Pan, "Cat swarm optimization," in *Proc. Pac. Rim Int. Conf. Artif. Intell.*, 2006, vol. 4099, pp. 854–858.
[105] G. Dhiman and V. Kumar, "Spotted hyena optimizer: A novel bio-inspired based metaheuristic technique for engineering applications," *Adv. Eng. Softw.*, vol. 114, pp. 48–70, 2017.
[106] X.-S. Yang and S. Deb, "Cuckoo search via levy flights," in *Proc. IEEE World Congr. Nature Biol. Inspired Comput.*, 2009, pp. 210–214.
[107] Y. Chu, H. Mi, H. Liao, Z. Ji, and Q. H. Wu, "A fast bacterial swarming algorithm for high-dimensional function optimization," in *Proc. IEEE World Congr. Comput. Intell.*, 2008, pp. 3135–3140.
[108] X.-L. Li, Z.-J. Shao, and J.-X. Qian, "Optimizing method based on autonomous animats: Fish-swarm algorithm," *Syst. Eng. Theory Pract.*, vol. 22, no. 11, pp. 32–38, 2002.
[109] K. N. Krishnanand and D. Ghose, "Glowworm swarm optimisation: A new method for optimising multi-modal functions," *Int. J. Comput. Intell. Stud.*, vol. 1, no. 1, pp. 93–119, 2009.
[110] A. H. Gandomi and A. H. Alavi, "Krill herd: A new bio-inspired optimization algorithm," *Commun. Nonlinear Sci. Numer. Simul.*, vol. 17, no. 12, pp. 4831–4845, 2012.
[111] X.-S. Yang, "Engineering optimizations via nature-inspired virtual bee algorithms," in *Proc. Int. Work-Conf. Interplay Between Natural Artif. Comput.*, 2005, vol. 3562, pp. 317–323.
[112] R. Tang, S. Fong, X.-S. Yang, and S. Deb, "Wolf search algorithm with ephemeral memory," in *Proc. 7th Int. Conf. Digit. Inf. Manage.*, 2012, pp. 165–172.
[113] T.-Q. Wu, M. Yao, and J.-H. Yang, "Dolphin swarm algorithm," *Frontiers Inf. Technol. Electron. Eng.*, vol. 17, no. 8, pp. 717–729, 2016.
[114] Z. H. Chen and H. Y. Tang, "Cockroach swarm optimization," in *Proc. 2nd Int. Conf. Comput. Eng. Technol.*, 2010, pp. 652–655.
[115] G. G. Wang, S. Deb, and L. D. S. Coelho, "Elephant herding optimization," in *Proc. 3rd Int. Symp. Comput. Bus. Intell.*, 2015, pp. 1–5.
[116] E. Cuevas, M. Cienfuegos, D. Zaldivar, and M. Perez-Cisneros, "A swarm optimization algorithm inspired in the behaviour of the social-spider," *Expert Syst. Appl.*, vol. 40, no. 16, pp. 6374–6384, 2013.
[117] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, 2016.
[118] B. Xing and W.-J. Gao, "Fruit fly optimization algorithm," in *Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms*. New York, NY, USA: Springer, 2013, pp. 167–170.
[119] R. Oftadeh, M. J. Mahjoob, and M. Shariatpanahi, "A novel meta-heuristic optimization algorithm inspired by group hunting of animals: Hunting search," *Comput. Math. Appl.*, vol. 60, no. 7, pp. 2087–2098, 2010.
[120] G.-G. Wang, S. Deb, and Z. Cui, "Monarch butterfly optimization," *Neural Computing and Applications*. New York, NY, USA: Springer, May 19, 2015, pp. 1–20.
[121] A. O. Topal and O. Altun, "A novel meta-heuristic algorithm: Dynamic virtual bats algorithm," *Inf. Sci.*, vol. 354, pp. 222–235, 2016.
[122] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: Crow search algorithm," *Comput. Struct.*, vol. 169, pp. 1–12, 2016.
[123] B. R. Rajakumar, "The lion's algorithm: A new nature-inspired search algorithm," *Procedia Technol.*, vol. 6, pp. 126–135, 2012.
[124] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: Theory and application," *Adv. Eng. Softw.*, vol. 105, pp. 30–47, 2017.
[125] M. Taherdangkoo, M. H. Shirzadi, and M. H. Bagheri, "A novel meta-heuristic algorithm for numerical function optimization: Blind, naked mole-rats (BNMR) algorithm," *Sci. Res. Essays*, vol. 7, no. 41, pp. 3566–3583, 2012.
[126] J. Hu, X. Zeng, and J. Xiao, "Artificial fish school algorithm for function optimization," in *Proc. IEEE 2nd Int. Conf. Inf. Eng. Comput. Sci.*, 2010, pp. 1–4.
[127] W. W. Li, H. Wang, and Z. J. Zou, "Function optimization method based on bacterial colony chemotaxis," *J. Circuits Syst.*, vol. 10, pp. 58–63, 2005.
[128] B. Niu and H. Wang, "Bacterial colony optimization," *Discr. Dyn. Nature Soc.*, vol. 2012, 2012, Art. no. 698057.
[129] Y. Zhou, F. He, and Y. Qiu, "Dynamic strategy based parallel ant colony optimization on GPUs for TSPs," *Sci. China Inf. Sci.*, vol. 60, 2017, Art. no. 068102. [Online]. Available: https://doi.org/10.1007/s11432–015-0594-2
[130] H. Lloyd and M. Amos, "Analysis of independent roulette selection in parallel ant colony optimization" in *Proc. Genetic Evol. Comput. Conf.*, 2017, pp. 19–26.
[131] M. Hajjem, H. Bouziri, E. Talbi, and K. Mellouli, "Parallel ant colony optimization for evacuation planning," in *Proc. Genetic Evol. Comput. Conf.*, 2017, pp. 51–52.
[132] J. Qu, X. Liu, M. Sun, and F. Qi, "GPU-based parallel particle swarm optimization methods for graph drawing," *Discr. Dyn. Nature Soc.*, vol. 2017, 2017, Art. no. 2013673.
[133] C. Zhang and Y. Li, "Cloud computing scheduling strategy based on multi-group parallel particle swarm optimization," *Int. J. Multimedia Ubiquitous Eng.*, vol. 12, no. 2, pp. 195–204, 2017.
[134] A. Nourmohammadzadeh, S. Hartmann, and H. Ma, "A parallel hybrid GA-PSO approach with dynamic rule-based parameter setting," in *Proc. Genetic Evol. Comput. Conf.*, 2017, pp. 215–216.
[135] C.-H. Yang, Y.-S. Lin, L.-Y. Chuang, and H.-W. Chang, "A particle swarm optimization-based approach with local search for predicting protein folding," *J. Comput. Biol.*, vol. 24, pp. 981–994, Oct. 2017. [Online]. Available: https://doi.org/10.1089/cmb.2016.0104
[136] V. M. Kureichik and A. Kazharov, "Using fuzzy logic controller in ant colony optimization," *Artif. Intell. Perspectives Appl.*, vol. 347, pp. 151–158, 2015.
[137] A. Tangherloni, L. Rundo, and M. S. Nobile, "Proactive particles in swarm optimization: A settings-free algorithm for real-parameter single objective optimization problems," in *Proc. IEEE Congr. Evol. Comput.*, 2017, pp. 1940–1947.
[138] Y. Chen, "SwarmDolphin—The swarm dolphin algorithm (SDA)," MathWorks, Natick, MA, USA. 2013. [Online]. Available: https://au.mathworks.com/matlabcentral/fileexchange/45965-swarmdolphin-the-swarm-dolphin-algorithm–sda-. Accessed on: Nov. 2, 2017.
[139] Y. Chen, "SwarmWolf—The artificial wolf pack algorithm (AWPA)," MathWorks, Natick, MA, USA. 2013. [Online]. Available: https://au.mathworks.com/matlabcentral/fileexchange/48469-swarmwolf-the-artificial-wolf-pack-algorithm–awpa-. Accessed on: Nov. 2, 2017.

Adam Slowik (M'07–SM'12) was born in Warsaw, Poland, in 1977. He received the Ph.D. degree in electronics with distinction from the Department of Electronics and Computer Science, Koszalin University of Technology, Koszalin, Poland, in 2007, and the Dr. Habil. (D.Sc.) degree in computer science from the Department of Mechanical Engineering and Computer Science, Czestochowa University of Technology, Czestochowa, Poland, in 2013.

Since October 2013, he has been an Associate Professor with the Department of Electronics and Computer Science, Koszalin University of Technology. His research interests include soft computing, computational intelligence, machine learning, and bioinspired global optimization algorithms and their applications.

Dr. Slowik is an Associate Editor for the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, and a Reviewer for many international scientific journals.

Halina Kwasnicka received the D.Sc. degree in computer science from the Faculty of Electronics, Wroclaw University of Science and Technology, Poland, in 2000.

She is currently a Professor of computer science, the Head of the Department of Computational Intelligence, and the Director of the Graduate School, Faculty of Computer Science and Management, Wroclaw University of Science and Technology, Wroclaw, Poland. In 2009, she was nominated by the President of Poland as the Full Professor (the highest scientific title in Poland). She is currently on the editorial boards of several journals and conferences. She was a leader of some research projects on applications of artificial intelligence methods in medicine, image analysis, and interpretations. She is active in international cooperation, participated in the EU projects, and was the coordinator of the Polish–Singapore Joint Research Project.