Particle Swarm Optimization: A Comprehensive Survey

Article *in* IEEE Access · January 2022

DOI: 10.1109/ACCESS.2022.3142859

Tareq M. Shami
The University of York
19 PUBLICATIONS **208** CITATIONS
SEE PROFILE

Ayman A. El-Saleh
A'Sharqiyah University
92 PUBLICATIONS **748** CITATIONS
SEE PROFILE

Mohammed Alswaitti
University of Luxembourg
23 PUBLICATIONS **601** CITATIONS
SEE PROFILE

Qasem Al-Tashi
University of Texas MD Anderson Cancer Center
37 PUBLICATIONS **1,000** CITATIONS
SEE PROFILE

IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Particle Swarm Optimization: A Comprehensive Survey

Tareq M. Shami[1], Ayman A. El-Saleh[2], (Senior Member, IEEE), Mohammed Alswaitti[3], (Member, IEEE), Qasem Al-Tashi[4,5] , Mhd Amen Summakieh[6] (Member, IEEE), and Seyedali Mirjalili[7,8] (Senior Member, IEEE)

[1]Department of Electronic Engineering, University of York, York YO10 5DD, U.K
[2]Department of Electronics and Communication Engineering, College of Engineering, A'Sharqiyah University, Ibra 400, Oman
[3]School of Electrical and Computer Engineering, Department of Information and Communication Technology, Xiamen University Malaysia, Sepang, 43900, Malaysia
[4]Department of Imaging Physics, The University of Texas MD Anderson Cancer Center, Houston, TX, USA
[5]Faculty of Administrative and Computer Sciences, University of Albaydha, Albaydha CV46+6X, Yemen
[6]Faculty of Engineering, Multimedia University, 63100 Cyberjaya, Selangor, Malaysia
[7]Centre for Artificial Intelligence Research and Optimization, Torrens University Australia, Fortitude Valley, QLD 4006, Australia
[8]Yonsei Frontier Lab, Yonsei University, Seoul, Republic of Korea

Corresponding author: Mohammed Alswaitti (e-mail: alswaitti.mohammed@xmu.edu.my).

This research was supported by the Research Management Center, Xiamen University Malaysia under XMUM Research Program Cycle 4 (Grant No: XMUMRF/2019-C4/IECE/0012).

ABSTRACT Particle swarm optimization (PSO) is one of the most well-regarded swarm-based algorithms in the literature. Although the original PSO has shown good optimization performance, it still severely suffers from premature convergence. As a result, many researchers have been modifying it resulting in a large number of PSO variants with either slightly or significantly better performance. Mainly, the standard PSO has been modified by four main strategies: modification of the PSO controlling parameters, hybridizing PSO with other well-known meta-heuristic algorithms such as genetic algorithm (GA) and differential evolution (DE), cooperation and multi-swarm techniques. This paper attempts to provide a comprehensive review of PSO, including the basic concepts of PSO, binary PSO, neighborhood topologies in PSO, recent and historical PSO variants, remarkable engineering applications of PSO, and its drawbacks. Moreover, this paper reviews recent studies that utilize PSO to solve feature selection problems. Finally, eight potential research directions that can help researchers further enhance the performance of PSO are provided.

INDEX TERMS Applications of PSO, Binary PSO, evolutionary computation, feature selection, hybrid algorithms, meta-heuristic algorithms, Particle Swarm Optimization, PSO variants

I. INTRODUCTION

A lot of engineering applications, such as electrical power systems and signal processing, require an efficient and effective algorithm that can solve their filed-related optimization problems. Real-world optimization problems have been solved by swarm algorithms such as particle swarm optimization (PSO) [1] and ant colony optimization (ACO) [2] as well as other meta-heuristic algorithms including genetic algorithm (GA) [3] and differential evolution (DE) [4].

Generally, most meta-heuristic algorithms can solve many different types of optimizations problems. Nevertheless, these algorithms may have one or more of the following drawbacks:

- Having a lot of parameters to be tuned.
- Requiring high programming skills to build the algorithm.
- High computational cost.
- The need of transforming algorithms into binary forms.

PSO was initially introduced by Kennedy and Eberhart [5] in 1995. The PSO algorithm has attracted a lot of researchers in the last decade due to its simple implementation and fewer controlling parameters. The idea and formulation of the PSO algorithm were stimulated from observing the societal behavior of birds flocking and fish schooling. In nature, a swarm of birds flies in the space following a leader who has the closest position to the food. The social behavior of birds can be translated into algorithmic operations, as in PSO, to solve optimization problems where the swarm of birds is interpreted as a swarm of particles and each particle represents a candidate solution. The swarm of particles searches the space in given dimensions and finds the best solution that optimizes the problem at hand. The following points summarize some of the facts that make the PSO algorithm an attractive optimization algorithm:

- PSO is simple to implement and code.
- PSO has only three controlling parameters (inertia weight, cognitive ratio, and social ratio). A slight change in any of these three controlling parameters results in a different performance as shown in [6] and [7].
- PSO is flexible to hybridize with other optimization algorithms.

PSO is efficient in controlling the balance between exploration and exploitation. Particles in the exploration phase explore the space extensively while the exploitation phase focuses on promising regions. The more balance between exploration and exploitation, the better the PSO performance.

The abovementioned advantages have made PSO a promising candidate for optimizing a wide variety of real-world optimization problems and applications. In the literature, there have been a few PSO review papers that can be split into two categories: the first category reviews PSO and its applications on a specific field [8, 9] whereas the second reviews existing PSO variants [10, 11]. Although the article in [10] reviewed recent studies on PSO, the authors considered PSO in continuous search space only whereas PSO in the binary form was somehow excluded. In addition, the authors did not consider several important aspects such as the applications of PSO in optimization problems.

A recent article [11] reviewed the research works carried on PSO but it was limited to binary PSO variants only. Recently, a survey paper on PSO has been published in [12] where several PSO variants in both continuous and discrete spaces are reviewed. However, the article does not include neighborhood topologies as well as the hybridization of PSO with other common meta-heuristic algorithms such as ACO and gravitational search algorithm (GSA). In addition, it is only focused on the application of PSO on solar photovoltaic systems without considering other engineering applications of PSO. Table I summarizes recent and important state-of-the-art PSO survey papers.

The main aim of this paper is to present a comprehensive review of PSO that includes continuous PSO, binary PSO, different PSO topologies, hybrid PSO variants, types of PSO variants (e.g., cooperative PSO and multi-swarm PSO), and the applications of PSO variants in optimization problems. More importantly, this review paper focuses on PSO-based feature selection. To the best of the authors' knowledge, there has been no publication on a comprehensive survey that covers the recent advances in PSO variant developments and the implementation of PSO to solve feature selection problems.

The main contributions of this review article can be summarized as follows:

1- A comprehensive and critical review of PSO and its variants is provided. The limitations of existing PSO variants are identified and some insightful recommendations are provided to overcome these limitations. In addition, clear guidance that includes that the essential steps to develop novel robust PSO variants is provided.

2- This paper attempts to provide a thorough review of the applications of PSO to feature selection problems due to their extreme importance in the artificial intelligence field. Moreover, a comprehensive review of PSO-based feature selection is still lacking.

3- Eight potential research directions are identified to further enhance the optimization performance of PSO.

The rest of this paper is organized as follows. Section II illustrates the formulation of the PSO algorithm and other basic concepts related to PSO. It also highlights different neighborhood topologies used in PSO. In Section III, the modifications introduced to the original PSO by inertia weight and constriction factor concepts are discussed. In addition, it reviews several strategies that have been used to control the PSO parameters and it critically reviews several recent high-performance PSO variants. Section III also reviews historical prominent variants of PSO. Section IV presents the PSO in binary form and its variants. In Section V, the steps required for validating novel PSO variants are provided. Section VI focuses on the application of PSO to solve feature selection problems. Moreover, prominent engineering applications of PSO are overviewed in Section VI. Section VII demonstrates the drawbacks of PSO while Section VII provides some potential research directions that can help PSO researchers to enhance the performance of PSO further. Finally, Section IX concludes the overall remarks of this paper.

TABLE I
SUMMARY OF THE STATE-OF-THE-ART PSO REVIEW PAPERS

Ref.	Basic PSO concepts	Inertia weight Modifications	Controlling parameters	Neighborhood topologies	PSO variants	Binary PSO	PSO applications	PSO-based feature selection
This paper	✓	✓	✓	✓	✓	✓	✓	✓
Jain et al. [13]	✓	✓	×	×	✓	×	✓	×

Bonyadi and Michalewicz [14]	✓	✓	✓	✓	✓	×	×	×
Sibalija [15]	✓	×	×	×	✓	×	✓	×
Xin et al. [16]	✓	×	×	✓	✓	×	✓	×
Rana et al. [17]	✓	×	×	×	✓	×	✓	×
Jordehi [18]	✓	×	✓	×	✓	×	✓	×
Song and Gu [19]	✓	×	✓	×	✓	×	✓	×
Juneja and Nagar [20]	✓	×	×	×	✓	×	×	×
Esmin et al. [21]	✓	×	×	×	✓	×	✓	×
Fang et al. [22]	✓	×	✓	×	✓	×	✓	×
Khare and Rangnekar [12]	✓	✓	×	×	✓	✓	✓	×
Tian [23]	✓	×	×	×	×	×	×	×
Kulkarni et al. [24]	✓	×	×	×	✓	×	✓	×
Jordehi [25]	✓	×	×	×	✓	×	×	×
Jordehi and Jasni [11]	✓	×	×	×	✓	✓	✓	×
Jordehi [26]	✓	×	×	×	✓	×	×	×
Engelbrecht [27]	✓	×	×	×	✓	×	×	×
Sengupta et al. [28]	✓	✓	×	✓	✓	✓	✓	×
Banks et al. [29]	✓	×	×	×	×	✓	✓	×
Banks et al. [30]	✓	×	×	✓	✓	×	✓	×

II. BASIC CONCEPTS OF PSO

A. PARTICLE SWARM OPTIMIZATION

The first PSO was presented by Kennedy and Eberhart as a continuous real-valued algorithm [5]. This version is referred to as the standard PSO (SPSO) throughout this paper. In SPSO, a swarm of particles flies in a D-dimensional search space seeking an optimal solution. Each particle i possesses a current velocity vector $V_i = [v_{i1}, v_{i2}, \dots, v_{iD}]$ and a current position vector $X_i = [x_{i1}, x_{i2}, \dots, x_{iD}]$, where D is the number of dimensions. The SPSO process starts by randomly initializing V_i and X_i. Then, in each iteration, the best position that has been found by particle i $Pbest_i = [Pbest_{i1}, Pbest_{i2}, \dots, Pbest_{iD}]$ and the best position that has been found by the whole swarm $Gbest = [Gbest_1, Gbest_2, \dots, Gbest_D]$ guide particle i to update its velocity and position by (1) and (2):

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1 \big(Pbest_{id}(t) - x_{id}(t)\big) + c_2 r_2 \big(Gbest_d(t) - x_{id}(t)\big), \quad (1)$$

$$x_{id}(t+1) = x_{id}(t) + v_{id}(t+1), \quad (2)$$

where c_1 and c_2 are the cognitive and social acceleration coefficients, and r_1 and r_2 are two uniform random values generated within [0,1]interval. The pseudo-code of the SPSO for solving a minimization problem is shown in Algorithm 1.

Algorithm 1 The pseudo-code of the SPSO for solving a minimization problem [1]

```
Initialization
Define the swarm size $S$ and the number of dimensions $D$
for each particle $i \in [1..S]$
Randomly generate $X_i$ and $V_i$, and evaluate the fitness of $X_i$ denoting it as $f(X_i)$
Set $Pbest_i = X_i$ and $f(Pbest_i)$= $f(X_i)$
end for
Set $Gbest = Pbest_1$ and $f(Gbest) = f(Pbest_1)$
for each particle $i \in [1..S]$
    if $f(Pbest_i) < f(Gbest)$ then
        $f(Gbest) = f(Pbest_i)$
    end if
end for
while $t <$ maximum number of iterations
for each particle $i \in [1..S]$
Evaluate its velocity $v_{id}(t+1)$ using Equation (1)
Update the position $x_{id}(t+1)$of the particle using Equation (2)
    if $f(x_i(t+1)) <$ $f(Pbest_i)$ then
    $Pbest_i = x_i(t+1)$
        $f(Pbest_i) = f(x_i(t+1)$
    end if
    if $f(Pbest_i) < f(Gbest)$ then
      $Gbest = Pbest_i$
    $f(Gbest) = f(Pbest_i)$
    end if
end for
$t = t + 1$
end while
return Gbest
```

B. VELOCITY CLAMPING

Velocity clamping was initially introduced by Eberhart and Kennedy [31] to avoid the velocity explosion and divergence. Velocity clamping limits the particles to move within a boundary in the search space by setting up a maximum velocity V_{max}. If the updated velocity of a particle is found to exceed the maximum velocity V_{max}, then it is set to V_{max} as follows:

$$v_{id}(t+1) = \min(v_{id}(t+1), V_{max}) \quad (3)$$

Although the velocity clamping helps to prevent the velocity from explosion, finding a proper value of V_{max} is very essential and it is not an easy task. A poor performance might occur if the V_{max} is not selected properly. For large values of V_{max} , the particles might fly in a very random manner and skip the optimal solution. On the contrary, for small values of V_{max}, the particles would have a very narrow search space which might result in being trapped in a local optimum. To resolve this critical problem, the maximum velocity V_{max} can be set as follows [32, 33]:

$$V_{max} = \delta(x_{max} - x_{min}) \quad (4)$$

where x_{max} and x_{min} are the maximum and minimum values of the search space boundary respectively, and $\delta \in (0,1]$.

C. POPULATION SIZE

Population size is defined as the number of particles in the swarm. It is a crucial parameter that characterizes the convergence performance of PSO. The main concern here is finding the optimal swarm size at which the best convergence performance of PSO can be attained. This concern has been addressed in [34, 35] where the effect of the swarm size on PSO performance was investigated. The conclusion drawn in [34, 36] states that a small number of particles does not support the swarm to explore more areas in the search space and produces poor solutions while a large number of particles improves the solution quality yet increases the computational complexity. Also, it is concluded that the optimal swarm size relies on the characteristics of the fitness function to be optimized. In the literature of PSO, it is common to set the population size to a size between 20 to 50 particles [37-41].

D. STOPPING CRITERIA

Typically, there are two types of stopping criteria that are used to terminate the PSO run. In the first stopping criterion, the execution of PSO stops when a predefined number of iterations is reached. This criterion has been widely used in the literature (e.g., [41], [40]). The second stopping criterion is the number of function evaluations (FEs) [37, 42-44], calculated as follows:

$$FEs = S \times T \quad (5)$$

where S is the swarm size and T is the maximum number of iterations.

E. CONTROLLING PARAMETERS OF PSO

In general, PSO has three main controlling parameters: inertia weight w, the cognitive component c_1, and the social component c_2. These parameters have a remarkable effect on the PSO performance where the best performance can only be obtained by a proper setting of these parameters. In the literature, many research efforts have been carried out to enhance the performance of PSO by tuning these controlling parameters through different mechanisms. The following subsections focus on the state-of-the-art mechanisms for tuning these three parameters.

1) INERTIA WEIGHT

The existing inertia weight mechanisms can be classified into three groups. The first group includes mechanisms where the inertia weight is either static or random. This type of mechanism does not require any feedback or historical knowledge input. In the second group, the inertia weight changes with time. In other words, the inertia weight is a function of the iteration number. This mechanism is known as time-varying inertia weight. The third group is called adaptive inertia weight where the inertia weight keeps adjusting its value based on a feedback parameter. These three mechanisms are further elaborated as follows:

THE STATIC AND RANDOM INERTIA WEIGHT

As mentioned earlier, the inertia weight was introduced by Shi and Eberhart. In their work, a range of inertia weight values have been tested and the results showed that a better performance is obtained when w is in the range [0.8,1.2].

In [47], the inertia weight was presented as a random value. This method is suitable for applications in a dynamic environment since it is not easy to predict whether a large or a small value of w is needed.

$$w = 0.5 + (random(0,1))/2 \tag{6}$$

where $random(0,1)$ is a random value in the range [0,1]. Therefore, w is limited to values in the range [0.5,1].

TIME VARYING INERTIA WEIGHT

In PSO, an extensive global search (exploration) is required at the early part of the process while the latter part requires focused local search (exploitation). A static inertia weight cannot meet such requirements. Thus, Shi and Eberhart [48] introduced the first time-varying inertia weight method called linearly-varying inertia weight (LVIW) to address this issue. The mathematical formula of this method is expressed as follows:

$$w(t) = (w_{max} - w_{min})\left(\frac{T-t}{T}\right) + w_{min}, \tag{7}$$

where w_{max} and w_{min} are the initial and final values of the inertia weight, respectively, T is the maximum number of iterations, and t is the number of the current iteration.

In their experimental study, Shi and Eberhart [48] noticed that better performance is achieved if the PSO run starts by choosing an inertia weight value of 0.9 and linearly decreasing it until it reaches a value of 0.4 by the end of the PSO run. This setting indicates that a global search is performed at the beginning of the PSO run and it gradually decreases to refine the search to be locally focused. LVIW is one of the most common, if not the most common, time-varying techniques that have been widely used by many researchers. Besides this technique, a lot of time-varying inertia weight techniques have been proposed with different performance achievements. The formulae of such techniques are presented in Table 2.

ADAPTIVE INERTIA WEIGHT

In this group, the value of the inertia weight is adjusted based on at least one feedback parameter. Utilizing the concept of success rate [49], an adaptive inertia weight technique has been proposed in [50]. This adaptive technique considers the percentage of success as the feedback parameter. The inertia weight of this adaptive strategy is expressed as follows:

$$w(t) = (w_{max} - w_{min})P_s(t) + w_{min}, \tag{8}$$

where w_{max} and w_{min} are in the range $[0,1]$ and $P_s \in [0,1]$ is the percentage of particles that succeeded to enhance their fitness in the previous iteration. Other adaptive inertia weight strategies are shown in Table II.

2) ACCELERATION COEFFICIENTS

The acceleration coefficients c_1 and c_2 guide the PSO search towards the optimal solution. In [5], it was pointed out that a relatively high value of c_1 compared to c_2 causes particles to extremely wander in the search space. Conversely, a relatively high value of c_2 might cause the problem of premature convergence. The authors recommended to statically set the values of c_1 and c_2 to 2. Since then, a lot of authors followed this recommendation in their PSO studies. Although this setting appears to be the most common static strategy for c_1 and c_2 , other settings such as $c_1 = c_2 = 1.49$ are also common. In [40], a hierarchical PSO with a time-varying acceleration coefficient (HPSO-TVAC) is proposed. At the beginning of the HPSO-TVAC process, it is suggested to have a large value of c_1 and a small value of c_2 to let particles perform extensive search. On the contrary, a small value of c_1 and a large value of c_2 help particles to focus more on exploitation at the end of the searching process. The following mathematical expressions illustrates how the values of c_1 and c_2 are gradually varied:

$$c_1 = (c_{1f} - c_{1i})\frac{t}{T} + c_{1i} \tag{9}$$

$$c_2 = (c_{2f} - c_{2i})\frac{t}{T} + c_{2i} \tag{10}$$

where the subscripts f and i denote the final and initial values, respectively. As suggested in [40], the values of c_{1f}, c_{1i}, c_{2f} and c_{2i} should be set to 0.5, 2.5, 2.5, 0.5, respectively.

E. NEIGHBORHOOD TOPOLOGIES IN PSO

Particles in a swarm are connected in a specific structure commonly known as a neighborhood topology within which they communicate with each other and share information. A study on how the neighborhood topology could influence the behavior of PSO operation was presented in [66]. Experimental results revealed that some neighborhood topologies perform better than others. The following subsections present various neighborhood topologies that have been used in PSO studies and applications.

1) STAR TOPOLOGY

The first PSO algorithm that was introduced in [5] was developed using a star topology where each particle considers all other particles as its neighbors. The star topology is also called *Gbest* in which all particles move towards the best global particle in the swarm. The velocity and position update equations for the star topology are the same equations in (1) and (2), respectively. The star topology achieves the fastest convergence among other topologies as it has a great exploration capability. However, it often suffers from convergence to local optima. The star topology has been widely used by many researchers in different applications due to its simple structure and fast convergence behavior.

2) RING TOPOLOGY

In the ring topology, each particle is connected to its two immediate neighbors forming a circle [66]. The ring topology is also known as $lbest$ in which a particle is attracted by its best local particle that has been found in its neighborhood. The velocity update equation for the ring topology is modified as follows:

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1 \big(Pbest_{id}(t) - x_{id}(t)\big) + c_2 r_2 \big(lbest_d(t) - x_{id}(t)\big) \qquad (11)$$

where $lbest_d(t)$ is the best local position found in the i^{th} particle neighborhood. The two neighbors of the i^{th} particle are the $(i-1)^{th}$ particle and $(i+1)^{th}$ particle. Particles in the ring topology fly towards their local best position. This provides diversity and protects the algorithm from becoming stuck at local optima. However, the convergence speed of the ring topology decreases since more information needs to be exchanged. In addition, the ring structure is not as simple as the star structure. One of the earliest PSO variants that implemented the ring topology is the fully informed particle swarm (FIPS) [67]. In FIPS, the particle's velocity relies on all the particles' best positions of its neighbors. Moreover, FIPS applies the concept of the constriction factor.

3) VON NEUMANN

The Von Neumann topology is a rectangle matrix, for example, (3×4), resulting in a population of 12 particles where each particle is connected to the particles below, above, on its right and left sides, and wrapping the edges. The Von Neumann showed superior performance over other topologies in many test problems [68].

4) DYNAMIC TOPOLOGY

In the dynamic topology, the neighborhood is refreshed and regrouped after a certain number of iterations. In [69], a dynamic neighborhood is developed where each particle, in the early stage of the PSO run, exchanges information with only a small number of particles. This enhances the exploration process in the early stage of the run. As the number of iterations increases, the neighborhood of each particle increases as well. At the end of the PSO run, all particles communicate with each other resulting in a higher exploitation capability. A dynamic neighborhood strategy named dynamic neighborhood learning PSO (DNLPSO) is presented in [70]. DNLPSO improved the CLPSO algorithm [71] by making the learning particle's neighborhood dynamic.

5) OTHER TOPOLOGIES

In [72], a complex neighborhood PSO was proposed where the neighborhood structure is a complex network that can be tuned during the PSO run. The star topology and the ring topology were combined in [73] to form a single PSO named unified PSO (UPSO). Other common topologies such as the pyramid, wheel, and cluster topologies are presented in [32, 74-76].

III. PSO VARIANTS

Since the introduction of PSO, many new PSO variants have been proposed to enhance its optimization performance. Mainly, PSO is modified by developing new controlling parameters strategies, hybridizing PSO with other well-known meta-heuristic algorithms, cooperation and multi-swarm approaches. This section reviews recent and historical PSO variants and identifies their limitations.

A. MODIFICATIONS OF PSO BY THE INERTIA WEIGHT AND CONSTRUCTION FACTOR METHODS

1) INERTIA WEIGHT

To improve the convergence speed of SPSO, Yuhui and Eberhart [45] modified the SPSO velocity update equation $v_{id}(t+1)$ by introducing a scaling factor that is multiplied by $v_{id}(t)$. This scaling factor is termed as *Inertia weight* and denoted by w. Based on this modification, the velocity update equation in (1) becomes now in the following form:

$$v_{id}(t+1) = w v_{id}(t) + c_1 r_1 \big(Pbest_{id}(t) - x_{id}(t)\big) + c_2 r_2 \big(Gbest_d(t) - x_{id}(t)\big) \qquad (12)$$

2) CONSTRICTION FACTOR

Clerc and Kennedy modified the velocity update equation of the SPSO by introducing the concept of constriction factor [46]. The role of the constriction factor is to ensure that the PSO algorithm converges without using velocity clamping. By using the constriction factor, the velocity update equation becomes as follows:

$$v_{id}(t+1) = \chi\big[v_{id}(t) + c_1 r_1 \big(Pbest_{id}(t) - x_{id}(t)\big) + c_2 r_2 \big(Gbest_d(t) - x_{id}(t)\big)\big] \qquad (13)$$

where $\chi = 2k / \left|2 - \phi - \sqrt{\phi^2 - 4\phi}\right|$, $\phi = c_1 + c_2$ and $k \in (0,1]$.

The value of ϕ must be > 4 to ensure convergence. The value of k controls the balance between exploration and exploitation. The exploration mode takes place when the value of k is large whereas the exploitation mode will be activated when the value of k is small. Eberhart and Shi stated that a combination of constriction factor and velocity clamping would speed up the convergence rate [28].

TABLE II
A SUMMARY OF SEVERAL STATE-OF-THE-ART INERTIA WEIGHT TECHNIQUES

No	Ref.	Technique	Formula
1	Yuhui and Eberhart [45]	Static w	$w = c$
2	Eberhart and Yuhui [51]	Random	$w = 0.5 + \frac{random(0,1)}{2}$
3	Yuhui and Eberhart [48]	Linear decreasing	$w(t) = (w_{max} - w_{min})\left(\frac{T - t}{T}\right) + w_{min}$
4	Lee and Jong-Bae [52]	Linear decreasing	$w(t) = w_{max} - \left(\frac{w_{max} - w_{min}}{T}\right) t$
5	Jiao et al. [53]	Nonlinear time-varying	$w(t) = w_{intial}\, u^{t}$
6	Fan and Chiu [54]	Nonlinear time-varying	$w(t) = \left(\frac{2}{t}\right)^{0.3}$
7	Chatterjee and Siarry [55]	Nonlinear time-varying	$w(t) = w_{min} + \left\{\frac{(T - t)^{n}}{(T)^{n}}\right\}(w_{max} - w_{min})$
8	Guimin et al. [56]	Exponential (e1)	$w(t) = w_{min} + (w_{max} - w_{min})\, e^{{-t}/{\left(\frac{T}{10}\right)}}$
9	Guimin et al. [56]	Exponential (e2)	$w(t) = w_{min} + (w_{max} - w_{min})\, e^{-\left[{t}/{\frac{T}{4}}\right]^{2}}$
10	Hui-Rong and Yue-Lin [57]	Exponential decreasing	$w(t) = (w_{max} - w_{min} - d_{1})e^{\frac{1}{1+d_{2}({t}/{T})}}$
11	Lu et al. [58]	Exponential decreasing	$w(t) = w_{min} + (w_{max} - w_{min})e^{-\frac{ct}{T}}$
12	Yong et al. [59]	Chaotic	$w(t) = (w_{max} - w_{min})\frac{T - t}{T} + z w_{min}$
13	Yang et al. [60]	Adaptive	$w_{i}^{t} = w_{initial} - \alpha(1 - h_{i}^{t}) + \beta s$
14	Panigrahi et al. [61]	Adaptive	$w_{i} = w_{min} + \frac{(w_{max} - w_{min})Rank_{i}}{S}$
15	Suresh et al. [62]	Adaptive	$w_{i} = w_{intial}\left(1 - \frac{dist_{i}}{dist_{max}}\right)$
16	Kaiyou et al. [63]	Adaptive	$w(t) = \frac{1 - \left(\frac{t}{T}\right)}{1 + s\left(\frac{t}{T}\right)}$
17	Arumugam and Rao [64]	Adaptive	$w_{i} = 1.1 - \frac{Gbest}{(Pbest_{i})_{avgerage}}$
18	Nickabadi and Ebadzadeh [50]	Adaptive	$w(t) = (w_{max} - w_{min})P_{s}(t) + w_{min}$
19	Zhi-Hui et al. [65]	Adaptive	$w(f) = \frac{1}{1 + 0.5e^{-2.6f}}$

However, the constriction factor approach still faces the problem of becoming trapped in local optima.

B. RECENT PSO VARIANTS

In the last few years, many PSO variants have been proposed to overcome the limitations of the original PSO algorithm and the historical PSO variants. This part critically reviews PSO variants that are published recently. The authors in [77] proposed a new PSO variant named prey-predator PSO (PP-PSO) that implements catch, escape, and breeding strategies that can assist in enhancing the convergence speed and reduce the computational time. The proposed approach is tested on 10 classical benchmarking functions and the CEC2017 test suite for 10, 30, and 100 dimensions. Although this approach has shown good performance, this good performance comes at the expense of an unreasonable number of function evaluations that can reach up to 10^6 function evaluations. Moreover, the proposed variant was not tested on real-world engineering problems. In [78], a multi-swarm PSO is proposed where a sub-swarm focuses on exploration while a different sub-swarm is performs exploitation. The performance of the proposed variant is tested on the CEC 2015 on 10 and 30 dimensions. The performance of this variant on high-dimensional problems as well as real-world engineering problems is not investigated. The work in [79] proposed a competition-based PSO variant where each particle is allocated a competition coefficient that allows to distinguish particles and divide them into three groups. The proposed method is tested on the CEC 2013 benchmarking functions for 10 and 30 dimensions and retarder designing problem as a real-world engineering problem. The impact of increasing the dimensions on the performance of the competition-based PSO needs to be investigated. The authors in [80] developed a new PSO variant that utilizes PSO with two differential mutations. The proposed approach was tested on 16 well-known benchmarking functions and CEC 2013 on 30 dimensions only.

In [81], a novel PSO variant is proposed where the main contribution is the utilization of the sigmoid function to update the PSO acceleration coefficients. The effectiveness of the proposed variant is evaluated by testing it only on 8 classical benchmarking functions on 30 dimensions. Further work is needed to evaluate the performance of this variant when it solves constrained optimization problems and real-world engineering problems. An improved social learning PSO is developed in [82] where the three best particles are updated using a differential mutation strategy. The developed approach is tested on the CEC 2013 test suite on 30 and 50 dimensions. Tough the proposed variant has shown good performance, it was compared with PSO variants only. Moreover, its performance on real-world optimization problems is not studied. The authors in [83] developed a novel PSO variant for constrained optimization problems. The proposed approach was tested on twenty four classical benchmarking functions for low dimensional problems as well as on the reservoir drainage plan optimization problem. With the help of mixed mutation strategies, a new PSO variant is proposed in [84] based on the idea of dividing the total population into an elitist population and a general population. The effectiveness of the proposed algorithm is evaluated by testing its performance on sixteen well-known benchmarking functions for 30, 50, and 100 dimensions. This multi-population PSO variant requires a massive number of function evaluations to achieve good performance. In addition, its performance was not validated on real-world constrained optimization problems. Also, The effectiveness of this variant was compared with PSO variants only.

The original PSO velocity update equations are modified in [85] by adding two new terms that aim to enhance the performance. The new PSO variants are tested on sixteen classical benchmarking functions for 50 dimensions without evaluating their effectiveness on real-world optimization problems. In addition, its performance is compared with PSO variants only. A novel PSO variant that is tested on CEC2013 for 30 dimensions is proposed in [86]. The main concept of the proposed approach is to split the whole population into several sub-swarms using a chaotic sequence. To achieve good performance, this variant requires a massive number of function evaluations which is computationally expensive. Utilizing complex-order derivatives, an improved version of PSO, that is tested on CEC 2017 for 20, 30, and 40, is proposed in [87]. The improved PSO is only compared with PSO variants without considering other well-known meta-heuristic algorithms. Based on forgetting ability and multi-exemplar, a new version of PSO is proposed in [88] where its effectiveness is tested on CEC 2013 for 30 dimensions. Although the proposed approach shows good performance in terms of average fitness and standard deviation for most of the tested functions, massive function evaluations are needed to achieve such performance. In [89], inertia weight PSO [90], CLPSO [91], LIPS [92], HPSO-TVAC [93], and FDR-PSO [94] algorithms are combined to produce a new single variant. The performance of the new variant is evaluated on CEC2005 for 10 and 30 dimensions, and it is compared with PSO variants only.

PSO has recently been hybridized with several meta-heuristic algorithms such as the whale optimization algorithm. PSO is hybridized with the whale optimization algorithm in [95] and its performance is evaluated on 18 classical benchmarking functions as well as on electronic design optimization problems. Although the proposed approach shows good performance, the results are based on only 20 independent runs which might be not enough to produce accurate results. In [96], a hybrid PSO algorithm is developed utilizing an adaptive learning strategy. The effectiveness of the hybrid approach is tested on 12 classical benchmarking test functions and CEC 2013 for only 30 dimensions. Moreover, its performance is compared with PSO variants only. By hybridizing PSO with sine cosine acceleration coefficients, a novel hybrid algorithm is introduced in [97]. The performance of the hybrid algorithm is evaluated on 12 well-known benchmarking functions for 10, 30, and 50 dimensions. However, its performance on constrained optimization problems is not investigated.

In summary, recent PSO variants have shown good optimization performance. However, all the PSO variants presented in this subsection except [79] [83] [95] did not consider constrained real-world optimization problems. Their performance on real-world constrained optimization problems needs to be investigated. The performance of [78-81] [83] [85-89] [95-97] is tested only on low dimension problems. Therefore, further work is needed to validate their effectiveness on high-dimensional problems. The PSO variants presented in [77] [81] [84] [85] [96] did not provide any statistical analysis which is essential to show the significance and the superiority of these variants. Finally, the PSO variants in [82] [84] [85] [87] [89] [96] compared their performance with PSO variants without considering other robust and well-known optimization algorithms. Table III summarizes the recent PSO discussed in this subsection and presents their ideas and limitations.

C. HISTORICAL PSO VARIANTS IN CONTINUOUS SEARCH SPACE

Since the introduction of SPSO in 1995, there has been a continuous research effort in enhancing the convergence speed, quality of achievable solutions, and stability of PSO. This has resulted in an enormous number of PSO variants some of which are dedicated to solving optimization problems in specific applications while the rest are used for general numerical optimization. This subsection discusses in detail the most important historical PSO variants that have been developed since the advent of PSO.

TABLE III

A SUMMARY OF RECENT PSO VARIANTS AND THEIR LIMITATIONS

Ref.	Idea	dim	Benchmarking functions	Maximum number of function evaluations	Real-world engineering problems	Statistical tests	Limitations
Zhang et al. [77]	Catch, escape, and breeding approaches	10, 30, 100,	6 unimodal 4 multimodal CEC 2017	1,000,000	None	None	- Computationally expensive - Untested on real-world problems.
Ye et al. [78]	Multi-swarm approach	10, 30	CEC 2015	250*dim	None	Wilcoxon test	- Untested on high dimensional problems - Untested on real-world problems.
Gou et al. [79]	Competition among particles	10, 30	CEC 2013	2000*dim	Retarder designing problem	Wilcoxon test	- Computationally expensive - Untested on high dimensional problems
Chen et al. [80]	A two sub-swarms approach with differential mutation operations	30	Sixteen classical benchmarking functions CEC 2013	300000	None	t-test	- Computationally expensive - Untested on high dimensional problems - Untested on real-world problems.
Liu et al. [81]	A novel adaptive weighting strategy	30	8 unimodal and multimodal functions	150000	None	None	- Computationally expensive - Untested on high dimensional problems - Untested on real-world problems. - Needs to be tested on diverse benchmarking functions
Zhang et al. [82]	Novel social learning strategy	30, 50	CEC2013	D ×10000	None	Wilcoxon signed rank	- Compared with PSO variants only - Untested on real-world problems.
Kohler et al. [83]	A novel particle update method	2-24	Twenty-four classical benchmarking functions	30,000	Reservoir drainage plan optimization	Wilcoxon signed rank test	- Untested on high dimensional problems - Needs to be tested on diverse real-world problems
Li et al. [84]	Multi-population with mutation strategies	30, 50, 100	Sixteen classical benchmark functions	200000	None	None	- Compared with PSO variants only - Computationally expensive - Untested on real-world problems.
Sedighizadeh et al. [85]	Addition of two new terms to the velocity equation	50	Sixteen classical benchmark functions	10,000	None	None	- Compared with PSO variants only - Untested on high dimensional problems - Untested on real-world problems.

Chen et al. [86]	Dynamic regrouping strategy	30	CEC2013	300000	None	Friedman-test, t-test	- Computationally expensive - Untested on high dimensional problems - Untested on real-world problems.
Machado et al. [87]	Adoption of complex-order derivatives	20, 30, 40	CEC 2017	1000*Dim	None	Friedman test	- Compared with PSO variants only - Untested on high dimensional problems - Untested on real-world problems.
Xia et al. [88]	Multi-exemplar and forgetting ability	30	CEC 2013	10000* Dim	None	Friedman test	- Computationally expensive - Untested on high dimensional problems - Untested on real-world problems.
Lynn and Suganthan [89]	A combination of different PSO algorithms are used	10, 30	CEC 2005	10000* Dim	None	Wilcoxon signed-rank test	- Compared with PSO variants only - Computationally expensive - Untested on high dimensional problems - Untested on real-world problems.
Laskar et al. [95]	PSO is hybridized with whale optimization algorithm	30	18 classical benchmark functions	50000	Three electronic design optimization problems	Friedman test and Wilcoxon test	- Untested on high dimensional problems - Considered only 20 independent runs - Needs to be tested on diverse benchmarking functions.
Wang et al. [96]	Adaptive learning strategy	30	12 classical benchmark test functions and CEC 2013	200000	None	None	- Compared with PSO variants only - Computationally expensive - Untested on high dimensional problems - Untested on real-world problems.
Chen et al. [97]	PSO hybridized with sine cosine acceleration coefficients	10, 30, 50	12 classical benchmarking functions	40000	None	Wilcoxon rank-sum test	- Untested on high dimensional problems - Untested on real-world problems. - Needs to be tested on diverse benchmarking functions.

1) COOPERATIVE PSO

Cooperation, in context to meta-heuristics, is defined as exchanging information between a number of agents to perform a specific task [98]. Though individual human beings can work separately and compete with each other to enhance their performance, better enhancement can be achieved by cooperation. Potter and De Jong [99] applied the cooperation concept in genetic algorithms (GAs). In [100], the same idea was extended to PSO, and a new PSO variant named cooperative particle swarm optimization (CPSO) was introduced.

In SPSO, each particle consists of a *D*-dimensional vector that represents a candidate solution. The updates of position and velocity equations that occur in each iteration treat this *D*-dimensional vector as one entity. Hence, there might be some components that are selected to represent the solution though they are moving far from this solution. These components are wrongly selected since SPSO considers the overall enhancement of the entire vector. Thus, CPSO [100] was introduced to tackle this problem.

CPSO proposed two models denoted as CPSO-S_k and CPSO-H_k. In CPSO-S_k, the entire vector is split into n swarms and each swarm has a 1-D vector. Every single component of the entire vector is optimized by the swarm that it belongs to. In this case, the evaluation of the optimization function is infeasible since the evaluation requires knowledge of the entire *D*-dimensional vector. To handle this, a *context vector* is invoked to form a vector that acts as a suitable input for the optimization function. The context vector can be formed by taking the values of the $Gbest$ particles from each of the n swarms and concatenating them to build up the input vector. To evaluate the fitness for the entire particles in the j^{th} swarm, the j^{th} component takes the value of the first particle of the j^{th} swarm while the rest of the context vector components are kept constant at the $Gbest$ values. The same procedure occurs for the rest of the particles in the j^{th} swarm. Experimentally, CPSO-S_k has been found to be easily stuck in sub-optimal regions of the search space. Thus, CPSO-H_k, which is a combination of CPSO-S_k and SPSO, is used to overcome this problem.

In [101], a new CPSO variant with the concept of dimension partition and adaptive velocity control was proposed. With this approach, the new variant was dedicated to optimizing multimodal functions by using the two-swarm cooperative technique while using adaptive velocity control. In this work, the population is split into two swarms where the SPSO is applied to the first swarm to perform a full dimensional search and a single-dimensional PSO is applied to the second swarm to perform a 1-D search. Information is shared between the two swarms in a communication phase. Unlike the conventional CPSO, the two swarms in this new CPSO variant work concurrently. As for the adaptive velocity control, V_{max} is changed dynamically based on how each particle flies in the search space. This new CPSO variant showed better performance when compared with other variants for most of the tested problems.

The work presented in [102] used the CPSO and inertia weight adaption together to come up with a new PSO variant named adaptive cooperative PSO (ACPSO). This method implemented the CPSO that was presented in [100] and provided an adaptive method that automatically controls the inertia weight. ACPSO was tested only on three benchmarking functions and the results showed that its solution quality and convergence behavior are better than CPSO for all the three tested functions. However, the performance of ACPSO still needs to be thoroughly investigated using other benchmarking functions to prove its effectiveness.

2) MULTI-SWARM PSO

The concept of multi-swarm PSO (MSPSO) has been applied in several PSO research works. In MSPSO, the population of particles is split into sub-swarms where each sub-swarm carries out a specific task. A sub-swarm task might be adjusted as time goes on and information is shared among sub-swarms.

One of the works that considered the use of the MSPSO concept was presented in [103]. This work presented a multi-swarm cooperative PSO (MCPSO) that divides the population into one master swarm and multiple slave swarms. Each slave swarm performs an independent single PSO run to control the diversity of the population whereas the formation of the master swarm depends on its own experience as well as the slave swarm experience. In MCPSO, the master swarm can update its particles by either a sequence of competitions or a sequence of collaboration with the slave swarms. The first case is known as the competitive MCPSO while the second is called the collaborative MCPSO. The performance of MCPSO was evaluated on six benchmarking functions and results have demonstrated that it can perform better than the SPSO [103, 104].

A Multi-swarm Self-adaptive CPSO (MSCPSO) was proposed in [105]. The total population in MSCPSO is split into four sub-swarms where information is shared among themselves. MSCPSO applied three strategies namely cooperative, diversity, and self-adaptive strategies to escape from becoming stuck in local optima, enhance diversity, and obtain better solutions. An attractive feature of this algorithm is that it does not add any complexity to the SPSO algorithm. In other words, its implementation is as simple and easy as the SPSO. MSCPSO was examined only on six benchmarking functions for 10 and 30 dimensions. Although MSCPSO has shown good performance on the six tested benchmarking functions in the cases of 10 and 30 dimensions, there is no proof that this algorithm can show good performance in the case of high-dimension search space or when other benchmarking functions are tested.

A tribal PSO (TPSO) is proposed in [106] where the population is split into several tribes or sub-swarms using a self-clustering algorithm. The process of the TPSO algorithm consists of four major steps: initializing population, using a clustering algorithm to generate tribes, performing the evaluation step where the performance of each particle is evaluated, and finally using the tribe's adaptation method to add and delete particles.

3) HYBRID PSO

In the field of meta-heuristics, hybridization is the process of selecting the best properties of two distinct algorithms that can solve the same problem and joining them together to come up with a novel algorithm that can achieve better results than the individual algorithms. PSO has been hybridized with many evolutionary algorithms such as GA, DE, and ACO to overcome its drawbacks, such as premature convergence. The hybridization of PSO with GA, DE, ACO as well as with other techniques is presented in the following.

HYBRIDIZATION OF PSO WITH GA

GA was initially introduced by John Holland [107] as one of the earliest evolutionary algorithms. Combining PSO with GA is a famous approach that has been widely considered due to the superior convergence performance as compared to the individual PSO and GA.

A hybrid PSO and GA (GA-PSO) was proposed in [108] to solve multimodal problems. The process of GA-PSO starts by creating a population size of $4D$ for a problem with D dimensions. The fitness of each individual is calculated, and individuals are ranked based on their fitness values. The selection, crossover, and mutation operators of GA are applied to the best $2N$ individuals whereas PSO is applied to the worst $2N$ individuals. This hybrid approached is tested on seventeen multimodal functions and it has shown better performance in terms of solution quality and convergence speed when compared with the continuous genetic algorithm (CGA) [109] and Nelder-Mead PSO (NMPSO) [110]. In [111], two-hybrid algorithms named GA-PSO and PSO-GA were introduced. In GA-PSO, the PSO initial population is created by GA, whereas in PSO-GA, the GA initial population is created by PSO. It has been observed that the PSO-GA performs better than GA-PSO, SPSO, and GA.

The work in [112] combined PSO with GA for field development optimization. The resultant hybrid algorithm is called genetical swarm optimization (GSO). In this hybrid algorithm, the population is split into two portions and it is reconstructed by GA and PSO operations in every iteration. A hybridization constant (HC) was introduced to indicate the population percentage that is constructed with GA where $HC = 0$ indicates that only PSO is used and $HC = 1$ indicates that only GA is implemented.

In [113], a hybrid PSO and GA named HPSOGA was proposed. In this approach, the population is split into two groups based on a hybrid probability P. The size of the first group is $M \times P$ where M is the number of particles in the whole population, and the size of the other group is $M - (M \times P)$. The first group updates its particles positions by PSO while particles in the other group are updated by the three GA operations: selection, crossover and mutation. HPSOGA showed that its performance is better than the performance of SPSO.

HYBRIDIZATION OF PSO WITH DE

DE is a population-based algorithm that was first presented by R. Storn and K. Price [114] in 1995 to solve optimization problems. The selection, mutation, and crossover operators of GA are also used in DE but they function differently. One of the advantages of DE is that it maintains diversity; however, unlike PSO, it is unable to keep track of the process history [115]. In [115], a hybrid DE with PSO (DEPSO) algorithm was proposed to solve economic dispatch problems. The overall procedure of this proposed algorithm is based on DE and letting PSO generates a second mutant operator. DEPSO showed its effectiveness in producing good solutions and efficient computation. DE and enhanced PSO (EPSO) were hybridized in [116] where they are executed in parallel and information is exchanged frequently. This approach was applied to design antenna arrays. DEPSO achieves a better global search than the individual DE and EPSO. In [117], a hybrid approach that combines PSO and DE is developed. In this approach, each of the PSO iterations is followed by implementing the three operators of DE (mutation, recombination, and selection) to the best personal positions. During the mutation procedure, six DE mutation techniques can be used. After that, a tournament is conducted to select the best position.

A hybrid PSO and DE (PSO-DE) was proposed in [118] to find the optimal design of water distribution systems. The basic idea behind this approach is that DE is not integrated with PSO at all iterations but only at a predefined interval of iterations. The results of PSO-DE, in solving three water distribution problems, showed better solution accuracy and computation efficiency than PSO. To confirm the effectiveness of PSO-DE, it should be used to solve more complex optimization problems and be compared with DE. In [119], the authors proposed a hybrid PSO and DE (DE-PSO) that is divided into two alternating phases, DE phase, and PSO phase. This hybrid version begins with the DE phase until a trail vector is created. The trail vector is added to the population if it satisfies a predefined requirement, else the proposed algorithm switches to the PSO phase and creates a new potential solution. DE-PSO is evaluated on several numerical benchmarking problems and the results have shown that DE-PSO outperforms the standard PSO and DE. In [120], a hybrid quantum PSO (QPSO) [121] with DE named DEQPSO is presented to solve a route planning problem. The first step in DEQPSO is to update the population by PSO then activate the DE algorithm. This proposed algorithm introduced a new form of vectors called the donor vector which makes the DE in this algorithm somewhat different from the classical DE. Based on simulation results, DEQPSO outperforms QPSO and DE in terms of optimal solution and convergence speed.

HYBRIDIZATION OF PSO WITH OTHER ALGORITHMS

Besides GA and DE, PSO has been hybridized with other algorithms such as ACO [122], gravitational search algorithm (GSA) [123], grey wolf optimizer (GWO) [124], and simulated annealing (SA) [125] [126].

In [122], a hybrid algorithm based on PSO and ACO was proposed. The developed hybrid algorithm is named as hybrid ant particle optimization algorithm (HAP). In each HAP iteration, separate executions of PSO and ACO are performed resulting in a new solution for PSO and another new solution for ACO. The best solution out of these two solutions is chosen to be the global best of the overall system. Particles and ant positions are updated based on the parameters of this obtained global best. HAP has shown that it can achieve better solutions as compared with SPSO and ACO. However, HAP was tested only on simple and low-dimensional benchmarking functions. Its performance in complex high-dimensional optimization problems needs to be investigated. A new hybrid method consisting of PSO and ACO that is used for energy optimization was proposed in [127]. The concept of this approach is to update the direction operator of movement if the best solution of ACO is affected by the best solution of PSO. This hybrid approach was also used in [128] to tune the controller coefficients in wind power plants. In [129], a hybrid PSO with ACO is proposed for the economic dispatch of a power system.

A novel hybrid PSO and GSA (HPSO-GSA) that is tested on only five benchmark functions is proposed in [130]. Results have shown that HPSO-GSA performs better than the individual performance of PSO and GSA for all the selected five benchmarking functions. Another combination of PSO and GSA is the gravitational particle swarm (GPS) [131]. In GPS, the velocities and positions of particles are updated based on the velocity of PSO as well as the acceleration of GSA. The results have demonstrated that GPS outperforms SPSO and GSA. However, the parameter setting in GPS is not optimal. Thus, further work is needed to produce better results through efficient parameter tuning. An improved hybrid version of PSO and GSA called centripetal accelerated PSO (CAPSO) was introduced in [41]. In CAPSO, the standard velocity of PSO shown in (1) is modified by adding two terms called acceleration and centripetal acceleration. This modification is introduced to accelerate the convergence speed and protect the algorithm from becoming trapped in local optima.

A hybrid algorithm that consists of PSO and Legendre pseudo-spectral method (LPM), namely PSO-LPM was proposed in [132]. PSO-LPM was used to solve planning problems. PSO-LPM starts the search process with the PSO algorithm only and it switches to the LPM algorithm if it finds that the change in the fitness function has become smaller than a predefined value. This hybrid approach provides better convergence speed and global search than both the separate PSO and LPM. In addition, its performance is not affected by random initialization. In [133], PSO was combined with the levy flight distribution method resulting in a new PSO variant called levy flight PSO (LFPSO). LFPSO alters the SPSO by adding two new ideas. The first idea is giving each particle a limit value. In each iteration of LFPSO, if a particle does not provide better solutions, the limit value is increased by 1. The second idea is using the Levy distribution method to reallocate the positions of particles that have exceeded the limit value. These two ideas aim to enhance the global search capability and avoid premature convergence to local optima. It has been demonstrated that LFPSO outperforms other PSO variants including CLPSO and HPSO-TVAC as well as other optimization methods like GA and DE.

OTHER HISTORICAL PSO VARIANTS

In [134], a modified PSO with time-varying acceleration coefficients (MPSO-TVAC) was presented. This algorithm proposed a new parameter termed '$rbest$' which provides additional information to each particle. As a result, better exploration is achieved leading to premature convergence avoidance. In this method, each particle chooses any random particle from $Pbest$ of all particles, other than its own $Pbest$, and it considers it as its own $rbest$. The velocity update equation of this algorithm is given in the following form:

$$v_{id}(t+1) = v_{id}(t) + c_1 r_1\big(Pbest_{id}(t) - x_{id}(t)\big) + c_2 r_2\big(Gbest_d(t) - x_{id}(t)\big) \quad + c_3 r_3\big(rbest_{id}(t) - x_{id}(t)\big) \tag{14}$$

where c_3 is an acceleration constant which attracts each particle to move in the direction of $rbest$, and r_3 is a uniform random value in the range [0,1]. In MPSO-TVAC, the acceleration coefficients c_1 and c_2 are varied with time and their formula is provided in [40] whereas the formula of c_3 is expressed as follows:

$$c_3 = c_1\big(1 - exp(-c_2 t)\big) \tag{15}$$

where t is the current iteration number.

In [135], a novel Gaussian PSO named Gaussian-distributed PSO (GDPSO) was presented. In GDPSO, the position of a particle is updated based on Gaussian distribution. This method does not require parameter tuning and its performance in solving high-dimension complex functions is superior to Gaussian PSO (GPSO) [136].

Based on the grey relational analysis, Leu and Yeh [137] proposed a PSO variant termed grey PSO. In each iteration of grey PSO, each particle is assigned a unique inertia weight, a cognitive component, and a social component. This algorithm achieves faster convergence speed and better solution accuracy as compared with PSO-LVIW [45], HPSO-TVAC [40], and APSO [65].

The work in [138] proposed an enhanced PSO incorporating a weighted particle (EPSOWP). EPSOWP calculates a weighted particle that guides the particles of a swarm towards the optimal solution. Based on simulation results, EPSOWP outperforms the SPSO, GA, and DE algorithms on some selected benchmarking functions. In [139], a team-oriented swarm optimization (TOSO) is proposed where the swarm is divided into two teams. The role of the first team is to perform exploration while the second team performs exploitation. The two teams interact with each other by sharing information about *Gbest*. This PSO variant omits the need for the inertia weight, cognitive coefficient, and social coefficient. Instead, it relies on only one parameter known as mutation probability (p_m). Unlike most of the PSO variants, this variant was tested in very high dimensions (up to 1000 dimension) cases. Although TOSO has shown good performance for various benchmarking functions, it still has drawbacks it terms of its exploration capability.

To avoid the problem of the premature convergence of SPSO while maintaining fast convergence, PSO with aging leader and challengers (ALC-PSO) was presented in [37]. In ALC-PSO, the swarm's leader possesses a lifespan that can be adjusted by the leader's leading power (stronger leading power indicates longer life for the leader) and its age increases with time. The other particles of the swarms (challengers) have the chance to claim the leadership once the leader has become old. The leader attracts other particles if its leading power is high; otherwise, new particles are allowed to compete to take the leadership. A median-oriented PSO (MPSO) was introduced in [140] to avoid becoming trapped in local optima and to accelerate the convergence speed. In this approach, each particle updates its velocity based on the current velocity and a median-oriented acceleration. This variant omits the need for the inertia weight w, cognitive coefficient c_1 C_1, and social coefficient c_2 C_2. Another PSO variant called orthogonal learning PSO (OLPSO) was proposed in [42]. This PSO variant uses an orthogonal learning strategy for PSO to achieve faster convergence speed and better solution quality. The role of the orthogonal learning strategy is to let the particles move in better directions. The results of OLPSO demonstrated its superiority in terms of convergence speed and solution quality as compared to the SPSO and some other PSO variants.

IV. PSO IN DISCRETE SEARCH SPACE

Kennedy and Eberhart introduced the binary version of PSO (BPSO) in [141]. The BPSO is applied to solve binary problems where each dimension of a particle can have two states only: 0 or 1. The values of 1 and 0 can have different meanings such as true or false, yes or no, selected or not selected, respectively. The updated velocity $v_{id}(t+1)$ in BPSO is the same as the updated velocity in the continuous PSO but with restricting the values of X_i, $Pbest_i$, and $Gbest$ to binary values. The result of $v_{id}(t+1)$ is real continuous values, though. The $v_{id}(t+1)$ can be limited to have values in the range of [0,1] based on a transfer function. One of the most common transfer functions is the sigmoidal function which is given as follows:

$$sig(v_{id}(t+1)) = \frac{1}{1+e^{-(v_{id}(t+1))}} \qquad (16)$$

Similar to PSO in continuous search space, a particle updates its velocity in BPSO using Equation (1). In BPSO, a particle updates its position based on a probabilistic equation given by:

$$x_{id}(t+1) = \begin{Bmatrix} 0\ if\ r_4 \geq sig(v_{id}(t+1)) \\ 1\ \ f\ r_4 \leq sig(v_{id}(t+1)) \end{Bmatrix} \qquad (17)$$

where r_4 is a uniformly distributed random value in the interval [0,1]. From Equation (17), it is observed that $sig(v_{id}(t+1))$ becomes 0 when the value of $v_{id}(t+1)$ is less than -10. At this state, the updated position $x_{id}(t+1)$ will remain 0 and no bit flip will occur. Similarly, $sig(v_{id}(t+1))$ becomes 1 when the value of $v_{id}(t+1)$ is greater than 10 and the updated position $x_{id}(t+1)$ will remain 1. In [141], it is recommended to limit the velocity to ± 6 where there will be a probability of 0.0025 for bits to be flipped. The work presented in [142] recommended tighter values to limit the velocity (± 4).

Unlike the continuous PSO, only limited research efforts have attempted to modify the standard BPSO to enhance its performance. As discussed earlier, the velocity in BPSO should be limited to ± 6 or ± 4. In [143], an essential binary particle swarm optimization (EPSO) is proposed based on the idea of omitting the velocity component of PSO. Thus, there is no need to limit the velocity. The EPSO adopted the concept of queen informants in ACO and applied it in PSO resulting in a modified form of EPSO denoted as EPSOq. In EPSOq, a new informer named *the queen informer* is added where it is updated after each loop by $Gbest$ only and its role is to provide information to other particles. The EPSO and EPSOq were applied to solve two suites of test functions and EPSOq showed better performance in terms of convergence rate and solution quality as compared to the standard BPSO and EPSO. However, the results of EPSOq are not optimal.

To overcome the problem of nonlinearity that results from the sigmoid function and the problem of the unusual behavior of the probability function of a bit-change, an improved binary particle swarm optimization (IBPSO) is proposed in [144]. In IBPSO, the XOR and AND operators are used in the velocity update equation $v_{id}(t+1)$, and the updated new position $x_{id}(t)$ depends on the current position. Utilizing the genotype-phenotype concept, a modified binary particle swarm optimization is introduced in [145]. In this approach, the standard BPSO is modified by letting the velocity and the position act as a particle and a solution. The position in the velocity update equation is a phenotype and the updated position equation is a genotype that depends on the current phenotype's position. This modified binary version is evaluated on ten benchmarking functions and the results have demonstrated that its performance is better than the standard BPSO. A novel binary particle swarm optimization (PBPSO) was proposed in [146] to address the problem of the long time spent by the sigmoid function. In [147], an adaptive mutation operator was added to the PBPSO resulting in a new binary variant called adaptive mutation PBPSO (AMPBPSO). In AMPBPSO, the new binary position update is based on an adaptive mutation probability which is evaluated by measuring the distance between the new binary position and its best position. The introduced adaptive mutation operator helps to maintain diversity and enhance local search.

The BPSO finds some difficulties to converge to the best solution because the binary positions are based on randomness. In addition, BPSO suffers from becoming trapped in local minima [141] [148]. A V-shaped transfer function is used in [149] instead of using the S-shaped transfer function to avoid unhealthy randomness. In [149], the new binary position depends on the V-shaped transfer function and it has three transition states: stays in its current position, changes its value to 1, or changes its value to 0. In this case, the randomness of binary positions is reduced. Though this method is capable of reducing the randomness of binary positions, it is incapable to solve the problem of convergence to local minima. To avoid this later problem, an enhancement to the work done in [149] was proposed in [150] by adding a mutation operator. The overall framework in [150] consists of a V-shaped transfer function, a new updating position formula, and a mutation operator. This combination enhances the convergence rate and diversity of particles and it also helps to escape from the local minima problem.

In [151], six new S-shaped and V-shaped transfer functions were introduced and tested on twenty-five benchmark functions. Based on the results, the V-shaped family of transfer functions outperforms the S-shaped family in terms of the convergence rate and escaping from local minima. Therefore, it is recommended to use the V-shape family, particularly the V4 transfer function, to enhance the standard BPSO performance.

In [152], the velocity update equation is modified to have three different equations for the three different cases: when $Gbest = Pbest_i = 1$, $Gbest = Pbest_i = 0$, and $Gbest \neq Pbest_i$. The velocity increases if $Gbest = Pbest_i = 1$, decreases if $Gbest = Pbest_i = 0$, and remains unchanged when $Gbest \neq Pbest_i$. This is justifiable by the consensus among $Gbest$ and $Pbest_i$ in the first two cases whereas the third case lacks this consensus. The proposed algorithm showed superior performance compared with other BPSO variants. A hierarchical BPSO (BPSOHS) inspired by multi-level learning behavior was proposed in [153]. Particles in the proposed approach are split into two groups: leaders and followers. In BPSOHS, the leaders' velocity and position updates are the same as the standard BPSO while the followers' velocities and positions are updated based on a random walk probability and a decision from the leaders. The idea of this method is to enable followers to fly towards leaders and at the same time to explore an extensive region near the leader space. Moreover, a mutation technique is implemented in order to avoid premature convergence.

Utilizing the sigmoid transfer function, the author in [154] proposed a new binary PSO version where the PSO acceleration coefficients are modified based on the fitness of each particle. The effectiveness of the proposed approach was tested on four problems in the continuous search space and its performance in optimizing binary problems is not validated. In addition, the number of independent runs is only 10 which is not enough to produce accurate results. At least 30 independent runs are needed to validate the performance of the proposed binary variant. Recently, the work in [155] converted the gaining-sharing knowledge-based continuous algorithm [156] into a novel binary PSO variant where both algorithms are based on the idea of gaining–sharing knowledge that humans experience during their lifespan. The new binary variant is tested on twenty two feature selection benchmark datasets and its performance is compared with the standard binary PSO and other well-known binary optimization algorithms such as binary GWO and binary salp swarm algorithm. The proposed approach was only tested on feature selection problems while its performance of multi-dimensional knapsack problems is not investigated. Thus, it would be interesting to study the performance of this binary variant when it solves multi-dimensional knapsack problems.

The work in [157] hybridized the binary PSO with the sine cosine algorithm to solve feature selection problems. A V-shaped transfer function is used and the performance of the hybrid variant is compared with some well-known binary PSO algorithms including the standard one as well as with other high-performance binary algorithms such as binary whale optimization algorithm and binary moth flame optimization algorithm. Although the hybrid approach has shown good performance, all results are obtained for 10 independent runs only which is not enough to achieve high accuracy. The authors in [158] developed a hybrid approach that combines binary PSO with tabu search to solve the set-union knapsack problem. The performance of this hybrid approach on feature selection problems is not investigated yet. Although [157] and [158] have achieved remarkable performance, this achievement comes at the expense of complexity.

A new binary PSO variant that is designed to solve feature selection problems is proposed in [159]. The idea of the proposed algorithm is to divide the entire population into sub-swarms where each sub-swarm implements a unique inertia weight strategy. Although the proposed approach has shown better classification performance compared with the binary PSO, GA, and binary GSA, it requires more computational time than the standard binary PSO. In [160], a time-varying mirrored transfer function is proposed and its performance is evaluated on CEC 2005 benchmark functions as well as on 0-1 multidimensional knapsack problems. Results have shown that the proposed transfer function outperforms the S-Shaped and V-shaped transfer functions. The performance of this new transfer function when used by other-metaheuristic algorithms is not studied yet. Thus, more research work is needed to further validate the effectiveness of this mirrored transfer function. In addition, its performance on feature selection problems needs to be investigated.

A. BPSO TRANSFER FUNCTIONS

The role of a transfer function is to map the velocity of a certain dimension of a particle into the probability of bit flipping. According to [161], three rules must be followed when selecting a transfer function:

- The probability of changing a bit from 0 to 1 or vice versa must be high for large absolute values of velocities.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3142859, IEEE Access

- The probability of unchanging a bit must be high for small absolute values of velocities.
- The outcome of a transfer function should be in the range of [0,1] as it acts as a probabilistic function.

Some transfer functions have been proposed in the literature such as the sigmoid function, the S-shaped family, and the V-shaped family [151]. Table IV lists the most common transfer functions that can efficiently convert a continuous search space into a binary one. The performance of binary algorithms is highly dependent on the selection of the transfer function. Thus, it is crucial to investigate the performance of new binary variants when different transfer functions are used to figure out which transfer function is the most suitable for each variant. Some of the transfer functions listed in Table IV were originally proposed for binary meta-heuristic algorithms and not binary PSO. However, they can be implemented in binary PSO and their performance on binary PSO is to be studied.

TABLE IV
A SUMMARY OF WELL-KNOWN TRANSFER FUNCTIONS

Category	Formula	Remarks	Ref.
S-shaped	$T_1(x) = \frac{1}{1+e^{-ax}}$	$a = 2, 1, \frac{1}{2}$, or $\frac{1}{3}$	Mirjalili and Lewis [151]
	$T_2(x,\varphi) = \frac{1}{1+e^{\frac{-ax}{\varphi}}}$	$\varphi = \varphi_{max} - t(\frac{\varphi_{max}-\varphi_{min}}{T})$ t= current iteration T= maximum number of iterations a = the function's slope	Islam et al. [162]
V-shaped	$T_3(x) = \lvert tanh(x) \rvert$		Mirjalili and Lewis [151]
	$T_4(x) = \left\lvert erf\left(\frac{\sqrt{\pi}}{2}x\right)\right\rvert$		Mirjalili and Lewis [151]
	$T_5(x) = \left\lvert \frac{x}{\sqrt{1+x^2}} \right\rvert$		Mirjalili and Lewis [151]
	$T_6(x) = \left\lvert \frac{2}{\pi} arctan\left(\frac{\pi}{2}x\right)\right\rvert$		Mirjalili and Lewis [151]
X-shaped	$T_7(x) = 0.5 + \frac{-x}{1+0.5\lvert -x \rvert}$		Beheshti [163]
	$T_8(x) = 0.5 + \frac{x-1}{1+0.5\lvert x-1 \rvert}$		Beheshti [163]
U-Shaped	$T_9(x) = \rho\lvert x^{\alpha}\rvert$	ρ determines the slope of the function α determines the width of the function	Mirjalili et al. [164]
Linear	$T_{10}(x) = \frac{x-R_{min}}{R_{max}-R_{min}}$	$[R_{max}, R_{min}]$ is a predefined range	Wang et al. [165]

V. VALIDATION OF NEW PSO VARIANTS

This part focuses on the steps that are required to validate the effectiveness of new PSO variants. These steps can be summarized as follows:

1. Development of a novel approach based on new ideas, parameter modifications, or hybridizations

The first step when developing a novel PSO variant is introducing new ideas particularly concepts that can help to balance exploration and exploitation. The most common concepts that help improve the performance of PSO are modification of controlling parameters particularly the inertia weight, hybridizing PSO with other prominent meta-heuristic algorithms, and multi-swarm approaches.

2. Testing the novel PSO variant on a wide range of benchmarking functions

The next step is to validate the performance of the new PSO variant to solve several unimodal, multimodal, and composite benchmarking functions. The most common classical benchmarking functions consist of twenty three unimodal and multimodal functions that are widely used by researchers [124, 151, 166, 167]. Although these functions can validate the exploration and exploitation abilities of a certain PSO variant, these functions do not fully represent real-world optimization problems since they are unconstrained problems. To represent real-world problems that contain a number of constraints, the CEC2017 test suite is introduced. Therefore, a strong PSO variant should be able to provide significant improvements when dealing with the CEC2017. Other widely used benchmarking functions suites are CEC2005 and CEC2019.

3. Testing the new PSO variant on real-world engineering problems

This is a crucial step to demonstrate the effectiveness of a proposed PSO variant. Real-world optimization problems are challenging since they have a number of constraints that must be satisfied. The introduction of constraints divides particles into valid and invalid particles. A valid particle is a one that can meet all constraints whereas a particle is considered invalid if it violates one or more constraints. One of the most common ways to penalize a particle when it does not satisfy all constraints is to assign its fitness a large value such as 10^{12} when solving a minimization problem. The most widely used engineering problems that serve as benchmarks to test the performance of a new optimization algorithm are welded beam design, speed reducer design, pressure vessel design, and tension/compression spring design.

4. Comparison with well-known PSO variants and other meta-heuristic approaches

The fourth step is to compare the performance of the developed PSO variant with other prominent PSO variants. However, this is not enough as the performance of the new

PSO variant must be compared with other outstanding meta-heuristic algorithms since their performance might be better on a certain set of functions compared with existing PSO variants.

5. High dimensional performance

A PSO variant might show strong performance when it deals with low dimension problems; nevertheless, it may have a poor performance when it solves high dimensional problems. As a consequence, it is crucial to validate the effectiveness of the new variant when it solves both low and high-dimensional problems. The performance of a PSO variant usually degrades as the number of dimensions increases; therefore, it is essential to investigate the performance of a new PSO variant on high-dimensional problems.

6. Sensitivity analysis

PSO controlling parameters have a direct influence on optimization performance. Some new PSO variants may add new parameters besides the three controlling parameters of the original PSO. Thus, it is crucial to provide a sensitivity analysis that illustrates the influence of these parameters on the performance of the new variant. In addition, it is important to show which parameters are sensitive to different settings and also show which parameters are robust.

7. Convergence analysis

Although the average fitness and standard deviation are two important metrics that help to validate the effectiveness of an optimization algorithm, a convergence analysis is required to further demonstrate the ability of an optimization algorithm to escape from local optima and converge to a global one.

8. Statistical significance analysis

Statistical significance analysis is an essential step that needs to be performed to show that a new PSO variant is statistically more significant than other existing PSO variants or meta-heuristic algorithms. In the literature, there have been a significant number of non-parametric statistical tests that help to demonstrate the superiority of one algorithm over others. Wilcoxon rank-sum test and Friedman test are the most two common statistical tests that are used to evaluate the performance of meta-heuristic algorithms.

VI. APPLICATIONS OF PSO

Due to its simplicity and robustness, PSO has been widely used as an efficient optimization tool for solving various optimization problems in many real-world applications such as feature selection, wireless communications, image processing and electrical power systems. The following present the applications of PSO in the aforementioned fields.

A. APPLICATIONS OF PSO TO FEATURE SELECTION

This part focuses on the applications of PSO on feature selection problems. It starts with an introduction to feature selection followed by a detailed explanation of how PSO is applied to solve feature selection problems. Finally, PSO-based feature selection studies are reviewed.

1) FEATURE SELECTION

Feature selection is a selection process that aims to select y features from x original features $(y < x)$ to optimize a certain metric [168], [169], [170]. Feature selection is a crucial process in machine learning and data mining as it can significantly help to remove unnecessary and redundant features [171]. For a large number of features, finding the optimal number of features is a complicated problem [172]. Generally, the selection of features is used for four reasons: simplifying data, reducing computational time, avoiding the dimensionality curse, and reducing overfitting. Figure 1 illustrates the feature selection process which goes through five steps: initialization, generation, evaluation, stopping criteria, and validation. In the initialization step, the number of all original features represents the dimensionality of the search space. The second step is responsible to select the best subset of features. Various searching approaches such as conventional schemes and meta-heuristic algorithms can be utilized to perform this task. Typically, searching can start with no features, all features, or a random selection of a subset of features [173], [174], [175]. Selected subsets in the second step is evaluated in step three to check their goodness. The fourth step requires good stopping criteria that terminate when good performance is achieved. The final step validates the effectiveness of the obtained subset of features on a test set.

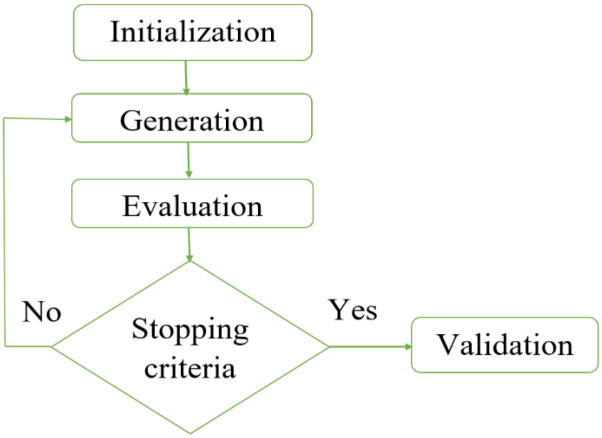

FIGURE 1. **The general steps of feature selection**

Figure 2 shows the key factors of feature selection which include searching algorithm, number of objectives, and evaluations measures. The first key factor of feature selection is the searching algorithm that attempts to find the best subsets of features. Feature selection is an NP-hard problem particularly for large datasets as it has 2^n possible solutions where n denotes the number of original features. Thus, searching algorithms play an important role in solving feature selection problems since they can achieve remarkable performance with a significant reduction in computational time. The number of objectives represents the second key factor where a single objective such as minimizing the classification error rate is considered or multiple objectives such as minimizing the number of features and minimizing the classification error rate are taken into account. Evaluation measures as the third key factor use an evaluation function that can determine the strength and the drawbacks of the selected subset which in turn help to guide the searching algorithm.

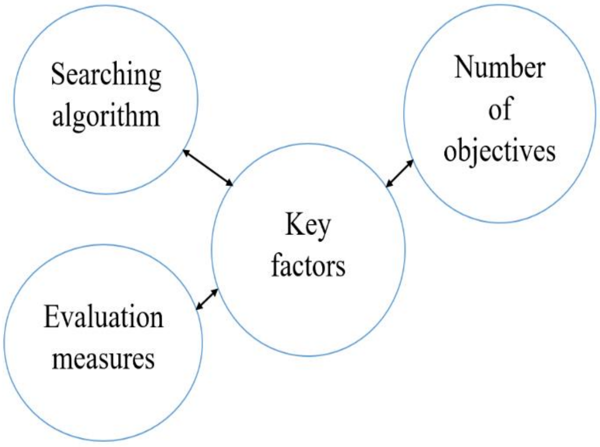

FIGURE 2. **The three key factors of feature selection**

Feature selection approaches can be classified into two main categories: filter and wrapper methods [176], [177], [169]. The main difference between the two is that wrapper approaches implement a classification algorithm to evaluate the goodness of the selected features whereas filter methods do not. As a consequence, wrapper approaches achieve better performance [174], [176], [178], [179]. Some research work [174], [178], [179] adds the embedded approach as a third category of feature selection approaches. In the embedded approach, the classifier and the selected features are integrated. Table V shows the strengths and drawbacks of the filter, wrapper, and embedded approaches.

TABLE V
A COMPARISON BETWEEN FEATURE SELECTION APPROACHES

Filter approaches	Wrapper approaches	Embedded approaches
Generalizable	Require an objective function	Combine classification with feature selection
Computationally cheap	Computationally expensive	Computationally expensive
Algorithm independent	Require a classifier or a learning algorithm	Benefit from the advantages of both filter and wrapper methods

2) PSO FEATURE SELECTION MECHANISM

Feature selection is a binary optimization problem by nature. To represent a solution that has the potential to solve the feature selection problem, a vector with n features dimension is needed where each element of a vector can have a value of either 0 or 1. A value of 0 indicates that a feature is not selected while 1 indicates the selection of the feature [3]. PSO can form a binary vector that can be used to solve the feature selection problem. Optimization problems where their variables are continuous values can be turned into a binary optimization problem by replacing the continuous variables with binary variables. PSO in its continuous version allows candidate solutions to update their positions where each variable can have a continuous value. In binary optimization, a position is updated by converting its value from 0 to 1 or from 1 to 0. As a result, for PSO to be able to solve feature selection problems, a transfer function is applied to convert the real positions of candidate solutions into binary ones [8]. Transfer functions rely on a probabilistic approach to update binary values from 1 to 0 or from 0 to 1. Several transfer functions have been proposed in the literature where the S-Shaped and V-Shaped transfer functions are the most common ones [21].

3) FEATURE SELECTION STUDIES BASED ON PSO

PSO has gained significant consideration in the domain of feature selection to solve different kinds of problems. For example, the study in [180] proposed an improve BPSO based on Lévy flight as a local search component and inertia weight coefficient as a global search component as well as mutation mechanism for population diversity enhancement. The KNN classifier for the classification process and the Sigmoid function are implemented for solution mappings. Sixteen classical datasets were used for validation. The findings showed promising performance compared to other benchmarking methods.

In another study [181], the authors used the BPSO to address the feature selection problem on input variables for intelligence joint moment prediction. Experimental data gathered from ten electromyography (EMG) data and six joints' angles were used for validation. ANN classifier is used for the classification process and the Sigmoid function is implemented for solution mappings. Findings showed that the proposed approach is able to reduce the number of input variables of five joint moments from 16 to less than 11.

In [159], the authors proposed a co-evolution binary particle swarm optimization with a multiple inertia weight strategy. The KNN classifier for the classification process and the Sigmoid transfer function were used to convert the search space into a binary one. Ten benchmark datasets collected from the UCI repository were used for validation and the proposed method was compared against four well-known feature selection methods. Findings demonstrated a competitive performance compared to other methods.

In [182], the BPSO was hybridized with differential evolution to solve feature selection issues in EMG signals classification. The EMG signals of ten healthy subjects obtained from a publicly accessible EMG database were used for validation. Discrete wavelet transform was applied to decompose signals into wavelet coefficients. The sigmoid transfer function was used to meet the nature of feature selection and the KNN classifier for the classification process. The performance of the proposed method was compared against four benchmarking feature selection methods. Findings demonstrated that the proposed method is beneficial for EMG signals classification. In the same domain of EMG signals classification, the work in [183] proposed a new personal best guide BPSO. The discrete wavelet transform decomposes a signal into multiresolution coefficients. The sigmoid transfer function was used to meet the nature of feature selection and the KNN classifier for the classification process.

Moreover, a study based on improved BPSO was proposed in [184] to address the feature selection problems in gene selection and cancer classification. This approach chooses a small dimensional set of prognostic genes to classify biological samples of binary and multi-class cancers using Naive–Bayes classifier and Sigmoid transfer function were used to meet the nature of feature selection. Eleven microarray datasets of different cancer types were used for validation. Experimental results were benchmarked with seven other well-known methods and findings demonstrated a better result of the proposed method in terms of classification accuracy and the number of selected genes.

A hybrid improved PSO with a shuffled frog leaping algorithm was proposed in [185] to address the feature selection problem. Naive Bayes (NB), KNN, and Support Vector Machine (SVM) classifiers were used for classification and the Sigmoid function was applied. For validation, a dataset that consists of 1600 reviews of the 20 most popular Chicago hotels were used. The findings revealed that the proposed method attains an optimized feature subset and achieves higher classification accuracy. In another research [186], the authors proposed a new multiswarm heterogeneous BPSO using a Win-Win method to solve feature selection problems in liver and kidney disease diagnosis.

In [187], Hamming distance is introduced as a proximity measure that can update the binary PSO velocity and select the important feature subsets. Experimental results on three benchmark datasets are evaluated using classification accuracies and validity indices as well. Utilizing rough set theory and its distinction table as a binary table, the work in [188] proposed a hybrid binary PSO variant that implements a statistical elimination strategy that can help to reduce the number of features efficiently. The authors in [189] developed a new multi-swarm PSO variant as a feature optimization technique in facial recognition systems. Results have demonstrated that the new PSO variant can significantly outperform the standard PSO and GA. To achieve better accuracy, another study conducted by [190] proposed a fuzzy rule-based binary PSO (FRBPSO) that is designed specifically to solve feature selection problems. Results on benchmarking high dimensional microarray datasets show the merits of the proposed FRBPSO method.

Based on feature sub-set correlation, the authors in [191] proposed a hybrid PSO with a new local search strategy for feature selection. In the proposed approach, PSO is designed to select features that have low correlation. Results have shown that the proposed PSO achieves higher accuracy compared with filter methods. Utilizing the SVM classifier, PSO is hybridized with GA in [192] as a wrapper feature selection tool to classify microarray data. Considering unreliable data in feature selection and based on bare-bones PSO, a multi-objective PSO approach is proposed in [193]. The work in [194] hybridized PSO with GA to improve feature selection in Digital Mammogram datasets. Utilizing the KNN classifier, Al-Tashi et al [195] proposed a hybrid PSO with GWO for wrapper feature selection. This work has used the sigmoid transfer function for converting the search space into a binary one.

Multi-objective variants of PSO have been widely applied to solve feature selection problems. For example, the authors in [196] developed a PSO-based multi-objective approach where features are ranked based on their frequency in the set of archives. The proposed multi-objective scheme is compared with three multi-objective PSO variants as well a multi-objective GA on nine benchmark datasets. Results have shown that the proposed approach is more efficient in reducing the number of features in large datasets while it achieves a satisfactory performance that is close to the performance achieved by other algorithms when it deals with datasets that have lower than 100 attributes. Nonetheless, the proposed approach suffers from slow convergence that restricts reaching the optimum Pareto front. Another work that utilizes multi-objective PSO is presented in [197] where a two-step algorithm is proposed for fault diagnosis of power transformers. The first step is responsible to select the most important features where the second step generates an ensemble classifier that is formed from the most accurate classifiers. The work in [198] has developed a multi-objective PSO feature selection approach to predict the dose of warfarin. The authors in this work have applied artificial neural networks as a technique to assess the selected features. The developed multi-objective PSO approach is compared with NSGA-II and results have demonstrated that PSO outperforms NSGA-II in terms of accuracy and the minimum number of features selected.

An improved multi-objective version of PSO is developed in [199] to study multi-label feature selection. The authors implemented an adaptive uniform mutation operator to enhance the exploration abilities while a local learning strategy is used to achieve better exploitation. Results have shown that the proposed scheme performs better than NSGA-II in terms of exploration. Based on the filter approach, a multi-objective BPSO is proposed in [200] for feature selection to obtain a non-dominated feature subset that results in a reduction in the number of selected features as well as higher classification accuracy. The work presented in [201] developed two multi-objective algorithms based on PSO (NSPSOFS and CMDPSOFS) for solving feature selection problems. The NSPSOFS algorithm is developed based on the concept of nondominated sorting in NSGAII to check the possibility of implementing a simple multi-objective PSO to solve the problems of feature selection. The second algorithm known as CMDPSOFS utilizes three different techniques:

mutation, dominance, and crowding. Testing the two algorithms on twelve classical UCI repository datasets, results have demonstrated the superiority of these two algorithms in reducing the number of features and decreasing the classification error rate when compared with NSGAII and the strength Pareto evolutionary algorithm 2 (SPEA2) [202] and Pareto archived evolutionary strategy (PAES) [203]. Table VI summarizes the existing studies on feature selection using PSO and its variants.

TABLE VI
A SUMMARY OF FEATURE SELECTION APPROACHES USING PSO AND ITS VARIANTS

Ref.	Algorithm	Domain	Feature selection type	Classifier
Ji et al. [180]	BPSO + Levy flight	16 Benchmark dataset from UCI	Wrapper	KNN
Xiong et al. [181]	BPSO	10 Electromyography (EMG) data and 6 joints' angles	Wrapper	ANN
Too et al. [159]	Co-evolution BPSO with a multiple inertia weight strategy.	10 Benchmark dataset from UCI	Wrapper	KNN
Too et al. [182]	Hybrid BPSO + DE	10 EMG datasets	Wrapper	KNN
Too et al. [183]	Personal best guide BPSO	EMG datasets	Wrapper	KNN
Jain et al. [184]	Improved BPSO	11 Microarray datasets of different cancer types	Wrapper	Naïve Bayes (NB)
Rajamohana and Umamaheswari [185]	Hybrid improved PSO + shuffled frog leaping algorithm	Dataset consists of 1600 reviews of the 20 most popular Chicago hotels	Wrapper	NB, KNN and Support Vector Machine (SVM)
Gunasundari et al. [186]	Multiswarm heterogeneous BPSO using a Win-Win method	Liver and kidney disease diagnosis.	Wrapper	Probabilistic Neural Networks (PNN)/ SVM
Banka and Dara [187]	Hamming distance + BPSO	Three benchmark datasets	Wrapper	KNN
Dara et al. [188]	Hybrid BPSO	Gene expression data	Filter	Rough set theory
Mistry et al. [189]	Multi-swarm PSO variant	Facial recognition systems	Wrapper	ANN, SVM and ensemble classifiers
Agarwal et al. [190]	Fuzzy rule based BPSO (FRBPSO)	High dimensional microarray datasets	-	-
Moradi and Gholampour [191]	Hybrid PSO with a new local search strategy	13 benchmark datasets	Wrapper	KNN
Talbi et al.[192]	Hybrid PSO + GA	Microarray datasets	Wrapper	SVM
Yong et al. [193]	Multi-objective PSO	Unreliable data	Wrapper	1-nearest neighbor method
Jona and Nagaveni [194]	Hybridized PSO + GA	Digital Mammogram datasets	Wrapper	KNN
Al-Tashi et al. [195]	Hybridized PSO + GWO	18 Benchmark dataset from UCI	Wrapper	KNN
Amoozegar and Minaei-Bidgoli [196]	Multi-objective PSO	9 Benchmark dataset from UCI	Wrapper	KNN
Peimankar et al. [197]	Multi-objective PSO	Fault diagnosis of power transformers	Wrapper	Ensemble classifier
Sohrabi and Tajik [198]	Multi-objective PSO	Dose of warfarin prediction	Wrapper	ANN
Zhang et al. [199]	Improved multi-objective PSO	6 Datasets from various applications, image processing, bioinformatics, music emotion.	Wrapper	KNN
Xue et al. [200]	Multi-objective BPSO	8 Benchmark dataset from UCI	Filter	-
Xue et al. [201]	NSPSOFS and CMDPSOFS	12 Benchmark dataset from UCI	Wrapper	KNN

B. WIRELESS COMMUNICATIONS

PSO has been extensively applied to solve diverse wireless communications optimization problems in different areas such as wireless sensor networks (WSNs) [204-207], cognitive radio networks (CRNs) [208-212], intelligent reflecting surfaces (IRSs) systems [213], edge computing [214-216], ad-hoc networks [217-219], signal processing [220-222] and antenna design [223-225]. PSO is used to tackle many WSNs challenges such as node localization, optimal deployment and clustering. A comprehensive survey on the applications of PSO in WSNs is presented in [226]. Since spectrum sensing in CRNs is a non-convex optimization problem, CRNs utilize PSO algorithms to optimize their performance in terms of energy efficiency, spectral efficiency and sensing time. The work in [227] has implemented the standard PSO algorithm to detect the presence of primary users. According to the simulation results, PSO can save more than 80x of energy consumption as well as sensing time. The performance of the proposed scheme can be further improved by applying enhanced PSO variants. A hybrid PSO-GSA approach is used in [123] to optimize energy efficiency in 5G CRNs. Results have shown that the proposed hybrid approach is more energy efficient than the standard PSO algorithm, Artificial Bee Colony (ABC), the energy detector scheme and the well-known cooperative spectrum sensing method.

In [213], PSO is used for beamforming optimization in IRSs to minimize the transmission power given that the signal-to-noise ratio (SNR) does not go below a certain threshold. Results have demonstrated that PSO can achieve near-optimal beamforming solutions. Considering vehicular ad-hoc networks, a task-distribution PSO is proposed to efficiently distribute tasks among vehicles that belong to the same cluster [228]. The results shows that the proposed PSO scheme outperforms GA in terms of overhead reduction while its overhead performance is comparable with linear programing. In [217], an adaptive PSO is developed to solve the clustering problem in ad-hoc networks.

One of the interesting and recent applications of PSO in the wireless domain is edge computing where intensive computational tasks are offloaded from core networks to the edge that is closer to the user. Considering a smart internet of things (IoT) system, a self-adaptive PSO algorithm that utilizes the GA operators (SPSO-GA) is recently proposed [214] to develop an energy-efficient approach that can efficiently make offloading decisions for deep neural networks (DNNs) layers with layer partition operations. Simulation results have shown that the SPSO-GA algorithm outperforms the GA and PSO-GA approaches in terms of energy consumption. The authors in [215] applied a PSO-GA algorithm to minimize the system cost when DNN layers are offloaded over the cloud, edge and user’s devices. According to the results, the proposed PSO-GA can significantly reduce the system cost compared with PSO and GA. Combining edge computing and cloud computing, the work in [216] implemented the BPSO algorithm with the GA operators to minimize data transmission time when the workflow is executed. Although the results have shown the superiority of the proposed approach in reducing data transmission time, data transmission energy is not considered. The works in [214-216] can be further improved by applying the recent high-performance PSO variants presented in Section III.

Considering a finite impulse response (FIR) filter, the authors in [229] applied a quantum-behaved PSO algorithm to develop an adaptive channel equalizer. Based on the results, the proposed approach achieves a lower bit error rate compared with GA, SPSO, and the classical least mean square method. The SPSO algorithm and its variants are also applied to design infinite impulse response (IIR) filters [220-222]. Another application of PSO in the wireless communications field is antenna array design. In a massive multiple-input multiple-output (MIMO) network, a contraction adaptive PSO algorithm is proposed in [223] to find the optimal positions of antenna array elements that can optimize an antenna’s performance when it transmits or receives data. Although the proposed approach has shown good results, its performance is compared with PSO variants only. It is evident from the state-of-the-art presented in this subsection that most of the work has considered single-objective optimization. In wireless communications, it is crucial to consider multiple objectives such as energy efficiency, spectral efficiency, and latency to develop a robust and reliable communication system. Therefore, it is essential to develop novel multi-objective PSO algorithms that takes several wireless metrics into account.

C. IMAGE PROCESSING

PSO has been successfully applied to solve many image processing optimization problems in diverse areas such as image segmentation [230-232], image enhancement [233], image compression [234] and image watermarking [235].

One of the recent and interesting applications of PSO in image processing is multilevel thresholding image segmentation. The work in [230] modified the standard PSO algorithm to perform image threshold segmentation in lung CT images where the aim is to identify lung tissue. In the proposed scheme, the symmetric disposition is implemented to adjust the positions of particles in each iteration. Although this work has shown fast segmentation speed as well as good segmentation accuracy, its performance is tested only on one lung CT image. In [231], PSO is used to segment medical images to detect brain tumors. This work can be further improved by applying recent robust PSO variants or other well-known meta-heuristic algorithms such as GWO and Equilibrium Optimizer (EO). Considering multilevel image thresholding, the work in [232] hybridized PSO with the firefly algorithm (FA) [236] to search for the optimal threshold values. Based on the results, the proposed hybrid scheme outperforms GA, PSO, and FA in terms of peak-signal-to-noise-ratio (PSNR).

In the area of image enhancement, PSO is utilized in [233] to address the inaccurate nature of retinal images. The effectiveness of the proposed approach is validated on two well-known image datasets and results have shown that PSO can significantly enhance the quality of retinal images compared with GA, ACO, and ABC. The authors in [234] have proposed to use PSO with Haar Wavelet Transform to compress medial images. The simulation results have shown that the proposed scheme can achieve high PSNR which indicates that the quality of compressed images is close the quality of original images. In [235], PSO is used with an intertwining logistic map to develop a blind watermarking scheme. Testing the proposed method on eight classical grayscale images, results have shown that PSO can efficiently optimize watermark embedding strength.

D. ELECTRICAL POWER SYSTEMS

PSO has been widely applied to optimize the performance of electrical power systems including economic dispatch [237-239], optimal power flow [240-242], state estimation [243], power system controllers [244, 245], unit commitment [246] and capacitor placement [247]. A detailed and thorough survey on the applications of PSO in electrical power systems is provided in [248]. The survey focused on ten areas including optimal power flow, economic dispatch, reactive power dispatch, and maintenance scheduling. Recently, the works in [249, 250] have provided a comprehensive review of the applications of PSO and its variants on the economic dispatch problem. The authors in [246] have recently reviewed the state-of-the-art applications of PSO to solve the unit commitment problems. According to [246], most of the PSO-based unit commitment research has considered single-objective optimization that minimizes cost only while multi-objective optimization that jointly minimizes cost and emission is not well studied yet.

The work in [247] applied a hybrid ABC-PSO algorithm in an IEEE 34-node and 69-node radial distribution networks to find the best capacitor placement and size that can help to reduce power loss. The proposed technique can achieve lower power loss when compared with the individual PSO and ABC algorithms; nevertheless, its performance is not compared with high-performance PSO variants as well as with other well-known meta-heuristic algorithms such as GWO and whale optimization algorithm (WOA). The work in [243] hybridized PSO with gravitational search algorithm to solve the problem of state estimation in distribution systems. Results have shown that the proposed hybrid scheme is more accurate and reliable than the standard PSO algorithm and the original GSA approach.

Table VII summarizes the major applications of PSO, shows the type of PSO used and the major publications for each application. It is observed from Table VII that each application is associated with a different PSO variant which indicates that there is no dominant PSO variant that can solve most of the optimization problems. In other words, each PSO variant is specialized in solving specific problems.

TABLE VII
A SUMMARY OF PSO APPLICATIONS IN VARIOUS ENGINEERING FIELDS

Sub Areas and Details	The type of PSO variants used and References
Wireless communication networks	
Wireless sensors networks	Improved PSO [204], immune cooperative PSO [205], extended binary PSO [206], chaotic PSO [207]
Cognitive radio networks	Binary PSO [208], adaptive discrete PSO [209] , binary quantum elite PSO [210], improved PSO [211], two-phase PSO [212]
Ad-Hoc networks	Adaptive PSO [217], multi-objective PSO [218], [219]
FIR digital filter design	Quantum-behaved PSO [229], adaptive PSO [251], genetic PSO with dynamically varying neighborhood technique [252]
IIR digital filter design	Hybrid Cellular PSO [220], single-swarm PSO [221], adaptive PSO [222]
Estimation of non-stationary signals	Self-adaptive PSO with adaptive filtering technique [253]
Antenna array design	Adaptive PSO [223], PSO with wavelet mutation [224], improved boolean PSO [225], hybrid enhanced PSO, and DE [116]
Design of electromagnetic devices	Binary PSO [262], modified quantum PSO [263]
Image processing	
Image enhancement	Exponential center symmetric inertia weight based PSO [233], Multi-objective PSO [254]
Image restoration	PSO with blind deconvolution technique [255]
Data clustering	Modified PSO [256], Hybridized PSO with mutation operator [257], adaptive PSO with fuzzy logic [258], PSO-GA [259]
Image segmentation	Enhanced PSO [260], improved PSO [230]
Image compression	PSO with Haar wavelet transform [234]
Image watermarking	Multi-dimensional PSO [235], guided dynamic PSO [261]
Electrical Power Systems	
Economic dispatch	Improved quantum PSO [237], evolutionary PSO [238], improved random drift PSO [239], hybrid PSO and gravitational search algorithm [130], improved PSO with a biogeography learning strategy [264], cultural quantum-behaved PSO

	[265], modified PSO with time varying acceleration coefficients (MPSO-TVAC) [134], hybrid double-weighted PSO [266]
Optimal Power flow	Modified PSO [240], hybrid PSO-dragonfly algorithm [241], hybrid PSO-artificial physics optimization [242], hybrid PSO-pattern search algorithm [267], enhanced PSO with stochastic weight and chaotic mutation [268].
State Estimation	Adaptive PSO [269], exponential decay PSO (EDPSO) [270], PSO with chaos theory [271], hybrid PSO and gravitational search algorithm (PSO-CGSA) [243]
Power System controllers	Hybrid PSO-pattern search [244], PSO with Pade approximation [245]
Capacitor Placement	Constriction factor PSO [272], hybrid PSO [247]
Generation expansion problem	Comprehensive learning PSO [273], fuzzy adaptive chaotic binary PSO [274]
Power system reliability and security	Modified PSO [275], binary PSO [276]
Reactive power and voltage control	adaptive discrete binary PSO [277], standard PSO [278], PSO with artificial physics optimization [279], hybrid PSO [280]
Short-Term load forecasting	switching delayed PSO [281], hybrid PSO [282]

VII. PSO DRAWBACKS

Despite the excellent performance of its variants, PSO suffers, in general, from some weaknesses that can be alleviated by introducing new modifications to the current PSO variants. The literature reports several concerns about PSO performance which can be outlined as follows:

A. PREMATURE CONVERGENCE

One of the major performance problems of PSO is premature convergence as pointed out in [40, 283]. This problem occurs due to the lack of population diversity especially in complex multimodal functions [40]. The work in [37, 71, 284] presented important PSO variants that have shown remarkable performance in terms of avoiding premature convergence. Nevertheless, much more research is needed to address this problem.

B. THE DIFFICULTY OF CONTROLING THE PSO PARAMETERS

Although there are only three parameters (c_1, c_2, w) to be controlled in PSO, it is difficult to control these parameters and find their appropriate setting at each iteration. Despite the extensive efforts of proposing several methods to control c_1, c_2 and w none of these methods guarantee that the optimal setting of c_1, c_2 and w can be achieved.

C. IMPROPER VELOCITY ADJUSTMENT

The improper velocity adjustment occurs when inappropriate values of c_1, c_2 and w are chosen. This makes the particles fly in undesired directions, causing stagnation around or near the optimum solution [285].

VIII. POTENTIAL RESEARCH DIRECTIONS

Although PSO variants have shown promising results in solving optimization problems, PSO can still be developed further to improve its performance when applied to solve complex real-world optimization problems. The following provides some potential future directions to be considered by researchers who are interested in PSO and its applications:

1) The original PSO and its recent variants presented in Section III can be hybridized with other recent high-performance metaheuristic algorithms such as Equilibrium optimizer (EO) [286], Marine Predators Algorithm (MPA) [287], Gradient-based optimizer (GBO) [288], Political Optimizer (PO) [289], The Arithmetic Optimization Algorithm (AOA) [290], and Archimedes optimization algorithm [291].

2) The recent PSO variants presented in Section III can be converted into binary PSO algorithms and utilized to solve binary problems such as feature selection and the 0-1 knapsack problem.

3) Some of the binary PSO variants presented in Section IV are applied to feature selection only while others are applied to solve the 0-1 knapsack problem. It would be interesting to apply each recent binary PSO variant to solve both problems and evaluate the performance.

4) Some of the binary transfer functions presented in Table IV have not been investigated and could be utilized to test the performance of PSO using such transfer functions.

5) The performance of PSO variants on high-dimensional problems is not well studied yet. The performance of recent PSO variants on high-dimensional problems can be investigated. In addition, Further work is needed to develop new PSO variants that can perform well on low and high-dimensional problems.

6) The development of new PSO variants that can solve multi-objective problems is a promising research direction to be considered.

7) Recent PSO variants can be applied to solve a wide range of real-world optimization problems such as data clustering [292], maintenance scheduling [293], lot-sizing optimization [294, 295], supply-chain network optimization [296, 297].

8) One promising research direction is to hybridize wrapper approaches that implements PSO variants with filter methods to solve feature selection problems.

IX. CONCLUSION

PSO is a simple, robust, and fast optimizer that can solve complex real-world optimization problems. To overcome the limitations of the standard PSO, extensive research efforts have been exerted to modify the original PSO algorithm into better variants by applying several methods including controlling the PSO parameters, hybridizing PSO with other searching algorithms, and using multi-swarm techniques. This work presents an overall review of the distinct research works that have been conducted on PSO. The review starts by explaining the basic concepts of PSO. Then, it describes the different topologies that can be used in PSO, provides a comprehensive review of the recent and historical prominent PSO variants. The review also includes PSO in binary presentation, remarkable engineering applications of PSO, and drawbacks of PSO. More specifically, this review paper has focused on PSO-based feature selection. Finally, this work provides some potential research directions that can help researchers further enhance the performance of PSO. In a nutshell, there are still rooms for improvement in PSO development to provide better performance when applied to complex high-dimensional real-world optimization problems.

REFERENCES

[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995, vol. 4, pp. 1942-1948: IEEE.

[2] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE computational intelligence magazine*, vol. 1, no. 4, pp. 28-39, 2006.

[3] J. Holland, "Adaptation in natural and artificial systems, univ. of mich. press," *Ann Arbor*, 1975.

[4] R. Storn and K. Price, "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces," *Journal of global optimization*, vol. 11, no. 4, pp. 341-359, 1997.

[5] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, 1995, vol. 4, pp. 1942-1948 vol.4.

[6] A. M. J. S. Eltamaly, "A novel strategy for optimal PSO control parameters determination for PV energy systems," vol. 13, no. 2, p. 1008, 2021.

[7] K. R. Harrison, A. P. Engelbrecht, B. M. J. S. Ombuki-Berman, and e. computation, "Optimal parameter regions and the time-dependence of control parameter values for the particle swarm optimization algorithm," vol. 41, pp. 20-35, 2018.

[8] M. Hajihassani, D. J. Armaghani, R. J. G. Kalatehjari, and G. Engineering, "Applications of particle swarm optimization in geotechnical engineering: a comprehensive review," vol. 36, no. 2, pp. 705-722, 2018.

[9] A. Elsheikh, M. J. I. j. o. e. s. Abd Elaziz, and technology, "Review on applications of particle swarm optimization in solar energy systems," vol. 16, no. 2, pp. 1159-1170, 2019.

[10] M. R. Bonyadi and Z. Michalewicz, "Particle swarm optimization for single objective continuous space problems: a review," *Evolutionary computation*, 2016.

[11] A. R. Jordehi and J. Jasni, "Particle swarm optimisation for discrete optimisation problems: a review," *Artificial Intelligence Review*, vol. 43, no. 2, pp. 243-258, 2015.

[12] A. Khare and S. Rangnekar, "A review of particle swarm optimization and its applications in solar photovoltaic system," *Applied Soft Computing*, vol. 13, no. 5, pp. 2997-3006, 2013.

[13] N. Jain, U. Nangia, and J. Jain, "A review of particle swarm optimization," *Journal of The Institution of Engineers (India): Series B*, vol. 99, no. 4, pp. 407-411, 2018.

[14] M. R. Bonyadi and Z. Michalewicz, "Particle swarm optimization for single objective continuous space problems: a review," *Evolutionary computation*, vol. 25, no. 1, pp. 1-54, 2017.

[15] T. V. Sibalija, "Particle swarm optimisation in designing parameters of manufacturing processes: A review (2008–2018)," *Applied Soft Computing*, vol. 84, p. 105743, 2019.

[16] B. Xin, J. Chen, J. Zhang, H. Fang, and Z.-H. Peng, "Hybridizing differential evolution and particle swarm optimization to design powerful optimizers: a review and taxonomy," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 5, pp. 744-767, 2011.

[17] S. Rana, S. Jasola, and R. Kumar, "A review on particle swarm optimization algorithms and their applications to data clustering," *Artificial Intelligence Review*, vol. 35, no. 3, pp. 211-222, 2011.

[18] A. R. Jordehi, "Particle swarm optimisation (PSO) for allocation of FACTS devices in electric transmission systems: A review," *Renewable and Sustainable Energy Reviews*, vol. 52, pp. 1260-1267, 2015.

[19] M.-P. Song and G.-C. Gu, "Research on particle swarm optimization: a review," in *Proceedings of 2004 International Conference on Machine Learning and Cybernetics (IEEE Cat. No. 04EX826)*, 2004, vol. 4, pp. 2236-2241: IEEE.

[20] M. Juneja and S. Nagar, "Particle swarm optimization algorithm and its parameters: A review," in *2016 International Conference on Control, Computing, Communication and Materials (ICCCCM)*, 2016, pp. 1-5: IEEE.

[21] A. A. Esmin, R. A. Coelho, and S. Matwin, "A review on particle swarm optimization algorithm and its variants to clustering high-dimensional data," Artificial Intelligence Review, vol. 44, no. 1, pp. 23-45, 2015.

[22] W. Fang, J. Sun, Y. Ding, X. Wu, and W. Xu, "A review of quantum-behaved particle swarm optimization," IETE Technical Review, vol. 27, no. 4, pp. 336-348, 2010.

[23] D. ping Tian, "A review of convergence analysis of particle swarm optimization," International Journal of Grid and Distributed Computing, vol. 6, no. 6, pp. 117-128, 2013.

[24] M. N. K. Kulkarni, M. S. Patekar, M. T. Bhoskar, M. O. Kulkarni, G. Kakandikar, and V. Nandedkar, "Particle swarm optimization applications to mechanical engineering-a review," Materials Today: Proceedings, vol. 2, no. 4-5, pp. 2631-2639, 2015.

[25] A. R. Jordehi, "Particle swarm optimisation for dynamic optimisation problems: a review," Neural Computing and Applications, vol. 25, no. 7, pp. 1507-1516, 2014.

[26] A. R. Jordehi, "A review on constraint handling strategies in particle swarm optimisation," Neural Computing and Applications, vol. 26, no. 6, pp. 1265-1275, 2015.

[27] A. P. Engelbrecht, "Particle swarm optimization with crossover: a review and empirical analysis," Artificial Intelligence Review, vol. 45, no. 2, pp. 131-165, 2016.

[28] S. Sengupta, S. Basak, R. A. J. M. L. Peters, and K. Extraction, "Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives," vol. 1, no. 1, pp. 157-191, 2019.

[29] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part I: background and development," Natural Computing, vol. 6, no. 4, pp. 467-484, 2007.

[30] A. Banks, J. Vincent, and C. Anyakoha, "A review of particle swarm optimization. Part II: hybridisation, combinatorial, multicriteria and constrained optimization, and indicative applications," Natural Computing, vol. 7, no. 1, pp. 109-124, 2008.

[31] R. C. Eberhart, P. K. Simpson, R. Dobbins, and R. W. Dobbins, Computational intelligence PC tools. AP Professional, 1996.

[32] A. P. Engelbrecht, Computational Intelligence: An Introduction. Wiley, 2007.

[33] M. Ghasemi, E. Akbari, A. Rahimnejad, S. E. Razavi, S. Ghavidel, and L. J. S. C. Li, "Phasor particle swarm optimization: a simple and efficient variant of PSO," vol. 23, no. 19, pp. 9701-9718, 2019.

[34] F. Van den Bergh and A. P. Engelbrecht, "Effects of swarm size on cooperative particle swarm optimisers," 2001.

[35] N. Himanshu, A. J. G. Burman, and G. Engineering, "Determination of critical failure surface of slopes using particle swarm optimization technique considering seepage and seismic loading," vol. 37, no. 3, pp. 1261-1281, 2019.

[36] G. A. F. Alfarisy, W. F. Mahmudy, and M. H. J. T. Natsir, "Optimizing laying hen diet using multi-swarm particle swarm optimization," vol. 16, no. 4, pp. 1712-1723, 2018.

[37] W. N. Chen et al., "Particle Swarm Optimization With an Aging Leader and Challengers," Evolutionary Computation, IEEE Transactions on, vol. 17, no. 2, pp. 241-258, 2013.

[38] A. P. Piotrowski, J. J. Napiorkowski, A. E. J. S. Piotrowska, and E. Computation, "Population size in particle swarm optimization," vol. 58, p. 100718, 2020.

[39] D. Wang, D. Tan, and L. J. S. C. Liu, "Particle swarm optimization algorithm: an overview," vol. 22, no. 2, pp. 387-408, 2018.

[40] A. Ratnaweera, S. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," Evolutionary Computation, IEEE Transactions on, vol. 8, no. 3, pp. 240-255, 2004.

[41] Z. Beheshti and S. M. Hj. Shamsuddin, "CAPSO: Centripetal accelerated particle swarm optimization," Information Sciences, vol. 258, no. 0, pp. 54-79, 2/10/ 2014.

[42] Z. Zhi-Hui, Z. Jun, L. Yun, and S. Yu-hui, "Orthogonal Learning Particle Swarm Optimization," Evolutionary Computation, IEEE Transactions on, vol. 15, no. 6, pp. 832-847, 2011.

[43] E. Koessler, A. J. O. Almomani, and Engineering, "Hybrid particle swarm optimization and pattern search algorithm," pp. 1-17, 2020.

[44] Z. Yang, A. J. N. C. Wu, and Applications, "A non-revisiting quantum-behaved particle swarm optimization based multilevel thresholding for image segmentation," vol. 32, no. 16, pp. 12011-12031, 2020.

[45] S. Yuhui and R. Eberhart, "A modified particle swarm optimizer," in Evolutionary Computation Proceedings, 1998. IEEE World Congress on Computational Intelligence., The 1998 IEEE International Conference on, 1998, pp. 69-73.

[46] M. Clerc and J. Kennedy, "The particle swarm - explosion, stability, and convergence in a multidimensional complex space," Evolutionary Computation, IEEE Transactions on, vol. 6, no. 1, pp. 58-73, 2002.

[47] M. Li, H. Chen, X. Shi, S. Liu, M. Zhang, and S. J. A. S. C. Lu, "A multi-information fusion “triple variables with iteration” inertia weight PSO algorithm and its application," vol. 84, p. 105677, 2019.

[48] S. Yuhui and R. C. Eberhart, "Empirical study of particle swarm optimization," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, 1999, vol. 3, p. 1950 Vol. 3.

[49] H.-G. Beyer and H.-P. Schwefel, "Evolution strategies–A comprehensive introduction," Natural computing, vol. 1, no. 1, pp. 3-52, 2002.

[50] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," Applied Soft Computing, vol. 11, no. 4, pp. 3658-3670, 2011.

[51] R. C. Eberhart and S. Yuhui, "Tracking and optimizing dynamic systems with particle swarms," in Evolutionary Computation, 2001. Proceedings of the 2001 Congress on, 2001, vol. 1, pp. 94-100 vol. 1.

[52] K. Y. Lee and P. Jong-Bae, "Application of Particle Swarm Optimization to Economic Dispatch Problem: Advantages and Disadvantages," in Power Systems Conference and Exposition, 2006. PSCE '06. 2006 IEEE PES, 2006, pp. 188-192.

[53] B. Jiao, Z. Lian, and X. Gu, "A dynamic inertia weight particle swarm optimization algorithm," Chaos, Solitons & Fractals, vol. 37, no. 3, pp. 698-705, 2008.

[54] S.-K. S. Fan and Y.-Y. Chiu, "A decreasing inertia weight particle swarm optimizer," Engineering Optimization, vol. 39, no. 2, pp. 203-228, 2007.

[55] A. Chatterjee and P. Siarry, "Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization," Computers & Operations Research, vol. 33, no. 3, pp. 859-871, 3// 2006.

[56] C. Guimin, H. Xinbo, J. Jianyuan, and M. Zhengfeng, "Natural Exponential Inertia Weight Strategy in Particle Swarm Optimization," in Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on, 2006, vol. 1, pp. 3672-3675.

[57] L. Hui-Rong and G. Yue-Lin, "Particle Swarm Optimization Algorithm with Exponent Decreasing Inertia Weight and Stochastic Mutation," in Information and Computing Science, 2009. ICIC '09. Second International Conference on, 2009, vol. 1, pp. 66-69.

[58] J. Lu, H. Hu, and Y. Bai, "Radial Basis Function Neural Network Based on an Improved Exponential Decreasing Inertia Weight-Particle Swarm Optimization Algorithm for AQI Prediction," in Abstract and Applied Analysis, 2014, vol. 2014: Hindawi Publishing Corporation.

[59] F. Yong, T. Gui-fa, W. Ai-Xin, and Y. Yong-Mei, "Chaotic Inertia Weight in Particle Swarm Optimization," in Innovative Computing, Information and Control, 2007. ICICIC '07. Second International Conference on, 2007, pp. 475-475.

[60] X. Yang, J. Yuan, J. Yuan, and H. Mao, "A modified particle swarm optimizer with dynamic adaptation," Applied Mathematics and Computation, vol. 189, no. 2, pp. 1205-1213, 2007.

[61] B. K. Panigrahi, V. Ravikumar Pandi, and S. Das, "Adaptive particle swarm optimization approach for static and dynamic economic load dispatch," Energy Conversion and Management, vol. 49, no. 6, pp. 1407-1415, 6// 2008.

[62] K. Suresh, S. Ghosh, D. Kundu, A. Sen, S. Das, and A. Abraham, "Inertia-Adaptive Particle Swarm Optimizer for Improved Global Search," in Intelligent Systems Design and Applications, 2008. ISDA '08. Eighth International Conference on, 2008, vol. 2, pp. 253-258.

[63] L. Kaiyou, Q. Yuhui, and H. Yi, "A new adaptive well-chosen inertia weight strategy to automatically harmonize global and local search ability in particle swarm optimization," in Systems and Control in Aerospace and Astronautics, 2006. ISSCAA 2006. 1st International Symposium on, 2006, pp. 4 pp.-980.

[64] M. S. Arumugam and M. V. C. Rao, "On the improved performances of the particle swarm optimization algorithms with adaptive parameters, cross-over operators and root mean square (RMS) variants for computing optimal control of a class of hybrid systems," Applied Soft Computing, vol. 8, no. 1, pp. 324-336, 1// 2008.

[65] Z. Zhi-Hui, Z. Jun, L. Yun, and H. S. H. Chung, "Adaptive Particle Swarm Optimization," Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, vol. 39, no. 6, pp. 1362-1381, 2009.

[66] J. Kennedy, "Small worlds and mega-minds: effects of neighborhood topology on particle swarm performance," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, 1999, vol. 3, p. 1938 Vol. 3.

[67] R. Mendes, J. Kennedy, and J. Neves, "The fully informed particle swarm: simpler, maybe better," Evolutionary Computation, IEEE Transactions on, vol. 8, no. 3, pp. 204-210, 2004.

[68] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," in Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on, 2002, vol. 2, pp. 1671-1676.

[69] P. N. Suganthan, "Particle swarm optimiser with neighbourhood operator," in Evolutionary Computation, 1999. CEC 99. Proceedings of the 1999 Congress on, 1999, vol. 3: IEEE.

[70] M. Nasir, S. Das, D. Maity, S. Sengupta, U. Halder, and P. N. Suganthan, "A dynamic neighborhood learning based particle swarm optimizer for global numerical optimization," Information Sciences, vol. 209, no. 0, pp. 16-36, 11/20/ 2012.

[71] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," Evolutionary Computation, IEEE Transactions on, vol. 10, no. 3, pp. 281-295, 2006.

[72] A. Godoy and F. J. Von Zuben, "A Complex Neighborhood based Particle Swarm Optimization," in Evolutionary Computation, 2009. CEC '09. IEEE Congress on, 2009, pp. 720-727.

[73] K. E. Parsopoulos and M. N. Vrahatis, "Unified Particle Swarm Optimization in Dynamic Environments," in Applications of Evolutionary Computing, vol. 3449, F. Rothlauf et al., Eds. (Lecture Notes in Computer Science: Springer Berlin Heidelberg, 2005, pp. 590-599.

[74] J. Kennedy and R. Mendes, "Population structure and particle swarm performance," 2002.

[75] R. Mendes, J. Kennedy, and J. Neves, "Watch thy neighbor or how the swarm can learn from its environment," 2003: Swarm Intelligence Symposium, 2003. SIS'03. Proceedings of the 2003 IEEE.

[76] Y. Xu, D. J. N. C. Pi, and Applications, "A reinforcement learning-based communication topology in particle swarm optimization," pp. 1-26, 2019.

[77] H. Zhang, M. Yuan, Y. Liang, and Q. Liao, "A novel particle swarm optimization based on prey–predator relationship," Applied Soft Computing, vol. 68, pp. 202-218, 2018.

[78] W. Ye, W. Feng, and S. Fan, "A novel multi-swarm particle swarm optimization with dynamic learning strategy," Applied Soft Computing, vol. 61, pp. 832-843, 2017.

[79] J. Gou, Y.-X. Lei, W.-P. Guo, C. Wang, Y.-Q. Cai, and W. Luo, "A novel improved particle swarm optimization algorithm based on individual difference evolution," Applied Soft Computing, vol. 57, pp. 468-481, 2017.

[80] Y. Chen, L. Li, H. Peng, J. Xiao, Y. Yang, and Y. Shi, "Particle swarm optimizer with two differential mutation," Applied Soft Computing, vol. 61, pp. 314-330, 2017.

[81] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, and X. Liu, "A novel sigmoid-function-based adaptive weighted particle swarm optimizer," IEEE transactions on cybernetics, 2019.

[82] X. Zhang, X. Wang, Q. Kang, and J. Cheng, "Differential mutation and novel social learning particle swarm optimization algorithm," Information Sciences, vol. 480, pp. 109-129, 2019.

[83] M. Kohler, M. M. Vellasco, and R. Tanscheit, "PSO+: A new particle swarm optimization algorithm for constrained problems," Applied Soft Computing, vol. 85, p. 105865, 2019.

[84] W. Li, X. Meng, Y. Huang, and Z.-H. Fu, "Multipopulation cooperative particle swarm optimization with a mixed mutation strategy," Information Sciences, vol. 529, pp. 179-196, 2020.

[85] D. Sedighizadeh, E. Masehian, M. Sedighizadeh, and H. Akbaripour, "GEPSO: A new generalized particle swarm optimization algorithm," Mathematics and Computers in Simulation, vol. 179, pp. 194-212, 2021.

[86] K. Chen, B. Xue, M. Zhang, and F. Zhou, "Novel chaotic grouping particle swarm optimization with a dynamic regrouping strategy for solving numerical optimization tasks," Knowledge-Based Systems, vol. 194, p. 105568, 2020.

[87] J. T. Machado, S. M. A. Pahnehkolaei, and A. Alfi, "Complex-order particle swarm optimization," Communications in Nonlinear Science and Numerical Simulation, vol. 92, p. 105448, 2021.

[88] X. Xia et al., "An expanded particle swarm optimization based on multi-exemplar and forgetting ability," Information Sciences, vol. 508, pp. 105-120, 2020.

[89] N. Lynn and P. N. Suganthan, "Ensemble particle swarm optimizer," Applied Soft Computing, vol. 55, pp. 533-548, 2017.

[90] Y. Shi and R. C. Eberhart, "Empirical study of particle swarm optimization," in Proceedings of the 1999 congress on evolutionary computation-CEC99 (Cat. No. 99TH8406), 1999, vol. 3, pp. 1945-1950: IEEE.

[91] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," IEEE transactions on evolutionary computation, vol. 10, no. 3, pp. 281-295, 2006.

[92] B.-Y. Qu, P. N. Suganthan, and S. Das, "A distance-based locally informed particle swarm model for multimodal optimization," IEEE Transactions on Evolutionary Computation, vol. 17, no. 3, pp. 387-402, 2012.

[93] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," IEEE Transactions on evolutionary computation, vol. 8, no. 3, pp. 240-255, 2004.

[94] T. Peram, K. Veeramachaneni, and C. K. Mohan, "Fitness-distance-ratio based particle swarm optimization," in Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03 (Cat. No. 03EX706), 2003, pp. 174-181: IEEE.

[95] N. M. Laskar, K. Guha, I. Chatterjee, S. Chanda, K. L. Baishnab, and P. K. Paul, "HWPSO: A new hybrid whale-particle swarm optimization algorithm and its application in electronic design optimization problems," Applied Intelligence, vol. 49, no. 1, pp. 265-291, 2019.

[96] F. Wang, H. Zhang, K. Li, Z. Lin, J. Yang, and X.-L. Shen, "A hybrid particle swarm optimization algorithm using adaptive learning strategy," Information Sciences, vol. 436, pp. 162-177, 2018.

[97] K. Chen, F. Zhou, L. Yin, S. Wang, Y. Wang, and F. Wan, "A hybrid particle swarm optimizer with sine cosine acceleration coefficients," Information Sciences, vol. 422, pp. 218-241, 2018.

[98] M. Karimi-Mamaghan, M. Mohammadi, P. Meyer, A. M. Karimi-Mamaghan, and E.-G. J. E. J. o. O. R. Talbi, "Machine Learning at the service of Meta-heuristics for solving Combinatorial Optimization Problems: A state-of-the-art," 2021.

[99] M. A. Potter and K. A. De Jong, "A cooperative coevolutionary approach to function optimization," in Parallel Problem Solving from Nature—PPSN III: Springer, 1994, pp. 249-257.

[100] F. Van den Bergh and A. P. Engelbrecht, "A Cooperative approach to particle swarm optimization," Evolutionary Computation, IEEE Transactions on, vol. 8, no. 3, pp. 225-239, 2004.

[101] W. Ruei-Yang, H. Yu-Ting, and L. Wei-Po, "A new cooperative particle swarm optimizer with dimension partition and adaptive velocity control," in Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on, 2012, pp. 103-109.

[102] L. Wang, "An improved cooperative particle swarm optimizer," (in English), Telecommunication Systems, vol. 53, no. 1, pp. 147-154, 2013/05/01 2013.

[103] B. Niu, Y. Zhu, X. He, and H. Wu, "MCPSO: A multi-swarm cooperative particle swarm optimizer," Applied Mathematics and Computation, vol. 185, no. 2, pp. 1050-1062, 2/15/ 2007.

[104] H. Rau, S. D. Budiman, G. A. J. T. R. P. E. L. Widyadana, and T. Review, "Optimization of the multi-objective green cyclical inventory routing problem using discrete multi-swarm PSO method," vol. 120, pp. 51-75, 2018.

[105] J. Zhang and X. Ding, "A Multi-Swarm Self-Adaptive and Cooperative Particle Swarm Optimization," Engineering Applications of Artificial Intelligence, vol. 24, no. 6, pp. 958-967, 9// 2011.

[106] C.-H. Chen and Y.-Y. Liao, "Tribal particle swarm optimization for neurofuzzy inference systems and its prediction applications," Communications in Nonlinear Science and Numerical Simulation, vol. 19, no. 4, pp. 914-929, 2014.

[107] J. H. Holland, Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. U Michigan Press, 1975.

[108] Y.-T. Kao and E. Zahara, "A hybrid genetic algorithm and particle swarm optimization for multimodal functions," Applied Soft Computing, vol. 8, no. 2, pp. 849-857, 2008.

[109] R. Chelouah and P. Siarry, "A Continuous Genetic Algorithm Designed for the Global Optimization of Multimodal Functions," (in English), Journal of Heuristics, vol. 6, no. 2, pp. 191-213, 2000/06/01 2000.

[110] S.-K. S. Fan, Y.-c. Liang, and E. Zahara, "Hybrid simplex search and particle swarm optimization for the global optimization of multimodal functions," Engineering Optimization, vol. 36, no. 4, pp. 401-418, 2004.

[111] J. Robinson, S. Sinton, and Y. Rahmat-Samii, "Particle swarm, genetic algorithm, and their hybrids: optimization of a profiled corrugated horn antenna," in *Antennas and Propagation Society International Symposium, 2002. IEEE*, 2002, vol. 1, pp. 314-317 vol.1.

[112] Z. Chai, A. Nwachukwu, Y. Zagayevskiy, S. Amini, S. J. J. o. P. S. Madasu, and Engineering, "An integrated closed-loop solution to assisted history matching and field optimization with machine learning techniques," vol. 198, p. 108204, 2021.

[113] N. M. Isa, A. L. Bukar, T. C. Wei, and A. Marwanto, "Optimal Sizing of Hybrid Fuel Cell and PV Employing Hybrid PSO-GA," in *2019 IEEE Conference on Energy Conversion (CENCON)*, 2019, pp. 159-164: IEEE.

[114] R. Storn and K. Price, *Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces*. ICSI Berkeley, 1995.

[115] S. Sayah and A. Hamouda, "A hybrid differential evolution algorithm based on particle swarm optimization for nonconvex economic dispatch problems," *Applied Soft Computing,* vol. 13, no. 4, pp. 1608-1619, 4// 2013.

[116] H. M. Elragal, M. A. Mangoud, and M. T. Alsharaa, "Hybrid differential evolution and enhanced particle swarm optimisation

technique for design of reconfigurable phased antenna arrays," *Microwaves, Antennas & Propagation, IET,* vol. 5, no. 11, pp. 1280-1287, 2011.

[117] M. G. Epitropakis, V. P. Plagianakos, and M. N. Vrahatis, "Evolving cognitive and social experience in Particle Swarm Optimization through Differential Evolution: A hybrid approach," *Information Sciences,* vol. 216, no. 0, pp. 50-92, 12/20/ 2012.

[118] A. Sedki and D. Ouazar, "Hybrid particle swarm optimization and differential evolution for optimal design of water distribution systems," *Advanced Engineering Informatics,* vol. 26, no. 3, pp. 582-591, 8// 2012.

[119] M. Pant, R. Thangaraj, C. Grosan, and A. Abraham, "Hybrid differential evolution - Particle Swarm Optimization algorithm for solving global optimization problems," in *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*, 2008, pp. 18-24.

[120] F. Yangguang, D. Mingyue, Z. Chengping, and H. Hanping, "Route Planning for Unmanned Aerial Vehicle (UAV) on the Sea Using Hybrid Differential Evolution and Quantum-Behaved Particle Swarm Optimization," *Systems, Man, and Cybernetics: Systems, IEEE Transactions on,* vol. 43, no. 6, pp. 1451-1465, 2013.

[121] J. Sun, B. Feng, and W. Xu, "Particle swarm optimization with particles having quantum behavior," in *Proceedings of the 2004 congress on evolutionary computation (IEEE Cat. No. 04TH8753)*, 2004, vol. 1, pp. 325-331: IEEE.

[122] M. S. Kıran, M. Gündüz, and Ö. K. Baykan, "A novel hybrid algorithm based on particle swarm and ant colony optimization for finding the global minimum," *Applied Mathematics and Computation,* vol. 219, no. 4, pp. 1515-1521, 2012.

[123] G. Eappen and T. J. P. C. Shankar, "Hybrid PSO-GSA for energy efficient spectrum sensing in cognitive radio network," vol. 40, p. 101091, 2020.

[124] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46-61, 2014.

[125] P. J. Van Laarhoven and E. H. Aarts, "Simulated annealing," in *Simulated annealing: Theory and applications*: Springer, 1987, pp. 7-15.

[126] X. Pan, L. Xue, Y. Lu, and N. Sun, "Hybrid particle swarm optimization with simulated annealing," *Multimedia Tools and Applications,* vol. 78, no. 21, pp. 29921-29936, 2019.

[127] J. Rangaraj and M. J. I. J. o. A. E. R. Anitha, "Implementing energy optimization by a novel hybrid technique for performance improvement in mobile ad hoc network," vol. 12, no. 22, pp. 12029-12035, 2017.

[128] O. M. Kamel, A. A. Z. Diab, T. D. Do, and M. A. J. I. A. Mossa, "A Novel Hybrid Ant Colony-Particle Swarm Optimization Techniques Based Tuning STATCOM for Grid Code Compliance," vol. 8, pp. 41566-41587, 2020.

[129] H. Suyono, E. Subekti, H. Purnomo, T. Nurwati, and R. N. Hasanah, "Economic Dispatch of 500 kV Java-Bali Power System using Hybrid Particle Swarm-Ant Colony Optimization Method," in *2020 12th International Conference on Electrical Engineering (ICEENG)*, 2020, pp. 5-10: IEEE.

[130] S. Jiang, Z. Ji, and Y. Shen, "A novel hybrid particle swarm optimization and gravitational search algorithm for solving economic emission load dispatch problems with various practical constraints," *International Journal of Electrical Power & Energy Systems,* vol. 55, no. 0, pp. 628-644, 2// 2014.

[131] H.-C. Tsai, Y.-Y. Tyan, Y.-W. Wu, and Y.-H. Lin, "Gravitational particle swarm," *Applied Mathematics and Computation,* vol. 219, no. 17, pp. 9106-9117, 2013.

[132] Y. Zhuang and H. Huang, "Time-optimal trajectory planning for underactuated spacecraft using a hybrid particle swarm optimization algorithm," *Acta Astronautica,* vol. 94, no. 2, pp. 690-698, 2// 2014.

[133] H. Haklı and H. Uğuz, "A novel particle swarm optimization algorithm with Levy flight," *Applied Soft Computing,* vol. 23, pp. 333-345, 2014.

[134] M. Abdullah, A. Bakar, N. Rahim, H. Mokhlis, H. Illias, and J. Jamian, "Modified Particle Swarm Optimization with Time Varying Acceleration Coefficients for Economic Load Dispatch with Generator Constraints," *Journal of Electrical Engineering & Technology*, vol. 9, no. 1, pp. 15-26, 2014.

[135] L. Joon-Woo and L. Ju-Jang, "Gaussian-distributed Particle Swarm Optimization: A novel Gaussian Particle Swarm Optimization," in *Industrial Technology (ICIT), 2013 IEEE International Conference on*, 2013, pp. 1122-1127.

[136] J.-W. Lee and J.-J. Lee, "Gaussian-distributed particle swarm optimization: A novel Gaussian particle swarm optimization," in *2013 IEEE International Conference on Industrial Technology (ICIT)*, 2013, pp. 1122-1127: IEEE.

[137] M.-S. Leu and M.-F. Yeh, "Grey particle swarm optimization," *Applied Soft Computing,* vol. 12, no. 9, pp. 2985-2996, 2012.

[138] N.-J. Li, W.-J. Wang, C.-C. James Hsu, W. Chang, H.-G. Chou, and J.-W. Chang, "Enhanced particle swarm optimizer incorporating a weighted particle," *Neurocomputing,* vol. 124, pp. 218-227, 2014.

[139] F. M. F. Hafiz and A. Abdennour, "A team-oriented approach to particle swarms," *Applied Soft Computing,* vol. 13, no. 9, pp. 3776-3791, 9// 2013.

[140] Z. Beheshti, S. M. H. Shamsuddin, and S. Hasan, "MPSO: median-oriented particle swarm optimization," *Applied Mathematics and Computation,* vol. 219, no. 11, pp. 5817-5836, 2013.

[141] J. Kennedy and R. C. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Systems, Man, and Cybernetics, 1997. Computational Cybernetics and Simulation., 1997 IEEE International Conference on*, 1997, vol. 5, pp. 4104-4108 vol.5.

[142] N. Tawalbeh, H. M. Abusamaha, and A. Al-Salaymeh, "Domestic appliances scheduling using BPSO and IoT," in *2019 IEEE PES Innovative Smart Grid Technologies Europe (ISGT-Europe)*, 2019, pp. 1-5: IEEE.

[143] E. Chen, J. Li, and X. Liu, "In search of the essential binary discrete particle swarm," *Applied Soft Computing,* vol. 11, no. 3, pp. 3260-3269, 2011.

[144] X. Yuan, H. Nie, A. Su, L. Wang, and Y. Yuan, "An improved binary particle swarm optimization for unit commitment problem," *Expert Systems with Applications,* vol. 36, no. 4, pp. 8049-8055, 5// 2009.

[145] S. Lee, S. Soak, S. Oh, W. Pedrycz, and M. Jeon, "Modified binary particle swarm optimization," *Progress in Natural Science,* vol. 18, no. 9, pp. 1161-1166, 2008.

[146] L. Wang, X. Wang, J. Fu, and L. Zhen, "A novel probability binary particle swarm optimization algorithm and its application," *Journal of software,* vol. 3, no. 9, pp. 28-35, 2008.

[147] W. Ling, F. Xiping, F. Jiating, W. Haikuan, and F. Minrui, "Optimal node placement in industrial Wireless Sensor Networks using adaptive mutation probability binary Particle Swarm Optimization algorithm," in *Natural Computation (ICNC), 2011 Seventh International Conference on*, 2011, vol. 4, pp. 2199-2203.

[148] I. M. Swesi and A. A. J. I. J. A. S. C. A. Bakar, "An enhanced binary particle swarm optimization (EBPSO) algorithm based a v-shaped transfer function for feature selection in high dimensional data," vol. 9, no. 3, 2017.

[149] L. Jian-Hua, Y. Rong-Hua, and S. Shui-Hua, "The analysis of binary particle swarm optimization," *Journal of Nanjing University (Natural Sciences),* vol. 47, no. 5, pp. 504-514, 2011.

[150] J. Yang, H. Zhang, Y. Ling, C. Pan, and W. Sun, "Task Allocation for Wireless Sensor Network Using Modified Binary Particle Swarm Optimization," *Sensors Journal, IEEE,* vol. 14, no. 3, pp. 882-892, 2014.

[151] S. Mirjalili and A. Lewis, "S-shaped versus V-shaped transfer functions for binary particle swarm optimization," *Swarm and Evolutionary Computation,* vol. 9, pp. 1-14, 2013.

[152] A. H. El-Maleh, A. T. Sheikh, and S. M. Sait, "Binary particle swarm optimization (BPSO) based state assignment for area minimization of sequential circuits," *Applied Soft Computing,* vol. 13, no. 12, pp. 4832-4840, 12// 2013.

[153] W. Bin, P. Qinke, Z. Jing, and C. Xiao, "A binary particle swarm optimization algorithm inspired by multi-level organizational learning behavior," *European Journal of Operational Research,* vol. 219, no. 2, pp. 224-233, 2012.

[154] Y. Mehmood, M. Sadiq, W. Shahzad, and F. Amin, "Fitness-based acceleration coefficients to enhance the convergence speed of novel binary particle swarm optimization," in *2018 International Conference on Frontiers of Information Technology (FIT)*, 2018, pp. 355-360: IEEE.

[155] P. Agrawal, T. Ganesh, and A. W. Mohamed, "A novel binary gaining–sharing knowledge-based optimization algorithm for feature selection," *Neural Computing and Applications,* vol. 33, no. 11, pp. 5989-6008, 2021.

[156] A. W. Mohamed, A. A. Hadi, and A. K. Mohamed, "Gaining-sharing knowledge based algorithm for solving optimization problems: a novel nature-inspired algorithm," *International Journal of Machine Learning and Cybernetics,* pp. 1-29, 2019.

[157] L. Kumar and K. K. Bharti, "A novel hybrid BPSO–SCA approach for feature selection," *Natural Computing,* pp. 1-23, 2019.

[158] G. Lin, J. Guan, Z. Li, and H. Feng, "A hybrid binary particle swarm optimization with tabu search for the set-union knapsack problem," *Expert Systems with Applications,* vol. 135, pp. 201-211, 2019.

[159] J. Too, A. R. Abdullah, and N. Mohd Saad, "A new co-evolution binary particle swarm optimization with multiple inertia weight strategy for feature selection," in *Informatics*, 2019, vol. 6, no. 2, p. 21: Multidisciplinary Digital Publishing Institute.

[160] Z. Beheshti, "A time-varying mirrored S-shaped transfer function for binary particle swarm optimization," *Information Sciences,* vol. 512, pp. 1503-1542, 2020.

[161] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "BGSA: binary gravitational search algorithm," *Natural Computing,* vol. 9, no. 3, pp. 727-745, 2010.

[162] M. J. Islam, X. Li, and Y. Mei, "A time-varying transfer function for balancing the exploration and exploitation ability of a binary PSO," *Applied Soft Computing,* vol. 59, pp. 182-196, 2017.

[163] Z. Beheshti, "A novel x-shaped binary particle swarm optimization," *Soft Computing,* vol. 25, no. 4, pp. 3013-3042, 2021.

[164] S. Mirjalili, H. Zhang, S. Mirjalili, S. Chalup, and N. Noman, "A Novel U-Shaped Transfer Function for Binary Particle Swarm Optimisation," in *9th International Conference on Soft Computing for Problem Solving, SocProS 2019*, 2020, pp. 241-259: Springer Gabler.

[165] L. Wang, X. Wang, J. Fu, and L. Zhen, "A Novel Probability Binary Particle Swarm Optimization Algorithm and its Application," *J. Softw.,* vol. 3, no. 9, pp. 28-35, 2008.

[166] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Advances in engineering software,* vol. 95, pp. 51-67, 2016.

[167] J. G. Digalakis and K. G. Margaritis, "On benchmarking functions for genetic algorithms," *International journal of computer mathematics,* vol. 77, no. 4, pp. 481-506, 2001.

[168] P. M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," *IEEE Transactions on computers,* vol. 26, no. 09, pp. 917-922, 1977.

[169] B. H. Nguyen, B. Xue, and M. Zhang, "A survey on swarm intelligence approaches to feature selection in data mining," *Swarm and Evolutionary Computation,* vol. 54, p. 100663, 2020.

[170] Q. Al-Tashi, H. M. Rais, S. J. Abdulkadir, S. Mirjalili, and H. Alhussian, "A review of grey wolf optimizer-based feature selection methods for classification," *Evolutionary machine learning techniques,* pp. 273-286, 2020.

[171] Q. Al-Tashi, H. M. Rais, S. J. Abdulkadir, and S. Mirjalili, "Feature selection based on Grey Wolf Optimizer for oil & gas reservoir classification," in *2020 International Conference on Computational Intelligence (ICCI),* 2020, pp. 211-216: IEEE.

[172] Q. Al-Tashi, H. Rais, and S. Jadid, "Feature selection method based on grey wolf optimization for coronary artery disease classification," in *International conference of reliable information and communication technology*, 2018, pp. 257-266: Springer.

[173] P. Langley, "Selection of relevant features in machine learning," in *Proceedings of the AAAI Fall symposium on relevance*, 1994, vol. 184, pp. 245-271.

[174] E. Hancer, B. Xue, and M. Zhang, "A survey on feature selection approaches for clustering," *Artificial Intelligence Review,* vol. 53, no. 6, pp. 4519-4545, 2020.

[175] A. Mukhopadhyay, U. Maulik, S. Bandyopadhyay, and C. A. C. Coello, "A survey of multiobjective evolutionary algorithms for data mining: Part I," *IEEE Transactions on Evolutionary Computation,* vol. 18, no. 1, pp. 4-19, 2013.

[176] M. Dash and H. Liu, "Feature selection for classification," *Intelligent data analysis,* vol. 1, no. 1-4, pp. 131-156, 1997.

[177] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of machine learning research,* vol. 3, no. Mar, pp. 1157-1182, 2003.

[178] H. Liu and Z. Zhao, "Manipulating data and dimension reduction methods: Feature selection," in *Computational Complexity: theory, techniques, and applications*: Springer New York, 2012, pp. 1790-1800.

[179] H. Liu, H. Motoda, R. Setiono, and Z. Zhao, "Feature selection: An ever evolving frontier in data mining," in *Feature selection in data mining*, 2010, pp. 4-13: PMLR.

[180] B. Ji, X. Lu, G. Sun, W. Zhang, J. Li, and Y. Xiao, "Bio-inspired feature selection: An improved binary particle swarm optimization approach," *IEEE Access,* vol. 8, pp. 85989-86002, 2020.

[181] B. Xiong, Y. Li, M. Huang, W. Shi, M. Du, and Y. Yang, "Feature selection of input variables for intelligence joint moment prediction based on binary particle swarm optimization," *IEEE Access,* vol. 7, pp. 182289-182295, 2019.

[182] J. Too, A. R. Abdullah, and N. Mohd Saad, "Hybrid binary particle swarm optimization differential evolution-based feature selection for EMG signals classification," *Axioms,* vol. 8, no. 3, p. 79, 2019.

[183] J. Too, A. R. Abdullah, N. Mohd Saad, and W. Tee, "EMG feature selection and classification using a Pbest-guide binary particle swarm optimization," *Computation,* vol. 7, no. 1, p. 12, 2019.

[184] I. Jain, V. K. Jain, and R. Jain, "Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification," *Applied Soft Computing,* vol. 62, pp. 203-215, 2018.

[185] S. Rajamohana and K. Umamaheswari, "Hybrid approach of improved binary particle swarm optimization and shuffled frog leaping for feature selection," *Computers & Electrical Engineering,* vol. 67, pp. 497-508, 2018.

[186] S. Gunasundari, S. Janakiraman, and S. Meenambal, "Multiswarm heterogeneous binary PSO using win-win approach for improved feature selection in liver and kidney disease diagnosis," *Computerized Medical Imaging and Graphics,* vol. 70, pp. 135-154, 2018.

[187] H. Banka and S. Dara, "A Hamming distance based binary particle swarm optimization (HDBPSO) algorithm for high dimensional feature selection, classification and validation," *Pattern Recognition Letters,* vol. 52, pp. 94-100, 2015.

[188] S. Dara, H. Banka, and C. S. R. Annavarapu, "A rough based hybrid binary PSO algorithm for flat feature selection and classification in gene expression data," *Annals of data science,* vol. 4, no. 3, pp. 341-360, 2017.

[189] K. Mistry, L. Zhang, S. C. Neoh, C. P. Lim, and B. Fielding, "A micro-GA embedded PSO feature selection approach to intelligent facial emotion recognition," *IEEE transactions on cybernetics,* vol. 47, no. 6, pp. 1496-1509, 2016.

[190] S. Agarwal, R. Rajesh, and P. Ranjan, "FRBPSO: a Fuzzy rule based binary PSO for feature selection," *Proceedings of the National Academy of Sciences, India Section A: Physical Sciences,* vol. 87, no. 2, pp. 221-233, 2017.

[191] P. Moradi and M. Gholampour, "A hybrid particle swarm optimization for feature subset selection by integrating a novel local search strategy," *Applied Soft Computing,* vol. 43, pp. 117-130, 2016.

[192] E.-G. Talbi, L. Jourdan, J. Garcia-Nieto, and E. Alba, "Comparison of population based metaheuristics for feature selection: Application to microarray data classification," in *2008 IEEE/ACS International Conference on Computer Systems and Applications*, 2008, pp. 45-52: IEEE.

[193] Z. Yong, G. Dun-wei, and Z. Wan-qiu, "Feature selection of unreliable data using an improved multi-objective PSO algorithm," *Neurocomputing,* vol. 171, pp. 1281-1290, 2016.

[194] J. Jona and N. Nagaveni, "A hybrid swarm optimization approach for feature set reduction in digital mammograms," *WSEAS Trans. Inf. Sci. Appl,* vol. 9, no. 11, pp. 340-349, 2012.

[195] Q. Al-Tashi, S. J. A. Kadir, H. M. Rais, S. Mirjalili, and H. Alhussian, "Binary optimization using hybrid grey wolf optimization for feature selection," *IEEE Access,* vol. 7, pp. 39496-39508, 2019.

[196] M. Amoozegar and B. Minaei-Bidgoli, "Optimizing multi-objective PSO based feature selection method using a feature elitism mechanism," *Expert Systems with Applications,* vol. 113, pp. 499-514, 2018.

[197] A. Peimankar, S. J. Weddell, T. Jalal, and A. C. Lapthorn, "Evolutionary multi-objective fault diagnosis of power transformers," *Swarm and Evolutionary Computation*, vol. 36, pp. 62-75, 2017.

[198] M. K. Sohrabi and A. Tajik, "Multi-objective feature selection for warfarin dose prediction," *Computational Biology and chemistry*, vol. 69, pp. 126-133, 2017.

[199] Y. Zhang, D.-w. Gong, X.-y. Sun, and Y.-n. Guo, "A PSO-based multi-objective multi-label feature selection method in classification," *Scientific reports*, vol. 7, no. 1, pp. 1-12, 2017.

[200] B. Xue, L. Cervante, L. Shang, W. N. Browne, and M. Zhang, "A multi-objective particle swarm optimisation for filter-based feature selection in classification problems," *Connection Science*, vol. 24, no. 2-3, pp. 91-116, 2012.

[201] B. Xue, M. Zhang, and W. N. Browne, "Particle swarm optimization for feature selection in classification: A multi-objective approach," *IEEE transactions on cybernetics*, vol. 43, no. 6, pp. 1656-1671, 2012.

[202] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength Pareto evolutionary algorithm," *TIK-report*, vol. 103, 2001.

[203] J. Knowles and D. Corne, "The pareto archived evolution strategy: A new baseline algorithm for pareto multiobjective optimisation," in *Proceedings of the 1999 Congress on Evolutionary Computation-CEC99 (Cat. No. 99TH8406)*, 1999, vol. 1, pp. 98-105: IEEE.

[204] B. Xia, T. Liu, T. Ding, and Z. J. W. P. C. Wang, "An Improved PSO Localization Algorithm for UWB Sensor Networks," vol. 117, no. 3, pp. 2207-2223, 2021.

[205] Y. Hu, Y. Ding, and K. Hao, "An Immune Cooperative Particle Swarm Optimization Algorithm for Fault-Tolerant Routing Optimization in Heterogeneous Wireless Sensor Networks," *Mathematical Problems in Engineering*, vol. 2012, p. 19, 2012, Art. no. 743728.

[206] Z. Zhi-hui, Z. Jun, D. Ke-jing, and X. Jing, "Extended Binary Particle Swarm Optimization Approach for Disjoint Set Covers Problem in Wireless Sensor Networks," in *Technologies and Applications of Artificial Intelligence (TAAI), 2012 Conference on*, 2012, pp. 327-331.

[207] Y.-m. ZHAO and A.-j. J. C. T. LU, "Application of Adaptive Chaos PSO Algorithm in WSN Coverage Optimization," p. 10, 2018.

[208] Z.-J. Teng, L.-Y. Xie, H.-L. Chen, and H. J. J. o. C. Zhang, "Application Research of Chaotic Binary Particle Swarm Optimization Algorithm in Dynamic Spectrum Allocation," vol. 31, no. 4, pp. 288-299, 2020.

[209] A. H. Mahdi, J. Mohanan, M. A. Kalil, and A. Mitschele-Thiel, "Adaptive Discrete Particle Swarm Optimization for Cognitive Radios," in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 6550-6554.

[210] J. Zhou, H. Qin, Y. Zhang, R. Yang, Y. Liu, and C. Li, "Binary Quantum Elite Particle Swarm Optimization Algorithm for Spectrum Allocation in Cognitive Wireless Medical Sensor Network," in *Journal of Physics: Conference Series*, 2021, vol. 1924, no. 1, p. 012030: IOP Publishing.

[211] Y. Yang, Q. Zhang, Y. Wang, T. Emoto, M. Akutagawa, and S. J. C. C. Konaka, "Adaptive resources allocation algorithm based on modified PSO for cognitive radio system," vol. 16, no. 5, pp. 83-92, 2019.

[212] A. A. El-Saleh, T. M. Shami, R. Nordin, M. Y. Alias, and I. Shayea, "Multi-objective optimization of joint power and admission control in cognitive radio networks using enhanced swarm intelligence," *Electronics*, vol. 10, no. 2, p. 189, 2021.

[213] V. D. P. Souto, R. D. Souza, B. F. Uchôa-Filho, A. Li, and Y. J. I. W. C. L. Li, "Beamforming optimization for intelligent reflecting surfaces without CSI," vol. 9, no. 9, pp. 1476-1480, 2020.

[214] X. Chen et al., "Energy-efficient offloading for DNN-based smart IoT systems in cloud-edge environments," vol. 33, no. 3, pp. 683-697, 2021.

[215] B. Lin, Y. Huang, J. Zhang, J. Hu, X. Chen, and J. J. I. T. o. I. I. Li, "Cost-driven off-loading for DNN-based applications over cloud, edge, and end devices," vol. 16, no. 8, pp. 5456-5466, 2019.

[216] B. Lin et al., "A time-driven data placement strategy for a scientific workflow combining edge computing and cloud computing," vol. 15, no. 7, pp. 4254-4265, 2019.

[217] J. S. Priya, M. Femina, and R. J. S. C. Samuel, "APSO-MVS: an adaptive particle swarm optimization incorporating multiple velocity strategies for optimal leader selection in hybrid MANETs," pp. 1-17, 2020.

[218] G. Husnain, S. Anwar, and F. Shahzad, "Performance evaluation of CLPSO and MOPSO routing algorithms for optimized clustering in Vehicular Ad hoc Networks," in 2017 14th International Bhurban Conference on Applied Sciences and Technology (IBCAST), 2017, pp. 772-778: IEEE.

[219] J. Toutouh and E. J. S. C. Alba, "Parallel multi-objective metaheuristics for smart communications in vehicular networks," vol. 21, no. 8, pp. 1949-1961, 2017.

[220] P. Lagos-Eulogio, J. C. Seck-Tuoh-Mora, N. Hernandez-Romero, and J. J. N. D. Medina-Marin, "A new design method for adaptive IIR system identification using hybrid CPSO and DE," vol. 88, no. 4, pp. 2371-2389, 2017.

[221] K. Yamamoto and K. Suyama, "Active enumeration of local minima for IIR Filter design using PSO," in 2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2017, pp. 910-917: IEEE.

[222] M. Dash, T. Panigrahi, and R. Sharma, "Robust Estimation of IIR System's Parameter Using Adaptive Particle Swarm Optimization Algorithm," in Computational Intelligence in Data Mining: Springer, 2019, pp. 41-50.

[223] X. Zhang, D. Lu, X. Zhang, Y. J. E. J. o. W. C. Wang, and Networking, "Antenna array design by a contraction adaptive particle swarm optimization algorithm," vol. 2019, no. 1, pp. 1-7, 2019.

[224] R. Bera, D. Mandal, R. Kar, S. P. J. C. Ghoshal, and E. Engineering, "Non-uniform single-ring antenna array design using wavelet mutation based novel particle swarm optimization technique," vol. 61, pp. 151-172, 2017.

[225] J. Liu, H. Miao, X. Yuan, and J. Shi, "Photonic Crystal Microstrip Antenna Array Design Using an Improved Boolean Particle Swarm Optimization," in Urban Intelligence and Applications: Springer, 2020, pp. 39-53.

[226] R. V. Kulkarni, G. K. J. I. T. o. S. Venayagamoorthy, Man,, and P. C. Cybernetics, "Particle swarm optimization in wireless-sensor networks: A brief survey," vol. 41, no. 2, pp. 262-267, 2010.

[227] J. Chen, S. Huang, H. Li, X. Lv, and Y. J. I. A. Cai, "PSO-based agent cooperative spectrum sensing in cognitive radio networks," vol. 7, pp. 142963-142973, 2019.

[228] M. Bany Taha, C. Talhi, H. Ould-Slimane, and S. J. T. o. E. T. T. Alrabaee, "TD-PSO: task distribution approach based on particle swarm optimization for vehicular ad hoc network," p. e3860, 2020.

[229] R. Sinha, A. Choubey, S. K. Mahto, and P. Ranjan, "Quantum behaved particle swarm optimization technique applied to FIR-based linear and nonlinear channel equalizer," in Advances in Computer Communication and Computational Sciences: Springer, 2019, pp. 37-50.

[230] Y. Zhao et al., "A Fast 2-D Otsu lung tissue image segmentation algorithm based on improved PSO," vol. 80, p. 103527, 2021.

[231] S. Mahalakshmi, T. J. I. J. o. S. Velmurugan, and Technology, "Detection of brain tumor by particle swarm optimization using image segmentation," vol. 8, no. 22, p. 1, 2015.

[232] T. R. Farshi and A. K. J. M. S. Ardabili, "A hybrid firefly and particle swarm optimization algorithm applied to multilevel image thresholding," vol. 27, no. 1, pp. 125-142, 2021.

[233] S. K. Ghosh, B. Biswas, and A. J. P. C. S. Ghosh, "A novel approach of retinal image enhancement using PSO system and measure of fuzziness," vol. 167, pp. 1300-1311, 2020.

[234] M. M. S. Rani, P. J. I. J. o. P. Chitra, and A. Mathematics, "A hybrid medical image coding method based on haar wavelet transform and particle swarm optimization technique," vol. 118, no. 8, pp. 3056-3067, 2018.

[235] X. Kang, Y. Chen, F. Zhao, and G. J. S. C. Lin, "Multi-dimensional particle swarm optimization for robust blind image watermarking using intertwining logistic map and hybrid domain," vol. 24, no. 14, pp. 10561-10584, 2020.

[236] X.-S. Yang, "Firefly algorithms for multimodal optimization," in International symposium on stochastic algorithms, 2009, pp. 169-178: Springer.

[237] Z. Xin-gang, L. Ji, M. Jin, and Z. J. E. S. w. A. Ying, "An improved quantum particle swarm optimization algorithm for environmental economic dispatch," vol. 152, p. 113370, 2020.

[238] M. Mohammadian, A. Lorestani, and M. J. E. Ardehali, "Optimization of single and multi-areas economic dispatch problems based on

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2022.3142859, IEEE Access

evolutionary particle swarm optimization algorithm," vol. 161, pp. 710-724, 2018.

[239] W. T. Elsayed, Y. G. Hegazy, M. S. El-bages, and F. M. J. I. T. o. I. I. Bendary, "Improved random drift particle swarm optimization with self-adaptive mechanism for solving the power economic dispatch problem," vol. 13, no. 3, pp. 1017-1026, 2017.

[240] U. Khaled, A. M. Eltamaly, and A. J. E. Beroual, "Optimal power flow using particle swarm optimization of renewable hybrid distributed generation," vol. 10, no. 7, p. 1013, 2017.

[241] S. Khunkitti, A. Siritaratiwat, S. Premrudeepreechacharn, R. Chatthaworn, and N. R. J. E. Watson, "A hybrid DA-PSO optimization algorithm for multiobjective optimal power flow problems," vol. 11, no. 9, p. 2270, 2018.

[242] K. Teeparthi, D. V. J. E. S. Kumar, and a. I. J. Technology, "Multi-objective hybrid PSO-APO algorithm based security constrained optimal power flow with wind and thermal generators," vol. 20, no. 2, pp. 411-426, 2017.

[243] Z. Ullah, M. R. Elkadeem, S. Wang, and J. J. I. A. Radosavljević, "A novel PSOS-CGSA method for state estimation in unbalanced DG-integrated distribution systems," vol. 8, pp. 113219-113229, 2020.

[244] R. K. Sahu, S. Panda, G. C. J. I. J. o. E. P. Sekhar, and E. Systems, "A novel hybrid PSO-PS optimized fuzzy PI controller for AGC in multi area interconnected power systems," vol. 64, pp. 880-893, 2015.

[245] J. Zhang and Y. Jiang, "Wide-area Power System Damping Control Coordination Based on Particle Swarm Optimization with Time Delay Considered," in IOP Conference Series: Earth and Environmental Science, 2017, vol. 86, no. 1, p. 012029: IOP Publishing.

[246] G. Shaari, N. Tekbiyik-Ersoy, and M. J. P. Dagbasi, "The state of art in particle swarm optimization based unit commitment: A review," vol. 7, no. 10, p. 733, 2019.

[247] S. Sharma, S. J. J. o. A. I. Ghosh, and H. Computing, "FIS and hybrid ABC-PSO based optimal capacitor placement and sizing for radial distribution networks," vol. 11, no. 2, pp. 901-916, 2020.

[248] B. Yang, Y. Chen, and Z. Zhao, "Survey on applications of particle swarm optimization in electric power systems," in 2007 IEEE international conference on control and automation, 2007, pp. 481-486: IEEE.

[249] G. Abbas, J. Gu, U. Farooq, M. U. Asad, and M. J. I. A. El-Hawary, "Solution of an economic dispatch problem through particle swarm optimization: a detailed survey-part I," vol. 5, pp. 15105-15141, 2017.

[250] G. Abbas, J. Gu, U. Farooq, A. Raza, M. U. Asad, and M. E. J. I. a. El-Hawary, "Solution of an economic dispatch problem through particle swarm optimization: A detailed survey–Part II," vol. 5, pp. 24426-24445, 2017.

[251] P. Shao, Z. Wu, X. Zhou, and D. C. J. S. C. Tran, "FIR digital filter design using improved particle swarm optimization based on refraction principle," vol. 21, no. 10, pp. 2631-2642, 2017.

[252] A. Ghosh, A. Ghosh, A. Chowdhury, A. Konar, K. Eunjin, and A. K. Nagar, "Linear phase low pass FIR filter design using Genetic Particle Swarm Optimization with dynamically varying neighbourhood technique," in Evolutionary Computation (CEC), 2012 IEEE Congress on, 2012, pp. 1-7.

[253] K. Basu and S. Nanda, "An adaptive filtering technique with self-adaptive PSO for estimation of non-stationary signals," in 2016 International Conference on Communication and Signal Processing (ICCSP), 2016, pp. 1057-1061: IEEE.

[254] R. Malik, R. Dhir, S. J. J. o. A. I. Mittal, and H. Computing, "Remote sensing and landsat image enhancement using multiobjective PSO based local detail enhancement," vol. 10, no. 9, pp. 3563-3571, 2019.

[255] G. R. Mamta and M. Dutta, "PSO based blind deconvolution technique of image restoration using cepstrum domain of motion blur," in Computational Vision and Bio Inspired Computing: Springer, 2018, pp. 947-958.

[256] S. Rengasamy, P. J. S. Murugesan, and E. Computation, "PSO based data clustering with a different perception," vol. 64, p. 100895, 2021.

[257] M. Sharma, J. K. J. S. C. I. Chhabra, and Systems, "Sustainable automatic data clustering using hybrid PSO algorithm with mutation," vol. 23, pp. 144-157, 2019.

[258] Y. Gupta and A. J. S. C. Saini, "A new swarm-based efficient data clustering approach using KHM and fuzzy logic," vol. 23, no. 1, pp. 145-162, 2019.

[259] R. L. Patibandla, B. T. Rao, P. S. Krishna, and V. R. J. J. o. C. R. Maddumala, "Medical data clustering using particle swarm optimization method," vol. 7, no. 6, pp. 363-367, 2020.

[260] T. Y. Tan, L. Zhang, C. P. Lim, B. Fielding, Y. Yu, and E. J. I. a. Anderson, "Evolving ensemble models for image segmentation using enhanced particle swarm optimization," vol. 7, pp. 34004-34019, 2019.

[261] Z. Zheng, N. Saxena, K. Mishra, and A. K. J. F. G. C. S. Sangaiah, "Guided dynamic particle swarm optimization for optimizing digital image watermarking in industry applications," vol. 88, pp. 92-106, 2018.

[262] B. Xia, Z. Ren, and C. S. J. I. t. o. M. Koh, "A novel reliability-based optimal design of electromagnetic devices based on adaptive dynamic Taylor Kriging," vol. 53, no. 6, pp. 1-4, 2017.

[263] [263]O. U. Rehman, S. Tu, S. Khan, H. Khan, S. J. I. J. o. A. E. Yang, and Mechanics, "A modified quantum particle swarm optimizer applied to optimization design of electromagnetic devices," vol. 58, no. 3, pp. 347-357, 2018.

[264] X. Chen, B. Xu, and W. Du, "An Improved Particle Swarm Optimization with Biogeography-Based Learning Strategy for Economic Dispatch Problems," Complexity, vol. 2018, p. 7289674, 2018/07/12 2018.

[265] T. Liu, L. Jiao, W. Ma, J. Ma, and R. Shang, "Cultural quantum-behaved particle swarm optimization for environmental/economic dispatch," Applied Soft Computing, vol. 48, pp. 597-611, 2016/11/01/ 2016.

[266] A. Goudarzi, Y. Li, and J. J. A. S. C. Xiang, "A hybrid non-linear time-varying double-weighted particle swarm optimization for solving non-convex combined environmental economic dispatch problem," vol. 86, p. 105894, 2020.

[267] F. Berrouk, K. J. J. o. C. Bounaya, Automation, and E. Systems, "Optimal power flow for multi-FACTS power system using hybrid PSO-PS algorithms," vol. 29, no. 2, pp. 177-191, 2018.

[268] A. Man-Im, W. Ongsakul, J. Singh, and M. N. J. E. E. Madhu, "Multi-objective optimal power flow considering wind power cost functions using enhanced PSO with chaotic mutation and stochastic weights," vol. 101, no. 3, pp. 699-718, 2019.

[269] M. He, S. Wang, C. Fernandez, C. Yu, X. Li, and E. D. J. I. J. o. E. S. Bobobee, "A novel adaptive particle swarm optimization algorithm based high precision parameter identification and state estimation of lithium-ion battery," vol. 16, no. 5, 2021.

[270] X. Yang, Y. Chen, B. Li, and D. J. E. Luo, "Battery states online estimation based on exponential decay particle swarm optimization and proportional-integral observer with a hybrid battery model," vol. 191, p. 116509, 2020.

[271] Y. Wang, G. Gao, X. Li, and Z. J. J. o. P. S. Chen, "A fractional-order model-based state estimation approach for lithium-ion battery and ultra-capacitor hybrid power source system considering load trajectory," vol. 449, p. 227543, 2020.

[272] K. Balu, V. J. E. P. C. Mukherjee, and Systems, "Siting and Sizing of Distributed Generation and Shunt Capacitor Banks in Radial Distribution System Using Constriction Factor Particle Swarm Optimization," vol. 48, no. 6-7, pp. 697-710, 2020.

[273] R. Krishnasamy, R. Aathi, and P. Jeyabalan, "Application of Comprehensive Learning Particle Swarm Optimization to Least Cost Generation Expansion Planning Problem with Solar Plant," in 2019 IEEE International Conference on Clean Energy and Energy Efficient Electronics Circuit for Sustainable Development (INCCES), 2019, pp. 1-5: IEEE.

[274] H. Doagou-Mojarrad, H. Rastegar, G. B. J. J. o. I. Gharehpetian, and F. Systems, "Probabilistic interactive fuzzy satisfying generation and transmission expansion planning using fuzzy adaptive chaotic binary PSO algorithm," vol. 30, no. 3, pp. 1629-1641, 2016.

[275] S. Rastgoufard and D. Charalampidis, "Tuned support vector regression by modified particle swarm optimization for online power system static security evaluation," in 2018 IEEE Texas Power and Energy Conference (TPEC), 2018, pp. 1-6: IEEE.

[276] N. Kumar and V. Mahajan, "Reconfiguration of distribution network for power loss minimization & reliability improvement using binary particle swarm optimization," in 2018 IEEE 8th Power India International Conference (PIICON), 2018, pp. 1-6: IEEE.

[277] J. Li, H. Huang, B. Lou, Y. Peng, Q. Huang, and K. Xia, "Wind farm reactive power and voltage control strategy based on adaptive discrete binary particle swarm optimization algorithm," in 2019 IEEE Asia Power and Energy Engineering Conference (APEEC), 2019, pp. 99-102: IEEE.

[278] W. Moondee and W. Srirattanawichaikul, "Reactive Power Management of MV Distribution Grid with Inverter-based PV Distributed Generations using PSO Algorithm," in IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, 2019, vol. 1, pp. 2239-2244: IEEE.

[279] T. M. Aljohani, A. F. Ebrahim, and O. J. E. Mohammed, "Single and multiobjective optimal reactive power dispatch based on hybrid artificial physics–particle swarm optimization," vol. 12, no. 12, p. 2333, 2019.

[280] S. Kanata, G. H. Sianipar, and N. U. Maulidevi, "Optimization of reactive power and voltage control in power system using hybrid artificial neural network and particle swarm optimization," in 2018 2nd International Conference on Applied Electromagnetic Technology (AEMT), 2018, pp. 67-72: IEEE.

[281] N. Zeng, H. Zhang, W. Liu, J. Liang, and F. E. J. N. Alsaadi, "A switching delayed PSO optimized extreme learning machine for short-term load forecasting," vol. 240, pp. 175-182, 2017.

[282] Y. K. Semero, J. Zhang, D. J. I. G. Zheng, Transmission, and Distribution, "EMD–PSO–ANFIS-based hybrid approach for short-term load forecasting in microgrids," vol. 14, no. 3, pp. 470-475, 2020.

[283] X. Tao, W. Guo, Q. Li, C. Ren, and R. J. A. S. C. Liu, "Multiple scale self-adaptive cooperation mutation strategy-based particle swarm optimization," vol. 89, p. 106124, 2020.

[284] K. Chaitanya, D. N. Somayajulu, and P. R. J. A. I. Krishna, "Memory-based approaches for eliminating premature convergence in particle swarm optimization," pp. 1-34, 2021.

[285] T. Krink, J. S. Vesterstrom, and J. Riget, "Particle swarm optimisation with spatial particle extension," in Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on, 2002, vol. 2, pp. 1474-1479.

[286] A. Faramarzi, M. Heidarinejad, B. Stephens, and S. Mirjalili, "Equilibrium optimizer: A novel optimization algorithm," Knowledge-Based Systems, vol. 191, p. 105190, 2020.

[287] A. Faramarzi, M. Heidarinejad, S. Mirjalili, and A. H. Gandomi, "Marine Predators Algorithm: A nature-inspired metaheuristic," Expert Systems with Applications, vol. 152, p. 113377, 2020.

[288] I. Ahmadianfar, O. Bozorg-Haddad, and X. Chu, "Gradient-based optimizer: A new metaheuristic optimization algorithm," Information Sciences, vol. 540, pp. 131-159, 2020.

[289] Q. Askari, I. Younas, and M. Saeed, "Political Optimizer: A novel socio-inspired meta-heuristic for global optimization," Knowledge-Based Systems, vol. 195, p. 105709, 2020.

[290] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz, and A. H. Gandomi, "The arithmetic optimization algorithm," Computer methods in applied mechanics and engineering, vol. 376, p. 113609, 2021.

[291] F. A. Hashim, K. Hussain, E. H. Houssein, M. S. Mabrouk, and W. Al-Atabany, "Archimedes optimization algorithm: a new metaheuristic algorithm for solving optimization problems," Applied Intelligence, vol. 51, no. 3, pp. 1531-1551, 2021.

[292] M. Alswaitti, M. Albughdadi, and N. A. M. Isa, "Density-based particle swarm optimization algorithm for data clustering," Expert Systems with Applications, vol. 91, pp. 170-186, 2018.

[293] C. Duan, C. Deng, A. Gharaei, J. Wu, and B. Wang, "Selective maintenance scheduling under stochastic maintenance quality with multiple maintenance actions," International Journal of Production Research, vol. 56, no. 23, pp. 7160-7178, 2018.

[294] A. Gharaei, S. A. Hoseini Shekarabi, and M. Karimi, "Modelling and optimal lot-sizing of the replenishments in constrained, multi-product and bi-objective EPQ models with defective products: Generalised cross decomposition," International Journal of Systems Science: Operations & Logistics, vol. 7, no. 3, pp. 262-274, 2020.

[295] A. Gharaei, M. Karimi, and S. A. Hoseini Shekarabi, "Joint economic lot-sizing in multi-product multi-level integrated supply chains: Generalized benders decomposition," International Journal of Systems Science: Operations & Logistics, vol. 7, no. 4, pp. 309-325, 2020.

[296] M. Rabbani, N. Foroozesh, S. M. Mousavi, and H. Farrokhi-Asl, "Sustainable supplier selection by a new decision model based on interval-valued fuzzy sets and possibilistic statistical reference point systems under uncertainty," International Journal of Systems Science: Operations & Logistics, vol. 6, no. 2, pp. 162-178, 2019.

[297] N. H. Shah, U. Chaudhari, and L. E. Cárdenas-Barrón, "Integrating credit and replenishment policies for deteriorating items under quadratic demand in a three echelon supply chain," International Journal of Systems Science: Operations & Logistics, vol. 7, no. 1, pp. 34-45, 2020.

TAREQ M. SHAMI received the B.Eng. (Hons.) degree and the M.Sc.Eng. degree in electronics majoring in telecommunications from Multimedia University, Melaka and Cyberjaya, Malaysia, in 2012 and 2017, respectively, his Ph.D. in Electronic Engineering from University of York in 2021. His Ph.D. research was part of the European-funded Marie Curie ITN-5GAuRA project. His current research interests include 5G small cell networks, radio resource management, cell-less architectures, optimization algorithms, multi-objective optimization, and feature selection.

AYMAN A. EL-SALEH (M’13-SM’17) received his B.Sc. degree in Communications Engineering from Omar El-Mukhtar University (OMU), Libya, in 1999, his M.Sc. in Microelectronics Engineering and his Ph.D. in Wireless Communications both from Universiti Kebangsaan Malaysia (UKM), in 2006 and 2012, respectively. He joined the Faculty of Engineering at Multimedia University (MMU) in October 2006 at which he was a Senior Lecturer and the Chairperson of Centre for Wireless Technology (CWT). In April 2017, he was appointed as the Lead of Connectivity Research Theme at MMU. He joined A’Sharqiyah University in Oman in September 2017 at which he is currently an Associate Professor. He is also a Senior Member of IEEE, and was an Executive Committee Member of IEEE Malaysia ComSoc/VTS Joint Chapter in 2015 and 2016. He is a reviewer of several indexed journals and published more than 70 journal and conference articles. His research interests include cognitive radio networks, heterogeneous LTE/LTE-Advanced cellular networks, and applications of artificial intelligence and evolutionary algorithms in wireless communications.

Mohammed Alswaitti received the B.Eng. degree in Computer Engineering from the Islamic University of Gaza (IUG), Palestine, in 2010, and the M.Sc. degree in Electronic Systems Design Engineering from the Universiti Sains Malaysia (USM) in 2011. He worked as a lecturer in Software Engineering and Information Technology Faculties at the University of Palestine and as an Instructor at the Educational Technology Centre at IUG. He pursued his PhD degree in Computational Intelligence under the Malaysian International Scholarship (MIS) scheme and worked as a Graduate Assistant at the Electrical and Electronic Engineering Department, USM, Malaysia. Currently, he is an Assistant Professor at the School of Electrical and Computer Engineering, Xiamen University Malaysia (XMUM). He acts as the Research Coordinator for the school where his research is focused on nature-inspired optimization-based clustering techniques, machine/deep learning applications, and pattern recognition. Recently, Alswaitti managed to publish several articles in the top tier journals in AI field through securing research funds and collaborations with research institutions all over the world. Besides, he is a regular keynote speaker at academic and industry conferences and workshops and does voluntary work as a reviewer and associate editor for journals and language editing services.

QASEM AL-TASHI is currently a Postdoctoral Fellow at The University of Texas MD Anderson Cancer Center, Houston, Texas, United States. He was a Research Scientist at Universiti Teknologi PETRONAS. He received the B.Sc. degree in software engineering from Universiti Teknologi Malaysia, in 2012, and the M.Sc. degree in software engineering from Universiti Kebangsaan Malaysia, in 2017. He obtained his PhD in Information Technology in early 2021 from Universiti Teknologi PETRONAS. He is also an Academic Staff with Albaydha University, Yemen. His research interests include artificial neural networks, multi-objective optimization, feature selection, swarm intelligence evolutionary algorithms, classification and data analytics. He is a section editor for the Journal of Applied Artificial Intelligence (JAAI) and Journal of Information Technology and Computing (JITC) and a reviewer for several high impact factor journals such as Artificial Intelligence Review, the IEEE ACCESS, Knowledge-Based Systems, Soft Computing, Journal of Ambient Intelligence and Humanized Computing, Applied Soft Computing, Neurocomputing, Applied Artificial Intelligence, and Plos One.

MHD AMEN SUMMAKIEH received his B.Eng. (Hons.) in Communication and Electronics Engineering from UCSI University, Malaysia and his M.Eng.Sc from Multimedia University, Malaysia in 2016 and 2020 respectively. His research interests include heterogeneous LTE-Advanced cellular networks, user association, metaheuristic algorithms and antennas design.

SEYEDALI (ALI) MIRJALILI (Senior Member, IEEE) is currently the Director of the Centre for Artificial Intelligence Research and Optimization, Torrens University Australia, Brisbane. He is internationally recognized for his advances in swarm intelligence and optimization, including the first set of algorithms from a synthetic intelligence standpoint—a radical departure from how natural systems are typically understood—and a systematic design framework to reliably benchmark, evaluate, and propose computationally cheap robust optimization algorithms. He has published over 250 publications with over 31900 citations and an H-index of 62. As the most cited researcher in robust optimization, he is in the list of 1% highly cited researchers and named one of the world’s most influential researchers by the Web of Science. He is working on the applications of multi-objective and robust meta-heuristic optimization techniques as well. His research interests include robust optimization, engineering optimization, multi-objective optimization, swarm intelligence evolutionary algorithms, and artificial neural networks. He is an Associate Editor of several journals, including Neurocomputing, Applied Soft Computing, Advances in Engineering Software, Applied Intelligence, and IEEE ACCESS