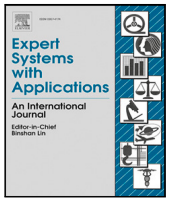
Expert Systems With Applications 225 (2023) 120069

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa



Snow ablation optimizer: A novel metaheuristic technique for numerical optimization and engineering design

Lingyun Deng *, Sanyang Liu

School of Mathematics and Statistics, Xidian University, Xi'an, 710126, China

ARTICLE INFO

Keywords:
Snow ablation optimizer
Novel metaheuristic algorithm
Premature convergence
Benchmark
Engineering design

ABSTRACT

This paper develops a novel nature-inspired metaheuristic technique named snow ablation optimizer (SAO) for numerical optimization and engineering design. The SAO algorithm mainly emulates the sublimation and melting behavior of snow to realize a tradeoff between exploitation and exploration in the solution space and discourage premature convergence. The competitiveness and effectiveness of SAO are validated utilizing 29 typical CEC2017 unconstrained benchmarks and 22 CEC2020 real-world constrained optimization issues which consist of 7 process synthesis and design issues and 15 mechanical engineering issues. Additionally, to further verify its strength, the developed SAO is applied to extract the core parameters in photovoltaic systems. The simulation outcomes have demonstrated that the developed SAO is a very promising technique that can yield better performance than other state-of-the-art rival methods. The source code of SAO is publicly available at https://github.com/denglingyun123/SAO-snow-ablation-optimizer.

1. Introduction

In the past two decades, metaheuristic algorithms (MAs) have been broadly utilized to handle many complicated engineering problems in various scientific fields. Especially when handling nonconvex, highly nonlinear, nonsmooth, and even dynamic real-world problems, compared with traditional mathematical optimization approaches, MAs are more general because of their gradient-free characteristics and simple structures (Li, Liu, & Yang, 2020; Su et al., 2023). These algorithms start the search process in the solution space according to the predefined rules. On the basis of the rules that the algorithm follows, MAs are mainly classified into two types: evolutionary algorithms and swarm intelligence algorithms, as depicted in Fig. 1.

As the first category, evolutionary algorithms emulate the competition rules of survival of the fittest in nature. Such algorithms mainly employ the crossover strategy and mutation operator to reflect the evolution of species. Genetic algorithm (GA) (Srinivas & Patnaik, 1994) is a representative evolutionary computation technique and has been utilized to effectively address numerous challenging problems in academia and industry. The inspiration of GA originates from the perspective of Darwinism. In addition, typical methods of this category contain evolution strategy (Hansen et al., 2003), differential evolution (Storn & Price, 1997), and evolutionary programming (Yao et al., 1999).

Another category is the swarm intelligence algorithm. Note that the characteristics of most swarm intelligence optimizers are mainly divided into the following four parts: bio-inspired, human-based, mathematics-based, and physics-based. Among them, bio-inspired algorithms mimic the biological behavior of fishes, insects, and other creatures. Recent and influential algorithms in the first group are particle swarm optimization (PSO) (Kennedy & Eberhart, 1995), grey wolf optimizer (GWO) (Sm et al., 2014), butterfly optimization algorithm (BOA) (Arora & Singh, 2018), and slime mould algorithm (SMA) (Deng & Liu, 2023; Li, Chen, et al., 2020). Besides, human-based methods typified by the teaching–learning-based optimizer (TLBO) (Rao et al., 2012) emulate the laws of human activities in communities, mathematics-based methods represented by sine cosine algorithm (SCA) (Mirjalili, 2016) and arithmetic optimization algorithm (AOA) (Abualigah, Diabat, et al., 2021) imitate the laws of mathematics in real life, and physics-based methods such as multi-verse optimizer (MVO) (Mirjalili et al., 2015), equilibrium optimizer (EO) (Faramarzi, Heidarinejad, Stephens, & Mirjalili, 2020), and simulated annealing (SA) (Ingber, 1993) simulate the laws of physics.

As the no-free-lunch theorem (NFL) (Wolpert & Macready, 1997) suggests, although MAs are broadly used, they cannot always yield excellent performance on all optimization problems. Therefore, researchers are motivated to either put forward novel methods or reinforce the convergence performance of existing algorithms by utilizing some effective improvements. The ultimate goal is to realize a tradeoff between exploitation and exploration (Hashim et al., 2019). Actually, in the field of metaheuristic algorithms, the exploration capability represents the property of the algorithm to expand and discover promising

* Corresponding author.
E-mail addresses: lingyundeng@stu.xidian.edu.cn (L. Deng), syliu@xidian.edu.cn (S. Liu).

https://doi.org/10.1016/j.eswa.2023.120069
Received 7 March 2023; Received in revised form 24 March 2023; Accepted 4 April 2023
Available online 12 April 2023

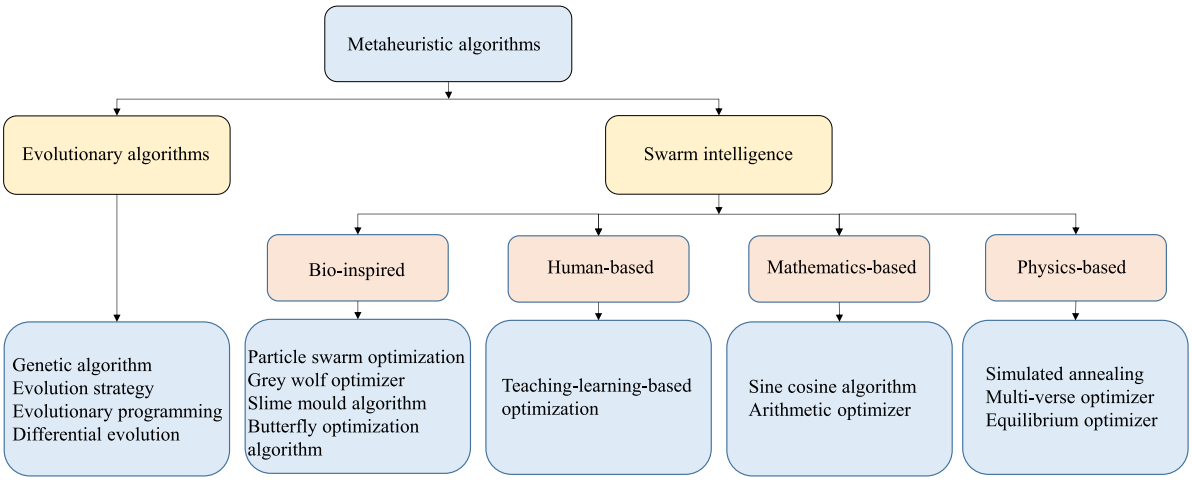


Fig. 1. Classification of MAs.

regions in the solution space. When applying MAs to cope with complicated multimodal problems, the exploration capability is supposed to be preserved to encourage the search agents to jump out from numerous local optima (Liang et al., 2006). Meanwhile, the exploitation capability stands for the capability to converge towards the global optimal solution with rapid convergence. Nevertheless, the major deficiency of most MAs is still the imbalance between exploration and exploitation that limits the performance of algorithms seriously. Hence, this article develops a novel physics-based metaheuristic technique named snow ablation optimizer (SAO) with the aiming of balancing exploration and exploitation.

1.1. Motivation

As one of the most powerful tools to cope with real-world optimization issues, MAs have attracted a high level of interest. Based on the above description, numerous researchers have worked on strengthening their performances through different approaches, thereby deriving a substantial number of interesting algorithms. Nevertheless, with more and more optimization problems becoming complicated, better optimization techniques are always needed. In addition, the usefulness and effectiveness of MAs encourage us to develop a novel metaheuristic technique. Furthermore, the SAO algorithm is devised to overcome the weaknesses of most MAs and achieve a tradeoff between the swarm's diversity and convergence eventually.

1.2. Contribution

Inspired by the sublimation and melting behavior of snow in nature, we propose a novel metaheuristic technique named snow ablation optimizer (SAO) for handling numerical optimization and engineering design issues. The major contributions are summarized in the following:

- A novel physics-based algorithm named snow ablation optimizer (SAO) that emulates the sublimation and melting behavior of snow is developed.
- 29 representative CEC2017 unconstrained benchmarks and 22 CEC2020 real-world constrained optimization issues are employed to verify the strength of the developed technique and simulation outcomes are contrasted with some well-regarded methods.
- Statistical analyses such as Wilcoxon's test and Friedman's test are utilized to investigate the strength of the raised SAO algorithm.
- The analysis of exploration and exploitation in SAO reveals the reason behind the great performance of the developed technique.
- The application of SAO to extract core parameters of photovoltaic systems also verifies the superiority of SAO over other rival algorithms.

The remainder of this article is arranged as follows. Section 2 describes the snow ablation optimizer in detail. Section 3 presents and analyzes the simulation outcomes. Conclusions and some future works are shown in Section 4.

2. Snow ablation optimizer (SAO)

In this part, the source of inspiration for SAO, which is on the basis of the sublimation and melting behavior of snow, is given. After that, the mathematical model of this algorithm is presented. Finally, we give the pseudo-code of SAO and analyze its time complexity.

2.1. Inspiration

Snow is one of the most fascinating and beautiful landscapes in nature. Especially in winter, the ablation of snow plays a significant role in the ecosystem, which affects the growth and development of crops and human health (Edwards et al., 2007). In physics, snow can transform into two forms: liquid water and steam. Corresponding to these two forms are two physical processes: melting and sublimation. As depicted in Fig. 2, in the process of melting, snow converts into liquid water, whereas it can straightly convert into steam through sublimation. Meanwhile, note that the liquid water converted from snow melting can also be turned into steam by evaporating.

The source of inspiration for snow ablation optimizer (SAO) arises from the sublimation and melting behavior of snow. In the following parts, the initialization stage, exploration stage, exploitation stage, and dual-population mechanism in the SAO algorithm will be introduced.

2.2. Initialization stage

In SAO, the procedure of iteration starts with a randomly produced swarm. As given in Eq. (1), the whole swarm is usually modeled as a matrix with N rows and Dim columns, where N represents the swarm's size, and Dim denotes the dimensionality of the solution space.

$$Z = L + \theta \times (U - L) = \begin{bmatrix} z_{1,1} & z_{1,2} & \cdots & z_{1,Dim-1} & z_{1,Dim} \\ z_{2,1} & z_{2,2} & \cdots & z_{2,Dim-1} & z_{2,Dim} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ z_{N-1,1} & z_{N-1,2} & \cdots & z_{N-1,Dim-1} & z_{N-1,Dim} \\ z_{N,1} & z_{N,2} & \cdots & z_{N,Dim-1} & z_{N,Dim} \end{bmatrix}_{N \times Dim}, \quad (1)$$

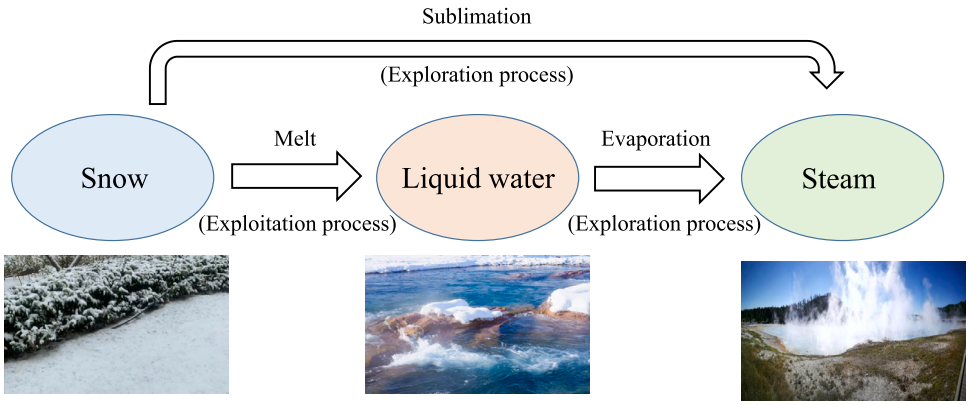
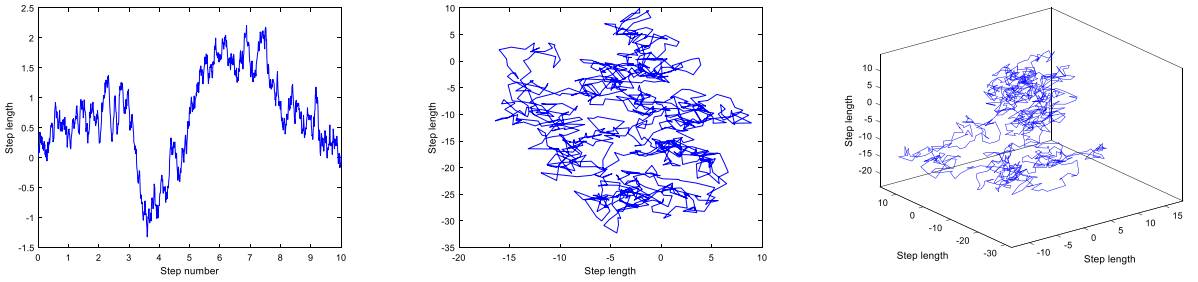


Fig. 2. Schematic diagram of inspiration source.



(a) 1-Dim Brownian distribution (b) 2-Dim Brownian trajectory (c) 3-Dim Brownian trajectory

Fig. 3. Brownian motion.

Among them, L and U indicate the lower bound and upper bound of the solution space, respectively. θ represents a number randomly produced in [0,1].

2.3. Exploration stage

In this part, the exploration strategy in SAO is described in detail. When the snow or the liquid water transformed from snow converts into steam, the search agents present a high-decentralized feature due to the irregular movement. In this study, Brownian motion is utilized to model this situation. As a stochastic process, Brownian motion is extensively applied to emulate the foraging behavior of animals (Faramarzi, Heidarinejad, Mirjalili, & Gandomi, 2020), the endless and irregular movement of particles (Abdel-Basset et al., 2018), the fluctuating behavior of the stock price (Merton, 1976), etc. For standard Brownian motion, the step size is attained by the probability density function based on the normal distribution with mean zero and variance one. The associated mathematical representation is shown in the following (Faramarzi, Heidarinejad, Mirjalili, & Gandomi, 2020):

$$f_{BM}(x;0,1) = \frac{1}{\sqrt{2\pi}} \times \exp(-\frac{x^2}{2}), \tag{2}$$

The 1-dimensional distribution of Brownian motion and the trajectory in 2- and 3-dimensional search spaces are shown in Fig. 3. As described in Fig. 3, utilizing dynamic and uniform step length, the Brownian motion enables some potential regions in the search space to be explored. Hence, it can well reflect the situation of the steam spreading out in the search space. The formula to calculate positions in the process of exploration is presented in the following:

$$Z_i(t+1) = Elite(t) + BM_i(t) \otimes (\theta_1 \times (G(t) - Z_i(t)) + (1-\theta_1) \times (\bar{Z}(t) - Z_i(t))), \tag{3}$$

Among them, $Z_i(t)$ denotes the ith individual during the tth iteration, $BM_i(t)$ indicates a vector including random numbers on the basis of Gaussian distribution denoting the Brownian motion, the sign $\otimes$ represents entry-wise multiplications, θ_1 indicates a number randomly chosen from [0,1]. Furthermore, $G(t)$ refers to the current best solution, $Elite(t)$ is a random individual selected from a set of several elites in the swarm, and $\bar{Z}(t)$ denotes the centroid position of the whole swarm. The corresponding mathematical expressions are presented in the following:

$$\bar{Z}(t) = \frac{1}{N} \sum_{i=1}^{N} Z_i(t), \tag{4}$$

$$Elite(t) \in [G(t), Z_{second}(t), Z_{third}(t), Z_c(t)], \tag{5}$$

where $Z_{second}(t)$ and $Z_{third}(t)$ represent the second-best individual and the third-best individual in the current population, respectively. $Z_c(t)$ denotes the centroid position of individuals whose fitness values ranked in the top 50%. In this study, for simplicity, the individuals whose fitness values ranked in the top 50% are named leaders. Additionally, $Z_c(t)$ is calculated utilizing the mathematical expression in Eq. (6).

$$Z_c(t) = \frac{1}{N_1} \sum_{i=1}^{N_1} Z_i(t), \tag{6}$$

where N_1 indicates the number of leaders, that is, N_1 is equal to half the size of the whole swarm, and $Z_i(t)$ represents the ith best leader. Therefore, during each iteration, the $Elite(t)$ is randomly selected from a set that consists of the current best solution, second-best individual, third-best individual, and centroid position of leaders.

Fig. 4 visually depicts the cross terms - $\theta_1 \times (G(t) - Z_i(t))$ and $(1-\theta_1) \times (\bar{Z}(t) - Z_i(t))$ in 2-dimensional parametric space. The parameter θ_1 is responsible for controlling the movement towards the current best individual and the movement to the centroid position of leaders. The integration of the above two cross terms is mainly utilized to reflect the interaction between individuals.

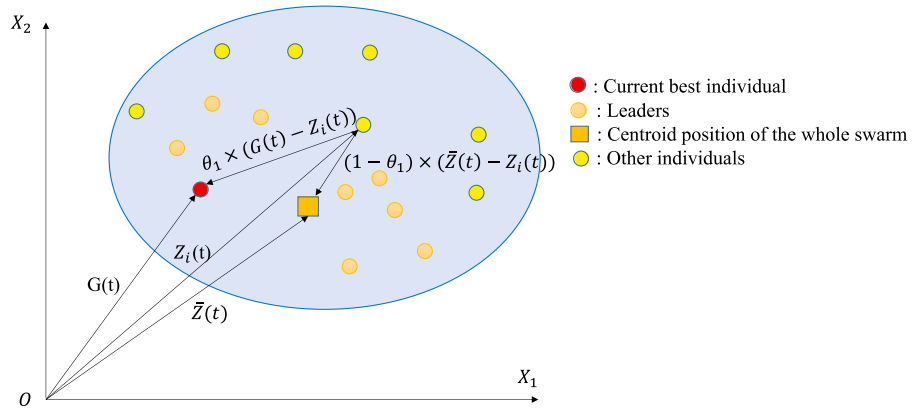


Fig. 4. Schematic diagram of the cross term in SAO.

2.4. Exploitation stage

In this part, the exploitative characteristic of SAO is introduced. Instead of expanding with a high-decentralized feature in the solution space, search agents are encouraged to exploit high-quality solutions around the current best solution when the snow converts into liquid water by melting behavior. As one of the most classical snowmelt models, the degree-day method (Zhou et al., 2021) is utilized to reflect the process of snow melting. The general form of this method is presented in the following:

$$M = DDF \times (T - T_1), \tag{7}$$

Among them, M denotes the snowmelt rate, a key parameter to emulate the melting behavior in the exploitation stage. T represents the average daily temperature. T_1 refers to the base temperature that is usually set to 0 (Zhou et al., 2021). This results in:

$$M = DDF \times T, \tag{8}$$

where DDF indicates the degree-day factor that ranges from 0.35 to 0.6 (Martinec & Rango, 1986). In each iteration, the mathematical expression to update the value of DDF is presented in the following:

$$DDF = 0.35 + 0.25 \times \frac{e^{\frac{t}{t_{\max}}} - 1}{e - 1}, \tag{9}$$

where t_{max} represents the termination criterion. The trend of DDF is depicted in Fig. 5.

Then in SAO, the snowmelt rate is calculated utilizing the following equation:

$$M = (0.35 + 0.25 \times \frac{e^{\frac{t}{t_{\max}}} - 1}{e - 1}) \times T(t), T(t) = e^{\frac{-t}{t_{\max}}}, \tag{10}$$

Then in the exploitation stage of SAO, the position updating equation is presented in the following:

$$Z_i(t+1) = M \times G(t) + BM_i(t) \otimes (\theta_2 \times (G(t) - Z_i(t)) + (1-\theta_2) \times (\bar{Z}(t) - Z_i(t))), \tag{11}$$

where M is the snowmelt rate, θ_2 indicates the random number chosen from [−1,1]. This parameter facilitates communication between individuals. In this stage, with the help of the cross terms - $\theta_2 \times (G(t) - Z_i(t))$ and $(1-\theta_2) \times (\bar{Z}(t) - Z_i(t))$, the individuals are more likely to exploit promising regions based on the knowledge of current best search agent and centroid position of the swarm.

2.5. Dual-population mechanism

Realizing a tradeoff between exploitation and exploration is very significant in metaheuristic algorithms. As mentioned in Section 2.1,

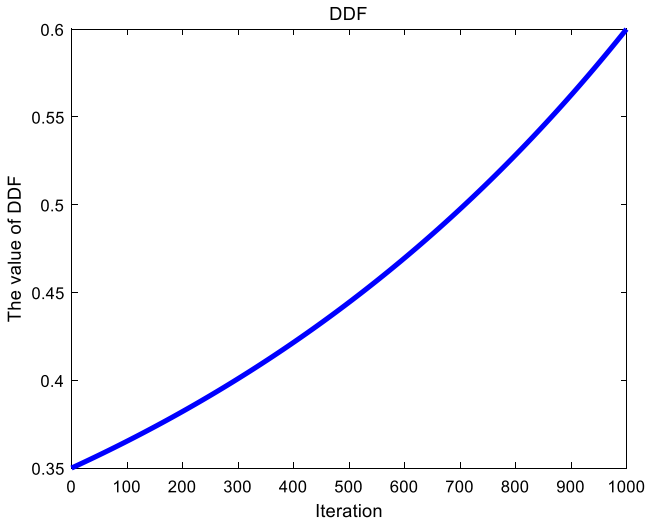


Fig. 5. Trends of the DDF.

some liquid water converted from the snow can also transform into steam to perform the exploration process. That is, over time, the likelihood of individuals doing irregular movements with a high-decentralized feature increases. Then the algorithm gradually has a tendency to explore the solution space. In our study, the dual-population mechanism is devised to reflect this situation and maintain exploitation and exploration. As presented in Algorithm 1, the whole population is randomly divided into two equal-sized subpopulations at the early phase of iteration. We denote the whole population and these two subpopulations as P, P_a, and P_b, respectively. In addition, the size of P, P_a, and P_b are denoted as N, N_a, and N_b, respectively. Among them, P_a is responsible for the exploration, whereas P_b is assigned to perform the exploitation. Then in the subsequent iterations, the size of P_b gradually declines and the size of P_a is accordingly increased.

Algorithm 1 Dual-population mechanism

1: Initialization: $t = 0, t_{max}, N_a = N_b = \frac{N}{2}$, where N denotes the population size
2: **while** $(t < t_{max})$ **do**
3: **if** $N_a < N$ **then**
4: $N_a = N_a + 1$, $N_b = N_b - 1$
5: **end if**
6: $t = t + 1$
7: **end while**

Table 1
Benchmark functions.

CEC2017	
$C2017_1, C2017_3$	unimodal problems
$C2017_4 - C2017_{10}$	simple multimodal problems
$C2017_{11} - C2017_{20}$	hybrid problems
$C2017_{21} - C2017_{30}$	composition problems

To sum up, the complete position updating equation of the SAO algorithm is shown in the following:

$$Z_i(t+1) = \begin{cases} Elite(t) + BM_i(t) \otimes (\theta_1 \times (G(t) - Z_i(t)) \\ \quad + (1-\theta_1) \times (\bar{Z}(t) - Z_i(t))), i \in index_a \\ M \times G(t) + BM_i(t) \otimes (\theta_2 \times (G(t) - Z_i(t)) \\ \quad + (1-\theta_2) \times (\bar{Z}(t) - Z_i(t))), i \in index_b \end{cases}, \quad (12)$$

As described in Eq. (1), the whole population is actually a position matrix. Hence, in Eq. (12), $index_a$ and $index_b$ denote a set of indexes including line numbers of individuals in P_a and P_b in the entire position matrix, respectively. The entire process of the SAO algorithm is summarized in Algorithm 2.

Algorithm 2 Snow ablation optimizer (SAO)

Initialization: the swarm $Z_i(i = 1, 2, \cdots, N)$, $t = 0, t_{max}, N_a = N_b = \frac{N}{2}$
Fitness evaluation
Record the current best individual $G(t)$
while $(t < t_{max})$ **do**
  Calculate the snowmelt rate M through Eq. (10)
  Randomly divide the whole population P into two subpopulations P_a and P_b
  for each individual **do**
    Update each individual's position through Eq. (12)
  end for
  Fitness evaluation
  Update $G(t)$
  $t = t + 1$
end while
Return $G(t)$

2.6. Time complexity of SAO

In this subsection, the time complexity of the snow ablation optimizer (SAO) is analyzed. Note that the SAO algorithm mainly consists of the following parts: initialization, position updating of individuals, fitness evaluation, and fitness sorting. The time complexity of each part is presented in the following: initialization $O(N * Dim)$, position updating of individuals $O(N * Dim)$, fitness evaluation $O(N)$, and fitness sorting $O(N * logN)$. Among them, Dim refers to the dimensionality of the solution space. Then the time complexity can be estimated in the following: $O(N * Dim + N * t_{max} * (logN + Dim + 1))$.

3. Experimental outcomes and analyses

In this part, the strength of the snow ablation optimizer (SAO) is validated through the CEC2017 test suite, CEC2020 real-world constrained optimization issues, and application to parameter extraction of photovoltaic systems. The whole experiment consists of the following five parts: (1) benchmark functions and experimental settings; (2) experiments on the CEC2017 test suite; (3) experiments on the CEC2020 real-world constrained optimization issues; (4) analysis of exploration and exploitation in SAO; and (5) application to parameter extraction for photovoltaic systems.

3.1. Benchmark functions and experimental settings

Experimental studies are performed on 29 10-dimensional and 30-dimensional benchmark problems from the CEC2017 test suite, respectively. Based on their characteristics, 29 CEC2017 unconstrained benchmarks are classified in Table 1. More details about these typical test problems can be found in the corresponding Ref. (Onay & Aydemr, 2022).

As reported in Table 1, unimodal problems are often utilized to validate the exploitation capability of algorithms because of the only one global optimum they possess. On the other hand, simple multimodal problems are more challenging to handle than unimodal problems because of the existence of numerous local optima. Moreover, hybrid problems and composition problems are likely to better imitate the real search space due to their different characteristics in different regions. Consequently, the potential performance of methods to cope with real-world optimization issues can be reflected by utilizing these typical benchmarks.

In our experiments, 30 search agents are employed and 1000 total iteration is utilized as the termination criterion. Moreover, eight state-of-the-art competitors are chosen to make a comparison with the snow ablation optimizer (SAO). They are AO (Abualigah, Yousri, et al., 2021), MVO (Mirjalili et al., 2015), EO (Faramarzi, Heidarinejad, Stephens, & Mirjalili, 2020), AVOA (Abdollahzadeh et al., 2021), HHO (Aaha et al., 2019), PSO-sono (Meng et al., 2022), SHADE (Tanabe & Fukunaga, 2013), and LSHADE-SPACMA (Mohamed et al., 2017). The parameter setup is kept consistent with the associated references, as given in Table 2. To alleviate the interference of randomness to the experiment, each algorithm is executed independently 20 times on each test problem. All experiments have been done utilizing MATLAB R2018a software under WINDOWS 10 64-bit operating system in Intel Core i5-5250U CPU @1.60 GHz. The outcome attained by the best algorithm is highlighted in boldface.

3.2. Experiments on the CEC2017 test suite

3.2.1. Convergence behavior of SAO

In this part, the convergence behavior of SAO is studied utilizing several CEC2017 benchmarks in the 2-dimensional parametric space. Specifically, in this experiment, the convergence behavior of SAO is reflected by the search history, convergence graph, history of average fitness, swarm's diversity, and diagram of trajectory in the first dimension. As depicted in Fig. 6, the first column is the description of the

Table 2
Parameter setup in rival methods.

Method	Reference	Parameter	Value
AO	Abualigah, Yousri, et al. (2021)	α, δ	0.1, 0.1
MVO	Mirjalili et al. (2015)	WEP_{max}, WEP_{min}	1, 0.2
EO	Faramarzi, Heidarinejad, Stephens, and Mirjalili (2020)	a_1, a_2, GP	2, 1, 0.5
AVOA	Abdollahzadeh et al. (2021)	P_1, P_2, P_3	0.6, 0.4, 0.6
HHO	Aaha et al. (2019)	E_0	(−1,1)
PSO-sono	Meng et al. (2022)	V_{max}, V_{min}, r	30, −30, 0.5
SHADE	Tanabe and Fukunaga (2013)	$Pbestrate$, $Arcrate$	0.1, 2
LSHADE-SPACMA	Mohamed et al. (2017)	H, N_{min}, $Pbest\_rate$, Arc_rate, F_{CP}, c	5, 4, 0.11, 1.4, 0.5, 0.8

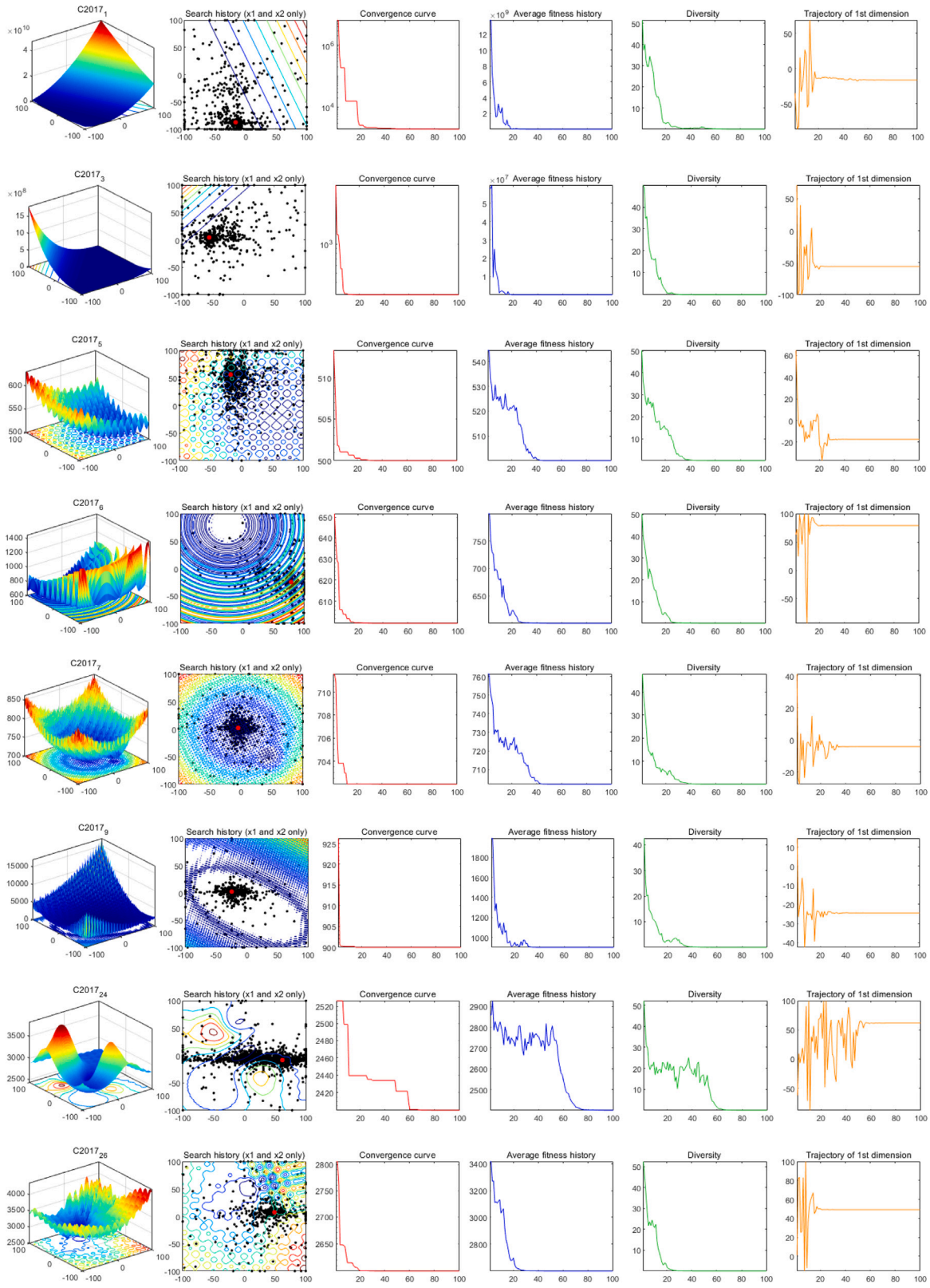


Fig. 6. Convergence behavior of SAO.

parametric space, and it reveals the smooth structure of unimodal problems such as C2017$_1$ and C2017$_3$. Meanwhile, a substantial number of local optima exist in simple multimodal problems and complicated hybrid problems well emulate the real solution space.

Then in the second column, the search history diagram visually presents all individuals' position history during the iteration procedure. Note that individuals tend to discover potential and promising areas in early iterations and finally cluster around the global optimal solution,

Table 3
Experimental outcomes attained by nine algorithms on the CEC2017 test suite with 10Dim.

Function	Metric	AO	MVO	EO	AVOA	HHO	PSO-sono	SHADE	LSHADE-SPACMA	SAO
$C2017_1$	Mean	1.07E+06	8.10E+03	3.13E+03	3.16E+03	5.43E+05	2.71E+03	**1.00E+02**	**1.00E+02**	1.99E+03
	SD	1.02E+03	4.81E+03	3.78E+03	1.83E+03	3.58E+05	3.57E+03	**0.00E+00**	**0.00E+00**	1.19E+03
$C2017_3$	Mean	833.07	300.04	300.00	303.52	307.19	300.00	**300.00**	**300.00**	300.00
	SD	2.38E+02	3.16E−02	2.25E−03	1.57E+01	9.52E+00	8.82E−08	**0.00E+00**	**0.00E+00**	1.30E−10
$C2017_4$	Mean	413.05	404.38	404.97	419.73	431.33	405.96	**400.00**	**400.00**	401.67
	SD	2.25E+01	1.61E+00	6.92E−01	2.98E+01	3.48E+01	1.35E+00	**0.00E+00**	**0.00E+00**	2.86E−01
$C2017_5$	Mean	5.32E+02	5.19E+02	5.16E+02	5.38E+02	5.49E+02	5.09E+02	5.05E+02	**5.04E+02**	5.14E+02
	SD	1.03E+01	8.03E+00	8.29E+00	1.88E+01	2.17E+01	5.24E+00	1.47E+00	**1.38E+00**	3.17E+00
$C2017_6$	Mean	614.31	601.11	600.00	613.13	636.41	600.38	**600.00**	600.00	600.00
	SD	5.74E+00	1.70E+00	1.78E−02	8.90E+00	1.05E+01	8.53E−01	**1.48E−05**	3.56E−04	1.13E−02
$C2017_7$	Mean	7.58E+02	7.25E+02	7.22E+02	7.68E+02	7.87E+02	7.19E+02	7.15E+02	**7.14E+02**	7.19E+02
	SD	1.32E+01	8.00E+00	7.00E+00	2.24E+01	1.79E+01	4.20E+00	1.93E+00	**2.01E+00**	4.09E+00
$C2017_8$	Mean	8.25E+02	8.20E+02	8.11E+02	8.33E+02	8.31E+02	8.11E+02	**8.04E+00**	8.05E+02	8.10E+02
	SD	6.67E+00	8.08E+00	4.54E+00	1.51E+01	9.79E+00	4.67E+00	**2.16E+00**	1.72E+00	4.18E+00
$C2017_9$	Mean	1059.92	900.45	900.62	1148.02	1424.84	901.53	900.37	900.26	**900.19**
	SD	1.42E+02	6.34E−01	3.02E−01	2.33E+02	2.40E+02	2.17E+00	7.32E−01	4.37E−01	**2.06E−01**
$C2017_{10}$	Mean	1.88E+03	1.78E+03	1.61E+03	1.92E+03	2.09E+03	1.41E+03	**1.24E+03**	1.36E+03	1.57E+03
	SD	2.67E+02	3.04E+02	2.64E+02	2.78E+02	3.86E+02	2.55E+02	**1.17E+02**	1.40E+02	1.29E+02
$C2017_{11}$	Mean	1171.62	1133.04	1106.91	1139.73	1169.96	1125.12	**1101.44**	1106.23	1108.81
	SD	3.28E+01	4.92E+01	4.14E+00	3.74E+01	5.48E+01	2.15E+01	**1.22E+00**	6.70E+00	3.37E+00
$C2017_{12}$	Mean	3.02E+06	8.50E+05	1.03E+04	1.28E+06	2.92E+06	1.11E+04	1.65E+03	**1.60E+03**	8.71E+03
	SD	3.34E+06	9.39E+05	7.75E+03	1.20E+06	3.39E+06	8.36E+03	2.87E+02	**2.22E+02**	4.03E+03
$C2017_{13}$	Mean	1.43E+04	1.00E+04	7.46E+03	1.02E+04	1.39E+04	1.57E+03	1.33E+03	**1.32E+03**	9.31E+03
	SD	1.41E+04	1.03E+04	6.58E+03	8.77E+03	1.04E+04	1.73E+02	6.20E+01	**5.15E+01**	4.92E+03
$C2017_{14}$	Mean	2.17E+03	1.70E+03	1.48E+03	1.88E+03	1.64E+03	1.45E+03	**1.41E+03**	1.42E+03	5.88E+03
	SD	8.81E+02	4.70E+02	3.17E+01	7.71E+02	1.92E+02	1.96E+01	**9.73E+00**	9.98E+00	6.17E+03
$C2017_{15}$	Mean	6.46E+03	2.20E+03	1.65E+03	4.59E+03	6.55E+03	1.62E+03	1.51E+03	**1.51E+03**	3.72E+03
	SD	7.94E+03	1.37E+03	9.09E+01	2.62E+03	2.15E+03	1.35E+02	1.43E+01	**7.87E+00**	2.83E+03
$C2017_{16}$	Mean	1.81E+03	1.79E+03	1.74E+03	1.78E+03	1.90E+03	1.65E+03	**1.61E+03**	1.62E+03	1.68E+03
	SD	1.32E+02	1.47E+02	6.19E+01	1.29E+02	1.24E+02	9.10E+01	**3.64E+01**	4.16E+01	7.93E+01
$C2017_{17}$	Mean	1772.17	1793.71	1742.94	1766.37	1775.52	1744.49	**1702.62**	1708.55	1741.57
	SD	2.85E+01	6.32E+01	2.77E+01	3.43E+01	4.66E+01	2.11E+01	**4.88E+00**	1.01E+01	2.73E+01
$C2017_{18}$	Mean	2.95E+04	1.88E+04	1.88E+04	1.29E+04	1.19E+04	2.31E+03	1.84E+03	**1.81E+03**	1.32E+04
	SD	1.55E+04	1.24E+04	1.26E+04	7.98E+03	7.27E+03	9.51E+02	2.35E+01	**1.02E+01**	1.01E+04
$C2017_{19}$	Mean	1.11E+04	2.68E+03	1.97E+03	7.99E+03	1.19E+04	1.92E+03	1.91E+03	**1.90E+03**	7.94E+03
	SD	8.59E+03	1.04E+03	4.91E+01	6.28E+03	8.75E+03	1.60E+01	1.50E+01	**8.87E−01**	6.97E+03
$C2017_{20}$	Mean	2.12E+03	2.07E+03	2.05E+03	2.14E+03	2.17E+03	2.03E+03	2.01E+03	**2.01E+03**	2.05E+03
	SD	5.69E+01	5.68E+01	5.32E+01	8.13E+01	7.52E+01	1.21E+01	8.18E+00	**3.14E+00**	5.83E+01
$C2017_{21}$	Mean	2313.89	2306.64	2295.84	2293.55	2337.74	2304.35	2296.87	**2245.21**	2292.64
	SD	3.82E+01	3.73E+01	4.08E+01	6.82E+01	5.04E+01	2.48E+01	3.32E+01	**1.55E+01**	4.74E+01
$C2017_{22}$	Mean	2306.58	2320.86	2300.63	2308.16	2306.53	2299.07	2300.51	2300.65	**2297.67**
	SD	1.61E+01	7.65E+01	3.63E+00	5.41E+00	2.38E+01	1.46E+01	4.35E+00	4.63E+00	**1.23E+00**
$C2017_{23}$	Mean	2642.35	2620.34	2615.77	2640.06	2677.54	2608.83	2609.43	**2608.11**	2616.37
	SD	1.33E+01	8.52E+00	6.07E+00	1.66E+01	3.03E+01	3.79E+00	2.95E+00	**2.27E+00**	7.07E+00
$C2017_{24}$	Mean	2740.69	2748.08	2745.69	2762.08	2815.91	2737.47	2737.71	**2737.45**	2743.12
	SD	8.34E+01	1.15E+01	6.37E+00	6.54E+01	4.49E+01	3.64E+00	3.04E+00	**2.26E+00**	7.28E+00
$C2017_{25}$	Mean	2926.78	2925.94	2933.29	**2913.97**	2919.58	2938.31	2916.91	2926.37	2922.72
	SD	2.32E+01	2.32E+01	3.15E+01	**2.04E+01**	7.48E+01	1.95E+01	2.35E+01	2.35E+01	1.07E+01
$C2017_{26}$	Mean	3052.78	2942.87	2969.37	3177.85	3470.52	3013.51	2968.22	2937.1	**2934.44**
	SD	1.47E+02	2.53E+02	2.38E+02	3.17E+02	5.83E+02	2.94E+02	1.98E+02	4.93E+01	**3.79E+01**
$C2017_{27}$	Mean	3102.81	3097.48	3091.68	3104.56	3150.11	**3080.91**	3091.25	3090.52	3095.73
	SD	6.85E+00	1.85E+01	2.48E+00	2.08E+01	4.44E+01	**2.15E+00**	3.81E+00	2.71E+00	5.07E+00
$C2017_{28}$	Mean	3395.92	3288.97	3316.08	3351.27	3434.87	3288.21	3329.65	3355.09	**3285.24**
	SD	9.36E+01	1.31E+02	1.36E+02	1.14E+02	1.68E+02	1.08E+02	1.36E+02	1.04E+02	**1.02E+02**
$C2017_{29}$	Mean	3224.37	3229.43	3189.89	3265.47	3374.57	3193.45	3153.91	**3151.61**	3188.35
	SD	4.33E+01	6.46E+01	4.97E+01	8.44E+01	9.83E+01	3.24E+01	3.58E+01	**1.52E+01**	6.08E+01
$C2017_{30}$	Mean	7.75E+05	3.13E+05	2.81E+05	1.96E+05	2.03E+06	1.50E+05	**3.20E+03**	3.40E+03	1.93E+05
	SD	9.35E+05	5.26E+05	4.33E+05	2.82E+05	2.71E+06	2.54E+05	**4.53E+00**	4.23E+01	3.86E+05

which indicates SAO realizes a great tradeoff between exploration and exploitation. Especially for hybrid problem $C2017_{24}$, the SAO has focused on exploiting the left region in the solution space for a long time, whereas the best outcomes are attained in the right space. Actually, this shows the excellent exploration capability possessed by the developed technique, which enables the swarm's diversity to be preserved and facilitates local optima avoidance.

The convergence graph in the third column is the most broadly utilized metric to validate the performance of metaheuristic techniques. As described in Fig. 6, the convergence graphs attained by SAO suggest

Table 4
Experimental outcomes attained by nine algorithms on the CEC2017 test suite with 30Dim.

Function	Metric	AO	MVO	EO	AVOA	HHO	PSO-sono	SHADE	LSHADE-SPACMA	SAO
$C2017_1$	Mean	4.34E+08	5.80E+05	4.09E+03	6.45E+03	3.00E+07	1.03E+05	1.00E+02	**1.00E+02**	3.96E+03
	SD	1.47E+08	1.94E+05	3.95E+03	5.61E+03	6.77E+06	2.17E+05	2.30E−02	**4.54E−04**	5.40E+03
$C2017_3$	Mean	5.75E+04	1.82E+03	2.83E+04	3.49E+04	3.99E+04	1.30E+04	**6.98E+02**	8.35E+02	9.14E+04
	SD	7.49E+03	1.20E+03	8.87E+03	8.12E+03	6.59E+03	6.27E+03	**2.43E+01**	2.97E+01	3.07E+04
$C2017_4$	Mean	6.49E+02	4.99E+02	5.00E+02	5.19E+02	5.70E+02	5.66E+02	4.58E+02	**4.57E+02**	4.92E+02
	SD	6.85E+01	1.84E+01	2.19E+01	2.98E+01	4.48E+01	6.50E+01	3.35E+01	**3.20E+01**	1.33E+01
$C2017_5$	Mean	7.07E+02	6.14E+02	5.86E+02	7.32E+02	7.52E+02	5.55E+02	**5.40E+02**	5.56E+02	5.67E+02
	SD	4.54E+01	3.49E+01	1.44E+01	3.86E+01	3.58E+01	2.33E+01	**8.27E+00**	1.18E+01	1.23E+01
$C2017_6$	Mean	653.71	622.28	600.74	651.97	664.06	604.52	600.13	**600.00**	600.42
	SD	7.44E+00	1.46E+01	9.50E−01	7.84E+00	6.28E+00	3.15E+00	3.24E−01	**9.75E−04**	5.37E−01
$C2017_7$	Mean	1.11E+03	8.52E+02	8.29E+02	1.16E+03	1.27E+03	8.21E+02	**7.79E+02**	7.97E+02	9.20E+02
	SD	5.57E+01	2.67E+01	2.86E+01	8.62E+01	6.49E+01	6.01E+01	**9.75E+00**	1.13E+01	5.48E+01
$C2017_8$	Mean	9.54E+02	9.08E+02	8.85E+02	9.64E+02	9.70E+02	8.67E+02	**8.41E+02**	8.56E+02	8.65E+02
	SD	2.69E+01	3.41E+01	2.82E+01	3.27E+01	1.83E+01	1.68E+01	**6.66E+00**	1.00E+01	1.82E+01
$C2017_9$	Mean	6.71E+03	4.09E+03	1.18E+03	5.32E+03	8.06E+03	1.11E+03	9.31E+02	**9.28E+02**	9.71E+02
	SD	1.11E+03	3.16E+03	4.55E+02	7.89E+02	7.85E+02	1.87E+02	3.59E+01	**2.36E+01**	2.58E+02
$C2017_{10}$	Mean	5.74E+03	4.86E+03	5.16E+03	5.60E+03	5.72E+03	4.67E+03	**3.88E+03**	4.62E+03	4.26E+03
	SD	5.75E+02	7.16E+02	8.60E+02	8.26E+02	6.90E+02	9.72E+02	**2.92E+02**	4.73E+02	7.68E+02
$C2017_{11}$	Mean	2352.47	1330.91	1217.39	1251.94	1293.22	1316.44	1227.69	**1199.73**	1213.49
	SD	6.49E+02	6.85E+01	4.19E+01	4.54E+01	4.24E+01	8.94E+01	4.85E+01	**3.76E+01**	5.05E+01
$C2017_{12}$	Mean	1.04E+08	1.02E+07	1.21E+06	6.53E+06	2.69E+07	3.39E+06	2.64E+04	**2.10E+04**	5.75E+05
	SD	1.16E+08	8.61E+06	7.99E+05	5.58E+06	2.06E+07	7.77E+06	1.83E+04	**1.06E+04**	5.97E+05
$C2017_{13}$	Mean	1.36E+06	1.64E+05	2.27E+04	1.48E+05	6.37E+05	2.01E+04	**2.58E+03**	3.09E+03	1.80E+04
	SD	1.26E+06	1.11E+05	2.10E+04	9.01E+04	2.40E+05	1.58E+04	**2.63E+02**	4.70E+02	1.95E+04
$C2017_{14}$	Mean	1.23E+06	2.95E+04	5.02E+04	2.88E+05	9.68E+05	5.57E+04	**1.57E+03**	1.61E+03	3.34E+05
	SD	6.81E+05	3.52E+04	5.80E+04	2.62E+05	9.47E+05	7.65E+04	**4.27E+01**	5.19E+01	3.04E+05
$C2017_{15}$	Mean	1.18E+05	4.85E+04	4.87E+03	3.68E+04	8.97E+04	9.23E+03	**1.85E+03**	3.89E+03	4.14E+03
	SD	6.20E+04	1.90E+04	3.22E+03	2.92E+04	4.98E+04	1.15E+04	**1.76E+02**	2.35E+02	3.83E+03
$C2017_{16}$	Mean	3.31E+03	2.77E+03	2.52E+03	3.18E+03	3.51E+03	2.40E+03	2.42E+03	**2.33E+03**	2.35E+03
	SD	4.78E+02	2.80E+02	3.61E+02	5.24E+02	4.32E+02	3.71E+02	2.42E+02	**2.01E+02**	3.07E+02
$C2017_{17}$	Mean	2.47E+03	2.11E+03	2.04E+03	2.66E+03	2.61E+03	2.00E+03	1.95E+03	**1.92E+03**	2.09E+03
	SD	2.85E+02	1.76E+02	1.77E+02	2.00E+02	2.58E+02	1.50E+02	1.32E+02	**8.46E+01**	1.72E+02
$C2017_{18}$	Mean	5.55E+06	3.54E+05	5.57E+05	1.45E+06	2.11E+06	3.68E+05	8.75E+03	**4.07E+03**	5.02E+05
	SD	5.86E+06	2.59E+05	3.58E+05	1.43E+06	2.74E+06	4.23E+05	2.08E+03	**1.28E+03**	3.80E+05
$C2017_{19}$	Mean	2.18E+06	9.89E+05	1.08E+04	4.54E+04	8.77E+05	1.48E+04	**2.08E+03**	2.17E+03	5.76E+03
	SD	1.70E+06	1.06E+06	1.41E+04	3.77E+04	7.93E+05	1.32E+04	**8.05E+01**	1.23E+02	3.93E+03
$C2017_{20}$	Mean	2526.16	2537.54	2364.34	2767.79	2884.66	2342.51	2265.24	**2256.87**	2401.25
	SD	1.56E+02	1.67E+02	1.76E+02	2.27E+02	2.08E+02	1.64E+02	9.91E+01	**7.96E+01**	1.89E+02
$C2017_{21}$	Mean	2491.88	2415.44	2364.71	2518.67	2574.67	2357.95	2361.21	**2342.14**	2366.25
	SD	3.75E+01	3.32E+01	2.32E+01	3.02E+01	4.72E+01	1.64E+01	9.81E+00	**4.55E+00**	1.65E+01
$C2017_{22}$	Mean	2.61E+03	4.89E+03	3.77E+03	6.08E+03	7.55E+03	2.52E+03	2.92E+03	2.75E+03	**2.41E+03**
	SD	1.52E+02	1.65E+03	2.16E+03	2.34E+03	9.19E+02	9.55E+02	8.26E+03	7.38E+02	**5.22E+02**
$C2017_{23}$	Mean	2958.97	2745.95	2712.96	2962.96	3227.63	2721.91	**2698.96**	2710.63	2713.14
	SD	6.06E+01	2.67E+01	2.53E+01	7.36E+01	1.90E+02	3.29E+01	**8.07E+00**	1.78E+01	1.21E+01
$C2017_{24}$	Mean	3.10E+03	2.92E+03	2.87E+03	3.13E+03	3.49E+03	2.88E+03	**2.86E+03**	2.87E+03	2.89E+03
	SD	6.79E+01	3.27E+01	1.83E+01	7.76E+01	1.27E+02	2.43E+01	**1.11E+01**	1.51E+01	1.93E+01
$C2017_{25}$	Mean	2993	2894.78	2900.65	2921.45	2937.83	2924.64	2888.19	2887.39	**2886.82**
	SD	2.73E+01	1.73E+01	1.95E+01	3.35E+01	2.38E+01	2.75E+01	2.68E+00	1.02+00	**3.09E+00**
$C2017_{26}$	Mean	5.91E+03	4.61E+03	4.21E+03	6.56E+03	7.78E+03	4.48E+03	4.11E+03	**3.98E+03**	4.16E+03
	SD	1.46E+03	7.64E+02	3.87E+02	1.59E+03	1.03E+03	6.04E+02	2.23E+02	**1.31E+02**	2.51E+02
$C2017_{27}$	Mean	3393.24	3226.8	3222.77	3273.18	3472.33	3240.35	3221.49	**3218.96**	3225.85
	SD	8.30E+01	1.82E+01	1.33E+01	3.83E+01	8.25E+01	3.23E+01	1.41E+01	**9.24E+00**	1.12E+01
$C2017_{28}$	Mean	3451.81	3244.98	3224.44	3278.38	3334.12	3308.34	3197.41	**3158.25**	3220.21
	SD	8.44E+01	4.39E+01	2.37E+01	3.05E+01	3.43E+01	5.49E+01	5.18E+01	**1.66E+01**	3.03E+01
$C2017_{29}$	Mean	4.81E+03	4.05E+03	3.69E+03	4.32E+03	4.91E+03	3.96E+03	**3.51E+03**	3.56E+03	3.79E+03
	SD	4.08E+02	1.91E+02	1.76E+02	3.12E+02	3.96E+02	2.00E+02	**1.12E+02**	2.03E+02	2.15E+02
$C2017_{30}$	Mean	1.35E+07	3.38E+06	1.12E+04	6.30E+05	5.02E+06	8.89E+04	**5.71E+03**	6.27E+03	9.47E+03
	SD	1.35E+07	2.15E+06	5.08E+03	3.20E+05	3.43E+06	1.45E+05	**5.70E+02**	6.17E+02	3.32E+03

that the algorithm has a rapid convergence rate on all eight benchmarks. For unimodal problems, due to the interaction and learning between individuals, SAO presents a good exploitative characteristic to approach the global optimum. When handling simple multimodal problems and hybrid problems, SAO sometimes falls temporarily into local optima, but the algorithm achieves a high precision under the guidance of the elites in the swarm. Meanwhile, in the last steps of iteration, the dynamic step length generated by Brownian motion

Table 5
Comparison outcomes attained by Wilcoxon's test on the 10-dimensional CEC2017 test suite.

Methods	Unimodal problems	Simple multimodal problems	Hybrid problems	Composition problems
SAO vs. AO	3.96E−08	4.86E−05	6.56E−03	8.05E−03
SAO vs. MVO	3.79E−06	8.31E−03	1.45E−02	2.35E−02
SAO vs. EO	4.51E−03	4.18E−02	3.78E−02	3.96E−02
SAO vs. AVOA	3.00E−02	1.58E−05	2.82E−02	8.89E−03
SAO vs. HHO	3.96E−08	1.29E−05	5.83E−03	4.14E−03
SAO vs. PSO-sono	4.19E−02	1.44E−01	1.50E−02	3.50E−02
SAO vs. SHADE	1.71E−01	1.12E−01	2.23E−06	1.98E−02
SAO vs. LSHADE-SPACMA	1.70E−01	4.96E−02	1.62E−02	3.00E−02

Table 6
Comparison outcomes attained by Wilcoxon's test on the 30-dimensional CEC2017 test suite.

Methods	Unimodal problems	Simple multimodal problems	Hybrid problems	Composition problems
SAO vs. AO	1.44E−07	3.97E−07	2.79E−03	4.32E−04
SAO vs. MVO	6.79E−08	1.51E−02	1.16E−02	9.33E−03
SAO vs. EO	2.71E−03	2.61E−02	1.41E−01	2.83E−02
SAO vs. AVOA	5.37E−03	4.50E−03	2.31E−03	1.41E−05
SAO vs. HHO	6.79E−08	6.27E−07	7.64E−03	7.46E−08
SAO vs. PSO-sono	4.62E−05	3.52E−02	3.82E−02	1.68E−01
SAO vs. SHADE	1.62E−07	4.37E−02	1.08E−01	2.68E−02
SAO vs. LSHADE-SPACMA	1.05E−07	1.07E−01	7.00E−02	1.79E−01

Table 7
Rankings of nine methods on 29 CEC2017 unconstrained benchmarks on the basis of Friedman's test.

Methods	Ranking	
	10Dim	30Dim
LSHADE-SPACMA	**2.051**	**1.689**
SHADE	2.189	1.844
SAO	3.741	3.689
PSO-sono	3.862	4.293
EO	4.586	4.517
MVO	6.017	5.793
AVOA	6.827	6.896
AO	7.586	7.931
HHO	8.137	8.344

can discourage premature convergence effectively. Also, in the fourth column, the descending behavior can be observed in the diagrams of average fitness history.

The swarm's diversity and the diagram of trajectory in the first dimension are shown in the fifth column and the sixth column, respectively. As shown in Fig. 6, since the swarm's diversity is well preserved in the initial steps of iteration, the trajectory of individuals presents abrupt and large changes, which indicates SAO is more possibly explore and discover potential and high-quality solutions.

3.2.2. Comparisons with other state-of-the-art rival algorithms

This subsection compares the SAO with AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA utilizing 10-dimensional and 30-dimensional CEC2017 test suites, respectively. The simulation outcomes attained by nine algorithms are summarized in Tables 3 and 4. On the basis of the mean and standard deviation, some comments are shown as follows:

- For unimodal problems $C2017_1$ and $C2017_3$, when the dimensionality of the problem is 10, the SAO algorithm attains better outcomes than all rival algorithms except SHADE and LSHADE-SPACMA. Especially for function $C2017_3$, SAO can straightly discover the global optimal solution, whereas AO, MVO, AVOA, and HHO cannot. Note that EO and PSO-sono also provide the best outcomes on $C2017_3$, but they are surpassed by SAO based on the standard deviation. In the case of 30Dim, the performance of SAO is not seriously deteriorated due to the increase in dimensionality.
- For simple multimodal problems $C2017_4$ - $C2017_{10}$, in the case of 10Dim, SAO outperforms AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA on 7, 7, 6, 7, 7, 4, 1, and 1 problems, respectively. In the case of 30Dim, SAO surpasses AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA on 7, 6, 6, 7, 7, 5, 0, and 0 problems, respectively.
- For hybrid problems $C2017_{11}$ - $C2017_{20}$, in the case of 10 Dim, AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA are surpassed by SAO on 9, 7, 5, 8, 8, 3, 0, and 0 problems, respectively. Moreover, in the case of 30Dim, SAO outperforms AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA on 10, 8, 7, 7, 10, 5, 0, and 0 problems, respectively.
- For composition problems $C2017_{21}$ - $C2017_{30}$, in the case of 10Dim, SAO outperforms AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA on 9, 10, 8, 9, 9, 6, 4, and 4 problems, respectively. Besides, SAO can attain the best outcomes on 3 problems, whereas AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA attain the best outcomes on 0, 0, 0, 1, 0, 1, 1, and 4 problems, respectively. As the dimensionality of the problem rises to 30, SAO surpasses AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, and LSHADE-SPACMA on 10, 10, 5, 10, 10, 8, 2, and 2 problems, respectively.

The Wilcoxon test (Derrac et al., 2011) is conducted on these nine algorithms on the basis of 10-dimensional and 30-dimensional CEC2017 test suites. As the test outcomes in Tables 5 and 6 present, in most cases, the attained p values are less than 5%. This indicates that the developed SAO statistically outperforms other competitors except for SHADE and LSHADE-SPACMA. Additionally, Friedman's test (Derrac et al., 2011) is utilized to validate the strength of SAO. As summarized in Table 7, SAO is ranked third among these nine algorithms, followed by PSO-sono, EO, MVO, AVOA, AO, and HHO.

Fig. 7 depicts the convergence graphs attained by AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA, and SAO on 6 typical benchmark problems such as $C2017_3$, $C2017_6$, $C2017_{21}$, $C2017_{22}$, $C2017_{26}$, and $C2017_{28}$ in the CEC2017 test suite with 10Dim. Note that SHADE and LSHADE-SPACMA are two most outstanding algorithms to tackle unconstrained benchmarks. Meanwhile, PSO-sono is a representative state-of-the-art PSO variants for single-objective optimization issues. As shown in Fig. 7, the SAO algorithm achieves a high level of convergence rate and accuracy, even if it is surpassed by SHADE and LSHADE-SPACMA when coping with some complicated benchmarks. Moreover, SAO presents superior accelerations in comparison to other competitors such as AO, MVO, EO and AVOA. The

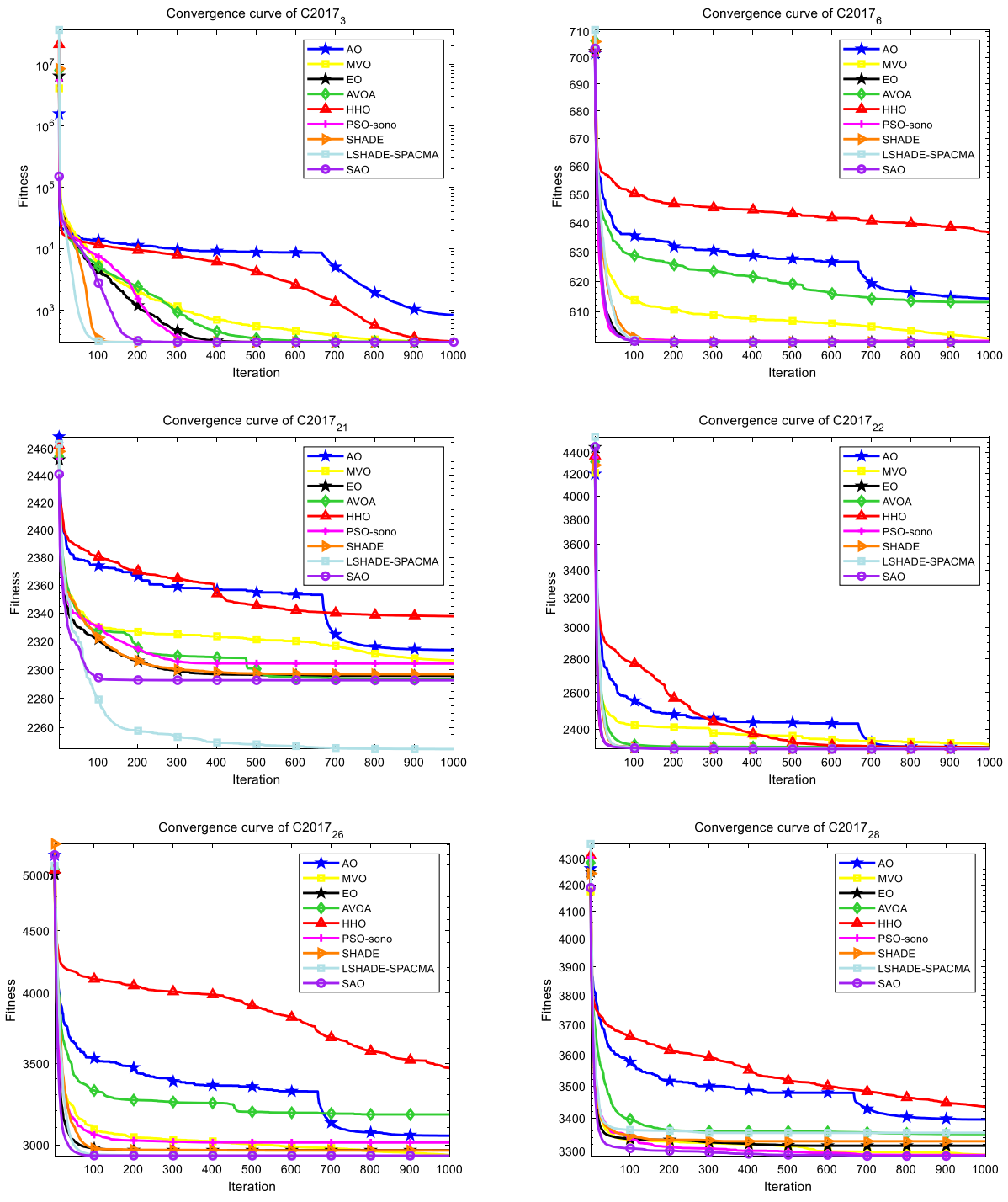


Fig. 7. Convergence graphs attained by nine algorithms on 6 CEC2017 benchmark problems with 10Dim.

convergence graphs attained by nine methods on C2017$_3$, C2017$_6$, C2017$_{21}$, C2017$_{22}$, C2017$_{26}$, and C2017$_{28}$ are shown in Fig. 8. When the dimensionality of the problem rises to 30, as depicted in Fig. 8, the advantages of SAO over other rival algorithms are not weakened. On the contrary, according to the increasing gap in the convergence graph, the advantages of the developed technique become clear.

3.3. Analysis of exploration and exploitation in SAO

In this part, the behavior of exploration and exploitation in SAO is analyzed through CEC2017 unconstrained benchmarks with 10Dim. The above experiments investigate the performance of SAO through some metrics such as standard deviation and mean value. Nevertheless, the reason behind the excellent performance is not revealed. Hence, this subsection explains how and why the SAO algorithm can yield better performance than other competitors on numerous benchmark instances. In Hussain et al. (2018), Hussain et al. put forward an approach to measure and analyze the capability of exploitation and exploration in metaheuristic algorithms. This approach builds on the mathematical representation of dimension-wise diversity raised in Hussain et al. (2018). The corresponding formulae are presented in the following:

$$Divs_j = \frac{1}{N}\sum_{i=1}^{N} median(z^j) - z_i^j, \tag{13}$$

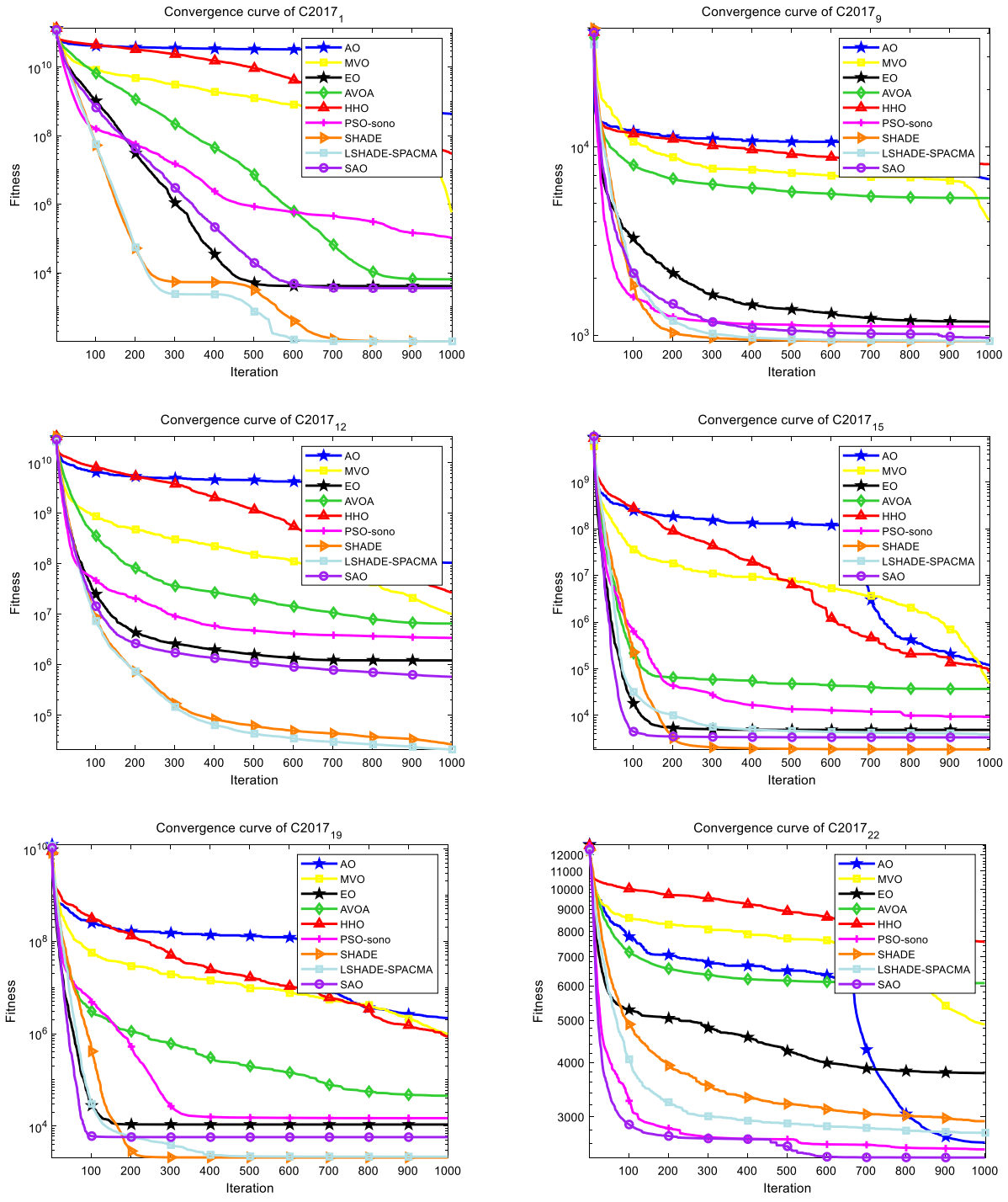


Fig. 8. Convergence graphs attained by nine algorithms on 6 CEC2017 benchmark problems with 30Dim.

$$Divs = \frac{1}{Dim}\sum_{j=1}^{Dim} Divs_j, \tag{14}$$

where $median(z^j)$ indicates the median of the jth dimension. Then the formulae to calculate the exploration percentage and exploitation percentage are summarized as follows:

$$Epl\% = \frac{Divs}{Divs_{\max}} \times 100, \tag{15}$$

$$Ept\% = \frac{|Divs - Divs_{\max}|}{Divs_{\max}} \times 100, \tag{16}$$

where Div_{max} represents the maximum diversity. $Epl\%$ and $Ept\%$ refer to the exploration percentage and exploitation percentage, respectively.

Fig. 9 depicts the simulation results of SAO on 6 CEC2017 benchmarks with 10Dim. As depicted in Fig. 9, the raised SAO maintains a high level of exploitation percentage when coping with unimodal problems and that exactly gives the reason behind the excellent capability of SAO to discover the global optimum on some unimodal problems such as C2017$_3$. On the other hand, when applying the SAO algorithm to handle challenging benchmarks such as composition problems and hybrid problems, it can also achieve an appropriate level of exploration percentage, and that is why the SAO can enable the swarm's diversity to be preserved and effectively discourage premature convergence. To

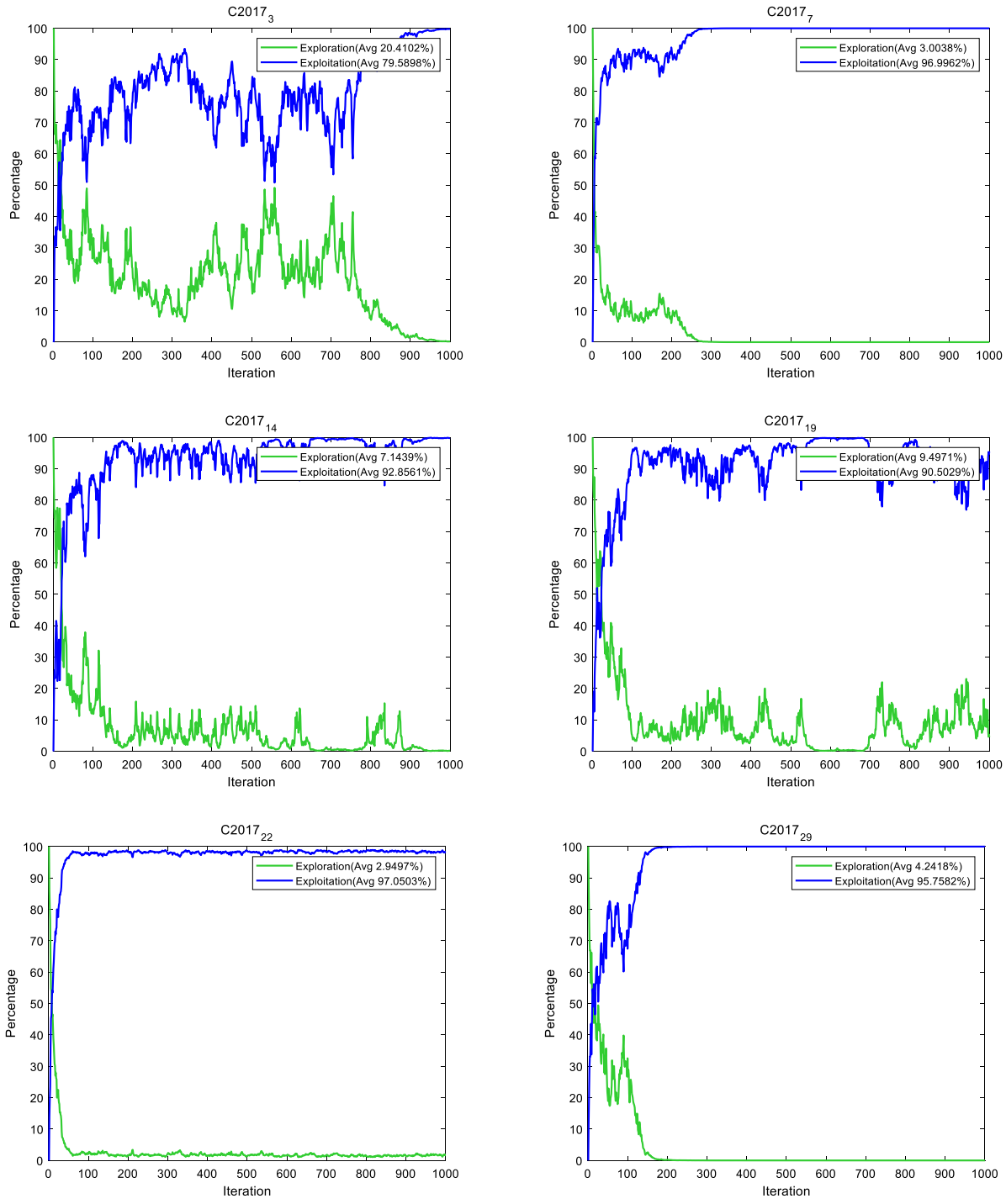


Fig. 9. Graphs of exploitation percentage and exploration percentage for SAO on 6 CEC2017 unconstrained benchmarks.

sum up, the SAO algorithm realizes a tradeoff between exploration and exploitation to a certain extent.

3.4. Experiments on the CEC2020 real-world constrained optimization issues

Different from the above CEC2017 unconstrained benchmarks, in this subsection, 22 CEC2020 real-world constrained optimization issues in diverse engineering fields are employed to verify the strength of the proposed SAO. These complicated nonconvex problems are classified into two types: process synthesis and design issues and mechanical engineering issues, which are constrained by equality and inequality constraints. Table 8 presents a brief description of these challenging issues and their mathematical expressions can be found in Kumar et al. (2020). Besides, the penalty function approach is employed as the constraint handling method. In the experiment, Each algorithm is executed 20 times independently, 1000 total iteration is utilized as the termination criterion and 30 individuals are employed.

Table 9 reports the experimental outcomes attained by nine algorithms on 22 real-world engineering applications. Some observations are presented as follows:

- For process synthesis and design problems R1-R7, SAO outperforms the other eight rival algorithms. More specifically, SAO

Table 8
22 real-world engineering issues. *Dim* denotes the dimensionality of the issue, N_1 and N_2 indicate the number of inequality constraints and equality constraints, respectively. f^* represents the best known feasible objective fitness value.

Category	No.	Name	N_1	N_2	*Dim*	f^*
Process synthesis and design problems	R1	Process synthesis problem 1	2	0	2	2.0000000000E+00
	R2	Process flow sheeting problem	3	0	3	1.0765430833E+00
	R3	Process synthesis problem 2	9	0	7	2.9248305537E+00
	R4	Multi-product batch plant	10	0	10	5.3638942722E+04
	R5	Process synthesis and design problem	1	1	3	2.5576545740E+00
	R6	Two-reactor problem	4	4	7	9.9238463653E+01
	R7	Process design problem	3	0	5	2.6887000000E+04
Mechanical engineering problems	R8	Speed reducer design	11	0	7	2.9944244658E+03
	R9	Compression/tension spring design (case1)	3	0	3	1.2665232788E−02
	R10	Welded beam design	5	0	4	1.6702177263E+00
	R11	Multiple disk clutch brake design	7	0	5	2.3524245790E−01
	R12	Step-cone pulley problem	8	3	5	1.6069868725E+01
	R13	Four-stage gear box problem	86	0	22	3.5359231973E+01
	R14	Rolling element bearing	9	0	10	1.4614135715E+04
	R15	Compression/tension spring design (case2)	8	0	3	2.6138840583E+00
	R16	Himmelblau's function	6	0	5	−3.0665538672E+04
	R17	Industrial refrigeration system design	15	0	14	3.2213000814E−02
	R18	Pressure vessel design	4	0	4	5.8853327736E+03
	R19	Three-bar truss design	3	0	2	2.6389584338E+02
	R20	Planetary gear train design	10	1	9	5.2576870748E−01
	R21	Gas transmission compressor design	1	0	4	2.9648954173E+06
	R22	Gear train design	1	1	4	0.0000000000E+00

provides the best outcomes on 4 problems, whereas AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA attain the best outcomes on 0, 0, 2, 0, 0, 3, 2, 3, and 5 problems, respectively. Hence, SAO has excellent potential to tackle these process synthesis and design problems.

- For mechanical engineering problems R8-R22, SAO is superior to AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA on 5, 5, 5, 4, 4, 4, 5, and 5 problems, respectively. Especially for the gear train design, the SAO algorithm can discover the global optimum eventually and attain the best standard deviation. Consequently, SAO is good at coping with these mechanical engineering problems.

Furthermore, Friedman's test is performed on nine algorithms on the basis of 22 real-world engineering problems. Test outcomes reported in Table 10 indicate that the SAO attains the best ranking, followed by EO, SHADE, LSHADE-SPACMA, AVOA, HHO, MVO, PSO-sono, and AO.

3.5. Application to parameter extraction for photovoltaic systems

In this part, a more crucial and meaningful real-world engineering application is employed to evaluate the advantages of SAO over other well-regarded competitors. In the field of new energy, photovoltaic (PV) systems are powerful tools that can use solar energy and directly transform it into electricity. Hence, designing an efficient and accurate model for PV systems through its parameters extracted on the basis of the measured current–voltage data is a vital assignment. SAO and the other eight algorithms are utilized to extract the core parameters for PV systems. Three classical PV models such as the single diode model (SDM), double diode model (DDM), and PV module model (PVMM) are employed. The equivalent circuit diagrams of these three models are depicted in Fig. 10.

Five core parameters need to be extracted in SDM: the source of photocurrent (I_{ph}), the reverse saturation current (I_{sd}), the series resistance (R_s), the shunt resistance (R_{sh}), and the ideal factor of diodes (n), as depicted in Fig. 10(a). The mathematical equation satisfied by these parameters is summarized below:

$$I_L = I_{ph} - I_{sh} - I_d, \tag{17}$$

$$I_d = I_{sd} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n \cdot k}) - 1], \tag{18}$$

$$I_{sh} = \frac{I_L \cdot R_s + V_L}{R_{sh}}, \tag{19}$$

where I_d represents the current of the diode calculated by Eq. (18), I_L indicates the output current, V_L denotes the output voltage, T refers to the battery temperature in Kelvin, q represents the electronic charge ($1.60217646 \times 10^{-19}$), and k indicates the Boltzmann's constant ($1.3806503 \times 10^{-23}$ J/k). Then in SDM, the calculation formula of I_L is presented in Eq. (20).

$$I_L = I_{ph} - \frac{I_L \cdot R_s + V_L}{R_{sh}} - I_{sd} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n \cdot k}) - 1], \tag{20}$$

Different from SDM, the DDM is raised by considering the effect of the recombination loss in depletion regions. Fig. 10(b) depicts the equivalent circuit diagram. Then the formula to calculate I_L in DDM is presented in Eq. (21).

$$I_L = I_{ph} - I_{sh} - I_{d1} - I_{d2} = I_{ph} - \frac{I_L \cdot R_s + V_L}{R_{sh}} - I_{sd1} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n_1 \cdot k}) - 1] - I_{sd2} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n_2 \cdot k}) - 1], \tag{21}$$

where I_{sd1} denotes the diffusion current, and I_{sd2} indicates the saturation current. n_1 and n_2 refer to the ideality factors of diodes. In DDM, seven core parameters (I_{ph}, I_{sd1}, R_s, R_{sh}, n_1, I_{sd2}, n_2) is supposed to be extracted.

As depicted in Fig. 10(c), several identical PV cells in parallel or series make up the single-diode-based PVMM. The mathematical expression of I_L in PVMM is given in Eq. (22).

$$I_L = I_{ph} \cdot N_p - I_{sd} \cdot N_p \cdot \exp[(\frac{(I_L N_s R_s / N_p + V_L) \cdot q}{T \cdot n \cdot k \cdot N_s}) - 1] - \frac{I_L R_s N_s / N_p + V_L}{N_s R_{sh} / N_p}, \tag{22}$$

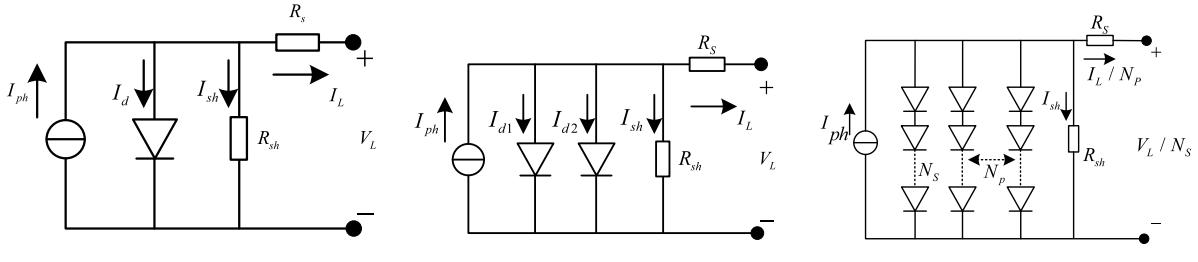
where N_p and N_s refer to the number of cells connected in parallel and series, respectively. In this model, five important parameters ($I_{ph}, I_{sd}, R_s, R_{sh}, n$) are supposed to be extracted. The bounds of different parameters in three PV models are reported in Table 11.

In this problem, the main target is to realize the minimization of error between the experimental data and the measured data. As a complicated nonlinear optimization problem, the root mean square error (RMSE) usually serves as the fitness function, and its mathematical definition is presented in Eq. (23).

$$RMSE(X) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} f_i(X, I_L, V_L)}, \tag{23}$$

Table 9
Experimental outcomes for problems R1-R22.

Problem	Metric	AO	MVO	EO	AVOA	HHO	PSO-sono	SHADE	LSHADE-SPACMA	SAO
R1	Mean	2.00E+00	2.00E+00	2.00E+00	2.00E+00	2.00E+00	**2.00E+00**	2.00E+00	2.00E+00	2.00E+00
	SD	1.45E−04	5.12E−06	4.60E−07	3.14E−07	4.60E−07	**2.37E−16**	3.14E−07	4.11E−07	4.11E−07
R2	Mean	1.25E+00	1.27E+00	1.10E+00	1.12E+00	1.12E+00	1.24E+00	1.18E+00	1.22E+00	**1.09E+00**
	SD	9.50E−02	9.69E−02	8.71E−02	7.89E−02	9.87E−02	3.79E−02	4.78E−02	8.81E−02	**2.50E−02**
R3	Mean	2.95E+00	3.50E+00	3.49E+00	3.36E+00	2.96E+00	4.99E+00	3.36E+00	3.18E+00	**2.93E+00**
	SD	2.68E−02	5.90E−01	9.00E−01	7.70E−01	2.82E−02	1.26E+00	9.11E−01	4.91E−01	**2.74E−02**
R4	Mean	7.24E+04	5.43E+04	5.46E+04	5.47E+04	6.37E+04	5.49E+04	5.47E+04	**5.43E+04**	5.43E+04
	SD	6.19E+03	2.32E+03	2.48E+03	2.51E+03	6.16E+03	3.18E+03	3.88E+03	**1.10E+03**	2.68E+03
R5	Mean	2.566	2.557	**2.557**	2.557	2.563	**2.557**	**2.557**	**2.557**	**2.557**
	SD	6.13E−03	5.49E−06	**0.00E+00**	1.67E−16	7.47E−03	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
R6	Mean	2.00E+02	1.54E+02	1.14E+02	1.44E+02	1.96E+02	1.67E+02	1.22E+02	1.46E+02	**1.01E+02**
	SD	3.88E+01	3.95E+01	2.80E+01	9.21E+01	3.61E+01	3.67E+01	4.10E+01	4.48E+01	**2.75E+01**
R7	Mean	2.69E+04	2.69E+04	**2.69E+04**	2,69E+04	2.69E+04	**2.69E+04**	**2.69E+04**	**2.69E+04**	**2.69E+04**
	SD	1.36E+01	2.71E+00	**3.88E−12**	7.01E−12	2.57E−02	**3.88E−12**	**3.88E−12**	**3.88E−12**	**3.88E−12**
R8	Mean	3.60E+03	3.04E+03	3.00E+03	3.00E+03	3.49E+03	3.01E+03	3.00E+03	**3.00E+03**	3.00E+03
	SD	5.11E+02	1.74E+01	4.58E+00	5.98E+00	7.70E+02	5.63E+01	3.84E−13	**2.54E−13**	3.43E−13
R9	Mean	1.27E−02	1.46E−02	1.42E−02	1.79E−02	1.27E−02	2.12E−02	**1.27E−02**	1.27E−02	1.27E−02
	SD	9.60E−04	7.73E−03	1.54E−06	1.28E−05	5.91E−04	4.18E−04	**2.17E−18**	2.45E−18	1.13E−09
R10	Mean	1.95E+00	1.69E+00	1.68E+00	1.72E+00	1.88E+00	1.68E+00	**1.68E+00**	**1.68E+00**	1.72E+00
	SD	1.57E−01	1.61E−02	2.44E−04	3.84E−02	1.36E−01	2.68E−04	**2.22E−16**	**2.22E−16**	9.54E−02
R11	Mean	2.36E−01	2.36E−01	2.36E−01	**2.36E−01**	2.36E−01	2.36E−01	2.36E−01	2.36E−01	2.36E−01
	SD	5.17E−04	5.51E−05	2.96E−17	**1.81E−17**	2.96E−17	2.96E−17	2.96E−17	2.96E−17	2.96E−17
R12	Mean	1.68E+01	1.62E+01	1.65E+01	1.67E+01	1.67E+01	1.69E+01	**1.61E+01**	**1.61E+01**	1.69E+01
	SD	3.14E−01	1.21E−01	2.54E−01	1.96E−01	3.00E−01	9.21E−01	**2.69E−15**	**2.69E−15**	1.46E−01
R13	Mean	6.18E+01	6.51E+01	**3.85E+01**	5.83E+01	4.90E+01	4.18E+01	4.10E+01	4.69E+01	4.03E+01
	SD	5.15E+00	8.64E+00	**1.14E+00**	1.06E+02	8.56E+00	5.93E+00	1.94E+00	2.44E+00	8.15E+00
R14	Mean	1.48E+04	1.47E+04	1.47E+04	1.47E+04	1.47E+04	1.49E+04	1.47E+04	1.47E+04	**1.47E+04**
	SD	5.08E+01	1.33E+01	2.28E−10	2.47E−08	1.24E+01	8.65E+00	1.81E−09	2.17E−10	**2.57E−12**
R15	Mean	2.92E+00	4.63E+00	2.67E+00	2.75E+00	2.95E+00	3.65E+00	2.79E+00	3.81E+00	**2.62E+00**
	SD	8.21E−01	3.63E+00	3.31E−01	3.96E−02	2.21E−01	1.14E+00	3.31E−01	7.89E−01	**2.70E−02**
R16	Mean	−3.06E+04	−3.06E+04	−3.07E+04	−3.06E+04	−3.05E+04	−2.98E+04	−3.05E+04	−3.07E+04	**−3.07E+04**
	SD	1.29E+02	2.24E+02	3.88E−12	1.99E+02	2.44E+02	3.19E+02	1.61E−03	1.69E−05	**1.94E−12**
R17	Mean	1.97E+00	1.29E+00	3.41E−02	3.45E−02	4.76E−02	9.21E−01	9.05E−01	1.19E+00	**3.30E−02**
	SD	1.55E+00	1.19E+00	3.04E−03	6.25E−04	7.99E−03	1.28E+00	1.20E+00	1.24E+00	**1.12E−04**
R18	Mean	6.44E+03	6.26E+03	6.17E+03	6.53E+03	6.25E+03	7.41E+03	**5.90E+03**	6.07E+03	5.99E+03
	SD	5.88E+02	5.43E+02	5.36E−02	1.06E+02	6.77E+02	7.49E+02	**1.07E+02**	2.64E+02	2.00E+02
R19	Mean	2.65E+02	2.64E+02	2.64E+02	2.64E+02	2.65E+02	2.64E+02	**2.64E+02**	**2.64E+02**	2.64E+02
	SD	2.46E−01	8.91E−04	1.77E−05	1.46E−02	1.49E−01	1.89E−03	**0.00E+00**	**0.00E+00**	9.71E−02
R20	Mean	1.22E+01	1.14E+00	**5.39E−01**	6.94E−01	5.98E−01	8.95E−01	6.86E−01	1.53E+00	5.84E−01
	SD	3.14E+01	5.65E−01	**1.13E−02**	1.97E−01	1.37E−01	2.75E−01	2.19E−01	2.84E−01	9.17E−02
R21	Mean	3.04E+06	3.01E+06	2.97E+06	2.97E+06	3.02E+06	2.99E+06	**2.97E+06**	**2.97E+06**	2.97E+06
	SD	4.52E+04	4.86E+04	1.45E−04	1.68E+03	1.78E+04	3.07E+03	**5.56E−10**	**5.56E−10**	2.15E+03
R22	Mean	6.72E−11	4.32E−13	2.53E−18	**0.00E+00**	**0.00E+00**	**0.00E+00**	4.16E−17	1.31E−17	**0.00E+00**
	SD	1.38E−10	8.85E−13	7.17E−18	**0.00E+00**	**0.00E+00**	**0.00E+00**	1.15E−16	3.70E−17	**0.00E+00**



(a) Single diode model (SDM) (b) Double diode model (DDM) (c) PV module model (PVMM)

Fig. 10. Equivalent circuit diagrams for photovoltaic cells.

Table 10
Rankings of nine methods on 22 CEC2020 real-world engineering issues on the basis of Friedman's test.

Methods	Ranking
SAO	**2.925**
EO	3.375
SHADE	4.025
LSHADE-SPACMA	4.300
AVOA	4.750
HHO	5.725
MVO	6.150
PSO-sono	6.375
AO	7.375

Table 11
Bounds of different parameters in three PV models.

Parameter	SDM/DDM		PVMM	
	LB	UB	LB	UB
$I_{ph}(A)$	0	1	0	2
$I_{sd1}, I_{sd2}, I_{sd}(\mu A)$	0	1	0	50
n_1, n_2, n	1	2	1	50
$R_{sh}(\Omega)$	0	100	0	2000
$R_s(\Omega)$	0	0.5	0	2

Among them, N denotes the number of measured data, and X refers to the solution vector including the unknown core parameters. Then the objective function of each model is expressed in the following:

- For SDM:

$$\begin{cases} f_i(X, I_L, V_L) = I_{ph} - I_{sd} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n \cdot k}) - 1] - \frac{I_L \cdot R_s + V_L}{R_{sh}} - I_L \\ X = \{I_{ph}, I_{sd}, R_s, R_{sh}, n\} \end{cases}, \tag{24}$$

- For DDM:

$$\begin{cases} f_i(X, I_L, V_L) = I_{ph} - I_{sd1} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n_1 \cdot k}) - 1] \\ -I_{sd2} \cdot [\exp(\frac{(I_L \cdot R_s + V_L) \cdot q}{T \cdot n_2 \cdot k}) - 1] \\ -\frac{I_L \cdot R_s + V_L}{R_{sh}} - I_L \\ X = \{I_{ph}, I_{sd1}, R_s, R_{sh}, n_1, I_{sd2}, n_2\} \end{cases}, \tag{25}$$

- For PVMM:

$$\begin{cases} f_i(X, I_L, V_L) = N_p \cdot I_{ph} - N_p \cdot I_{sd} \cdot \exp[(\frac{(I_L N_s R_s / N_p + V_L) \cdot q}{T \cdot n \cdot k \cdot N_s}) - 1] \\ -\frac{I_L R_s N_s / N_p + V_L}{N_s R_{sh} / N_p} - I_L \\ X = \{I_{ph}, I_{sd}, R_s, R_{sh}, n\} \end{cases}, \tag{26}$$

In this experiment, the benchmark measured current–voltage data are attained from Easwarakhanthan et al. (1986), where a commercial RTC France silicon solar cell with 57 mm diameter (under 1000 W/m^2 at 33 °C). This benchmark dataset is broadly utilized to test the efficacy of algorithms proposed for parameters extraction (Kharchouf et al., 2022; Ridha et al., 2022; Yu et al., 2017). Besides, 1000 total iteration is utilized as the termination criterion and 30 individuals are employed. The best outcomes attained by nine algorithms based on 20 independent executions on SDM, DDM, and PVMM are reported in Tables 12–14, respectively. Some observations are presented as follows:

- For SDM, the best outcomes attained by AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA, and SAO are 5.9041E−02, 1.5286E−03, 1.1551E−03, 1.0946E−03, 1.4178E−03, 9.8602E−04, 9.8602E−04, 9.8602E−04, and 1.0007E−03, respectively. Compared with other competitors except PSO-sono, SHADE, and LSHADE-SPACMA, the SAO algorithm can obtain the best RMSE.
- For DDM, the optimal RMSE over 20 independent runs attained by AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA, and SAO are 6.0578E−02, 1.1095E−03, 9.8598E−04, 1.1942E−03, 1.2445E−03, 9.8366E−04, 9.8248E−04, 9.8248E−04, and 9.8324E−04. Therefore, the SAO outperforms AO, MVO, EO, AVOA, HHO, and PSO-sono.
- For PVMM, the best outcomes provided by AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA, and SAO are 9.331E−02, 2.4942E−03, 2.4266E−03, 2.4357E−03, 1.2887E−02, 2.4251E−03, 2.4251E−03, 2.4251E−03, and 2.4254E−03, respectively.

The statistical results for the RMSE attained by AO, MVO, EO, AVOA, HHO, PSO-sono, SHADE, LSHADE-SPACMA, and SAO on three PV models are reported in Table 15. We can observe that SAO outperforms AO, MVO, AVOA, and HHO. Also, SAO can provide competitive results when compared to PSO-sono, SHADE, and LSHADE-SPACMA. Furthermore, Friedman's test is conducted on nine algorithms on the basis of three PV models. As summarized in Table 16, among these nine algorithms, SAO attains a good ranking. Specifically, according to the average performance, SAO performs better than AO, MVO, EO, AVOA, and HHO, and slightly worse than PSO-sono, SHADE, and LSHADE-SPACMA.

Figs. 11–13 depict the difference between the experimental data attained by SAO and the measured data on SDM and DDM, respectively. As shown in Figs. 11–13, we can observe that the experimental data attained by the SAO algorithm can fit perfectly the measured data.

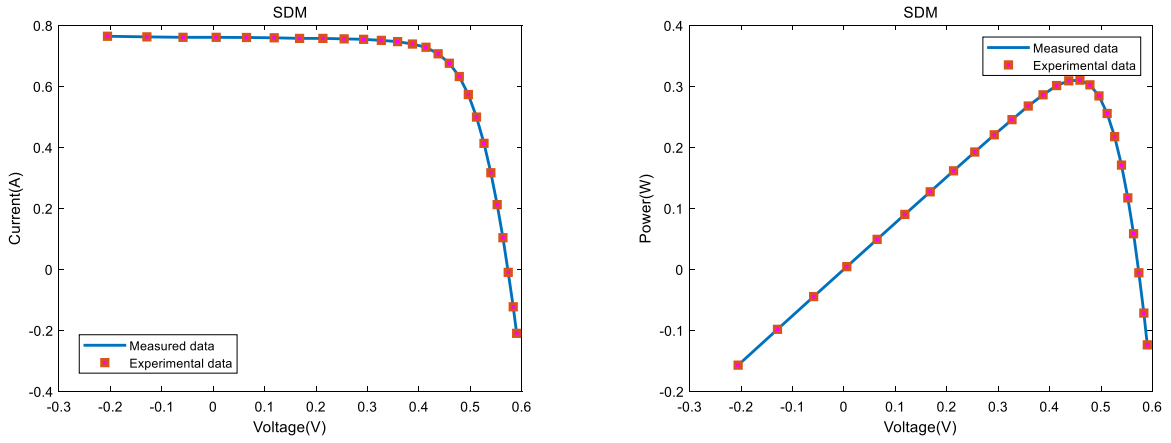
4. Conclusion

In this article, inspired by the sublimation and melting behavior of snow, a novel metaheuristic technique named the snow ablation optimizer (SAO) is raised with the aim of balancing exploitation and exploration and discouraging premature convergence. In the exploration stage of the SAO algorithm, the search agents tend to discover more potential and promising regions, whereas they turn to exploit high-quality solutions around the current best solution during the exploitation stage.

The strength of the SAO algorithm is verified through 29 CEC2017 unconstrained benchmarks and 20 CEC2020 real-world constrained optimization issues. Additionally, the developed technique is employed

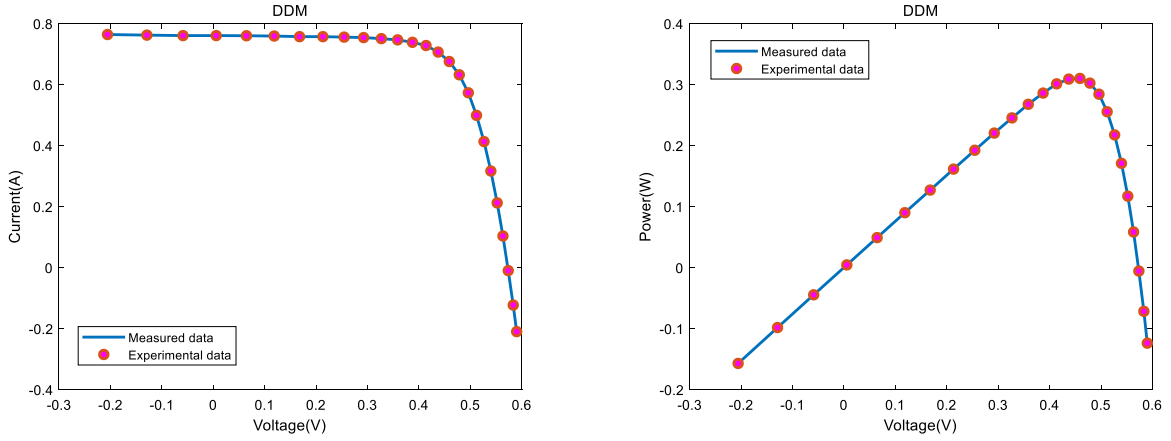
Table 12
The best outcomes attained by nine methods over 20 independent runs on SDM.

Methods	Optimal variables					RMSE
	$I_{ph}(A)$	$I_{sd}(\mu A)$	$R_s(\Omega)$	$R_{sh}(\Omega)$	n	
AO	0.724216	0.607958	0.0516663	27.0936	1.56212	5.9041E−02
MVO	0.76252	0.385612	0.0352405	42.749	1.4995	1.5286E−03
EO	0.760639	0.43864	0.0351438	65.9151	1.51259	1.1551E−03
AVOA	0.760621	0.413816	0.0353887	62.3453	1.50653	1.0946E−03
HHO	0.76005	0.239914	0.0372831	54.3757	1.45171	1.4178E−03
PSO-sono	0.760776	0.323021	0.0363771	53.7185	1.48119	**9.8602E−04**
SHADE	0.760776	0.323021	0.0363771	53.7185	1.48119	**9.8602E−04**
LSHADE-SPACMA	0.760776	0.323021	0.0363771	53.7185	1.48119	**9.8602E−04**
SAO	0.760816	0.295288	0.036731	51.3886	1.4722	1.0007E−03



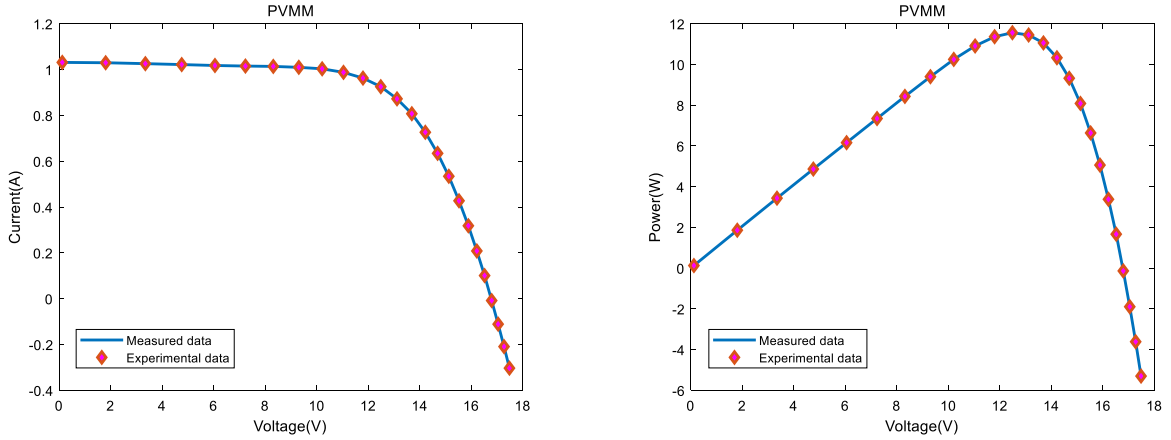
(a) $V-I$ characteristic (b) $V-P$ characteristic

Fig. 11. Comparisons between measured data and experimental data attained by SAO for SDM.



(a) $V-I$ characteristic (b) $V-P$ characteristic

Fig. 12. Comparisons between measured data and experimental data attained by SAO for DDM.



(a) $V-I$ characteristic (b) $V-P$ characteristic

Fig. 13. Comparisons between measured data and experimental data attained by SAO for PVMM.

Table 13
The best outcomes attained by nine methods over 20 independent runs on DDM.

Methods	Optimal variables							RMSE
	$I_{ph}(A)$	$I_{sd1}(\mu A)$	$R_s(\Omega)$	$R_{sh}(\Omega)$	n_1	$I_{sd2}(\mu A)$	n_2	
AO	0.80648	0.680011	0.0574887	18.6472	1.90551	0.161456	1.43113	6.0578E−02
MVO	0.76022	0.234578	0.0373124	55.5148	1.45193	0.352137	2	1.1095E−03
EO	0.760813	0.999599	0.0370484	55.0058	1.99253	0.186609	1.43472	9.8598E−04
AVOA	0.760141	0.703766	0.0373927	58.1967	1.90393	0.258884	1.45913	1.1942E−03
HHO	0.760195	0.434397	0.0351994	76.6725	1.53251	0.0393049	1.44289	1.2445E−03
PSO-sono	0.760778	0.542787	0.0365657	55.6795	1.99993	0.25589	1.46171	9.8366E−04
SHADE	0.760781	0.74934	0.0367404	55.4854	2	0.225974	1.45102	**9.8248E−04**
LSHADE-SPACMA	0.760781	0.74934	0.0367404	55.4854	2	0.225974	1.45102	**9.8248E−04**
SAO	0.760774	0.606521	0.0366366	55.6498	1.96842	0.236754	1.45536	9.8324E−04

Table 14
The best outcomes attained by nine methods over 20 independent runs on PVMM.

Methods	Optimal variables					RMSE
	$I_{ph}(A)$	$I_{sd}(\mu A)$	$R_s(\Omega)$	$R_{sh}(\Omega)$	n	
AO	1.1271983	1.2666945	1.4741654	1281.5935	44.973713	9.331E−02
MVO	1.0295786	4.2065418	1.1817839	1207.1767	49.376275	2.4942E−03
EO	1.030247	3.5670769	1.1992199	1028.4195	48.734337	2.4266E−03
AVOA	1.030276	3.298946	1.208045	986.773	48.43451	2.4357E−03
HHO	1.022035	0.1386828	1.545559	996.4929	38.71288	1.2887E−02
PSO-sono	1.030514	3.482263	1.201271	981.9822	48.6429	**2.4251E−03**
SHADE	1.030514	3.482263	1.201271	981.9822	48.6429	**2.4251E−03**
LSHADE-SPACMA	1.030514	3.482263	1.201271	981.9822	48.6429	**2.4251E−03**
SAO	1.030593	3.433816	1.202717	967.1524	48.58935	2.4254E−03

Table 15
Statistical outcomes of RMSE attained by nine methods on three PV models.

Model	Methods	RMSE			
		Min	Max	Mean	SD
SDM	AO	5.9041E−02	2.6682E−01	1.7922E−01	6.5974E−02
	MVO	1.5286E−03	1.084E−02	5.5717E−03	3.7212E−03
	EO	1.1551E−03	3.8151E−02	1.074E−02	1.6922E−02
	AVOA	1.0946E−03	3.5622E−03	2.2895E−03	8.6407E−04
	HHO	1.4178E−03	3.9497E−02	1.1198E−02	1.2578E−02
	PSO-sono	9.8602E−04	9.8905E−04	9.8644E−04	1.0563E−06
	SHADE	9.8602E−04	9.8602E−04	9.8602E−04	2.8951E−17
	LSHADE-SPACMA	**9.8602E−04**	**9.8602E−04**	**9.8602E−04**	**1.9412E−17**
	SAO	1.0007E−03	2.1816E−03	1.6459E−03	4.0721E−04
DDM	AO	6.0578E−02	3.6267E−01	2.1331E−01	7.1544E−02
	MVO	1.1095E−03	8.8732E−03	3.8886E−03	1.9281E−03
	EO	9.8598E−04	3.3392E−02	4.1524E−03	8.7194E−03
	AVOA	1.1942E−03	4.154E−03	2.4764E−03	7.6928E−04
	HHO	1.2445E−03	6.574E−02	1.134E−02	1.2549E−02
	PSO-sono	9.8366E−04	2.3088E−03	1.3915E−03	4.9853E−04
	SHADE	**9.8248E−04**	**1.5781E−03**	**1.0305E−03**	**1.4488E−04**
	LSHADE-SPACMA	9.8248E−04	1.7509E−03	1.0311E−03	1.5685E−04
	SAO	9.8324E−04	2.6279E−03	1.9411E−03	4.3455E−04
PVMM	AO	9.331E−02	1.4221E+00	5.1337E−01	3.0579E−01
	MVO	2.4942E−03	2.7429E−01	2.4823E−02	7.4307E−02
	EO	2.4266E−03	2.7425E−01	1.9244E−02	6.509E−02
	AVOA	2.4357E−03	2.7431E−01	5.4181E−02	1.0468E−01
	HHO	1.2887E−02	4.3802E−01	1.2805E−01	1.1043E−01
	PSO-sono	2.4251E−03	3.8836E−03	2.478E−03	2.1244E−04
	SHADE	2.4251E−03	2.6081E−03	2.4287E−03	2.5886E−05
	LSHADE-SPACMA	**2.4251E−03**	**2.4294E−03**	**2.4252E−03**	**7.6454E−07**
	SAO	2.4254E−03	2.9972E−02	3.9398E−03	5.2021E−03

to extract the core parameters in photovoltaic systems. The simulation outcomes over 20 independent runs indicate that the SAO algorithm can yield better performance than other competitors such as AO, MVO, EO, AVOA, and HHO. Importantly, SAO can provide competitive results when compared to state-of-the-art algorithms such as PSO-sono, SHADE, and LSHADE-SPACMA. Due to its simple structure and excellent performance on benchmark problems, SAO can be applied to cope with more challenging problems in other scientific fields. Hence, as a future direction, applications of SAO to robot path planning and image segmentation will be investigated.

Table 16
Rankings of nine methods on three PV models based on Friedman's test.

Methods	Ranking
LSHADE-SPACMA	**1.500**
SHADE	**1.500**
PSO-sono	3.000
SAO	4.000
AVOA	5.667
MVO	6.000
EO	6.333
HHO	8.000
AO	9.000

CRediT authorship contribution statement

Lingyun Deng: Conceptualization, Software, Writing – original draft, Visualization. **Sanyang Liu:** Funding acquisition, Conceptualization, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Grant Nos. 12271419).

References

Aaha, B., Sm, C., Hf, D., Ia, D., Mm, E., & Hc, F. (2019). Harris hawks optimization: Algorithm and applications. *Future Generation Computer Systems*, *97*, 849–872.

Abdel-Basset, M., Abdel-Fatah, L., & Sangaiah, A. (2018). Metaheuristic algorithms: A comprehensive review. In *Computational intelligence for multimedia big data on the cloud with engineering applications* (pp. 185–231).

Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers & Industrial Engineering*, *158*, Article 107408.

Abualigah, L., Diabat, A., Mirjalili, S., Abd Elaziz, M., & Gandomi, A. H. (2021). The arithmetic optimization algorithm. *Computer Methods in Applied Mechanics and Engineering*, *376*, Article 113609.

Abualigah, L., Yousri, D., Abd Elaziz, M., Ewees, A. A., Al-Qaness, M. A., & Gandomi, A. H. (2021). Aquila optimizer: a novel meta-heuristic optimization algorithm. *Computers & Industrial Engineering*, *157*, Article 107250.

Arora, S., & Singh, S. (2018). Butterfly optimization algorithm: a novel approach for global optimization. *Soft Computing*.

Deng, L., & Liu, S. (2023). A multi-strategy improved slime mould algorithm for global optimization and engineering design problems. *Computer Methods in Applied Mechanics and Engineering*, *404*, Article 115764.

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation*, *1*, 3–18.

Easwarakhanthan, T., Bottin, J., Bouhouch, I., & Boutrit, C. (1986). Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers. *International Journal of Solar Energy*, *4*(1), 1–12.

Edwards, A. C., Scalenghe, R., & Freppaz, M. (2007). Changes in the seasonal snow cover of alpine regions and its effect on soil processes: A review. *Quaternary International*, *162–163*, 172–181.

Faramarzi, A., Heidarinejad, M., Mirjalili, S., & Gandomi, A. H. (2020). Marine predators algorithm: A nature-inspired metaheuristic. *Expert Systems with Applications*, *152*, Article 113377.

Faramarzi, A., Heidarinejad, M., Stephens, B., & Mirjalili, S. (2020). Equilibrium optimizer: A novel optimization algorithm. *Knowledge-Based Systems*, *191*, Article 105190.

Hansen, N., Müller, S. D., & Koumoutsakos, P. (2003). Reducing the time complexity of the derandomized evolution strategy with covariance matrix adaptation (CMA-ES). *Evolutionary Computation*, *11*(1), 1–18.

Hashim, F. A., Houssein, E. H., Mabrouk, M. S., Al-Atabany, W., & Mirjalili, S. (2019). Henry gas solubility optimization: A novel physics-based algorithm. *Future Generation Computer Systems*, *101*, 646–667.

Hussain, K., Salleh, M., Cheng, S., & Shi, Y. (2018). On the exploration and exploitation in popular swarm-based metaheuristic algorithms. *Neural Computing and Applications*, *31*, 1–19.

Ingber, L. (1993). Simulated annealing: Practice versus theory. *Mathematical and Computer Modelling*, *18*(11), 29–57.

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. In *Proceedings of ICNN'95 - international conference on neural networks, vol. 4* (pp. 1942–1948).

Kharchouf, Y., Herbazi, R., & Chahboun, A. (2022). Parameter's extraction of solar photovoltaic models using an improved differential evolution algorithm. *Energy Conversion and Management*, *251*, Article 114972.

Kumar, A., Wu, G., Ali, M. Z., Mallipeddi, R., Suganthan, P. N., & Das, S. (2020). A test-suite of non-convex constrained optimization problems from the real-world and some baseline results. *Swarm and Evolutionary Computation*, *56*, Article 100693.

Li, S., Chen, H., Wang, M., Heidari, A. A., & Mirjalili, S. (2020). Slime mould algorithm: A new method for stochastic optimization. *Future Generation Computer Systems*, *111*, 300–323.

Li, Q., Liu, S.-Y., & Yang, X.-S. (2020). Influence of initialization on the performance of metaheuristic optimizers. *Applied Soft Computing*, *91*, Article 106193.

Liang, J., Qin, A., Suganthan, P., & Baskar, S. (2006). Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Transactions on Evolutionary Computation*, *10*(3), 281–295.

Martinec, J., & Rango, A. (1986). Parameter values for snowmelt runoff modelling. *Journal of Hydrology*, *84*(3), 197–219.

Meng, Z., Zhong, Y., Mao, G., & Liang, Y. (2022). PSO-sono: A novel PSO variant for single-objective numerical optimization. *Information Sciences*, *586*, 176–191.

Merton, R. C. (1976). Option pricing when underlying stock returns are discontinuous. *Journal of Financial Economics*, *3*(1), 125–144.

Mirjalili, S. (2016). SCA: A Sine cosine algorithm for solving optimization problems. *Knowledge-Based Systems*, *96*.

Mirjalili, S., Mirjalili, S., & Hatamlou, A. (2015). Multi-verse optimizer: a nature-inspired algorithm for global optimization. *Neural Computing and Applications*, *27*.

Mohamed, A. W., Hadi, A. A., Fattouh, A. M., & Jambi, K. M. (2017). LSHADE with semi-parameter adaptation hybrid with CMA-ES for solving CEC 2017 benchmark problems. In *2017 IEEE congress on evolutionary computation* (pp. 145–152).

Onay, F. K., & Aydemr, S. B. (2022). Chaotic hunger games search optimization algorithm for global optimization and engineering problems. *Mathematics and Computers in Simulation*, *192*, 514–536.

Rao, R., Savsani, V., & Vakharia, D. (2012). Teaching–learning-based optimization: An optimization method for continuous non-linear large scale problems. *Information Sciences*, *183*(1), 1–15.

Ridha, H. M., Hizam, H., Mirjalili, S., Othman, M. L., Ya'acob, M. E., & Ahmadipour, M. (2022). Parameter extraction of single, double, and three diodes photovoltaic model based on guaranteed convergence arithmetic optimization algorithm and modified third order Newton Raphson methods. *Renewable and Sustainable Energy Reviews*, *162*, Article 112436.

Sm, A., Smm, B., & Al, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, 46–61.

Srinivas, M., & Patnaik, L. (1994). Genetic algorithms: a survey. *Computer*, *27*(6), 17–26.

Storn, R., & Price, K. (1997). Differential evolution – A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*(4), 341–359.

Su, H., Zhao, D., Asghar Heidari, A., Liu, L., Zhang, X., Mafarja, M., & Chen, H. (2023). RIME: A physics-based optimization. *Neurocomputing*.

Tanabe, R., & Fukunaga, A. (2013). Success-history based parameter adaptation for differential evolution. In *2013 IEEE congress on evolutionary computation* (pp. 71–78).

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, *1*(1), 67–82.

Yao, X., Liu, Y., & Lin, G. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, *3*(2), 82–102.

Yu, K., Liang, J., Qu, B., Chen, X., & Wang, H. (2017). Parameters identification of photovoltaic models using an improved JAYA optimization algorithm. *Energy Conversion and Management*, *150*, 742–753.

Zhou, G., Cui, M., Wan, J., & Zhang, S. (2021). A review on snowmelt models: Progress and prospect. *Sustainability*, *13*(20).