Spiral-based chaotic chicken swarm optimization algorithm for parameters identification of photovoltaic models

Miao Li¹ · Chunquan Li¹ · Zhengyu Huang¹ · Jiehui Huang¹ · Gaige Wang² · Peter X. Liu³

Accepted: 26 June 2021 / Published online: 26 August 2021 © The Author(s), under exclusive licence to Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract
Photovoltaic (PV) systems are becoming increasingly significant because they can convert solar energy into electricity. The conversion efficiency is related to the PV models' parameters, so it is crucial to identify the parameters of PV models. Recently, various metaheuristic methods have been proposed to identify the parameters, but they cannot provide sufficient accurate and reliable performance. To address this problem, this paper proposes a spiral-based chaos chicken swarm optimization algorithm (SCCSO) including three strategies: (1) the information-sharing strategy provides the latest information of the roosters for searching global optimal solution, beneficial to improve the exploitation ability; (2) the spiral motion strategy can enable hens and chicks to move toward their corresponding targets with a spiral trajectory, improving the exploration ability; and (3) a self-adaptive-based chaotic disturbance mechanism is introduced around the global optimal solution to generate a promising solution for the worst chick at each iteration, thereby improving the convergence speed of the chicken flock. Besides, SCCSO is used for identifying different PV models such as the single-diode, the double-diode, and PV module models. Comprehensive analysis and experimental results show that SCCSO provides better robustness and accuracy than other advanced metaheuristic methods.

Keywords Chicken swarm optimization algorithm (CSO) · Spiral shrinkage · Parameters identification · Photovoltaic models

1 Introduction
With the depletion of non-renewable energy, it is more and more urgent to find clean alternative energy, and thus, people pay more attention to the utilization of renewable energy (Quis et al. 2019a). Among all renewable energy sources, solar energy is regarded as inexhaustible clean energy, which has attracted worldwide attention due to its low development cost and huge development potential (Manel et al. 2018). Solar energy can be converted into electricity through PV systems (Zhang et al. 2020b). Nevertheless, the utilization efficiency of solar energy is greatly affected by weather and other environmental factors as PV arrays of PV systems are easily damaged in those tough conditions (Eseye et al. 2017; Zhang et al. 2020a). As a result, to control and optimize PV systems, it is significant to build accurate PV models by measured voltage–current data for evaluating the actual operation behavior of PV arrays (Li et al. 2020; Zaimi et al. 2019).

Several mathematical models have successfully depicted the performance and nonlinear behavior of PV systems. Among them, the single-diode model and double-diode model are widely adopted in various practical problems (Aly et al. 2019; Liu 2020). The accuracy of these PV models mainly depends on their parameters, such as photo-generated current, reverse saturation current, series resistance, and the ideal factor of the diode (Askarzadeh and Rezazadeh 2012). However, the parameters usually change and even are unavailable owing to aging, faults, and unstable operating conditions of the PV systems (Harrou et al. 2018; Jiao et al. 2020). Therefore, to effectively
simulate, evaluate, and control PV systems, it is indispensable to identify the model parameters accurately and reliably.

According to the types of the chosen PV models, two different sets of parameters need to be identified: five for the single-diode model and seven for the double-diode model (Oliva et al. 2014). In essence, the parameter identification of PV models can be regarded as an optimization problem and solved by building an effective objective function. To obtain the corresponding optimal solutions, various optimal algorithms have been designed. They can be roughly divided into deterministic methods and metaheuristic methods (Abdel-Basset et al. 2021a). The deterministic methods need to be combined with several model constraints, including differentiability and convexity (Elsheikh et al. 2019). It is apt for the deterministic methods to fall into the local optimum, and their performance largely depends on their initial solutions (Gao et al. 2018; Ridha et al. 2020). In contrast, the performance of the metaheuristic methods is almost not affected by model constraints (Jordehi 2018), so they are used to solve various parameters identification of PV models (Askarzadeh and Rezagadeh 2012; Oliva et al. 2014).

As an emerging biological metaheuristic algorithm, the most notable feature of CSO is that it divides the population into three groups, establishes different iterative update strategies, and makes full use of the information of different individuals in the population to achieve an effective balance between exploration and development capabilities. This avoids the problem of premature convergence and local convergence of the algorithm (Deb et al. 2020). Furthermore, compared with other biological metaheuristic algorithms, CSO has the advantages of robust control parameters and higher search efficiency (He et al. 2020). More importantly, compared with the simulated annealing algorithm (SA) (El-Naggar et al. 2012), the genetic algorithm (GA) (Jervase et al. 2001), and the particle swarm optimization (PSO) (Ye et al. 2009), CSO can provide better parameters identification performance of PV models.

Although CSO has many advantages summarized above, it still has some shortcomings such as poor ability in jumping out from the local optimum, insufficient information utilization, and low convergence speed. Specifically, the shortcomings are given as follows: (1) the roosters in the rooster sub-flock only learn from a randomly selected rooster, which causes the information of other roosters cannot be fully utilized. (2) The chicks neither learn from the rooster when foraging nor use the latest information in the chicken flock. Thus, they may fail to concentrate near food effectively and degenerate the convergence speed. (3) The ability of CSO to jump out of the local optimum is not strong enough, because hens and chicks also have a high probability of falling into the local optimum when roosters fall into the local optimum.

To overcome these mentioned shortcomings, this paper proposes a spiral-based chaotic chicken swarm optimization algorithm (SCCSO), which involves an information-sharing strategy, a spiral motion strategy, and a self-adaptive-based chaotic disturbance mechanism. First, an information-sharing strategy is employed for roosters to fully use their latest information about food. Two roosters are randomly selected before the update of each rooster. When updating, each rooster draws near the rooster with a better fitness value while keeps away from the other rooster with a worse fitness value, which is propitious to improve the exploitation ability. Second, the spiral motion strategy allows both hens and chicks to move to their respective targets in a spiral trajectory, which expands the searching range and improves the exploration ability. Especially, to fully use the information in the chicken flock, chicks follow roosters in early iteration stages while coming after hens in the later iteration stages. Third, a self-adaptive-based chaotic disturbance mechanism is introduced to improve the convergence speed. By imposing a certain disturbance on the global optimal solution, a new individual is generated to update the individual with the worst fitness value, which effectively improves the convergence speed. To verify the accuracy and robustness of SCCSO, it is compared with other advanced algorithms on different PV models including single-diode, double-diode, and PV module models. Experimental results show that the proposed SCCSO provides better robustness and accuracy compared with other advanced metaheuristic algorithms.

The main contributions of this paper are given as follows.

(1) A new algorithm SCCSO is proposed to solve the parameters identification problems of PV models.

(2) In SCCSO, an information-sharing strategy is designed to search for a globally optimal solution by the latest information in roosters, improving the exploitation ability of SCCSO. Specifically, each rooster obtains the position information from the leader rooster and another randomly selected rooster to update the position of each rooster in an optimal range, thus improving the information utilization and enhancing the exploitation ability.

(3) In SCCSO, instead of a straight trajectory, a spiral motion strategy is proposed to expand the scope of searching for global optimal solutions and improve the exploration ability. Precisely, hens and chicks move toward the roosters with the spiral trajectory instead of a straight trajectory. Chicks have already gathered around the optimal solutions at the later
iteration, so following their mothers to search for food benefits the exploitation ability.

(4) A self-adaptive-based chaotic disturbance mechanism is developed by guiding the chick with the worst fitness value close to the global optimal solution, speeding up the convergence speed of the chicken flock. In detail, after sorting fitness values of chickens, the chick with the worst moves toward the leader rooster, thus accelerating the convergence speed of the chicken flock without tripping into local optimum.

(5) The effectiveness of SCCSO is comprehensively evaluated in the parameters identification problems of various PV models.

The following is the structure of the remaining papers. Section 2 contains an overview of related work. Section 3 shows the modeling and problem formulation of the photovoltaic system. Section 4 briefly introduces the original CSO. Section 5 introduces the proposed SCCSO algorithm in detail. Section 6 analyzes and compares the experimental results. Finally, Sect. 7 concludes the article.

2 Related work

2.1 Applications for metaheuristics

Inspired by nature behavior and physical phenomena, metaheuristic algorithms are developed by combining randomness and some defined principles. Since metaheuristic algorithms perform better than many traditional optimization methods in complex optimization problems, they are widely used in many scientific and engineering problems. For instance, Bernal et al. provided a comparison between the firefly algorithm (FA) and the galactic swarm optimization (GSO) method in the optimization of a fuzzy controller to find a better method for this problem (Bernal et al. 2020). Castillo et al. developed the bee colony optimization algorithm by a generalized type-2 fuzzy logic approach for the optimization of fuzzy controller design (Castillo and Amador-Angulo 2018). Olivas et al. improved the original PSO by using the interval type-2 fuzzy logic and employed it on three benchmark functions (Olivas et al. 2014). Moreover, metaheuristic algorithms are applied in other optimization problems such as spectrum allocation (Padmanaban and Sathiyamoorthy 2020), wireless sensor networks (Mann and Singh 2017), pedestrian detection (Sri Preethaa and Sabari 2020), handwritten signature verification (Hancer et al. 2021), location routing problem (Rabbani et al. 2020), and illness detection (Devarajan et al. 2020).

2.2 Approaches for PV models parameters estimation

Many metaheuristic algorithms have already been applied in PV model parameters estimation. Niu et al. (2014) proposed an improved TLBO (ITLBO) with an elite strategy to obtain parameters of PV models according to the current–voltage results of solar cells. Chen et al. developed a generalized opposite teaching learning-based optimization algorithm (GOTLBO) to solve two-parameter identification problems of solar cell models, including single-diode model and double-diode model (Chen et al. 2016). Yu et al. (2017a) introduced the self-adaptive teaching–learning-based optimization algorithm (SATLBO) to accurately and reliably identify the PV model parameters. Abdel-Basset et al. applied a modified teaching–learning-based optimization (MTLBO) to the accurate and efficient parameter estimation of PV models (Abdel-Basset et al. 2021b). Jaya algorithm proposed by Rao can perform effective exploration and be widely used to identify the parameters of PV models (Venkata Rao 2016). Additionally, other Jaya-based algorithms are also widely utilized in PV models’ parameter identification. For instance, the new Jaya algorithm based on elite opposition (EO-Jaya) (A et al.), an improved JAYA optimization algorithm (IJJAYA) (Yu et al. 2017b), and the performance-guided JAYA (PGJAYA) algorithm (Yu et al. 2019). Yu et al. (2018) designed a multiple learning backtracking search algorithm (MLBSA) for the parameter identification of different PV models. Xiong et al. (2019) proposed a modified search strategy-assisted crossover whale optimization algorithm (MCSWOA) to extract accurate parameters of PV models. Allam et al. (2016) used the moth-flame optimization algorithm (MFO) for the parameter extraction process of a three diode model, a double diode, and the modified double-diode models of the same cell/module. An improved cuckoo search optimization is presented for the parameters extraction of PV cells (Gude and Jana 2020), and a teaching–learning-based artificial bee colony (TLABC) is employed for the solar PV parameters estimation problems (Chen et al. 2018). Besides, an improved brainstorming optimization algorithm (IBSO) (Yan et al. 2019) and hybrid symbiotic differential evolution moth–flame optimization algorithm (HSDEMFO) are also presented to accurately identify parameters of PV models (Wu et al. 2020). However, because the parameter identification problems of PV models possess multimodal and nonlinear features, these problems contain lots of local optimums. Most metaheuristic algorithms are not easy to sink into global optimal solutions when dealing with these problems. Therefore, it is still a challenging task to develop a competitive
algorithm for identifying the parameters of different PV models accurately and reliably.

2.3 CSO for PV models parameters identification

Inspired by the foraging behavior and hierarchical order of chickens in nature, Meng et al. proposed a biological metaheuristic algorithm in 2014, namely the chicken swarm optimization algorithm (CSO), which is employed to solve lots of optimization problems in the real world (Meng et al. 2014a). In CSO, the information in the chicken flock can be fully utilized to achieve the balance between the exploration ability and the exploitation ability. Moreover, CSO can obtain better results than some classical swarm intelligence optimization algorithms such as PSO, DE, and BA when solving twelve benchmark function problems and other practical engineering problems (Meng et al. 2014a). Owing to its advantage in solving optimization problems, CSO has been applied in many aspects such as the optimal design of fuzzy controllers (Amador-Angulo et al. 2021), the collaborative beam forming in wireless sensor network (Al Shayokh and Shin 2017), the evaluation of regional water resources carrying capacity (Yu et al. 2020), the robot path planning (Mu et al. 2016), the fault detection issues (Moldovan et al. 2018), and other issues (Ahmed et al. 2018; Sanchari et al. 2018; Taie et al. 2017).

3 Modeling and problem formulation

Several PV models have been developed to describe the current–voltage characteristics of solar cells in the literature. Only the single-diode model and double-diode model have been put into factual optimization problems. Brief descriptions and objective functions of these models are presented in this section.

3.1 Single-diode model

The equivalent circuit of the single-diode model is shown in Fig. 1.¹

As we can see, it is composed of a current source connected in parallel with the diode, a shunt resistor representing the leakage current passing through, and a series resistor of the load current considering the related loss. According to Kirchhoff’s current law (Diantoro et al. 2018), the output current is calculated by (1).

\[ I_L = I_{PH} - I_D - I_{SH} \]  

where \( I_L \) is the output current, \( I_{PH} \) is the photo-generated current, and \( I_{SH} \) is the current through the shunt resistor. Both of them are calculated by (2) and (3), respectively.

\[ I_D = I_{SD} \exp \left( \frac{V_L + R_L I_L}{kT} \right) - 1 \]  

\[ I_{SH} = \frac{V_L + R_L I_L}{R_{SH}} \]  

where \( I_{SD} \) is the reverse saturation current, \( V_L \) is the output voltage of the port, \( n \) is the ideal factor of the diode, and \( V_t \) is the Thermal voltage calculated by (4).

\[ V_t = \frac{kT}{q} \]  

where \( K \) is Boltzmann constant (1.3806503 \( \times 10^{-23} \) J/K), \( t \) is the working temperature of the diode in Kelvin, \( q \) is the charge amount of meta charge (1.60217646 \( \times 10^{-19} \) C). By combining (1–5), is obtained to reflect the relationship between the output current, the output voltage, and the PV model parameters.

\[ I_L = I_{PH} - I_{SD} \exp \left( \frac{(V_L + R_L I_L)}{nKt}q \right) - 1 - \frac{V_L + R_L I_L}{R_{SH}} \]  

From (5), it can be seen that five unknown parameters \( (I_{PH}, I_{SD}, R_L, R_{SH}, \text{ and } n) \) need identifying for the single-diode model. The more accurate these parameters are identified, the better the characteristics of PV systems can be reflected.

3.2 Double-diode model

Figure 2 shows the equivalent circuit of the double-diode model.

As we can see, it is formed by connecting a diode in parallel at both ends of the diode connected in parallel with the shunt resistor and current source. By combining (2) and (4), the output current can be calculated as.

¹ The Figs. 1, 2, 3, and 4 were drawn with Visio.
PV module, which is composed of a plurality of solar cells connected in parallel and series. The output current \( I_L \) is calculated by (7).

\[
\frac{I_L}{N_p} = I_{PH} - I_{SD} \left[ \exp \left( \frac{V_L + R_L I_L}{n_1 V_t} \right) - 1 \right] - \frac{V_L + R_L I_L}{R_{SH}} \frac{1}{N_S} \]  
(7)

where \( N_p \) and \( N_S \) represent the number of solar cells connected in parallel and series, respectively. Here, the unknown parameters to be identified are \( I_{PH}, I_{SD}, R_L, R_{SH}, \) and \( n \).

\[ fig:3 \] Equivalent circuit of the PV module model

\[ fig:2 \] Equivalent circuit of the double-diode model

\[ I_L = I_{PH} - I_{SD1} \left[ \exp \left( \frac{V_L + R_L I_L}{n_1 V_t} \right) - 1 \right] - I_{SD2} \left[ \exp \left( \frac{V_L + R_L I_L}{n_2 V_t} \right) - 1 \right] - V_L + R_L I_L \frac{1}{R_{SH}} \]  
(6)

where \( I_{D1} \) and \( I_{D2} \) are diffusion current and saturation current, respectively. \( I_{SD1} \) and \( I_{SD2} \) are reverse saturation currents of rectifier diode and compound diode, respectively. \( n_1 \) and \( n_2 \) are the ideal factors of two diodes. From (6), there are seven unknown parameters \( (I_{PH}, I_{SD1}, I_{SD2}, R_L, R_{SH}, n_1, n_2) \) that need identifying for the double-diode model.

3.3 Photovoltaic module model

Figure 3 provides the equivalent circuit of a single-diode PV module, which is composed of a plurality of solar cells connected in parallel and series.

The current \( I_L \) is calculated by (7).

\[ \frac{I_L}{N_p} = I_{PH} - I_{SD} \left[ \exp \left( \frac{V_L + R_L I_L}{n_1 V_t} \right) - 1 \right] - \frac{V_L + R_L I_L}{R_{SH}} \frac{1}{N_S} \]  
(7)

where \( N_p \) and \( N_S \) represent the number of solar cells connected in parallel and series, respectively. Here, the unknown parameters to be identified are \( I_{PH}, I_{SD}, R_L, R_{SH}, \) and \( n \).

\[ fig:2 \] Equivalent circuit of the double-diode model

3.4 Objective function

The PV models’ parameters identification problem can be transformed into optimization problems, minimizing the difference between the experimental and simulated data. The error functions of the single and double-diode models are, respectively, defined by (8) and (9):

\[
\begin{align*}
F_{\text{error}}(V_L, I_L, X) & = I_{PH} - I_{SD} \left[ \exp \left( \frac{(V_L + R_L I_L)}{n_1 K_t} \right) - 1 \right] - V_L + R_L I_L \frac{1}{R_{SH}} - I_L \\
& = X = \{I_{PH}, I_{SD}, R_L, R_{SH}, n\} \\
\end{align*}
\]

\[
\begin{align*}
F_{\text{error}}(V_L, I_L, X) & = I_{PH} - I_{SD} \left[ \exp \left( \frac{(V_L + R_L I_L)}{n_1 K_t} \right) - 1 \right] - V_L + R_L I_L \frac{1}{R_{SH}} - I_L \\
& = X = \{I_{PH}, I_{SD1}, I_{SD2}, R_L, R_{SH}, n_1, n_2\} \\
\end{align*}
\]

The root mean square error (RMSE) (Qais et al. 2020a) is used as the objective function of the overall difference between experimental and simulated current data:

\[
\text{RMSE}(X) = \sqrt{\frac{1}{N_m} \sum_{i=1}^{N_m} F_{\text{error}}^2(V_L, I_L, X)} 
\]

where \( X \) is the solution vector composed of the PV model parameters to be identified, and \( N_m \) is the number of measured \( I-V \) data pairs, \( i \in \{1, 2, \ldots, N\} \).

4 Original CSO algorithm

In 2014, inspired by the foraging behavior and hierarchy of the chicken flock in nature, Meng et al. (2014a) proposed a chicken swarm optimization algorithm (CSO), which is applied to solve various optimization problems. For simplicity, the behavior of the chicken flock is idealized by the following regulations:

(1) The chicken flock is divided into rooster sub-flock, hen sub-flock, and chick sub-flock according to the fitness value of each individual in the chicken flock. The individuals with better fitness values are defined as roosters, followed by hens, and those with the worst fitness values act as chicks. Particularly, the individual with the best fitness value is defined as the leader rooster, and some of the hens are randomly selected as mother hens. The mother–child relationship is established by randomly choosing mother hens and chicks. Similarly, the spouse relationship
between hens and roosters is also done by randomly selecting hens and roosters.

(2) The entire chicken flock is divided into several clusters. Each cluster consists of a rooster, some hens, and some chicks.

(3) The identity of each individual is updated according to the corresponding fitness value at every $G$ generation. When the identity of each individual is updated, the mother–child and the spouse relationship are also updated.

(4) The individuals in each sub-flock search for food under the leadership of roosters and the food is defined as the global optimal solution. Specifically, hens follow their spouses for food and steal food from other hens or roosters. Chicks follow their corresponding mother hens to forage. The better the fitness values of individuals are, the more dominant the individuals are in the process of foraging or stealing food.

In CSO, the position of each individual is regarded as a candidate solution for optimization problems. $N$ is the number of individuals in the chicken flock. Each individual searches food in $D$ space with dimensions and updates its identity at every $G$ generation. The range of the serial numbers of each individual is $\{1, 2, 3, \ldots, N_1, N_2 + 1, \ldots, N_b, N_h + 1, \ldots, N_r\}$, where $N_1$, $N_2$, and $N_r$ are the maximum serial numbers of the roosters, hens, and chicks in each sub-flock after sorting, respectively. More details are shown as follows.

### 4.1 Rooster foraging

Each rooster forages under the guidance of other roosters in the rooster sub-flock. The original position and the updated position of each rooster are defined, respectively, as $X_{ij}$ and $X_{ij}^{\text{new}}$, where $i \in \{1, 2, 3, \ldots, N_r\}$, $j \in \{1, 2, 3, \ldots, D\}$. $X_{ij}$ is the position of a spouse, and $X_{ij}^{\text{new}}$ is the position of the individual in which the $i$th hen wants to steal food, where $c \in \{1, 2, 3, \ldots, N_r\}$, $d \in \{1, 2, 3, \ldots, N_h\}$. The stealing and searching abilities of hens are related to their fitness values. The smaller the fitness value is, the stronger the foraging and stealing abilities are. These foraging behaviors are described by the following (13–15):

$$X_{ij}^{\text{new}} = X_{ij} + S_1 \cdot \text{Rand} \cdot (X_{cij} - X_{ij}) + S_2 \cdot \text{Rand} \cdot (X_{dij} - X_{ij})$$  \hspace{1cm} (13)

$$S_1 = \exp\left(\frac{f_i - f_c}{|f_i| + \zeta}\right)$$  \hspace{1cm} (14)

$$S_2 = \exp(f_a - f_i)$$  \hspace{1cm} (15)

where $\text{Rand}$ is a random number ranging within $[0, 1]$.

### 4.2 Hens foraging

In the chicken flock, hens can obtain food by following their spouses or stealing from other roosters and hens. The original position and the updated position of each hen are defined, respectively, as $X_{ij}$ and $X_{ij}^{\text{new}}$, where $i \in \{N_r + 1, N_r + 2, \ldots, N_h\}$, $j \in \{1, 2, 3, \ldots, D\}$. $X_{ij}$ is the position of a spouse, and $X_{ij}^{\text{new}}$ is the position of the individual in which the $i$th hen wants to steal food, where $c \in \{1, 2, 3, \ldots, N_r\}$, $d \in \{1, 2, 3, \ldots, N_h\}$. The stealing and searching abilities of hens are related to their fitness values. The smaller the fitness value is, the stronger the foraging and stealing abilities are. These behaviors are described by the following (13–15):

$$X_{ij}^{\text{new}} = X_{ij} + \sigma \cdot (X_{mj} - X_{ij})$$  \hspace{1cm} (16)

where $X_{ij}$ and $X_{ij}^{\text{new}}$ are the original position and the updated position of each chick, respectively. For each chick, $i \in \{N_h + 1, N_h + 2, \ldots, N_m\}$ and $j \in \{1, 2, 3, \ldots, D\}$. $X_{mj}$ is the position of the mother hen corresponding to the $i$th chick, where $m \in \{N_r + 1, N_r + 2, \ldots, N_h\}$. $\sigma$ is the following probability, which indicates the probability for each chick following its mother hen to forage. Considering the differences between each chick, $\sigma$ is randomly generated between 0 and 2.

### 4.3 Chicken foraging

Chicks follow their mothers to forage food. The smaller their fitness values, the easier it is for them to find food in the foraging process. The foraging behavior of the chicks is shown in (16):

$$X_{ij}^{\text{new}} = X_{ij} + \sigma \cdot (X_{mj} - X_{ij})$$  \hspace{1cm} (16)

where $X_{ij}$ and $X_{ij}^{\text{new}}$ are the original position and the updated position of each chick, respectively. For each chick, $i \in \{N_h + 1, N_h + 2, \ldots, N_m\}$ and $j \in \{1, 2, 3, \ldots, D\}$. $X_{mj}$ is the position of the mother hen corresponding to the $i$th chick, where $m \in \{N_r + 1, N_r + 2, \ldots, N_h\}$. $\sigma$ is the following probability, which indicates the probability for each chick following its mother hen to forage. Considering the differences between each chick, $\sigma$ is randomly generated between 0 and 2.

### 4.4 Algorithm framework

CSO imitates the foraging behavior and hierarchy of the chicken flock in nature. In CSO, each chicken is considered as a potential solution to a specific optimization problem. The chicken flock is divided into specific optimization problems. The chicken flock is divided into sub-flocks, and each sub-flock uses a different updating mechanism to update the position, generating new solutions. The best solution
obtained at the end of the iteration is the solution to the optimization problem. The pseudocode of CSO is shown in Algorithm 1.

Algorithm 1: Pseudo-code of the original CSO algorithm

Randomly assign initial values to the individuals in the chicken flock;
Define parameters such as \( N_r, N_h, \) and \( N_c \);
\( t = 0; \)
Calculate fitness values for each individual;
While \( t < T_{\text{max}} \) do
   If \((t \% G == 0)\)
      Rank the fitness values and divide the flock into different sub-groups;
   End if
   For \( i = 1: N_r \)
      Update \( X_{i,j} \) with Eq. (11);
   End for
   For \( i = (N_r + 1): N_m \)
      Update \( X_{i,j} \) with Eq. (13);
   End for
   For \( i = (N_m + 1): N_c \)
      Update \( X_{i,j} \) with Eq. (16);
   End for
   If \( f(X_{i,j}) < f(X_{i,j}^{\text{new}}) \)
      \( X_{i,j} = X_{i,j}^{\text{new}}; \)
   Else
      \( X_{i,j} = X_{i,j}^{\text{new}}; \)
   End if
   \( t = t + 1; \)
End while

Here, \( t \) is the current iteration number, and \( T_{\text{max}} \) is the maximum iteration number.

5 Proposed SCCSO algorithm

5.1 Grouping principle

In SCCSO, all individuals are rearranged according to the fitness values of all individuals in ascending order, and thus, the identity of each individual can be determined. The sorted individuals are divided into three sub-flocks: rooster sub-flock, hen sub-flock, and chicken sub-flock. Without loss of generality, SCCSO is used to search the minimum. Therefore, individuals with smaller fitness values are appointed as roosters while with the larger fitness values are chicks, and the rest are hens. Like CSO, SCCSO also follows the same regulation that the smaller the fitness value of chickens, the stronger the ability to find and search for food. The number of individuals in each sub-flock is represented by \( N_r, N_h, \) and \( N_c \), respectively, and their ratio is defined as \( r \) with a value of 1:2:1 in this paper. The total number of individuals is \( N = N_r + N_h + N_c \) and 5.2 improved strategy

5.1.1 Information-sharing strategy

The information-sharing strategy is developed to update roosters that are individuals with smaller fitness values. Specifically, \( X_{r_1} \) and \( X_{r_2} \) are the positions of two roosters that are randomly selected from the rooster sub-flock; without loss of generality, \( X_{r_1} \) and \( X_{r_2} \) are defined as the positions of the rooster with better and worse fitness values, respectively. For each \( i \in \{1, 2, 3, \ldots, N_r\} \), \( X_i \) is the position of the updated rooster at each iteration. It is allowed to follow \( X_{r_1} \) for foraging food; however, it will steal food from \( X_{r_2} \). In this way, each updated rooster has access to share the information of other roosters, thus improving the exploitation ability. From the above analysis, the position of each rooster is updated by (17).

\[
X_i^{\text{new}} = X_i + \text{Rand} n \ast [(X_{r_1} - X_i) - (X_{r_2} - X_i)] \tag{17}
\]

where \( X_i^{\text{new}} \) is the updated position of the \( i \)th rooster. \( \text{Rand} n \) is a standard normal distribution and its mean value and the standard deviation are set to 0 and 1, respectively. If the updated position of a rooster is not good as the original position, the rooster will ignore the updated position and stays at the original position until the next update.

5.1.2 Spiral movement strategy

The spiral movement strategy is introduced to improve the foraging behaviors of the hens and chicks. First, for each hen, its position is defined as \( X_i \), where \( i \in \{N_r + 1, N_r + 2, \ldots, N_r + N_h\} \). The spouse of each hen is randomly selected from the rooster sub-flock whose position is defined as \( X_r \), where \( r \in \{1, 2, 3, \ldots, N_r\} \). The updated position of each hen can be obtained by the distance between the hen and its spouse with the spiral movement strategy as follows:

\[
\begin{align*}
X_i^{\text{new}} &= \theta_1 \ast \text{Rand} n \ast \text{Dis}_{i,r} \ast \exp(\text{bl}) \ast \cos(2\pi l) + X_r \\
\text{Dis}_{i,r} &= X_r - X_i
\end{align*}
\tag{18}
\]

where \( \text{Dis}_{i,r} \) means the distance between each hen and its spouse. \( \theta_1 \ast \cos(2\pi l) \) denotes the spiral movement strategy, which can effectively enhance the exploration capability due to the spiral movement. Specifically, the spiral movement allows each hen to bypass the rooster and explore a wider space, instead of being limited to exploring...
the searching space between them. Here, \( b \) is a constant for defining the shape of the spiral movement, and it is set to \(-2\) in this paper. \( l \) is a random number in the range \([-1, 1]\), shown in (19) as follows. Particularly, \( \theta l \) is a variable parameter and here the value is set to 4.

\[
l = (a - 1) \times \text{Rand} + 1
\]

(19)

\[
a = u \times \left(1 + \frac{\text{FES}}{\text{FESmax}}\right)
\]

(20)

where \( \text{Rand} \) is a random number ranging between \([0, 1]\), \( u \) is set to \(-0.5\). FES is the current iteration number, and \( \text{FESmax} \) is the maximum iteration number.

Afterward, for each chick, its position is also defined as \( X_i \), where \( i \in \{N_r + N_h + 1, N_r + N_h + 1, \ldots, N\} \). At the early iteration stage, each chick follows a rooster for foraging food while at the later iteration stage, it helps a mother hen to seek food and thus move toward the mother hen. Therefore, a rooster and a mother hen are first randomly selected from the rooster and hen sub-flock and their positions are, respectively, defined as \( X_r \) and \( X_m \), where \( r \in \{1, 2, 3, \ldots, N_r\} \), \( m \in \{N_r + 1, N_r + 2, \ldots, N_r + N_h\} \). The updated position of each chick can be obtained by the distance between each chick and its corresponding rooster with the same spiral movement strategy at the early iteration stage:

\[
\begin{cases}
X_{i}^{\text{new}} = \theta_2 \times \text{Rand} \times \text{Dis}_{i,r} \times \exp(bl) \times \cos(2\pi l) + X_r \\
\text{Dis}_{i,r} = X_r - X_i
\end{cases}
\]

(21)

where \( \text{Dis}_{i,r} \) indicates the distance between the \( i \)th chick and its corresponding rooster. \( \theta l \) is a variable parameter and set to 3. The setting values of the other parameters are the same as those mentioned above.

Conversely, at the later iteration stage, the updated position of each chick can be represented by the distance between each chick and its corresponding mother hen with the spiral movement strategy:

\[
\begin{cases}
X_{i}^{\text{new}} = \theta_2 \times \text{Rand} \times \text{Dis}_{i,m} \times \exp(bl) \times \cos(2\pi l) + X_m \\
\text{Dis}_{i,m} = X_m - X_i
\end{cases}
\]

(22)

where \( \text{Dis}_{i,m} \) is the distance between the \( i \)th chick and its corresponding mother hen. In this strategy, both chicks and hens moved toward the roosters with a spiral trajectory at the early integration stage. The whole chicken flock can explore a wider searching range, and gather near the food quickly, thus improving the convergence speed. At the later iteration, chicks have almost gathered around the food. Chicks follow their mothers to search for food, which is beneficial to local exploitation.

### 5.1.3 Self-adaptive-based chaotic disturbance mechanism

The chaotic maps can create a chaotic sequence with better dynamic and statistics properties in a specific order (Ismail et al. 2019). Compared with random numbers created by imposing ordinary probability distributions, the use of chaotic sequences helps the individuals to perform searches at a higher speed (Coelho and Mariani 2008). By increasing the convergence speed and preventing sinking into the local optimum, the chaotic sequence can be used to improve the performance of the solution obtained by the individuals (Lu et al. 2014).

Due to such a fact, a self-adaptive-based chaotic disturbance mechanism is developed for guiding the movement of the individual with the worst fitness value to move toward the leader rooster at each iteration, which can improve the overall quality of the chicken flock. Assume that the position of the individual with the worst fitness value is defined as \( X_w \), and its updated position is defined as \( X_w^{\text{new}} \). In the early iteration stage, \( X_w^{\text{new}} \) is generated by imposing a certain disturbance on the leader rooster whose position is defined as \( X_r \). And in the later iteration stage, \( X_b \) may be very close to the food, so it is necessary to retain more information about \( X_b \) for \( X_w^{\text{new}} \). With the self-adaptive-based chaotic disturbance mechanism, the updating method of \( X_w \) is given as follows:

\[
X_w^{\text{new}} = \begin{cases} 
X_b, & \text{if } \text{Rand}_2 \geq 1 - \frac{\text{FES}}{\text{FESmax}} \\
X_b + \text{Rand}_1 \times (2C_k - 1), & \text{otherwise}
\end{cases}
\]

(23)

\[
C_{k+1} = 4 \times C_k \times (1 - C_k)
\]

(24)

where \( \text{Rand}_1 \) and \( \text{Rand}_2 \) are different random numbers that are generated within \([0, 1]\). \( k \) is the iteration number, \( C_k \) is the value at \( k \)th chaotic iteration, and the initial value of the chaotic sequence is randomly generated within \([0, 1]\). The result obtained by \( 1 - \frac{\text{FES}}{\text{FESmax}} \) is a self-adaptive changed value. According to the comparison between the self-adaptive changed value and \( \text{Rand}_2 \), there is no obvious boundary between pre-iteration and post-iteration so that the natural transition is realized.
5.2 Framework of SCCSO

Based on the above descriptions, the flowchart of SCCSO is shown in Fig. 4, and the pseudocode of SCCSO is summarized in Algorithm 2.

**Algorithm 2:** Pseudo-code of SCCSO

1. Initialize variables, such as $N$, $\mathcal{D}$, $N_r$, $N_h$, and $N_c$;
2. Initialize $X_i$ and calculate $\text{Fit}(X_i)$, $i \in \{1, 2, 3, \ldots, N\}$;
3. $FES = 0$;
4. **While** $FES < FES_{\text{max}}$ **do**
   - Sort $\text{Fit}(X_i)$ and determine the compositions of each group;
   - For $i = 1$: $N_r$ do
     - Update $X_i$ with Eq (17);
   - End for
   - For $i = (N_r + 1)$: $(N_r + N_h)$ do
     - Update $X_i$ with Eq (18);
   - End for
   - For $i = (N_r + N_h + 1)$: $N$ do
     - If $FES/FES_{\text{max}} < \Xi$ then
       - Update $X_i$ with Eq (21);
     - Else
       - Update $X_i$ with Eq (22);
     - End if
   - End for
   - Update $X_N$ with Eq (23);
   - $FES = FES + N$;
   - Replace $X_i$ with $X_i^{\text{new}}$ if $\text{Fit}(X_i^{\text{new}}) < \text{Fit}(X_i)$;
5. **End while**

Here, $\mathcal{D}$ is the dimension of the searching range. $\text{Fit}$ is the fitness function, and $\text{Fit}(X_i)$ expresses the fitness value of each individual in the chicken flock. $\Xi$ is the constant controlling the time that the chicks follow the roosters foraging; here, it is set to 0.

5.3 Time computational complexity of SCCSO

Following is the time computational complexities of SCCSO:

1. Initialization requires $O(N \times D)$ time, where $N$ represents the size of the chicken population, and $D$ is the dimension of the searching range, $N \gg D$.
2. Fitness calculation requires $O(N \times D)$ time.
3. Fitness sorting requires $O(N \times \log N)$ time.
4. Position updating requires $O(N \times D)$ time.

For $N \gg D$, so $O(N \times \log N) \gg O(N \times D)$. In conclusion, the total time complexity is $O(N \times \log N)$ time per generation, the same as CSO’s. Therefore, the total time
complexity of SCCSO for the maximum iteration number is \( O(N \times \log N \times \text{FES}_{\text{max}}) \), where \( \text{FES}_{\text{max}} \) is the maximum iteration number.

5.4 Tests on benchmark functions

To confirm the performance of the proposed algorithms, they are usually tested on a set of mathematics functions with known global optima. Here, eight benchmark functions from the literature (Liang et al. 2005) including unimodal, multimodal, and composite functions are employed for comparison. The unimodal function is used for benchmarking the exploitation ability of algorithms, while the multimodal function is applied to examine the ability of exploration and local optima avoidance. Moreover, the composite functions are utilized to evaluate the performance of algorithms in balancing the exploration and exploitation ability. The searching spaces of these functions are illustrated in Fig. 5.\(^2\) The mathematical formulations of the eight benchmark functions are listed in Table 1.

5.4.1 Tests on unimodal and multimodal benchmark functions

To test the performance of SCCSO on unimodal and multimodal benchmark functions, it is compared with the original CSO and PSO. Their crucial parameters are set according to literature (Meng et al. 2014b; Ye et al. 2009) as follows. In the original CSO, the number of the roosters, hens, chickens, and mother hens are set as \(0.2*N\), \(0.6*N\), \(0.2*N\), and \(0.1*N\), where \(N\) represents the population size. The updating frequency \(G\) is 10, and the following rate \(FL\) ranges in \([0.5, 0.9]\). In PSO, the acceleration coefficients \(c_1\) and \(c_2\) are both 1.49445, and the inertia weight \(\omega\) is 0.729. For the compared algorithms, the population size and maximum iteration are set to 50 and 500, respectively. Table 2 shows the average values and standard deviation values of the three compared algorithms on unimodal and multimodal benchmark functions with 50, 100, and 500 dimensions. To show the statistical results clearly, the overall best and second-best results of the three compared algorithms are underlined in bold italic and bold, respectively.

From Table 2, we can see that SCCSO shows better performance compared with CSO and PSO in solving the high-dimensional unimodal problem and the multimodal problem with the three different dimensions. CSO has better performance than PSO in the multimodal problem with the three different dimensions. However, SCCSO is not good as PSO in solving the unimodal problems with 50-dimension and 100-dimension. According to the no free lunch theorem (Wolpert and Macready 1997), no algorithm is suitable for solving all optimization problems. Note that the PV model parameters identification problem is a multi-mode problem. This is also our motivation to

\(^2\) The Figs. 5, 6, 7, 8, 9, 10, 11 were drawn with MATLAB.
5.4.2 Tests on composite benchmark functions

To verify the effectiveness of SCCSO on solving complex problems, it is employed to optimize six composite benchmark functions in Table 1. The key parameters of CSO and PSO are set the same as in Sect. 5.6.1. Comparative results of the three algorithms are tabulated in Table 3. In Table 3, SCCSO obtains the best performance on F18–F22 among the three compared algorithms. This confirms the effectiveness of SCCSO on solving complex problems. PSO obtains the best results on F23. Compared with the original CSO, SCCSO can achieve better optimization results and robustness, which proves the effectiveness of the designed improving strategies from SCCSO.

6 Experiments on PV models and discussion

To verify the effectiveness of SCCSO, it is applied to the parameters identification of different PV models, such as a single-diode, double-diode, and PV module models. The current and voltage data of various solar cells and modules originate from reference (Easwarakhanthan et al. 1986), which has been vastly applied in various techniques of PV models’ parameters identification (Chen et al. 2016; Gong et al.; Niu et al. 2014; Oliv et al. 2017; Yu et al. 2017b). In
reference, the commercial R.T.C. French silicon solar cells, 57 mm in diameter, operates at 33 °C at an irradiance of 1000. A solar module consisting of 36 polysilicon cells in series, called photowATt-PWP201, operates at 45 °C at an irradiance of 1000. Table 4 shows the rational upper and lower limitations of different PV model parameters.

To demonstrate the competitive performance of SCCSO, it is compared with seven other advanced algorithms which perform excellently in the parameter identification of PV models. They are chicken swarm optimization (CSO) (Meng et al. 2014a), particle swarm optimization (PSO) (Kennedy and Eberhart 2002), moth-flame optimization algorithm (MFO) (Mirjalili 2015), improved JAYA optimization algorithm (IJAYA) (Yu et al. 2017b), performance-guided JAYA algorithm (PGJAYA) (Yu et al. 2019), improved brainstorming optimization algorithm (IBSO) (Yan et al. 2019), and improved moth-flame optimization (IMFO) (Sheng et al. 2019). For fairness, the maximum evaluation time and run number of each algorithm in each experiment are set to 50,000 and 30, respectively. Furthermore, the mentioned seven algorithms except for CSO all have been employed to solve the parameter identification problem of PV models, so their crucial parameters can be set according to the corresponding references in Table 5. Since the original CSO has never been used to solve the problem, we adjust its parameters through several experiments and obtain the best parameters for this problem. The parameter configuration of the involved algorithms is shown in Table 5.

In the following sections, comparisons are conducted first on the performance of each algorithm about accuracy, robustness, and convergence speed by analyzing experimental results and convergence curves. Next, comparisons are made for the best RMSE values gained by the above-mentioned algorithms for 30 runs. Additionally, analysis is performed on the sensitivity of crucial parameters. Then, authentication and discussion are made about the effectiveness of the strategies proposed in SCCSO. To show the statistical results clearly, the overall best and second-best results of RMSE are underlined in bold italic and bold, respectively.

### 6.1 Analysis of data and convergence curves

In this section, we evaluated the performance of all advanced algorithms including accuracy, robustness, and convergence speed by analyzing experimental data and convergence curve. Experimental data are shown in Table 6, where the minimum (Min) and average (Mean) value of RMSE reflects the accuracy and average accuracy of algorithms, respectively. The standard deviation of RMSE (SD) is related to the reliability of algorithms. Additionally, the Wilcoxon signed-rank test with a significant level of 5% (Alcalá-Fdez 2008) is introduced to estimate the difference between SCCSO and other

| Table 3 | Results obtained by composite benchmark functions |
|---------|---------------------------------------------------|
| F      | SCCSO    | CSO      | PSO      |
|        | ave | std   | ave | std | ave | std |
| F18    | 3.00E+00 | 1.04E-14 | 3.00E+00 | 4.71E-07 | 3.00E+00 | 1.06E-14 |
| F19    | - 3.86E+00 | 1.89E-14 | - 3.86E+00 | 2.82E-03 | - 3.86E+00 | 1.96E-14 |
| F20    | - 3.27E+00 | 5.76E-02 | - 3.27E+00 | 6.18E-02 | - 3.27E+00 | 5.94E-02 |
| F21    | - 9.12E+00 | 2.17E+00 | - 9.06E+00 | 2.28E+00 | - 7.66E+00 | 3.02E+00 |
| F22    | - 9.22E+00 | 2.48E+00 | - 8.65E+00 | 2.95E+00 | - 9.10E+00 | 2.53E+00 |
| F23    | - 9.85E+00 | 1.68E+00 | - 9.62E+00 | 2.31E+00 | - 9.95E+00 | 1.78E+00 |

| Table 4 | Parameters ranges of three PV models |
|---------|-------------------------------------|
| Parameter | Single/double diode | PV module |
|          | Lower bound | Upper bound | Lower bound | Upper bound |
| I_PH (A) | 0 | 1 | 0 | 2 |
| I_SD, I_SD1, I_SD2 (μA) | 0 | 1 | 0 | 50 |
| R_L (Ω) | 0 | 0.5 | 0 | 2 |
| R_SH (Ω) | 0 | 100 | 0 | 2000 |
| n, n1, n2 | 1 | 2 | 1 | 50 |
Table 5 Parameter configuration of different algorithms

| Algorithm          | Parameter configuration                                      |
|--------------------|-------------------------------------------------------------|
| SCCSO              | \( N = 100, \ RN = 30, MN = 40, CN = 30, t = 0.25 \)         |
| CSO                | \( N = 100, \ RN = 15, \ HN = 70, \ MN = 50, \ CN = 15, \ G = 20, \ FL \in [0.4, 0.9] \) |
| PSO (Ye et al. 2009)| \( N = 50, c_1 = c_2 = 2, w \in [0.4, 0.9] \)               |
| MFO (Allam et al. 2016)| \( N = 50 \)                                              |
| IMFO (Sheng et al. 2019)| \( N = 100, m = 4, P = 0.4 \)                             |
| IJAYA (Yu et al. 2017b)| \( N = 20 \)                                              |
| PGJAYA (Yu et al. 2019)| \( N = 20 \)                                              |
| IBSO (Yan et al. 2019)| \( N = 50, M = 5, r_1 = r \) and, \( p_{r_1} = 0.8 \)       |

Table 6 Experimental results of different algorithms for three models

| Model            | Algorithm | RMSE | Wilcoxon signed-rank test |
|------------------|-----------|------|--------------------------|
|                  |           | Min  | Mean         | Max          | SD            |
| Single-diode model| SCCSO     | 9.8602E−04 | 9.8602E−04 | 9.8602E−04 | 2.3572E−17    |
|                  | CSO       | 1.0004E−03 | 1.4457E−03 | 2.1624E−03 | 3.7225E−04 | + |
|                  | PSO       | 5.8126E−02 | 1.9598E−01 | 2.9761E−01 | 6.5951E−02 | + |
|                  | MFO       | 9.9272E−04 | 2.0594E−03 | 4.7694E−03 | 6.5366E−04 | + |
|                  | IMFO      | 9.8603E−04 | 9.8999E−04 | 1.0212E−03 | 6.8178E−06 | + |
|                  | IJAYA     | 9.8608E−04 | 9.9132E−04 | 1.0419E−03 | 1.0010E−05 | + |
|                  | PGJAYA    | 9.8602E−04 | 9.8602E−04 | 9.8603E−04 | 8.1518E−10 | + |
|                  | IBSO      | 9.8602E−04 | 9.8605E−04 | 9.8627E−04 | 6.5208E−08 | + |
| Double-diode model| SCCSO     | 9.8248E−04 | 9.8366E−04 | 9.8609E−04 | 1.4171E−06 |
|                  | CSO       | 9.9273E−04 | 1.9532E−03 | 2.9254E−03 | 5.2375E−04 | + |
|                  | PSO       | 8.8789E−02 | 2.2411E−01 | 4.0839E−01 | 7.0402E−02 | + |
|                  | MFO       | 9.8287E−04 | 3.1610E−03 | 3.3398E−02 | 5.7468E−03 | + |
|                  | IMFO      | 9.8298E−04 | 1.0635E−03 | 1.8118E−03 | 1.6481E−04 | + |
|                  | IJAYA     | 9.8388E−04 | 1.0192E−03 | 1.4391E−03 | 9.9613E−05 | + |
|                  | PGJAYA    | 9.8294E−04 | 9.8680E−04 | 1.0042E−03 | 3.8186E−06 | + |
|                  | IBSO      | 9.8554E−04 | 1.0259E−03 | 1.3097E−03 | 7.4098E−05 | + |
| PV module model   | SCCSO     | 2.4251E−03 | 2.4251E−03 | 2.4251E−03 | 1.5950E−17 |
|                  | CSO       | 2.4333E−03 | 2.5366E−03 | 2.7329E−03 | 6.9708E−05 | + |
|                  | PSO       | 1.2988E−01 | 1.3409E+00 | 5.0262E+00 | 1.2713E+00 | + |
|                  | MFO       | 2.4982E−03 | 2.6051E−03 | 2.7428E−03 | 3.4529E−05 | + |
|                  | IMFO      | 2.4256E−03 | 2.4510E−03 | 2.7083E−03 | 6.8809E−05 | + |
|                  | IJAYA     | 2.4252E−03 | 2.4309E−03 | 2.4639E−03 | 7.9201E−06 | + |
|                  | PGJAYA    | 2.4251E−03 | 2.4253E−03 | 2.4289E−03 | 8.3628E−07 | + |
|                  | IBSO      | 2.4251E−03 | 2.4251E−03 | 2.4252E−03 | 2.0783E−08 | + |

In Table 9, the results obtained by SCCSO on the three models are superior to the other seven advanced algorithms in accuracy and reliability. More specifically, in the single-diode model, besides SCCSO, PGJAYA and IBSO also obtain the best Min, and IMFO gains the second best. Although PGJAYA also gets the best Mean, its reliability is inferior to SCCSO. In the double-diode model, SCCSO also performs best as it gets the best in all aspects except for a small gap between SD and the best SD. In the PV module model, SCCSO also achieves the best results in all aspects. Besides, IBSO and PGJAYA obtain the best Min, algorithms. “+” and “≈” represent that the performance of SCCSO is significantly better than or similar to that of other algorithms, respectively.
and IBSO ranks second in accuracy and reliability. Obviously, the Wilcoxon signed-rank test results show the superior performance of SCCSO on all mentioned PV models to other competitive algorithms.

Furthermore, boxplots are applied to visually show the distribution of results obtained by every algorithm in 30 independent runs on three models, as shown in Fig. 6. Note that the symbol ‘+’ represents the disordered values. Comparing the span of solution distributions, we can see that SCCSO performs better in accuracy and robustness than other advanced algorithms.

To investigate the computational efficiency of SCCSO, Fig. 7 shows the average computational time of all algorithms on three models in 30 runs. For each algorithm, the CPU time is obtained on a PC Intel Core 7 Duo 1.80 GHz with 8 GB RAM that runs on Windows 10 with MATLAB R2018a implementation. From Fig. 7, the proposed algorithm and PGJAYA consume the lowest computational overhead among all algorithms. This indicates that SCCSO can obtain superior results under the condition of limited computational overhead.

According to the average RMSE values obtained by each algorithm, the convergence curves are drawn and shown in Fig. 8. Note that the average RMSE values obtained by PSO are beyond the upper limitation of 0.1, so its convergence curves are not presented in those pictures. By amplifying the curves, it is obvious that SCCSO performs excellently in convergence speed as it gets a fast speed and obtains the best convergence values in the end among all algorithms.

To summarize, the above analysis and comparisons of experimental results show that SCCSO performs better in robustness, accuracy, and convergence speed compared with the other competitive algorithms.

6.2 Detailed experimental results analysis

In this section, the best RMSE values and the related parameters identified by different algorithms in 30 independent runs are analyzed detailedly.

6.2.1 Single-diode model

Table 7 presents the analyzed experimental results on the single-diode model, including the adjusted five parameters and the related RMSE values. We can observe that SCCSO, PGJAYA, and IBSO acquire the best RMSE value (9.8602E-04), while IMFO achieves the second-best value (9.8603E-04), followed by IJAYA, MFO, CSO, and PSO. Although the difference of RMSE value corresponding to each algorithm is not much different, a slight difference can improve the accuracy of adjusted parameters significantly for the objective function. Besides, Table 8 shows the individual absolute errors (IAE) between the...
Fig. 8 Convergence curves of SCCSO and compared algorithms for (a) single-diode model, (b) double-diode model, (c) PV module model
Table 7  Comparisons of experimental results on single-diode model

| Algorithm | $I_P$ (A) | $I_{SD}$ (µA) | $R_L$ (Ω) | $R_{SH}$ (Ω) | $n$ | RMSE   |
|-----------|----------|--------------|----------|-------------|-----|--------|
| SCCSO     | 0.7608   | 0.32302      | 0.0364   | 53.7185     | 1.4812 | $9.8602E-04$ |
| CSO       | 0.7607   | 0.35248      | 0.0360   | 56.7696     | 1.4900 | 1.0004E-03 |
| PSO       | 0.4796   | 0.63782      | 0.03199  | 76.0057     | 1.6396 | 1.9598E-01 |
| MFO       | 0.7607   | 0.34312      | 0.0361   | 55.4217     | 1.4873 | 9.9272E-04 |
| IMFO      | 0.7608   | 0.32225      | 0.0364   | 53.6451     | 1.4809 | $9.8603E-04$ |
| IJAYA     | 0.7608   | 0.32312      | 0.0364   | 53.6988     | 1.4812 | $9.8608E-04$ |
| PGJAYA    | 0.7608   | 0.32302      | 0.0364   | 53.7189     | 1.4812 | $9.8602E-04$ |
| IBSO      | 0.7608   | 0.32299      | 0.0364   | 53.7230     | 1.4812 | $9.8602E-04$ |

Table 8  IAE of SCCSO on single-diode model

| Item | Measured data | Calculated data | Simulated current data | Simulated power data |
|------|---------------|-----------------|------------------------|----------------------|
|      | $U$ (V)       | $I$ (A)         | $P$ (W)                | $I_{sim}$ (A)        | $I_{AE}$         | $P_{sim}$ (W) | $I_{AE}$ |
| 1    | 0.2057        | 0.7640          | 0.1572                 | 0.76408770           | 0.00008770       | 0.15717284   | 0.00001804 |
| 2    | 0.1291        | 0.7620          | 0.0984                 | 0.76266309           | 0.00066309       | 0.09845980   | 0.00008560 |
| 3    | 0.0588        | 0.7605          | 0.0447                 | 0.76135531           | 0.00085531       | 0.04476769   | 0.00005029 |
| 4    | 0.0057        | 0.7605          | 0.0043                 | 0.76015309           | 0.00034601       | 0.00432288   | 0.00001970 |
| 5    | 0.0646        | 0.7600          | 0.0491                 | 0.75905521           | 0.00094479       | 0.04903497   | 0.00006103 |
| 6    | 0.1185        | 0.7590          | 0.0899                 | 0.75804235           | 0.00095765       | 0.08982802   | 0.00011348 |
| 7    | 0.1678        | 0.7570          | 0.1270                 | 0.75709165           | 0.0009165        | 0.12703998   | 0.00015380 |
| 8    | 0.2132        | 0.7570          | 0.1614                 | 0.75614136           | 0.00085684       | 0.16120934   | 0.00018306 |
| 9    | 0.2545        | 0.7555          | 0.1923                 | 0.75508687           | 0.00041313       | 0.19216961   | 0.00010514 |
| 10   | 0.2924        | 0.7540          | 0.2205                 | 0.75366388           | 0.00033612       | 0.22037132   | 0.00009828 |
| 11   | 0.3269        | 0.7505          | 0.2453                 | 0.75139097           | 0.00089097       | 0.24562971   | 0.00029126 |
| 12   | 0.3585        | 0.7465          | 0.2676                 | 0.74735385           | 0.00085385       | 0.26792636   | 0.00030611 |
| 13   | 0.3873        | 0.7385          | 0.2860                 | 0.74011722           | 0.00161722       | 0.28664740   | 0.00062635 |
| 14   | 0.4137        | 0.7280          | 0.3012                 | 0.72738222           | 0.00061778       | 0.30091803   | 0.00025557 |
| 15   | 0.4373        | 0.7065          | 0.3090                 | 0.70697265           | 0.00047265       | 0.30915914   | 0.00020669 |
| 16   | 0.4590        | 0.6755          | 0.3101                 | 0.67528015           | 0.00021985       | 0.30995359   | 0.00010091 |
| 17   | 0.4784        | 0.6320          | 0.3023                 | 0.63075827           | 0.00124173       | 0.30175476   | 0.00059404 |
| 18   | 0.4960        | 0.5730          | 0.2842                 | 0.57192836           | 0.00107164       | 0.28367647   | 0.00053153 |
| 19   | 0.5119        | 0.4990          | 0.2554                 | 0.49660702           | 0.00060702       | 0.25574883   | 0.00031073 |
| 20   | 0.5265        | 0.4130          | 0.2174                 | 0.41364879           | 0.00064879       | 0.21778609   | 0.00034159 |
| 21   | 0.5398        | 0.3165          | 0.1708                 | 0.31751011           | 0.00101011       | 0.17139196   | 0.00054526 |
| 22   | 0.5521        | 0.2120          | 0.1170                 | 0.21215494           | 0.00015494       | 0.11713074   | 0.00085544 |
| 23   | 0.5633        | 0.1035          | 0.0583                 | 0.10225131           | 0.00124869       | 0.05759816   | 0.00070339 |
| 24   | 0.5736        | 0.0100          | 0.0057                 | 0.00871754           | 0.00128246       | 0.00500038   | 0.00073562 |
| 25   | 0.5833        | 0.01230         | 0.0717                 | 0.12550741           | 0.00250741       | 0.07320847   | 0.00146257 |
| 26   | 0.5900        | 0.2100          | 0.1239                 | 0.20847233           | 0.00152767       | 0.12299867   | 0.00090133 |
| Sum of IAE |                |                |                       | 0.02152687          | 0.00873078       |         |         |
experimental data and simulated data (Easwarakhanthan et al. 1986). All the IAE values of current are no more than 0.02152687 while those of power is less than 0.00873078, which shows the accuracy of the adjusted five parameters. Additionally, the best-adjusted parameters obtained by SCCSO are applied to construct the \(I–V\) and \(P–V\) curves in Fig. 94. The simulated data acquired by SCCSO (blue five-pointed stars) are extremely unanimous with the experimental data (purple lines) within the whole voltage range.

### 6.2.2 Double-diode model

Table 9 lists the compared results of the adjusted seven parameters as well as the related RMSE values. Among all algorithms, SCCSO obtains the best RMSE value (9.8248E−04) while MFO gains the second-best value (9.8287E−04), followed by PGJAYA, IMFO, IJAYA, IBSO, CSO, and PSO. Table 10 presents the IAE values of current and power, which are less than 0.02127523 and 0.00877664, respectively. Thus, we can conclude that SCCSO can achieve high accuracy parameters.

From the \(I–V\) and \(P–V\) curves presented in Fig. 104, the experimental and simulated data almost coincide with each other within the whole voltage range.

### 6.2.3 Photovoltaic module model

Table 11 presents the five adjusted parameters and related RMSE values. Interestingly, SCCSO and PGJAYA obtain the best RMSE value (2.42507E−03), while the second-best RMSE value (2.42508E−03) is obtained by IBSO. As mentioned above, the slight difference in objective function has a significant impact on the accuracy of the adjusted parameters. Hence, SCCSO has strong competitiveness in the accuracy of parameter adjustment. Table 12 displays the IAE values of current and power, which are no more than 0.04892368 and 0.51688808, respectively. Furthermore, Fig. 11 shows the \(I–V\) and \(P–V\) curves, and we can see that the simulated data are in great agreement with the experimental data within the whole voltage range.

### 6.3 Sensitivity analysis of crucial parameters

As discussed in Sect. 4, \(\tau\) and \(\gamma\) are crucial to improve the performance of SCCSO. \(\tau\) is the ratio of individuals in each sub-flock, and \(\gamma\) is the constant controlling when the chicks follow the rooster foraging. In this section, to investigate the sensitivity of \(\tau\) and \(\gamma\), some experiments are conducted without changing other experimental settings. In
Fig. 10 Comparisons between experimental data and simulated data obtained by SCCSO on double-diode model: a $I$–$V$ characteristic, b $P$–$V$ characteristic
Table 11 Comparisons of experimental results on PV module

| Algorithm | $I_{ph}$ (A) | $I_{sd}$ (μA) | $R_{l}$ (Ω) | $R_{sh}$ (Ω) | n | RMSE |
|-----------|--------------|---------------|-------------|--------------|---|------|
| SCCSO     | 1.0305       | 3.48226       | 1.2013      | 981.9823     | 48.6428 | 2.42507E–03 |
| CSO       | 1.0301       | 3.74181       | 1.1937      | 1067.3017    | 48.9195 | 2.43329E–03 |
| PSO       | 0.9976       | 3.32953       | 1.1262      | 1190.5412    | 49.2876 | 1.34096E+00 |
| MFO       | 1.0293       | 4.32274       | 1.1786      | 1290.2168    | 49.4843 | 2.49818E–03 |
| IMFO      | 1.0303       | 3.53721       | 1.1998      | 1009.1064    | 48.7025 | 2.42560E–03 |
| IJAYA     | 1.0305       | 3.50358       | 1.2004      | 983.5366     | 48.6665 | 2.42525E–03 |
| PGJAYA    | 1.0305       | 3.48261       | 1.2013      | 982.1342     | 48.6432 | 2.42508E–03 |
| IBSO      | 1.0305       | 3.48318       | 1.2012      | 982.4548     | 48.6438 | 2.42508E–03 |

Table 12 IAE of SCCSO on PV module

| Item | Measured data | Calculated data | Simulated current data | Simulated power data |
|------|---------------|-----------------|------------------------|----------------------|
|      | $U$(V) | $I$(A) | $P$(W) | $I_{sim}$(A) | $IAE_I$ | $P_{sim}$(W) | $IAE_P$ |
| 1    | 0.1248 | 1.0315 | 0.1287 | 1.02911916 | 0.00238084 | 0.12843407 | 0.00029713 |
| 2    | 1.8093 | 1.0300 | 1.8636 | 1.02738107 | 0.00261893 | 1.85884058 | 0.00473842 |
| 3    | 3.3511 | 1.026  | 3.4382 | 1.02574180 | 0.00025820 | 3.43763634 | 0.00865262 |
| 4    | 4.7622 | 1.0220 | 4.8670 | 1.02410715 | 0.00210715 | 4.87700309 | 0.01003469 |
| 5    | 6.0538 | 1.0180 | 6.1628 | 1.0229180 | 0.00429180 | 6.18875013 | 0.02598173 |
| 6    | 7.2364 | 1.0155 | 7.3486 | 1.0193068 | 0.00443068 | 7.38062638 | 0.03206218 |
| 7    | 8.3189 | 1.0140 | 8.4354 | 1.01636311 | 0.00025820 | 8.45502304 | 0.01968544 |
| 8    | 9.3097 | 1.0100 | 9.4028 | 1.01049615 | 0.00049615 | 9.40741602 | 0.00461902 |
| 9    | 10.2163 | 1.0035 | 10.2521 | 1.00062897 | 0.00287103 | 10.2227575 | 0.02933130 |
| 10   | 11.0449 | 0.9880 | 10.9124 | 0.98454838 | 0.00345162 | 10.87423839 | 0.03812281 |
| 11   | 11.8018 | 0.9630 | 11.3651 | 0.9592168 | 0.00347832 | 11.32408292 | 0.04105048 |
| 12   | 12.4929 | 0.9255 | 11.5622 | 0.92288882 | 0.00266118 | 11.52893307 | 0.03324588 |
| 13   | 13.1231 | 0.8725 | 11.4499 | 0.87299966 | 0.0009966 | 11.45121264 | 0.01037089 |
| 14   | 13.6983 | 0.8075 | 11.0614 | 0.80727426 | 0.00022574 | 11.05828505 | 0.00390220 |
| 15   | 14.2221 | 0.7265 | 10.3324 | 0.72836348 | 0.00183648 | 10.35847423 | 0.02611858 |
| 16   | 14.6995 | 0.6345 | 9.3268 | 0.63711800 | 0.00263080 | 9.3651003 | 0.03877278 |
| 17   | 15.1346 | 0.5345 | 8.0894 | 0.53621306 | 0.00171306 | 8.11537023 | 0.0292653 |
| 18   | 15.5311 | 0.4275 | 6.6395 | 0.42951132 | 0.00201032 | 6.67087334 | 0.03128089 |
| 19   | 15.8929 | 0.3185 | 5.0619 | 0.31877448 | 0.00027448 | 5.06625098 | 0.00436233 |
| 20   | 16.2229 | 0.2085 | 3.3825 | 0.20738951 | 0.00111049 | 3.36445923 | 0.01801542 |
| 21   | 16.5241 | 0.1010 | 1.6689 | 0.09616717 | 0.00483283 | 1.58907596 | 0.07985184 |
| 22   | 16.7987 | 0.0080 | 0.1344 | 0.00382539 | 0.0002539 | 0.13985567 | 0.00546607 |
| 23   | 17.0499 | – 0.1110 | – 1.8925 | – 0.11093648 | – 0.11955576 | – 0.19145594 | 0.0108296 |
| 24   | 17.2793 | – 0.2090 | – 3.6114 | – 0.20924727 | – 0.31656462 | – 0.42725825 | 0.0376266 |
| 25   | 17.4885 | – 0.3030 | – 5.2990 | – 0.30086359 | – 0.00213641 | – 5.26165284 | 0.0376266 |
| Sum of IAE | | | | 0.04892368 | 0.5168808 |

previous experiments, $r$ is set to 1:2:1 and is 0.4. Here, different values are set for $r$ :3:14:10, 1:1:1, and 1:3:1, and different: 0.3 and 0.5. Table 13 displays the experimental results of SCCSO using combinations of $r$ and for three models.

From Table 13, the variants all obtain the best results on the single-diode model and PV module models, but only SCCSO achieves the best results on the double-diode model with the best stability. Hence, by comparing the results with the combinations, we can conclude that the combination of $r$ = 1:2:1 and = 0.4 is a suitable choice for SCCSO.
6.4 Authentication of strategy effectiveness

To verify the effectiveness of the proposed strategies, it is essential to conduct several experiments. In this section, three variants are proposed based on SCCSO, which are SCCSO without the information-sharing strategy instead of the original rooster optimization equation in CSO (denoted as SCCSO-1), SCCSO without spiral motion trajectory equation instead of the original update equation of hen and chicks (denoted as SCCSO-2), and SCCSO without the self-adaptive-based chaotic disturbance mechanism (denoted as SCCSO-3). In the experiments, the parameter
settings stay the same as mentioned. Table 14 presents the experimental results of SCCSO and its three variants on PV models in 30 independent runs. The results obtained by SCCSO perform better in all PV models than SCCSO-1. Although SCCSO-2 obtained the best results on the single-diode model and PV module models, the results on the double-diode model are not as good as the results obtained by SCCSO. As for SCCSO-3, it obtains the best results on the single-diode model, but the performance of the other two models is not as good as SCCSO. Comparisons indicate the extraordinary performance of SCCSO in terms of accuracy, and avoiding local optimal. As a consequence, using a single strategy is insufficient to achieve satisfactory results, but combining all proposed strategies leads to the best results.

### 6.5 Discussion

The above experimental results indicate that compared with the advanced algorithms, SCCSO obtains better performance on robustness and accuracy. The reason is due to the three strategies we used in SCCSO, covering the information-sharing strategy, the spiral motion strategy, and the self-adaptive-based chaotic disturbance mechanism. The specific discussion shows as follows:

1. The information-sharing strategy is developed to overcome the shortness of low utilization of the information. In CSO, roosters only learn from a randomly selected rooster so that the information of other roosters cannot be fully utilized. By the information-sharing strategy, the roosters get information from the leader rooster and another randomly selected rooster. Then, the roosters update their positions in an optimal range in the chicken flock, thus improving the information utilization and ensuring the exploitation ability.

2. The spiral motion strategy is designed to decrease the probability of falling into the local optimum. In CSO, when the cock falls into the local optimum, the hens and the chicks are affected by the rooster and have a high probability of falling into the local optimum. By the spiral motion strategy, the searching range for hens and chicks is expanded, which is favorable to get rid of the local optimum and balance the early exploration and the later exploitation. Specifically, hens and chicks move to the roosters with the spiral trajectory, so the whole chicken flock can gather near roosters at a fast speed, accelerating the convergence speed. When at the later iteration, chicks have already gathered around the food. At this time, chicks follow their mothers to search for food benefits the exploitation ability.

3. The self-adaptive-based chaotic disturbance mechanism is used to accelerate the convergence speed of the chicken flock. In CSO, chicks neither learn from rooster when foraging nor use the latest information in the chicken flock. Thus, they may fail to concentrate near food effectively, degenerating the convergence speed of the chicken flock. By the self-adaptive-based chaotic disturbance mechanism, the worst chick moves to the position of the leader rooster directly. After sorting fitness values of chickens, the new worst chick also moves to the new leader rooster, thus accelerating the convergence speed of the chicken flock without tripping into local optimum.

In summary, SCCSO is an effective choice to solve the problem of nonlinear multimodal PV model parameter identification. However, according to the no free lunch
7 Conclusion

It is crucial to determine accurate and reliable PV model parameters for the evaluation and optimization of PV systems. However, the parameter identification problems of PV models have multimodal and nonlinear features so that most existing metaheuristic algorithms fail to obtain globally optimal solutions. To overcome this problem, this paper proposed a spiral-based chaos chicken swarm optimization algorithm (SCCSO) to identify the parameters of PV models. The experimental results show that SCCSO performs better in robustness and accuracy than other advanced metaheuristic algorithms on the single-diode, double-diode, and PV module models. The reasons behind the fact are (1) SCCSO develops the information-sharing strategy to provide the latest information of roosters for exploring local optimal solutions, which is beneficial to improve the exploitation ability; (2) SCCSO develops the spiral motion strategy to expand the searching range of hens and chicks, improving the exploration ability; (3) SCCSO introduces the self-adaptive-based chaotic disturbance mechanism to update the worst chick based on the global optimal solution, improving the convergence speed of the chicken flock; (4) SCCSO rationally combines the above three strategies and thus achieves satisfactory results. Thus, the proposed SCCSO is a promising candidate technique for the parameter identification problems of PV models.

In the future, we plan to use SCCSO for solving other multi-objective and constrained optimization problems in the power system, such as the fault ride-through improvement and the maximum power point tracking of speed wind generators (Qais et al. 2019b, 2020b).

Author contributions ML was involved methodology, software, formal analysis, writing—original draft. CL helped in funding acquisition, writing—review and editing, supervision. ZH contributed to methodology, validation, writing—review and editing. JH was involved in conceptualization, writing—review and editing. LW helped in writing—review and editing, supervision. PXL contributed to writing—review and editing, supervision.

Funding This work was supported in part by the National Natural Science Foundation of China under Grants 61863028, 81660299, and 61503177, and in part by the Science and Technology Department of Jiangxi Province of China under Grants 20204ABC03A39, 20161ABC21007, 2017BBE50071, and 20171BAB202033.

Data availability All data generated or analyzed during this study are included in this published article.

Code availability My manuscript has data included as electronic supplementary material.

Declarations

Conflict of 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.

Informed consent Written informed consent for publication was obtained from all participants.

References

Abdel-Basset M, El-Shahat D, Chakrabortty RK, Ryan M (2021a) Parameter estimation of photovoltaic models using an improved marine predators algorithm. Energy Convers Manag. https://doi.org/10.1016/j.enconman.2020.113491

Abdel-Basset M, Mohamed R, Chakrabortty RK, Sallam K, Ryan MJ (2021b) An efficient teaching-learning-based optimization algorithm for parameters identification of photovoltaic models: analysis and validations. Energy Convers Manag. https://doi.org/10.1016/j.enconman.2020.113614

Ahmed K, Hassanien AE, Ezzat E, Bhattacharyya S (2018). Swarming behaviors of chicken for predicting posts on facebook branding pages. In: The international conference on advanced machine learning technologies and applications (AMLTA2018)

Al Shayyokh M, Shin SY (2017) Bio inspired distributed WSN localization based on chicken swarm optimization wireless personal communications. Wireless Pers Commun 97:5691–5706. https://doi.org/10.1007/s11277-017-4803-1

Alcalá-Félez J et al (2008) KEEL: a software tool to assess evolutionary algorithms for data mining problems. Soft Comput 13:307–318. https://doi.org/10.1007/s10009-008-0323-y

Allam D, Yousri DA, Eteiba MB (2016) Parameters extraction of the three diode model for the multi-crystalline solar cell/module using moth-flame optimization algorithm. Energy Convers Manag 123:535–548. https://doi.org/10.1016/j.enconman.2016.06.052

Aly SP, Ahzi S, Barth NJAE (2019) An adaptive modelling technique for parameters extraction of photovoltaic devices under varying sunlight and temperature conditions. Appl Energy 236:728–742. https://doi.org/10.1016/j.apenergy.2018.12.036

Amador-Angulo L, Castillo O, Peraza C, Ochoa P (2021) An efficient chicken search optimization algorithm for the optimal design of fuzzy controllers. Axioms 10(1):30. https://doi.org/10.3390/axioms10010030

Askarzadeh A, Rezaazadeh A (2012) Parameter identification for solar cell models using harmony search-based algorithms. Sol Energy 86:3241–3249. https://doi.org/10.1016/j.solener.2012.08.018

Bernal E, Lagunes ML, Castillo O, Soria J, Valdez F (2020) Optimization of Type-2 fuzzy logic controller design using the GSO and FA algorithms. Int J Fuzzy Syst 23:42–57. https://doi.org/10.1007/s40815-020-00976-w

Castillo O, Amador-Angulo L (2018) A generalized type-2 fuzzy logic approach for dynamic parameter adaptation in bee colony...
optimal parameter identification of solar cell models using a new algorithm based on PSO and SVM. Renew Energy 118:357–367. https://doi.org/10.1016/j.renene.2017.11.011

Devarajan D, Ramesh SM, Gomathy B (2020) A metaheuristic biologically inspired algorithm for engineering design optimization. Soft Comput 24:5313–5331. https://doi.org/10.1007/s00500-019-04280-0

Devarajan D, Ramesh SM, Gomathy B (2020) A metaheuristic segmentation framework for the detection of retinal disorders from fundus images using a hybrid ant colony optimization. Soft Comput 24:13347–13356. https://doi.org/10.1007/s00500-020-04753-7

Diantoro M, Suprayogi T, Hidayat A, Taufiq A, Fuad A, Suryana R (2018) Shockley’s equation fit analyses for solar cell parameters from I–V curves. Int J Photoenergy 2018:1–7. https://doi.org/10.1155/2018/9214820

Easwarakhanthan T, Bottin J, Bouhouch I, Boutrit C (1986) Nonlinear minimization algorithms for determining the solar cell parameters with microcomputers. Int J Solar Energy 4:1–12. https://doi.org/10.1016/0143-7912(86)90089-3

El-Naggar KM, Alfranshi MR, Alhajri MF (2012) Al-Othman AKJSE simulated annealing algorithm for photovoltaic parameter identification. Solar Energy 86:266–274. https://doi.org/10.1016/j.solener.2011.09.032

Elsheikh AH, Sharshir SW, Abd Elaziz M, Kabeel AE, Wang G, Zhang H (2019) Modeling of solar energy systems using artificial neural network: a comprehensive review. Sol Energy 180:622–639. https://doi.org/10.1016/j.solener.2019.01.037

Esseye AT, Zhang J, Zheng DJRE (2017) Short-term photovoltaic solar power forecasting using a hybrid Wavelet-PSO-SVM model based on SCADA and Meteorological information. Renew Energy 115:357–367. https://doi.org/10.1016/j.renene.2017.11.011

Gao X, Cui Y, Hu J, Xu G, Wang Z, Qu J, Wang H (2018) Parameter extraction of solar cell models using improved shuffled complex evolution algorithm. Energy Convers Manag 157:460–479. https://doi.org/10.1016/j.enconman.2017.12.033

Gong L, Cao W, Zhao J. An improved PSO algorithm for high accurate parameter identification of PV model. In: International conference on environment and electrical engineeringindustrial and commercial power systems Europe

Gude S, Jana KC (2020) Parameter extraction of photovoltaic cell using an improved cuckoo search optimization. Sol Energy 204:280–293. https://doi.org/10.1016/j.solener.2020.04.036

Gx A, Jz A, Ds B, Lz C, Xyjec A (2020) Parameter extraction of solar photovoltaic models with an either-or teaching learning based algorithm. Energy Convers Manag 224:113395. https://doi.org/10.1016/j.enconman.2020.113395

Hancer E, Hodoshinsky I, Sarin K, Slezkin A (2021) A wrapper metaheuristic framework for handwritten signature verification. Soft Comput. https://doi.org/10.1007/s00500-021-05717-1

Harrou F, Sun Y, Taghezouit B, Saidi A, Hamlati M-E (2018) Reliable fault detection and diagnosis of photovoltaic systems based on statistical monitoring approaches. Renewable Energy 116:22–37. https://doi.org/10.1016/j.renene.2017.09.048

He Y, Wang M, Guan F, Zhao WJEP (2020) Research on the method of electricity demand analysis and forecasting: the case of China. Electric Power Syst Res 187:106408. https://doi.org/10.1016/j.epsr.2020.106408

Ismail FH, Housseni EH, Hassanien AE (2019) Chaotic Bird swarm optimization algorithm. In: Hassanien AE, Tolba MF, Shaalan K, Azar AT (eds.). In: Proceedings of the international conference on advanced intelligent systems and informatics 2018, vol 845. Advances in Intelligent Systems and Computing. pp 294–303. https://doi.org/10.1007/978-3-319-99010-1_27

Jervase JA, Bourdoucen H, Al-Lawati AJMS (2001) Technology solar cell parameter extraction using genetic algorithms. Measurement Sci Technol 12:1922–1925. https://doi.org/10.1088/0957-0233/12/11/322

Jiao S et al (2020) Orthogonally adapted Harris hawks optimization for parameter estimation of photovoltaic models. Energy. https://doi.org/10.1016/j.energymodel.2020.117804

Jordehi AR (2018) Enhanced leader particle swarm optimisation (ELPSO): an efficient algorithm for parameter estimation of photovoltaic (PV) cells and modules. Solar Energy 159:78–87. https://doi.org/10.1016/j.solener.2017.10.063

Kennedy J, Eberhart R (2002) Particle Swarm Optimization. In: 1cnn95-international conference on neural networks

Li S, Gu Q, Gong W, Ning BJEC (2020) An enhanced adaptive differential evolution algorithm for parameter extraction of photovoltaic models. Energy Convers Manag 205:112443. https://doi.org/10.1016/j.enconman.2019.112443

Liang JJ, Suganthan PN, Deb K (2005) Novel composition test functions for numerical global optimization. In: 2005 IEEE Swarm intelligence symposium. IEEE, New York.

Liu Y et al (2020) Horizontal and vertical crossover of Harris hawk optimizer with Nelder-Mead simplex for parameter estimation of photovoltaic models. Energy Convers Manag. https://doi.org/10.1016/j.enconman.2020.113211

Lu H, Wang X, Fei Z, Qu M (2014) The effects of using chaotic map on improving the performance of multiobjective evolutionary algorithms. Math Prob Eng 2014:1–16. https://doi.org/10.1155/2014/924652

Manel M, Anis S, Faouzi MMJEC (2018) Particle swarm optimisation with adaptive mutation strategy for photovoltaic solar cell/module parameter extraction. Energy Convers Manag 175:151–163. https://doi.org/10.1016/j.enconman.2018.08.081

Mann PS, Singh S (2017) Improved metaheuristic-based energy-efficient clustering protocol with optimal base station location in wireless sensor networks. Soft Comput 23:1021–1037. https://doi.org/10.1007/s00500-017-2815-0

Meng X, Liu Y, Gao X, Zhang H (2014a) A new bio-inspired algorithm: chicken swarm optimization. In: advances in swarm intelligence. lecture notes in computer science. pp 86–94. https://doi.org/10.1007/978-3-319-11857-4_10

Meng X, Liu Y, Gao X, Zhang H (2014b) A new bio-inspired algorithm: chicken swarm optimization. In: Tan Y, Shi Y, Coelho CAC (eds.). Advances in swarm intelligence, Pt1, vol 8794. Lecture Notes in Computer Science. pp 86–94

Mirjalili SJK-BS (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl-Based Syst 89:228–249. https://doi.org/10.1016/j.knosys.2015.07.006

Moldovan D, Chifu V, Pop C, Ciocarla I, Anghel I, Salomie I (2018). Chicken swarm optimization and deep learning for manufacturing processes. In: Gasner P, Cebuc E (eds). In: 2018 17th Roedunet IEEE international conference: networking in education and research. RoEdunet international conference

Mu Y, Zhang L, Chen X, Gao X, Ieke (2016). Optimal trajectory planning for robotic manipulators using chicken swarm optimization. In: 2016 8th international conference on intelligent
human-machine systems and cybernetics. pp 369–373. https://doi.org/10.1109/dmasc.2016.107
Niu Q, Zhang H, Li K (2014) An improved TLBO with elite strategy for parameters identification of PEM fuel cell and solar cell models. Int J Hydrog Energy 39:3837–3854. https://doi.org/10.1016/j.ijhydene.2013.12.110
Oliv D, Mohamed AEA, Hassanien AEJAE (2017) Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm. Appl Energy 200:141–154. https://doi.org/10.1016/j.apenergy.2017.05.029
Oliva D, Cuevas E, Pajares G (2014) Parameter identification of solar cells using artificial bee colony optimization. Energy 72:93–102. https://doi.org/10.1016/j.energy.2014.05.011
Olivas F, Valdez F, Castillo O, Melin P (2014) Dynamic parameter adaptation in particle swarm optimization using interval type-2 fuzzy logic. Soft Comput 20:1057–1070. https://doi.org/10.1007/s00500-014-1567-3
Padmanaban B, Sathiyaamoorthy S (2020) A metaheuristic optimization model for spectral allocation in cognitive networks based on ant colony algorithm (M-ACO). Soft Comput 24:15551–15560. https://doi.org/10.1007/s00500-020-04882-z
Qais M, Hassanien HM, Alghuwainem S, Nooh ASJE (2019a) Coyote optimization algorithm for parameters extraction of three-diode photovoltaic models of photovoltaic modules. Energy 187:116001. https://doi.org/10.1016/j.energy.2019.116001
Qais MH, Hassanien HM, Alghuwainem S (2019b) Enhanced salp swarm algorithm: application to variable speed wind generators. Eng Appl Artif Intell 80:82–96. https://doi.org/10.1016/j.engappai.2019.01.011
Qais MH, Hassanien HM, Alghuwainem S (2020a) Parameters extraction of three-diode photovoltaic model using computation and Harris Hawks optimization. Energy. https://doi.org/10.1016/j.energy.2020.117040
Qais MH, Hassanien HM, Alghuwainem S (2020b) Whale optimization algorithm-based Sugeno fuzzy logic controller for fault ride-through improvement of grid-connected variable speed wind generators. Eng Appl Artif Intell. https://doi.org/10.1016/j.engappai.2019.103328
Rabbani M, Nikouhin A, Farrokh-Arsl (2020) Using modified metaheuristic algorithms to solve a hazardous waste collection problem considering workload balancing and service time windows. Soft Comput 25:1885–1912. https://doi.org/10.1007/s00500-020-05261-4
Ridha HM, Heidari AA, Wang M, Chen H (2020) Boosted mutation-guided JAYA algorithm for parameters identification of single-diode solar cell models. Energy Convers Manag. https://doi.org/10.1016/j.enconman.2019.112875
Sancha D, Kari T, Karuna K, Pinakeshwar MJE (2018) Impact of electric vehicle charging station load on distribution network. Energies 11:178. https://doi.org/10.3390/en11101178
Sheng H, Li C, Wang H, Yan Z, Xiong Y, Cao Z, Kuang Q (2019) Parameters extraction of photovoltaic models using an improved moth-flame optimization. Energies. https://doi.org/10.3390/en12183527
Sri Preethaa KR, Sabari A (2020) Intelligent video analysis for enhanced pedestrian detection by hybrid metaheuristic approach. Soft Comput 24:12303–12311. https://doi.org/10.1007/s00500-020-04674-5
Taie SA, Ghonaim W, Ieee (2017). Title CSO-Based Algorithm With Support Vector Machine for brain tumor’s disease diagnosis. In: 2017 IEEE International conference on pervasive computing and communications workshops. International conference on pervasive computing and communications
Venkata Rao R (2016) Jaya: a simple and new optimization algorithm for solving constrained and unconstrained optimization problems. Int J Ind Eng Comput. https://doi.org/10.5267/j.ijiec.2015.8.004
Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evolut Comput 1(1):67–82. https://doi.org/10.1109/4235.585893
Yu K, Liang JJ, Qu B, Yue C, Ge S, Chen X, Liang J (2019) A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module. Appl Energy 237:241–257. https://doi.org/10.1016/j.apenergy.2019.01.008
Yu X, Xie J, Jiang R, Zuo G, Liang J (2020) Assessment of water resource carrying capacity based on the chicken swarm optimization-projection pursuit model. Arab J Geosci. https://doi.org/10.1007/s12517-019-5010-z
Zaimi M, Achouby HE, Ibral A, Assaid EMJSE (2019) Determining combined effects of solar radiation and panel junction temperature on all model-parameters to forecast peak power and photovoltaic yield of solar panel under non-standard conditions. Sol Energy 191:341–359. https://doi.org/10.1016/j.solener.2019.09.007
Zhang J, Liu Y, Li Y, Ding K, Wu JJE (2020a) A reinforcement learning based approach for on-line adaptive parameter extraction of photovoltaic array models. Energy Conver Manag 214:112875. https://doi.org/10.1016/j.enconman.2020.112875
Zhang Y, Ma M, Jin ZJE (2020b) Comprehensive learning Jaya algorithm for parameter extraction of photovoltaic models. Energy. https://doi.org/10.1016/j.energy.2020.118644

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