Extremal Nelder–Mead colony predation algorithm for parameter estimation of solar photovoltaic models

Boyang Xu1 | Ali Asghar Heidari2 | Siyang Zhang3 | Huiling Chen1 | Qike Shao1

1Department of Computer Science and Artificial Intelligence, Wenzhou University, Wenzhou, China
2School of Surveying and Geospatial Engineering, College of Engineering, University of Tehran, Tehran, Iran
3School of Information Engineering, Wenzhou Business College, Wenzhou, China

Abstract
Measurement data based on current and voltage of photovoltaic (PV) systems and the establishment of more accurate and stable solar system models are of typical significance for the design, control, evaluation and optimization of PV systems. Accurate and stable parameter evaluation for PV systems needs to be based on more efficient optimization techniques to achieve efficient energy conversion from solar energy. Therefore, this paper proposes a novel and efficient optimization technique enhanced colony predation algorithm to solve the complex PV parameter identification problem named ECPA. By fusing extremal optimization strategy and Nelder-Mead simplex method enables ECPA to further develop in the neighborhood of potential optimal solutions while improving the position of inferior agent candidates, and finally has the ability to search globally beyond the local optimum. To verify the optimization efficiency of ECPA, the first part verifies the efficiency of ECPA in solving complex high-dimensional and multimodal problems by conducting competitive comparison experiments at the IEEE CEC 2020 benchmark case. In the second part, ECPA is compared with nine similar published state-of-the-art algorithms, and competitive tests for PV parameter identification under single diode model, double diode model, triple diode model and PV module model (PV) are conducted. Finally, we focused on three different commercial PV models (thin film ST40, monocrystalline SM55, and multicrystalline KC200GT) to test the accuracy of ECPA in evaluating PV parameters. The test results show that ECPA is able to maintain a high level of accuracy and stability when dealing with commercial PV models in complex environments. The experimental results demonstrate that ECPA outperforms other algorithms in terms of data fitting, stability, convergence speed and convergence accuracy. All the competitive experimental results show that ECPA can be a novel technique with the best performance for identifying the parameters to be determined in solar PV systems.
1 | INTRODUCTION

As the supply of large areas of nonrenewable energy sources is decreasing, the use of renewable energy sources is becoming more and more important. And nowadays, solar energy has become one of the most valuable and efficient energy sources.1 Solar photovoltaic (PV) systems have become the main energy technology in the world after wind energy, which is a near-pollution-free distributed power generation system. Large-scale investments and funding of PV systems have been recorded in detail more recently,2 but PV systems also have disadvantages such as low efficiency and instability, so the accuracy and efficiency of PV system model design is even more crucial for researchers. Designing a stable and accurate solar PV model is even more of a challenging project. Modeling of PV systems often requires accurate mathematical expressions and parameter extraction, however, in many solar industries the parameters given in the manufacturer’s data sheets are not accurate enough, and we need to more accurately evaluate the performance of PV arrays, power–voltage (P–V) and current–voltage (I–V),3,4 as a way to optimize the efficiency of light-generated electricity. The single diode model (SDM) and the double diode model (DDM) are frequently used by researchers today.5 Both models contain electronic components capable of analyzing the nonlinear effects of real PV model circuits.

Classical methods for identifying PV models include some gradient-based deterministic methods, such as exact, Newtonian, and curve fitting,6,7 but these methods often have some limitations, such as being too sensitive to the generated initial values requiring user-defined parameters,8,9 easily falling into local optima, and so on. So, the use of metaheuristic techniques in the field of solar energy helps us to solve some problems effectively. It does not have any requirements for constructing the objective function and has a wide range of search behaviors. Thus, it can solve the parameter evaluation problem of complex solar PV systems more accurately. So the application of some metaheuristic techniques can better solve such problems.10,11 Awadallah et al.12 used the bacterial foraging algorithm to study the relevant variations of PV system parameters extracted from manufacturer’s data sheets. Zhang et al.13 studied the distribution optimization problem of generalized normal and its application direction in solar model parameter evaluation. Yang et al.14 applied grouped grey wolf optimizer to the problem related to maximum power point tracking of doubly fed wind turbine. Bastidas-Rodriguez et al.15 applied genetic algorithm to identify the parameters of SDM of PV panel. Askarzadeh et al.16 applied artificial swarm optimization algorithm to the problem of solar PV cell parameter identification and obtained highly accurate results. Nayak et al.17 applied grey wolf optimization algorithm (GWO) to single diode PV module parameter estimation and obtained accurate results. Abbassi et al.18 applied salp swarm-inspired algorithm to solar cell model parameter identification problem. Askarzadeh et al.19 designed a novel bird mating optimization method for the extraction of PV system parameters under different operating conditions. Compared with the traditional analytical methods, the metaheuristic algorithm does not need to be designed into an exact mathematical model,20,21 but only needs to define the search range of parameters and a designed objective function to achieve the solution for complex problems, which largely reduces the computational burden.

Compared with the traditional gradient-based methods, metaheuristic algorithms do show better results in solving tedious derivation arithmetic problems. These metaheuristics has shown great effectiveness in solving the practical problems such as multiattribute decision making,22–25 object tracking,26,27 design of power electronic circuit,28,29 neural network training,30 gate resource allocation,31,32 fault diagnosis of rolling bearings,33,34 traveling salesman problem,35 fractional-order controller,36 feature selection,37,38 and medical diagnosis.39,40 However, when faced with these multimodal nonlinear objective functions of PV models containing multiple local optima, convergence is often slow and enters into deceptive convergence. According to the No Free Lunch (NFL) theorem,41 there does not exist an optimizer that can solve all the optimization parameters extraction in an all-purpose way. Therefore, some variants of metaheuristic algorithms have emerged with relatively good results. Jordehi et al.42 proposed a time-varying acceleration coefficient particle swarm optimization algorithm for efficient evaluation of pending parameters of PV system components. Hasanien et al.43 improved the whale optimization algorithm WOA for accurate modeling of dual diodes to further improve the utilization of PV systems for light energy. Qais et al.44 completed a
A comprehensive review of recent developments in SAPV system design based on multiobjective optimization and multicriteria decision making (MCDM) approaches, including mathematical models for estimating the output power of PV modules and batteries, is also presented by Ridha et al.\textsuperscript{60} Wang et al.\textsuperscript{61} designed an enhanced antlion optimizer to efficiently and accurately estimate the PV model parameters. Zhang et al.\textsuperscript{62} proposed an orthogonal moth flame optimization (MFO) with local search for identifying the parameters of the PV cell model, which was named NMSOLMFO. Abbassi et al.\textsuperscript{63} proposed an opposition-based learning modified sarp swarm algorithm (OLMSSA) for the accurate identification of the two diode model parameters of the electrical equivalent circuit of the PV cell. Chen et al.\textsuperscript{64} proposed an improved sine cosine algorithm to efficiently approach the unknown parameters of solar cells and PV modules.

However, most of the aforementioned hybrid metaheuristic algorithms can only obtain satisfactory root mean square error values and do not use relatively few computational resources. Some algorithms still suffer from degradation of the fitness function, excessive interactivity and slow transformation speed. Moreover, for algorithms based on a single search, they are more dependent on the starting point where the algorithm starts performing the search. In multimodal optimization problems, experimental data sets that contain noise caused by the process of collecting imperfect data may lead to deterioration of the performance of the multipeak multimodal characteristics of the solar PV system objective function using metaheuristic algorithms, further leading to problems such as low solution accuracy and falling into deceptive optimality. Therefore, the existing metaheuristic method still needs continuous improvement to further enhance its search capability. Developing a more competitive technique to solve this multimodal function optimization problem more comprehensively in any environment where the parameters of the PV model need to be accurately evaluated to achieve the actual energy efficient utilization of the PV system. This is still a complex and challenging optimization problem.

 Colony predation algorithm (CPA)\textsuperscript{65} is a state-of-the-art stochastic continuous optimization algorithm, which is constructed with the idea of collective predation behavior of animals in nature, using the relevant mathematical mapping to perform operations such as scattering prey, encircling prey, and supporting the closest companion of hunters while using the difference in success rates to simulate the animal. The selective neglect of prey in hunting can reflect the coexistence of hunters. Due to its high search efficiency and the absence of complex concepts related to the introduction of gradients, CPA...
requires fewer parameters to be tuned, and is being followed in various fields such as local sequence comparison, maximum power tracking for PV models, multiobjective problems, and training of neural networks. However, due to the simplicity of the structure, CPA still suffers from some shortcomings. For example, when dealing with highly complex nonlinear multiple deceptive optimal optimization problems, it leads to less than perfect search space and poorer diversity of populations, then the convergence trend and speed at later stages will be greatly reduced. The performance of traditional CPA algorithms in the search space still has much room for improvement. So far, no CPA-related techniques have emerged to deal with the complex multimodal nonlinear optimization problems of PV models and PV parameter extraction problems. CPA also suffers from incompleteness and immaturity of convergence for the population search space. Therefore, in this study, an enhanced colony predation algorithm (ECPA) based on the extremal optimization (EO) scheme and Nelder–Mead simplex strategies (NMs) is proposed for the first time. The scheme makes full use of the differential vectors and the target information generated by the search population, enhances the intensification of the algorithm search, extends the dimensionality of the optimization problem, and possesses a stronger search and exploitation capability to extract the parameters of the solar PV system. To validate the performance of the algorithm, this study compares with some advanced state-of-the-art algorithms in solar PV such as generalized oppositional teaching learning based optimization (GOTLBO), \(^6^6\) GOFPANM, \(^6^7\) improved JAYA (IJAYA) optimization algorithm, \(^6^8\) multiple learning back-tracking search algorithm (MLBSA), \(^6^9\) random learning gradient based optimization (RLGO), \(^7^0\) enhanced Harris hawks optimization (EHHO), \(^7^1\) slime mould algorithm (SMA), \(^7^2\) gradient-based optimizer (GBO), \(^7^3\) weighted mean of vectors (INFO) \(^7^4\) on different PV models (including single diode, double diode, triple diode, and PV modules) for efficient extraction of PV parameters, ECPA compared with other ECPA shows the fastest extraction capability and the most accurate results compared to other strong competing algorithms. In addition, statistical and competitive experiments confirm the highly stable performance of ECPA, highlighting higher efficacy results, by performing stability analysis of EPCA for varying temperature and irradiation under two real datasets obtained from the manufacturer's data sheet.

In short, the main contributions of this study are as follows:

1. Based on the idea of introducing EO scheme and NMs local search strategy, an enhanced evolutionary algorithm ECPA is proposed in this study. The proposed ECPA algorithm can identify the unknown parameters of multiple PV models more accurately, rapidly and stably.

2. The EO strategy incorporated into CPA is firstly able to generate variants that successively mutate the inferior genes in the original CPA, enhancing the diversity of the population while ensuring the breadth of the algorithm search. At the same time, the introduction of NMs local search strategy in the iterative process can enhance the algorithm's trend of mining the optimal solution, accurately scanning the optimal solution neighborhood, and improving the accuracy of the final convergence to the global optimal solution. Therefore, ECPA can ensure a more mature balance between exploration capability and exploitation capability.

3. Based on experimental results on the IEEE CEC 2020 function test suite, the efficiency of the ECPA technique in solving single-mode functions, multimode functions, hybrid functions, and composite functions is verified.

4. Based on several PV models, we compare ECPA with some state-of-the-art algorithms through detailed and comprehensive experiments and analyses as a way to verify the highest efficiency of ECPA and its strong competitiveness in dealing with complex parameter extraction problems of PV systems.

5. Based on three solar PV models, ECPA was tested at different temperatures and irradiances to verify its efficiency and high stability.

The structure of this paper is as follows. Section 2 introduces the definition related to the PV problem. Section 3 presents the algorithm proposed in this paper in detail. Section 4 validates the performance of ECPA by conducting competitive comparison experiments of the proposed algorithm on the IEEE CEC 2020 function test suite and four PV models. Simulation experiments are also introduced for stability experimental analysis. Finally, Section 5 provides a summary of this study and an outlook and plan for the future.

## 2 | PROBLEM DEFINITION

With the highest potential to solve the system of converting light energy into electrical energy, the stable establishment of PV models and the evaluation of unknown parameters can directly affect the efficiency of converting solar energy into electrical energy. Therefore, researchers have set up some accurate PV models that can stably demonstrate the basic current-voltage characteristics of solar modules. The models commonly
used by researchers are single diode model (SDM), double diode model (DDM), triple diode model (TDM) and PV module. The above-mentioned equivalent models and the corresponding objective functions are described in detail below.

2.1 | Solar cell model

2.1.1 | Single diode model

SDM are often used in the characterization of solar PV cells and in the field of PVs because of their simpler structural model and high simulation accuracy. Among others, SDM can be exploited in low cost and fast response PV modules. Figure 1 shows the basic structure of the SDM, which consists of three parts. The first part is a current source, which is connected in parallel with the diode. The second part is a current divider capable of indicating the leakage current. The third part is a series-connected resistive cell, whose action corresponds to the energy consumption associated with the load current. In this case, the output current of the SDM is expressed as the equation below.

\[ I_L = I_{ph} - I_d - I_{sh}, \]

where \( I_L \) is the output current, \( I_{ph} \) stands for photogenerated current, \( I_d \) denotes diode current, which can be computed by Equation (2) according to Shockley equation, and \( I_{sh} \) is shunt resistor current calculated by Equation (3).

\[ I_d = I_{sd} \exp \left( \frac{q(V_L + R_s I_d)}{n k T} - 1 \right), \]

\[ I_{sh} = \frac{V_L + R_s I_d}{R_{sh}}, \]

where \( R_s \) and \( R_{sh} \) are on behalf of the series and shunt resistances, respectively. \( V_L \) denotes the output voltage and \( I_{sd} \) is the reverse saturated current of SDM. And \( n \) denotes the diode ideality factor, \( k \) means Boltzmann constant \((1.3806503 \times 10^{-23} \text{ J/K})\), \( q \) stands for the charge of the electron \((1.60217646 \times 10^{-19} \text{ C})\), and \( T \) stands for the Kelvin temperature of the battery. Based on this, Equation (1) can be reset to the following equation:

\[ I_L = I_{ph} - I_{sd} \exp \left( \frac{q(V_L + R_s I_d)}{n k T} - 1 \right) - \frac{V_L + R_s I_d}{R_{sh}}. \]

From Equation (4) we can see that the single diode has five unknown parameters \((I_{ph}, I_{sd}, R_s, R_{sh}, n)\). Optimization techniques to more consistently and accurately evaluate these parameters have led to a significant increase in the energy conversion rate of solar PV cells.

2.1.2 | Double diode model

The structural simplicity of SDM makes it widely used in PV systems, but it has the limitation that it does not calculate the effects related to the compound current losses. Therefore, the DDM was introduced to improve the accuracy of the model. In the DDM, an additional diode is added than the SDM. The role of the first diode is responsible for modeling the compound currents and uncertainties, and the second is used as a rectifier. A shunt resistor is also present to divert the current generated by the solar energy. Figure 2 shows the structural circuit diagram of the DDM model. The simple expression for the output current is represented by the following equation:

\[ I_L = I_{ph} - I_{d1} - I_{d2} - I_{sh}, \]

where \( I_{d1} \) and \( I_{d2} \) denote the currents of the first and second diodes. A more detailed expression for the output current is shown below.
\[ I_L = I_{ph} - I_{sd1} \left[ \exp \left( \frac{q(V_L + R_s I_L)}{n_1 kT} \right) - 1 \right] - I_{sd2} \]

\[ \left[ \exp \left( \frac{q(V_L + R_s I_L)}{n_2 kT} \right) - 1 \right] - \frac{V_L + R_s I_L}{R_{sh}}, \]

where \( I_{sd1} \) and \( I_{sd2} \) mean the diffusion and saturation currents, \( n_1 \) and \( n_2 \) represent the ideal factors for diffusion and conforming diodes, respectively. The seven different parameters \( (I_{ph}, I_{sd1}, I_{sd2}, n_1, n_2, R_s, R_{sh}) \) that need to be extracted under the DDM model can be seen in Equation (6).

### 2.1.3 Three diode model

The TDM is used to solve industrial optimization problems for large-scale solar industry, to better determine the current components of solar PV cells, and to fit the current curves well but the hardware implementation is cumbersome. Figure 3 shows the equivalent circuit of the TDM model. The current generated by the solar light source through the triple diode and two equivalent resistors is recorded by \( I_{d1}, I_{d2}, I_{d3}, I_{sh}, \) and \( I_s \). Based on Kirchhoff's current law, the simple output current expression of the TDM is shown in equation as follows:

\[ I_L = I_{ph} - I_{d1} - I_{d2} - I_{d3} - I_{sh}. \tag{7} \]

The following equation shows a more detailed expression for the output current under the DDM model.

\[ I_L = I_{ph} - I_{sd1} \exp \left[ \frac{q(V_L + R_s I_L)}{n_1 kT} \right] - 1 \right] - I_{sd2} \]

\[ \exp \left[ \frac{q(V_L + R_s I_L)}{n_2 kT} \right] - 1 \right] - I_{sd3} \]

\[ \exp \left[ \frac{q(V_L + R_s I_L)}{n_3 kT} \right] - 1 \right] - \frac{V_L + R_s I_L}{R_{sh}}, \tag{8} \]

where \( I_{sd3} \) denotes the current of third diode and \( n_3 \) stands for the ideal factor. Under the TDM model, the same parameters represent the same meaning as under the SDM and DDM models. Therefore, TDM contains nine control parameters, namely \( I_{ph}, I_{sd1}, I_{sd2}, I_{sd3}, n_1, n_2, n_3, R_s, \) and \( R_{sh} \).

### 2.2 PV module model

PV module model is an important device for converting light energy into electrical energy, which is mainly obtained from some solar cells connected in series or in parallel. The equivalent circuit of the PV module is shown in Figure 4. The output current expression of the PV module can be systematically expressed as the following equations:

\[ I_L = I_{ph} N_p - I_{sd} N_p \left[ \exp \left( \frac{q\left( \frac{V_L}{N_s} + N_s \frac{R_s}{N_s} \right)}{n_1 kN_s T} \right) - 1 \right] \]

\[ - \frac{V_L}{N_s} + N_s \frac{R_s}{N_s} \]

\[ I_L = I_{ph} N_p - I_{sd} N_p \left[ \exp \left( \frac{q\left( \frac{V_L}{N_s} + N_s \frac{R_s}{N_s} \right)}{n_1 kN_s T} \right) - 1 \right] \]

\[ - I_{sd2} \left[ \exp \left( \frac{q\left( \frac{V_L}{N_s} + N_s \frac{R_s}{N_s} \right)}{n_2 kN_s T} \right) - 1 \right] \]

\[ - \frac{V_L}{N_s} + N_s \frac{R_s}{N_s} \]

In Equations (9) and (10), \( N_p \) denotes the number of parallel solar cells while \( N_s \) is the number of series ones. The PV model also has five parameters that need to be evaluated precisely: \( I_{ph}, I_{sd}, R_s, R_{sh}, \) and \( n. \)
2.3 | Objective function

In solar PV systems, to solve the most accurate model and the minimum value of the simulated objective function in each PV model, the primary goal is to simulate an objective function that accurately reflects the real performance of the PV system. Also to make the error between the experimentally derived current data and the measured current data as small as possible. In this paper, for the four PV models under SDM, DDM, TDM, and PV components, the objective functions are shown in Equations (11), (12), (13), and (14). Their overall differences are indicated by the root mean square error (RMSE) values exhibited by Equation (15). A smaller value of the RMSE indicator means that the data simulated by the objective set of unknown parameters.

\[
f(V_{L}, I_{L}, X) = I_{ph} - I_{ad} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{nkT} \right] - \frac{V_{L} + R_{S}I_{L}}{R_{sh}} - I_{L},
\]

\[
f(V_{L}, I_{L}, X_{D}) = I_{ph} - I_{ad1} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{1}kT} \right] - I_{ad2} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{2}kT} \right] - I_{ad3} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{3}kT} \right] - \frac{V_{L} + R_{S}I_{L}}{R_{sh}} - I_{L},
\]

\[
f(V_{L}, I_{L}, X_{T}) = I_{ph} - I_{ad1} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{1}kT} \right] - I_{ad2} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{2}kT} \right] - I_{ad3} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{n_{3}kT} \right] - \frac{V_{L} + R_{S}I_{L}}{R_{sh}} - I_{L},
\]

\[
f(V_{L}, I_{L}, X_{PV}) = I_{ph}N_{p} - I_{ad}N_{p} \exp \left[ \frac{q(V_{L} + R_{S}I_{L})}{nkT} \right] - \frac{V_{L} + R_{S}I_{L}}{R_{sh}} - I_{L},
\]

\[
RMSE(X) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} f(V_{L}, I_{L}, X)^2},
\]

In Equation (15), \( N \) stands for the number of current data in the experiment and \( X \) means the set of unknown parameters. \( X_{D} \) denotes the parameter set of \( [I_{ph}, I_{ad}, R_{S}, R_{sh}, n] \). \( X_{D} \) is on behalf of the parameter set of \( [I_{ph}, I_{ad1}, I_{ad2}, R_{S}, R_{sh}, n_{1}, n_{2}] \). \( X_{r} \) represents the parameter set of \( [I_{ph}, I_{ad1}, I_{ad2}, I_{ad3}, R_{S}, R_{sh}, n_{1}, n_{2}, n_{3}] \). \( X_{PV} \) is the parameter set of \( [I_{ph}, I_{ad}, R_{S}, R_{sh}, n] \).

3 | THE PROPOSED ECPA METHOD

3.1 | An overview of the CPA

Many optimizers have been proposed to tackle the practice problems, such as Runge Kutta optimizer (RUN), slime mould algorithm (SMA), weighted mean of vectors (INFO), Harris hawks optimization (HHO) and hunger games search (HGS). They can be the great potential tools to solve many problems in the field of engineering, medicine, medicine, education, finance, and energy. The CPA is an advanced algorithm developed based on the group predation behavior of animals. Group predation is usually carried out by a number of herd animals, and this group predation behavior usually improves the survival rate of each individual animal through communication and cooperation to capture higher quality prey. This group behavior is usually more efficient than individual animal predation. Group predation increases predation success by dividing and surrounding the target. The principle of survival of the fittest is also used in the algorithm to select individual animal leaders by this rule.

3.1.1 | Mathematical model

This section describes the mathematical model of the CPA algorithm and its establishment. This model is based on the behavior of group predation and is easy to understand and highly feasible. Figure 5 shows the search directions of groups and individuals in two-dimensional space and in three-dimensional space. The scattered predators can determine the update target by \( (X_{best}, Y_{best}) \).

Figure 6 illustrates the process by which the search agent updates its position by the position of the animal leader and other hunters in the two-dimensional search.
region. The orange individuals on the top right form a circle of encircled individuals represent a single predator encountering difficulties, through the left side of the green individual mutual assistance to find a partner, mutual communication, so as to complete the process of common hunting. The left side of the figure represents the stage where the blue leader leads the orange predator below to surround the prey, \( \tan(l^*\pi/4) \) represents the curve where the hunter surrounds the prey, \( S \) represents the search radius formed by the search force of the prey, and \( D \) represents the distance between the prey and the predator. The right side represents the potential location of the predator, which searches for the next prey potential location after no search target is found in the neighborhood, and \( D_1 \) represents the distance between them. From this process, it can be seen that the optimal search position is based on some random position inside the circle defined by the dispersed predators and the leader during the search. The gray circle represents the desired direction of the search agent to update its position.
3.1.2 | Communication and cooperation

Individual animals that hunt in groups improve their hunting success by communicating and cooperating. The following Equation (16) represents the pattern of search behavior of group-hunting animals that hunt through communication and cooperation.

\[ \vec{X}_j(t + 1) = r\vec{X}_j(t) + (1 - r)(\vec{X}_i(t) + \vec{X}_\omega(t))/2 \] (16)

where \( \vec{X}_j(t + 1) \) represents the updated position of the search agent. \( \vec{X}_j(t) \) represents that the animal is pursuing food and the range of \( r \) belongs to \([0,1]\). \( \vec{X}_i(t) \) and \( \vec{X}_\omega(t) \) represent the two positions in the \( j \)th dimension that are located closest to the prey.

3.1.3 | Disperse food

Individual animals are driven to different directions to separate prey from the population during the dispersal food phase. The predatory behavior of an individual animal during hunting is modeled by below equation

\[ \vec{X}(t + 1) = \vec{X}_{\text{best}} - S(\vec{r}(\text{ub} - \text{lb}) + \text{lb}), \] (17)

where \( \vec{X}(t + 1) \) represents the location of the population and \( \vec{X}_{\text{best}} \) represents the location of the optimal solution prey. \( \vec{r} \) represents the random number in \([0,1]\), \( \text{ub} \) and \( \text{lb} \) represent the upper and lower boundaries, respectively. \( S \) represents the power of the target individual, which can be expressed by Equation (19).

\[ S_0 = a - t \left( \frac{a}{N} \right), \] (18)
\[ S = 2S_0 r_2 - S_0, \] (19)

where \( N \) is the number of individual animals and \( t \) represents the current number of iterations. \( r_2 \) is a random number in \([0,1]\). As the number of iterations increases, \( S_0 \) decreases from \( a \) to 0. In the case where \( \omega \) equals 9, \( a \) is defined as follows.

\[ a = e^{-w \omega \left( 1 - \frac{t}{\text{Max.FES}} \right)}. \] (20)

3.1.4 | Surrounding the prey

Animal hunters will approach and surround their targets with a second strategy, a search that can be represented by the following equation.

\[ \vec{X}(t + 1) = \vec{X}_{\text{best}} - 2SD\alpha \tan \left( \frac{\pi}{4} \right) \] (21)

where \( l \) represents a random number between \([0,1]\) and \( \tan \left( \frac{\pi}{4} \right) \) is the curve surrounded by the hunter. \( D \) represents the distance between the prey and the predator expressed in Equation (22).

\[ D = |\vec{X}_{\text{best}} - \vec{X}(t)|, \] (22)

where \( \vec{X}(t) \) represents the number of hunters.

The probability of execution of the two predatory behaviors is expressed in Equation (23).

\[ \vec{X}(t + 1) = \begin{cases} \vec{X}_{\text{best}} - S(\vec{r}(\text{ub} - \text{lb}) + \text{lb}), & r_2 \geq 0.5, \\ \vec{X}_{\text{best}} - 2SD\alpha \tan \left( \frac{\pi}{4} \right), & r_2 < 0.5. \end{cases} \] (23)

3.1.5 | Assisting the closest individual

When a single hunter is in trouble when hunting, this is when an assistance strategy is introduced, represented by the below equation.

\[ \vec{X}(t + 1) = \vec{P}_{\text{nearest}}, \] (24)

where \( \vec{P}_{\text{nearest}} \) represents the position of the closest hunter in the mutual aid group. \( \vec{P}_j \) represents the closest hunter to the target, as indicated below. \( r_3 \) represents a random number in \([0,1]\).

\[ \vec{P}_j = r_3 \cdot \vec{X}_j. \] (25)

3.1.6 | Finding prey

If no search target is found in the neighborhood, individuals identify a substitute, and this pattern is shown below.

\[ D_1 = \text{abs}(2r_4 \vec{X}_{\text{rand}} \cdot \vec{X}(t)), \] (26)
\[ \vec{X}(t + 1) = \vec{X}_{\text{rand}} - SD_1, \] (27)

where \( D_1 \) is the distance of the animal population transition. \( r_4 \) is a randomly generated number in \([0,1]\).
is a randomly generated new individual, expressed by Equation (28), and \( r_5 \) is a randomly generated number in \([0,1]\).

\[
\tilde{x}_{\text{rand}} = r_5 ((ub - lb) + lb).
\] (28)

The animal group and the prey target determine the probability that an individual animal assists the nearest individual and finds the target, expressed by the below equation

\[
\tilde{x}(t + 1) = \begin{cases} 
\tilde{p}_{\text{nearest}} & \text{abs}(r_6) < 1, \\
\tilde{x}_{\text{rand}} - SD_1 & \text{abs}(r_6) > 1,
\end{cases}
\] (29)

where \( r_6 \) is a randomly generated number in \([-2,2]\).

When the search agent exceeds the upper or lower bound, Equation (30) is introduced to replace the position beyond the bound using the current optimal position.

\[
\tilde{x}_{\text{ob}}(j) = \tilde{x}_{\text{best}}(j),
\] (30)

where \( \tilde{x}_{\text{ob}}(j) \) denotes the position beyond the boundary and \( \tilde{x}_{\text{best}}(j) \) denotes the current optimal position.

### 3.2 An overview of the EO

It is well known that metaheuristics can effectively solve many real-world problems such as image segmentation,94–96 plant disease recognition,97 big data optimization problems,98 scheduling problems,99,100 green supplier selection,101 economic emission dispatch problem,102 engineering design,103,104 and combination optimization problems.105 EO106 is also one of these kind of metaheuristics. EO106 emerged to validate the concept of self-organized criticality (SOC), a concept used to represent burstiness. EO reserves only a single chromosome for defining local fitness functions of multiple genes during optimization, which are characterized by the ability to linearly combine into more high-quality solutions for solving the target. And the optimization of high-quality solutions can be achieved by co-evolution of genes to reach the goal.

Inspired by the Bak–Sneppen model, EO is able to pick a single solution from multiple solutions of the problem population to adjust the mutation and finally find the optimal target solution. EO is characterized by focusing on the adjustment of elements in a single solution strategy, being able to generate variants in the optimization process, and getting new fitness values for ranking in each evaluation process, thus randomly replacing the solutions about the variables. This motivates a large-scale dynamic movement of the population. Thus, it brings us a new way of thinking that allows the worse solutions or suboptimal solutions to be adjusted by mutations, for successive mutations of inferior genes, continuously improving their constituents and thus not being limited to optimal solutions in a narrow space. The pseudocode of EO can be represented by Algorithm 1.

**Algorithm 1. The pseudocode of EO**

1. Generate a random solution \( X = (x_1, x_2, ..., x_{i-1}, x_i, ..., x_D) \), \( i \in (1,2, ..., D) \). Put the current solution as the best solution, that is, \( X_{\text{best}} = X \). At this point the value of the objective function is assigned, that is \( F(X_{\text{best}}) = F(X) \).
2. Facing solution \( X \):
   1. While ensuring that other variables remain unchanged, the variables \( x_i \) that make up \( X \) are introduced one by one. This generates \( D \) new solutions \( N_X = \{X_{x_1}, X_{x_2}, ..., X_{x_i}, ..., X_{xd} \} \).
   2. For each variable \( x_i \), the fitness evaluation \( I_f = OBJ(X_{x_i}) - OBJ(X_{\text{best}}) \) was performed.
   3. After comparing the fitness values and finding the optimal fitness value, assuming it is the \( m \)th variable, set \( l_m < I_f \) for all \( j \).
   4. The fitness value of \( f(X_{\text{new}}) \) is updated after adding the mutation of \( X_i \), where \( X_{\text{new}} = (x_1, ..., x_{x_i}, ..., x_D) \), where \( x_{x_i} \) is the element of mutation.
   5. If \( OBJ(X_{\text{new}}) < OBJ(X) \) after the operation by \( OBJ \), then update \( X \) and \( F(X) \) as follows, that is, \( X = X_{\text{new}}, F(X) = F(X_{\text{new}}) \). Otherwise, keep \( X \) the same as the previous time.
   6. If a better solution is obtained, then the global optimal solution is set to \( X_{\text{new}} \), that is, \( X_{\text{best}} = X_{\text{new}} \).
3. If the termination condition is not met, then it returns to the second part.
4. Output the best solution \( X_{\text{best}} \) and \( F(X_{\text{best}}) \).

### 3.3 NMIs method

The use of the NMIs method technique107 is designed to solve minimization problems without gradient constraints. The NMIs iterative technique is used to search for the optimal solution by generating a simplex polyhedron with \( D + 1 \) vertices in the face of a \( D \)-dimensional minimization function. While iterating, the fitness value of each vertex of the simplex polyhedron is calculated, and some poorly performing vertices emerge. The poorly behaved vertices are then reflected, expanded, continuously compressed and contracted as follows to change the vertex positions, thus generating a new simplex polyhedron. This allows the vertex positions to
be continuously updated during the iterative process, so that the simplex is always transformed towards the optimal solution target.

The details of the implementation of NMs technology can be achieved in six steps as follows.

Step 1: The reflection coefficient $\alpha$, expansion coefficient $\beta$, compression coefficient $\gamma$ and shrinkage coefficient $\delta$ are set in advance, and then the fitness values computed for each vertex of the simplex are recorded and sorted using the equation

$$f(x_1) \leq f(x_2) \leq f(x_3) \leq \cdots \leq f(x_D) \leq f(x_{D+1}).$$

(31)

Step 2: Use Equation (32) to calculate the position of the reflection point and its fitness value $f(x_r)$ calculated by the function. After replacing $x_{D+1}$ with $x_r$ to make a judgment: if the fitness function value of $x_r$ is between $x_D$ and $x_1$, that is, $f(x_1) \leq f(x_r) \leq f(x_D)$, then perform the sixth step. If the fitness function value of $x_r$ is smaller than the fitness function value of $x_1$, step three is executed at this point. If the value of the fitness function of $x_r$ is greater than the value of the fitness function of $x_D$, then the fourth step is executed.

$$x_r = (1 + \alpha)\bar{x} - \alpha x_{D+1}.$$  

(32)

where $\alpha$ denotes the reflection factor, $\bar{x}$ is the center position of the vertex, and $\bar{x}$ can be calculated using the equation below

$$\bar{x} = \frac{1}{n} \sum_{i=1}^{n} x_i / D.$$  

(33)

Step 3: Use Equation (34) to calculate the position of the unfolding point $x_e$ and the value of the fitness function $f(x_e)$ calculated by the function. At this point, if $f(x_e) \leq f(x_r)$, then the position of the unfolding point $x_e$ is used to replace $x_{D+1}$, otherwise $x_r$ is used to replace $x_{D+1}$. Then the circular judgment condition in step 6 is executed.

$$x_e = (1 - \beta)\bar{x} + \beta x_r.$$  

(34)

where $\beta$ is the expansion coefficient.

Step 4: Compare the values of the fitness functions of $x_e$ and $x_{D+1}$. If $f(x_e) \leq f(x_{D+1})$, then use Equation (35) to calculate the external compression point $x_{oc}$, otherwise use Equation (36) to calculate the internal compression point $x_{ic}$.

$$x_{oc} = (1 - \gamma)\bar{x} + \gamma x_e,$$  

(35)

$$x_{ic} = (1 - \gamma)\bar{x} + \gamma x_{D+1}.$$  

(36)

where $\gamma$ represents the compression coefficient.

When the external compression point $x_{oc}$ is computed the judgment of the fitness value is performed. If $f(x_{oc}) < f(x_i)$, then replace $x_{D+1}$ with the external compression point $x_{oc}$ before executing the sixth step, otherwise skip to the fifth step. Correspondingly, when the internal compression point $x_{ic}$ is computed, if $f(x_{ic}) < f(x_{D+1})$, then replace $x_{D+1}$ with the internal compression point $x_{ic}$ and skip to step 6, otherwise go to step 5.

Step 5: Use Equation (37) to shrink all the points with $x_i$ removed, with the aim of being able to generate a new simplex polyhedron, and then perform step 6.

$$V_i = \delta x_i + (1 - \delta)x_1, \quad i = 2, ..., D + 1,$$  

(37)

where $\delta$ is the shrinkage coefficient.

Step 6: Determine whether the termination condition of the iteration is satisfied, if so, stop the step of updating the simplex polyhedra, otherwise continue the iterative process.

### 3.4 The proposed ECPA

The basic CPA algorithm suffers from the disadvantages of low solution efficiency and immature convergence, poor original population diversity, and immature exploration and exploitation balance when solving optimization complex optimization problems, and an enhanced algorithm with respect to CPA has not yet appeared for stable and highly accurate identification of unknown parameters of PV systems. Therefore, this study proposes an evolutionary version of CPA, ECPA, which incorporates the idea of EO in CPA, improves the location of candidate solutions for inferior agents, and enhances the variational scalability of the algorithm. After improving the diversity of the population, the fusion NMs strategy further enhances the algorithm’s accurate development capability, allowing the dimensionality of the optimization problem to be extended and converge to the optimal solution faster. PV experiments further verified that ECPA’s precise development capabilities have further improved the accuracy of extracting PV parameters. The performance of the ECPA algorithm is substantially improved by the improved ECPA algorithm, which has better metrics in both exploration and exploitation stages.

In this study first starts initializing the agent population by cyclically acquiring individuals and calculating the fitness value of each individual, then updating $S_0$ with

$$S_0 = a - t \left( \frac{d}{N} \right)$$

to prepare for the next update of the prey intensity. Immediately afterward, we enter the communication and cooperation phase, choosing a population of one dimension and using the two best individuals to change the
position of the population. The loop then updates the prey intensity using $S = 2S_0r_2 - S_0$, which decreases with the number of iterations. The loop chooses to disperse the prey or surround it depending on the size relationship between $r_2$ and 0.5, choosing to execute the following fundamental strategy of population evolution.

$$
\vec{X}(t + 1) = \begin{cases} 
\vec{X}_{best} - S(\vec{r}_1(ub - lb) + lb), \\
\vec{X}_{best} - 2SDe^l\tan\left(\frac{l\pi}{4}\right),
\end{cases}
$$

(38)

Then, after entering the phase of searching for prey based on the relationship between animal hunters and prey and their probability of finding the target, mutations are adjusted using EO to generate variants, and a selected scenario can generate $D$ mutation scenarios. In each evaluation process replace the solution about the variant. In this case, the mutation step is adjusted by the following form.

$$
x_m = x_{Nm} + N_m(0, 1),
$$

(39)

$$
x_m = x_{Nm} + C_m(0, 1),
$$

(40)

where $x_m$ and $x_{Nm}$ are the $m$th decision variable before the mutation occurs and the decision variable after the mutation, respectively, and $N_m(0,1)$ and $C_m(0,1)$ are the random numbers satisfying Gaussian and Cauchy distributions, respectively.

Figure 7 represents the process of new solutions generated by ECPA after the introduction of EO, and it is clear that the mutation operation has a crucial role in the change of the composition of the inferior genes and the solution out of the narrow space.

While EO motivates the population to move dynamically to continuously adjust the suboptimal solutions, the strategy of NMs is introduced to search for better solutions in the continuous space. In this case, after constructing a simplex with the current optimal solution, the simplex is updated, and the position of the optimal solution in the population is selected as the starting position for the search, and the improved simplex trajectory is searched in the tolerance domain of the current solution. If a better improved solution can be found, then it is used as a trial run trajectory for the next simplex search cycle. Otherwise, the iterative process is repeated while the search steps are gradually reduced until the termination condition is satisfied. In this way, the undesirable cost of repeated redundant calculations is avoided, and the potential optimal solution neighborhood of the algorithm is further developed to improve the accuracy of convergence to the optimal solution.

We found that there is an important parameter $\mu$ in the ECPA algorithm, and the number of iterations of ECPA and the rate of convergence of the algorithm to the global optimal solution depend to some extent on the value of $\mu$. After testing, if $\mu$ will be set too large, our compression and expansion operations in NMs local search strategy dominate the weight and the variational strategy of EO is drastically scaled down. It also greatly wastes the average CPU time occupied by the algorithm. If the parameter $\mu$ is set too small, the local search capability of NMs is not fully exploited, and the convergence accuracy of ECPA becomes poor. After testing, we set the value of $\mu$ to be more suitable as $d + 1$, and further iterate the NMs local search method $\mu$ times to search the global optimal solution.

Therefore, the pseudocode of the ECPA method proposed in this study is shown in Algorithm 2 and the flowchart is illustrated in Figure 8.

Algorithm 2. The pseudocode of ECPA

1. Initialize the parameters of the ECPA algorithm, such as $N$, dim, Max_FEs, values;
2. Initialize the location of all individuals $X_i (i = 1, 2, ..., N)$;
3. While ($fes \leq \text{Max}_\text{Fes}$)
   a. For $i = 1: N$ do
      i. Check the solution if the search space is exceeded and replace the exceeded position with Equation (30);
      ii. Calculate the fitness value for each target individual;
      iii. Update the best location $\vec{X}_{best}$;
   b. End For
4. Use Equation (18) and Equation (19) to update the target individual’s strength $S$;
5. Update $a$ by Equation (5);
6. Update $r$;
7. For $j = 1: \text{dim}$ do
   a. Update $X_i$, $X_k$, $i$, $r$;
   b. If rand() < $fes/\text{Max}_\text{Fes}$
      i. $\vec{X}_i = \vec{X}_{best}$;
   c. End If
   d. Calculate the $X'_j$ by Equation (16);
   e. End For
8. Perform the operation of Extremal optimization to update the best location $\vec{X}_{best}$;
9. For $i = 1: N$ do
   a. Update the target individual’s strength $S$ again;
   b. If abs($S$) < $2^a/3$
      i. Calculate $X_i$ using Equation (23);
   c. Else
      i. Calculate $X_i$ using Equation (29);
   d. End If
   e. End For
10. Perform NMs operation with the current best position as the initial value;
11. Output $\vec{X}_{best}$
3.5 Computational complexity analysis

The computational complexity of our proposed ECPA algorithm is related to the maximum number of evaluations ($Max\_FEs$), the population size ($N$), and the dimensionality of the problem ($D$). So the overall computational complexity is

$$\text{O}\_\text{ECPA} = \text{Max\_FEs}\*(O(\text{Calculate the fitness value for all individuals}) + O(\text{Updating the location of all individuals}) + O(\text{Communication and collaboration})).$$

In this case, our location update process includes the location update of all agents in the original CPA, which updates all agents by EO and NMs local search strategy (which also includes multiple evaluations of fitness values). The complexity of computing the fitness values of all agents is $O(n)$. The computational complexity of updating the agent's position in the original CPA is $O(N \times \left(1 + \frac{D \times N^2}{4}\right))$. The computational complexity of updating the agent's position by EO is $O(D + N \log N)$. The computational complexity of updating the agent's location by NMs local search strategy is $O(D)$. Therefore, the final computational complexity of ECPA is $O(\text{ECPA}) = O(\text{Max\_FEs}(n + N \times \left(1 + \frac{D \times N^2}{4}\right) + D)).$
EXPERIMENTAL INVESTIGATIONS AND ANALYSIS

Fair experimental setup is very important for computational science issues, such as location-based services and information retrieval services. In the experimental part, we validate the performance and results analysis of the ECPA in a more complete way. First, the efficiency and accuracy of the algorithm is verified by a well-known series of benchmark function tests (IEEE CEC 2020 function set), where it is evaluated against a variety of state-of-the-art algorithms to test the potential of ECPA to solve complex problems. Then, we apply ECPA to several classical parameter extraction problems of solar PV cell systems and compare it with some well-known algorithms to verify the effectiveness and accuracy of the ECPA method on solar PV cells. All the experimental results verify that the ECPA method can accurately and efficiently complete problems such as nonlinear multimodality under PV models, and can extend the dimensionality of optimization problems and find the optimal solution more precisely.

The Wilcoxon signed-rank test with a significance level of 0.05 was used in our experiments to evaluate the performance tests of ECPA and related competing algorithms. The symbols “+,” “−,” and “=” were used to determine the performance differences between ECPA and the competitors. For example, “+” is used when the performance of ECPA is better than the competitor, “−” is used when the performance result is not enough to catch up with the competitor, and “=” is used when the performance result is not significantly different from the competing algorithms. The performance comparison between ECPA and some advanced algorithms can be seen more intuitively and simply by using the Wilcoxon signed-rank test. In this study, the experiments were done on a desktop computer with AMD 3700X CPU, 3.60 GHz, Win10 system and 16 GB of RAM using MATLAB 2015b.

Detailed results on the IEEE CEC 2020 benchmark functions

In this section, to verify the effectiveness of the proposed ECPA in solving complex function optimization
problems, we test and compare the ECPA with some advanced algorithms including OBSCA,114 SCADE,115 HGWO,116 CESCA, CBA,117 AMFOA,118 RCBA,119 BMWOA,120 MSFOA,121 EM on the IEEE CEC 2020 function set comparisons. These advanced competing algorithms are often used by researchers to solve complex nonlinear multimodal high-dimensional problems.122–124 The IEEE CEC 2020 function set includes functions that provide a complete and multifaceted assessment of the algorithms’ strengths and weaknesses. It consists of a total of 10 functions: including basic function F11, hybrid functions F21 on CEC 2014 and unimodal function F1, basic function F7, F19, hybrid functions F17, F16, composition functions F22, F24, F25 on CEC 2017. More detailed evaluation criteria and problem definitions for these functions can be found in the IEEE CEC 2020 related literature. A complete analytical test of the functions allows a better evaluation of the optimal solutions to different optimization problems.85,86,123–130 The comparison of ECPA with advanced competing algorithms on a CEC 2020 function set with dimensionality set to 30 is set according to the requirements of the competition. The maximum number of evaluations (Max_FEs) was set to 300,000 and the population size was set to 40. To eliminate the effect of algorithm randomness, all competing algorithms were run 30 times and the final results were averaged. Table 1 shows that all competing algorithms maintained the same parameters as the original literature when tested on CEC 2020.

The optimization statistics generated by running ECPA with all competing algorithms on the CEC 2020 function set are documented in Table 2. The table details the statistics for all competing algorithms, including the mean (Mean) and standard deviation (SD). Bold represents the performance of the algorithm in finding the optimal value. Among them, ECPA is able to find the theoretical optimum by showing the best results on F1, F2, F4, F5, F7, F9, and F10. ECPA has the most experimental results in bold for all functions, indicating that it finds the theoretical optimum on the most functions compared to other competing algorithms. This also demonstrates that ECPA is able to perform the most efficient development when solving complex optimization function problems.

The results of the evaluation of ECPA against other advanced algorithms and based on the Wilcoxon singed-rank test are documented in Table 3. The table includes both the overall comparative performance of the different algorithms on all functions and a more detailed segmented comparative performance by function classification. Based on the results of the full function comparison, it can be seen that ECPA performs much better than its competitors on all 10 functions of the CEC 2020 test suite.

A detailed comparison of the results after classifying the functions by extreme points shows that ECPA works better than any of the algorithms on the simple single-peaked function of F1, which has a better solution to the problem of evaluating a single extreme point. Second, ECPA still shows a competitive performance on the three simple multipeak functions F2–F4. Its ability to solve multipeak functions is no less than any other advanced competing algorithms. In particular, when facing the algorithms CBA, AMFOA, RCBA, CESCA, BMWOA, and MSFOA, ECPA outperforms these competitive algorithms for all three functions. When facing the better performing

| Algorithm | Parameters |
|-----------|------------|
| SCADE     | beta_min = 0.2; beta_max = 0.8; pCR = 0.8; \( \eta = a - \text{fes}(a/\text{Max}_\text{FEs}) \) |
| RCBA      | \( f_{\text{min}} = 0; f_{\text{max}} = 2; r = \text{ones}(1,N)*0.5 \) |
| BMWOA     | \( a = 2 - \text{FEs}^{(2)}(\text{Max}_\text{FEs}); a2 = -1 + \text{FEs}^{(-1)}(\text{Max}_\text{FEs}) \) |
| MSFOA     | \( W_0 = 1; \alpha = 0.95; M = 5 \) |
| HGWO      | \( a = 2 - \text{FEs}^{(2)}(\text{Max}_\text{FEs}); A = 2*\text{rand}(0,1)*a - a; C = 2*\text{rand}(0,1) \) |
| OBSCA     | \( \eta = a - \text{fes}(a/\text{Max}_\text{FEs}); a = 2; \tau = (2*\pi)\text{rand}() \) |
| CBA       | \( C_w = 3; f_{\text{max}} = 2.5; \eta = \tau = 0.5 + \text{rand}(0,1); \omega = \left( \frac{\text{fES}}{\text{Max}_\text{FEs}} \right) + 0.1; u = \left( \frac{\text{fES}}{\text{Max}_\text{FEs}} \right)^{0.4} + 0.3 \) |
| CESCA     | \( \tau = a - \text{fes}(a/\text{Max}_\text{FEs}); a = 2; \tau = (2*\pi)\text{rand}() \) |
| AMFOA     | \( W_0 = 1; \alpha = 0.95; M = 5 \) |
| EM        | IndexLB = 0; LSITER = 5; delta = 0.1 |
| ECPA      | \( a = \exp(9 - 18*\text{fes}/\text{Max}_\text{FEs}); S0 = a*(1 - \text{fes}/\text{Max}_\text{FEs}); S = 2*S0*\text{rand} - S0 \) |

**Table 1** Parameter records of the competition algorithms.
| Function | ECPA | CBA | RCBA |
|----------|------|-----|------|
|          | Mean | SD  | Mean | SD  | Mean | SD  |
| F1       | 1.7139E+02 | 3.9072E+02 | 4.9601E+05 | 1.9910E+06 | 1.9380E+04 | 7.9513E+03 |
| F2       | 4.9591E+03 | 5.2043E+02 | 5.7803E+03 | 6.7214E+02 | 5.7963E+03 | 5.7414E+02 |
| F3       | 9.5056E+02 | 1.1396E+02 | 1.8157E+03 | 2.2063E+02 | 1.7691E+03 | 2.6793E+02 |
| F4       | 1.9000E+03 | 0.0000E+00 | 1.9624E+05 | 1.6384E+01 | 1.9272E+03 | 5.5168E+00 |
| F5       | 3.1094E+03 | 3.6619E+02 | 3.6191E+05 | 2.4185E+05 | 1.2531E+05 | 8.1991E+04 |
| F6       | 3.0384E+03 | 3.1368E+02 | 3.3537E+03 | 4.6225E+02 | 3.0495E+03 | 4.4291E+02 |
| F7       | 9.1263E+03 | 6.8606E+03 | 2.0431E+05 | 1.1318E+05 | 9.7095E+04 | 6.0468E+04 |
| F8       | 2.4240E+03 | 2.0141E+01 | 2.4775E+03 | 6.4535E+01 | 2.4855E+03 | 4.7540E+01 |
| F9       | 2.6000E+03 | 0.0000E+00 | 2.8088E+03 | 4.7550E+01 | 2.9047E+03 | 5.6576E+02 |
| F10      | 2.7000E+03 | 0.0000E+00 | 3.0360E+03 | 4.4581E+01 | 3.0326E+03 | 3.5982E+01 |

| Function | AMFOA | EM  | OBSCA |
|----------|-------|-----|-------|
|          | Mean  | SD  | Mean  | SD  |
| F1       | 1.3593E+11 | 2.9019E+09 | 4.4356E+08 | 8.5585E+07 | 2.6632E+10 | 5.5110E+09 |
| F2       | 1.2491E+04 | 3.3391E+02 | 5.1422E+03 | 7.1117E+02 | 7.5612E+03 | 4.1793E+02 |
| F3       | 1.5241E+03 | 2.0515E+01 | 9.4810E+02 | 4.4886E+01 | 1.1968E+03 | 3.9797E+01 |
| F4       | 1.9066E+03 | 9.5351E+00 | 2.6802E+03 | 3.2904E+02 | 3.5596E+03 | 1.7888E+02 |
| F5       | 1.2240E+09 | 2.6475E+08 | 2.0050E+05 | 5.7817E+04 | 1.0739E+07 | 5.8932E+06 |
| F6       | 1.0606E+04 | 1.5909E+03 | 2.6802E+03 | 3.2904E+02 | 3.5596E+03 | 1.7888E+02 |
| F7       | 1.2680E+10 | 2.7970E+09 | 2.7041E+05 | 5.4279E+05 | 3.4847E+06 | 2.5004E+06 |
| F8       | 2.6891E+03 | 2.0665E+01 | 2.3797E+03 | 2.2926E+01 | 2.4881E+03 | 1.8935E+01 |
| F9       | 2.6000E+03 | 6.0605E-05 | 2.6185E+03 | 8.9282E-01 | 2.6022E+03 | 6.0832E+00 |
| F10      | 2.7000E+03 | 4.9598E+00 | 2.8088E+03 | 1.2160E+01 | 2.7751E+03 | 2.5890E+02 |

| Function | SCADE | HGWO | CESCA |
|----------|-------|------|-------|
|          | Mean  | SD  | Mean  | SD  |
| F1       | 2.9554E+10 | 5.0152E+09 | 1.0253E+10 | 1.9029E+09 | 9.9669E+10 | 6.6707E+09 |
| F2       | 8.3416E+03 | 2.4599E+02 | 7.6370E+03 | 5.1748E+02 | 9.3089E+03 | 2.1377E+02 |
| F3       | 1.2176E+03 | 3.8883E+01 | 1.0675E+03 | 1.8026E+01 | 1.5355E+03 | 4.7137E+01 |
| F4       | 1.9000E+03 | 0.0000E+00 | 1.9000E+03 | 0.0000E+00 | 1.9726E+03 | 5.9323E+01 |
| F5       | 1.8215E+07 | 9.1861E+06 | 1.1435E+07 | 5.4628E+06 | 1.2922E+08 | 4.3161E+07 |
| F6       | 3.5097E+03 | 1.3456E+02 | 3.1187E+03 | 1.6957E+02 | 5.4295E+03 | 4.3965E+02 |
| F7       | 8.0259E+06 | 4.4051E+06 | 4.1800E+06 | 1.9487E+06 | 1.1641E+08 | 5.2957E+07 |
| F8       | 2.5143E+03 | 1.6469E+01 | 2.4313E+03 | 1.0053E+01 | 2.6293E+03 | 2.2664E+01 |
| F9       | 2.6000E+03 | 0.0000E+00 | 3.1595E+03 | 3.7873E+02 | 2.8582E+03 | 1.1656E+02 |
| F10      | 2.7000E+03 | 0.0000E+00 | 3.4105E+03 | 9.4875E+01 | 4.3274E+03 | 5.1111E+02 |

| Function | BMWOA | MSFOA |
|----------|-------|-------|
|          | Mean  | SD    | Mean  | SD   |
| F1       | 5.0286E+08 | 2.5530E+08 | 1.3442E+11 | 2.2513E+10 |
| F2       | 7.3863E+03 | 7.4694E+02 | 9.5740E+03 | 6.5662E+02 |
EM algorithm, ECPA outperforms EM in one function and is evenly matched in two functions. Again, ECPA still outperforms all the remaining competing algorithms on the three hybrid functions, F5–F7, to the best level. Finally, when comparing ECPA with the advanced competing algorithms on the very complex composition functions, ECPA does not decrease its optimization power as the complexity of the functions increases. ECPA still shows the best performance and achieves the best evaluation results for all composition functions. In summary, ECPA is able to find exact optimal solutions not only for simple unimodal functions and basic functions, but also for very complex hybrid functions and composition functions by extending the dimensionality of the optimization problem to avoid local optimal solutions and thus produce high-quality optimization results.

Table 4 records the ranking results of ECPA after the Friedman test compared with the above competing algorithms.

| Function | BMWOA Mean | SD | MSFOA Mean | SD |
|----------|------------|----|------------|----|
| F3       | 1.2590E+03 | 8.1028E+01 | 2.8596E+03 | 2.7219E+02 |
| F4       | 1.9000E+03 | 1.5087E−08 | 5.4641E+05 | 3.2669E+05 |
| F5       | 8.3352E+06 | 8.8437E+06 | 3.3275E+08 | 1.9229E+08 |
| F6       | 3.1255E+03 | 3.9390E+02 | 6.1093E+03 | 1.2101E+03 |
| F7       | 2.5806E+06 | 2.0410E+06 | 4.8031E+08 | 7.1711E+08 |
| F8       | 2.4711E+03 | 1.9963E+01 | 2.7249E+03 | 6.0390E+01 |
| F9       | 2.6001E+03 | 1.2151E−01 | 6.0343E+03 | 5.3205E+02 |
| F10      | 2.7013E+03 | 1.1772E+00 | 1.5340E+04 | 3.2580E+03 |

**Table 3** Comparison of ECPA with other advanced competing algorithms on CEC 2020 based on Wilcoxon signed-rank test

| Comparison groups | Unimodal function (F1) + (better)/− (worse)/= (no sig.) | Basic functions (F2–F4) | Hybrid functions (F5–F7) | Composition functions (F8–F10) | All functions |
|-------------------|--------------------------------------------------------|----------------------|---------------------|-------------------------------|--------------|
| ECPA vs. CBA      | 1/0/0                                                   | 3/0/0                | 3/0/0               | 3/0/0                         | 10/0/0       |
| ECPA vs. RCBA     | 1/0/0                                                   | 3/0/0                | 2/0/1               | 3/0/0                         | 9/0/1        |
| ECPA vs. AMFOA    | 1/0/0                                                   | 3/0/0                | 3/0/0               | 3/0/0                         | 10/0/0       |
| ECPA vs. EM       | 1/0/0                                                   | 1/0/2                | 2/1/0               | 2/1/0                         | 6/2/2        |
| ECPA vs. OBSCA    | 1/0/0                                                   | 2/0/1                | 3/0/0               | 3/0/0                         | 9/0/1        |
| ECPA vs. SCADE    | 1/0/0                                                   | 2/0/1                | 3/0/0               | 1/0/2                         | 7/0/3        |
| ECPA vs. HGWO     | 1/0/0                                                   | 2/0/1                | 2/0/1               | 2/0/1                         | 7/0/3        |
| ECPA vs. CESCA    | 1/0/0                                                   | 3/0/0                | 3/0/0               | 3/0/0                         | 10/0/0       |
| ECPA vs. BMWOA    | 1/0/0                                                   | 3/0/0                | 2/0/1               | 3/0/0                         | 9/0/1        |
| ECPA vs. MSFOA    | 1/0/0                                                   | 3/0/0                | 3/0/0               | 3/0/0                         | 10/0/0       |

**Table 4** Ranking of the algorithms generated by the Friedman test results

| Algorithms | Friedman ranking |
|------------|------------------|
| CBA        | 5.33667          |
| RCBA       | 5.27333          |
| AMFOA      | 8.62000          |
| EM         | 3.82333          |
| OBSCA      | 5.36833          |
| SCADE      | 5.52667          |
| HGWO       | 5.43500          |
| CESCA      | 9.11333          |
| BMWOA      | 5.23333          |
| MSFOA      | 10.36667         |
| ECPA       | **1.90333**      |
algorithms. Based on the Friedman test ranking results, it can be seen that ECPA achieves the minimum value of 1.90333 for all algorithms and ranks first compared to other advanced algorithms. The next algorithms are EM and BMWOA, and so on. The results of the evaluation with the advanced algorithms show that ECPA has a better ability to find the best in solving core search problems and complex multimode evaluation problems. Although the candidate solution positions of the inferior agents are changed by extremal optimization strategies to perform extended variants, the NMs local search strategy is used to avoid redundant computations in the potentially continuous search space and further develop better solutions. Therefore, the convergence curves of competing algorithms on the CEC 2020 test suite shown in Figure 9 similarly demonstrate that ECPA maintains the best convergence accuracy and the fastest convergence speed compared to other competing algorithms in the process of finding the optimal solution. Based on this, ECPA can be a potential optimization technique for accurate mining of optimal solutions.

### 4.2 Evaluation based on PV models

In this section, we want to verify the performance of ECPA for complex parameter extraction of solar PV systems, using ECPA for four classical parameter extraction of SDM, DDM, TDM, and PV modules that convert light energy into electricity. In this case, each algorithm was tested for 30 independent runs on different problems, the maximum number of evaluations was set to 20,000 and the population size was taken to 30. When modeling these modules, the input variables were set to some meteorological data, such as temperature and irradiation, and the output variables were power, current, or voltage. The following two sets of data were used in this set of experiments, which correlate with the Photowatt-PWP 201 PV module and the RTC France PV cell, respectively. The first set of data was obtained from the RTC France PV cell, where the temperature was set to 33°C and the light intensity was set to 1000 W/m². Meanwhile, the second data set is from the Photowatt-PWP 201 PV module, a solar system containing 36 polycrystalline cells, where the temperature is set to 45°C and the light intensity is also set to 1000 W/m². These data sets were used extensively in the performance demonstration process to validate a new method of its kind for solar PV systems. Second, to fairly compare the performance differences between the different algorithms, the upper and lower bounds of the parameters over the different models were kept consistent with the original paper, as shown in Table 5. To further verify the performance merits of the proposed ECPA method, it is compared here with some state-of-the-art algorithms, such as GOFPANM, GOTLBO, IJAYA, MLBSA, RLGBO, EHHO, SMA, GBO, and INFO. These similar well-known metaheuristics possess a strong capability to solve nonlinear multimodal PV parameter extraction problems. The parameter list of the contrast algorithm we adopted in the PV experiments is recorded in Table 6, and they are consistent with the original literature.

The performance of the proposed ECPA algorithm can be discerned from several performance metrics as follows.

1. **RMSE**: The value of RMSE, the objective function, indicates how good the quality results of different algorithms produce optimization. The smaller the value of RMSE, the closer the simulation value and the real value are to the experimental optimization.
2. **The Wilcoxon singed-rank test metric** was used to visually express the variability of the different algorithms, where the significance level was taken as 0.05.
3. **Convergence trend curve**: The optimal solution obtained by 30 independent averaging experiments is plotted into a curve, which can clearly visualize the speed, direction and trend of convergence of different algorithms toward the optimal solution.
4. **Absolute error (IAE) and relative error (RE)**: two precise error indicators enable a clearer survey of the gap between the simulated and real data at different voltages, thus verifying whether the proposed method is able to achieve an accurate state for identifying unknown parameters of the PV system. The expressions are shown as follows:

\[
\text{IAE} = |I_{\text{measure}} - I_{\text{simulated}}|, \quad (41)
\]

\[
\text{RE} = \frac{I_{\text{measure}} - I_{\text{simulated}}}{I_{\text{measure}}}. \quad (42)
\]

#### 4.2.1 Experimental results of SDM model

In this section, we examine the capability of ECPA in handling the unknown parameters under the SDM model. Figure 10 vividly illustrates the I–V and P–V characteristic curves plotted by the ECPA technique proposed in this study when simulating the actual performance of the SDM model. It can be clearly seen that the performance curves simulated by ECPA in different voltage ranges match well with the actual data set used in this experiment, demonstrating the high degree of fit of ECPA. Figure 11 shows the error characteristics of IAE and RE between the estimated current data and the measured current data under the SDM model. From the error characteristics plot, it can be
seen that the value of IAE is always below 2.50741E−03
and the value of RE is always between $[-1.99782E−02,
1.47108E−01]$, indicating that ECPA has the ability to
extract the parameters accurately. Next, Table 7 shows
more precisely the statistics of IAE between the
simulated and measured values of current and power
generated by the ECPA under the simulated SDM model.
The sum of the current IAE is 2.15269E−02 and the sum

**FIGURE 9** Convergence curves based on F1, F2, F3, F5, F7 functions on the CEC 2020 test suite
of the power IAE is \(8.73077 \times 10^{-3}\). The maximum value of all IAEs is only \(2.50741 \times 10^{-3}\). In summary, all the low error values clearly demonstrate that the ECPA is able to accurately simulate the actual performance of the SDM model.

Table 8 precisely records the RMSE values and optimal parameters obtained by ECPA with the above published advanced techniques such as GOTLBO, GOFPANM, IJAYA, MLBSA, RLGBO, EHHO, SMA, GBO, INFO evaluated on the SDM model. The table shows that ECPA achieves the optimal RMSE value of \(9.8602 \times 10^{-4}\) together with some published advanced algorithms GOFPANM, MLBSA, GBO, INFO and RLGBO, which is smaller than the remaining competing algorithms. Based on the Wilcoxon singed-rank test it can be seen that ECPA has the potential to solve the PV parameter extraction problem in the SDM case.

In Table 9 the results of the RMSE statistics and Wilcoxon singed-rank test demonstrated by the ECPA method under the processing of the SDM model are recorded in detail, and the performance of the advanced algorithms mentioned above is compared under 30 independent experiments. In this table it is possible to see the minimum, maximum, standard deviation, and mean values of the RMSE metrics demonstrated by the different competing algorithms in the evaluation of the simulated SDM model. The results from the Wilcoxon singed-rank test metric show that ECPA successfully achieves the lowest standard deviation of the RMSE metric of all competing algorithms and outperforms them all, showing a more accurate and stable performance. Figure 12 records the convergence curves of ECPA and the above competitors under the SDM model, from which it is obvious that the ECPA algorithm converges to the optimal solution much faster than the original CPA algorithm, and faster than all the remaining competitors. Moreover, when all competing algorithms of the same kind converge to the optimal solution, it can be seen from the local zoomed-in plot that the ECPA algorithm is able to achieve the best accuracy. We can see that the three slower convergence algorithms, CPA, SMA, and EHHO, have fallen into the local optimum prematurely in the middle of the iteration, resulting in poor convergence to the final result. The EPCA algorithm has made a qualitative change to the convergence speed and accuracy of the original CPA algorithm, effectively improving the shortcomings of the slow and immature convergence of CPA. The well-known and quite effective GOFPANM lags behind the ECPA proposed in this study in terms of both convergence speed and convergence accuracy, reaching the second place, while the remaining techniques such as GOTLBO and RLGBO are far from the performance of ECPA. Based on the above discussion about ECPA, it can be seen that ECPA is much better than its similar competitors in evaluating the SDM model parameters.

### 4.2.2 Experimental results of DDM model

This section describes the experimental results of the parameter evaluation using the ECPA technique under the DDM model. First, the P–V and I–V curves are shown in Figure 13 for the simulated and experimental data using the ECPA technique in the DDM model. As with the SDM model, the simulated current data matches the measured current data, demonstrating the efficient and
accurate performance of the ECPA simulated DDM model. Next, Figure 14 shows the IAE and RE metrics between the estimated and measured values of the ECPA currents under the DDM model. Where the value of IAE is not greater than $2.54343 \times 10^{-3}$ and the value of RE is in the interval of $[-2.02594 \times 10^{-02}, 1.37430 \times 10^{-01}]$, proving that ECPA is able to simulate the performance of DDM accurately. Table 10 records the sum of IAE between the experimental and measured values of current and power under the DDM model, which are $2.12752 \times 10^{-02}$ and $8.77665 \times 10^{-03}$, respectively. The low level of error ensures the accuracy of the simulation.

For the DDM model, Table 11 records the results of ECPA’s comparison with some similar competitors on DDM. Among them, the RMSE metric of ECPA reaches $9.8248 \times 10^{-04}$, which is in the highest standard among all competing algorithms. From the Wilcoxon singed-rank test, we can see that the ECPA algorithm has outstanding performance and possesses a strong ability to simulate the actual performance of the DDM model.

The statistical data of the RMSE metrics obtained by the ECPA algorithm proposed in this study with competing algorithms under the DDM model are recorded in Table 12. Among them, the maximum value, minimum value, mean and standard deviation of the RMSE statistical metrics of ECPA achieve the most advantageous results compared to the above-mentioned competitors. Referring to the Wilcoxon rank sum test result metrics it can be seen that ECPA outperforms all competitors in terms of the effectiveness of the evaluation parameters on the DDM model. Second, the convergence curves of ECPA and the above competitors converge to the optimal solution under the DDM model can be seen in Figure 15, where it can be intuitively seen that ECPA maintains the fastest convergence speed and optimal convergence accuracy both at the beginning,
middle, and end of the iteration. Based on the above discussion on the various aspects under the DDM case, ECPA is able to maturely smooth the exploration phase and exploitation phase, outperforms all competitors, and can produce higher quality optimization solutions.

4.2.3 Experimental results of TDM model

Similarly, we modeled the ECPA on the TDM for the experiments. For the experiments on TDM, the \(I-V\) and \(P-V\) characteristics that can describe the simulated and experimental data can be represented by Figure 16. The high degree of fit shown in the figure again demonstrates that the simulated data over the specified voltage range gives a good indication of the actual performance of the TDM. Figure 17 also shows the IAE and RE values of the ECPA for the estimated and measured currents under the TDM model, with the IAE values always less than \(2.54345 \times 10^{-3}\) and the RE values fluctuating in the range \([-2.02595 \times 10^{-2}, 1.37425 \times 10^{-1}]\). The low error level also ensures the high accuracy optimization results of ECPA.

The IAE metrics for current and power under the TDM model are recorded in detail in Table 13. 2.12752E\(-2\) is the sum of IAEs belonging to current and 8.77667E\(-3\) is the sum of IAEs belonging to power. The low level
| Algorithm | $I_{ph}$ (A) | $I_{ad}$ (A) | $R_s$ (Ω) | $R_{sh}$ (Ω) | $n$ | RSME | sig |
|-----------|----------------|----------------|-----------|-------------|-----|-------|-----|
| GOTLBO    | 0.760796       | 3.18387E−07    | 0.0364    | 52.9922     | 1.4797 | 9.8690E−04 | +   |
| GOFPANM   | 0.760776       | 3.23021E−07    | 0.0364    | 53.7185     | 1.4812 | 9.8602E−04 | =   |
| IJAYA     | 0.760725       | 3.24017E−07    | 0.0364    | 54.2724     | 1.4815 | 9.8690E−04 | =   |
| MLBSA     | 0.760776       | 3.23021E−07    | 0.0364    | 53.7185     | 1.4812 | 9.8602E−04 | =   |
| RLGBO     | 0.760776       | 3.23021E−07    | 0.0364    | 53.7185     | 1.4812 | 9.8602E−04 | =   |
| EHHO      | 0.760253       | 3.71316E−07    | 0.0359    | 64.1091     | 1.4953 | 1.0569E−03 | +   |
| SMA       | 0.729659       | 4.83833E−07    | 0.0232    | 52.6564     | 1.6638 | 1.3022E−01 | +   |
| GBO       | 0.760776       | 3.23022E−07    | 0.0364    | 53.7189     | 1.4812 | 9.8602E−04 | =   |
| INFO      | 0.760776       | 3.23021E−07    | 0.0364    | 53.7185     | 1.4812 | 9.8602E−04 | =   |
| CPA       | 0.760787       | 3.75498E−07    | 0.0357    | 56.7208     | 1.4965 | 1.0295E−03 | +   |
| ECPA      | 0.760776       | 3.23015E−07    | 0.0364    | 53.7180     | 1.4812 | 9.8602E−04 | =   |

**Table 8** Comparison of different algorithms on SDM case

| Algorithm | Max     | Min     | Mean    | SD      | sig |
|-----------|---------|---------|---------|---------|-----|
| GOTLBO    | 1.48663E−03 | 9.86899E−04 | 1.11084E−03 | 1.37145E−04 | +   |
| GOFPANM   | 9.86022E−04  | 9.86022E−04  | 9.86022E−04  | 8.13789E−12  | +   |
| IJAYA     | 1.43908E−03  | 9.86745E−04  | 1.11186E−03  | 1.38578E−04  | +   |
| MLBSA     | 2.21804E−03  | 9.86022E−04  | 1.03108E−03  | 2.24430E−04  | +   |
| RLGBO     | 1.43848E−03  | 9.86022E−04  | 1.05002E−03  | 1.27865E−04  | +   |
| EHHO      | 3.84355E−02  | 1.05687E−03  | 6.32020E−03  | 8.04555E−03  | +   |
| SMA       | 3.01565E−01  | 1.30219E−01  | 2.22567E−01  | 5.13776E−02  | +   |
| GBO       | 1.43848E−03  | 9.86022E−04  | 1.06132E−03  | 1.38014E−04  | +   |
| INFO      | 9.98063E−04  | 9.86022E−04  | 9.86428E−04  | 2.19760E−06  | +   |
| CPA       | 1.00000E+00  | 1.02954E−03  | 1.33046E−01  | 2.98367E−01  | +   |
| ECPA      | 9.86022E−04  | 9.86022E−04  | 9.86022E−04  | 1.69365E−12  | +   |

**Table 9** Competitive statistics of RMSE metrics under SDM model

**Figure 12** Comparison of the convergence of ECPA with competing algorithms in the SDM model
of absolute errors again proves that ECPA is more stable and accurate in evaluating parameters on TDM.

In Table 14, the parameters and RMSE metrics of ECPA are recorded in comparison with the above competing algorithms under the TDM model. Among all the solving algorithms, ECPA achieves the best RMSE value of $9.8248 \times 10^{-04}$ together with GOFPANM, which is also superior to the remaining competitors. This is followed by INFO, RLGBO, GBO, MLBSA, IJAYA, and EHHO. Also, the metrics based on the Wilcoxon rank sum test show that ECPA achieves the best performance under the TDM model and is highly competitive in evaluating the unknown PV parameters.

Then, in Table 15, the four statistical metrics on the RMSE values for ECPA and similar competing algorithms for the more complex TDM model are represented. The statistical metrics based on the Wilcoxon rank sum test show that the ECPA approach is able to outperform all competitors and achieve better results. Finally, the optimization convergence graph of ECPA compared with its competitors on TDM is shown in Figure 18. It is clear from the graph that ECPA again outperforms all competing algorithms, gets the optimal convergence accuracy and the fastest convergence speed, and is able to make better use of individual information to jump out of the immature optimal solution. This shows that ECPA can be a very competitive technique for the study and extraction of PV parameters.

4.2.4 | Experimental results of PV model

This section records the analytical experiments performed by ECPA on the simulated PV assembly. Similarly, the $I-V$ and $P-V$ curves of current and power obtained from the simulations on the PV model are
recorded in Figure 19. As with the previous three PV models, the same conclusion is reached, and the perfect fit between the estimated and experimental data proves that ECPA can also accurately simulate the performance of TDM with high stability. In addition, Figure 20 shows a record plot of the ECPA technique by simulating the current data compared to the error values IAE and RE produced by the experimental data. Among them, the recorded values of IAE are less than $4.83284\times10^{-3}$ and the values of RE are in the range $[-3.90848\times10^{-2}, 5.02545\times10^{-2}]$. Among the more detailed values recorded in Table 16, the sum of IAEs for current and power reached $4.89237\times10^{-2}$ and $5.16890\times10^{-1}$. The results demonstrate that the proposed ECPA technique is capable of estimating the unknown parameters of PV components with low error and stability and accuracy.

The optimal RMSE values and performance parameters determined by ECPA and some published competing algorithms of the same type for the PV model under the PV component are documented by Table 17. As can be seen, the ECPA method competitively achieves again the RMSE value of $2.42507\times10^{-3}$, the optimal metric, with a high potential for evaluating the PV model parameters.

Table 18 further documents the statistical results of the RMSE values for the ECPA and competing algorithms under the PV model. From the table, it can be seen that the

---

**TABLE 10** IAE of ECPA on DDM

| Item | Measured data | Simulated current data | Simulated power data |
|------|---------------|------------------------|----------------------|
|      | $V$ (V) | $I$ (A)  | $I_{sim}$ (A) | $IAE_i$ (A) | $P_{sim}$ (W) | $IAE_p$ (W) |
| 1    | -0.2057 | 0.7640    | 0.763983416 | 0.000016584 | -0.157151389 | 0.00003411 |
| 2    | -0.1291 | 0.7620    | 0.762604101 | 0.0000604101 | -0.098452189 | 0.000077989 |
| 3    | -0.0588 | 0.7605    | 0.761337704 | 0.0000837704 | -0.044766657 | 0.000049257 |
| 4    | 0.0057  | 0.7605    | 0.760173794 | 0.000326206 | -0.004332991 | 0.00001859 |
| 5    | 0.0646  | 0.7600    | 0.759107687 | 0.0000892313 | 0.049038357 | 0.000057643 |
| 6    | 0.1185  | 0.7590    | 0.758121427 | 0.0000878573 | 0.089837389 | 0.000104111 |
| 7    | 0.1678  | 0.7570    | 0.757188621 | 0.000186821 | 0.127056251 | 0.000031651 |
| 8    | 0.2132  | 0.7570    | 0.756243614 | 0.000756386 | 0.161231138 | 0.0000161262 |
| 9    | 0.2545  | 0.7555    | 0.755177307 | 0.000322693 | 0.192192625 | 0.000082125 |
| 10   | 0.2924  | 0.7540    | 0.753722357 | 0.000277643 | 0.220388417 | 0.000081183 |
| 11   | 0.3269  | 0.7505    | 0.751399135 | 0.0000899135 | 0.245632377 | 0.000293927 |
| 12   | 0.3585  | 0.7465    | 0.747301441 | 0.000801441 | 0.267907567 | 0.000287317 |
| 13   | 0.3873  | 0.7385    | 0.740010655 | 0.001510655 | 0.286606127 | 0.000585077 |
| 14   | 0.4137  | 0.7280    | 0.727246946 | 0.000753054 | 0.300862061 | 0.000311539 |
| 15   | 0.4373  | 0.7065    | 0.706850292 | 0.000350292 | 0.309105633 | 0.000153183 |
| 16   | 0.4590  | 0.6755    | 0.675210539 | 0.000289461 | 0.309921637 | 0.000132863 |
| 17   | 0.4784  | 0.6320    | 0.630760758 | 0.001239242 | 0.301759946 | 0.000592854 |
| 18   | 0.4960  | 0.5730    | 0.571994736 | 0.001005264 | 0.283709389 | 0.000498611 |
| 19   | 0.5119  | 0.4990    | 0.499706140 | 0.000706140 | 0.255799573 | 0.000361473 |
| 20   | 0.5265  | 0.4130    | 0.413733678 | 0.000733678 | 0.217830781 | 0.000862821 |
| 21   | 0.5398  | 0.3165    | 0.317546209 | 0.001046209 | 0.171411443 | 0.000564743 |
| 22   | 0.5521  | 0.2120    | 0.212122996 | 0.000129996 | 0.117113106 | 0.00067906 |
| 23   | 0.5633  | 0.1035    | 0.102163275 | 0.001336725 | 0.057548573 | 0.000752977 |
| 24   | 0.5736  | -0.0100   | -0.008791751 | 0.001208249 | -0.005042948 | 0.000693052 |
| 25   | 0.5833  | -0.1230   | -0.125543432 | 0.002543432 | -0.073229484 | 0.001483584 |
| 26   | 0.5900  | -0.2100   | -0.208371578 | 0.001628422 | -0.122939231 | 0.000960769 |
| Sum of IAE | NA     | NA        | NA        | 2.12752E−02 | NA        | 8.77665E−03 |
TABLE 11  Comparison of different algorithms on DDM case

| Algorithm  | $I_{ph}$ (A) | $I_{sd1}$ (A) | $I_{sd2}$ (A) | $R_s$ (Ω) | $R_{sh}$ (Ω) | $n_1$ | $n_2$ | RSME | sig |
|-----------|--------------|---------------|---------------|-----------|-------------|-------|-------|------|-----|
| GOTLBO    | 0.76063      | 2.09918E−07   | 1.63097E−07   | 0.0367    | 54.5531     | 1.5901| 1.4407| 1.0016E−03 | +   |
| GOFPANM   | 0.76078      | 2.25974E−07   | 7.49344E−07   | 0.0367    | 55.4854     | 1.4510| 2.0000| 9.8248E−04 | =   |
| IJAYA     | 0.76074      | 4.02922E−07   | 2.63879E−07   | 0.0366    | 54.5093     | 1.9805| 1.4641| 9.8341E−04 | +   |
| MLBSA     | 0.76077      | 5.69547E−07   | 2.09888E−07   | 0.0368    | 55.2238     | 1.8929| 1.4460| 9.8384E−04 | +   |
| RLGBO     | 0.76079      | 9.29128E−07   | 2.05144E−07   | 0.0368    | 55.8528     | 1.4429| 1.4510| 9.8275E−04 | +   |
| EHHO      | 0.76072      | 4.31401E−07   | 3.72599E−07   | 0.0346    | 67.2448     | 1.5141| 1.9726| 1.3258E−03 | +   |
| SMA       | 0.77286      | 3.04286E−07   | 8.01849E−07   | 0.0062    | 39.4151     | 1.8941| 1.5988| 5.8204E−02 | +   |
| GBO       | 0.76073      | 2.61384E−07   | 4.78711E−07   | 0.0366    | 55.5441     | 1.4634| 1.9963| 9.8375E−04 | +   |
| INFO      | 0.76078      | 7.37901E−07   | 2.25355E−07   | 0.0367    | 55.4715     | 1.9941| 1.4508| 9.8255E−04 | +   |
| CPA       | 0.76057      | 5.17633E−07   | 6.08020E−07   | 0.0329    | 92.9175     | 1.9475| 1.5530| 1.9398E−03 | +   |
| ECPA      | 0.76078      | 2.25970E−07   | 7.49382E−07   | 0.0367    | 55.4855     | 1.4510| 2.0000| 9.8248E−04 | +   |

TABLE 12  Competitive statistics of RMSE metrics under DDM model

| Algorithm | Max     | Min     | Mean    | SD      | sig |
|-----------|---------|---------|---------|---------|-----|
| GOTLBO    | 2.44346E−03 | 1.00163E−03 | 1.34767E−03 | 3.43664E−04 | +   |
| GOFPANM   | 9.89172E−04 | 9.82485E−04 | 9.84562E−04 | 2.03779E−06 | +   |
| IJAYA     | 1.50228E−03 | 9.83411E−04 | 1.13922E−03 | 1.63881E−04 | +   |
| MLBSA     | 1.16372E−03 | 9.83838E−04 | 1.01723E−03 | 5.11670E−05 | +   |
| RLGBO     | 1.43846E−03 | 9.82747E−04 | 1.06636E−03 | 1.56082E−04 | +   |
| EHHO      | 1.80916E−02 | 1.32584E−03 | 5.98293E−03 | 5.32951E−03 | +   |
| SMA       | 3.07698E−01 | 5.82039E−02 | 2.04135E−01 | 7.52167E−02 | +   |
| GBO       | 1.89676E−03 | 9.83754E−04 | 1.18610E−03 | 2.47451E−04 | +   |
| INFO      | 1.47219E−03 | 9.82553E−04 | 1.08369E−03 | 1.78968E−04 | +   |
| CPA       | 1.00000E+00 | 1.93977E−03 | 4.16465E−01 | 4.65841E−01 | +   |
| ECPA      | 9.88641E−04 | 9.82485E−04 | 9.84353E−04 | 1.93792E−06 | +   |

FIGURE 15  Comparison of the convergence of ECPA with competing algorithms in the DDM model
ECPA method not only achieves the optimal \(2.42507 \times 10^{-03}\) for the maximum, minimum, and mean values of the RMSE metrics, but also achieves the optimal \(5.07771 \times 10^{-13}\) for the standard deviation. The Wilcoxon rank sum test metric also proves that the ECPA solution has better accuracy and stability. Finally, Figure 21 visualizes the convergence curve plotted by ECPA for solving the PV model parameter estimation. Again, at each stage of the algorithm execution, ECPA maintains the fastest search speed and the best convergence accuracy compared to its strong competitors, so the ECPA method can be a more efficient and accurate technique to solve the PV system parameter evaluation problem.

### 4.2.5 Average CPU time

To more fully evaluate the computational efficiency of the ECPA algorithm, the key metric of CPU time consumed by the algorithm needs to be further documented as well. In this section, the average CPU time in milliseconds consumed by ECPA is recorded for 30 experiments run independently under four different models. Detailed results of the times are recorded in Table 19, and more visual results are also plotted by Figure 22. From the figure, it can be visualized that the algorithm time spent on evaluating the optimal parameters on different PV models varies for different techniques. Combining the results generated from the experiments performed on the four models shows that the original CPA algorithm, although spending the least average CPU time compared to some complex algorithms, performs so poorly for PV parameter extraction and performance simulation of the actual PV model that it often falls into immature convergence. However, the ECPA algorithm, after a mature balance between exploration and exploitation, is able to better solve the PV parameters and simulate the actual PV model.
performance with qualitative improvements in both solution accuracy and convergence speed, while only adding a few milliseconds and not significantly increasing the CPU time consumed. On the contrary, the GOFPANM algorithm, which is second only to ECPA in terms of solution performance, consumes a substantial increase in CPU time, reaching a 100 times or more than the ECPA algorithm. In particular, on the complex model of TDM, the average CPU time spent by ECPA is very close to that on the SDM, DDM, and PV models, while the time spent by GOFPANM on the TDM model is very disparate compared to the other three models. Furthermore, although the CPU time spent by GOTLBO, IJAYA, MLBSA, EHHO, SMA, GBO, INFO, and ECPA are very close, the results obtained from the experiments show that the actual performance demonstrated in simulating the actual parameters of the PV model is far behind that of ECPA. Therefore, the consistency of the ECPA method in evaluating the performance and efficiency of PV systems further demonstrates the efficiency of ECPA as a potential method that can evaluate the most accurate parameters.

| Item | Measured data | Simulated current data | Simulated power data |
|------|---------------|------------------------|----------------------|
|      | $V$ (V) | $I$ (A) | $I_{\text{sim}}$ (A) | IAE$_{I}$ (A) | $P_{\text{sim}}$ (W) | IAE$_{P}$ (W) |
| 1    | $-0.2057$ | $0.7640$ | $0.763983369$ | $0.000016631$ | $-0.157151379$ | $0.000003421$ |
| 2    | $-0.1291$ | $0.7620$ | $0.762604072$ | $0.0000060472$ | $-0.098452186$ | $0.000073986$ |
| 3    | $-0.0588$ | $0.7605$ | $0.761337691$ | $0.0000837691$ | $-0.044766656$ | $0.000049256$ |
| 4    | $0.0057$   | $0.7605$ | $0.761737979$ | $0.000326203$ | $0.004332991$ | $0.00001859$  |
| 5    | $0.0646$   | $0.7600$ | $0.759107702$ | $0.000892998$ | $0.049038358$ | $0.00057642$  |
| 6    | $0.1185$   | $0.7590$ | $0.758121453$ | $0.000878547$ | $0.089837392$ | $0.000104108$ |
| 7    | $0.1678$   | $0.7570$ | $0.757188654$ | $0.000186854$ | $0.127056256$ | $0.000031656$ |
| 8    | $0.2132$   | $0.7570$ | $0.756243649$ | $0.000756351$ | $0.161231146$ | $0.000161254$ |
| 9    | $0.2545$   | $0.7555$ | $0.755177338$ | $0.000326262$ | $0.192192632$ | $0.000082118$ |
| 10   | $0.2924$   | $0.7540$ | $0.753722375$ | $0.000277625$ | $0.220388423$ | $0.000081177$ |
| 11   | $0.3269$   | $0.7505$ | $0.751399134$ | $0.000899134$ | $0.245632377$ | $0.000293928$ |
| 12   | $0.3585$   | $0.7465$ | $0.747301416$ | $0.000801416$ | $0.267907558$ | $0.000287308$ |
| 13   | $0.3873$   | $0.7385$ | $0.740010611$ | $0.001501061$ | $0.286606110$ | $0.000585060$ |
| 14   | $0.4137$   | $0.7280$ | $0.727246892$ | $0.000753108$ | $0.300862039$ | $0.000311561$ |
| 15   | $0.4373$   | $0.7065$ | $0.706850247$ | $0.000350247$ | $0.309105613$ | $0.000153163$ |
| 16   | $0.4590$   | $0.6755$ | $0.675210518$ | $0.000289482$ | $0.309921628$ | $0.000132872$ |
| 17   | $0.4784$   | $0.6320$ | $0.630706768$ | $0.001239232$ | $0.301755951$ | $0.000592849$ |
| 18   | $0.4960$   | $0.5730$ | $0.571994772$ | $0.001005228$ | $0.283709407$ | $0.000498593$ |
| 19   | $0.5119$   | $0.4990$ | $0.499706187$ | $0.000706187$ | $0.255799597$ | $0.000361497$ |
| 20   | $0.5265$   | $0.4130$ | $0.413733713$ | $0.000733713$ | $0.217830800$ | $0.000386300$ |
| 21   | $0.5398$   | $0.3165$ | $0.317546219$ | $0.001046219$ | $0.171411449$ | $0.000564749$ |
| 22   | $0.5521$   | $0.2120$ | $0.212122974$ | $0.000122974$ | $0.117130949$ | $0.000678949$ |
| 23   | $0.5633$   | $0.1035$ | $0.102163228$ | $0.001336772$ | $0.057548546$ | $0.000753004$ |
| 24   | $0.5736$   | $-0.0100$ | $-0.008791790$ | $0.001208210$ | $-0.005042971$ | $0.000693029$ |
| 25   | $0.5833$   | $-0.1230$ | $-0.125543450$ | $0.002543450$ | $-0.073229494$ | $0.001483594$ |
| 26   | $0.5900$   | $-0.2100$ | $-0.208371531$ | $0.001628469$ | $-0.122939203$ | $0.000960797$ |
| Sum of IAE | NA | NA | NA | 2.12752E$-02$ | NA | 8.77667E$-03$ |
### Table 14: Comparison among different algorithms on TDM

| Algorithm    | $I_{ph}$ (A) | $I_{sd1}$ (A) | $I_{sd2}$ (A) | $I_{sd3}$ (A) | $R_s$ (Ω) | $R_{sh}$ (Ω) | $n_1$ | $n_2$ | $n_3$ | RSME | sig |
|-------------|--------------|--------------|--------------|--------------|-----------|-------------|-----|-----|-----|------|-----|
| GOTLBO      | 0.76081      | 1.180E−07    | 1.218E−07    | 2.273E−07    | 0.03647   | 55.17204    | 1.61704 | 1.88967 | 1.45916 | 9.8976E−04 | +   |
| GOFPANM     | 0.76078      | 2.260E−07    | 4.112E−13    | 7.493E−07    | 0.03674   | 55.48544    | 1.45102 | 1.45079 | 2.00000 | 9.8248E−04 | =   |
| IJAYA       | 0.76085      | 2.242E−07    | 2.917E−07    | 7.493E−07    | 0.03645   | 55.50943    | 1.98944 | 1.88962 | 1.80788 | 9.8459E−04 | +   |
| MLBSA       | 0.76078      | 5.702E−07    | 2.465E−07    | 7.642E−10    | 0.03665   | 55.02026    | 1.99999 | 1.45856 | 1.44959 | 9.8271E−04 | +   |
| RLGBO       | 0.76078      | 7.822E−07    | 2.361E−07    | 2.224E−07    | 0.03675   | 55.32192    | 2.00000 | 1.44970 | 1.8265E | +   |
| EHHO        | 0.76078      | 9.995E−07    | 1.305E−07    | 9.995E−07    | 0.03690   | 68.38694    | 1.99640 | 1.40826 | 1.99949 | 1.0441E−03 | +   |
| SMA         | 0.78826      | 1.000E−12    | 1.000E−06    | 6.909E−07    | 0.03662   | 61.47905    | 1.00000 | 2.00000 | 1.57917 | 6.9149E−02 | +   |
| GBO         | 0.76078      | 4.925E−07    | 5.121E−09    | 2.579E−07    | 0.03657   | 55.56671    | 1.98986 | 1.99920 | 1.46236 | 9.8378E−04 | +   |
| INFO        | 0.76081      | 7.822E−07    | 2.361E−23    | 2.224E−07    | 0.03675   | 55.32192    | 2.00000 | 1.44970 | 1.8265E | +   |
| CPA         | 0.75869      | 1.670E−07    | 8.493E−08    | 3.096E−07    | 0.03810   | 92.48047    | 1.54948 | 1.39006 | 1.86727 | 1.8180E−03 | ++  |
| ECPA        | 0.76078      | 2.259E−07    | 2.719E−08    | 7.225E−07    | 0.03674   | 55.48621    | 1.45100 | 2.00000 | 2.00000 | 9.8248E−04 | +   |

### Table 15: Statistics of RMSE obtained by competing algorithms on TDM

| Algorithm    | Max       | Min       | Mean      | SD        | sig |
|-------------|-----------|-----------|-----------|-----------|-----|
| GOTLBO      | 2.91773E−03 | 9.89758E−04 | 1.48219E−03 | 3.96541E−04 | +   |
| GOFPANM     | 9.94427E−04 | 9.82488E−04 | 9.85451E−04 | 2.49695E−06 | +   |
| IJAYA       | 1.96859E−03 | 9.86037E−04 | 1.22805E−03 | 2.36893E−04 | +   |
| MLBSA       | 1.46441E−03 | 9.84585E−04 | 1.11274E−03 | 1.78402E−04 | +   |
| RLGBO       | 1.42333E−03 | 9.82712E−04 | 1.07685E−03 | 1.50788E−04 | +   |
| EHHO        | 3.15813E−02 | 1.04411E−03 | 6.54243E−03 | 6.84803E−03 | +   |
| SMA         | 3.66903E−01 | 6.91491E−02 | 2.13797E−01 | 7.97426E−02 | +   |
| GBO         | 1.86484E−03 | 9.83779E−04 | 1.25733E−03 | 2.39768E−04 | +   |
| INFO        | 3.66903E−01 | 9.82647E−04 | 1.11810E−03 | 7.84024E−04 | +   |
| CPA         | 1.00000E+00 | 1.81800E−03 | 6.09188E−01 | 4.88079E−01 | +   |
| ECPA        | 9.87808E−04 | 9.82485E−04 | 9.84104E−04 | 1.76360E−06 | +   |

![Figure 18](image) Comparison of the convergence of ECPA with competing algorithms in the TDM model
in a short period of time while achieving the highest performance.

4.3 | ECPA results through manufacturer’s datasheet

This part of the experiment identifies valid parameters for thin-film (ST40), monocrystalline (SM55) and polycrystalline (KC200GT) PV modules to verify the stability and effectiveness of the proposed ECPA method under two PV models. The datasets used in the experiments were obtained from reliable information in the manufacturer’s data sheets. Among them, these data set suites were always used by the researchers to detect the efficiency of the evaluation of different unknown parameters for the PV models with more precise test implications. The above three PV modules were used by Alam et al. in the flower pollination algorithm (FPA) to test the evaluation capability on PV models, and Yu et al. also used the data sets from these PV modules to test the detailed performance of the JAYA (PGJAYA) algorithm. In summary, these three extracted accurate data sets play a crucial role in the evaluation of the efficiency and stability of the algorithm under PV systems.

Two sets of experiments under SDM and DDM models were conducted by controlling different temperatures and different irradiations and extracting data precisely from the above data set. First, the first set of experiments was conducted when the temperature was constant at 20°C, and by varying the irradiation in the light conditions to 200, 400, 600, 800, and 1000 W/m², the exact parameter tables mined by the ECPA method are shown in Tables 20, 21, and 22, and ECPA demonstrates very low RMSE values. At this point, to
verify the accuracy of the parameters extracted by the evaluation, the $I$–$V$ characteristic diagrams of the approximate currents simulated by ECPA under the three PV modules ST40, SM55 and KC200GT are plotted by Figures 23, 24, and 25. It can be visually seen that the tapped PV parameters make the estimated and measured curves match highly and the ECPA demonstrates a high degree of stability performance. Next, the second set of experiments was conducted, when the irradiation of the control light condition was a constant 1000 W/m$^2$, and the ambient temperature was set at 25°C, 40°C, 50°C, and 70°C for comparison experiments. At this point the results evaluated by the ECPA method for the real parameters of the SDM and DDM models are recorded in Tables 23, 24, and 25. Similarly, the $I$–$V$ characteristic diagrams of the simulated currents were used to verify the match between the simulated data and the measured data extracted from the data set by the ECPA method for different temperatures, as documented in Figures 26, 27, and 28. The two sets of experiments allow us to conclude that the ECPA technique proposed in this study is able to simulate the actual performance of the PV model stably and extract the important unknown parameters of the PV model accurately when the ambient temperature and the irradiation of the light undergo turbulent changes.

| Item | Measured data | Simulated current data | Simulated power data |
|------|---------------|------------------------|----------------------|
|      | $V$ (V)       | $I$ (A)                | $I_{sim}$ (A)        | $I$ (A) | $P_{sim}$ (W) | $I$ (A) |
| 1    | 0.1248        | 1.0315                 | 1.029119192          | 0.002380808 | 0.128434075 | 0.000297125 |
| 2    | 1.8093        | 1.0300                 | 1.027381086          | 0.002618914 | 1.858840598 | 0.004738402 |
| 3    | 3.3511        | 1.0260                 | 1.025741793          | 0.000258207 | 3.437363322 | 0.000865278 |
| 4    | 4.7622        | 1.0220                 | 1.024107136          | 0.002107136 | 4.877003004 | 0.010034604 |
| 5    | 6.0538        | 1.0180                 | 1.022291773          | 0.004291773 | 6.188749938 | 0.025981538 |
| 6    | 7.2364        | 1.0155                 | 1.019930640          | 0.004430640 | 7.380626084 | 0.032061884 |
| 7    | 8.3189        | 1.0140                 | 1.016363059          | 0.002363059 | 8.455022649 | 0.019658049 |
| 8    | 9.3097        | 1.0100                 | 1.010496103          | 0.000496103 | 9.407415567 | 0.04181567  |
| 9    | 10.2163       | 1.0035                 | 1.000628925          | 0.002871075 | 10.222725285 | 0.029331765 |
| 10   | 11.0449       | 0.9880                 | 0.984548343          | 0.003451657 | 10.874237992 | 0.038123208 |
| 11   | 11.8018       | 0.9630                 | 0.959521654          | 0.003478346 | 11.324082661 | 0.041050739 |
| 12   | 12.4929       | 0.9255                 | 0.922838813          | 0.002661187 | 11.528933005 | 0.033245945 |
| 13   | 13.1231       | 0.8725                 | 0.872599674          | 0.000099674 | 11.451212778 | 0.001308028 |
| 14   | 13.6983       | 0.8075                 | 0.807274287          | 0.000225713 | 11.058285369 | 0.003091881 |
| 15   | 14.2221       | 0.7265                 | 0.728336509          | 0.001836509 | 10.358474664 | 0.026119014 |
| 16   | 14.6995       | 0.6345                 | 0.637138032          | 0.002638032 | 9.365610507 | 0.038777757 |
| 17   | 15.1346       | 0.5345                 | 0.536213091          | 0.001713091 | 8.115370653 | 0.025926953 |
| 18   | 15.5311       | 0.4275                 | 0.429511346          | 0.002011346 | 6.670783663 | 0.031238413 |
| 19   | 15.8929       | 0.3185                 | 0.318774494          | 0.000274494 | 5.066251158 | 0.004362508 |
| 20   | 16.2229       | 0.2085                 | 0.207389509          | 0.00110491 | 3.364459259 | 0.018015391 |
| 21   | 16.5241       | 0.1010                 | 0.096167165          | 0.004832835 | 1.589075848 | 0.079858252 |
| 22   | 16.7987       | −0.0080                | −0.008325396         | 0.000325396 | −0.139855838 | 0.00566238 |
| 23   | 17.0499       | −0.1110                | −0.110936495         | 0.000063505 | −1.891456148 | 0.001082752 |
| 24   | 17.2793       | −0.2090                | −0.209247278         | 0.00247278 | −3.615646485 | 0.004272785 |
| 25   | 17.4885       | −0.3030                | −0.300863594         | 0.002136406 | −5.261652962 | 0.037362538 |

| Sum of IAE | NA | NA | 4.89237E−02 | NA | 5.16890E−01 |
# TABLE 17 Comparison results for PV model

| Item   | $I_{ph}$ (A) | $I_{sd}$ (A) | $R_s$ (Ω) | $R_{sh}$ (Ω) | n | RMSE sig |
|--------|--------------|--------------|-----------|--------------|----|----------|
| GOTLBO | 1.03052      | 3.44316E−06  | 1.20200   | 968.52219    | 48.60038 | 2.42637E−03 + |
| GOFPANM | 1.03051      | 3.48226E−06  | 1.20127   | 981.98229    | 48.64283 | 2.42507E−03 = |
| IJAYA  | 1.03046      | 3.51682E−06  | 1.20022   | 990.77992    | 48.68077 | 2.42527E−03 + |
| MLBSA  | 1.03051      | 3.48226E−06  | 1.20127   | 981.98384    | 48.64284 | 2.42507E−03 = |
| RLGBO  | 1.03051      | 3.48226E−06  | 1.20127   | 981.98084    | 48.64283 | 2.42507E−03 = |
| EHHO   | 1.02574      | 2.87896E−06  | 1.23398   | 1865.23123   | 47.90173 | 3.08151E−03 + |
| SMA    | 0.85920      | 8.10115E−07  | 0.68625   | 774.99571    | 44.21801 | 1.19818E−01 + |
| GBO    | 1.03051      | 3.48251E−06  | 1.20127   | 982.26559    | 48.64310 | 2.42507E−03 = |
| INFO   | 1.03051      | 3.48226E−06  | 1.20127   | 981.98236    | 48.64284 | 2.42507E−03 = |
| CPA    | 1.02267      | 3.32035E−06  | 1.12921   | 1112.50479   | 48.48872 | 8.66247E−03 + |
| ECPA   | 1.03051      | 3.48224E−06  | 1.20127   | 981.97170    | 48.64281 | 2.42507E−03 = |

# TABLE 18 Statistics of RMSE obtained by competing algorithms on PV model

| Algorithm | Max       | Min       | Mean       | SD         | sig |
|-----------|-----------|-----------|------------|------------|-----|
| GOTLBO    | 2.74662E−03 | 2.42637E−03 | 2.48742E−03 | 9.30119E−05 | +   |
| GOFPANM   | 2.42507E−03 | 2.42507E−03 | 2.45467E−03 | 8.89495E−05 | +   |
| IJAYA     | 2.99555E−03 | 2.42527E−03 | 2.48704E−03 | 1.11521E−04 | +   |
| MLBSA     | 2.89530E−03 | 2.42507E−03 | 2.45467E−03 | 8.89495E−05 | +   |
| RLGBO     | 2.74251E−01 | 2.42507E−03 | 2.12270E−02 | 6.88630E−02 | +   |
| EHHO      | 4.38170E−01 | 3.08151E−03 | 6.55448E−02 | 1.05279E−01 | +   |
| SMA       | 1.00000E+00 | 1.19818E−01 | 5.09072E−01 | 1.89754E−01 | +   |
| GBO       | 2.55077E−03 | 2.42507E−03 | 2.43802E−03 | 2.59910E−05 | +   |
| INFO      | 6.56520E+00 | 2.42507E−03 | 7.64909E−01 | 1.50543E+00 | +   |
| CPA       | 1.00000E+00 | 8.66247E−03 | 8.04056E−01 | 3.78762E−01 | +   |
| ECPA      | 2.42507E−03 | 2.42507E−03 | 2.42507E−03 | 5.07771E−13 |     |

**FIGURE 21** Comparison of the convergence of ECPA with competing algorithms in the PV model
TABLE 19  Average CPU time of competitive algorithms on four PV models (ms)

| Item | ECPA  | GOTLBO | GOFPANM | IJAYA | MLBSA | RLGO | EHHO | SMA | GBO | INFO | CPA |
|------|-------|--------|---------|-------|-------|------|------|-----|-----|------|-----|
| SDM  | 22.4260 | 21.8161 | 4785.4516 | 19.3063 | 19.0807 | 20.5234 | 23.0620 | 19.1964 | 19.1724 | 18.9359 | 17.4313 |
| DDM  | 23.2104 | 22.3682 | 7119.3313 | 18.4964 | 18.3396 | 19.8813 | 22.6219 | 19.0193 | 19.0490 | 19.1656 | 17.8000 |
| TDM  | 22.9052 | 21.7599 | 9291.8932 | 17.8786 | 17.6344 | 19.1927 | 21.8844 | 18.5146 | 18.6708 | 18.7089 | 17.4625 |
| PV   | 21.6578 | 21.0667 | 4685.6031 | 19.4948 | 19.1422 | 20.5745 | 23.2365 | 19.0995 | 19.0974 | 18.9661 | 17.2964 |

FIGURE 22  Average CPU time of ECPA compared with other competing algorithms on four PV models

TABLE 20  Estimation of ST40 parameters using ECPA at different irradiances and the same temperature of 25°C

| Parameters | Irradiance |
|------------|------------|
|            | 200 W/m²   | 400 W/m²   | 600 W/m²   | 800 W/m²   | 1000 W/m²  |
| SDM        |            |            |            |            |            |
| $I_{ph}$ (A) | 0.533137328 | 1.067544952 | 1.604810356 | 2.138014261 | 2.675799818 |
| $I_{sd}$ (A) | 1.42971E−06 | 1.84852E−06 | 1.44187E−06 | 1.15821E−06 | 1.52898E−06 |
| $R_s$ (Ω)  | 1.185706678 | 1.080603387 | 1.112612769 | 1.125282297 | 1.113215120 |
| $R_{sh}$ (Ω) | 344.9840184 | 362.5015039 | 347.6870904 | 332.896835  | 357.596497  |
| $n_1$      | 1.747111231 | 1.778514047 | 1.745124233 | 1.718684331 | 1.750340344 |
| RMSE       | 4.77201E−04 | 6.30725E−04 | 6.74036E−04 | 7.73905E−04 | 7.34099E−04 |
| DDM        |            |            |            |            |            |
| $I_{ph}$ (A) | 0.533248145 | 1.067529700 | 1.604774442 | 2.137827099 | 2.675799817 |
| $I_{sd1}$ (A) | 4.33406E−07 | 1.83805E−06 | 1.33899E−06 | 2.94244E−11 | 1.52880E−06 |
| $I_{sd2}$ (A) | 2.58026E−05 | 3.79261E−08 | 1.96085E−06 | 1.21562E−06 | 3.23820E−13 |
| $R_s$ (Ω)  | 1.442600810 | 1.080547183 | 1.116743193 | 1.124795675 | 1.113225921 |
| $R_{sh}$ (Ω) | 356.6061365 | 362.8945703 | 349.3612557 | 336.4094110 | 357.5983749 |
| $n_1$      | 1.606718347 | 1.784578974 | 1.736635002 | 1.208495394 | 1.750326676 |
| $n_2$      | 3.061556909 | 1.667657351 | 2.860894360 | 1.725751899 | 1.729278210 |
| RMSE       | 4.55474E−04 | 6.31744E−04 | 6.77171E−04 | 7.86618E−04 | 7.34099E−04 |
### Table 21  
Estimation of SM55 parameters using ECPA at different irradiations and the same temperature of 25°C

| Parameters | 200 W/m² | 400 W/m² | 600 W/m² | 800 W/m² | 1000 W/m² |
|------------|----------|----------|----------|----------|-----------|
| $I_{ph}$ (A) | 0.692013715 | 1.382843964 | 2.070898191 | 2.760380985 | 3.450104156 |
| $I_{sd}$ (A) | 1.30912E−07 | 1.00439E−07 | 1.55494E−07 | 3.30507142 | 0.329155854 |
| $R_s$ (Ω) | 0.380397623 | 427.0584105 | 450.0317518 | 459.8843916 | 483.8760151 |
| $n$ | 1.370891394 | 1.352004261 | 1.387523390 | 1.381144407 | 1.395736829 |
| RMSE | 5.20542E−04 | 7.07608E−04 | 8.23949E−04 | 6.69216E−04 | 1.15022E−03 |

### Table 22  
Estimation of KC200GT parameters using ECPA at different irradiations and the same temperature of 25°C

| Parameters | 200 W/m² | 400 W/m² | 600 W/m² | 800 W/m² | 1000 W/m² |
|------------|----------|----------|----------|----------|-----------|
| $I_{ph}$ (A) | 1.646133747 | 3.287851232 | 4.934308104 | 6.571125421 | 8.21689308 |
| $I_{sd}$ (μA) | 5.32604E−10 | 1.48942E−09 | 3.86161E−09 | 9.91520E−10 | 2.24202E−09 |
| $R_s$ (Ω) | 0.379319097 | 0.335387980 | 0.337337800 | 0.356650142 | 0.343813883 |
| $R_n$ (Ω) | 690.9887710 | 752.0305726 | 742.9984321 | 755.2939167 | 763.6023949 |
| $n$ | 1.004244581 | 1.055024114 | 1.104022895 | 1.037115878 | 1.076411792 |
| RMSE | 1.41875E−03 | 1.42620E−03 | 1.29767E−03 | 1.64751E−03 | 1.53903E−03 |

| Parameters | 200 W/m² | 400 W/m² | 600 W/m² | 800 W/m² | 1000 W/m² |
|------------|----------|----------|----------|----------|-----------|
| $I_{ph}$ (A) | 1.6416107285 | 3.287850399 | 4.934307782 | 6.570961140 | 8.216842190 |
| $I_{sd}$ (A) | 4.83112E−10 | 4.50486E−10 | 3.86161E−09 | 9.91520E−10 | 2.24202E−09 |
| $R_s$ (Ω) | 0.382464864 | 0.353612099 | 0.337337745 | 0.360386735 | 0.343839393 |
| $R_n$ (Ω) | 693.33492922 | 752.0144040 | 743.0167469 | 787.3127145 | 767.1529583 |
| $n$ | 1.000017009 | 1.063741319 | 1.103711587 | 1.076411792 | 3.850109410 |
| RMSE | 1.41454E−03 | 1.42645E−03 | 1.29767E−03 | 1.51068E−03 | 1.53893E−03 |
FIGURE 23 Estimated and measured ST40 properties at the same temperature and under different irradiation. (A) Single diode model (SDM) and (B) double diode model (DDM).

FIGURE 24 Estimated and measured SM55 properties at the same temperature and under different irradiation. (A) Single diode model (SDM) and (B) double diode model (DDM).

FIGURE 25 Estimated and measured KC200GT properties at the same temperature and under different irradiation. (A) Single diode model (SDM) and (B) double diode model (DDM).
TABLE 23  Estimation of ST40 parameters using ECPA at different temperatures and the same irradiation of 1000 W/m²

| Parameters | Temperatures | 25°C | 40°C | 50°C | 70°C |
|------------|--------------|------|------|------|------|
| SDM        |              |      |      |      |      |
| $I_{ph}$ (A) |              | 2.675799332 | 2.680912203 | 2.691968060 | 2.692329482 |
| $I_{sd}$ (A) |              | 1.52887E-06 | 5.66629E-06 | 1.86808E-05 | 8.75214E-05 |
| $R_{s}$ (Ω) |              | 1.113220715 | 1.129294243 | 1.149590463 | 1.125889237 |
| $R_{sh}$ (Ω) |              | 357.5969515 | 364.1031140 | 295.0300872 | 367.7528496 |
| $n$        |              | 1.750331493 | 1.722562130 | 1.717576917 | 1.727319018 |
| RMSE       |              | 7.34099E-04 | 1.32141E-03 | 1.82326E-03 | 7.77718E-04 |
| DDM        |              |      |      |      |      |
| $I_{ph}$ (A) |              | 2.675686219 | 2.680798650 | 2.688901054 | 2.692329477 |
| $I_{sd1}$ (A) |              | 9.58702E-07 | 1.01114E-07 | 1.86572E-09 | 8.75213E-05 |
| $I_{sd2}$ (A) |              | 9.51564E-07 | 5.79537E-06 | 3.81945E-05 | 5.61665E-10 |
| $R_{s}$ (Ω) |              | 1.116203874 | 1.129926168 | 1.271904170 | 1.12588711 |
| $R_{sh}$ (Ω) |              | 360.5769305 | 366.7055432 | 381.6014442 | 367.7532252 |
| $n$        |              | 1.713414625 | 1.554284243 | 1.016591054 | 1.72731969 |
| RMSE       |              | 7.45421E-04 | 1.32669E-03 | 1.58958E-03 | 7.77718E-04 |

TABLE 24  Estimation of SM55 parameters using ECPA at different temperatures and the same irradiation of 1000 W/m²

| Parameters | Temperatures | 25°C | 40°C | 60°C |
|------------|--------------|------|------|------|
| SDM        |              |      |      |      |
| $I_{ph}$ (A) |              | 3.450103669 | 3.469137175 | 3.494609802 |
| $I_{sd}$ (A) |              | 1.71176E-07 | 1.14510E-06 | 6.90967E-06 |
| $R_{s}$ (Ω) |              | 0.329144438 | 0.313096115 | 0.318704964 |
| $R_{sh}$ (Ω) |              | 483.9146774 | 533.0796934 | 484.8695861 |
| $n$        |              | 1.395763810 | 1.417838765 | 1.405144373 |
| RMSE       |              | 1.14622E-03 | 3.7881E-03  | 3.78039E-03 |
| DDM        |              |      |      |      |
| $I_{ph}$ (A) |              | 3.450103564 | 3.469065374 | 3.494608469 |
| $I_{sd1}$ (A) |              | 1.71154E-07 | 3.58867E-07 | 6.90950E-06 |
| $I_{sd2}$ (A) |              | 1.42068E-08 | 1.21908E-06 | 1.34448E-17 |
| $R_{s}$ (Ω) |              | 0.329147705 | 0.316904117 | 0.318705726 |
| $R_{sh}$ (Ω) |              | 483.9004829 | 541.788911 | 484.8839654 |
| $n$        |              | 1.395752859 | 1.349550930 | 1.405141753 |
| RMSE       |              | 1.14621E-03 | 3.89753E-03 | 3.78039E-03 |
The unity of efficiency and stability determines that ECPA can be a new competitive method to deal with the parameter evaluation of PV systems.

Owing to the strong optimization capability, the ECPA can also be a great potential tool to deal with some complex problems such as drug discovery, disease module identification, and pharmacoinformatic data mining. It also can be combined with other tools for tackling the practicle problems such as fault diagnosis, active surveillance, urban road planning, human motion capture, and kayak cycle phase segmentation.

### TABLE 25
Estimation of KC200GT parameters using ECPA at different temperatures and the same irradiation of 1000 W/m²

| Parameters | Temperatures |          |          |          |
|------------|--------------|----------|----------|----------|
|            | 25°C         | 50°C     | 75°C     |          |
| SDM        |              |          |          |          |
| $I_{ph}$ (A) | 8.216892762  | 8.29505561 | 8.377663878 |          |
| $I_{sd}$ (A) | 2.24181E−09  | 1.25954E−07 | 1.63080E−06 |          |
| $R_s$ (Ω)  | 0.343814913  | 0.335654348 | 0.342498214 |          |
| $R_{sh}$ (Ω) | 763.4733387  | 953.8752252 | 790.5069951 |          |
| $n$        | 1.076407256  | 1.11729274 | 1.10147803 |          |
| RMSE       | 1.53903E−03  | 2.74651E−03 | 4.47293E−03 |          |
| DDM        |              |          |          |          |
| $I_{ph}$ (A) | 8.216858068  | 8.29508004 | 8.377662978 |          |
| $I_{sd1}$ (A) | 2.20345E−09  | 6.95879E−12 | 4.64020E−14 |          |
| $I_{sd2}$ (A) | 5.15054E−09  | 1.25959E−07 | 1.63082E−06 |          |
| $R_s$ (Ω)  | 0.343908527  | 0.335656497 | 0.342497964 |          |
| $R_{sh}$ (Ω) | 765.8512056  | 953.7567005 | 790.5575257 |          |
| $n_1$      | 1.075616661  | 1.030106987 | 3.919760669 |          |
| $n_2$      | 1.652379076  | 1.117311128 | 1.101479928 |          |
| RMSE       | 1.53941E−03  | 2.74653E−03 | 4.47293E−03 |          |

FIGURE 26 Estimated and measured ST40 properties under the same irradiation and at different temperatures. (A) Single diode model (SDM) and (B) double diode model (DDM).
5 | CONCLUSIONS AND FUTURE DIRECTIONS

In this study, an enhanced ECPA algorithm is proposed to better handle the optimization problem of high-dimensional multimodal functions to accurately and stably evaluate the parameters to be evaluated in solar PV systems, and ultimately to achieve more efficient utilization of solar energy as a renewable energy source. The proposed ECPA incorporates extremal optimization and local search strategy based on NMs, which allows more frequent communication between individuals in the population, thus enabling a more comprehensive exploration and more accurate exploitation of the algorithm. ECPA adjusts the mutation scheme of individuals through extremal optimization strategy to promote the migration of the population toward the direction close to the optimal solution, and the incorporation of NMs local search strategy can continuously improve the genes of inferior individuals so that they can escape from the immature convergence solution, thus overcoming the defects such as stagnation and easy to fall into immature convergence in the development stage of the original CPA.

Based on the multiple competitive experiments explored in this study and their results the following conclusions can be drawn.
I. ECPA is tested on the IEEE CEC 2020 function set with 10 advanced competitive algorithms such as OBSCA, SCADE, and HGWO, and the competitive performance and result comparison of ECPA are well analyzed. The mean, standard deviation, Wilcoxon singed-rank test metrics and Friedman test ranking results of ECPA are further reliable and accurate compared to other competitive algorithms, both on simple unimodal and multimodal functions and on complex composite and hybrid functions.

II. Based on the convergence curves, it can be seen that ECPA is able to achieve better convergence accuracy and faster convergence in the process of CEC 2020 function test suite and PV model unknown parameter extraction.

III. The competitive experiments for PV parameter evaluation demonstrate that ECPA has the lowest RMSE value, maximum value, minimum value, mean value and standard deviation when comparing nine already existing competitive algorithms such as GOTLBO, GOFPANM and IJAVA, showing a more accurate and efficient level.

IV. The stability simulation experiments under the manufacturer’s datasheet showed that the ECPA maintained the same highly stable performance under different irradiation and temperature.

V. All evaluation experiments and results demonstrate that ECPA can be a novel and promising technique to solve optimization problems of multimodal functions and parametric evaluation of solar PV parameters. Thus, it can become a novel and high standard technique for identifying parameters to be determined in complex PV systems.

It is worth noting that the values of the parameters in ECPA need to be carefully tuned because the values of the parameters affect the quality of the solution and the parameter settings depend on the different problems. Second, the same limitation as many advanced hybrid metaheuristic algorithms exists, that is, the ECPA consumes additional computational resources compared to the original CPA due to the revision of the new structure, and it is still worthwhile to further investigate how to balance the computational resources consumed and the accuracy exhibited by the algorithm.

Although the above experiments have verified the efficiency of ECPA in solving PV parameter problems, there are still more aspects that deserve further exploration to explore a wider range of application scenarios. For example, ECPA can be used to extend to other solar cell datasets to expand its performance more. Second, to further improve the stability of ECPA in actual PV system applications, more environmental variables can be added for stability extension testing, such as wind speed, humidity, air pressure, and other variables. In the future, ECPA can also be extended to solve discrete optimization problems, multiobjective problems, or parametric evaluation of renewable energy sources such as geothermal energy, biomass energy, and so on.

**NOMENCLATURE**

- **CPA**: colony predation algorithm
- **DDM**: double diode model
- **ECPA**: enhanced colony predation algorithm
- **IAE**: absolute error
- **KC200GT**: multicrystal photovoltaic module
- **N**: the number of individual
- **NMs**: Nelder-Mead simplex algorithm
- **Photowatt-PWP 201**: photovoltaic module
- **PV**: photovoltaic
- **RE**: relative error
- **RMSE**: root mean square error
- **RTC France**: solar cell
- **SDM**: single diode model
- **SM55**: mono-crystalline PV module
- **ST40**: thin-film PV module
- **STC**: standard test conditions
- **TDM**: three diode model
- **I_{L}**: output current
- **I_{ph}**: photo-generated current
- **I_{d1}, I_{d2}, I_{d3}**: diode current
- **I_{sh}**: shunt resistor current
- **I_{sd1}, I_{sd2}, I_{sd3}**: reverse saturated current
- **R_s**: series resistor
- **n, n_1, n_2, n_3**: diode ideality factor
- **k**: Boltzmann constant
- **T**: Kelvin temperature
- **N_p**: number of parallel solar cells
- **N_s**: number of series solar cells
- **I_{SC}**: short circuit current
- **G**: irradiance
- **α**: temperature coefficient
- **I_{SC,STC}**: short circuit current at STC
- **G_{STC}**: irradiance at STC
- **T_{STC}**: Kelvin temperature at STC
- **S**: the power of the target individual
- **D_1**: the distance of the animal population transition
- **r_i**: random number
- **α**: reflection coefficient
- **β**: expansion coefficient
- **γ**: compression coefficient
- **δ**: Shrinkage coefficient
ub  upper boundary
lb  lower boundary
Max_FEs  maximum number of function evaluations
D  the dimension of the problem
Rsh  parallel resistor

ACKNOWLEDGMENTS
This study was supported in part by the National Natural Science Foundation of China (62076185 and U1809209), Wenzhou Science and Technology Bureau (ZG2020030), Humanities and Social Science Research Planning Fund Project of the Ministry of Education (20YJA790090). We acknowledge Fangjun Kuang for her contribution to the analysis of the data, and the comments of the reviewers.

ORCID
Ali Asghar Heidari  https://orcid.org/0000-0001-6938-9948
Huiling Chen  http://orcid.org/0000-0002-7714-9693

REFERENCES
1. Yousri D, Babu TS, Allam D, Ramachandaramurthy VK, Etiba MB. A novel chaotic flower pollination algorithm for global maximum power point tracking for photovoltaic system under partial shading conditions. *IEEE Access*. 2019;7:121432-121445.
2. Elattar EE, ElSayed SK. Modified JAYA algorithm for optimal power flow incorporating renewable energy sources considering the cost, emission, power loss and voltage profile improvement. *Energy*. 2019;178:598-609.
3. Bradaschia F, Cavalcanti MC, do Nascimento AJ, da Silva EA, de Souza Azevedo GM. Parameter identification for PV modules based on an environment-dependent double-diode model. *IEEE J Photovolt.* 2019;9(5):1388-1397.
4. Ebrahimim SM, Salahshour E, Malekzadeh M, Gordillo F. Parameters identification of PV solar cells and modules using flexible particle swarm optimization algorithm. *Energy*. 2019;179:358-372.
5. Alam D, Yousri D, Etieba M. Flower pollination algorithm based solar PV parameter estimation. *Energy Convers Manage*. 2015;101:410-422.
6. Chan D, Phillips J, Phang J. A comparative study of extraction methods for solar cell model parameters. *Solid-State Electron*. 1986;29(3):329-337.
7. Easwarakhanth T, Bottin J, Bouhouch I, Boutrit C. Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers. *Int J Solar Energy*. 1986;4(1):1-12.
8. Faris H, Mafarja MM, Heidari AA, et al. An efficient binary salt optimization algorithm with crossover scheme for feature selection problems. *Knowl-Based Syst*. 2018;154:43-67.
9. Luo J, Chen H, Heidari AA, Xu Y, Zhang Q, Li C. Multi-strategy boosted mutative whale-inspired optimization approaches. *Appl Math Model*. 2019;73:109-123.
10. Humada AM, Hojabri M, Mekhilef S, Hamada HM. Solar cell parameters extraction based on single and double-diode models: a review. *Renew Sustain Energy Rev*. 2016;56:494-509.
11. Zhang Q, Wang YX, Ma HT, Yuan J. Parameter estimation of photovoltaic models via cuckoo search. *J Appl Math*. 2013;2013:511-514.
12. Awadallah MA. Variations of the bacterial foraging algorithm for the extraction of PV module parameters from nameplate data. *Energy Convers Manage*. 2016;113:312-320.
13. Zhang Y, Jin Z, Mirjalili S. Generalized normal distribution optimization and its applications in parameter extraction of photovoltaic models. *Energy Convers Manage*. 2020;224:113301.
14. Yang B, Zhang X, Yu T, Shu H, Fang Z. Grouped grey wolf optimizer for maximum power point tracking of doubly-fed induction generator based wind turbine. *Energy Convers Manage*. 2017;133:427-443.
15. Bastidas-Rodriguez JD, Petrone G, Ramos-Paja CA, Spagnuolo G. A genetic algorithm for identifying the single diode model parameters of a photovoltaic panel. *Math Comput Simul*. 2017;131:38-54.
16. Askarzadeh A, Rezaazadeh A. Artificial bee swarm optimization algorithm for parameters identification of solar cell models. *Appl Energy*. 2013;102:943-949.
17. Nayak B, Mohapatra A, Mohanty KB. Parameter estimation of single diode PV module based on GWO algorithm. *Renew Energy Focus*. 2019;30:1-12.
18. Abbassi R, Abbassi A, Heidari AA, Mirjalili S. An efficient salp swarm-inspired algorithm for parameters identification of photovoltaic cell models. *Energy Convers Manage*. 2019;179:362-372.
19. Askarzadeh A, dos Santos Coelho L. Determination of photovoltaic modules parameters at different operating conditions using a novel bird mating optimizer approach. *Energy Convers Manage*. 2015;89:608-614.
20. Yang B, Wang J, Zhang X, et al. Comprehensive overview of meta-heuristic algorithm applications on PV cell parameter identification. *Energy Convers Manage*. 2020;208:112595.
21. Nesmachnow S. An overview of metaheuristics: accurate and efficient methods for optimisation. *Int J Metaheuristics*. 2014;3(4):320-347.
22. Fan C, Hu K, Feng S, Ye J, Fan E. Heronian mean operators of linguistic neutrosophic multisets and their multiple attribute decision-making methods. *Int J Distrib Sensor Netw*. 2019;15(4):1550147719843059.
23. Cui W-H, Ye J. Logarithmic similarity measure of dynamic neutrosophic cubic sets and its application in medical diagnosis. *Comput Indus*. 2019;111:198-206.
24. Fan C, Fan E, Hu K. New form of single valued neutrosophic uncertain linguistic variables aggregation operators for decision-making. *Cognit Syst Res*. 2018;52:1045-1055.
25. Ye J, Cui W. Modeling and stability analysis methods of neutrosophic transfer functions. *Soft Comput*. 2020;24(12):9039-9048.
26. Hu K, Ye J, En F, Shen S, Huang L, Jiadian P. A novel object tracking algorithm by fusing color and depth information based on single valued neutrosophic cross-entropy. *J Intell Fuzzy Syst*. 2017;32(3):1775-1786.
27. Hu K, He W, Ye J, Zhao L, Peng H, Pi J. Online visual tracking of weighted multiple instance learning via neutrosophic
similarity-based objectness estimation. *Symmetry*. 2019;11(6):832.

28. Liu X-F, Zhan Z-H, Zhang J. Resource-aware distributed differential evolution for training expensive neural-network-based controller in power electronic circuit. IEEE Transactions on Neural Networks and Learning Systems; 2021:1-11. doi:10.1109/TNNLS.2021.3075205

29. Zhan Z-H, Liu XF, Zhang H, et al. Cloudde: a heterogeneous differential evolution algorithm and its distributed cloud version. *IEEE Trans Parallel Distrib Syst*. 2016;28(3):704-716.

30. Yang Z, Li K, Guo Y, Ma H, Zheng M. Compact real-valued teaching-learning based optimization with the applications to neural network training. *Knowl-Based Syst*. 2018;159:51-62.

31. Deng W, Xu J, Zhao H, Song Y. A novel gate resource allocation method using improved PSO-based QEA. *IEEE Trans Intell Transport Syst*. 2020;23:1737-1745. doi:10.1109/TITS.2020.3025796

32. Deng W, Xu J, Song Y, Zhao M. An effective improved co-evolution ant colony optimization algorithm with multi-strategies and its application. *Int J Bio-Inspired Computat*. 2020;16(3):158-170.

33. Lai X, Zhou Y. Analysis of multiobjective evolutionary algorithms on the biobjective traveling salesman problem (1, 2). *Multimedia Tools and Applications*. 2020;79(41):30839-30860.

34. Li G, Li Y, Chen H, Deng W. Fractional-order controller for course-keeping of underactuated surface vessels based on frequency domain specification and improved particle swarm optimization algorithm. *Appl Sci*. 2022;12:3139.

35. Hu J, Chen H, Heidari AA, et al. Orthogonal learning covariance matrix for defects of grey wolf optimizer: insights, balance, diversity, and feature selection. *Knowl-Based Syst*. 2021;213:106684.

36. Hu J, Gui W, Heidari AA, et al. Dispersed foraging slime mould algorithm: continuous and binary variants for global optimization and wrapper-based feature selection. *Knowl-Based Syst*. 2022;237:107761.

37. Xia J, Wang Z, Yang D, et al. Performance optimization of support vector machine with oppositional grasshopper optimization for acute appendicitis diagnosis. *Comput Biol Med*. 2022;143:105206.

38. Xia J, Yang D, Zhou H, et al. Evolving kernel extreme learning machine for medical diagnosis via a disperase foraging sines cosine algorithm. *Comput Biol Med*. 2022;141:105137.

39. Wolpert DH, Macready WG. No free lunch theorems for optimization. *IEEE Trans Evolut Computat*. 1997;1(1):67-82.

40. Jordehi AR. Time varying acceleration coefficients particle swarm optimisation (TVACPSO): a new optimisation algorithm for estimating parameters of PV cells and modules. *Energy Convers Manage*. 2016;129:262-274.

41. Hasanien HM. Performance improvement of photovoltaic power systems using an optimal control strategy based on whale optimization algorithm. *Electric Power Syst Res*. 2018;157:168-176.

42. Qais MH, Hasanien HM, Alghuwainem S. Identification of electrical parameters for three-diode photovoltaic model using analytical and sunflower optimization algorithm. *Appl Energy*. 2019;250:109-117.

43. Ye M, Wang X, Xu Y. Parameter extraction of solar cells using particle swarm optimization. *J Appl Phys*. 2009;105(9):094502.

44. Chen X, Yu K. Hybridizing cuckoo search algorithm with biogeography-based optimization for estimating photovoltaic model parameters. *Sol Energy*. 2019;180:192-206.

45. Xu B, Heidari AA, Kuan F, Zhang S, Chen H, Cai Z. Performance optimization of photovoltaic systems: reassessment of political optimization with a quantum Nelder-Mead functionality. *Sol Energy*. 2022;234:39-63.

46. Song S, Wang P, Heidari AA, Zhao X, Chen H. Adaptive hawks optimization with persistent trigonometric differences for photovoltaic model parameter extraction. *Eng Appl Artificial Intell*. 2022;109:104608.

47. Yu S, Heidari AA, Liang G, Chen C, Chen H, Shao Q. Solar photovoltaic model parameter estimation based on orthogonally-adapted gradient-based optimization. *Optik*. 2022;252:168513.

48. Weng X, Heidari AA, Liang G, Chen H, Ma X. An evolutionary Nelder–Mead slime mould algorithm with random learning for efficient design of photovoltaic models. *Energy Reports*. 2021;7:8784-8804.

49. Fan Y, Wang P, Heidari AA, Chen H, Turabieh H, Mafarja M. Random reselection particle swarm optimization for optimal design of solar photovoltaic modules. *Energy*. 2022;239:121865.

50. Zhou W, Wang P, Heidari AA, et al. Metaphor-free dynamic spherical evolution for parameter estimation of photovoltaic modules. *Energy Reports*. 2021;7:5175-5202.

51. Yu S, Chen Z, Heidari AA, Zhou W, Chen H, Xiao L. Parameter identification of photovoltaic models using a sine cosine differential gradient based optimizer. *IET Renew Power Gen*. 2022;16:1535-1561.

52. Wang M, Zhang Q, Chen H, Heidari AA, Mafarja M, Turabieh H. Evaluation of constraint in photovoltaic cells using ensemble multi-strategy shuffled frog leading algorithms. *Energy Convers Manage*. 2021;244:114484.

53. Ahmadianfar I, Gong W, Heidari AA, Golilzar NA, Samadi-Kouchehsaraee A, Chen H. Gradient-based optimization with ranking mechanisms for parameter identification of photovoltaic systems. *Energy Reports*. 2021;7:3979-3997.

54. Liu Y, Heidari AA, Ye X, Liang G, Chen H, He C. Boosting slime mould algorithm for parameter identification of photovoltaic models. *Energy*. 2021;234:121164.

55. Weng X, Heidari AA, Liang G, et al. Laplacian Nelder-Mead spherical evolution for parameter estimation of photovoltaic models. *Energy Convers Manage*. 2021;243:114233.

56. Ridha HM, Hizam H, Gomes C, et al. Parameters extraction of three diode photovoltaic models using boosted LSHADE algorithm and Newton Raphson method. *Energy*. 2021;224:120136.
59. Fan Y, Wang P, Heidari AA, Zhao X, Turabieh H, Chen H. Delayed dynamic step shuffling frog-leaping algorithm for optimal design of photovoltaic models. *Energy Reports*. 2021;7:228-246.

60. Ridha HM, Gomes C, Hizam H, Ahmadiour F, Heidari AA, Chen H. Multi-objective optimization and multi-criteria decision-making methods for optimal design of standalone photovoltaic system: a comprehensive review. *Renew Sustain Energy Rev*. 2021;135:110202.

61. Wang M, Zhao X, Heidari AA, Chen H. Evaluation of constraint in photovoltaic models by exploiting an enhanced ant lion optimizer. *Sol Energy*. 2020;211:503-521.

62. Zhang H, Heidari AA, Wang M, Zhang L, Chen H, Li C. Orthogonal Nelder-Mead moth flame method for parameters identification of photovoltaic modules. *Energy Convers Manage*. 2020;211:112764.

63. Abbassi A, Abbassi R, Heidari AA, et al. Parameters identification of photovoltaic cell models using enhanced exploratory salp chains-based approach. *Energy*. 2020;198:117333.

64. Chen H, Jiao S, Heidari AA, Wang M, Chen X, Zhao X. An opposition-based sine cosin approach with local search for parameter estimation of photovoltaic models. *Energy Convers Manage*. 2019;195:927-942.

65. Tu J, Chen H, Wang M, Gandomi AH. The colony predation algorithm. *J Bionic Eng*. 2021;18(3):674-710.

66. Chen X, Yu K, Du W, Zhao W, Liu G. Parameters identification of solar cell models using generalized oppositional teaching learning based optimization. *Energy*. 2016;99: 170-180.

67. Xu S, Wang Y. Parameter estimation of photovoltaic modules using a hybrid flower pollination algorithm. *Energy Convers Manage*. 2017;144:53-68.

68. Yu K, Liang JJ, Qu BY, Chen X, Wang H. Parameters identification of photovoltaic models using an improved JAYA optimization algorithm. *Energy Convers Manage*. 2017;150:742-753.

69. Yu K, Liang JJ, Qu BY, Cheng Z, Wang H. Multiple learning backtracking search algorithm for estimating parameters of photovoltaic models. *Appl Energy*. 2018;226:408-422.

70. Zhou W, Wang P, Heidari AA, Zhao X, Turabieh H, Chen H. Random learning gradient based optimization for efficient design of photovoltaic models. *Energy Convers Manage*. 2021;230:113751.

71. Jiao S, Chong G, Huang C, et al. Orthogonally adapted Harris hawks optimization for parameter estimation of photovoltaic models. *Energy*. 2020;203:117804.

72. Li S, Chen H, Wang M, Heidari AA, Mirjalili S. Slime mould algorithm: a new method for stochastic optimization. *Future Gener Comput Syst*. 2020;111:300-323.

73. Ahmadifar I, Bozorg-Haddad O, Chu X. Gradient-based optimizer: a new metaheuristic optimization algorithm. *Inf Sci*. 2020;540:131-159.

74. Ahmadifar I, Heidari AA, Noshadian S, Chen H, Gandomi AH. INFO: an efficient optimization algorithm based on weighted mean of vectors. *Expert Syst Appl*. 2022;195:116516.

75. Xu W, Qu S, Zhao L, Zhang H. An improved adaptive sliding mode observer for middle-and high-speed rotor tracking. *IEEE Trans Power Electron*. 2020;36(1):1043-1053.

76. Zeng H-B, Liu X-G, Wang W. A generalized free-matrix-based integral inequality for stability analysis of time-varying delay systems. *Appl Math Comput*. 2019;354:1-8.

77. Zeng H-B, Liu XG, Wang W, Xiao SP. New results on stability analysis of systems with time-varying delays using a generalized free-matrix-based inequality. *J Franklin Inst*. 2019;356(13):7312-7321.

78. Moayedi H, Rezaei A. An artificial neural network approach for under-reamed piles subjected to uplift forces in dry sand. *Neural Comput Appl*. 2019;31(2):327-336.

79. Moayedi H, Hayati S. Applicability of a CPT-based neural network solution in predicting load-settlement responses of bored pile. *Int J Geomech*. 2018;18(6):06018009.

80. Ahmadifar I, Heidari AA, Gandomi AH, Chu X, Chen H. RUN beyond the metaphor: an efficient optimization algorithm based on Runge Kutta method. *Expert Syst Appl*. 2021;181:115079.

81. Heidari AA, Mirjalili S, Faris H, Aljaraah I, Mafarja M, Chen H. Harris hawks optimization: algorithm and applications. *Future Generation Comput Syst Int J Sci*. 2019;97:849-872.

82. Yang Y, Chen H, Heidari AA, Gandomi AH. Hunger games search: visions, conception, implementation, deep analysis, perspectives, and towards performance shifts. *Expert Syst Appl*. 2021;177:114864.

83. Tu J, Chen H, Liu J, et al. Evolutionary biogeography-based whale optimization methods with communication structure: towards measuring the balance. *Knowl-Based Syst*. 2021;212:106642.

84. Song S, Wang P, Heidari AA, et al. Dimension decided harris hawks optimization with Gaussian mutation: balance analysis and diversity patterns. *Knowledge-Based Systems*. 2021;215:106425.

85. Fu J, Zhang Y, Wang Y, et al. Optimization of metabolomic data processing using NOREVA. *Nat Protoc*. 2022;17(1):129-151.

86. Li B, Tang J, Yang Q, et al. NOREVA: normalization and evaluation of MS-based metabolomics data. *Nucleic Acids Res*. 2017;45(W1):W162-W170.

87. Yang Q, Wang Y, Zhang Y, et al. NOREVA: enhanced normalization and evaluation of time-course and multi-class metabolomic data. *Nucleic Acids Res*. 2020;48(W1):W436-W448.

88. Cai Z, Gu J, Luo J, et al. Evolving an optimal kernel extreme learning machine by using an enhanced grey wolf optimization strategy. *Expert Syst Appl*. 2019;138:112814.

89. Wei Y, Lv H, Chen M, et al. Predicting entrepreneurial intention of students: an extreme learning machine with Gaussian barebone Harris hawks optimizer. *IEEE Access*. 2020;8:76841-76855.

90. Yu C, Chen M, Cheng K, et al. SGGA: annealing-behaved grasshopper optimizer for global tasks. *Engineering with Computers*. 2021;1-28.

91. Zhang Y, Liu R, Heidari AA, et al. Towards augmented kernel extreme learning models for bankruptcy prediction: algorithmic behavior and comprehensive analysis. *Neurocomputing*. 2021;430:185-212.

92. Persons MH, Walker SE, Rypstra AL, Marshall SD. Wolf spider predator avoidance tactics and survival in the presence of diet-associated predator cues (araneae: lycosidae). *Anim Behav*. 2001;61(1):43-51.
93. Khater M, Murariu D, Gras R. Predation risk tradeoffs in prey: effects on energy and behaviour. *Theoretical Ecology*. 2016;9(3):251-268.

94. Hussien AG, Heidari AA, Ye X, Liang G, Chen H, Pan Z. Boosting whale optimization with evolution strategy and Gaussian random walks: an image segmentation method. *Eng Comput*. 2022.

95. Su H, Zhao D, Yu F, et al. Horizontal and vertical search artificial bee colony for image segmentation of COVID-19 X-ray images. *Comput Biol Med*. 2022;142:105181.

96. Chen X, Huang H, Heidari AA, et al. An efficient multilevel thresholding image segmentation method based on the slime mould algorithm with bee foraging mechanism: a real case with lupus nephritis images. *Comput Biol Med*. 2022;142:105179.

97. Yu D, Wang X, Liang P, Sun X. Apple leaf disease recognition method with improved residual network. *Multimedia Tools Appl*. 2022;81:27923-27948.

98. Yi J-H, Deb S, Dong J, Alavi AH, Wang GG. An improved NSGA-III algorithm with adaptive mutation operator for big data optimization problems. *Future Generation Computer Sys*. 2018;88:571-585.

99. Han X, Han Y, Chen Q, et al. Distributed flow shop scheduling with sequence-dependent setup times using an improved iterated greedy algorithm. *Complex System Model Simul*. 2021;1(3):198-217.

100. Gao D, Wang G-G, Pedrycz W. Solving fuzzy job-shop scheduling problem using DE algorithm improved by a selection mechanism. *IEEE Trans Fuzzy Systems*. 2020;28(12):3265-3275.

101. Liu P, Gao H. A novel Green supplier selection method based on the interval type-2 fuzzy prioritized choquet Bonferroni means. *IEEE/CAA Journal of Automatica Sinica*. 2020;8:1549-1566. doi:10.1109/JAS.2020.1003444

102. Dong R, Chen H, Heidari AA, Turabieh H, Mafarja M, Wang S. Boosted kernel search: framework, analysis and case studies on the economic emission dispatch problem. *Knowl-Based Syst*. 2021;233:107529.

103. Zhang H, Liu T, Ye X, et al. Differential evolution-assisted salp swarm algorithm with chaotic structure for real-world problems. *Eng Comput*. 2022;1:35.

104. Shan W, Qiao Z, Heidari AA, Chen H, Turabieh H, Teng Y. Double adaptive weights for stabilization of moth flame optimizer: balance analysis, engineering cases, and medical diagnosis. *Knowl-Based Syst*. 2021;214:106728.

105. Zhao F, Di S, Cao J, Tang J, Jonrinaldi. A novel cooperative multi-stage hyper-heuristic for combination optimization problems. *Complex Syst Model Simul*. 2021;1(2):91-108.

106. Boettcher S, Percus AG. Extremal optimization: an evolutionary local-search algorithm. *Computational Modeling and Problem Solving in the Networked World*. Springer; 2003:61-77.

107. Nelder JA, Mead R. A simplex method for function minimization. *Comput J*. 1965;7(4):308-313.

108. Wu Z, Li G, Shen S, Lian X, Chen E, Xu G. Constructing dummy query sequences to protect location privacy and query privacy in location-based services. *World Wide Web*. 2021;24(1):25-49.

109. Wu Z, Wang R, Li Q, et al. A location privacy-preserving system based on query range cover-up for location-based services. *IEEE Trans Veh Technol*. 2020;69:5244-5254.

110. Wu Z, Li R, Zhou Z, Guo J, Jiang J, Su X. A user sensitive subject protection approach for book search service. *J Assoc Inform Sci Technol*. 2020;71(2):183-195.

111. Wu Z, Shen S, Lian X, Su X, Chen E. A dummy-based user privacy protection approach for text information retrieval. *Knowl-Based Syst*. 2020;195:105679.

112. Wu Z, Shen S, Zhou H, Li H, Lu C, Zou D. An effective approach for the protection of user commodity viewing privacy in e-commerce website. *Knowl-Based Syst*. 2021;220:106952.

113. Derrac J, Garcia S, Molina D, Herrera F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evol Comput*. 2011;1(1):3-18.

114. Abd Elaziz M, Oliva D, Xiong S. An improved opposition-based sine cosine algorithm for global optimization. *Expert Syst Appl*. 2017;90:484-500.

115. Nenavath H, Jatoth RK. Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking. *Appl Soft Comput*. 2018;62:1019-1043.

116. Niu M, Hu Y, Sun S, Liu Y. A novel hybrid decomposition-ensemble model based on VMD and HGWO for container throughput forecasting. *Appl Math Model*. 2018;57:163-178.

117. Adarsh BR, Raghunathan T, Jayabarathi T, Yang XS. Economic dispatch using chaotic bat algorithm. *Energy*. 2016;96:666-675.

118. Sun K, Dou Z, Zhu Y, et al. Scheduling model of power system based on forecasting error of wind power plant output. *IEEE Trans Electrical Electronic Eng*. 2021;16(4):526-535.

119. Liang H, Liu Y, Shen Y, Li F, Man Y. A hybrid bat algorithm for economic dispatch with random wind power. *IEEE Trans Power Syst*. 2018;33(5):5052-5061.

120. Heidari AA, Aljarah I, Faris H, Chen H, Jie L, Mirjalili S. An enhanced associative learning-based exploratory whale optimizer for global optimization. *Neural Comput Appl*. 2019;32:1-27.

121. Su HP, Rickert K, Burlein C, et al. A novel multi-scale cooperative mutation fruit fly optimization algorithm. *Knowl-Based Syst*. 2016;114:24-35.

122. Liu W, Zhang Z, Fan J, Jiang D, Daemen JMJ. Research on the stability and treatments of natural gas storage caverns with different shapes in bedded salt rocks. *IEEE Access*. 2020;8:18995-19007.

123. Liu W, Zhang X, Fan J, Li Y, Wang L. Evaluation of potential for salt cavern gas storage and integration of brine extraction: cavern utilization, Yangtze River Delta region. *Nat Resour Res*. 2020;29(5):3275-3290.

124. Jinlong L, Wenjie X, Jianjing Z, Wei L, Xilin S, Chunhe Y. Modeling the mining of energy storage salt caverns using a structural dynamic mesh. *Energy*. 2020;193:116730.

125. Gao W, Wu H, Siddiqui MK, Baig AQ. Study of biological networks using graph theory. *Saud J Biol Sci*. 2018;25(6):1212-1219.

126. Gao W, L.G. Guirao J, Basavanagoud B, Wu J. Partial multi-dividing ontology learning algorithm. *Inf Sci*. 2018;467:35-58.

127. Gao W, Wang W, Dimitrov D, Wang Y. Nano properties analysis via fourth multiplicative ABC indicator calculating. *Arabian J Chem*. 2018;11(6):793-801.
128. Su Y, Li S, Zheng C, Zhang X. A heuristic algorithm for identifying molecular signatures in cancer. *IEEE Trans Nanobiosci*. 2019;19(1):132-141.

129. Li L, Gao Z, Wang YT, et al. SCMFMDA: predicting microRNA-disease associations based on similarity constrained matrix factorization. *PLoS Comput Biol*. 2021;17(7):e1009165.

130. Wu Q-W, Cao RF, Xia J, Ni JC, Zheng CH, Su Y. Extra trees method for predicting LncRNA-Disease association based on multi-layer graph embedding aggregation. *IEEE/ACM Transactions of Computational Biology and Bioinformatics*; 2021. doi:10.1109/TCBB.2021.3113122

131. Abd Elaziz M, Oliva D. Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm. *Energy Convers Manage*. 2018;171:1843-1859.

132. Li YH, Li XX, Hong JJ, et al. Clinical trials, progression-speed differentiating features and swiftness rule of the innovative targets of first-in-class drugs. *Brief Bioinform*. 2020;21(2):649-662.

133. Zhu F, Li XX, Yang SY, Chen YZ. Clinical success of drug targets prospectively predicted by in silico study. *Trends Pharmacol Sci*. 2018;39(3):229-231.

134. Su Y, Liu C, Niu Y, Cheng F, Zhang X. A community structure enhancement-based community detection algorithm for complex networks. *IEEE Trans Syst Man Cybernet Syst*. 2019;51(5):2833-2846.

135. Tian Y, Su X, Su Y, Zhang X. EMODMI: A multi-objective optimization based method to identify disease modules. *IEEE Trans Emerging Topics Comput Intell*. 2020;5(4):570-582.

136. Yin J, Sun W, Li F, et al. VARIDT 1.0: variability of drug transporter database. *Nucleic Acids Res*. 2020;48(D1):D1042-D1050.

137. Zhu F, Shi Z, Qin C, et al. Therapeutic target database update 2012: a resource for facilitating target-oriented drug discovery. *Nucleic Acids Res*. 2012;40(D1):D1128-D1136.

138. Cui H, Guan Y, Chen H. Rolling element fault diagnosis based on VMD and sensitivity MCKD. *IEEE Access*. 2021;9:120297-120308.

139. Deng W, Li Z, Li X, Chen H, Zhao H. Compound fault diagnosis using optimized MCKD and sparse representation for rolling bearings. *IEEE Trans Instrum Meas*. 2022;71:1-9. doi:10.1109/TIM.2022.3159005

140. Pei H, Yang B, Liu J, Chang KCC. Active surveillance via group sparse Bayesian learning. *IEEE Trans Pattern Anal Mach Intell*. 2020;44:1133-1148. doi:10.1109/TPAMI.2020.3023092

141. Ran X, Zhou X, Lei M, TePas W, Deng W. A novel k-means clustering algorithm with a noise algorithm for capturing urban hotspots. *Appl Sci*. 2021;11(23):11202.

142. Qiu S, Zhao H, Jiang N, et al. Sensor network oriented human motion capture via wearable intelligent system. *Int J Intell Syst*. 2021;37:1646-1673. doi:10.1002/int.22689

143. Qiu S, Hao Z, Wang Z, et al. Sensor combination selection strategy for kayak cycle phase segmentation based on body sensor networks. *IEEE Internet Things J*. 2021;9:4190-4201. doi:10.1109/JIOT.2021.3102856

144. Yu K, Liang J, Qu B, Luo Y, Yue C. Dynamic selection preference-assisted constrained multiobjective differential evolution. *IEEE Trans Syst Man Cybernet Syst*. 2021;52:2954-2965. doi:10.1109/TSMC.2021.3061698

145. Yu K, Liang J, Qu B, Yue C. Purpose-directed two-phase multiobjective differential evolution for constrained multiobjective optimization. *Swarm Evolut Computat*. 2021;60:100799.

**How to cite this article:** Xu B, Heidari AA, Zhang S, Chen H, Shao Q. Extremal Nelder–Mead colony predation algorithm for parameter estimation of solar photovoltaic models. *Energy Sci Eng*. 2022;10:4176-4219. doi:10.1002/ese3.1273