Neural Computing and Applications (2024) 36:8757–8773
https://doi.org/10.1007/s00521-024-09534-z

ORIGINAL ARTICLE

Estimation of solar cell parameters through utilization of adaptive sine–cosine particle swarm optimization algorithm

Mohamed Issa[1,3] · Ahmed M. Helmi[2,3] · Mohamed Ghetas[4]

Received: 3 July 2023 / Accepted: 22 January 2024 / Published online: 2 March 2024

Abstract
Due to the growing demand for clean and sustainable energy sources, there has been an increasing interest in solar cells and photovoltaic panels. Nevertheless, determining the right design parameters to achieve the most efficient energy output that aligns with the energy system's needs can be quite challenging. This complexity arises from the intricate models and the inherent inaccuracies in the available information. To tackle this challenge, this paper introduces the adaptive sine–cosine particle swarm optimization algorithm (ASCA-PSO) as a method for estimating the parameters of solar cells and photovoltaic modules. The ASCA-PSO approach combines the strengths of the SCA and PSO algorithms in a two-tier process. In this process, SCA search agents explore the search space, while the PSO search agents leverage the outcomes derived from SCA exploration. This study evaluates the effectiveness of ASCA-PSO in accurately estimating the parameters of single- and double-diode models using data from two commercial solar cells. The findings are compared with those of cutting-edge methods. It is demonstrated that ASCA-PSO can identify global solutions for multifaceted and intricate objective functions. Furthermore, it proves to be a viable option for designing solar cells even in the presence of noise.

Keywords Solar cells · Photovoltaic modules · Parameter estimation · Sine–cosine algorithm · Particle swarm optimization

1 Introduction

Based on recent research, it is projected that by 2050, 85% of the global population will live in urban areas, creating a significant need for specialized services to be available around the clock. [1]. Approximately 75% of the world's energy production is consumed by cities, and they are responsible for generating 80% of greenhouse gas emissions [1, 2]. To address the issue of pollution, it is necessary to develop new environmentally friendly energy sources [2]. In recent years, there has been a growing global concern for environmental protection policies that aim to promote the development of clean fuel technology [3]. Solar energy is one of the most lucrative renewable sources being explored today, as it can help meet the increasing demand for energy supplies [4]. Solar energy is recognized for its quiet operation, ease of deployment, and lack of pollutants [5].

Photovoltaic modules (PVs), consisting of solar cells (SCs), can convert solar energy into electricity without the need for any additional intervention. This means that both SC and PV modules must be capable of functioning in conditions that are influenced by weather [6], in addition, the maintenance of SC and PV modules should be cost-effective [7]. Meeting these objectives requires rigorous design of both SC and PV modules. Before a PV system is installed, its efficiency must be optimized, and its capacity must be evaluated using a reliable and effective simulator [8].

The design of SCs involves the use of mathematical models to estimate the parameters that define the cell. These models are utilized to simulate the internal variables

✉ Mohamed Issa
Mamohamedali@eng.zu.edu.eg

1 Faculty of Computer Science and Information Technology, Egypt-Japan University of Science and Technology, New Borg El Arab, Egypt

2 Department of Computer Engineering, College of Engineering and Information Technology, Buraydah Private Colleges, 51418 Buraydah, Saudi Arabia

3 Department of Computer and Systems Engineering, Faculty of Engineering, Zagazig University, Zagazig 44519, Egypt

4 Faculty of Computer Science and Engineering, Galala University, Suez, Egypt

that govern the behavior of the SC and establish the relationship between current and voltage (I–V). Two models are commonly employed for this task: the single-diode (SD) model and the double-diode (DD) model. In [9], the essential concepts of single- and double-diode modeling of PV solar cells are described.

The single-diode (SD) and double-diode (DD) models are electronic circuits used to depict the nonlinear behavior of SC. In the SD and DD models, various factors are considered, including the photo-generated current, diode saturation current, series resistance, and diode ideality factor. The configuration of these elements precisely reflects the performance of the SC or PV modules. The SD model employs five parameters to describe the SC, while the DD model uses seven. To achieve a precise balance between current and voltage (I–V) and obtain current estimates that closely match the measured values, it is essential to accurately estimate these parameters.

Several approaches have been proposed in recent years to optimize the estimation of SC model parameters [9, 10]. The methods for estimating SC model parameters can be categorized into three types: numerical techniques, analytical techniques, and soft-computing techniques. Numerical techniques employ nonlinear optimization approaches, such as the Newton–Raphson technique [11], the Levenberg–Marquardt algorithm [12], and the conductivity method for estimating the parameters of SC [13]. One of the main disadvantages of numerical techniques is their sensitivity to the initialization of parameters. This can result in becoming trapped in local minima, as the nature of the SC parameter estimation problem is multimodal [14].

Several attempts have been made to estimate parameters using analytical methods, such as employing elementary functions [15] and Lambert W-function [16], and there is a review of the common methods used in [9] that shows the advantages and drawbacks of each method. The main issue with analytical methods is that they require more approximations due to the large number of parameters to be estimated (five for a single diode and seven for a double diode), which can lead to increased computation time for solving the system [9]. That is clear in [17] when the number of parameters was reduced from seven to four to be solved analytically. Another drawback of analytical methods is that they require additional coefficients, and their values may not be readily available in the datasheets [9]. As a result, analytical methods may provide less accurate parameter estimates for PV solar cells due to the approximations required and the lack of available data [18].

Soft computing is a third approach that aims to overcome the disadvantages of numerical and analytical methods. Soft-computing techniques typically utilize meta-heuristic algorithms (MAs), which search for optimal global solutions based on a search strategy that imitates natural behavior. Meta-heuristic algorithms employ different metaphors, such as the genetic algorithm (GA), to achieve this goal [19] and differential evolution (DE) [20], which is based on the evolutionary theory. Meanwhile, physics-based algorithms include methods such as the sine–cosine algorithm (SCA) [21] and the gravitational search algorithm (GSA) [22]. There also exists another group of methods based on animals and insects like particle swarm optimization (PSO) [23], artificial bee colony (ABC) [24, 25], or moth-flame optimization (MFO) [26]. MA succeeded in optimizing many applications such as the optimization of radiative transfer function [27], induction motor design [28], Handwritten Arabic Manuscript Image Binarization [29], feature selection [30], PID controller tuning parameters [31], and multiple sequence alignment [32].

Meta-heuristic algorithms can explore complex and multimodal search spaces using various operators to find the optimal solution. In the context of estimating SC parameters using MA, the root-mean-square error (RMSE) is commonly used as an objective function. The RMSE compares the values obtained from the dataset with the parameters calculated by the diode models. There are several different types of MA discussed in the literature. For example, in [33], the GA is used to increase the accuracy of the parameters estimated by the DD. The PSO is applied to estimate the parameters of solar cells using SD and DD models [34–38] and in another development of PSO using chaos theory to increase exploration [39]. Besides, PSO was used for solar fabrication with the aid of neural networks [40]. PSO also succeeded in estimating the parameters of solar cells but in a more complex model of the circuit where it modeled as three diodes [41].

Another interesting approach uses simulated annealing (SA) to compute the values of the SD and DD [42], according to the authors, the results obtained using simulated annealing (SA) were superior to those obtained using other approaches that were compared. More recently, a new method called cat swarm optimization (CSO) has been proposed for determining the optimal parameters of SC using both the SD and DD models [43]. Moreover, the CSO has also been compared with different methods to verify the quality of the solutions. In Rajasekar [44], the bacterial foraging algorithm (BFA) is introduced as an alternative to accurately model the characteristics of an SC using a new equation proposed by the authors. In the same context, in Askarzadeh and Rezazadeh [45], various versions of the harmony search (HS) algorithm have been proposed for identifying unknown parameters in both single- and double-diode models of solar cells. While these methods are generally efficient, they may still suffer from accuracy issues. In real-world scenarios, such as PV or SC model identification, it is crucial to have accurate outputs to minimize the costs associated with energy systems [46].

Other related works are the pattern search (PS) [47] and the bird-mating optimizer [48].

In Das et al. [49], a hybrid scheme between PSO and DE [20]. This technique involves integrating particle swarm optimization (PSO) with differential evolution (DE) for the optimization of digital filter circuits. The primary advantage of this integration is that it enhances the exploration scheme of PSO, thereby reducing the likelihood of becoming trapped in local optima. This is achieved by adding a new term in the estimation of particle velocity, which is the percentage difference in distances between two random particles. This promotes greater exploration of the particles. Additionally, particles are moved to new positions if they provide better fitness than their previous positions. However, the main drawback of this technique is its slow convergence, and it may provide low accuracy for optimizing mathematical benchmark functions.

The no-free-lunch (NFL) theorem states that there is no single optimization technique that is universally suitable for all optimization problems. In other words, different optimization techniques may perform better or worse depending on the specific problem being addressed. It emphasizes the importance of selecting an appropriate optimization technique based on the characteristics of the problem at hand [50]. Given the NFL theorem, this paper proposes the use of a hybrid optimization algorithm that combines the best features of two different methods. Specifically, it introduces the use of a recently proposed method called the sine–cosine algorithm, which is known for its simplicity and efficiency. By combining the sine–cosine algorithm with other optimization techniques, the proposed hybrid algorithm aims to achieve better performance than either method alone [21] with the PSO [23]. This method is called an adaptive sine–cosine optimization algorithm integrated with particle swarm optimization (ASCA-PSO), and it has two optimization layers [51]. The proposed hybrid optimization algorithm consists of two layers. The first layer uses particle swarm optimization (PSO) to exploit the most prominent regions of the search space. Meanwhile, the second layer employs the sine–cosine algorithm (SCA) to explore different sections of the same space using its unique operators. By combining these two methods, the proposed ASCA-PSO algorithm aims to achieve a better balance between exploration and exploitation of the search space, thereby improving the overall performance compared to either method alone. The ASCA-PSO algorithm has demonstrated its capabilities for solving complex optimization problems such as local sequence alignment.

The primary advantage of ASCA-PSO is its ability to perform both exploitation and exploration processes in parallel, which speeds up the convergence of the best solution and improves the quality of the solutions by combining the benefits of SCA for exploring the search space with the accurate tuning provided by PSO for exploitation. This makes it an ideal algorithm for enhancing the estimation of parameters for PV solar cells, which typically involve numerous parameters that must be accurately tuned. By efficiently exploring the search space, ASCA-PSO can find optimal solutions in multidimensional search spaces while achieving a high degree of convergence.

ASCA-PSO is executed in two layers, with SCA in the bottom layer and PSO in the upper layer, which helps to increase the accuracy of the search process. This is achieved through the combination of operators and the co-evolutionary learning scheme, which guides the algorithm toward optimal solutions. However, due to the nonlinearity of the parameter estimation problem for solar cells, SCA may produce poor results when used alone, in comparison to PSO. This is because PSO can exploit the search space to find the best solution, while SCA explores the search space. The objective function used in ASCA-PSO is the root-mean-square error (RMSE), which measures the differences between a dataset and the values estimated using the optimal diode models (SD and DD).

Thus, this paper's primary contributions are outlined as follows: firstly, introducing a reliable and precise tool for identifying PV solar cell parameters through a hybrid method of PSO and SCA. Secondly, it showcases the ability of ASCA-PSO to simultaneously conduct exploration and exploitation processes, thereby improving the accuracy of SC parameters. Thirdly, applying ASCO-PSO to estimate parameters for two SC models, namely the SD and DD circuit models. Finally, the paper compares the performance of ASCA-PSO with other algorithms used for estimating PV solar cell parameters.

The organization of this paper considers the following sections: Sect. 2 presents the diode models used for SC and how the parameter estimation can be formulated as an optimization problem. In Sect. 3, the preliminaries of ASCA-PSO are presented. Section 4 describes the experimental methodology and presents the results. Meanwhile, Sect. 5 includes some conclusions and future work.

2 Formulation of PV parameter estimation problem

In photovoltaic (PV) systems, diodes are used to manage the flow of electrical current. There are two common types of diodes used in PV systems: single diodes and double diodes. A single diode, also known as a bypass diode or blocking diode, is a basic diode component used in PV modules. Its primary function is to prevent reverse current flow through a specific cell or group of cells in a PV module when they are shaded or operating under low-light conditions. When a solar cell is shaded, it can act as a resistor and impede the flow of current, potentially reducing the overall performance of the module. The single diode is connected in parallel to the shaded cell or group of cells. When the voltage across the shaded cell(s) becomes negative (due to shading), the diode becomes forward biased and allows the current to flow through it, by passing the shaded cells.

A double-diode configuration is a more advanced setup used in some high-efficiency PV modules. It includes both a forward-biased diode (like the single diode) and a reverse-biased diode. The forward-biased diode functions as described for the single diode, allowing current to bypass shaded cells. However, the additional reverse-biased diode offers extra protection and performance benefits. The reverse-biased diode is connected in series with the PV cell or string of cells and helps reduce losses caused by voltage drop and thermal effects. It prevents current from flowing in the reverse direction, which can improve the overall efficiency and reliability of the PV module.

The choice between single- and double-diode configurations depends on the specific design and requirements of the PV system. Double-diode configurations are typically found in high-end, high-efficiency solar modules, while single diodes are more common in standard PV modules. The use of diodes in PV systems is crucial to maximizing energy harvest and protecting the cells from damage under various operating conditions.

In this section, the basic concepts of the general two solar cell models, the single diode (SD) and the double diode (DD) are discussed.

2.1 Single-diode circuit model

The model depicted in Fig. 1 employs a single diode to shunt the photogenerated current source, with the diode serving as the rectifier in the circuit. Typically, the SD model requires the estimation of five parameters, as the configuration has a significant impact on the model's output.

In general, the SC current (I_t) is calculated using the following equation:

$$I_{\mathrm{t}} = I_{\mathrm{ph}} - I_{\mathrm{d}} - I_{\mathrm{sh}} \tag{1}$$

where I_{sh}, I_{t}, I_{d}, and I_{ph} are the shunt resistor current, the terminal, the diode, and the photogenerated, respectively. If the internal parameters of the diode are adjusted based on the equivalent Shockley diode equation to achieve high-performance output, Eq. (1) can be expressed as:

$$I_{\mathrm{t}} = I_{\mathrm{ph}} - I_{\mathrm{sd}}\left[\exp\left(\frac{q(V_{\mathrm{t}} + R_{\mathrm{s}}.I_{\mathrm{t}})}{n.k.T}\right) - 1\right] - \frac{V_{\mathrm{t}} + R_{\mathrm{s}}.I_{\mathrm{t}}}{R_{\mathrm{sh}}} \tag{2}$$

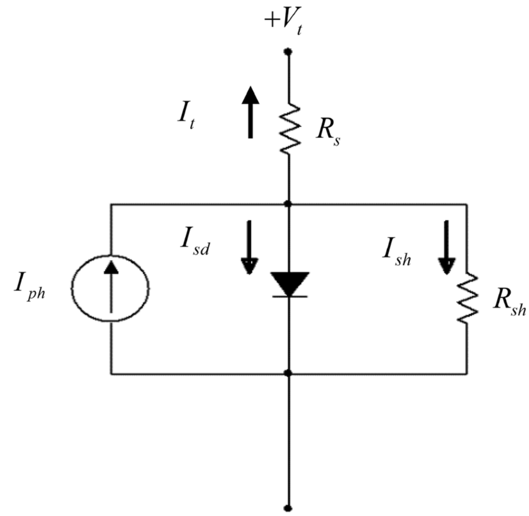

Fig. 1 The equivalent circuit of the SD model

where V_{t}, I_{sd}, R_{sh}, and R_{s} represent the terminal voltage, the diode saturation currents, the shunt, and the series resistances, respectively. The variable n is the non-physical ideality factor. Also, $q = 1.602 \times 10^{-19}$ C (coulombs) represents the magnitude of the charge on an electron. Meanwhile, $k = 1.380 \times 10^{-23}$ $^{\mathrm{J}}\!/_{\mathrm{K}}$ is the Boltzmann constant, and T is the cell temperature in Kelvin (K).

2.2 Double-diode circuit model

This section presents a description of the DD model, which is represented by a rectifier employing one diode. The second diode is utilized to design the recombination current and other non-idealities of the SC. Figure 2 illustrates the DD model. Based on Eq. (1) can be rewritten as follows:

$$I_{\mathrm{t}} = I_{\mathrm{ph}} - I_{\mathrm{d1}} - I_{\mathrm{d2}} - I_{\mathrm{sh}} \tag{3}$$

where I_{d1}, I_{d2}, and are the currents of the first and second diode, respectively. The Shockley equivalence is used to update the internal configuration of the diodes given in Eq. (3) to become the following:

$$\begin{aligned} I_{\mathrm{t}} = I_{\mathrm{ph}} - I_{\mathrm{sd1}}\left[\exp\left(\frac{q(V_{\mathrm{t}} + R_{\mathrm{s}}.I_{\mathrm{t}})}{n_1.k.T}\right) - 1\right] \\ - I_{\mathrm{sd2}}\left[\exp\left(\frac{q(V_{\mathrm{t}} + R_{\mathrm{s}}.I_{\mathrm{t}})}{n_2.k.T}\right) - 1\right] - \frac{V_t + R_s.I_t}{R_{sh}}. \end{aligned} \tag{4}$$

where I_{sd1}, and I_{sd2} represent the diffusion and saturation current for the d_1 and d_2 diodes, respectively. n_1 and n_2 are the diffusion and recombination diode ideality factors, respectively. From Eq. (4), the DD circuit contains seven undefined parameters (i.e.,R_{s}, R_{sh}, I_{ph}, I_{sd1}, I_{sd2}, $n_{1,}$ and n_2) needed to be estimated as their parameters.

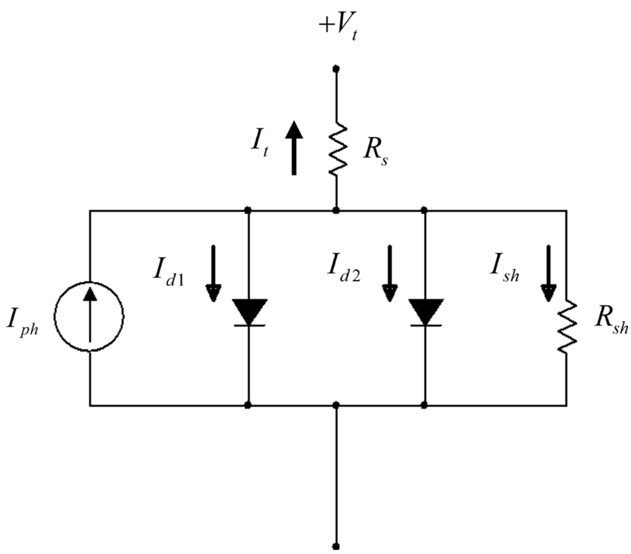

Fig. 2 The equivalent circuit of the SD model

Both single- and double-diode configurations in photovoltaic (PV) systems have their advantages, but they also come with limitations.

SD's limitations such as:

- Limited shading mitigation: Single diodes are effective at mitigating shading issues for small-scale shading scenarios. However, in cases of complex or partial shading, where multiple cells or strings are affected, single diodes may not be as effective in maximizing power output.
- Voltage drop: Single diodes can introduce some voltage drop when they are being conducted. This voltage drop can lead to energy losses, especially in systems where minimizing losses is crucial.
- Temperature sensitivity: Single diodes can be sensitive to temperature variations. The forward voltage drop across a diode decreases as temperature increases, which can impact its effectiveness in different environmental conditions.

DD's limitations such as:

- Complexity: Double-diode configurations are more complex and can be costlier to implement compared to single diodes. They require additional components and wiring.
- Added resistance: The presence of an additional diode in the circuit can introduce additional electrical resistance, potentially leading to minor energy losses.
- Niche application: Double-diode configurations are typically used in high-efficiency or advanced PV modules, making them less common in standard PV installations. They may not be necessary for all applications.
- Maintenance and reliability: With more components comes an increased potential for maintenance issues and reliability concerns. Double diode configurations require proper design and quality control to ensure they function as intended over the long term.

3 Adaptive sine–cosine and particle swarm optimization algorithm

This section is divided into two parts, the first subsection introduces the basics of particle swarm optimization (PSO) and the sine–cosine algorithm (SCA). The second part explains all the steps of the adaptive sine–cosine and particle swarm optimization algorithm (ASCA-PSO).

3.1 Preliminaries

3.1.1 Particle swarm optimization

The particle swarm optimization (PSO) [26] mimics the behavior of birds flocking. It is a search strategy based on global communication between the particles (search agents) where the particles adapt their movements toward the particle that finds the best solution. Its movement is adapted according to Eqs. (5) and (6) toward the particles P^{gbest} and P_i^{best} which represents the best global position between all particles and the best local position that particle I passed during the previous iterations.

$$v_i(t+1) = w * v_i(t) + c_1 \mathrm{rand}\left(P^{\mathrm{best}}{}_i - P_i(t)\right) + c_2 \mathrm{rand}\left(P^{\mathrm{gbest}} - P_i(t)\right) \quad (5)$$

$$P_i(t+1) = P_i(t) + v_i(t+1) \quad (6)$$

where v_i is the velocity of the ith particle, P_i is the position of particle i, t is the iteration number, and *rand* is a uniformly distributed random variable in the range [0–1]. c_1 and c_2 are the best local and global positions weight coefficients in order. w is the inertia coefficient that controls the effect of the previous velocity on the new velocity. P_i^{best} is the best local position (solution) found by particle I, and P^{gbest} is the best global solution found by all particles.

3.1.2 Sine–cosine algorithm

The sine–cosine algorithm (SCA) [23] is an optimization algorithm that uses sine and cosine operators to adapt the movements of search agents to explore the search space for the best solution. The movements of particles are controlled toward the best solution found according to Eq. (7).

$$P_i^{t+1} = \left\{ \begin{array}{l} P_i^t + r_1 \sin(r_2) \left| r_3 P^{\text{gbest}} - P_i^t \right| r_4 < 0.5 \\ P_i^t + r_1 \cos(r_2) \left| r_3 P^{\text{gbest}} - P_i^t \right| r_4 \geq 0.5 \end{array} \right\} \quad (7)$$

where P_i denotes the position of search agent i and P^{gbest} is the best global solution between all search agents. r_1 determines how far the next solution is from the current one and determines the exploration scale through the search space. r_2 determines the direction of the next movement toward or outward the best solution, and r_3 controls the effect of destination (P^{gbest}) on current movement. r_4 is used to balance the usage of sine and cosine functions as in Eq. (7). Values of r_1, r_2, r_3, and r_4 are modified during each iteration to increase the diversity of solutions. Equation (8) is used for balancing exploitation and exploration by updating the value of r_1 according to the iteration number where t is the current iteration, T is the maximum number of iterations, and a is a constant that should be set by the coder.

$$r_1 = a\left(1 - \frac{t}{T}\right) \quad (8)$$

3.2 The ASCA-PSO

The adaptive sine–cosine and particle swarm optimization algorithm (ASCA-PSO) [51] enhanced the convergence and the quality of solutions produced by the standard SCA. The two algorithms are merged in two layers as in Fig. 3, the bottom layer has groups of search agents that update its movements based on SCA represented as (x_{ij}), where i = 1, 2, 3, …, M, and j = 1, 2, 3, …,N. While, i and j represent the index of solutions in the top and bottom layer, respectively. The top layer consists of search agents controlled by PSO and each agent represents the global solution found by the agents in the corresponding bottom layer. Each search agent of the top layer is represented by (y^i). ($_{\text{best}}$) represents the best solution found among the particles of PSO in the top layer. This classification of hybridization of meta-heuristics belongs to the high-level and Co-evolutionary hybrid meta-heuristics [43].

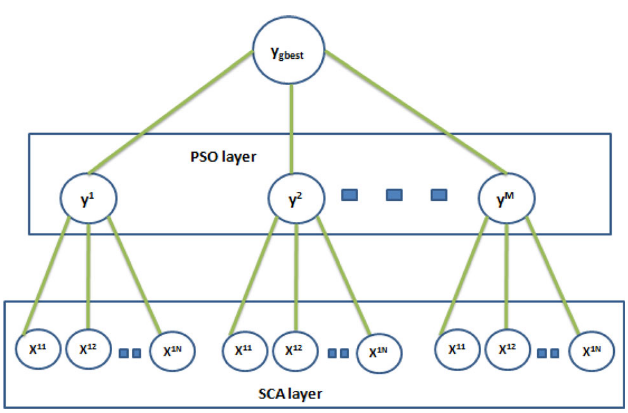

Fig. 3 The two-layer structure of ASCA-PSO

SCA updates the movements of the search agents toward the best solution found by (y_i) according to Eq. (9). The search agents of the top layer update their movements based on PSO toward the best solution ($y^{\text{pbest}}{}_i$) from all search agents of the top and bottom layers. The movements are updated using Eqs. (10) and (11) where ($y^{\text{pbest}}{}_i$) represents the best solution that y^i of particle I have over all previous iterations. y_{gbest} is the best global solution between whole search agents in the top and bottom layers.

$$x_{ij}^{t+1} = \left\{ \begin{array}{l} x_{ij}^t + r_1 \sin(r_2) \left| r_3 y_i^t - x_{ij}^t \right| r_4 < 0.5 \\ x_{ij}^t + r_1 \cos(r_2) \left| r_3 y_i^t - x_{ij}^t \right| r_4 \geq 0.5 \end{array} \right. \quad (8)$$

$$v_i^{t+1} = w * v_i^t + c_1 \text{rand}\left(y_i^{\text{pbest}} - y_i^t\right) + c_2 \text{rand}\left(y^{\text{gbest}} - y_i^t\right) \quad (9)$$

$$y_i^{t+1} = y_i^t + v_i^t \quad (10)$$

Each search agent of the bottom layer (x_{ij}) is influenced by the best solution of the group in the top layer (y^i). Moreover, each search agent of the top layer (y^i) is also influenced by the best solution found (y_{gbest}) between the whole set of search agents which increases the diversity of solutions found in the search space. In addition, performing exploration besides exploitation in the same iteration raises the chance of finding the global optimum solution with a higher convergence speed than SCA while avoiding being trapped in local minima. Figure 4 shows the flowchart of the proposed ASCA-PSO algorithm.

4 Parameter estimation of PV cell using ASCA-PSO

This section presents the implementation of the ASCA-PSO for parameter estimation of PV cells. The algorithm and problem could be adapted to accurately estimate the best solutions. The optimization problem is defined *as minimizing the* $RMSE(X)$ that depends on the variables of each diode model. In the ASCA-PSO, the population X contains the candidate solutions defined as $X = [x_1, x_2, ..., x_N]$ and each element is constructed as $x_i = \left[x_{i,1}, x_{i,2}, ..., x_{i,d}\right]$ where $d \in [5, 7]$. The variable d corresponds to the dimension of the problem and depends on the parameters to estimate using the diode models for that reason its value could be five or seven. The set of solutions is randomly initialized and then evaluated in the objective functions defined by the root-mean-square error (RMSE). The ASCA-PSO then starts the iterative search process. Here is important to mention that to compute the RMSE first is necessary to test the set of parameters (candidate solution) on the model of the solar cells to compute the output current of the circuit. Algorithm 1 describes the details for computing the parameters of a diode model.

Algorithm 1 Parameter estimation of PV cell using ASCA-PSO algorithm

1: Initialize N_{sca} search agents (number of search agents in each group in the bottom layer)
2: Initialize N_{pso} search agents (number of search agents in top layer)
3: Initialize the global best solution
5: Evaluate RMSE for each agent in the bottom layer and assign it as the best solution ($RMSE^{gbest}$) if it finds better RMSE then update the value of the head of this group toward $RMSE^{gbest}$ based on updating equation of PSO algorithm.
6: Update the search agents in each group in the bottom layer using the updating equation of the SCA algorithm toward the best solution of the group (head of the group in the top layer).
7: During step 6, compute RMSE for each agent if it finds better RMSE then assign it a a solution to $RMSE^{gbest}$ and update the value of the head of this group toward $RMSE^{gbest}$ used on the updating equation of PSO algorithm.
8: Update the positions of the search agents in each group in the top layer using the update equation of the PSO algorithm toward the global best solution $RMSE^{gbest}$, if finds a better RMSE then assign it as a solution to $RMSE^{gbest}$
9: Repeat from 6 to 8 for T iterations
10: Output the parameters of the PV cell represented by the global best solution that achieves the best RMSE found $RMSE^{gbest}$

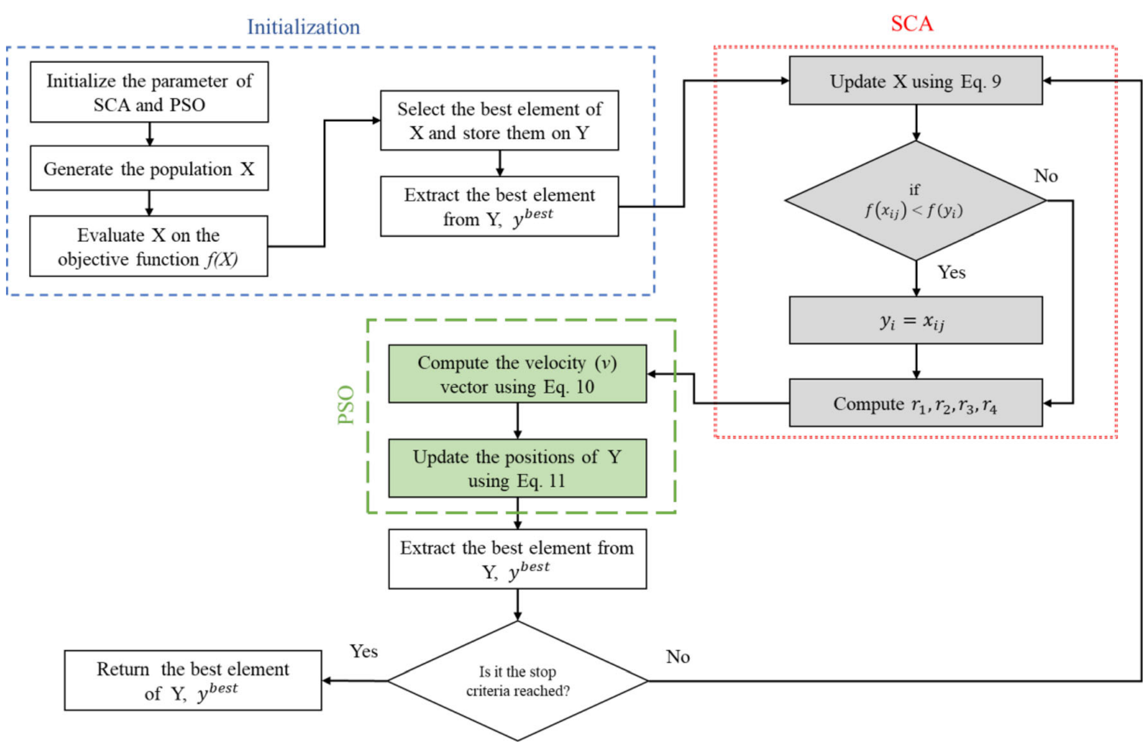

Fig. 4 Flowchart of ASCA-PSO

Table 1 The used ranges of PV cell parameters in the SD model [39]

Parameter	Lower bound	Upper bound
I_{ph} (A)	0	1
I_{sd} (A)	0	1
n	1	2
R_s (Ω)	0	0.5
R_p (Ω)	0	100

5 Experimental results

This section discusses the verification of the performance and efficiency of the ASCA-PSO approach for estimating the parameters of the PV solar cell design. The evaluation criteria were used for testing the performance of the optimization technique as follows:

Table 4 The ranges of PV cell parameters in the DD circuit model [39]

Parameter	Lower bound	Upper bound
I_{ph} (A)	0	1
I_{sd1} (A)	0	1
I_{sd2} (A)	0	1
n_1	1	2
n_2	1	2
R_s (Ω)	0	0.5
R_{sh} (Ω)	0	100

- *Statistical mean:* is the average of solutions (S_i) that are produced by executing the optimization algorithm for M times and is calculated according to Eq. (12).

Table 2 Statistics of the RMSE values, achieved by different optimization algorithms for SD using the R.T.C module

	ASCA-PSO	PSO	SCA	GA	BSA	PS
Max	0.000995	0.129	0.0104	0.01516	0.00278	0.00205
Min	0.000987	0.0029	0.00272	0.004102	0.00144	0.00205
Mean	0.000989	0.082	0.00540	0.009877	0.00581	0.00205
Std	0.0000146	0.029	0.00206	0.002719	0.00972	0

	Newton	GOTLBO	LETLBO	TLABC	GBABC	PCE
Max	0.0097	0.00198	0.00112	1.039×10^{-3}	0.001284	0.0009860
Min	0.0097	0.00984	0.00098	9.860×10^{-4}	0.000988	0.0009860
Mean	0.0097	0.001334	0.00101	9.985×10^{-4}	0.001044	0.0009860
Std	0	0.000299	0.00031	1.860×10^{-5}	0.000070	3.05×10^{-12}

Table 3 Circuit model parameters for the SD model are achieved by different optimization algorithms

	ASCA-PSO	PSO	SCA	GA	BSA	PS	GOTLBO
I_{ph} (A)	0.766	0.708	0.767	0.766535	0.761	0.761	0.7607
I_{sd} (A)	3.07×10^{-7}	2.14×10^{-7}	2.31×10^{-7}	7.45×10^{-7}	4.79×10^{-7}	9.80×10^{-7}	0.331
n	1.58	1.51	1.50	1.570175	1.52	1.60	1.483
R_s (Ω)	0.035	0.0363	0.037	0.031438	0.034	0.031	0.036
R_{sh} (Ω)	50	49.5	19.3	29.482993	79.59	100.0	54.11
RMSE	0.000989	0.082	0.00540	0.00410	0.00144	0.00295	0.00134

	PCE	DE	Newton	HS	CSO	GGHS	TLABC
I_{ph} (A)	0.760776	0.7608	0.7608	0.7607	0.7607	0.7609	0.7607
I_{sd} (A)	0.323021	3.23×10^{-7}	3.22×10^{-7}	3.04×10^{-7}	3.23×10^{-7}	3.26×10^{-7}	0.3230
n	1.481074	1.4806	1.4837	1.4753	1.481	1.482	1.4811
R_s (Ω)	0.036377	0.0364	0.0364	0.0366	0.036	0.0363	0.0363
R_{sh} (Ω)	53.718525	53.71	53.7634	53.59	53.71	53.06	53.716
RMSE	0.000982	0.0234	0.00970	0.000994	0.000986	0.000949	0.000985

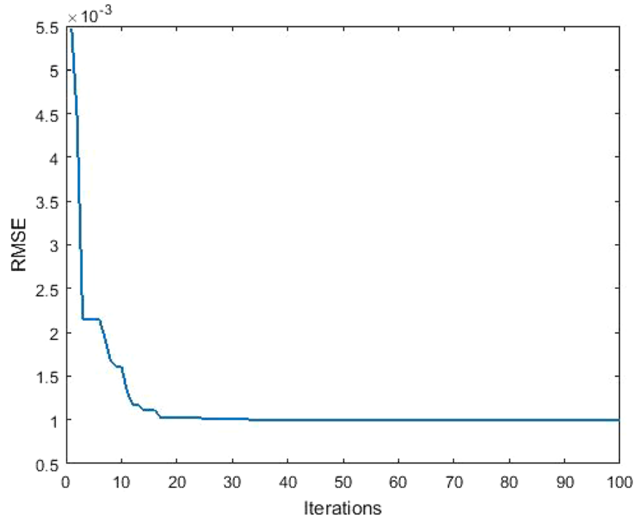

Fig. 5 The convergence curve of ASCA-PSO for SD model of R.T.C France module

$$\text{Mean} = \frac{1}{\text{M}}\sum_{\text{i}=1}^{\text{M}} S_i \quad (11)$$

where S_i is the obtained solution of the run time i.

- *Statistical standard deviation (Std):* is an indicator of the variation of the best fitness values found for running the optimization algorithm for M run times. Also, it represents robustness and stability. It is computed as in Eq. (13):

$$\text{Std} = \sqrt{\frac{1}{M-1}\sum_{i=1}^{M}(S_i - \text{Mean})^2} \quad (12)$$

- *Root-mean-square error (RMSE):* it represents a standard deviation of the difference between measured (I_m) and estimated ($I_\mathbf{c}$) values. It is computed using the following formula:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N}(I_\text{m} - I_\text{c})^2}{N}} \quad (13)$$

The RMSE is used as an objective function that verifies if the model with the parameters estimated can properly perform the output of the SC.

- *Absolute error (E_{abs})*: It is the absolute difference between the measured values (I_m) and estimated values (I_c) as in Eq. (15).

$$E_\text{abs} = |I_\text{m} - I_\text{c}| \quad (14)$$

The E_abs are used to measure how close the I_c is from the ideal (measured) current. In this implementation, it helps to verify if the current value computed by the ASCA-PSO is better than the measured current.

Two commercial solar cell modules are used for verification which are the R.T.C module and the STM6-40/36 module with 36 mono-crystalline cells. In the experimental testing of ASCA-PSO in comparison with standard SCA and PSO. The following parameters are used which are chosen within the allowed range specified by the authors of algorithms and by several runs each parameter is tuned individually to deliver the best results:

- The number of search agents was 1600 and the number of iterations was set to 100.
- For PSO, c_1 and c_2 had a value of 2 and 0.2 for w.
- For SCA, a and r_3 had a value of 2 and 1, respectively.
- For ASCA-PSO, c_1 and c_2 had a value of 0.5. w has a value of 0.2. a and r_3 have values of 10 and 2 respectively.
- For SCA and ASCA-PSO r_2 and r_4 were randomized each iteration and r_1 was updated according to Eq. (8).

Table 5 Circuit model parameters for the DD circuit model achieved by different optimization algorithms

	ASCA-PSO	PSO	SCA	BSA	PS	PCE
Max	0.0017	0.5598	0.0444	0.00286	0.00816	0.001025
Min	0.000177	0.0087	0.0011	0.0011	0.00816	0.000982
Mean	0.000997	0.1985	0.0094	0.00466	0.00816	0.000986
Std	0.0012	0.12	0.0123	0.000835	0	5.99×10^{-7}

	GA	GOTLBO	LETLBO	TLABC	GBABC
Max	0.0144	0.001787	0.00157	0.001504	0.001272
Min	0.00592	0.000983	0.000985	0.000984	0.0009907
Mean	0.00861	0.001243	0.001083	0.001055	0.001052
Std	0.0018	0.000209	0.000126	0.000155	7.00×10^{-5}

Table 6 Circuit model parameters for the DD circuit model are achieved by different optimization algorithms

	ASCA-PSO	PSO	SCA	BSA	PS	GOTLBO	PCE
I_{ph} (A)	0.761	0.88148	0.761545	0.767	0.763	0.7607	0.760781
I_{sd1} (A)	1.03×10^{-6}	4.45×10^{-7}	3.1×10^{-7}	4×10^{-7}	2.8×10^{-7}	0.8001	0.226015
$I_{sd2}(A)$	9.87×10^{-8}	7.78×10^{-8}	5.2×10^{-8}	1×10^{-12}	1×10^{-12}	0.2204	0.749340
n_1	1.838	1.878855	1.833986	1.47	1.00	1.9999	1.450923
n_2	1.388	1.340615	1.320963	2	1.00	1.4489	2
R_s (Ω)	0.037	0.047286	0.044317	0.0353	0.0586	0.0367	0.03674
R_p (Ω)	55.93	58.64958	58.20994	54.45	18.21	56.075	55.483160
RMSE	0.000997	0.1985	0.00940	0.0112	0.00820	0.001243	0.000986

	GA	SA	HS	GGHS	CSO	TLABC
I_{ph} (A)	0.768	0.7623	0.7617	0.7605	0.7607	0.7608
I_{sd1} (A)	6.60×10^{-7}	4.767×10^{-7}	1.2×10^{-7}	3×10^{-7}	2.2×10^{-7}	0.4239
$I_{sd2}(A)$	4.57×10^{-7}	0.100×10^{-7}	2.5×10^{-7}	1×10^{-7}	7.27×10^{-7}	0.2401
n_1	1.60	1.517	1.494	1.496	1.451	1.9075
n_2	1.62	2.00	1.499	1.929	1.997	1.4567
R_s (Ω)	0.0291	0.0345	0.0354	0.0356	0.0367	0.0366
R_{sh} (Ω)	51.11	43.10	46.82	62.78	55.38	54.667
RMSE	0.00591	0.0166	0.00126	0.00107	0.000982	0.001243

All experimental tests were performed using Matlab software on a machine with a 64-bit Intel Dual Core processor with 2.0 GHz and 4GB RAM.

The performance of the proposed algorithm is evaluated by analyzing the complexity and measuring the processing time. The complexity of the ASCA-PSO algorithm is $O(T \times M \times N \times (C_{SCA} + C_{PSO}))$ where M and N are the size of search agents in the top and bottom layer in order and C_{SCA} and C_{PSO} is the time cost of updating all one search agent per one iteration for SCA and PSO in order, and T is the number of iterations. The time complexity of SCA and PSO respectively is O($T \times N \times$ C_{SCA}) and O($T \times N \times$ C_{PSO}). According to the previous setting parameters values the program coded in ASCA-PSO, SCA, and PSO for estimating the parameters is run 30 times. The average time for estimating the parameters consumes 16.4 s by ASCA-PSO, while SCA consumes 9.3 s and PSO executes 10.5 s.

5.1 Experimental series 1: parameter estimation of solar cells for RTC France silicon solar cell at 33 °C and full irradiation (1000 w/m^2)

5.1.1 Single-diode (SD) circuit model

This subsection shows the estimation of the parameters of the SD circuit model of R.T.C (with 26 data samples of experimental measurement of currents versus voltage) using ASCA-PSO and compared with other related work such as backtracking search algorithm (BSA) [52], gravitational search algorithm (GSA) [53], PS, Newton algorithm [47], SA [42], DE [18], HS and Grouping-based Global Harmony Search (GGHS) [45], CSO [43], GOTLBO [54], Teaching–learning-based optimization with learning experience of other learners (LETLBO) [54], Teaching–Learning–Based Artificial Bee Colony (TLABC) [55], Population Classification Evolution Algorithm (PCE) [56] and Gaussian Bare-bones Artificial Bee Colony (GBABC) [57].

The range of five decision parameters of the SD circuit model for the R.T.C module is listed in Table 1 [39].

In Table 2, the estimated current (I_c) using the ASCA-PSO method is shown in comparison with PSO and SCA besides the absolute error between the estimated current using optimization techniques and the measured current experimentally. In this case, the value of I_c is computed by using Eq. (2), the value of the measured voltage (V_m), and the parameters estimated by the ASCA-PSO. The absolute error (E_{abs}) is obtained by the difference between I_c and the measured current (I_m). ASCA-PSO technique produces an absolute error smaller than that produced using SCA and PSO. This reflects the efficiency of ASCA-PSO for finding the values of parameters that produce current approximated to the measured current experimentally better than SCA or PSO separately.

From Tables 3 and 4, it is observed that the ASCA-PSO outperforms all the algorithms by delivering a mean RMSE that is less than that delivered using most of the related work. ASCA-PSO delivers RMSE near that delivered using HS, CSO, and GGHS. In addition, a minimum standard

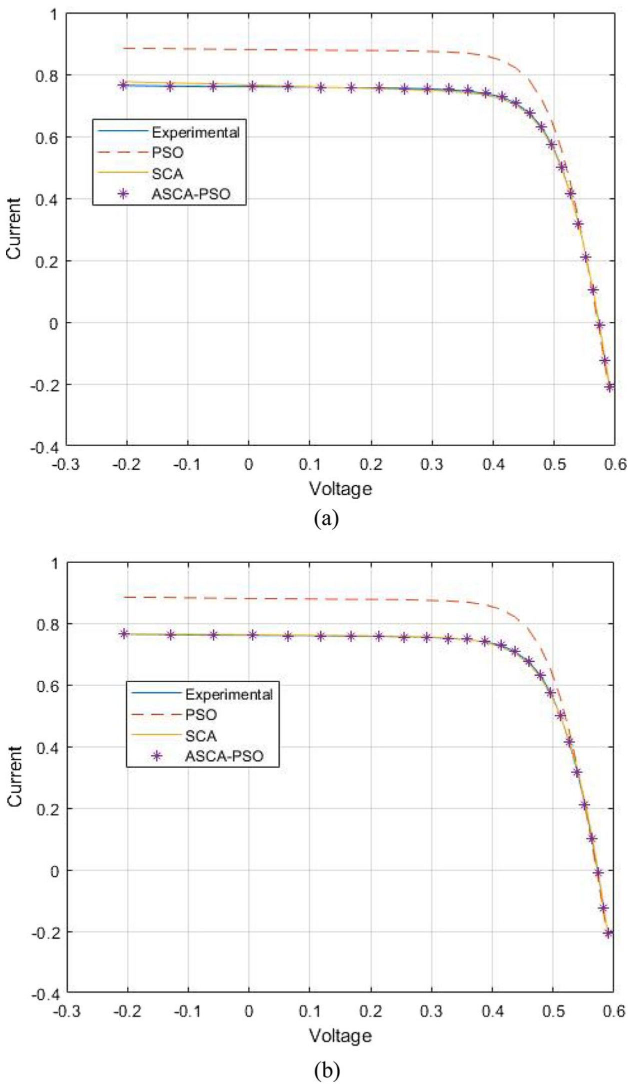

Fig. 6 Measured voltage versus current computed by using ASCA-PSO, SCA, and PSO for **a** SD **b** DD R.T.C models

deviation reflects the stability and robustness of the ASCA-PSO technique. Moreover, Fig. 5 indicates that the ASCA-PSO method converges after a few iterations for reaching the optimum RMSE which reflects the quick convergence rate performance of ASCA-PSO. Hence, the statistical test for the SD circuit model of the R.T.C module shows the efficiency of ASCA-PSO for delivering accurate results in terms of quality of solution and convergence speed with efficient robustness.

In Fig. 5, the convergence curve of the ASCA-PSO is plotted along the iterative process for searching the best parameters of the SD model. Here, the data from the R.T.C France module are used. From this figure, it can be seen that the algorithm converges in approximately less than 30 iterations after small adjustments. This is the behavior of the combination of SCA and PSO. Such behavior can be interpreted as follows. During the first iterations, the algorithm performs exploration and after it reaches an optimal solution, the exploitation starts working trying to reach more precise solutions.

From Table 2, it can be observed that the lower the value the mean RMSE is reached by the ASCA-PSO. The second-best algorithm is the GA followed in third place by the TLABC. The worst algorithm according to the meaning of the RMSE is the PSO. In the same context, the standard deviation of ASCAPSO is lower in comparison to similar approaches.

Table 3 presents the parameters estimated for the SD model using different methods. From such results, it can be analyzed the differences between the elements that permit obtaining the total current of the SC.

5.1.2 Double-diode (DD) circuit model

In this section, the DD circuit model of the R.T.C PV solar cell was used to ensure the verification of the performance of the ASCA-PSO. The estimated parameters are seven which increase the complexity of the problem. The range of values of parameters are as listed in Table 4 [39].

From Table 5, it is indicated that the ASCA-PSO method keeps its robustness with minimum standard deviation despite the increased number of parameters. Table 7 presents the estimated parameters using ASCA-PSO computed at best RMSE better than most of the related work except CSO which produces similar RMSE approximately. The power of ASCA-PSO appears with an increasing number of parameters for the DD model where it preserves the same efficiency as in the SD model where algorithms like HS and GGHS perform better for the SD model circuit.

An analysis of the mean value of the RMSE from Table 6 indicates that for the ASCA-PSO such value is lower, followed by the GBABC and the TLABC. Moreover, the Std provided evidence of the stability of the ASCA-PSO during the iterations.

From Table 6, evidence of the quality of the parameter is given. From such results, it is possible to see that the best solution is obtained by the ASCA-PSO. Here are also presented the differences between all the parameters used in the DD model. On the other hand, Fig. 6 shows a plotted graph of estimated current using the ASCA-PSO technique versus measured voltages in comparison with PSO, SCA, and measured current experimentally for SD and DD circuit models. From the figure, it is concluded that ASCA-PSO estimates current approximately like measured current better than SCA. However, PSO produces the worst results which reflect the ability of SCA to raise the performance of PSO by hybridization.

The reason behind the multimodal nature of PV model estimation is the large range of values from each

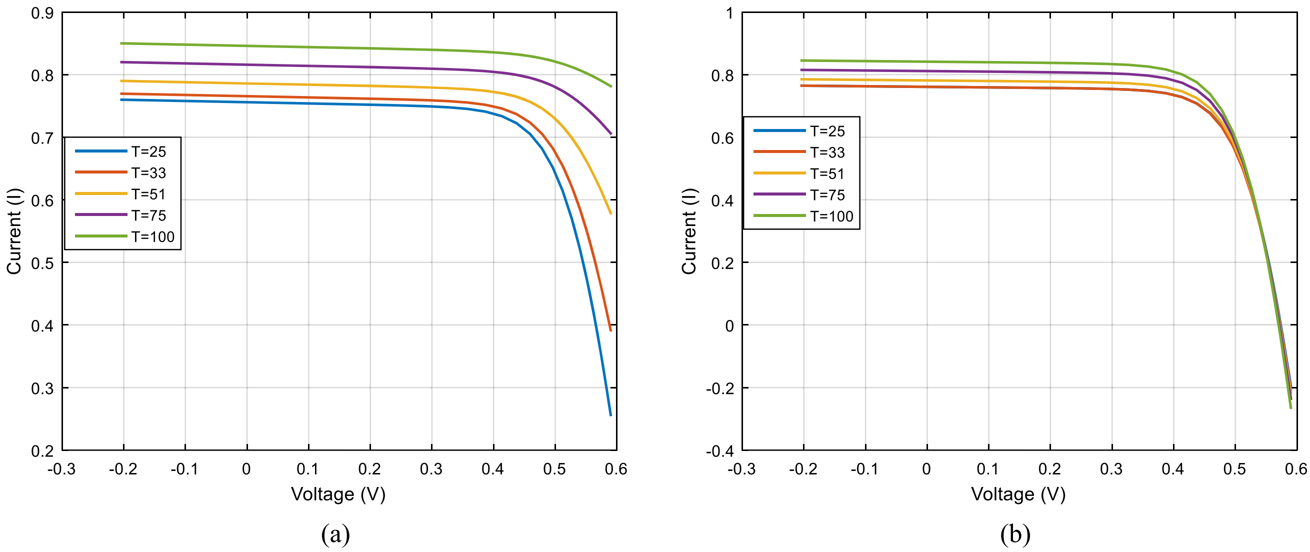

Fig. 7 Measured voltage versus current computed by using ASCA-PSO for different temperatures for **a** SD and **b** DD models

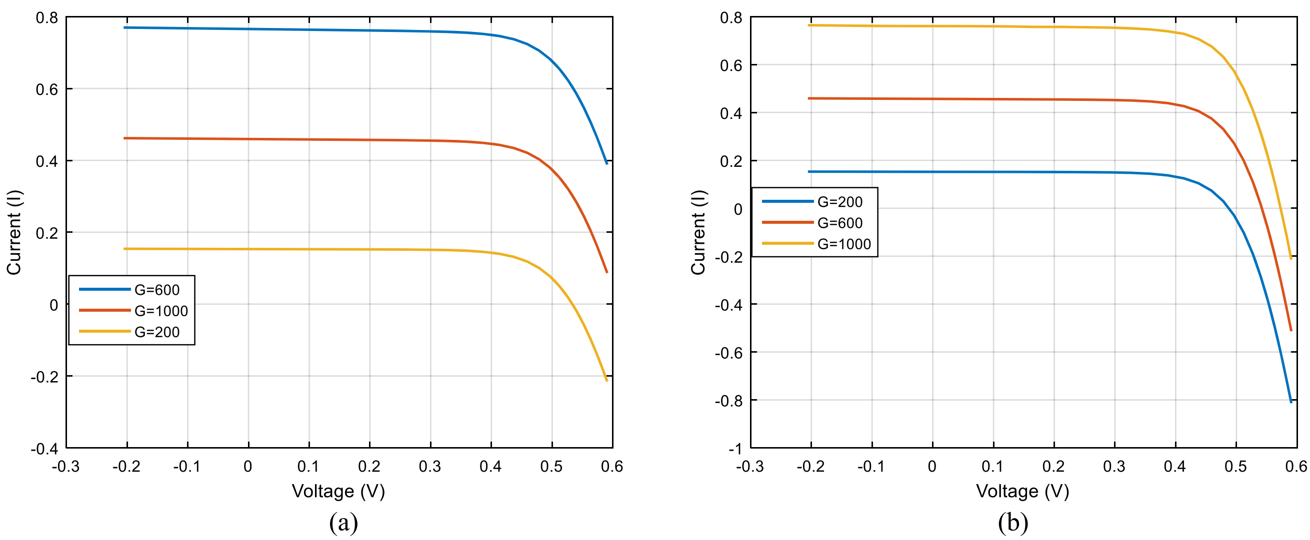

Fig. 8 Measured voltage versus current computed by using ASCA-PSO for different irradiation for **a** SD and **b** DD models

Table 7 The used ranges of PV cell parameters in the SD model for the STM6-40/36 module [59]

Parameter	Lower bound	Upper bound
I_{ph} (A)	1	2
I_{sd} (A)	0	10^{-6}
N	0	100
R_s (Ω)	0	1
R_p (Ω)	300	800

Table 8 Statistics of the RMSE values, achieved by different optimization algorithms

	ASCA-PSO	PSO	SCA	BSA	ABC
Max	0.0024	0.0484	0.0457	0.0094	0.0035
Min	0.0020	0.0138	0.0081	0.0036	0.0023
Mean	0.0023	0.0295	0.0241	0.0060	0.0035
Std	0.000057	0.0107	0.0104	0.0014	0.00077

For the SD circuit model of STM6-40/36 model

parameter. In addition, several parameters increase the dimensionality of the search space. Thus, using algorithms with exploitation advantages is better for improving the quality of the solution. Hence, the meta-heuristic algorithms that have a good balance between exploration and exploitation generate better results. Hence, performing

Table 9 Circuit model parameters for the SD circuit model of STM6-40/36 module estimated by ASCA-PSO in comparison with other works

	ASCA-PSO	PSO	SCA	BSA	ABC
I_{ph} (A)	1.668	1.64	1.74	1.65	1.67
$I_{sd}(A)$	4.00×10^{-7}	1.51×10^{-7}	2.52×10^{-7}	6.33×10^{-7}	4.65×10^{-7}
n	55.42	52.82	54.51	51.76	50.47
R_s (Ω)	0.521	0.28	0.86	0.52	0.5
R_{sh} (Ω)	400.54	200.94	100.52	723.39	495.52
RMSE	0.0023	0.0295	0.0241	0.0061	0.0023

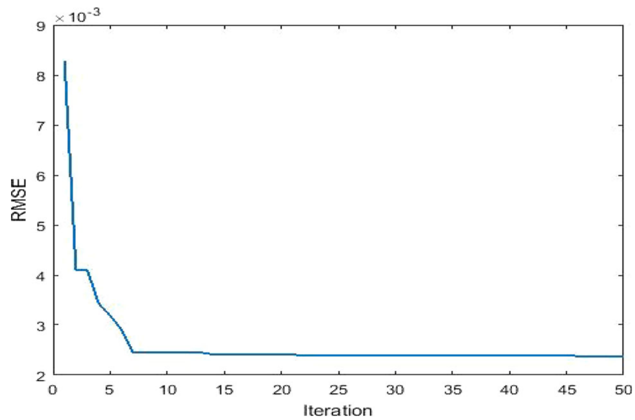

Fig. 9 The convergence curve of ASCA-PSO for the SD circuit model of STM6-40/36 module

exploration (SCA) side with exploitation (PSO) like ASCA-PSO will present results better than using each one separately. Also, the power-voltage analysis is presented in Fig. 7 for SD and DD models.

Figures 7 and 8 present the temperature and irradiation analysis effects respectively at different operating conditions. In these figures, it is graphically shown the differences between the PSO, SCA, and ASCA-PSO. The proposed approach then can find the configuration of parameters that better fits the data provided. Figure 8 presents a comparative study that concerns irradiation. Here, the algorithm proposed maintains its performance under different values.

5.2 Experimental series 2: results using STM6-40/36 module at 51 °C and irradiation of 1000 W/m^2

In this experimental series, the STM6-40/36 (36 monocrystalline photovoltaic cells aligned in series) was used to ensure testing of the performance and efficiency of the ASCA-PSO approach. The experimental data (measured voltage versus current) has been extracted at full radiation and $T = 51$ °C. The tests were performed on the single- and double-diode circuits model in the following sections. The ASCA-PSO is compared with SCA, PSO, and other related work such as BSA and ABC [58].

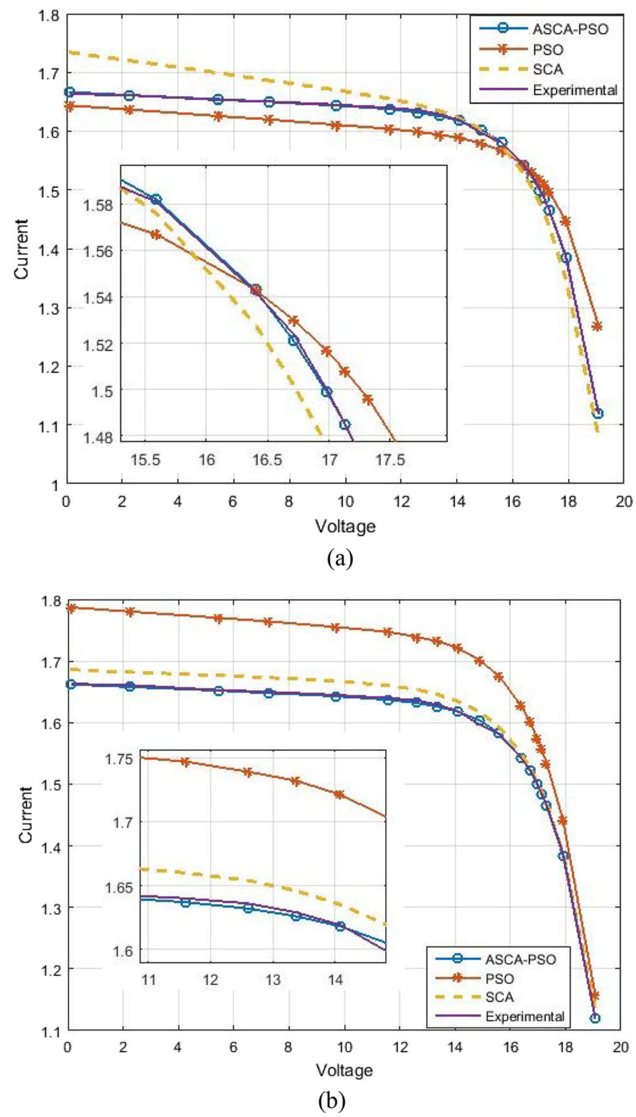

Fig. 10 Measured voltage versus current computed by using ASCA-PSO, SCA, and PSO for **a** SD and **b** DD circuit models

5.2.1 SD circuit model

The results of applying the ASCA-PSO technique for estimating the parameters of the SD circuit model of the STM6-40/36 module are presented and analyzed. In

Table 10 The ranges of PV cell parameters in the DD circuit model STM6-40/36 module [59]

Parameter	Lower bound	Upper bound
I_{ph} (A)	1	2
$I_{sd1}(A)$	10^{-12}	10^{-6}
$I_{sd2}(A)$	10^{-12}	10^{-6}
n_1	0	100
n_2	0	100
R_s (Ω)	0.01	1
R_{sh} (Ω)	300	800

Table 11 Statistics of the RMSE values, achieved by different optimization algorithms

	ASCA-PSO	PSO	SCA	BSA	ABC
Max	0.0046	0.1595	0.0221	0.00911	0.0053
Min	0.0022	0.0194	0.0046	0.00430	0.0020
Mean	0.0028	0.0681	0.0126	0.0066	0.0034
Std	9.12×10^{-5}	0.0368	0	0.0014	0.00081

For DD circuit model STM6-40/36 module

Table 7, the limits of values of the five parameters of the SD circuit model that were used in the tests are listed.

Table 8 presents a comparison between ASCA-PSO, SCA, and PSO for estimating the currents at different operating voltages and provides the absolute error relative to the experimentally measured current.

Table 8 shows the efficiency of ASCA-PSO over other algorithms for delivering RMSE smaller than most of the related work except ABC which has approximately a similar RMSE. However, ASCA-PSO has the minimum standard deviations which ensure the robustness and stability of ASCA-PSO over various PV modules. In a comparative study of the mean value of RMSE, it is possible to see that the ASCA-PSO is first ranked with a lower value, then the second is the ABC and the third is the BSA. The worst is PSO.

The estimated parameters of the SD circuit model using the ASCA-PSO technique in comparison with other related work are presented in Table 9. Here are shown the differences in the parameters that permit to obtain better performance.

For measuring the convergence of the rate of ASCA-PSO using the measurements of the STM6-40/36 module Fig. 9 shows that after a few iterations, the curve is converged at the optimum RMSE which reflects the quick performance rate of the convergence of ASCA-PSO. In Fig. 10, the curve shows that the algorithm converges after 5 iterations and then just adjusts the value. It is expected from the coevolutionary behavior of the ASCA-PSO.

5.2.2 DD circuit model

The limits of values of the seven parameters of the DD circuit model of the STM6-40/36 module are listed in Table 10. ASCA-PSO ensures its capability for enhancing the performance of PSO and SCA by decreasing absolute errors despite the increasing number of estimated parameters.

From Table 11, ASCA-PSO provides the minimum RMSE although ABC provides an RMSE smaller than that delivered in the SD circuit model. Also, ASCA-PSO keeps its superiority for the minimum standard deviation in comparison with other related works; it represents that the proposed approach is more stable. In this context, the mean value of the RMSE is also lower for the ASCA-PSO. Meanwhile, the worst value is forming the BSA.

Table 12 provides the optimal values for the parameters of the solar cell using a DD model. The RMSE is lower than the similar approaches, and it is also possible to see the differences between the values. The parameters then could be applied over the model and generate similar output to the dataset used in the iterative process.

The curves that are shown in Fig. 10 prove the efficiency of ASCA-PSO for estimating the parameters of SD and DD circuit models that produce approximately similar currents to the measured currents experimentally. Figure shows the power versus voltage analysis of ASCA-PSO

Table 12 Circuit model parameters for the DD circuit model of STM6-40/36 module estimated by ASCA-PSO in comparison with related works

	ASCA-PSO	PSO	SCA	BSA	ABC
I_{ph} (A)	1.661	1.791	1.68	1.66	1.66
$I_{sd1}(A)$	8.31×10^{-8}	3.35×10^{-7}	2.66×10^{-8}	1.198×10^{-6}	8.9×10^{-6}
$I_{sd2}(A)$	1.54×10^{-7}	7.84×10^{-8}	6.14×10^{-6}	8.91×10^{-6}	1×10^{-12}
n_1	55.53	53.07	46.98	100	71.46
n_2	50.63	57.07	68.54	52.874	27.79
R_s (Ω)	0.69	0.79	0.49	0.5000	1.23
R_{sh} (Ω)	508	316	603	924.81	938
RMSE	0.0028	0.0681	0.0126	0.0066	0.00334

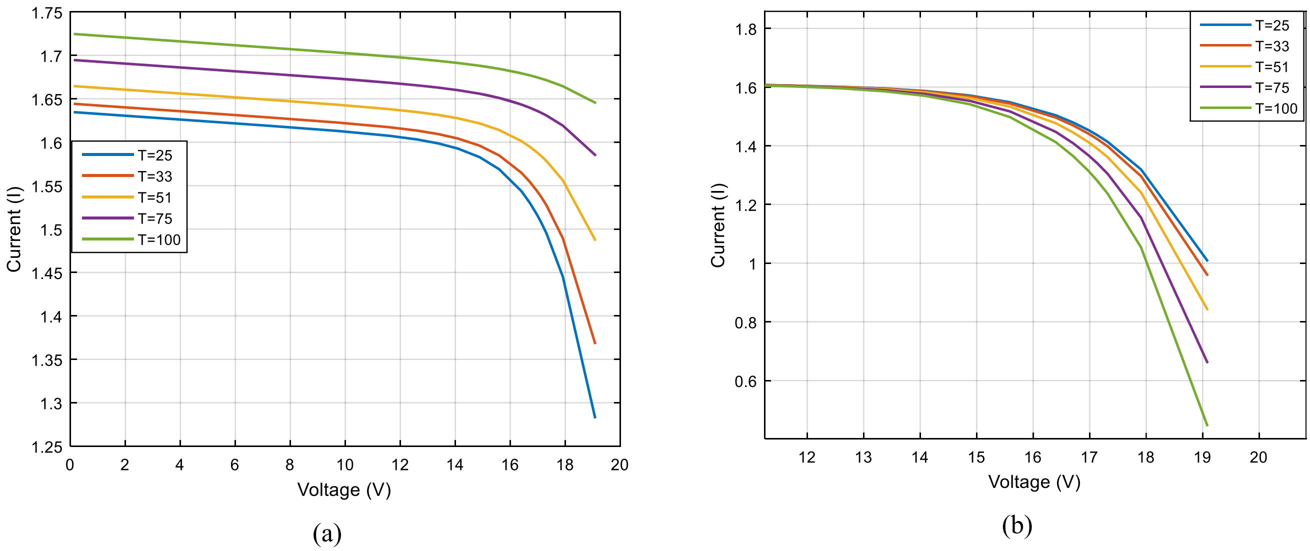

Fig. 11 Measured voltage versus current computed by using ASCA-PSO for different temperatures for **a** SD and **b** DD circuit models on the STM6-40/36 module

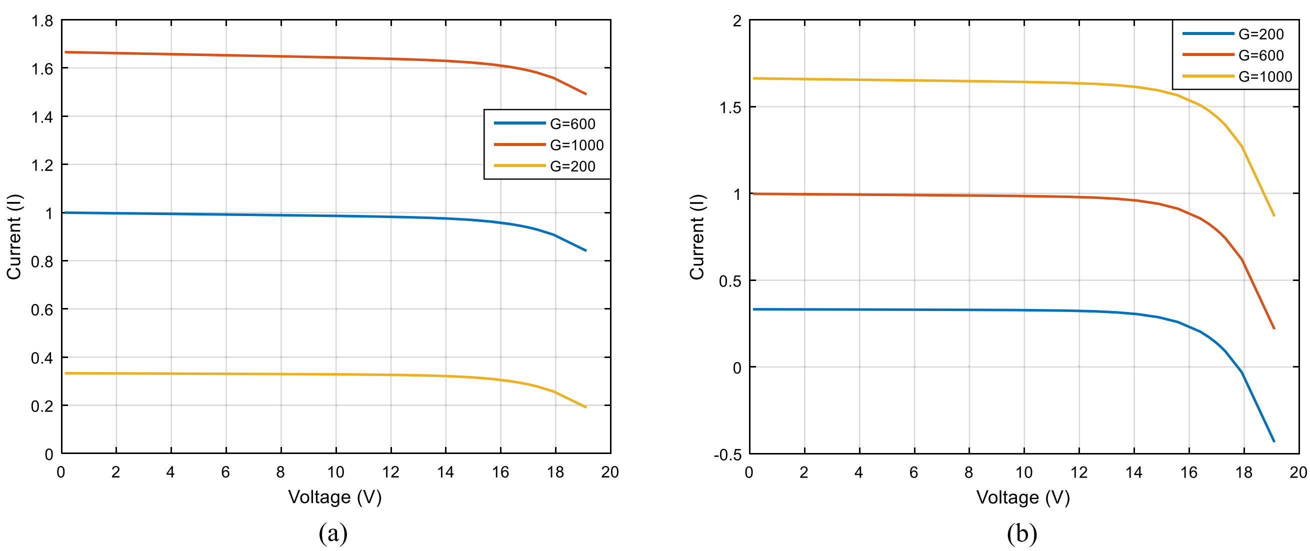

Fig. 12 Measured voltage versus current computed by using ASCA-PSO for different temperatures for **a** SD and **b** DD circuit models ON THE STM6-40/36 MODULE

in comparison with PSO, SCA, and the measured power. From the plotted results, it can be concluded that ASCA-PSO provides estimated currents more approximated to the measured current experimentally better than PSO and SCA.

Figures 11 and 12 show the temperature and irradiation effects on the parameter values estimated using ASCA-PSO at different operating conditions of the STM6-40/36 module. The use of different temperatures and irradiances is useful to verify that the proposed model (estimated by the ASCA-PSO) can work under different environmental conditions. It is expected that the values obtained allow the SC to be adapted and follow a similar response.

6 Conclusions

This paper examines the effectiveness of ASCA-PSO, a recently developed optimization algorithm, in estimating the parameters of photovoltaic cells for both single- and double-circuit models. The proposed algorithm employs a two-layer structure, with the top layer consisting of search agents controlled by PSO, each representing the global solution found by the agents in the corresponding bottom layer. The bottom layer comprises groups of search agents that update their movements based on SCA. This combination allows for simultaneous exploration and exploitation of the search space in the same iteration, thus enhancing the convergence rate and solution quality. The performance of ASCA-PSO is evaluated using two commercial photovoltaic modules, the R.T.C module and the STM6-40/36 module, each with 36 mono-crystalline cells and employing both single- and double-circuit models. The experimental results are compared to other related works and demonstrate the ability of ASCA-PSO to find global solutions for multimodal and complex objective functions with greater precision and stability, even in the presence of noise. The proposed model has the potential for application in more types of solar cells and more complex problems in the future.

Author contributions MI provided data curation, supervision, and writing—original draft preparation, methodology, conceptualization, and experimental tests. AH performed conceptualization, supervision, methodology, and writing—original draft preparation. MR revised writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding Open access funding provided by The Science, Technology & Innovation Funding Authority (STDF) in cooperation with The Egyptian Knowledge Bank (EKB). This research received no external funding.

Data availability The datasets generated during and/or analyzed during the current study are available in another published study (https://doi.org/https://doi.org/10.1016/j.renene.2017.04.014),

Declarations

Conflict of interest The authors clarify that there is no conflict of interest to report.

References

1. Carli R, Dotoli M, Pellegrino R (2017) A hierarchical decision-making strategy for the energy management of smart cities. IEEE Trans Autom Sci Eng 14(2):505–523
2. Lazaroiu GC, Roscia M (2012) Definition methodology for the smart cities model. Energy 47(1):326–332
3. Singh GK (2013) Solar power generation by PV (photovoltaic) technology: a review. Energy 53:1–13
4. Kabir E et al (2018) Solar energy: potential and prospects. Renew Sustain Energy Rev 82:894–900
5. Xu X, Wang H, Zuo Y (2011) Method for diagnosing photovoltaic array fault in a solar photovoltaic system. In: Power and energy engineering conference (APPEEC), 2011 Asia-Pacific, IEEE
6. Ma T, Yang H, Lu L (2014) Solar photovoltaic system modeling and performance prediction. Renew Sustain Energy Rev 36:304–315
7. Brano VL et al (2010) An improved five-parameter model for photovoltaic modules. Sol Energy Mater Sol Cells 94(8):1358–1370
8. Alam D, Yousri D, Eteiba M (2015) Flower pollination algorithm based solar PV parameter estimation. Energy Convers Manage 101:410–422
9. Chin VJ, Salam Z, Ishaque K (2015) Cell modeling and model parameters estimation techniques for photovoltaic simulator application: a review. Appl Energy 154:500–519
10. Avalos O et al (2019) A comparative study of evolutionary computation techniques for solar cells parameter estimation. Computación y Sistemas 23(1):231
11. Easwarakhanthan T et al (1986) Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers. Int J Solar Energy 4(1):1–12
12. Ma T, Yang H, Lu L (2014) Development of a model to simulate the performance characteristics of crystalline silicon photovoltaic modules/strings/arrays. Sol Energy 100:31–41
13. Saleem H, Karmalkar S (2009) An analytical method to extract the physical parameters of a solar cell from four points on the illuminated J–V curve. IEEE Electron Device Lett 30(4):349–352
14. Gong W, Cai Z (2013) Parameter extraction of solar cell models using repaired adaptive differential evolution. Sol Energy 94:209–220
15. Chan D, Phillips J, Phang J (1986) A comparative study of extraction methods for solar cell model parameters. Solid-State Electron 29(3):329–337
16. Jain A, Kapoor A (2004) Exact analytical solutions of the parameters of real solar cells using Lambert W-function. Sol Energy Mater Sol Cells 81(2):269–277
17. Ishaque K, Salam Z, Taheri H (2011) Simple, a fast and accurate two-diode model for photovoltaic modules. Sol Energy Mater Sol Cells 95(2):586–594
18. Ishaque K, Salam Z (2011) An improved modeling method to determine the model parameters of photovoltaic (PV) modules using differential evolution (DE). Sol Energy 85(9):2349–2359
19. Holland JH (1992) Genetic algorithms. Sci Am 267(1):66–73
20. Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. J Global Optim 11(4):341–359
21. Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl-Based Syst 96:120–133
22. Yazdani S, Nezamabadi-pour H, Kamyab S (2014) A gravitational search algorithm for multimodal optimization. Swarm Evol Comput 14:1–14
23. Kennedy (1995) Particle swarm optimization. Neural Netw

24. Karaboga D, Basturk B (2007) Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems. In: Foundations of fuzzy logic and soft computing, p 789–798
25. Karaboga D, Akay B (2009) A comparative study of artificial bee colony algorithm. Appl Math Comput 214(1):108–132
26. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm. Knowl-Based Syst 89:228–249
27. Lobato FS, Steffen V Jr, Silva Neto AJ (2010) A comparative study of the application of differential evolution and simulated annealing in radiative transfer problems. J Braz Soc Mech Sci Eng 32(SPE):518–526
28. Hannan M et al (2017) Optimization techniques to enhance the performance of induction motor drives: a review. Renew Sustain Energy Rev 81:1611–1626
29. Elfattah MA, et al (2016) Handwritten Arabic manuscript image binarization using sine cosine optimization algorithm. In: International conference on genetic and evolutionary computing, Springer
30. Emary E, Zawbaa HM, Hassanien AE (2016) Binary grey wolf optimization approaches for feature selection. Neurocomputing 172:371–381
31. Kim DH, Park JI (2005) Intelligent PID controller tuning of AVR system using GA and PSO. In: International conference on intelligent computing, Springer
32. Issa M, Hassanien AE (2017) Multiple sequence alignment optimization using meta-heuristic techniques. In: Ferraggine VE, Doorn JH, Rivero LC (eds) Handbook of research on machine learning innovations and trends. IGI Global, Pennsylvania, pp 409–423
33. Jervase JA, Bourdoucen H, Al-Lawati A (2001) Solar cell parameter extraction using genetic algorithms. Meas Sci Technol 12(11):1922
34. Ye M, Wang X, Xu Y (2009) Parameter extraction of solar cells using particle swarm optimization. J Appl Phys 105(9):094502
35. Geetha T et al (2013) An efficient survey on multi colony-particle swarm optimization (MC-PSO) algorithm. Int J Emerg Technol Adv Eng 3(4):672–677
36. Cong J. et al (2012) Solar cells performance testing and modeling based on particle swarm algorithm. In: 2012 International conference on computer science and information processing (CSIP), IEEE
37. Qin H, Kimball JW (2011) Parameter determination of photovoltaic cells from field testing data using particle swarm optimization. In: 2011 IEEE power and energy conference at Illinois, IEEE
38. Khanna V et al (2014) Estimation of photovoltaic cells model parameters using particle swarm optimization. Physics of semiconductor devices. Springer, Berlin, pp 391–394
39. Wei H, et al (2011) *Extracting solar cell model parameters based on chaos particle swarm algorithm*. In: 2011 international conference on electric information and control engineering, IEEE
40. Kim H-S, Hong SJ, Han S-S (2009) A comparison and analysis of genetic algorithm and particle swarm optimization using neural network models for high efficiency solar cell fabrication processes. In: 2009 IEEE international conference on fuzzy systems, IEEE
41. Khanna V et al (2015) A three diode model for industrial solar cells and estimation of solar cell parameters using PSO algorithm. Renew Energy 78:105–113
42. El-Naggar K et al (2012) Simulated annealing algorithm for photovoltaic parameters identification. Sol Energy 86(1):266–274
43. Guo L et al (2016) Parameter identification and sensitivity analysis of solar cell models with cat swarm optimization algorithm. Energy Convers Manage 108:520–528
44. Rajasekar N, Kumar NK, Venugopalan R (2013) Bacterial foraging algorithm based solar PV parameter estimation. Sol Energy 97:255–265
45. Askarzadeh A, Rezazadeh A (2012) Parameter identification for solar cell models using harmony search-based algorithms. Sol Energy 86(11):3241–3249
46. Allam D, Yousri D, Eteiba M (2016) Parameters extraction of the three diode model for the multi-crystalline solar cell/module using Moth-Flame Optimization Algorithm. Energy Convers Manage 123:535–548
47. Jordehi AR (2016) Parameter estimation of solar photovoltaic (PV) cells: a review. Renew Sustain Energy Rev 61:354–371
48. Askarzadeh A, Rezazadeh A (2013) Extraction of maximum power point in solar cells using bird mating optimizer-based parameters identification approach. Sol Energy 90:123–133
49. Das S, Abraham A, Konar A (2008) Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. Advances of computational intelligence in industrial systems. Springer, Berlin, pp 1–38
50. Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. IEEE Trans Evol Comput 1(1):67–82
51. Issa M et al (2018) ASCA-PSO: adaptive sine cosine optimization algorithm integrated with particle swarm for pairwise local sequence alignment. Expert Syst Appl 99:56–70
52. Yu K et al (2018) Multiple learning backtracking search algorithm for estimating parameters of photovoltaic models. Appl Energy 226:408–422
53. Jordehi AR (2017) Gravitational search algorithm with linearly decreasing gravitational constant for parameter estimation of photovoltaic cells. In: Evolutionary computation (CEC), 2017 IEEE Congress on, IEEE
54. Chen X et al (2016) Parameters identification of solar cell models using generalized oppositional teaching learning based optimization. Energy 99:170–180
55. Chen X et al (2018) Teaching–learning–based artificial bee colony for solar photovoltaic parameter estimation. Appl Energy 212:1578–1588
56. Zhang Y et al (2016) A population classification evolution algorithm for the parameter extraction of solar cell models. Int J Photoenergy 2016:129–144
57. Zhou X et al (2016) Gaussian bare-bones artificial bee colony algorithm. Soft Comput 20(3):907–924
58. Oliva D, Cuevas E, Pajares G (2014) Parameter identification of solar cells using artificial bee colony optimization. Energy 72:93–102
59. Tong NT, Pora W (2016) A parameter extraction technique exploiting intrinsic properties of solar cells. Appl Energy 176:104–115