Mathematical Biosciences and Engineering

MBE, 19(6): 5610–5637.
DOI: 10.3934/mbe.2022263
Received: 23 February 2022
Revised: 24 March 2022
Accepted: 25 March 2022
Published: 30 March 2022

http://www.aimspress.com/journal/MBE

Research article

An enhanced adaptive comprehensive learning hybrid algorithm of Rao-1 and JAYA algorithm for parameter extraction of photovoltaic models

Yujun Zhang[1], Yufei Wang[1], Shuijia Li[2], Fengjuan Yao[3], Liuwei Tao[4], Yuxin Yan[5], Juan Zhao[1,*] and Zhengming Gao[3,6]

[1] School of electronics and information engineering, Jingchu University of Technology, Jingmen 448000, China
[2] School of Computer Science, China University of Geosciences, Wuhan 430074, China
[3] School of computer engineering, Jingchu University of Technology, Jingmen, 448000, China
[4] School of foreign languages, Jingchu University of Technology, Jingmen, 448000, China
[5] Academy of arts, Jingchu University of Technology, Jingmen 448000, China
[6] Institute of intelligent information technology, Hubei Jingmen industrial technology research institute, Jingmen 448000, China

* **Correspondence:** Email: ajuan323@jcut.edu.cn.

Abstract: In order to maximize the acquisition of photovoltaic energy when applying photovoltaic systems, the efficiency of photovoltaic system depends on the accuracy of unknown parameters in photovoltaic models. Therefore, it becomes a challenge to extract the unknown parameters in the photovoltaic model. It is well known that the equations of photovoltaic models are nonlinear, and it is very difficult for traditional methods to accurately extract its unknown parameters such as analytical extraction method and key points method. Therefore, with the aim of extracting the parameters of the photovoltaic model more efficiently and accurately, an enhanced hybrid JAYA and Rao-1 algorithm, called EHRJAYA, is proposed in this paper. The evolution strategies of the two algorithms are initially mixed to improve the population diversity and an improved comprehensive learning strategy is proposed. Individuals with different fitness are given different selection probabilities, which are used to select different update formulas to avoid insufficient using of information from the best individual and overusing of information from the worst individual. Therefore, the information of different types of individuals is utilized to the greatest extent. In the improved update strategy, there are two different adaptive coefficient strategies to change the priority of information. Finally, the combination of the linear population reduction strategy and the dynamic lens opposition-based learning strategy, the convergence speed of the algorithm and ability to escape from local optimum can be improved. The results of various experiments prove that the proposed EHRJAYA has superior performance and rank in the leading position among the famous algorithms.

Keywords: parameter extraction; photovoltaic models; JAYA algorithm; Rao-1 algorithm; EHRJAYA algorithm

1. Introduction

With the development of the world and the consumption of resources, it is increasingly necessary to find better, clean and renewable new energy. Among the new types of energy, photovoltaic energy [1] is known as the most potential renewable energy for long-term development. However, the collection of photovoltaic energy depends on the photovoltaic model, and the performance of the photovoltaic model is determined by the unknown parameters [2] in the model. So far, a variety of photovoltaic models have been created, including single-diode model [3] (SDM), double-diode model [4] (DDM), three-diode model [5] (TDM), etc., but the most widely [6] used are the single-diode model and double-diode model. Parameters will change when faced with uncertain factors in use, which will affect the efficiency of photovoltaic models. Therefore, it is very necessary to extract the unknown parameters of the photovoltaic model for evaluating the performance of the model before it is used.

Many methods have been proposed to extract unknown parameters in photovoltaic models. They can be roughly divided into three categories: analytical methods [7,8], deterministic methods [9,10] and meta-heuristics. The first two methods are dependent on the initial value set by the model and the necessary assumptions, so the achieved solution is not accurate. Moreover, these two methods are prone to fall into local optimum, which will result in the inability to find the optimal solution. Compared with the other two methods, the meta-heuristic has natural advantages. It has a simple structure, high efficiency, strong ability to jump out of the local optimum, and does not depend on the specific setting of parameters. Therefore, meta-heuristics are used by many scholars for parameter extraction of photovoltaic models. For example, Simulated Annealing Algorithm [11] (SA), Whale Optimization Algorithm [12] (WOA), Differential Evolution Algorithm [13] (DE), Harmony Search Algorithm [14] (HS), Cuckoo Search Algorithm [15] (CS), Genetic Algorithm [16] (GA), Artificial Bee Colony Algorithm [17] (ABC), Teaching-Learning-Based Optimization Algorithm [18] (TLBO), Ant Lion Optimizer [19] (ALO), Arithmetic Optimization Algorithm [20] (AOA), Sine Cosine Algorithm [21] (SCA), Rao-1 Algorithm [22], Empire Competition Algorithm [23] (ICA), Marine predators algorithm [24] (MPA), Harris Hawk optimization algorithm [25] (HHO), Runge Kutta Optimizer [26] (RUN), Artificial Hummingbird Algorithm [27] (AHA), etc. Recently, many excellent algorithms have been proposed for photovoltaic model of parameter extraction. Abdullrahman A. et al. [28] used Bonobo Optimizer for parameter extraction of photovoltaic models. Chen et al. [29] proposed a bi-subgroup optimization algorithm (BSOA) for parameter extraction of proton exchange membrane fuel cell (PEMFC). Rezk et al. [30] proposed a robust methodology based on the Gradient-based Optimizer (GBO) for extracting optimal parameters for PEM fuel cells (PEMFC). Babu et al. [31] introduced a robust method based on stochastic fractal search (SFS) optimization algorithm for extracting parameters of photovoltaic models. Therefore, no matter what type of photovoltaic model it is, the meta-heuristic algorithm can be used to optimize it. On the one hand, this also proves that the meta-heuristic algorithm is very useful. On the other hand, in order to minimize the complexity and computational resources, there is still a need to find better algorithms.

The JAYA algorithm [32] and Rao-1 algorithm [33] are swarm-based optimization algorithms. The structure of both algorithms is very simple, and there is no need to set redundant parameters. That is, there is only one common parameter, namely the population size. This means that the algorithm runs very fast. In addition, there is only one evolution strategy for these two algorithms. Therefore, they have been deeply improved by many scholars to solve various complex problems [34–41].

In this paper, an enhanced hybrid JAYA and Rao-1 algorithm, called EHRJAYA, is proposed. In the EHRJAYA, various improvement strategies are introduced. First, the evolution strategies of the two algorithms are mixed, which improves the population diversity of the algorithm and avoids a single strategy from reducing the performance of the algorithm. Then, an improved comprehensive learning strategy is introduced. Different selection probabilities populations with different fitness are given to, so that different update formulas are selected, which avoid insufficient using of information from the best individual and overusing of information from the worst individual. Then, two different adaptive coefficient strategies are introduced into the evolution strategies of the two algorithms. The common point of these two adaptive coefficients is to guide the population towards the optimal individual and away from the worst individual. Then, linear population reduction strategy is introduced, to improve the convergence speed. Finally, dynamic lens opposition-based learning strategy is introduced, improving the situation where the algorithm gets stuck in local optimum.

To verify the performance of the proposed EHRJAYA, five photovoltaic models are selected, including the single-diode model, the double-diode model, the Photowatt-PWP201 model, the STM6-40/36 model, and the STP6-120/36 model. The comparison results of the comparison algorithms in the experiment or the final comparison results with the well-known reported algorithms, the EHRJAYA has excellent performance and locates in a leading position.

The main contributions of this paper are as follows:

- An enhanced hybrid JAYA and Rao-1 algorithm is proposed for extracting the parameters of photovoltaic models efficiently.
- An enhanced adaptive comprehensive learning strategy is proposed.
- Two adaptive coefficients are introduced to avoid underutilizing the information from the best individual and overusing the information from the worst individual.
- Linear population reduction strategy and dynamic lens opposition-based learning strategy are combined to improve the convergence speed of the algorithm and the situation of falling into local optimum.
- The superior performance of the EHRJATA is verified by five photovoltaic models. Compared with many well-known algorithms, the superiority of the algorithm is further confirmed.

The rest of this paper is arranged as follows: In Section 2, the definition of Photovoltaic (PV) model and objective function is introduced. In Section 3, the JAYA and Rao-1 algorithm is introduced. In Section 4, the EHRJAYA is introduced. In Section 5, experiments and result analysis are carried out by the EHRJAYA and the comparison algorithms. In Section 6, the EHRJAYA is summarized.

2. Definition of Photovoltaic (PV) Model

2.1. Single-diode model (SDM)

The characteristics of solar cells can be accurately described by SDM. SDM can be expressed by the formula (1)–(3).

$$I_L = I_{pv} - I_d - I_p \tag{1}$$

$$I_d = I_{sd}\left[\exp(\frac{(V_L + I_L R_s)\times q}{nkT}) - 1\right] \tag{2}$$

$$I_p = \frac{V_L + I_L R_s}{R_P} \tag{3}$$

where, I_L is the terminal current, I_p is the shunt resistor current, I_d is the diode current, I_{pv} is the current generated by solar irradiation, I_{sd} is the diode saturation current, V_L is the output voltage, R_s and R_P are the series and shunt resistances respectively, n is the diode characteristic factor, $k = 1.3806503 \times 10^{-23} J/K$ and $q = 1.60217646 \times 10^{-19} C$ are both constants, T is the temperature of junction in Kelvin.

Therefore, the output current of the SDM can be expressed by the formula (4).

$$I_L = I_{pv} - I_{sd}\left[\exp(\frac{(V_L + I_L R_s)\times q}{nkT}) - 1\right] - \frac{V_L + I_L R_s}{R_P} \tag{4}$$

It can be seen from the formula that this model needs to extract five unknown parameters including I_{pv}, I_{sd}, R_s, R_P and n.

2.2. Double-diode model (DDM)

DDM adds a diode on the basis of SDM, so the effect of loss of recombination current is considered. This model can be expressed by the formula (5)–(7).

$$I_L = I_{pv} - I_{d1} - I_{d2} - I_p \tag{5}$$

$$I_{d_1} = I_{sd_1}\left[\exp(\frac{(V_L + I_L R_s)\times q}{n_1 kT}) - 1\right] \tag{6}$$

$$I_{d_2} = I_{sd_2}\left[\exp(\frac{(V_L + I_L R_s)\times q}{n_2 kT}) - 1\right] \tag{7}$$

where, I_{sd_1} and I_{sd_2} are the diffusion current and saturation current, and n_1 and n_2 are the ideality factors of the two diodes, respectively.

Therefore, the output current of the DDM can be expressed by the formula (8).

$$I_L = I_{pv} - I_{sd_1}\left[\exp(\frac{(V_L + I_L R_s)\times q}{n_1 kT}) - 1\right] - I_{sd_2}\left[\exp(\frac{(V_L + I_L R_s)\times q}{n_2 kT}) - 1\right] - \frac{V_L + I_L R_s}{R_P} \tag{8}$$

It can be seen from the formula that this model needs to extract seven unknown parameters including I_{pv}, I_{sd_1}, I_{sd_2}, R_s, R_P, n_1 and n_2.

2.3. Photovoltaic (PV) module model

The PV module model is built on multiple PV cells connected in parallel and in series.

Therefore, it can be expressed by the formula (9).

$$I_L = I_{pv}N_p - I_{sd}N_p\left[\exp(\frac{(V_LN_p + I_LR_sN_s) \times q}{nN_sN_pkT}) - 1\right] - \frac{V_LN_p + I_LR_sN_s}{R_PN_s} \quad (9)$$

where, N_s represents the number of photovoltaic cells in series, and N_p represents the number of photovoltaic cells in parallel.

It can be seen from the formula that this model needs to extract five unknown parameters, including I_{pv}, I_{sd}, R_s, R_P and n.

2.4. Objective function of PV model

The parameters of the above models are estimated when using data provided by the supplier. Usually, an objective function is needed to estimate the error of the experiment. In this paper, the root mean square error (RMSE) is adopted as the objective function for optimization because it can reflect the degree of error between the measured data and the real data.

$$min \quad RMSE(x) = \sqrt{\frac{\sum_{i=1}^{N}(I_i - I_L)^2}{N}} \quad (10)$$

where, N is the number of datasets, I_L is the calculated current, and I_i is the data provided by the supplier.

It can be seen from formula (10) that when the value of RMSE is smaller, the extracted parameters are more accurate.

3. Rao-1 and JAYA algorithm

3.1. JAYA algorithm

In the JAYA algorithm, the property of no public parameters is very attractive, which means that the algorithm is unlimited and simple in structure. The idea of the algorithm is very simple, and it aims to move towards the optimal position and away from the worst position, which can be expressed by the formula (11).

$$Y_{i,j} = X_{i,j} + rand_1 \times \left(X_{best,j} - |X_{i,j}|\right) - rand_2 \times \left(X_{worst,j} - |X_{i,j}|\right) \quad (11)$$

where, $X_{best,j}$ represents the best solution, $X_{worst,j}$ represents the worst solution, and $rand_1$and $rand_2$ are random numbers between 0 and 1.

If the fitness value of the updated solution is better than the previous solution, then the updated solution can be accepted, otherwise, the previous solution is kept, which shows as formula (12).

$$X_i = \begin{cases} Y_i, & if\ f(Y_i) < (X_i) \\ X_i, & otherwise \end{cases} \quad (12)$$

3.2. Rao-1 algorithm

Different from the JAYA algorithm, the difference information between the best search agent and the worst search agent is used by the Rao-1 algorithm to mutate, which can be expressed by the formula (13).

$$Z_{i,j} = X_{i,j} + rand_3 \times \left(X_{best,j} - X_{worst,j}\right) \tag{13}$$

After mutation, the mutated solution or the original solution is selected by the same update formula as the JAYA algorithm.

$$X_i = \begin{cases} Z_i, & if\ f(Z_i) < (X_i) \\ X_i, & otherwise \end{cases} \tag{14}$$

4. The proposed EHRJAYA algorithm

For both the JAYA and Rao-1 algorithm, there is potential for further improvement, because these two algorithms have clear structures, simple evolution strategies, and clear concepts. However, the evolution strategies of the two algorithms are relatively simple, the search and development are not thorough enough, and they do not even have the conditions to jump out of the local optimum. Therefore, some improvement strategies are put forward.

4.1. Hybrid evolutionary strategy

Algorithms with a single evolution strategy may have insufficient population diversity, which will lead to low exploration performance. Therefore, the evolution strategy of Rao-1 algorithm is introduced. Random numbers are judged, and two evolutionary strategies are randomly selected, which can be expressed by the formula (15).

$$X'_{i,j} = \begin{cases} Y_{i,j}, & rand < 0.5 \\ Z_{i,j}, & otherwise \end{cases} \tag{15}$$

Through the above method, one of these two strategies will be randomly selected in one population evolution, which avoids the limitation caused by a single evolution strategy and improves the population diversity of the algorithm.

4.2. Enhanced comprehensive learning strategy

In the population, there are the best and worst search agents. If these agents all pass the same update formula, it may lead to underutilization of the best search agents and overuse of the worst search agents. Therefore, the best and worst search agents need to be separated when using different update formulas to improve the utilization efficiency of search agents as a whole. A reinforced comprehensive learning strategy is proposed. Different search agents are given different selection probabilities. Therefore, by judging this probability, they are used to select different update formulas.

In order to establish the relationship between the search agent and the selection probability of the search agent, first the fitness values are sorted.

$$index = sort(fitness) \tag{16}$$

Then, the indexes of the sorted search agents are assigned to the corresponding search agents.

$$P(index(i)) = \frac{i}{NP\_current} \tag{17}$$

Therefore, search agents with smaller fitness are assigned smaller values. For search agents corresponding to different fitness, different update formulas (18) are introduced.

$$Y_{i,j} = \begin{cases} X_{i,j} + rand_1 \times (X_{best,j} - |X_{i,j}|) - rand_2 \times (X_{worst,j} - |X_{i,j}|), & P \leq 1/3 \\ X_{i,j} + rand_1 \times (X_{best,j} - |X_{i,j}|) - rand_2 \times (Mean(X) - |X_{i,j}|), & 1/3 < P \leq 2/3 \\ X_{i,j} + rand_1 \times (X_{best,j} - |X_{i,j}|) + rand_2 \times (X_{r1,j} - X_{r2,j}), & otherwise \end{cases} \tag{18}$$

where $Mean(X)$ represents the mean of all search agents, and $r1$ and $r2$ are randomly selected in the entire population except the current search agent. Among the three update formulations, the superior search agent can be utilized to the maximum, the medium search agent can be utilized evenly, and the other solutions are randomly selected by the worst search agent for improvement.

Similarly, we also improve the Rao-1 algorithm using the above-mentioned comprehensive learning strategy.

$$Z_{i,j} = \begin{cases} X_{i,j} + rand_3 \times (X_{best,j} - X_{worst,j}) - rand_4 \times (X_{r1,j} - X_{worst,j}), & P \leq 1/3 \\ X_{i,j} + rand_3 \times (X_{best,j} - X_{worst,j}) - rand_4 \times (Mean(X) - X_{r2,j}), & 1/3 < P \leq 2/3 \\ X_{i,j} + rand_3 \times (X_{best,j} - X_{worst,j}) + rand_4 \times (X_{r1,j} - X_{r2,j}), & otherwise \end{cases} \tag{19}$$

In the update formula in the improved Rao-1 algorithm, different types of search agents have corresponding leadership mechanisms. Therefore, the search agents in the population can be utilized to the maximum, improving the performance of the algorithm.

4.3. Two adaptive coefficient updating strategies

4.3.1. The first method of coefficient adaptation

In the evolution strategy of the JAYA algorithm, the priority of the information from the best and worst search agents is the same, which easily leads to the algorithm not being able to make better use of the information of the best individual or overusing the information of the worst individual. Therefore, an adaptive coefficient strategy [42] is introduced

$$A_1 = \begin{cases} \dfrac{mean(f(X))}{f(X_{best})}, f(X_{best}) \neq 0 \\ 1, f(X_{best}) = 0 \end{cases} \tag{20}$$

$$A_2 = \begin{cases} \dfrac{mean(f(X))}{f(X_{worst})}, f(X_{worst}) \neq 0 \\ 1, f(X_{worst}) = 0 \end{cases} \tag{21}$$

where, X_{best} and X_{worst} are the global best solution and the worst solution, respectively. In the process of optimization, the value of A_1 is greater than 1, and the value of A_2 is less than 1. As the iteration continues to increase, they eventually approach 1. Therefore, the improved evolution strategy can be expressed as follows.

$$Y_{i,j} = \begin{cases} X_{i,j} + A_1 \times rand_1 \times (X_{best,j} - |X_{i,j}|) - A_2 \times rand_2 \times (X_{worst,j} - |X_{i,j}|), & P \leq 1/3 \\ X_{i,j} + A_1 \times rand_1 \times (X_{best,j} - |X_{i,j}|) - A_2 \times rand_2 \times (Mean(X) - |X_{i,j}|), & 1/3 < P \leq 2/3 \\ X_{i,j} + A_1 \times rand_1 \times (X_{best,j} - |X_{i,j}|) + A_2 \times rand_2 \times (X_{r1,j} - X_{r2,j}), & otherwise \end{cases} \quad (22)$$

4.3.2. The second method of coefficient adaptation

This adaptive coefficient [43] strategy adaptively adjusts towards the optimal solution and away from the optimal solution by accumulating successful trend factors. The improved evolution strategy is represented as formula (23).

$$Z_{i,j} = \begin{cases} X_{i,j} + T1_i \times (X_{best,j} - X_{worst,j}) - T2_i \times (X_{r1,j} - X_{worst,j}), & P \leq 1/3 \\ X_{i,j} + T1_i \times (X_{best,j} - X_{worst,j}) - T2_i \times (Mean(X) - X_{r2,j}), & 1/3 < P \leq 2/3 \\ X_{i,j} + T1_i \times (X_{best,j} - X_{worst,j}) + T2_i \times (X_{r1,j} - X_{r2,j}), & otherwise \end{cases} \quad (23)$$

where, $T1_i$ and $T2_i$ are calculated by Cauchy distribution according to $\mu T(*)$ and 0.1. When $T(*)_i$ is greater than 1, it is assigned a value of 1, and when $T(*)_i$ is less than 0, it is recalculated. The calculation formula is as formula (24), (25).

$$T(*)_i = randc(\mu T(*), 0.1) \quad (24)$$

$$\mu T(*) = (1 - b) \times \mu T(*) + b \times mean(S_{T(*)}) \quad (25)$$

where, the initial value of $\mu T(*)$ is 0.5, b is 0.1, $mean(S_{T(*)})$ is the mean of the set of successfully generated trend factors.

4.4. Linear population reduction strategy

The JAYA and the Rao-1 algorithm are easily affected by the size of the population. Therefore, the linear reduction of the population [44] can be introduced to improve the operating efficiency of the algorithm, that is, the search agent corresponding to the worst fitness is eliminated after each evaluation.

$$NP_{G+1} = round\left[\left(\frac{NP_{min} - NP_{init}}{MaxNFES}\right) \times NFES + NP_{init}\right] \quad (26)$$

where, NP_{min} is the population size at the end of the algorithm iteration, which is set to 3, NP_{init} is the initial population size, $NFES$ is the current number of evaluations, $MaxNFES$ is the maximum number of evaluations, NP_G is the population size of the current generation, and NP_{G+1} is the population size of the next generation. The poor search agents are gradually reduced, on the one hand, the overall level of search agents and the convergence speed of the algorithm are improved. On the other hand, the continuous reduction of the population will lead to the overall transition from the exploration phase to the local development phase.

4.5. Lens Opposition-based Learning strategy

The most commonly used opposition-based learning strategy [45] only consider the opposite of the candidate solutions, however, this fixed value may not satisfy the dynamic changes of the solutions, so lens opposition-based learning strategy is introduced. The opposites of the candidate solutions in different situations can be effectively selected by the dynamic lens opposition-based learning strategy, which improves the population diversity of the algorithm.

$$X_{i,j}^{*}=\frac{min(X_j)+max(X_j)}{2}+\frac{min(X_j)+max(X_j)}{2\times k}-\frac{X_{i,j}}{k} \tag{27}$$

$$k=u_{max}-(u_{max}-u_{min})\times\frac{NFES}{MaxNFES} \tag{28}$$

Where, u_{max} and u_{min} are 2 and 0 respectively, $NFES$ is the current number of evaluations, $MaxNFES$ is the maximum number of evaluations. When K=1, it can be found that the lens opposition-based learning strategy is commonly used opposition-based learning strategy. Therefore, the commonly used opposition-based learning strategy is only one of the cases of the lens opposition-based learning strategy.

4.6. Overview of the EHRJAYA

Algorithm 1: The pseudo-code of the proposed EHRJAYA

```
Set population size $NP$, the maximum number of evaluations $MaxNFES$, dimension $Dim$
Initialize the positions of Individuals $X_i(i=1,2,\dots,NP)$
Set $NFES=0,\ NP=NP_{max}=50,\ NP_{min}=3,\ NFES=NP.$
While ($NFES\le MaxNFES$)
   Calculate coefficients $A_1,A_2$ and $T_1,T_2$ using Eqs (20), (21), (24) and (25)
   For $i=1:NP$
   Calculate the selection probability $P$ using Eqs (16) and (17)
   End For
   For $i=1:NP$
        If rand<0.5 then
        Update the new position $X'$ using Eq (22)
        Else
        Update the new position $X'$ using Eq (23)
        If $f(X')<f(X)$ then
        $X=X'$
   End if
   End For
   Calculate the new population $NP_{G+1}$ using Eq (26)
   If $rand<0.3$ then
        Calculate the new position $X_{i,j}^{*}$ using Eq (27)
   End If
Memory saving
End While
Return $X_{best}$
```

Through the introduction of the above five improved strategies, the EHRJAYA is proposed. It is worth noting that these strategies are dynamically adaptive, which means that the proposed EHRJAYA has strong adaptability when faced with different problems. The complexity of the proposed EHRJAYA is $O(G_{max} \times NP \times (log(NP) + Dim)$,where G_{max} is the maximum number of iterations, Dim is the population dimension, and NP is the population size. The pseudocode of the EHRJAYA is provided by Algorithm 1. It can be seen from Algorithm 1 that at the beginning of the iteration, two kinds of adaptation coefficients are calculated and the selection probability of each individual is performed. Then, an evolution strategy is randomly selected, and each search agent chooses different update formula according to different selection probabilities. Finally, lens opposition-based learning is performed.

5. Experiments and results

5.1. Parameter settings and PV model selection

In order to verify the performance of the proposed EHRJAYA algorithm, adaptive guided differential evolution algorithm (AGDE) [46], multiple learning backtracking search algorithm (MLBSA) [47], triple archives particle swarm optimization (TAPSO) [48], generalized oppositional teaching learning-based optimization (GOTLBO) [49], improved JAYA optimization algorithm (IJAYA) [50], Rao-1, improved teaching-learning-based optimization (ITLBO) [51], JAYA, performance-guided JAYA algorithm (PGJAYA) [52], and teaching-learning-based artificial bee colony algorithms (TLABC) [53] are selected as comparison algorithms. The setting of specific parameters is shown in Table 1. The maximum number of evaluations is set to 30000 for the single-diode, the double-diode, Photowatt-PWP201, STM6-40/36 and STP6-120/36. All experiments are run 30 times. All of the simulation experiments would be carried out with HP DL380 Gen 10 server with 32GB RAM and Intel Xeon Bronze 3106×2 cores, and MATLAB 2017b software.

Three different PV models are chosen to test the performance of the EHRJAYA on four PV datasets. For single-diode and double-diode model, R.T.C France solar cell of the 57 mm diameter commercial is selected. For the PV module model, monocrystalline STM6-40/36 and polycrystalline STP6-120/36 is selected. The setting of specific relevant parameters is presented in Tables 2 and 3.

Table 1. Parameter settings of algorithms.

Algorithm	Parameter
EHRJAYA	$NP = 50$
PGJAYA	$NP = 50$
JAYA	$NP = 50$
AGDE	$NP = 50, \varepsilon = 0.01$
TLABC	$NP = 50, limit = 200, F = rand(0,1)$
MLBSA	$NP = 50, \varepsilon = 0.01$
TAPSO	$NP = 50, \omega = 0.7298, P_C = 0.5, P_m = 0.02$
GOTLBO	$NP = 50, Jr = 0.3$
IJAYA	$NP = 50$
Rao-1	$NP = 50$
ITLBO	$NP = 50$

Table 2. Correlation data of three PV models.

Parameter	The single/double-diode model	Photowatt-PWP201	STM6-40/36	STP6-120/36
NP	1	1	1	1
NS	1	36	36	36
Data Volume	26	25	20	24
temperature	25℃	45℃	51℃	55℃
Radiance	1000 W/m^2	1000 W/m^2	1000 W/m^2	1000 W/m^2

Table 3. Parameter settings of three PV models.

Parameter	R.T.C. France solar cell		Photowatt-PWP201		STM6-40/36		STP6-120/36	
	LB	UB	LB	UB	LB	UB	LB	UB
$\boldsymbol{I_{pv}}$(A)	0	1	0	2	0	2	0	8
$\boldsymbol{I_{sd_1}}$, $\boldsymbol{I_{sd_2}}$, $\boldsymbol{I_{sd}}(\boldsymbol{\mu}$A)	0	1	0	50	0	50	0	50
$\boldsymbol{R_P}$(Ω)	0	100	0	2	0	1000	0	1500
$\boldsymbol{R_S}$(Ω)	0	0.5	0	2000	0	0.36	0	0.36
$\boldsymbol{n_1}$, $\boldsymbol{n_2}$, $\boldsymbol{n}$	1	2	1	50	1	60	1	50

5.2. Experimental results of the single-diode model

For the single diode model, the extracted parameters and the corresponding RMSE are shown in **Table 4**. All algorithms except IJAYA and TAPSO obtained the best RMSE. It is preliminarily verified that the performance of the EHRJAYA does not lose to other algorithms on the single-diode model.

The superiority of the algorithm in the process of extracting parameters of this model can be proved by the convergence analysis. The convergence curve is shown in Figure 1. From the details in the Figure 1, it can be seen that the proposed EHRJAYA converges faster and performs better than other comparison algorithms on the single-diode model.

These extracted parameters cannot be intuitively seen to be correct. Therefore, these parameters are re-substituted into the objective function, and the simulated current and power values are re-calculated. The results are shown in Figure 2. The fit of the simulated and measured values is very high, which indirectly proves the accuracy of the algorithm.

Table 4. Extracted parametric results on the single-diode model.

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}}(\boldsymbol{\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	$\boldsymbol{n}$	RMSE
EHRJAYA	7.60775530E-01	3.23020841E-01	3.63770923E-02	5.37185275E+01	1.48118359E+00	**9.86021878E-04**
PGJAYA	7.60775530E-01	3.23020839E-01	3.63770922E-02	5.37185235E+01	1.48118360E+00	**9.86021878E-04**
JAYA	7.60775530E-01	3.23020833E-01	3.63770924E-02	5.37185272E+01	1.48118360E+00	**9.86021878E-04**
AGDE	7.60775530E-01	3.23020816E-01	3.63770925E-02	5.37185260E+01	1.48118359E+00	**9.86021878E-04**
TLABC	7.60775530E-01	3.23020824E-01	3.63770925E-02	5.37185285E+01	1.48118359E+00	**9.86021878E-04**
MLBSA	7.60677240E-01	3.79295956E-01	3.57846147E-02	5.98687132E+01	1.49747817E+00	**9.86021878E-04**
TAPSO	7.60799109E-01	3.29069246E-01	3.63049519E-02	5.40075682E+01	1.48305900E+00	9.86023131E-04
GOTLBO	7.60677240E-01	3.79295956E-01	3.57846147E-02	5.98687132E+01	1.49747817E+00	**9.86021878E-04**
IJAYA	7.60753937E-01	3.33910823E-01	3.62442162E-02	5.47278261E+01	1.48452218E+00	9.89877058E-04
Rao-1	7.60684299E-01	4.59902816E-01	3.49183645E-02	6.51608749E+01	1.51766650E+00	**9.86021878E-04**
ITLBO	7.60799109E-01	3.29069246E-01	3.63049519E-02	5.40075682E+01	1.48305900E+00	**9.86021878E-04**

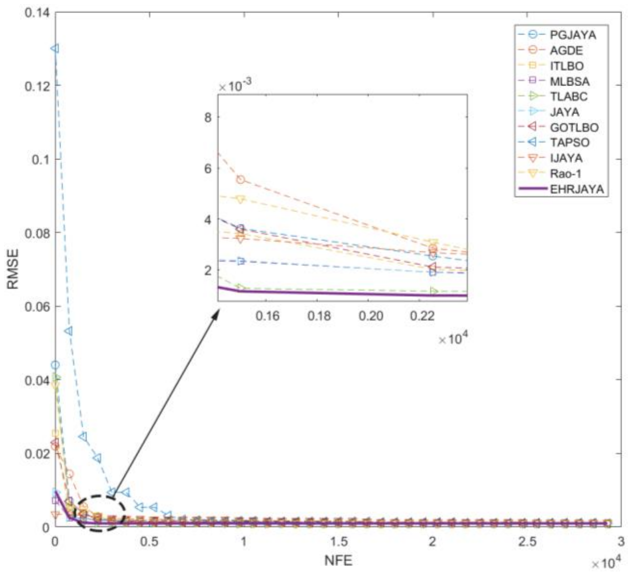
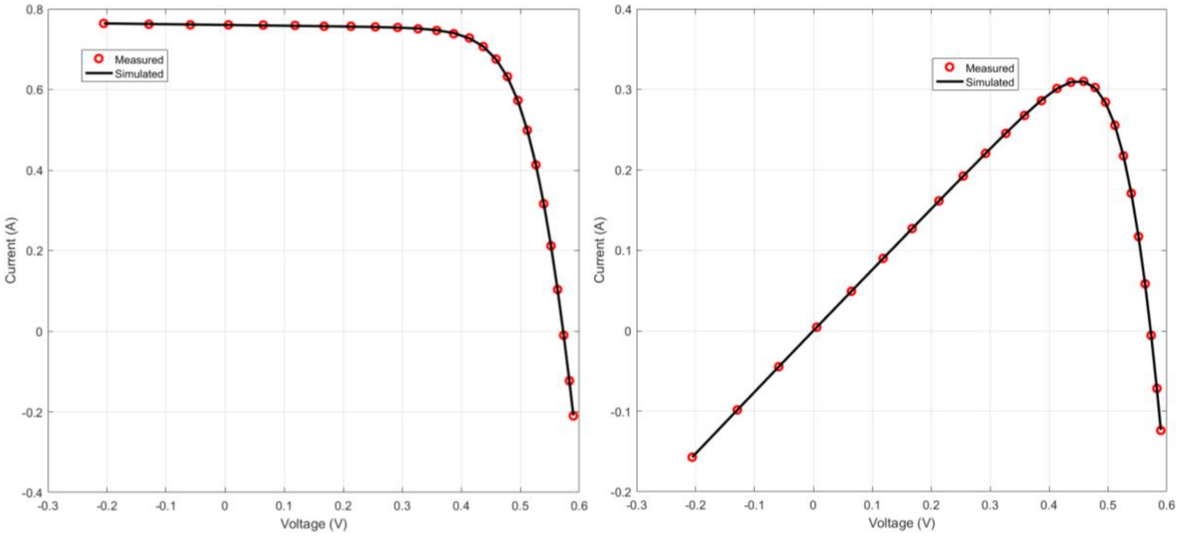


Figure 1. Comparison of algorithms on the convergence curve on the single-diode model.



(a)

(b)

Figure 2. The Fitting curve between the measured data and the simulated data is obtained by the EHRJAYA on the single-diode model: (a) Fitting curve of output current on the single-diode model, (b) Fitting curve of output power on the single-diode model.

5.3. *Experimental results of the double-diode model*

For the double-diode model, the parameters that need to be extracted become seven, both the complexity of the problem and the requirements for the algorithm are very high. The extracted seven parameters and their corresponding the RMSE are shown in Table 5. Similarly, the best RMSE is obtained by the EHRJAYA. This proves that the EHRJAYA still has superior performance on complex models.

From the convergence curve in **Figure 3**, it can be seen that the EHRJAYA has a faster convergence speed and higher level of performance on complex models.

Moreover, the correctness of the extracted 7 parameters is also verified. The fitted curve of the simulated and measured values is shown in **Figure 4**. As can be seen from the figure, the degree of fitting is very high and the extraction parameters are very accurate. This further proves that the algorithm also has superior performance in handling complex models.

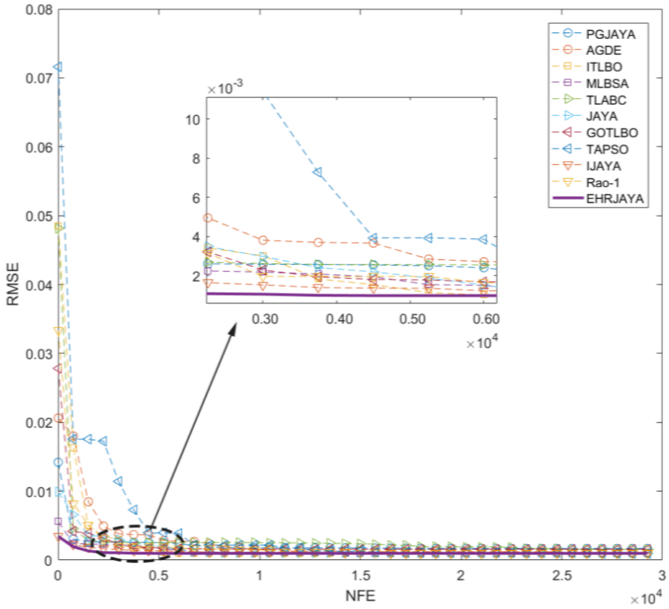
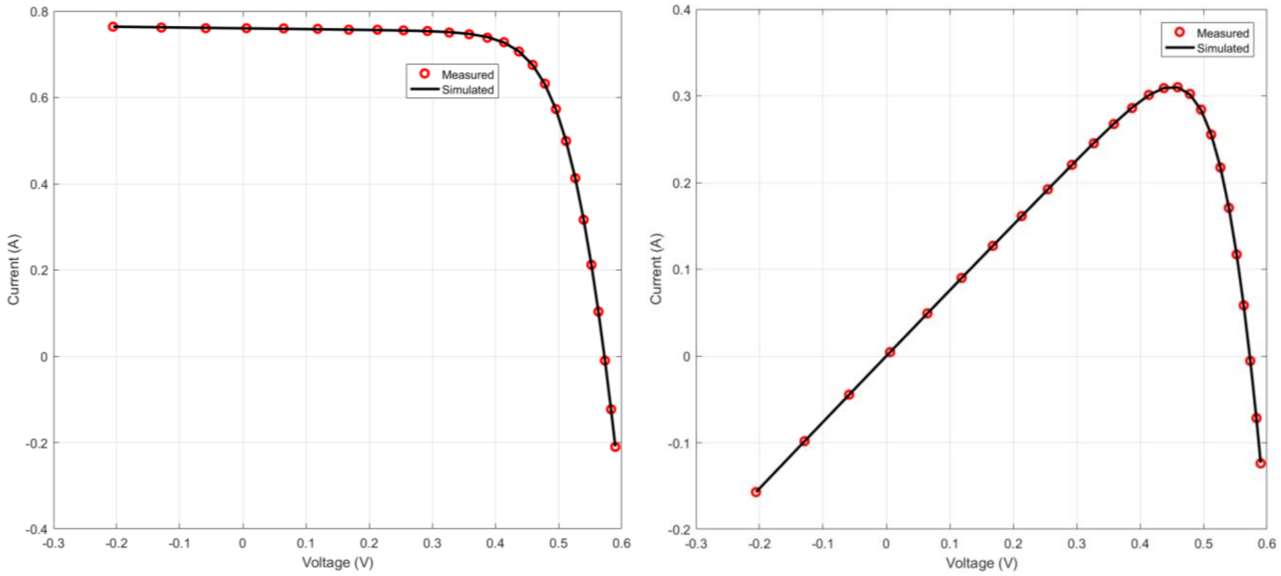


Figure 3. Comparison of algorithms on the convergence curve on the double-diode model.



(a) (b)

Figure 4. The Fitting curve between the measured data and the simulated data is obtained by the EHRJAYA on the double-diode model: (a) Fitting curve of output current on the double-diode model, (b) Fitting curve of output current on the double-diode model.

Table 5. Extracted parametric results on the double-diode model.

Algorithm	I_{pv}(A)	I_{sd_1}(μA)	R_S(Ω)	R_P(Ω)	n_1	I_{sd_2}(μA)	n_2	RMSE
EHRJAYA	7.60781258E-01	7.47538298E-01	3.67396247E-02	5.54786495E+01	1.99996900E+00	2.26166373E-07	1.45108745E+00	**9.82484851E-04**
PGJAYA	7.60782691E-01	9.92185064E-01	3.68703169E-02	5.61498543E+01	1.99983984E+00	1.98907570E-07	1.44043723E+00	9.82943724E-04
JAYA	7.60779768E-01	2.19514534E-01	3.67330644E-02	5.54364923E+01	1.44916774E+00	6.46927132E-07	1.94399658E+00	9.83141687E-04
AGDE	7.60464543E-01	3.09640419E-04	3.52271922E-02	6.42351712E+01	1.61543153E+00	4.32189133E-07	1.51111988E+00	9.83079356E-04
TLABC	7.60755291E-01	4.29452667E-01	3.67026694E-02	5.52382559E+01	1.83765086E+00	2.15082897E-07	1.44889303E+00	9.84565015E-04
MLBSA	7.60318211E-01	5.97174284E-01	3.36845757E-02	9.16732420E+01	1.54737246E+00	1.59162846E-07	1.93634296E+00	9.82682288E-04
TAPSO	7.60289000E-01	4.99352011E-01	3.46531370E-02	7.78089094E+01	1.52627417E+00	1.21692088E-10	1.75688634E+00	9.85014194E-04
GOTLBO	7.60464543E-01	3.09640419E-04	3.52271922E-02	6.42351712E+01	1.61543153E+00	4.32189133E-07	1.51111988E+00	9.84454849E-04
IJAYA	7.60416869E-01	2.32769739E-01	3.57427014E-02	6.87619507E+01	1.60252267E+00	2.22831049E-07	1.47085017E+00	9.95452170E-04
Rao-1	7.60865496E-01	3.24116226E-01	3.64840355E-02	5.21341610E+01	1.48396790E+00	6.23455807E-10	1.26524009E+00	9.84037665E-04
ITLBO	7.60289000E-01	4.99352011E-01	3.46531370E-02	7.78089094E+01	1.52627417E+00	1.21692088E-10	1.75688634E+00	9.83052011E-04

5.4. Experimental results of the Photowatt-PWP201

For the Photowatt-PWP201, this is a PV module model. The extracted parameters and corresponding RMSE are shown in Table 6. The best RMSE is obtained by algorithms other than TAPSO and IJAYA. The best RMSE is obtained by the EHRJAYA using 20,000 evaluations. This means that the EHRJAYA is still competitive when faced with PV module models.

The convergence curve is shown in Figure 5. The proposed EHRJAYA still converges very fast, and it is worth noting that the TLABC also converges very fast.

In order to verify the accuracy of the extracted parameters, Figure 6 shows the fitted curve of the simulated and measured values. It can be seen from Figure 6 that the fitting degree of the curve is very high, which means that the EHRJAYA also has very high accuracy in the parameter extraction of the PV module model.

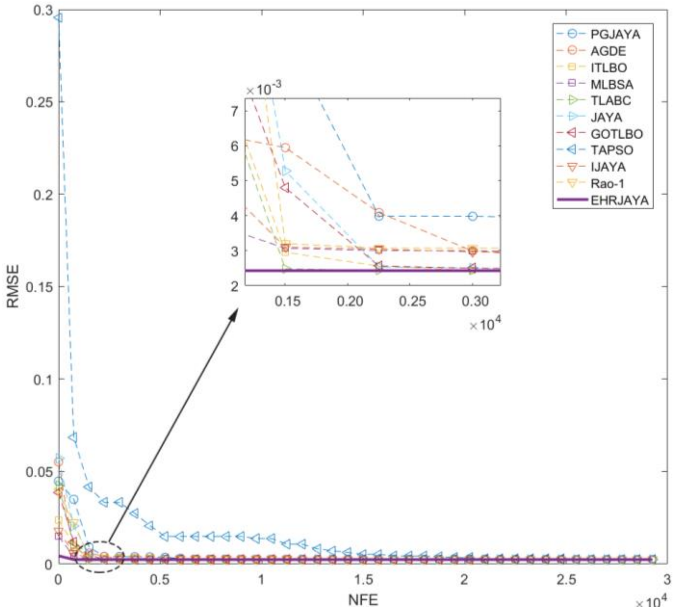
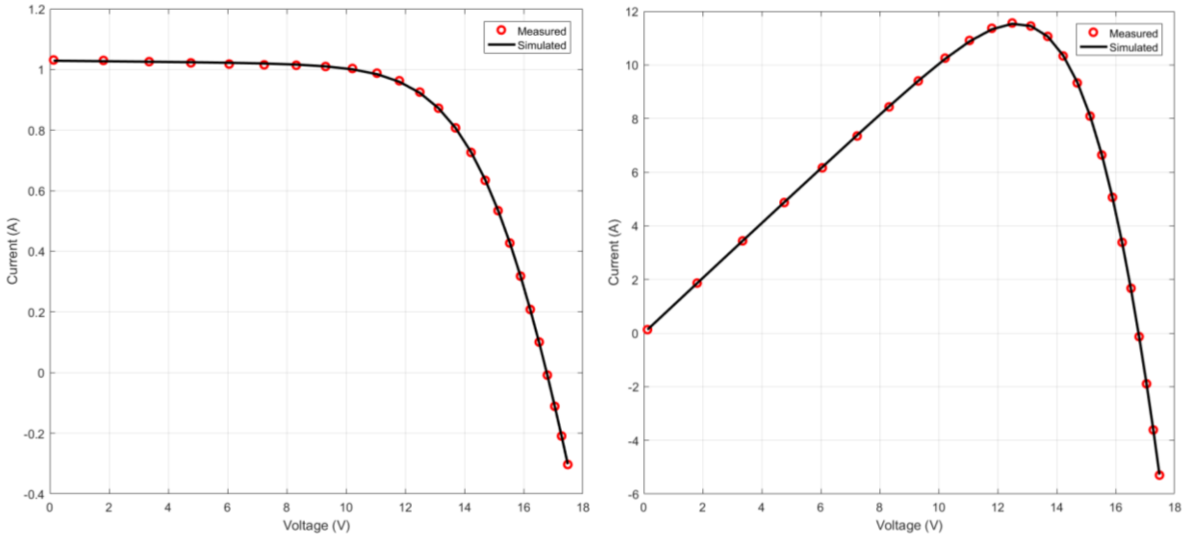


Figure 5. Comparison of algorithms on the convergence curve on the Photowatt-PWP201.

Table 6. Extracted parametric results on the Photowatt-PWP201

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}}$($\boldsymbol{\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	$\boldsymbol{n}$	RMSE
EHRJAYA	1.03051430E+00	3.48226293E-00	1.20127100E+00	9.81982222E+02	4.86428349E+01	**2.42507487E-03**
PGJAYA	1.03051430E+00	3.48226304E-00	1.20127100E+00	9.81982241E+02	4.86428350E+01	**2.42507487E-03**
JAYA	1.03051430E+00	3.48226296E-00	1.20127101E+00	9.81982252E+02	4.86428349E+01	**2.42507487E-03**
AGDE	1.03051430E+00	3.48226270E-00	1.20127102E+00	9.81982290E+02	4.86428346E+01	**2.42507487E-03**
TLABC	1.03051430E+00	3.48226291E-00	1.20127101E+00	9.81982283E+02	4.86428349E+01	**2.42507487E-03**
MLBSA	1.03072874E+00	3.23284848E-00	1.20878887E+00	9.23249849E+02	4.83597087E+01	**2.42507487E-03**
TAPSO	1.03051430E+00	3.48226322E-00	1.20127100E+00	9.81982348E+02	4.86428352E+01	2.43611127E-03
GOTLBO	1.03054073E+00	3.46271957E-00	1.20175352E+00	9.76177979E+02	4.86213220E+01	**2.42507487E-03**
IJAYA	1.03050146E+00	3.49374291E-00	1.20088384E+00	9.84565626E+02	4.86554782E+01	2.42621415E-03
Rao-1	1.03051430E+00	3.48226270E-00	1.20127102E+00	9.81982290E+02	4.86428346E+01	**2.42507487E-03**
ITLBO	1.03051430E+00	3.48226270E-00	1.20127102E+00	9.81982290E+02	4.86428346E+01	**2.42507487E-03**



(a) (b)

Figure 6. The Fitting curve between the measured data and the simulated data is obtained by the EHRJAYA on the Photowatt-PWP201: (a) Fitting curve of output current on the Photowatt-PWP201, (b) Fitting curve of output power on the Photowatt-PWP201.

5.5. Experimental results of the STM6-40/36

Table 7. Extracted parametric results on the STM6-40/36.

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}}$($\boldsymbol{\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	$\boldsymbol{n}$	RMSE
EHRJAYA	1.66390478E+00	1.73865695E-00	4.27377121E-03	1.59282944E+01	1.52030293E+00	**1.72981371E-03**
PGJAYA	1.66390478E+00	1.73865682E-00	4.27377142E-03	1.59282938E+01	1.52030292E+00	**1.72981371E-03**
JAYA	1.66390478E+00	1.73865689E-00	4.27377127E-03	1.59282940E+01	1.52030292E+00	**1.72981371E-03**
AGDE	1.66193319E+00	3.33258320E-00	2.23208753E-03	2.07914452E+01	1.59536111E+00	**1.72981371E-03**
TLABC	1.66391959E+00	1.73165475E-00	4.28565149E-03	1.58999591E+01	1.51986178E+00	1.72985308E-03
MLBSA	1.66161097E+00	3.97763969E-00	1.66711256E-03	2.27245278E+01	1.61707253E+00	1.73031964E-03
TAPSO	1.66205955E+00	3.16226227E-00	2.35677203E-03	2.03848642E+01	1.58894553E+00	5.92497629E-03
GOTLBO	1.66193319E+00	3.33258320E-00	2.23208753E-03	2.07914452E+01	1.59536111E+00	1.72981397E-03
IJAYA	1.66225636E+00	4.63423560E-00	6.70300559E-04	2.10617576E+01	1.63649780E+00	2.59459752E-03
Rao-1	1.66199771E+00	3.40742177E-00	2.06225519E-03	2.06586741E+01	1.59812395E+00	1.72981697E-03
ITLBO	1.66398813E+00	1.70380605E-00	4.34302882E-03	1.58195363E+01	1.51808366E+00	**1.72981371E-03**

For the STM6-40/36, this is a type of PV module model, which also has a certain complexity. The extracted parameters and the corresponding best RMSE are shown in **Table 7**. In this model, the best RMSE is only obtained by the EHRJAYA, PGJAYA, JAYA, AGDE, and ITLBO.

The convergence curve of this model is shown in **Figure 7**. It can be seen intuitively in the figure that the EHRJAYA performs better than other comparison algorithms.

The extracted parameters need to be further verified for correctness. The fitting curve of the simulated value and the measured value is shown in **Figure 8**. Similarly, the fit performed very well, so the extracted parameters are fairly accurate.

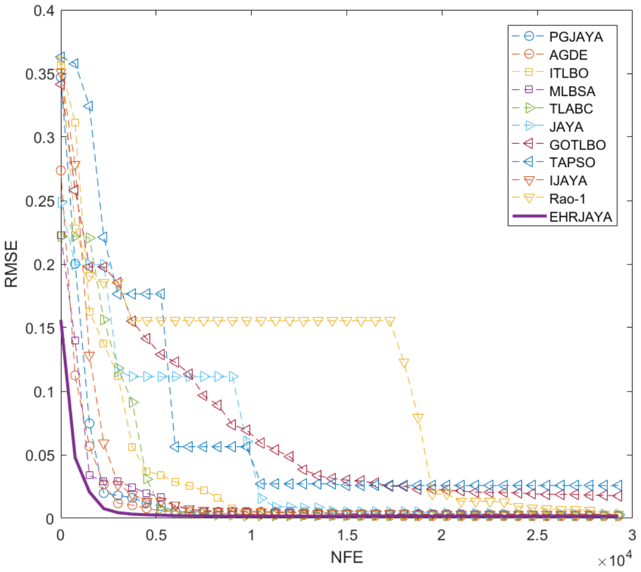
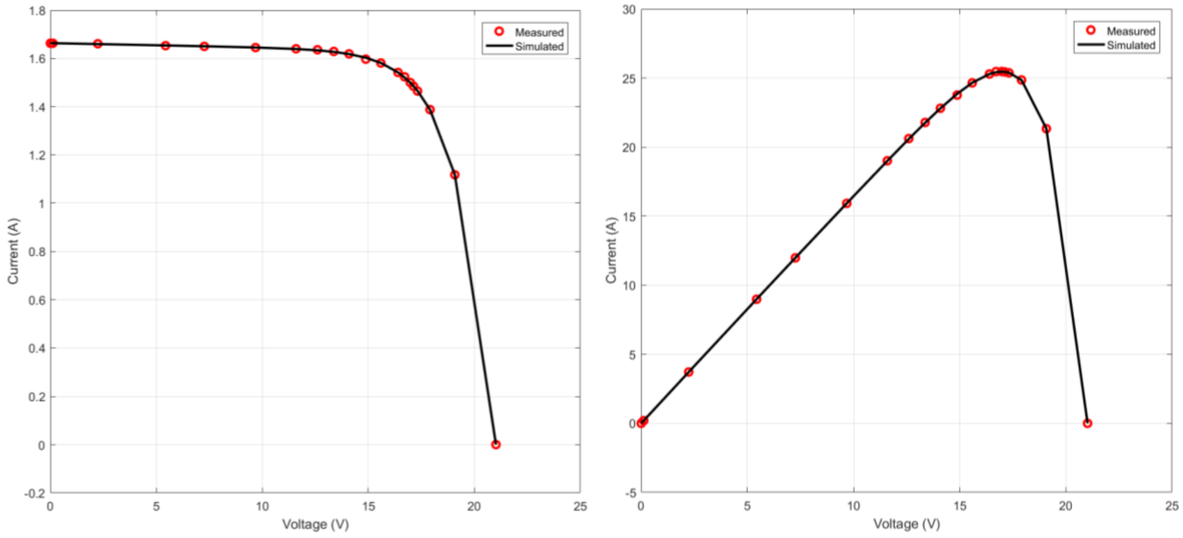


Figure 7. Comparison of algorithms on the convergence curve on the STM6-40/36.



(a) (b)

Figure 8. The Fitting curve between the measured data and the simulated data is obtained by the EHRJAYA on the STM6-40/36: (a) Fitting curve of output current on the STM6-40/36, (b) Fitting curve of output power on the STM6-40/36.

5.6. Experimental results of the STP6-120/36

For the STP6-120/36, the extracted parameters and corresponding RMSE are shown in **Table 8**. In this model, the best RMSE is obtained only by EHRJAYA, PGJAYA, JAYA, AGDE, Rao-1 and ITLBO.

The convergence curve of this model is shown in **Figure 9**. The EHRJAYA shows absolute advantages compared with other contrasting algorithms.

The fitting curves of output current and power are shown in **Figure 10**. Similarly, the simulated values are in good agreement with the measured values, which means the fit is very good.

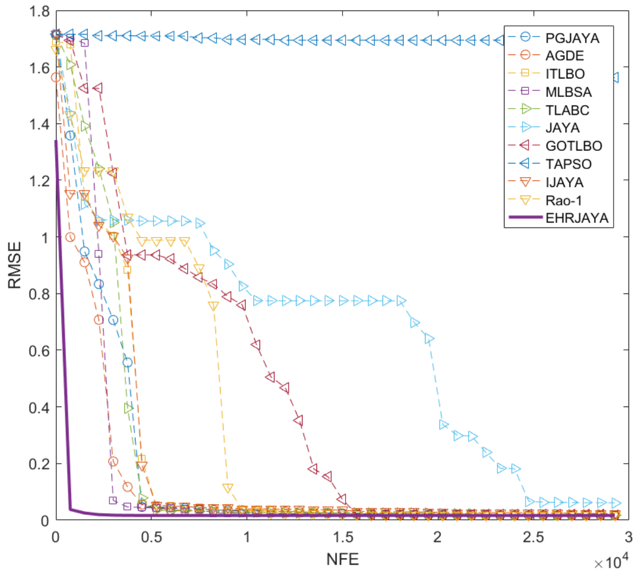
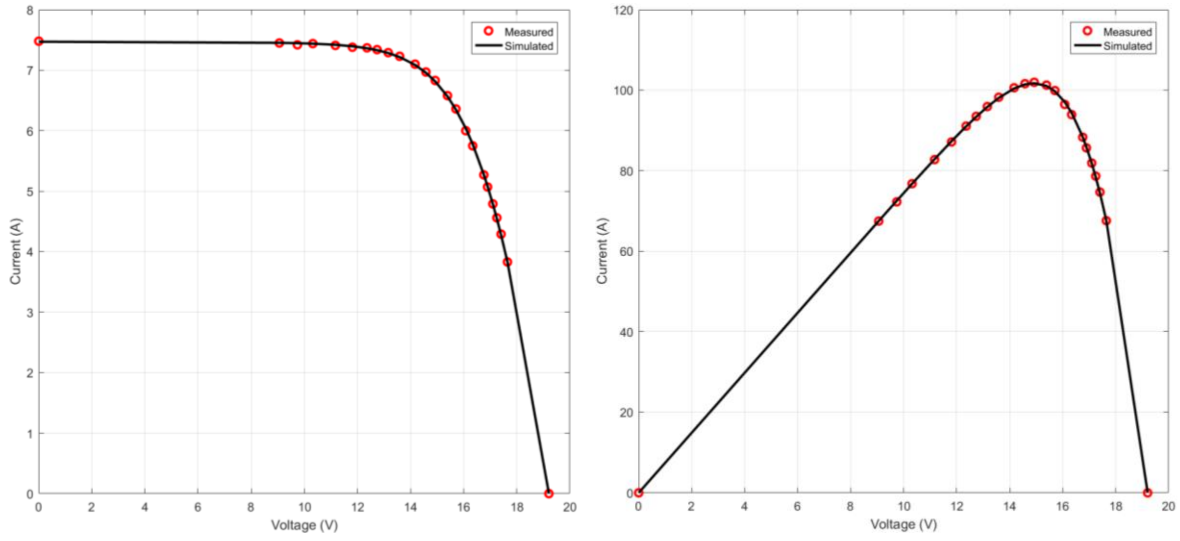


Figure 9. Comparison of algorithms on the convergence curve on the STP6-120/36.



(a) (b)

Figure 10. The Fitting curve between the measured data and the simulated data is obtained by the EHRJAYA on the STP6-120/36: (a) Fitting curve of output current on the STP6-120/36, (b) Fitting curve of output power on the STP6-120/36.

Table 8. Extracted parametric results on the STP6-120/36.

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}}$($\boldsymbol{\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	$\boldsymbol{n}$	RMSE
EHRJAYA	7.47252991E+00	2.33499502E-00	4.45946346E-03	2.22199074E+02	1.26010347E+00	**1.66006031E-02**
PGJAYA	7.47252991E+00	2.33499531E-00	4.59463455E-03	2.22199147E+01	1.26010349E+00	**1.66006031E-02**
JAYA	7.47252992E+00	2.33499502E-00	4.59463460E-03	2.22199062E+01	1.26010348E+00	**1.66006031E-02**
AGDE	7.48228394E+00	6.16278443E-00	4.09192670E-03	1.23156031E+03	1.34692010E+00	**1.66006031E-02**
TLABC	7.47290405E+00	2.36521524E-00	4.58821357E-03	2.21050016E+01	1.26118315E+00	1.66021524E-02
MLBSA	7.47050390E+00	4.14318493E-00	4.31875565E-03	3.79926319E+02	1.31008048E+00	1.66318539E-02
TAPSO	7.47290405E+00	2.36521524E-00	4.58821357E-03	2.21050016E+01	1.26118315E+00	7.94391655E-02
GOTLBO	7.46321879E+00	3.29196793E-00	4.42563991E-03	9.34597784E+02	1.28948549E+00	1.66009776E-02
IJAYA	7.47890441E+00	7.31117235E-00	3.99767796E-03	1.44316136E+03	1.36371602E+00	1.69032348E-02
Rao-1	7.48759808E+00	5.44470060E-00	4.09637821E-03	2.34570146E+01	1.33515922E+00	**1.66006031E-02**
ITLBO	7.46316616E+00	3.37366836E-00	4.40676487E-03	6.31085686E+02	1.29160745E+00	**1.66006031E-02**

5.7. Statistical analysis

The superiority of the algorithm cannot be proved from the above-mentioned aspect of obtaining the best RMSE alone. Therefore, it is necessary to analyze the RMSE obtained by all algorithms. As shown in **Table 9**, the RMSE includes the best value, the worst value, the mean value, and the standard deviation in 30 experiments, and the Wilcoxon Signed Ranks test is performed to judge the superiority of the algorithm. From the data in the table, the following conclusions can be drawn:

- For the single-diode model (SDM), the EHRJAYA, PGJAYA, JAYA, AGDE and ITBLO all perform very well in four aspects of RMSE, so from the p-value, the results of these five algorithms are very similar. However, as can be seen from the value of the standard deviation (1.10513598E-17), EHRJAYA is more stable than the other four algorithms. It is worth noting that the EHRJAYA is very different from the other algorithms except these four algorithms. From the ranking, it can be seen that the EHRJAYA shows very strong competitiveness, and among these comparison algorithms, it ranks first with the other four. In summary, the proposed EHRJAYA performs better than other algorithms in SDM.
- For the double-diode model (DDM), only the EHRJAYA and ITBLO perform relatively well. However, from the value of the standard deviation (1.60552555E-06), EHRJAYA is more stable than ITBLO. It can be seen from the p-value that the results of the EHRJAYA are different from those of the algorithms except the ITLBO. On the other hand, it can be seen from the ranking that the EHRJAYA ranks first among all algorithms. Therefore, in summary, the EHRJAYA performs best compared to other algorithms on the double-diode model.
- For the Photowatt-PWP201 (PWP), only the EHRJAYA, JAYA, AGDE and ITBLO perform well in four aspects of RMSE. However, similarly, from the value of the standard deviation (1.27666432E-17), EHRJAYA is more stable than JAYA, AGDE and ITBLO. From the p-value, it can be seen that the results of the EHRJAYA are different from other comparison algorithms except these three algorithms, because these three algorithms show better performance, and the difference in results is not obvious. From the ranking, the EHRJAYA is tied for the first place with these three algorithms among all the algorithms. Therefore, the proposed EHRJAYA performs best compared to other algorithms in PWP.
- For the STM6-40/36 (STM6), the EHRJAYA performs the best in terms of the four aspects (including Best, Worst, Mean, Std) of RMSE, and it ranks first among all algorithms in terms of ranking. It can be seen from the p-value that the results of the EHRJAYA are

different from that of most algorithms. Therefore, in this model, the EHRJAYA performs the best compared to other comparison algorithms.

- For the STP6-120/36 (STP6), the EHRJAYA performs the best in the four aspects (including Best, Worst, Mean, Std) of RMSE at the same time. From the ranking point of view, the EHRJAYA is better than other algorithms, ranking first among all algorithms. The results of the EHRJAYA are different from the results of all the compared algorithms as can be seen from the p-values. Therefore, in this model, the EHRJAYA is still the best performer.

Table 9. Statistical results of RMSE of different algorithms in different models.

Model	Algorithm	RMSE				Wilcoxon Signed Ranks test				
		Best	Worst	Mean	Std	R^+	R^-	*p*-value	Ranking	Sig.
SDM	EHRJAYA	**9.86021878E-04**	**9.86021878E-04**	**9.86021878E-04**	**1.10513598E-17**				**4.1167**	
	PGJAYA	**9.86021878E-04**	9.86021893E-04	9.86021878E-04	2.80277724E-12	232.5	232.5	≥ 0.2	**4.1167**	≈
	JAYA	**9.86021878E-04**	**9.86021878E-04**	**9.86021878E-04**	**1.91861852E-17**	232.5	232.5	≥ 0.2	**4.1167**	≈
	AGDE	**9.86021878E-04**	**9.86021878E-04**	**9.86021878E-04**	**4.30733413E-17**	232.5	232.5	≥ 0.2	**4.1167**	≈
	TLABC	**9.86021878E-04**	1.40946520E-03	1.04578461E-03	1.03870962E-04	447.0	18.0	1.92E-06	8.1667	+
	MLBSA	**9.86021878E-04**	1.02012387E-03	9.87302147E-04	6.22172770E-06	412.5	52.5	1.73E-06	6.35	+
	TAPSO	9.86023131E-04	2.18902086E-03	1.15246672E-03	3.14762098E-04	465.0	0.0	1.73E-06	9.6667	+
	GOTLBO	**9.86021878E-04**	1.70888890E-03	1.03074464E-03	1.37930201E-04	367.0	68.0	1.92E-06	6.4167	+
	IJAYA	9.89877058E-04	2.13601446E-03	1.22589718E-03	2.51763833E-04	465.0	0.0	1.73E-06	10.3	+
	Rao-1	**9.86021878E-04**	1.02442569E-03	9.87303865E-04	7.01119985E-06	262.0	203.0	1.67E-04	4.5167	+
	ITLBO	**9.86021878E-04**	9.86021878E-04	9.86021878E-04	**2.53140932E-17**	232.5	232.5	≥ 0.2	**4.1167**	
DDM	EHRJAYA	**9.82489448E-04**	**9.87630370E-04**	**9.84816894E-04**	**1.60552555E-06**				**2.3333**	
	PGJAYA	9.82943724E-04	2.31019435E-03	1.12878237E-03	3.25009725E-04	454.0	11.0	2.96E-03	6.4333	+
	JAYA	9.83141687E-04	1.11973139E-03	9.91288054E-04	2.47521608E-05	362.0	73.0	3.49E-03	4.1	+
	AGDE	9.83079356E-04	1.41337194E-03	1.03070996E-03	1.06602868E-04	399.0	66.0	4.11E-03	4.1	+
	TLABC	9.84565015E-04	1.66517270E-03	1.10590659E-03	1.53534053E-04	462.0	3.0	3.11E-05	6.9333	+
	MLBSA	9.82682288E-04	1.31622002E-03	1.00553814E-03	6.15956153E-05	418.0	47.0	7.71E-04	4.9333	+
	TAPSO	9.85014194E-04	2.02980721E-03	1.35571098E-03	3.02282845E-04	462.0	3.0	1.73E-06	9.0667	+
	GOTLBO	9.84454849E-04	2.51787981E-03	1.22496555E-03	3.98438216E-04	464.0	1.0	3.18E-06	7.1667	+
	IJAYA	9.95452170E-04	2.37365389E-03	1.48027649E-03	3.46315975E-04	465.0	0.0	1.73E-06	9.9	+
	Rao-1	9.84037665E-04	2.99488092E-03	1.24774598E-03	4.17692613E-04	456.0	9.0	2.35E-06	7.3333	+
	ITLBO	9.83052011E-04	9.87669922E-04	9.85556248E-04	9.16514133E-06	354.0	81.0	3.05E-03	3.7	+
PWP	EHRJAYA	**2.42507487E-03**	**2.42507487E-03**	**2.42507487E-03**	**1.27666432E-17**				**4.1667**	
	PGJAYA	**2.42507487E-03**	4.19348570E-03	2.54627251E-03	3.74503998E-04	330.0	105.0	1.05E-04	6.0667	+
	JAYA	**2.42507487E-03**	**2.42507487E-03**	**2.42507487E-03**	**1.51268278E-17**	232.5	232.5	≥ 0.2	**4.1667**	≈
	AGDE	**2.42507487E-03**	**2.42507487E-03**	**2.42507487E-03**	**1.89117715E-17**	232.5	232.5	≥ 0.2	**4.1667**	≈
	TLABC	**2.42507487E-03**	2.75840377E-03	2.44463842E-03	6.26870189E-05	426.0	39.0	1.73E-06	6.75	+
	MLBSA	**2.42507487E-03**	2.50656518E-03	2.42962914E-03	1.66464730E-05	338.5	126.5	1.73E-06	5.3833	+
	TAPSO	2.43611127E-03	1.21730621E-02	3.68602259E-03	2.23096076E-03	465.0	0.0	1.73E-06	10.5	+
	GOTLBO	2.42507487E-03	2.55533866E-03	2.43878597E-03	2.91304642E-05	397.0	68.0	2.01E-06	6.4333	+
	IJAYA	2.42621415E-03	4.57482543E-03	2.56071055E-03	3.86826839E-04	465.0	0.0	1.73E-06	9.8	+
	Rao-1	**2.42507487E-03**	4.45239447E-03	2.49265219E-03	3.70136226E-04	232.0	203.0	6.32E-04	4.4	≈
	ITLBO	**2.42507487E-03**	**2.42507487E-03**	**2.42507487E-03**	**1.35311403E-17**	232.5	232.5	≥ 0.2	**4.1667**	≈
STM6	EHRJAYA	**1.72981371E-03**	**1.72981371E-03**	**1.72981371E-03**	**7.43618690E-18**				**1.9833**	
	PGJAYA	**1.72981371E-03**	2.87306133E-03	1.86924081E-03	2.84969156E-04	426.0	39.0	1.73E-06	3.9	+
	JAYA	**1.72981371E-03**	1.94536126E-01	1.32040335E-02	3.63281800E-02	447.0	18.0	4.73E-06	6.2167	+
	AGDE	**1.72981371E-03**	2.20591920E-03	1.74954872E-03	8.73082253E-05	407.5	27.5	1.73E-06	3.3833	+
	TLABC	1.72985308E-03	5.82080492E-03	2.51382277E-03	7.43167719E-04	465.0	0.0	1.73E-06	6.4333	+
	MLBSA	1.73031964E-03	2.61729089E-03	2.08481135E-03	2.26380918E-04	465.0	0.0	1.73E-06	5.7667	+
	TAPSO	5.92497629E-03	3.25889564E-01	5.10311809E-02	5.48221079E-02	465.0	0.0	1.73E-06	10.6667	+
	GOTLBO	1.72981397E-03	1.20576548E-01	1.03111883E-02	2.20818747E-02	465.0	0.0	1.73E-06	8.4833	+
	IJAYA	2.59459752E-03	5.33629220E-03	3.54989942E-03	5.77839624E-04	465.0	0.0	1.73E-06	8.3333	+
	Rao-1	1.72981697E-03	6.59098520E-02	1.77343532E-02	2.32076442E-02	465.0	0.0	1.73E-06	8.6333	+
	ITLBO	**1.72981371E-03**	1.76726899E-03	1.73111707E-03	6.83427314E-06	259.0	175.5	≥ 0.2	2.2	≈
STP6	EHRJAYA	**1.66006031E-02**	**1.66006031E-02**	**1.66006031E-02**	**1.77966291E-16**				**2.0333**	
	PGJAYA	**1.66006031E-02**	3.38972496E-02	1.77225050E-02	3.60388252E-03	349.5	85.5	1.73E-06	3.7167	+
	JAYA	**1.66006031E-02**	9.48164230E-01	1.09094120E-01	2.75490691E-01	407.5	27.5	3.14E-05	5.25	+
	AGDE	**1.66006031E-02**	1.72180964E-02	1.66286480E-02	1.13010775E-04	460.0	5.0	1.73E-06	4	+
	TLABC	1.66021524E-02	3.04170354E-02	2.13527483E-02	3.92803569E-03	465.0	0.0	1.73E-06	7.6333	+
	MLBSA	1.66318539E-02	4.15852992E-02	1.83916953E-02	4.45656162E-03	465.0	0.0	1.73E-06	6.7333	+
	TAPSO	7.94391655E-02	1.47184780E+00	8.62747692E-01	3.96609968E-01	465.0	0.0	1.73E-06	10.9	+
	GOTLBO	1.66009776E-02	6.22455148E-01	6.38093633E-02	1.31746769E-01	465.0	0.0	1.73E-06	8.5333	+
	IJAYA	1.69032348E-02	4.04018974E-02	2.86892099E-02	7.06230859E-03	465.0	0.0	1.73E-06	8.7333	+
	Rao-1	**1.66006031E-02**	7.74018520E-01	6.85545719E-02	1.90221137E-01	463.5	1.5	1.73E-06	6.1667	+
	ITLBO	**1.66006031E-02**	1.66071028E-02	1.66008751E-02	1.20890475E-06	289.5	175.5	4.26E-03	2.3	+

In conclusion, after the statistical analysis of the five models, the EHRJAYA is better than the comparison algorithm in the comprehensive performance.

5.8. Comparison of the EHRJAYA with other well-known improved algorithms

To further prove the superiority of the EHRJAYA algorithm, in this section, the EHRJAYA algorithm will be compared with some well-known improved algorithms, including self-adaptive teaching-learning-based optimization (SATLBO), chaotic whale optimization algorithm (CWOA), hybrid differential evolution with whale optimization algorithm (DE/WOA), opposition-based whale optimization algorithm (OBWOA), hybridizing cuckoo search algorithm with biogeography-based optimization (BHCS), flexible particle swarm optimization algorithm (FPSO), improved Lozi map based chaotic optimization Algorithm (ILCOA), backtracking search algorithm with reusing differential vectors (BSARDVs), similarity-guided differential evolution (SGDE), classified perturbation mutation based particle swarm optimization algorithm (CPMPSO), niche-based particle swarm optimization in parallel computing architecture (NPSOPC), backtracking search algorithm with competitive learning (CBSA), comprehensive learning JAYA (CLJAYA), hybrid adaptive teaching–learning-based optimization and differential evolution (ATLDE), enhanced JAYA algorithm (EJAYA), enhanced adaptive butterfly optimization algorithm (EABOA), shuffled frog leaping with memory pool (SFLBS), reinforcement learning-based differential evolution (RLDE), improved equilibrium optimizer (IEO), modified teaching learning based optimization (MTLBO), modified Rao-1 algorithm (MRao-1). From **Table 10** to **Table 14** the following conclusions can be drawn:

- For the single-diode model, all algorithms except the NPSOPC can obtain the best RMSE, in addition, only the proposed EHRJAYA consumes the least evaluation times, only 20,000 times.
- For the double-diode model, the best RMSE is obtained by the DE/WOA, CPMPSO, CBSA, ATLDE, EJAYA, RLDE and EHRJAYA, but still the EHRJAYA consumes the least computing resources, and the evaluation times are 20,000.
- For the Photowatt-PWP201, the best RMSE is obtained by the EHRJAYA and other well-known algorithms, however, the EHRJAYA uses fewer evaluations than other algorithms.
- For the STM6-40/36, there are certain challenges in this model. Only half of the algorithms achieved the best RMSE, including BHCS, ELBA, ATLDE, EJAYA, RLDE, IEO and EHRJAYA. It is worth noting that among these algorithms, the proposed EHRJAYA is second only to ELBA in the number of evaluations used. From the no free lunch theorems [54], this is acceptable, and there is currently no single algorithm that can solve all problems perfectly.
- For the STP6-120/36, the best RMSEs are obtained for all algorithms except the CWOA. Similarly, still only the EHRJAYA uses the least computing resources, and the number of evaluations is only 26,000.

Table 10. Comparison of extracted parameters between the EHRJAYA and other mature algorithms on the single-diode model.

Algorithm	I_{pv}(A)	I_{sd}(μA)	R_S(Ω)	R_P(Ω)	n	RMSE	NFES
SATLBO (2017) [55]	0.7608	0.3232	0.0363	53.7295	1.4812	9.8602E-04	50,000
CWOA (2017) [56]	0.76077	0.3239	0.03636	53.742465	1.4812	9.8602E-04	50,000
DE/WOA (2018) [57]	0.760776	0.323021	0.036377	53.718524	1.481184	9.8602E-04	50,000
OBWOA (2018) [12]	0.76077	0.3232	0.0363	53.6836	1.5208	9.8602E-04	1,500,000

Continued on next page

Algorithm	I_{pv}(A)	I_{sd}(μA)	R_S(Ω)	R_P(Ω)	n	RMSE	NFES
BHCS (2019) [58]	0.76078	0.32302	0.03638	53.71852	1.48118	9.8602E-04	50,000
FPSO (2019) [59]	0.76077552	0.323020	0.036370	53.718520	1.48110817	9.8602E-04	NA
ILCOA (2019) [60]	0.760775	0.323021	0.036377	53.718679	1.481108	9.8602E-04	10,000*NP
BSARDVs (2020) [61]	0.760776	0.323021	0.036377	53.718520	1.481184	9.8602E-04	25,000
SGDE (2020) [13]	0.76078	0.32302	0.036377	53.71853	1.481184	9.8602E-04	50,000
CPMPSO (2020) [62]	0.760776	0.323021	0.036377	53.71852	1.481184	9.8602E-04	50,000
NPSOPC (2020) [63]	0.7608	0.3325	0.03639	53.7583	1.4814	9.8856E-04	NA
CBSA (2020) [64]	0.760776	0.323021	0.036377	53.71852	1.481184	9.8602E-04	50,000
ATLDE (2020) [18]	0.76077553	0.32302082	0.03637712	53.71852699	1.48118359	9.8602E-04	30,000
EJAYA (2021) [43]	0.76078	0.32302	0.03638	53.71852	1.48118	9.8602E-04	30,000
EABOA (2021) [65]	0.760771077	0.322929	0.036379593	53.76600144	1.481153457	9.8602E-04	50,000
SFLBS (2021) [66]	0.76078	0.323021	0.03638	53.7185	1.481184	9.8602E-04	60,000
RLDE (2021) [67]	0.7608	0.3231	0.0364	53.7185	1.4812	9.8602E-04	30,000
EHRJAYA	0.7607529	0.3297826	0.036291482	53.928534	1.48328376	9.8602E-04	**20,000**

Table 11. Comparison of extracted parameters between the EHRJAYA and other mature algorithms on the double-diode model.

Algorithm	I_{pv}(A)	I_{sd_1}(μA)	R_S(Ω)	R_P(Ω)	n_1	I_{sd_2}(μA)	n_2	RMSE	NFES
SATLBO (2017) [55]	0.7608	0.2509	0.0366	55.1170	1.4598	0.5454	1.9994	9.8280E-04	50,000
CWOA (2017) [56]	0.76077	0.24150	0.03666	55.20160	1.45651	0.60000	1.98990	9.8272E-04	50,000
DE/WOA (2018) [57]	0.760781	0.225974	0.036740	55.485437	1.451017	0.749346	2.000000	9.8248E-04	50,000
OBWOA (2018) [12]	0.76076	0.22990	0.03671	55.3990	1.49154	0.61956	2.000000	9.8251E-04	1,500,000
BHCS (2019) [58]	0.76078	0.74935	0.03674	55.48544	2.00000	0.22597	1.45102	9.8249E-04	50,000
FPSO (2019) [59]	0.76078	0.22731	0.036737	55.39230	1.45160	0.72786	1.99969	9.8253E-04	NA
ILCOA (2019) [60]	0.76078	0.22601	0.036739	55.5320	1.45101	0.74921	2.00000	9.8257E-04	10,000*NP
SGDE (2020) [13]	0.76079	0.28070	0.036480	55.3667	1.46966	0.24996	1.93228	9.8441E-04	50,000
CLJAYA (2020) [68]	0.76078	0.226051	0.03674	55.48599	1.45105	0.74876	1.99999	9.8249E-04	20,000
CPMPSO (2020) [62]	0.76078	0.74935	0.3674	55.48544	2	0.22597	1.45102	9.8248E-04	50,000
NPSOPC (2020) [63]	0.76078	0.25093	0.3663	55.117	1.45982	0.545418	1.99941	9.8208E-04	NA
CBSA (2020) [64]	0.76078	0.22597	0.3674	55.48544	1.451017	0.74935	2	9.8248E-04	50,000
ATLDE (2020) [18]	0.76078	0.22597	0.036740	55.48544744	1.451016	0.74934885	2.00000000	9.8248E-04	30,000
EJAYA (2021) [43]	0.76078	0.22597	0.03674	55.48509	1.45102	0.74934	2	9.8248E-04	30,000
EABOA (2021) [65]	0.76082	0.25072	0.03662	55.3660129	1.459884	0.72069	1.99997318	9.8607E-04	50,000
RLDE (2021) [67]	0.7608	0.226	0.0367	55.4847	2	0.7492	1.451	9.8248E-04	30,000
EHRJAYA	0.7608	0.1639	0.03648	54.20251	1.841254	0.2801918	1.4698422	9.8248E-04	**20,000**

Table 12. Comparison of extracted parameters between the EHRJAYA and other mature algorithms on the Photowatt-PWP201.

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}(\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	n	RMSE	NFES
CLJAYA (2020) **[68]**	1.030514	3.48226280	1.201271	981.982279	48.64283	2.4251E-03	30,000
CBSA (2020) **[64]**	1.0275389	4.747459	1.340999	1087.81738	49.927517	2.4251E-03	25000
IEO (2020) **[69]**	1.030514254	3.48	1.201269	981.9956	48.64292	2.4251E-03	1,500,000
EJAYA (2021) **[43]**	1.03051	3.48226	1.20127	981.98235	48.64283	2.4251E-03	30,000
EABOA (2021) **[65]**	1.0304416	3.5084	1.200630203	991.9830745	48.67132719	2.4251E-03	50,000
SFLBS (2021) **[66]**	1.030514	3.48226	1.201271	981.9804	48.6428	2.4251E-03	60,000
MTLBO (2021) **[70]**	1.0305143	3.4823	1.201271	981.9823732	48.6428349	2.4251E-03	50,000
MRao-1(2021) **[71]**	1.030514	3.4823	1.201271	981.9821	48.64131	2.4251E-03	50,000
EHRJAYA	1.0305143	3.482263	1.201271	981.982523	48.6428353	2.4251E-03	**20,000**

Table 13. Comparison of extracted parameters between the EHRJAYA and other mature algorithms on the STM6-40/36.

Algorithm	I_{pv}(A)	I_{sd}(μA)	R_S(Ω)	R_P(Ω)	n	RMSE	NFES
CWOA (2017) [56]	1.7	1.6338	0.0050	15.4	1.5	1.8000E-03	50,000
HFAPS (2018) [72]	1.6663	1.0703	0.24849	490.03	53.016	1.9700E-03	50,000
OBWOA(2018) [12]	1.6642	1.65025	0.0044	15.5299	1.51424	1.7530E-03	1,500,000
BHCS (2019) [58]	1.66390	1.73866	0.00427	15.92829	1.52030	1.7298E-03	50,000
FPSO (2019) [59]	1.2323	7.4732	0.0049	9.6889	1.2086	1.3000E-03	NA
ILCOA (2019) [60]	1.2001	7.4812	0.0049	9.6991	1.2067	1.6932E-02	10,000*NP
ELBA (2020) [73]	1.663905	1.738657	0.004274	15.928294	1.520305	1.7298E-03	15,000
ATLDE (2020) [18]	1.66390478	1.73865697	0.00427377	15.92829439	1.52030293	1.7298E-03	30,000
EJAYA (2021) [43]	1.6639	1.73866	0.00427	15.92829	1.5203	1.7298E-03	30,000
RLDE (2021) [67]	1.6639	1.7387	0.00427	15.9283	1.5203	1.7298E-03	30,000
IEO (2020) [69]	1.663904802	1.74	0.004274	15.92827	1.520303	1.7298E-03	1,500,000
EHRJAYA	1.663904	1.738619	0.00427383	15.92811	1.5203	1.7298E-03	**24,000**

Table 14. Comparison of extracted parameters between the EHRJAYA and other mature algorithms on the STP6-120/36.

Algorithm	$\boldsymbol{I_{pv}}$(A)	$\boldsymbol{I_{sd}}$($\boldsymbol{\mu}$A)	$\boldsymbol{R_S}$(Ω)	$\boldsymbol{R_P}$(Ω)	n	RMSE	NFES
CWOA (2017) **[56]**	7.4760	1.2	0.00000490	9.7942	1.2069	1.7601E-02	50,000
ITLBO (2019) **[51]**	7.4725	2.335	0.0046	22.2199	1.2601	1.6601E-02	50,000
BHCS (2019) **[58]**	7.47253	2.33499	0.00459	22.21990	1.26010	1.6601E-02	50,000
ATLDE (2020) **[18]**	7.47252992	2.33499485	0.00459463	22.21989607	1.26010347	1.6601E-02	30,000
EJAYA (2021) **[43]**	7.47253	2.33499	0.00459	22.21989	1.2601	1.6601E-02	30,000
RLDE (2021) **[67]**	7.4725	2.335	0.0046	22.2199	1.2601	1.6601E-02	30,000
IEO (2020) **[69]**	7.472531264	2.23	0.004595	22.21989	1.260101	1.6601E-02	1,500,000
EHRJAYA	7.472742	2.3668696	0.004588927	22.49115	1.261246	1.6601E-02	**26,000**

6. Conclusions and future work

In order to find an algorithm with better performance for photovoltaic model extraction, an enhanced hybrid JAYA and Rao-1 algorithm, called EHRJAYA, is proposed in this paper. In the proposed EHRJAYA, the evolution strategies of the two algorithms are mixed, and the population diversity of the algorithm is improved. Then, an improved comprehensive learning strategy is introduced. Different selection probabilities are assigned to different individuals, which are used to select different update formulas to avoid insufficient using of information from the best individual and overusing of information from the worst individual. Then, two different adaptive coefficients are introduced into the two evolution strategies, so that the overall population tends to the optimal search agent and away from the worst search agent. Finally, the combination of linear population reduction strategy and dynamic lens opposition-based learning strategy improves the convergence speed of the algorithm and the ability to escape from local optimum. Ten well-known algorithms are selected as the comparison algorithms for the experiments, and the statistical analysis of the experimental results preliminarily proves that the EHRJAYA performs best compared with the comparison algorithms. Finally, compared with other well-known reported algorithms, the results further prove that the proposed EHRJAYA has strong competitiveness and is in a leading position among the famous algorithms.

In future work, EHRJAYA will be ready to be used to solve complex problems in higher dimensions, or even to create multi-objective versions, and so on.

Acknowledgments

The authors would also like to thank the supports of the following projects: The scientific research team project of Jing Chu University of technology with grant number TD202001. National Training Program of Innovation and Entrepreneurship for Undergraduates with grant number 202111336006. The key research and development project of Jing men with grant numbers 2019YFZD009. Provincial teaching reform research project of Hubei universities with grant number 2020683.

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. S. Li, W. Gong, X. Yan, C. Hu, D. Bai, L. Wang, Parameter estimation of photovoltaic models with memetic adaptive differential evolution, *Sol. Energy*, **190** (2019), 465–474. https://doi.org/10.1016/j.solener.2019.08.022
2. Z. Liao, Q. Gu, S. Li, Z. Hu, B. Ning, An improved differential evolution to extract photovoltaic cell parameters, *IEEE Access*, **8** (2020), 177838–177850. https://doi.org/10.1109/ACCESS.2020.3024975
3. S. Li, Q. Gu, W. Gong, B. Ning, An enhanced adaptive differential evolution algorithm for parameter extraction of photovoltaic models, *Energy Convers. Manage.*, **205** (2020), 112443. https://doi.org/10.1016/j.enconman.2019.112443
4. Z. Liao, Z. Chen, S. Li, Parameters Extraction of Photovoltaic Models Using Triple-Phase Teaching-Learning-Based Optimization, *IEEE Access*, **8** (2020), 69937-69952. https://doi.org/10.1109/ACCESS.2020.2984728
5. H. M. Ridha, H. Hizam, C. Gomes, A. A. Heidari, H. Chen, M. Ahmadipour, et al., Parameters extraction of three diode photovoltaic models using boosted LSHADE algorithm and Newton Raphson method, *Energy*, **224** (2021), 120136. https://doi.org/10.1016/j.energy.2021.120136
6. S. Li, W. Gong, Q. Gu, A comprehensive survey on meta-heuristic algorithms for parameter extraction of photovoltaic models, *Renewable Sustainable Energy Rev.*, **141** (2021), 110828. https://doi.org/10.1016/j.rser.2021.110828
7. D. Chan, J. Phang, Analytical methods for the extraction of solar-cell single- and double-diode model parameters from I-V characteristics, *IEEE Trans. Electron Devices*, **34** (1987), 286-293. https://doi.org/10.1109/T-ED.1987.22920
8. H. Saleem, S. Karmalkar, An Analytical Method to Extract the Physical Parameters of a Solar Cell From Four Points on the Illuminated, *IEEE Electron Device Lett.s*, **30** (2009), 349–352. https://doi.org/10.1109/LED.2009.2013882
9. T. Easwarakhanthan, J. Bottin, I. Bouhouch, C. Boutrit, Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers, *Int. J. Sol. Energy*, **4** (1986), 1–12. https://doi.org/10.1080/01425918608909835
10. A. Ortiz-Conde, F. J. García Sánchez, J. Muci, New method to extract the model parameters of solar cells from the explicit analytic solutions of their illuminated I–V characteristics, *Sol. Energy Mate. Sol. Cells*, **90** (2006), 352–361. https://doi.org/10.1016/j.solmat.2005.04.023

11. R. Messaoud, Extraction of uncertain parameters of single-diode model of a photovoltaic panel using simulated annealing optimization, *Energy Rep.*, **6** (2020), 350–357. https://doi.org/10.1016/j.egyr.2020.01.016
12. M. Abd Elaziz, D. Oliva, Parameter estimation of solar cells diode models by an improved opposition-based whale optimization algorithm, *Energy Convers. Manage.*, **171** (2018), 1843–1859. https://doi.org/10.1016/j.enconman.2018.05.062.
13. J. Liang, K. Qiao, M. Yuan, K. Yu, B. Qu, S. Ge, et al., Evolutionary multi-task optimization for parameters extraction of photovoltaic models, *Energy Convers. Manage.*, **207** (2020), 112509. https://doi.org/10.1016/j.enconman.2020.112509
14. A. Askarzadeh, A. Rezazadeh, Parameter identification for solar cell models using harmony search-based algorithms, *Sol. Energy*, **86** (2012), 3241–3249. https://doi.org/10.1016/j.solener.2012.08.018
15. T. Kang, J. Yao, M. Jin, S. Yang, T. Duong, A Novel Improved Cuckoo Search Algorithm for Parameter Estimation of Photovoltaic (PV) Models, *Energies*, **11** (2018), 1–31. https://doi.org/10.3390/en11051060
16. M. R. AlRashidi, M. F. AlHajri, K. M. El-Naggar, A. K. Al-Othman, A new estimation approach for determining the I–V characteristics of solar cells, *Sol. Energy*, **85** (2011), 1543–1550. https://doi.org/10.1016/j.solener.2011.04.013
17. A. Askarzadeh, A. Rezazadeh, Artificial bee swarm optimization algorithm for parameters identification of solar cell models, *Appl. Energy*, **102** (2013), 943–949. https://doi.org/10.1016/j.apenergy.2012.09.052
18. S. Li, W. Gong, L. Wang, X. Yan, C. Hu, A hybrid adaptive teaching–learning-based optimization and differential evolution for parameter identification of photovoltaic models, *Energy Convers. Manage.*, **225** (2020), 113474. https://doi.org/10.1016/j.enconman.2020.113474
19. G. Kanimozhi, K. Harish, Modeling of solar cell under different conditions by Ant Lion Optimizer with LambertW function, *Appl. Soft Comput.*, **71** (2018), 141–151. https://doi.org/10.1016/j.asoc.2018.06.025.
20. H. M. Ridha, H. Hizam, S. Mirjalili, M. L. Othman, M. E. Ya'acob, L. Abualigah, A novel theoretical and practical methodology for extracting the parameters of the single and double diode photovoltaic models, *IEEE Access*, **10** (2022), 11110–11137. https://doi.org/10.1109/ACCESS.2022.3142779
21. W. Zhou, P. Wang, A. A. Heidari, X. Zhao, H. Turabieh, M. Mafarja, et al., Metaphor-free dynamic spherical evolution for parameter estimation of photovoltaic modules, *Energy Rep.*, **7** (2021), 5175–5202. https://doi.org/10.1016/j.egyr.2021.07.041
22. A. Farah, A. Belazi, F. Benabdallah, A. Almalaq, M. Chtourou, M.A. Abido, Parameter extraction of photovoltaic models using a comprehensive learning Rao-1 algorithm, *Energy Convers. Manage.*, **252** (2022), 115057. https://doi.org/10.1016/j.enconman.2021.115057
23. J. Luo, J. Zhou, X. Jiang, A Modification of the Imperialist Competitive Algorithm With Hybrid Methods for Constrained Optimization Problems, *IEEE Access*, **9** (2021), 161745–161760. https://doi.org/10.1109/ACCESS.2021.3133579
24. M. Sattar, A. Al Sumaiti, H. Ali, A. Diab, Marine predators algorithm for parameters estimation of photovoltaic modules considering various weather conditions, *Neural Comput. Appl.*, **33** (2021), 11799–11819. https://doi.org/10.1007/s00521-021-05822-0

25. S. Jiao, G. Chong, C. Huang, H. Hu, M. Wang, A. A. Heidari, et al., Orthogonally adapted Harris hawks optimization for parameter estimation of photovoltaic models, *Energy*, **203** (2020), 117804. https://doi.org/10.1016/j.energy.2020.117804
26. H. Shaban, E. H. Houssein, M. Pérez-Cisneros, D. Oliva, A. Y. Hassan, A. Ismaeel, et al., Identification of parameters in photovoltaic models through a runge kutta optimizer, *Mathematics*, **9** (2021). https://doi.org/10.3390/math9182313
27. A. Ramadan, S. Kamel, M. H. Hassan, E. M. Ahmed, H. M. Hasanien, Accurate photovoltaic models based on an adaptive opposition artificial hummingbird algorithm, *Electronics*, **11** (2022). https://doi.org/10.3390/electronics11030318
28. A. Al-Shamma'a, H. Omotoso, F. Alturki, H. Farh, A. Alkuhayli, K. Alsharabi, et al., Parameter estimation of photovoltaic cell/modules using bonobo optimizer, *Energies*, **15** (2022), 140. https://doi.org/10.3390/en15010140
29. Y. Chen, D. Pi, B. Wang, J. Chen, Y. Xu, Bi-subgroup optimization algorithm for parameter estimation of a PEMFC model, *Expert Syst. Appl.*, **196** (2022), 116646. https://doi.org/10.1016/j.eswa.2022.116646
30. H. Rezk, S. Ferahtia, A. Djeroui, A. Chouder, A. Houari, M. Machmoum, et al., Optimal parameter estimation strategy of PEM fuel cell using gradient-based optimizer, *Energy*, **239** (2022), 122096. https://doi.org/10.1016/j.energy.2021.122096
31. H. Rezk, T. S. Babu, M. Al-Dhaifallah, H.A. Ziedan, A robust parameter estimation approach based on stochastic fractal search optimization algorithm applied to solar PV parameters, *Energy Rep.*, **7** (2021), 620–640. doi:https://doi.org/10.1016/j.egyr.2021.01.024
32. R. V. Rao, Jaya: A simple and new optimization algorithm for solving constrained and unconstrained optimization problems, *Int. J. Ind. Eng. Comput.*, **7** (2016), 19–34. https://doi.org/10.5267/j.ijiec.2015.8.004
33. R. V. Rao, Rao algorithms: three metaphor-less simple algorithms for solving optimization problems, *Int. J. Ind. Eng. Comput.*, **11** (2020), 107–130. https://doi.org/10.5267/j.ijiec.2019.6.002
34. Y. Zhang, Z. Jin, Comprehensive learning Jaya algorithm for engineering design optimization problems, *J. Intell. Manuf.*, https://doi.org/10.1007/s10845-020-01723-6
35. Y. Zhang, A. Chi, S. Mirjalili, Enhanced Jaya algorithm: A simple but efficient optimization method for constrained engineering design problems, *Knowl. Based Syst.*, **233** (2021), 107555. https://doi.org/10.1016/j.knosys.2021.107555
36. M. Afifi, H. Rezk, M. Ibrahim, M. El-Nemr, Multi-objective optimization of switched reluctance machine design using Jaya algorithm (MO-Jaya), *Mathematics*, **9** (2021), 1107. https://doi.org/10.3390/math9101107
37. S. Basak, B. Bhattacharyya, B. Dey, Combined economic emission dispatch on dynamic systems using hybrid CSA-JAYA Algorithm, *Int. J. Syst. Assur. Eng. Manage.*, (2022). https://doi.org/10.1007/s13198-022-01635-z
38. D. Saadaoui, M. Elyaqouti, K. Assalaou, D. B. hmamou, S. Lidaighbi, Multiple learning JAYA algorithm for parameters identifying of photovoltaic models, *Mater. Today Proc.*, (2021). https://doi.org/10.1016/j.matpr.2021.11.106
39. L. Wang, Z. Wang, H. Liang, C. Huang, Parameter estimation of photovoltaic cell model with Rao-1 algorithm, *Optik*, **210** (2020), 163846. https://doi.org/10.1016/j.ijleo.2019.163846

40. K. Junhua, L. Shuijia, G. Wenyin, Photovoltaic models parameter estimation via an enhanced Rao-1 algorithm, *Math. Biosci. Eng.*, **19** (2022), 1128–1153. https://doi.org/10.3934/mbe.2022052
41. L. Bhukya, A. Annamraju, S. Nandiraju, A novel maximum power point tracking technique based on Rao-1 algorithm for solar PV system under partial shading conditions, *Int. Trans. Electr. Energy Syst.*, **31** (2021), e13028. https://doi.org/10.1002/2050-7038.13028
42. X. Yu, X. Wu, W. Luo, Parameter Identification of Photovoltaic Models by Hybrid Adaptive JAYA Algorithm, *Mathematics*, **10** (2022), 183. https://doi.org/10.3390/math10020183
43. X. Yang, W. Gong, Opposition-based JAYA with population reduction for parameter estimation of photovoltaic solar cells and modules, *Appl. Soft Comput.*, **104** (2021), 107218. https://doi.org/10.1016/j.asoc.2021.107218
44. E. H. Houssein, E. Çelik, M. A. Mahdy, R. M. Ghoniem, Self-adaptive Equilibrium Optimizer for solving global, combinatorial, engineering, and Multi-Objective problems, *Expert Syst. Appl.*, **195** (2022), 116552. https://doi.org/10.1016/j.eswa.2022.116552
45. Y.Xiao, X. Sun, Y. Guo, S. Li, Y. Zhang, Y. Wang, An improved gorilla troops optimizer based on Lens opposition-based learning and adaptive β-Hill climbing for global optimization, *Comput. Model. Eng. Sci.* **131** (2022). https://doi.org/10.32604/cmes.2022.019198
46. A. W. Mohamed, A. K. Mohamed, Adaptive guided differential evolution algorithm with novel mutation for numerical optimization, *Int. J. Mach. Learn. Cybern.*, **10** (2019), 253–277. https://doi.org/10.1007/s13042-017-0711-7
47. K. Yu, J. J. Liang, B. Y. Qu, Z. Cheng, H. Wang, Multiple learning backtracking search algorithm for estimating parameters of photovoltaic models, *Appl. Energy*, **226** (2018), 408–422. https://doi.org/10.1016/j.apenergy.2018.06.010
48. X. Xia, L. Gui, F. Yu, H. Wu, B. Wei, Y.L. Zhang, et al., Triple Archives Particle Swarm Optimization, *IEEE Trans. Cybern.*, **50** (2020), 4862–4875. https://doi.org/10.1109/TCYB.2019.2943928
49. X. Chen, K. Yu, W. Du, W. Zhao, G. Liu, Parameters identification of solar cell models using generalized oppositional teaching learning based optimization, *Energy*, **99** (2016), 170–180. https://doi.org/10.1016/j.energy.2016.01.052
50. K. Yu, J. J. Liang, B. Y. Qu, X. Chen, H. Wang, Parameters identification of photovoltaic models using an improved JAYA optimization algorithm, *Energy Convers.Manage.*, **150** (2017), 742–753. https://doi.org/10.1016/j.enconman.2017.08.063.
51. S. Li, W. Gong, X. Yan, C. Hu, D. Bai, L. Wang, et al., Parameter extraction of photovoltaic models using an improved teaching-learning-based optimization, *Energy Convers.Manage.*, **186** (2019), 293–305. https://doi.org/10.1016/j.enconman.2019.02.048
52. K. Yu, B. Qu, C. Yue, S. Ge, X. Chen, J. Liang, A performance-guided JAYA algorithm for parameters identification of photovoltaic cell and module, *Appl. Energy*, **237** (2019), 241–257. https://doi.org/10.1016/j.apenergy.2019.01.008
53. X. Chen, B. Xu, C. Mei, Y. Ding, K. Li, Teaching–learning–based artificial bee colony for solar photovoltaic parameter estimation, *Appl. Energy*, **212** (2018), 1578–1588. https://doi.org/10.1016/j.apenergy.2017.12.115
54. D. H. Wolpert, W. G. Macready, No free lunch theorems for optimization, *IEEE Trans. Evol. Comput.*, **1** (1997), 67–82. https://doi.org/10.1109/4235.585893

55. K. Yu, X. Chen, X. Wang, Z. Wang, Parameters identification of photovoltaic models using self-adaptive teaching-learning-based optimization, *Energy Convers. Manage.* **145** (2017), 233–246. https://doi.org/10.1016/j.enconman.2017.04.054
56. D. Oliva, M. Abd El Aziz, A. Ella Hassanien, Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm, *Appl. Energy*, **200** (2017), 141–154. https://doi.org/10.1016/j.apenergy.2017.05.029
57. G. Xiong, J. Zhang, X. Yuan, D. Shi, Y. He, G. Yao, Parameter extraction of solar photovoltaic models by means of a hybrid differential evolution with whale optimization algorithm, *Sol. Energy*, **176** (2018), 742–761. https://doi.org/10.1016/j.solener.2018.10.050
58. X. Chen, K. Yu, Hybridizing cuckoo search algorithm with biogeography-based optimization for estimating photovoltaic model parameters, *Sol. Energy*, **180** (2019), 192–206. https://doi.org/10.1016/j.solener.2019.01.025.
59. S. M. Ebrahimi, E. Salahshour, M. Malekzadeh, G. Francisco, Parameters identification of PV solar cells and modules using flexible particle swarm optimization algorithm, *Energy*, **179** (2019), 358–372. https://doi.org/10.1016/j.energy.2019.04.218
60. N. Pourmousa, S. M. Ebrahimi, M. Malekzadeh, M. Alizadeh, Parameter estimation of photovoltaic cells using improved Lozi map based chaotic optimization Algorithm, *Sol. Energy*, **180** (2019), 180–191. https://doi.org/10.1016/j.solener.2019.01.026
61. Y. Zhang, C. Huang, Z. Jin, Backtracking search algorithm with reusing differential vectors for parameter identification of photovoltaic models, *Energy Convers. Manage.*, **223** (2020), 113266. https://doi.org/10.1016/j.enconman.2020.113266
62. J. Liang, S. Ge, B. Qu, K. Yu, F. Liu, H. Yang, et al., Classified perturbation mutation based particle swarm optimization algorithm for parameters extraction of photovoltaic models, *Energy Convers. Manage.*, **203** (2020), 112138. https://doi.org/10.1016/j.enconman.2019.112138
63. X. Lin, Y. Wu, Parameters identification of photovoltaic models using niche-based particle swarm optimization in parallel computing architecture, *Energy*, **196** (2020), 117054. doi:https://doi.org/10.1016/j.energy.2020.117054
64. Y. Zhang, M. Ma, Z. Jin, Backtracking search algorithm with competitive learning for identification of unknown parameters of photovoltaic systems, *Expert Syst. Appl.*, **160** (2020), 113750. https://doi.org/10.1016/j.eswa.2020.113750
65. W. Long, T. Wu, M. Xu, M. Tang, S. Cai, Parameters identification of photovoltaic models by using an enhanced adaptive butterfly optimization algorithm, *Energy*, **229** (2021), 120750. https://doi.org/10.1016/j.energy.2021.120750
66. Y. Liu, A. A. Heidari, X. Ye, C. Chi, X. Zhao, C. Ma, et al., Evolutionary shuffled frog leaping with memory pool for parameter optimization, *Energy Rep.*, **7** (2021), 584–606. https://doi.org/10.1016/j.egyr.2021.01.001
67. Z. Hu, W. Gong, S. Li, Reinforcement learning-based differential evolution for parameters extraction of photovoltaic models, *Energy Rep.*, **7** (2021), 916–928. https://doi.org/10.1016/j.egyr.2021.01.096
68. Y. Zhang, M. Ma, Z. Jin, Comprehensive learning Jaya algorithm for parameter extraction of photovoltaic models, *Energy*, **211** (2020), 118644. https://doi.org/10.1016/j.energy.2020.118644
69. M. Abdel-Basset, R. Mohamed, S. Mirjalili, R. K. Chakrabortty, M. J. Ryan, Solar photovoltaic parameter estimation using an improved equilibrium optimizer, *Sol. Energy*, **209** (2020), 694–708. https://doi.org/10.1016/j.solener.2020.09.032

70. M. Abdel-Basset, R. Mohamed, R. K. Chakrabortty, K. Sallam, M. J. Ryan, An efficient teaching-learning-based optimization algorithm for parameters identification of photovoltaic models: Analysis and validations, *Energy Convers. Manage.*, **227** (2021), 113614. https://doi.org/10.1016/j.enconman.2020.113614

71. X. Jian, Y. Zhu, Parameters identification of photovoltaic models using modified Rao-1 optimization algorithm, *Optik*, **231** (2021), 166439. https://doi.org/10.1016/j.ijleo.2021.166439

72. A. M. Beigi, A. Maroosi, Parameter identification for solar cells and module using a Hybrid Firefly and Pattern Search Algorithms, *Sol. Energy*, **171** (2018), 435–446. https://doi.org/10.1016/j.solener.2018.06.092

73. L. Deotti, J. Pereira, I. d. Silva Júnior, Parameter extraction of photovoltaic models using an enhanced Lévy flight bat algorithm, *Energy Convers. Manage.*, **221** (2020), 113114. https://doi.org/10.1016/j.enconman.2020.113114