

Parameter extraction of solar cell models using repaired adaptive differential evolution[☆]

Wenyin Gong^{*a}, Zhihua Cai^a

^a*School of Computer Science,
China University of Geosciences, Wuhan 430074, P.R. China*

Abstract

Parameter extraction of solar cell models plays an important role in the simulation and design calculation of photovoltaic (PV) systems. In this paper, in order to fast and accurately extract the solar cell parameters, an improved adaptive differential evolution with crossover rate repairing technique and ranking-based mutation is proposed. The proposed method is referred to as R_{cr} -IJADE, which is an improved version of JADE. In R_{cr} -IJADE, including the parameter adaptation presented in JADE, the crossover rate repairing technique and the ranking-based mutation are also synergized to improve the performance of JADE when solving the parameter extraction problems of solar cell models. In order to verify the performance of R_{cr} -IJADE, it is used to extract the parameters of different solar cell models, *i.e.*, single diode, double diode, and PV module. Compared with other parameter extraction techniques, experimental results indicate the superiority of R_{cr} -IJADE in terms of the quality of final solutions, success rate, and convergence speed. In addition, the simulated data with the extracted parameters of R_{cr} -IJADE are in very good agreement with the experimental data in all cases.

Key words: Solar cell models, parameter extraction, differential evolution, parameter adaptation, repairing technique, ranking-based mutation

1. Introduction

Because of several promising features like renewability, less pollution, ease of installation, and noise-free, the photovoltaic (PV) system such as solar cell has been obtained increasing interest recently [1]. For PV systems, it is very important to select a model to closely emulate the characteristics of PV cells [2]. Several models have been introduced to describe the current-voltage (I-V) relationship in solar cells over recent years [3, 4, 5, 6]. In practice, two main equivalent circuit models have been widely used: single and double diode models [7]. However, no matter what kinds of models, there are important PV parameters that need to be accurately extracted for the simulation, design, performance evaluation, and control of solar cell systems.

Generally, in the literature, there are two types of approaches for the purpose of parameter extraction of solar cell models: i) analytical approaches [8, 9, 10, 6, 11] and ii) numerical approaches [12, 13, 14]. Since the parameter extraction of solar cell models is a non-linear, multi-variable, and multi-modal problem with many local optima, traditional extraction techniques may be difficult to extract the parameters accurately. Therefore, in the later approaches, the use of artificial intelligence techniques for solar cell parameter extraction has received considerable attention recently [15, 16], such as genetic algorithms (GAs) [12, 17], particle swarm optimization (PSO) [18, 19, 20, 21, 22], differential evolution (DE) [23, 24, 2, 25], pattern search (PS) [26, 7], simulated annealing (SA) [27], harmony search (HS) [28], artificial bee swarm optimization (ABSO) [1], and so on.

[☆]This work was partly supported by the National Natural Science Foundation of China under Grant No. 61203307 and 61075063, the Fundamental Research Funds for the Central Universities at China University of Geosciences (Wuhan) under Grant No. CUG130413 and CUG090109, and the Research Fund for the Doctoral Program of Higher Education under Grant No. 201110145120009.

^{*}Corresponding author. Tel: +86-27-67883716.

Email addresses: wenyinong@yahoo.com; wygong@cug.edu.cn (Wenyin Gong), zhcai@cug.edu.cn (Zhihua Cai)

Differential evolution (DE), proposed by Storn and Price in 1997 [29], is a simple yet efficient evolutionary algorithm for the numerical optimization. Due to several advantages like ease of use, simple structure, and robustness, DE has been successfully used in diverse fields, such as data mining, sensor fusion, engineering design [30], etc. More applications can be found in the survey paper [31], and the references therein.

In the original DE algorithm, there are some pitfalls: i) the parameter setting of DE is sensitive; ii) the choice of optimal mutation strategy is difficult for a specific problem; and iii) DE is good at exploring the search space, however it is slow at exploitation of the solutions. Therefore, there are several advanced DE variants available in the literature to remedy some of drawbacks of DE, such as jDE [32], SaDE [33], JADE [34], DEGL [35], CoDE [36], and so on. The advanced DE variants obtain better performance than the original DE algorithm through benchmark functions. As mentioned above, in [23, 24, 2, 25], DE has been used to solve the parameter extraction problems of solar cell models. However, in these references only the original DE algorithm is employed.

Inspired by the promising performance obtained by the advanced DE variants, in this paper, we proposed an improved DE method so as to fast and accurately extract the parameters of solar cell models. The proposed method is called R_{cr} -IJADE for short. R_{cr} -IJADE is an improved version of JADE [34], where two improvements are synergized. i) To make the algorithm adapt the optimal crossover rate (Cr) quickly, a crossover rate repairing technique is implemented. And ii) the ranking-based mutation operator presented in [37] is employed to accelerate the convergence speed, and hence, to reduce the computational efforts. In order to verify the performance of R_{cr} -IJADE, it is used to extract the parameters of different solar cell models, *i.e.*, single diode, double diode, and PV module. Experimental results demonstrate the superiority of our approach when comparing with other parameter extraction techniques.

The main contributions of this paper are as follows:

- i) An improved JADE algorithm, R_{cr} -IJADE, is proposed. In R_{cr} -IJADE, two parameters of DE (*i.e.*, Cr and F) are adaptively controlled, which makes the algorithm be very useful to real-world applications. In addition, the ranking-based mutation operator can accelerate the convergence speed of R_{cr} -IJADE, and hence, it is able to reduce the computational efforts.
- ii) R_{cr} -IJADE is comprehensively investigated through the parameter extraction problems of different solar cell models via the experimental I-V data.
- iii) Compared with other different techniques, the superior performance of R_{cr} -IJADE is confirmed. Therefore, R_{cr} -IJADE can be an effective alternative to solve other complex optimization problems of PV systems.

The rest of this paper is organized as follows. In Section 2, the solar cell models used in this work together with the objective function to be optimized are introduced. The DE and JADE algorithms are briefly described in Section 3. Section 4 presents our proposed R_{cr} -IJADE method in detail, followed by the experimental results and analysis in Section 5. Finally, in Section 6, we conclude this paper.

2. Problem formulation

2.1. Solar cell models

In the literature, several models are available to describe the I-V characteristics of solar cells. In practice, two commonly used models are single and double diode models. In this subsection, we first introduce these two models briefly.

2.1.1. Double diode model

In the double diode model, the output current of solar cell can be formulated as follows [3, 26]:

$$I_L = I_{ph} - I_{d1} - I_{d2} - I_{sh} \quad (1)$$

where I_L is the cell output current. I_{ph} indicates the photogenerated current. I_{d1} and I_{d2} are the first and second diode currents, respectively. I_{sh} represents the shunt resistor current.

According to the Shockley equation, the two diode currents I_{d1} and I_{d2} can be calculated as

$$I_{d1} = I_{sd1} \left(\exp \left(\frac{V_L + I_L R_s}{a_1 V_t} \right) - 1 \right) \quad (2)$$

$$I_{d2} = I_{sd2} \left(\exp \left(\frac{V_L + I_L R_s}{a_2 V_t} \right) - 1 \right) \quad (3)$$

where V_L is the cell output voltage. I_{sd1} and I_{sd2} are respectively the diffusion and saturation currents. a_1 and a_2 denote the diffusion and recombination diode ideality factors, respectively. R_s is the series resistance, and V_t is the junction thermal voltage as

$$V_t = \frac{kT}{q} \quad (4)$$

where k is the Boltzmann constant ($1.3806503 \times 10^{-23}$ J/K), T is the temperature of the junction in Kelvin, and q is the electron charge ($1.60217646 \times 10^{-19}$ C).

The shunt resistor current I_{sh} is formulated as

$$I_{sh} = \frac{V_L + I_L R_s}{R_{sh}} \quad (5)$$

where R_{sh} denotes the shunt resistance.

In the above double diode model, there are seven parameters (*i.e.*, I_{ph} , I_{sd1} , I_{sd2} , R_s , R_{sh} , a_1 , and a_2) that need to be extracted from the I-V data of the solar cell.

2.1.2. Single diode model

Due to the simplicity and accuracy, the single diode model is also widely considered. In this model, the output current of cell is calculated as follows:

$$I_L = I_{ph} - I_{sd} \left(\exp \left(\frac{V_L + I_L R_s}{a V_t} \right) - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}} \quad (6)$$

For the single diode model, five parameters to be extracted are I_{ph} , I_{sd} , R_s , R_{sh} , and a .

2.1.3. Photovoltaic module

The single diode model of a PV module, which consists of N_s connected cells in series per string, is given as [22]:

$$I_L = I_{ph} - I_{sd} \left(\exp \left(\frac{V_L + I_L R_s}{aN_s V_t} \right) - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}} \quad (7)$$

2.2. Objective function

In order to extract the parameters of different solar cell models from the I-V data using the optimization techniques, we first need to define the objective function to be optimized. In this work, similar to [18, 25, 1], the root mean square error (RMSE) is used as the objective function, which is described as

$$F(\mathbf{x}) = \sqrt{\frac{1}{N} \sum_{k=1}^N f_k(V_L, I_L, \mathbf{x})^2} \quad (8)$$

where N is the number of experimental data. In Equation (8), for the double diode model

$$\begin{aligned} f(V_L, I_L, \mathbf{x}) = & I_{ph} - I_{sd1} \left(\exp \left(\frac{V_L + I_L R_s}{a_1 V_t} \right) - 1 \right) \\ & - I_{sd2} \left(\exp \left(\frac{V_L + I_L R_s}{a_2 V_t} \right) - 1 \right) \\ & - \frac{V_L + I_L R_s}{R_{sh}} - I_L \end{aligned} \quad (9)$$

$$\mathbf{x} = \{I_{ph}, I_{sd1}, I_{sd2}, R_s, R_{sh}, a_1, a_2\} \quad (10)$$

Table 1: Ranges of the parameters of the double and single diode models.

Parameter	Lower bound	Upper bound
I_{ph} (A)	0	1
I_{sd} (μ A)	0	1
R_s (Ω)	0	0.5
R_{sh} (Ω)	0	100
a	1	2

Table 2: Ranges of the parameters of the PV module.

Parameter	Lower bound	Upper bound
I_{ph} (A)	0	2
I_{sd} (μ A)	0	50
R_s (Ω)	0	2
R_{sh} (Ω)	0	2000
a	1	50

For the single diode model

$$f(V_L, I_L, \mathbf{x}) = I_{ph} - I_{sd} \left(\exp \left(\frac{V_L + I_L R_s}{a V_t} \right) - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}} - I_L \quad (11)$$

$$\mathbf{x} = \{I_{ph}, I_{sd}, R_s, R_{sh}, a\} \quad (12)$$

\mathbf{x} is decision vector which consists of the parameters to be extracted. For each parameter, it is bounded in the search space. In the double and single diode models, the lower and upper boundaries of each parameter are shown in Table 1, which is the same as used in [1]. While in the PV module, the lower and upper boundaries of each parameter are tabulated in Table 2.

Obviously, in order to make the simulated data better fit the experimental data, the objective function $F(\mathbf{x})$ in Equation (8) needs to be minimized. The smaller the objective function, the better the solutions obtained.

3. DE and JADE

In this section, the original DE algorithm and the JADE algorithm are briefly presented.

3.1. The DE algorithm

The DE algorithm is originally proposed by Storn and Price in 1997 [29], which is mainly used for the numerical optimization problems. In DE, there are four operations, *i.e.*, initialization, mutation, crossover, and selection, which will be described as follows.

3.1.1. Population initialization

Generally, the population of DE consists of μ solutions (vectors). The population is initialized at random within the boundaries. For example, for the i -th vector \mathbf{x}_i it is initialized as follows:

$$x_{i,j} = L_j + \text{rndreal}(0, 1) \cdot (U_j - L_j) \quad (13)$$

where L_j and U_j are respectively the lower bound and upper bound of x_j , *i.e.*, $x_j \in [L_j, U_j]$. $i = 1, \dots, \mu$, $j = 1, \dots, D$. D is the number of decision variables. $\text{rndreal}(0, 1)$ is a uniformly distributed random real number in $(0, 1)$.

3.1.2. Mutation

The mutation operation is also named as *differential* mutation, which is the core operator of DE. For each *target* vector \mathbf{x}_i , a *mutant* vector \mathbf{v}_i is generated by the mutation operator. For example, for the “DE/rand-to-best/1” mutation, it is formulated as:

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{\text{best}} - \mathbf{x}_{r_1}) + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \quad (14)$$

where F is the mutation scaling factor, $r_1, r_2, r_3 \in \{1, \dots, \mu\}$ are mutually different integers randomly generated, and $r_1 \neq r_2 \neq r_3 \neq i$. \mathbf{x}_{best} is the best-so-far solution in the current population.

3.1.3. Crossover

In order to diversify the current population, following mutation, DE employs the crossover operator to produce the *trial* vector \mathbf{u}_i between \mathbf{x}_i and \mathbf{v}_i . The most commonly used operator is the *binomial* crossover performed on each component as follows:

$$u_{i,j} = \begin{cases} v_{i,j}, & \text{if } (\text{rndreal}(0, 1) < Cr \text{ or } j == j_{\text{rand}}) \\ x_{i,j}, & \text{otherwise} \end{cases} \quad (15)$$

where Cr is the crossover rate and j_{rand} is a randomly generated integer within $\{1, D\}$. The notation “ $a == b$ ” indicates a is equal to b .

3.2. Selection

Finally, to keep the population size constant in the following generations, the selection operation is employed to determine whether the trial or the target vector survives to the next generation. In DE, the one-to-one tournament selection is used as follows:

$$\mathbf{x}_i = \begin{cases} \mathbf{u}_i, & \text{if } F(\mathbf{u}_i) \leq F(\mathbf{x}_i) \\ \mathbf{x}_i, & \text{otherwise} \end{cases} \quad (16)$$

where $F(\mathbf{x})$ is the objective function to be optimized.

3.3. The JADE algorithm

As above-description, DE has three parameters, *i.e.*, μ , Cr , and F . Previous studies [38, 32] indicate that the parameter setting is crucial to the performance of DE. To remedy this drawback, Zhang and Sanderson presented an adaptive DE variant (JADE) in [34], where the parameters Cr and F are adaptively controlled according their successful experience in the last generation. The adaptation techniques proposed in JADE are briefly introduced as follows.

3.3.1. Crossover rate adaptation

At each generation, for each target vector the crossover rate Cr_i is independently generated as follows:

$$Cr_i = \text{rndn}_i(\mu_{Cr}, 0.1) \quad (17)$$

and truncated to the interval $[0, 1]$. Where μ_{Cr} is the mean value to generate Cr_i . It is updated as follows:

$$\mu_{Cr} = (1 - c) \cdot \mu_{Cr} + c \cdot \text{mean}_A(S_{Cr}) \quad (18)$$

where c is a constant in $[0, 1]$; $\text{mean}_A(\cdot)$ is the usual arithmetic mean operation; and S_{Cr} is the set of all successful crossover rates Cr_i at generation g .

3.3.2. Mutation factor adaptation

In order to maintain the population diversity, for each target vector the mutation factor F_i is independently calculated as:

$$F_i = \text{rndc}_i(\mu_F, 0.1) \quad (19)$$

and then truncated to be 1.0 if $F_i > 1.0$ or regenerated if $F_i \leq 0$. $\text{rndc}_i(\mu_F, 0.1)$ is a random number generated according to the Cauchy distribution with location parameter μ_F and scale parameter 0.1. The location parameter μ_F is updated in the following manner:

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}_L(S_F) \quad (20)$$

where S_F is the set of all successful mutation factors F_i at generation g ; and $\text{mean}_L(\cdot)$ is the Lehmer mean:

$$\text{mean}_L(S_F) = \frac{\sum_{i=1}^{|S_F|} F_i^2}{\sum_{i=1}^{|S_F|} F_i} \quad (21)$$

4. Repaired JADE: R_{cr}-IJADE

In order to fast and accurately extract the parameters of different solar cell models, we propose the improved JADE method, *i.e.*, R_{cr}-IJADE. In R_{cr}-IJADE, the parameter adaptation techniques mentioned above are used. Additionally, two improvements (*i.e.*, crossover rate repairing and ranking-based mutation) are implemented in the following subsections.

4.1. Crossover rate repairing technique

The most commonly used crossover operator is the binomial crossover (see Equation (15)) in the DE algorithm. In order to analyze the behavior of the binomial crossover, we let \mathbf{b}_i be a binary string generated for each target vector \mathbf{x}_i as follows:

$$b_{i,j} = \begin{cases} 1, & \text{if } (\text{rndreal}(0, 1) < Cr \text{ or } j == j_{\text{rand}}) \\ 0, & \text{otherwise} \end{cases} \quad (22)$$

Therefore, the binomial crossover of DE in Equation (15) can be reformulated as

$$u_{i,j} = b_{i,j} \cdot v_{i,j} + (1 - b_{i,j}) \cdot x_{i,j} \quad (23)$$

where $i = 1, \dots, \mu$ and $j = 1, \dots, D$. According to Equations (22) and (23), we can see that the binary string \mathbf{b}_i is stochastically related to Cr ; however, the trial vector \mathbf{u}_i is *directly* related to its binary string \mathbf{b}_i , but not directly related to its crossover rate Cr . Based on this consideration, we propose the crossover repairing technique, where the crossover rate is repaired by its corresponding binary string, *i.e.* by using the average number of components taken from the mutant. Suppose that Cr'_i is the repaired crossover rate, it is calculated as

$$Cr'_i = \frac{\sum_{j=1}^D b_{i,j}}{D} \quad (24)$$

where \mathbf{b}_i is the binary string calculated in Equation (22). The crossover rate is repaired after its binary string is generated by Equation (22) based on Cr_i . If the trial vector \mathbf{u}_i is a successful vector, Cr'_i will be stored in S_{Cr} , instead of storing Cr_i .

4.2. Ranking-based mutation operator

In JADE [39], the authors presented a modified “DE/rand-to-best/1” mutation (*i.e.*, “DE/rand-to- p best/1”) as

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F_i \cdot (\mathbf{x}_{\text{best}}^p - \mathbf{x}_{r_1}) + F_i \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \quad (25)$$

where $\mathbf{x}_{\text{best}}^p$ refers to the p best solution, which is randomly selected from the top $100p\%$ solutions, with $p \in (0, 1]$. $r_1, r_2, r_3 \in \{1, \dots, \mu\}$ and $r_1 \neq r_2 \neq r_3 \neq i$.

Algorithm 1: Ranking-based vector selection

Input: The target vector index i
Output: The selected vector indexes r_1, r_2, r_3
while $\text{rndreal}(0, 1) > pv_{r_1}$ **or** $r_1 == i$ **do**
 └ Randomly select $r_1 \in \{1, \mu\}$;
Randomly select $r_2 \in \{1, \mu\}$;
while $\text{rndreal}(0, 1) > pv_{r_2}$ **or** $r_2 == r_1$ **or** $r_2 == i$ **do**
 └ Randomly select $r_2 \in \{1, \mu\}$;
Randomly select $r_3 \in \{1, \mu\}$;
while $r_3 == r_2$ **or** $r_3 == r_1$ **or** $r_3 == i$ **do**
 └ Randomly select $r_3 \in \{1, \mu\}$;

In Equation (25), since \mathbf{x}_{r_1} , \mathbf{x}_{r_2} , and \mathbf{x}_{r_3} are only randomly selected from the population, it may make the algorithm be good at exploring the search space, but be slow at exploitation of the solutions. Therefore, in order to reduce the computational efforts and make the parameter extraction process of solar cell models faster, in this work, the ranking-based vector selection technique [37] is used in the above mutation operator.

In the ranking-based vector selection technique, the population is firstly ranked from the best to the worst. Then, the rank R_i of each vector \mathbf{x}_i in the sorted population is assigned as

$$R_i = \mu - i, \quad i = 1, 2, \dots, \mu \quad (26)$$

After that, the selection probability of each vector is calculated as follows:

$$pv_i = \left(\frac{R_i}{\mu}\right)^2 \quad (27)$$

Finally, the vectors (\mathbf{x}_{r_1} and \mathbf{x}_{r_2}) in the mutation are selected according to their selection probabilities as shown in Algorithm 1. Obviously, better solutions have more chance to be chosen as \mathbf{x}_{r_1} and \mathbf{x}_{r_2} in the “DE/rand-to-pbest/1” mutation to generate the mutant vector.

4.3. Boundary-handling technique

In DE, after performing the mutation operation to generate a new solution, some variables may be out of their corresponding boundaries, *i.e.*, $x_j \notin [L_j, U_j]$, if that, the following boundary-handling technique is applied:

$$x_j = L_j + \text{rndreal}(0, 1) \cdot (U_j - L_j) \quad (28)$$

Actually, it is the *reinitialization* method as mentioned in [40].

4.4. The R_{cr} -IJADE algorithm

Combining the parameter adaptation techniques presented in JADE [34] with the crossover rate repairing technique and the ranking-based “DE/rand-to-pbest/1” mutation, our proposed R_{cr} -IJADE is developed. The pseudo-code of R_{cr} -IJADE is introduced in Algorithm 2, where NFES means the number of function evaluations and Max_NFES is the maximal NFES, which is the termination criterion in this work. Note that we do not use the maximal generations as the termination criterion, since for different algorithms the consumed NFES at one generation are not the same. From Algorithm 2, we can see that there are only two parameters (μ and Max_NFES) that need to be given by the user, while other parameters (*i.e.*, μ_{Cr} , μ_F , c , p) are kept the default values as used in JADE [34]. This makes R_{cr} -IJADE be easy to be applied to real-world applications.

5. Experimental results and analysis

In this section, the performance of R_{cr} -IJADE is evaluated for parameter extraction of different solar cell models, *i.e.*, single diode model, double diode model, and PV module. The experimental I-V data of a solar cell and a solar module are used for this purpose. The data is obtained from [41], where a 57 mm diameter commercial (R.T.C France)

Algorithm 2: The pseudo-code of the R_{cr}-IJADE algorithm

```

Input: Control parameters:  $\mu$  and Max_NFEs
Output: The best final solution
Set  $\mu_{Cr} = 0.5, \mu_F = 0.5, c = 0.1, p = 0.05$ ;
Initialize the population  $\mathbb{P}$  randomly;
Calculate the objective function value of each solution in the population;
NFEs =  $\mu$ ;
while NFEs < Max_NFEs do
     $S_{Cr} = \phi, S_F = \phi$ ;
    Sort the population from the best to the worst;
    for  $i = 1$  to  $\mu$  do
        Generate  $Cr_i$  and  $F_i$  with Equations (17) and (19), respectively;
        Calculate the ranking  $R_i$  and the selection probability  $pv_i$ ;
    for  $i = 1$  to  $\mu$  do
        Select  $r_1, r_2$ , and  $r_3$  as shown in Algorithm 1;
        Produce the mutant vector  $\mathbf{v}_i$  with ranking-based "DE/rand-to-pbest/1";
        Get the binary string  $\mathbf{b}_i$  as shown in Equation (22);
        Calculate the repaired crossover rate  $Cr'_i$  with Equation (24);
        for  $j = 1$  to  $D$  do
             $u_{i,j} = b_{i,j} \cdot v_{i,j} + (1 - b_{i,j}) \cdot x_{i,j}$ ;
        Calculate the objective function value of  $\mathbf{u}_i$ ;
    for  $i = 1$  to  $\mu$  do
        if  $\mathbf{u}_i$  is better than its parent  $\mathbf{x}_i$  then
             $\mathbf{x}_i = \mathbf{u}_i$ ;
             $Cr'_i \rightarrow S_{Cr}$ ;
             $F_i \rightarrow S_F$ ;
    Update the  $\mu_{Cr}$  and  $\mu_F$  with Equations (18) and (20), respectively;
    NFEs = NFEs +  $\mu$ ;

```

silicon solar cell (at 33°C) and a solar module (Photowatt-PWP 201, at 45°C) in which 36 polycrystalline silicon cells are connected in series.

As shown in Algorithm 2, there are two parameters that need to be given by the user. In this work, the population size $\mu = 50$ is used for all experiments. For the single diode model and the PV module, the Max_NFEs is set to be 10,000, while for the double diode model Max_NFEs = 20,000. R_{cr}-IJADE is coded in standard C++.

In order to show the superior performance of R_{cr}-IJADE, in this section, it is firstly compared with the reported results in the literature with respect to the RMSE values and extracted parameters. Then, R_{cr}-IJADE is directly compared with other advanced DE variants according to the statistical results of different performance criteria.

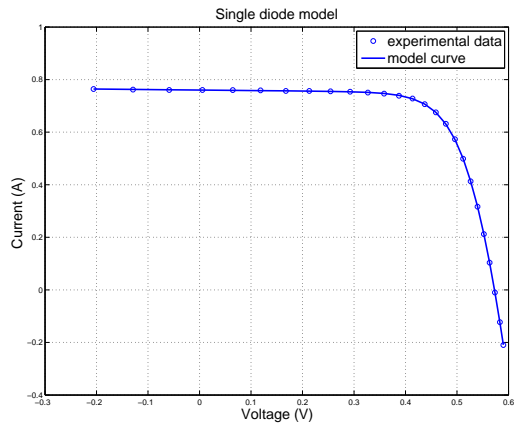
Table 3: Comparison among different parameter extraction techniques for the single diode model.

Item	GA [26]	CPSO [20]	PS [7]	SA [27]	IGHS [28]	ABSO [1]	R _{cr} -IJADE
I_{ph} (A)	0.7619	0.7607	0.7617	0.7620	0.7608	0.7608	0.760776
I_{sd} (μA)	0.8087	0.4000	0.9980	0.4798	0.3435	0.3062	0.323021
R_s (Ω)	0.0299	0.0354	0.0313	0.0345	0.0361	0.0366	0.036377
R_{sh} (Ω)	42.3729	59.0120	64.1026	43.1034	53.2845	52.2903	53.718526
a	1.5751	1.5033	1.6000	1.5172	1.4874	1.4758	1.481184
RMSE	0.01908	0.00139	0.01494	0.01900	9.9306E-04	9.9124E-04	9.8602E-04

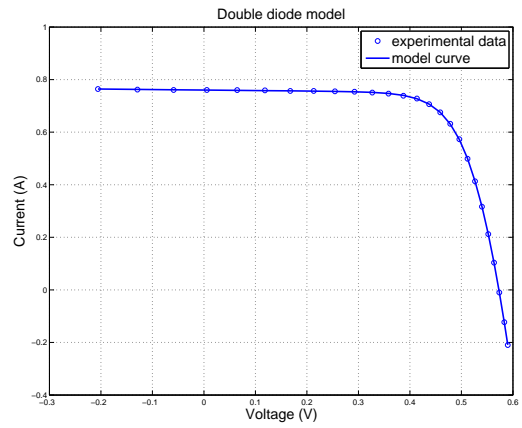
5.1. Results on single diode model

For the single diode model, the extracted parameters and RMSE value of R_{cr}-IJADE are compared with those of GA [26], CPSO [20], PS [7], SA [27], IGHS [28], and ABSO [1]. These methods are chosen for comparison due to the good performance obtained by them in the single diode model. The experimental results are reported in Table 3. The overall best and the second best RMSE values among the compared methods are highlighted in **grey boldface** and **boldface**, respectively.

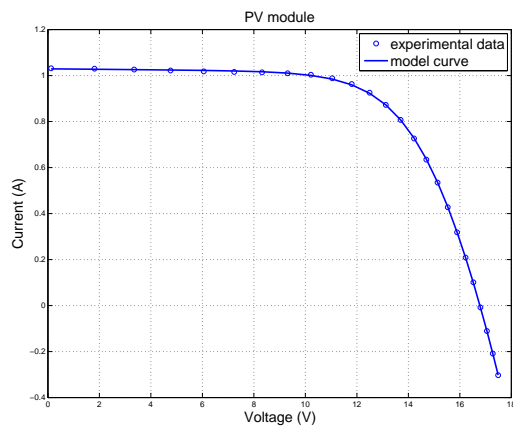
From Table 3, it is clear to observe that R_{cr}-IJADE provides the best RMSE value among all compared methods, followed by ABSO, IGHS, CPSO, PS, SA, and GA. Although the RMSE values of ABSO and IGHS are very close to that of R_{cr}-IJADE, however, since the experimental data is adopted, no information is available about the accurate values of the parameters; therefore, any reduction in the objective function value is significant because it results



(a)



(b)



(c)

Figure 1: Comparison on the I-V characteristics between the experimental data and simulated data obtained by R_{cr} -IJADE for 1(a): the single diode model, 1(b): the double diode model, and 1(c): the PV module.

in improvement in the knowledge about the real values of the parameters [26]. It is worth emphasizing that the Max_NFEs of R_{cr} -IJADE is only 10,000, which is much less than those of ABSO (150,000), IGHS (150,000), and CPSO (45,000). The Max_NFEs for GA, PS, and SA are not available in the literature.

Table 4: Curve fitting results of R_{cr} -IJADE for the single diode model.

Item	V_L (V)	I_L measured (A)	I_L calculated (A)	IAE
1	-0.2057	0.7640	0.76409559	0.00009559
2	-0.1291	0.7620	0.76266611	0.00066611
3	-0.0588	0.7605	0.76135473	0.00085473
4	0.0057	0.7605	0.76014966	0.00035034
5	0.0646	0.7600	0.75905702	0.00094298
6	0.1185	0.7590	0.75804472	0.00095528
7	0.1678	0.7570	0.75709510	0.00009510
8	0.2132	0.7570	0.75615050	0.00084950
9	0.2545	0.7555	0.75508177	0.00041823
10	0.2924	0.7540	0.75367033	0.00032967
11	0.3269	0.7505	0.75139542	0.00089542
12	0.3585	0.7465	0.74735737	0.00085737
13	0.3873	0.7385	0.74010420	0.00160420
14	0.4137	0.7280	0.72740088	0.00059912
15	0.4373	0.7065	0.70694631	0.00044631
16	0.4590	0.6755	0.67530400	0.00019600
17	0.4784	0.6320	0.63089105	0.00110895
18	0.4960	0.5730	0.57208973	0.00091027
19	0.5119	0.4990	0.49949902	0.00049902
20	0.5265	0.4130	0.41349030	0.00049030
21	0.5398	0.3165	0.31721532	0.00071532
22	0.5521	0.2120	0.21210468	0.00010468
23	0.5633	0.1035	0.10271603	0.00078397
24	0.5736	-0.0100	-0.00924563	0.00075437
25	0.5833	-0.1230	-0.12437754	0.00137754
26	0.5900	-0.2100	-0.20919680	0.00080320
Sum of IAE				0.01770357

In addition, the I-V characteristic obtained by R_{cr} -IJADE and the individual absolute error (IAE) [7] between the experimental data and simulated data are shown in Figure 1(a) and Table 4, respectively. The results clearly indicate that the simulated data generated by R_{cr} -IJADE are highly coincide with the experimental data in the single diode model, which means that the extracted parameters of R_{cr} -IJADE are very accurate.

Table 5: Comparison among different parameter extraction techniques for the double diode model.

Item	PS [7]	SA [27]	IGHS [28]	ABSO [1]	R_{cr} -IJADE
I_{ph} (A)	0.7602	0.7623	0.7608	0.7608	0.760781
I_{sd1} (μ A)	0.9889	0.4767	0.9731	0.2671	0.225974
R_s (Ω)	0.0320	0.0345	0.0369	0.0366	0.036740
R_{sh} (Ω)	81.3008	43.1034	56.8368	54.6219	55.485443
a_1	1.6000	1.5172	1.9213	1.4651	1.451017
I_{sd2} (μ A)	0.0001	0.0100	0.1679	0.3819	0.749347
a_2	1.1920	2.0000	1.4281	1.9815	2.000000
RMSE	0.01518	0.01664	9.8635E-04	9.8344E-04	9.8248E-04

5.2. Results on double diode model

In this model, seven parameters need to be extracted. The results of R_{cr} -IJADE are indirectly compared with those of PS [7], SA [27], IGHS [28], and ABSO [1]. The extracted parameters and RMSE values of different methods are tabulated in Table 5 and the curve fitting results of R_{cr} -IJADE are reported in Table 6. Additionally, the I-V characteristic of R_{cr} -IJADE is plotted in Figure 1(b). Similar to the results in the single diode model, in this case, our proposed R_{cr} -IJADE still obtains the best RMSE value but with less Max_NFEs (20,000) compared with other methods. Moreover, by returning the extracted parameters to the double diode model, the simulated data of R_{cr} -IJADE are in very good agreement with the experimental data almost in all data points as shown in Table 6 and Figure 1(b).

Table 6: Curve fitting results of R_{cr} -IJADE for the double diode model.

Item	V_L (V)	I_L measured (A)	I_L calculated (A)	IAE
1	-0.2057	0.7640	0.76409268	0.00009268
2	-0.1291	0.7620	0.76265394	0.00065394
3	-0.0588	0.7605	0.76135755	0.00085755
4	0.0057	0.7605	0.76016253	0.00033747
5	0.0646	0.7600	0.75906000	0.00094000
6	0.1185	0.7590	0.75805065	0.00094935
7	0.1678	0.7570	0.75709635	0.00009635
8	0.2132	0.7570	0.75614465	0.00085535
9	0.2545	0.7555	0.75508115	0.00041885
10	0.2924	0.7540	0.75366874	0.00033126
11	0.3269	0.7505	0.75139511	0.00089511
12	0.3585	0.7465	0.74734939	0.00084939
13	0.3873	0.7385	0.74010214	0.00160214
14	0.4137	0.7280	0.72738784	0.00061216
15	0.4373	0.7065	0.70695162	0.00045162
16	0.4590	0.6755	0.67530112	0.00019888
17	0.4784	0.6320	0.63088766	0.00111234
18	0.4960	0.5730	0.57207477	0.00092523
19	0.5119	0.4990	0.49949417	0.00049417
20	0.5265	0.4130	0.41349125	0.00049125
21	0.5398	0.3165	0.31721918	0.00071918
22	0.5521	0.2120	0.21210831	0.00010831
23	0.5633	0.1035	0.10272032	0.00077968
24	0.5736	-0.0100	-0.00924461	0.00075539
25	0.5833	-0.1230	-0.12437667	0.00137667
26	0.5900	-0.2100	-0.20919499	0.00080501
Sum of IAE				0.01770933

Table 7: Comparison among different parameter extraction techniques for the PV module.

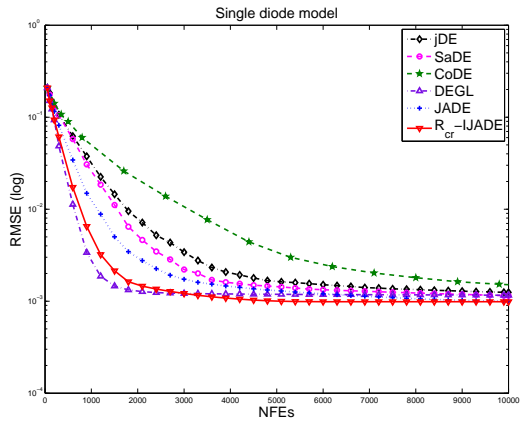
Item	Newton [41]	Method in [42]	CPSO [20]	Method in [43]	PS [7]	SA [27]	R_{cr} -IJADE
I_{ph} (A)	1.0318	1.0339	1.0286	1.0310	1.0313	1.0331	1.030514
I_{sd} (μ A)	3.2875	3.0760	8.3010	3.8236	3.1756	3.6642	3.482263
R_s (Ω)	1.2057	1.2030	1.0755	1.0920	1.2053	1.1989	1.201271
R_{sh} (Ω)	555.5556	555.5556	1850.1000	689.6600	714.2857	833.3333	981.982240
a	48.4500	48.1862	52.2430	48.9300	48.2889	48.8211	48.642835
RMSE	0.7805	0.6130	0.0035	0.0102	0.0118	0.0027	0.002425

5.3. Results on PV module

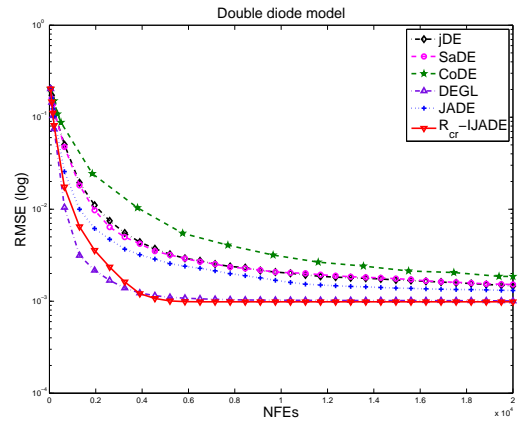
For the PV module, the experimental results are reported in Table 7, where R_{cr} -IJADE is compared with Newton [41], method in [42], CPSO [20], method in [43], PS [7], and SA [27]. Like the previous cases, R_{cr} -IJADE also gets the best RMSE value among all compared methods in the PV module. Moreover, according the curve fitting results shown in Table 8 and the I-V characteristic plotted in Figure 1(c), we can see that the simulated data obtained by R_{cr} -IJADE fit the experimental data very well.

5.4. Statistical results and convergence speed

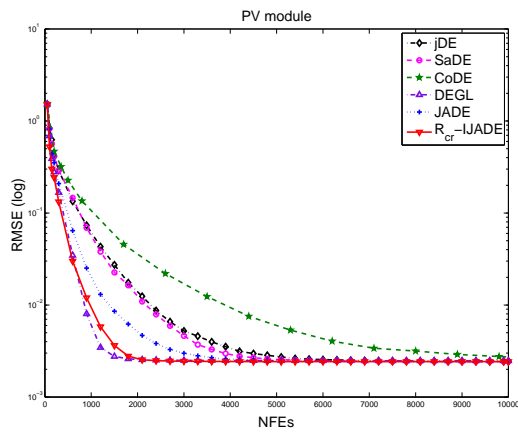
In the previous subsections, the superiority of R_{cr} -IJADE has been confirmed by indirectly comparing it with other parameter extraction techniques in the literature. As mentioned above, there are several DE variants that have obtained promising results via benchmark functions. Therefore, in order to further evaluate the performance of R_{cr} -IJADE, it is compared with five advanced DE variants (*i.e.*, jDE [32], SaDE [33], CoDE [36], DEGL [35], and JADE [34]) to solve the parameter extraction problems of solar cell models. The Max_NFEs for the three solar cell models are the same as used in R_{cr} -IJADE in the previous experiments. In addition, to make a fair comparison, all DE variants use the same population size, *i.e.*, $\mu = 50$. All other parameters are kept the same as used in their original literature. Note that the parameter settings of JADE are the same as R_{cr} -IJADE, the only differences are that in JADE the crossover rate repairing and ranking-based mutation are not employed. Since the DE algorithm is stochastic, all DE variants are executed over 100 independent runs to make the comparison meaningful. All DE variants are coded in standard C++.



(a)



(b)



(c)

Figure 2: Convergence graphs of different DE variants for 2(a): the single diode model , 2(b): the double diode model, and 2(c): the PV module.

Table 8: Curve fitting results of R_{cr} -IJADE for the PV module.

Item	V_L (V)	I_L measured (A)	I_L calculated (A)	IAE
1	0.1248	1.0315	1.02912049	0.00237951
2	1.8093	1.0300	1.02738564	0.00261436
3	3.3511	1.0260	1.02573499	0.00026501
4	4.7622	1.0220	1.02409557	0.00209557
5	6.0538	1.0180	1.02227575	0.00427575
6	7.2364	1.0155	1.01991719	0.00441719
7	8.3189	1.0140	1.01635389	0.00235389
8	9.3097	1.0100	1.01048191	0.00048191
9	10.2163	1.0035	1.00068707	0.00281293
10	11.0449	0.9880	0.98465514	0.00334486
11	11.8018	0.9630	0.95969425	0.00330575
12	12.4929	0.9255	0.92305160	0.00244840
13	13.1231	0.8725	0.87258829	0.00008829
14	13.6983	0.8075	0.80731392	0.00018608
15	14.2221	0.7265	0.72796294	0.00146294
16	14.6995	0.6345	0.63646347	0.00196347
17	15.1346	0.5345	0.53569189	0.00119189
18	15.5311	0.4275	0.42882216	0.00132216
19	15.8929	0.3185	0.31867170	0.00017170
20	16.2229	0.2085	0.20785189	0.00064811
21	16.5241	0.1010	0.09835838	0.00264162
22	16.7987	-0.0080	-0.00817367	0.00017367
23	17.0499	-0.1110	-0.11096908	0.00003092
24	17.2793	-0.2090	-0.20912100	0.00012100
25	17.4885	-0.3030	-0.30202427	0.00097573
Sum of IAE				0.04177271

Table 9: Comparison on different performance criteria among different DE variants for the single diode model. "NA" means not available.

Algorithm	RMSE						NFEs $_{\epsilon}$ ($\epsilon = 0.001$)		S_r
	Min	Median	Max	Mean	Std	sig.	Mean	Std	
jDE	1.011467E-03	1.221579E-03	1.674285E-03	1.236149E-03	1.48E-04	+	NA	NA	0.00
SaDE	9.972754E-04	1.140693E-03	1.386479E-03	1.145756E-03	9.26E-05	+	9850.00	NA	0.01
CoDE	1.101993E-03	1.494410E-03	2.081250E-03	1.505273E-03	1.93E-04	+	NA	NA	0.00
DEGL	9.860219E-04	1.084221E-03	1.827544E-03	1.162104E-03	2.06E-04	+	3308.62	2098.05	0.29
JADE	9.860219E-04	9.889961E-04	1.142030E-03	1.001006E-03	2.88E-05	+	8529.17	1094.38	0.72
R_{cr} -IJADE	9.860219E-04	9.860219E-04	9.860219E-04	9.860219E-04	5.12E-16		4430.50	1084.74	1.00

"+" indicates R_{cr} -IJADE is significantly better than its competitor according to the Wilcoxon signed-rank test at $\alpha = 0.05$, hereinafter.

Table 10: Comparison on different performance criteria among different DE variants for the double diode model. "NA" means not available.

Algorithm	RMSE						NFEs $_{\epsilon}$ ($\epsilon = 0.001$)		S_r
	Min	Median	Max	Mean	Std	sig.	Mean	Std	
jDE	1.070336E-03	1.470803E-03	2.114012E-03	1.488346E-03	2.66E-04	+	NA	NA	0.00
SaDE	9.917107E-04	1.460147E-03	2.221314E-03	1.496245E-03	2.82E-04	+	19400.00	NA	0.01
CoDE	1.225893E-03	1.820441E-03	2.563051E-03	1.840997E-03	3.07E-04	+	NA	NA	0.00
DEGL	9.825195E-04	9.859891E-04	1.583493E-03	1.021629E-03	9.55E-04	+	6155.92	3450.34	0.76
JADE	9.826437E-04	1.241540E-03	2.125234E-03	1.283669E-03	2.73E-04	+	13034.62	2189.05	0.26
R_{cr} -IJADE	9.824849E-04	9.826140E-04	9.860244E-04	9.826140E-04	9.86E-05		4407.50	834.86	1.00

Table 11: Comparison on different performance criteria among different DE variants for the PV module. "NA" means not available.

Algorithm	RMSE						NFEs $_{\epsilon}$ ($\epsilon = 0.01$)		S_r
	Min	Median	Max	Mean	Std	sig.	Mean	Std	
jDE	2.425084E-03	2.428733E-03	2.606556E-03	2.437621E-03	2.44E-05	+	2303.00	424.91	1.00
SaDE	2.425075E-03	2.425405E-03	2.470321E-03	2.428499E-03	6.70E-06	+	2117.50	333.06	1.00
CoDE	2.437571E-03	2.631745E-03	3.499591E-03	2.689244E-03	2.10E-04	+	3756.50	755.67	1.00
DEGL	2.425075E-03	2.425120E-03	3.303094E-03	2.457298E-03	1.14E-04	+	847.00	97.40	1.00
JADE	2.425075E-03	2.425075E-03	2.639902E-03	2.428243E-03	2.18E-05	+	1395.00	262.13	1.00
R_{cr} -IJADE	2.425075E-03	2.425075E-03	2.425075E-03	2.425075E-03	2.90E-17		976.50	169.29	1.00

5.4.1. Performance criteria

To compare the performance between different algorithms, the following performance criteria are used.

- **RMSE:** As described in Equation (8), it is used to measure the quality of the fit between the simulated data and experimental data obtained by the algorithm.
- **NFEs_ε:** The NFEs_ε is used to record the number of function evaluations in each run for finding a solution \mathbf{x} satisfying $F(\mathbf{x}) - F(\mathbf{x}^*) \leq \epsilon$, where $F(\mathbf{x}^*) = 0$ is the optimal objective function value in this work, and ϵ is a very small positive value to be pre-given for different problems.
- **Success rate (S_r):** It is equal to the number of success runs over total runs. A success run means that within Max_NFEs the algorithm finds a solution \mathbf{x} satisfying $F(\mathbf{x}) - F(\mathbf{x}^*) \leq \epsilon$.
- **Convergence graphs:** The graphs show the *mean* RMSE performance of the total runs.

5.4.2. Compared with other DEs

The results of different performance criteria of different DE variants are respectively described in Tables 9, 10, and 11 for the single diode model, double diode model, and PV module. For the RMSE performance, the minimal, median, maximal, mean, and standard deviation over 100 runs are reported. In addition, to compare the significance of RMSE values between two algorithms, the paired Wilcoxon signed-rank test is used, and “+” indicates R_{cr}-IJADE is significantly better than its competitor according to the Wilcoxon signed-rank test at $\alpha = 0.05$. With respect to the NFEs_ε performance, the mean and standard deviation values are recorded. The convergence graphs of different DE variants are plotted in Figure 2.

Based on the results shown in Tables 9 - 11 and Figure 2, it can be observed that

- In terms of the RMSE values, the proposed R_{cr}-IJADE consistently gets the overall best results among different DE variants in all cases. According to the Wilcoxon’s test, R_{cr}-IJADE significantly surpasses other DE variants for different solar cell models. In addition, the standard deviation values of R_{cr}-IJADE are the smallest, which means that R_{cr}-IJADE is the most robust method compared with other five DE variants.
- For the NFEs_ε performance, for the single diode model and PV module, DEGL requires the smallest NFEs_ε values to reach the ϵ , followed by R_{cr}-IJADE. However, when the extracted parameters increase, for the double diode model, R_{cr}-IJADE is the best one, followed by DEGL. This phenomenon indicates that R_{cr}-IJADE has better scalability than DEGL.
- Considering the success rate S_r , it is clear that only R_{cr}-IJADE can consistently provide $S_r = 1.0$ for different solar cell models, while other DE methods can not successfully solve the parameter extraction problems in the single and double diode models in all runs. The highest success rate confirms the effectiveness and efficiency of our proposed R_{cr}-IJADE.
- With respect to the convergence speed, Figure 2 shows that in the early stage DEGL converges the fastest, followed by R_{cr}-IJADE, JADE, SaDE, jDE, and CoDE. However, DEGL stagnates quickly, which makes DEGL not obtain highly accurate final solutions. R_{cr}-IJADE is capable of successively converging toward the optimal solutions during the whole evolutionary process.

6. Conclusions

In this paper, an improved DE variant, R_{cr}-IJADE, is proposed to fast and accurately extract the parameters of solar cell models. R_{cr}-IJADE is the improved version of JADE, where the crossover rate repairing technique and ranking-based mutation are implemented to enhance the performance of JADE. In R_{cr}-IJADE, this is only one algorithmic parameter (μ) that needs to be pre-given by the user, which makes it be ease of use for real-world problems. R_{cr}-IJADE is comprehensively evaluated through the parameter extraction problems of different solar cell models, *i.e.*, single diode model, double diode model, and PV module. Moreover, the performance of R_{cr}-IJADE is indirectly compared with the reported results of different techniques and directly compared with other advanced DE variants based on different performance criteria. Experimental results confirm our expectation that the proposed R_{cr}-IJADE method is able to extract the parameters of different solar cell models fast and accurately. Additionally, compared with other methods, R_{cr}-IJADE can provide more precise and robust solutions, obtain higher success rate, and converge faster.

Therefore, R_{cr} -IJADE can be an efficient and reliable alternative for other complex optimization problems of solar cell models.

The source code of R_{cr} -IJADE can be obtained from the first author upon request.

References

- [1] A. Askarzadeh, A. Rezaezadeh, Artificial bee swarm optimization algorithm for parameters identification of solar cell models, *Applied Energy* 102 (0) (2013) 943 – 949.
- [2] K. Ishaque, Z. Salam, H. Taheri, A. Shamsudin, A critical evaluation of EA computational methods for photovoltaic cell parameter extraction based on two diode model, *Solar Energy* 85 (9) (2011) 1768 – 1779.
- [3] M. Wolf, G. Noel, R. Stirn, Investigation of the double exponential in the current-voltage characteristics of silicon solar cells, *IEEE Transactions on Electron Devices* 24 (4) (1977) 419 – 428.
- [4] M. Villalva, J. Gazoli, E. Filho, Comprehensive approach to modeling and simulation of photovoltaic arrays, *IEEE Transactions on Power Electronics* 24 (5) (2009) 1198 – 1208.
- [5] K. Ishaque, Z. Salam, H. Taheri, Simple, fast and accurate two-diode model for photovoltaic modules, *Solar Energy Materials and Solar Cells* 95 (2) (2011) 586 – 594.
- [6] A. J. Bühler, A. Krenzinger, Method for photovoltaic parameter extraction according to a modified double-diode model, *Progress in Photovoltaics: Research and Applications* In press.
- [7] M. AlHajri, K. El-Naggar, M. AlRashidi, A. Al-Othman, Optimal extraction of solar cell parameters using pattern search, *Renewable Energy* 44 (0) (2012) 238 – 245.
- [8] D. Chan, J. Phang, Analytical methods for the extraction of solar-cell single- and double-diode model parameters from I-V characteristics, *IEEE Transactions on Electron Devices* 34 (2) (1987) 286 – 293.
- [9] A. Ortiz-Conde, F. J. G. Sanchez, J. Muci, New method to extract the model parameters of solar cells from the explicit analytic solutions of their illuminated I-V characteristics, *Solar Energy Materials and Solar Cells* 90 (3) (2006) 352 – 361.
- [10] H. Saleem, S. Karmalkar, An analytical method to extract the physical parameters of a solar cell from four points on the illuminated J-V curve, *IEEE Electron Device Letters* 30 (4) (2009) 349 – 352.
- [11] A. Orioli, A. D. Gangi, A procedure to calculate the five-parameter model of crystalline silicon photovoltaic modules on the basis of the tabular performance data, *Applied Energy* 102 (0) (2013) 1160 – 1177.
- [12] J. A. Jervase, H. Bourdouce, A. Al-Lawati, Solar cell parameter extraction using genetic algorithms, *Meas. Sci. Technol.* 12 (2001) 1922 – 1925.
- [13] M. Chegaar, Z. Ouennoughi, F. Guechi, Extracting dc parameters of solar cells under illumination, *Vacuum* 75 (4) (2004) 367 – 372.
- [14] Y. Chen, X. Wang, D. Li, R. Hong, H. Shen, Parameters extraction from commercial solar cells I-V characteristics and shunt analysis, *Applied Energy* 88 (6) (2011) 2239 – 2244.
- [15] A. Mellit, S. A. Kalogirou, Artificial intelligence techniques for photovoltaic applications: A review, *Progress in Energy and Combustion Science* 34 (5) (2008) 574 – 632.
- [16] R. Baños, F. Manzano-Agugliaro, F. Montoya, C. Gil, A. Alcayde, J. Gómez, Optimization methods applied to renewable and sustainable energy: A review, *Renewable and Sustainable Energy Reviews* 15 (4) (2011) 1753 – 1766.
- [17] M. Zagrouba, A. Sellami, M. Bouaïcha, M. Ksouri, Identification of PV solar cells and modules parameters using the genetic algorithms: Application to maximum power extraction, *Solar Energy* 84 (5) (2010) 860 – 866.
- [18] M. Ye, X. Wang, Y. Xu, Parameter extraction of solar cells using particle swarm optimization, *Journal of Applied Physics* 105 (2009) 094502.
- [19] L. Sandrolini, M. Artioli, U. Reggiani, Numerical method for the extraction of photovoltaic module double-diode model parameters through cluster analysis, *Applied Energy* 87 (2) (2010) 442 – 451.
- [20] W. Huang, C. Jiang, L. Xue, D. Song, Extracting solar cell model parameters based on chaos particle swarm algorithm, in: *Electric Information and Control Engineering (ICEICE)*, 2011 International Conference on, 2011, pp. 398 – 402.
- [21] E. Q. B. Macabebe, C. J. Sheppard, E. E. van Dyk, Parameter extraction from I-V characteristics of PV devices, *Solar Energy* 85 (1) (2011) 12 – 18.
- [22] J. J. Soon, K.-S. Low, Photovoltaic model identification using particle swarm optimization with inverse barrier constraint, *IEEE Transactions on Power Electronics* 27 (9) (2012) 3975 – 3983.
- [23] W. da Costa, J. Fardin, D. Simonetti, L. V. Neto, Identification of photovoltaic model parameters by differential evolution, in: *Industrial Technology (ICIT)*, 2010 IEEE International Conference on, 2010, pp. 931 – 936.
- [24] K. Ishaque, Z. Salam, An improved modeling method to determine the model parameters of photovoltaic (PV) modules using differential evolution (DE), *Solar Energy* 85 (9) (2011) 2349 – 2359.
- [25] K. Ishaque, Z. Salam, S. Mekhilef, A. Shamsudin, Parameter extraction of solar photovoltaic modules using penalty-based differential evolution, *Applied Energy* 99 (0) (2012) 297 – 308.
- [26] M. AlRashidi, M. AlHajri, K. El-Naggar, A. Al-Othman, A new estimation approach for determining the I-V characteristics of solar cells, *Solar Energy* 85 (7) (2011) 1543 – 1550.
- [27] K. El-Naggar, M. AlRashidi, M. AlHajri, A. Al-Othman, Simulated annealing algorithm for photovoltaic parameters identification, *Solar Energy* 86 (1) (2012) 266 – 274.
- [28] A. Askarzadeh, A. Rezaezadeh, Parameter identification for solar cell models using harmony search-based algorithms, *Solar Energy* 86 (11) (2012) 3241 – 3249.
- [29] R. Storn, K. Price, Differential evolution—A simple and efficient heuristic for global optimization over continuous spaces, *J. of Global Optim.* 11 (4) (1997) 341–359.
- [30] K. Price, R. Storn, J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Springer-Verlag, Berlin, 2005.
- [31] S. Das, P. N. Suganthan, Differential evolution: A survey of the state-of-the-art, *IEEE Trans. on Evol. Comput.* 15 (1) (2011) 4–31.

- [32] J. Brest, S. Greiner, B. Bošković, M. Mernik, V. Žumer, Self-adapting control parameters in differential evolution: A comparative study on numerical benchmark problems, *IEEE Trans. on Evol. Comput.* 10 (6) (2006) 646–657.
- [33] A. K. Qin, V. L. Huang, P. N. Suganthan, Differential evolution algorithm with strategy adaptation for global numerical optimization, *IEEE Trans. on Evol. Comput.* 13 (2) (2009) 398–417.
- [34] J. Zhang, A. C. Sanderson, JADE: Adaptive differential evolution with optional external archive, *IEEE Trans. on Evol. Comput.* 13 (5) (2009) 945–958.
- [35] S. Das, A. Abraham, U. K. Chakraborty, A. Konar, Differential evolution using a neighborhood-based mutation operator, *IEEE Trans. on Evol. Comput.* 13 (3) (2009) 526–553.
- [36] Y. Wang, Z. Cai, Q. Zhang, Differential evolution with composite trial vector generation strategies and control parameters, *IEEE Trans. on Evol. Comput.* 15 (1) (2011) 55–66.
- [37] W. Gong, Z. Cai, Differential evolution with ranking-based mutation operators, *IEEE Transactions on Systems, Man, and Cybernetics: Part B – Cybernetics* (2013) 1 – 16In press.
- [38] J. Liu, J. Lampinen, A fuzzy adaptive differential evolution algorithm, *Soft Comput.* 9 (6) (2005) 448–462.
- [39] J. Zhang, A. C. Sanderson, *Adaptive Differential Evolution: A Robust Approach to Multimodal Problem Optimization*, Springer-Verlag, Berlin, 2009.
- [40] J. Arabas, A. Szczepankiewicz, T. Wroniak, Experimental comparison of methods to handle boundary constraints in differential evolution, in: *Parallel Problem Solving from Nature - PPSN XI*, Vol. 6239 of LNCS, 2010, pp. 411 – 420.
- [41] T. Easwarakhanthan, J. Bottin, I. Bouhouch, C. Boutrit, Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers, *International Journal of Solar Energy* 4 (1) (1986) 1–12.
- [42] K. Bouzidi, M. Chegaar, N. Nehaoua, New method to extract the parameters of solar cells from their illuminated I-V curve, in: *4th International Conference on Computer Integrated Manufacturing*, 2007.
- [43] M. Chegaar, N. Nehaoua, A. Bouhemadou, Organic and inorganic solar cells parameters evaluation from single icv plot, *Energy Conversion and Management* 49 (6) (2008) 1376 – 1379.