Parameter extraction of solar cell models using repaired adaptive differential evolution

Wenyin Gong∗, Zhihua Cai∗

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 Rcr-IJADE, which is an improved version of JADE. In Rcr-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 Rcr-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 Rcr-IJADE in terms of the quality of final solutions, success rate, and convergence speed. In addition, the simulated data with the extracted parameters of Rcr-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, 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. 20110145120009.

∗Corresponding author. Tel: +86-27-67883716.

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

Preprint submitted to Elsevier April 30, 2013
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]:

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

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

$$I_{d1} = I_{ad1} \left( \exp \left( \frac{V_L + I_{sh}R_s}{a_1V_t} \right) - 1 \right)$$  \(2\)
\[ I_{d2} = I_{sd2} \left( \exp \left( \frac{V_L + I_L R_s}{a_2 V_t} \right) - 1 \right) \]  

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

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

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 (\( 1.60217664 \times 10^{-19} \) C).

The shunt resistor current \( I_{sh} \) is formulated as

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

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:

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

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]:

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

### 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

\[ F(x) = \sqrt{\frac{1}{N} \sum_{k=1}^{N} f_k (V_L, I_L, x)^2} \]  

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

\[ f(V_L, I_L, x) = I_{ph} - I_{sd1} \left( \exp \left( \frac{V_L + I_L R_s}{a_1 V_t} \right) - 1 \right) \]  

\[ - I_{sd2} \left( \exp \left( \frac{V_L + I_L R_s}{a_2 V_t} \right) - 1 \right) \]  

\[ - \frac{V_L + I_L R_s}{R_{sh}} - I_L \]  

\[ x = \{ I_{ph}, I_{sd1}, I_{sd2}, R_s, R_{sh}, a_1, a_2 \} \]
Table 1: Ranges of the parameters of the double and single diode models.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ph}$ (A)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$I_{sd}$ (µA)</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$R_s$ (Ω)</td>
<td>0</td>
<td>0.5</td>
</tr>
<tr>
<td>$R_{sh}$ (Ω)</td>
<td>0</td>
<td>100</td>
</tr>
<tr>
<td>$a$</td>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Lower bound</th>
<th>Upper bound</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ph}$ (A)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>$I_{sd}$ (µA)</td>
<td>0</td>
<td>50</td>
</tr>
<tr>
<td>$R_s$ (Ω)</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>$R_{sh}$ (Ω)</td>
<td>0</td>
<td>2000</td>
</tr>
<tr>
<td>$a$</td>
<td>1</td>
<td>50</td>
</tr>
</tbody>
</table>

For the single diode model

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

$$x = \{I_{ph}, I_{sd}, R_s, R_{sh}, a\}$$  \hspace{1cm} (12)

$x$ 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 $F(x)$ 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 $\mu$ solutions (vectors). The population is initialized at random within the boundaries. For example, for the $i$-th vector $x_i$, it is initialized as follows:

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

where $L_j$ and $U_j$ are respectively the lower bound and upper bound of $x_j$, i.e., $x_j \in [L_j, U_j]$, $i = 1, \cdots, \mu$, $j = 1, \cdots, D$. $D$ is the number of decision variables. randreal(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 $x_i$, a mutant vector $v_i$ is generated by the mutation operator. For example, for the “DE/rand-to-best/1” mutation, it is formulated as:

$$v_i = x_{r_1} + F \cdot (x_{best} - x_{r_1}) + F \cdot (x_{r_2} - x_{r_3})$$

(14)

where $F$ is the mutation scaling factor, $r_1, r_2, r_3 \in \{1, \cdots, \mu\}$ are mutually different integers randomly generated, and $r_1 \neq r_2 \neq r_3 \neq i$. $x_{best}$ 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 $u_i$ between $x_i$ and $v_i$. The most commonly used operator is the binomial crossover performed on each component as follows:

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

(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:

$$x_i = \begin{cases} u_i, & \text{if } F(u_i) \leq F(x_i) \\ x_i, & \text{otherwise} \end{cases}$$

(16)

where $F(x)$ is the objective function to be optimized.

3.3. The JADE algorithm

As above-description, DE has three parameters, i.e., $\mu, Cr$, 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:

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

(17)

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

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

(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 $g$. 

\[5\]
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:

$$F_i = \text{rndc}([\mu_F, 0, 1])$$  \hspace{1cm} (19)

and then truncated to be 1.0 if $F_i > 1.0$ or regenerated if $F_i \leq 0$. $\text{rndc}([\mu_F, 0, 1])$ is a random number generated according to the Cauchy distribution with location parameter $\mu_F$ and scale parameter 0.1. The location parameter $\mu_F$ is updated in the following manner:

$$\mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}_l(S_F)$$  \hspace{1cm} (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:

$$\text{mean}_l(S_F) = \frac{\sum_{i=1}^{\mid S_F \mid} F_i^2}{\sum_{i=1}^{\mid S_F \mid} F_i}$$  \hspace{1cm} (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 $b_i$ be a binary string generated for each target vector $x_i$ as follows:

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

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

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

where $i = 1, \cdots, \mu$ and $j = 1, \cdots, D$. According to Equations (22) and (23), we can see that the binary string $b_{i,j}$ is stochastically related to $Cr$; however, the trial vector $u_i$ is directly related to its binary string $b_i$, 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

$$Cr'_{i} = \frac{\sum_{j=1}^{D} b_{i,j}}{D}$$  \hspace{1cm} (24)

where $b_i$ 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 $u_i$ is a successful vector, $Cr'_{i}$ will be stored in $S_{Cr_i}$ 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-pbest/1”) as

$$v_i = x_i + F_i \cdot (x^p_{\text{best}} - x_i) + F_i \cdot (x_{r_1} - x_i)$$  \hspace{1cm} (25)

where $x^p_{\text{best}}$ refers to the $p$th best solution, which is randomly selected from the top 100$p$% solutions, with $p \in (0, 1]$. $r_1, r_2, r_3 \in \{1, \cdots, \mu\}$ 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** \( \text{rndreal}(0,1) > pv_{r_1} \text{ or } r_1 == i \) **do**

Randomly select \( r_1 \in [1,\mu] \);

**while** \( \text{rndreal}(0,1) > pv_{r_2} \text{ or } r_2 == r_1 \text{ or } r_2 == i \) **do**

Randomly select \( r_2 \in [1,\mu] \);

Randomly select \( r_3 \in [1,\mu] \);

**while** \( r_3 == r_2 \text{ or } r_3 == r_1 \text{ or } r_3 == i \) **do**

Randomly select \( r_3 \in [1,\mu] \);

In Equation (25), since \( x_{r_1}, x_{r_2}, \) and \( x_{r_3} \) 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 \( x_i \) in the sorted population is assigned as

\[
R_i = \mu - i, \quad i = 1, 2, \ldots, \mu
\]  

(26)

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

\[
pv_i = \left(\frac{R_i}{\mu}\right)^2
\]  

(27)

Finally, the vectors \( (x_{r_1} \text{ and } x_{r_2}) \) 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 \( x_{r_1} \) and \( x_{r_2} \) 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., \( x_j \not\in [L_j, U_j] \), if that, the following boundary-handling technique is applied:

\[
x_j = L_j + \text{rndreal}(0,1) \cdot (U_j - L_j)
\]  

(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 (\( \mu \) and Max_NFEs) that need to be given by the user, while other parameters (i.e., \( \mu_{cr}, \mu_{F}, c, p \)) 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)
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.

### 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 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...
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 Rcr-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.

In addition, the I-V characteristic obtained by Rcr-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 Rcr-IJADE are highly coincide with the experimental data in the single diode model, which means that the extracted parameters of Rcr-IJADE are very accurate.

### Table 4: Curve fitting results of Rcr-IJADE for the single diode model.

<table>
<thead>
<tr>
<th>Item</th>
<th>$V_L$ (V)</th>
<th>$I_L$ measured (A)</th>
<th>$I_L$ calculated (A)</th>
<th>IAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.2057</td>
<td>0.7640</td>
<td>0.76409559</td>
<td>0.00009559</td>
</tr>
<tr>
<td>2</td>
<td>-0.1291</td>
<td>0.7620</td>
<td>0.76206611</td>
<td>0.00006611</td>
</tr>
<tr>
<td>3</td>
<td>-0.0588</td>
<td>0.7655</td>
<td>0.76124731</td>
<td>0.00024731</td>
</tr>
<tr>
<td>4</td>
<td>0.0057</td>
<td>0.7655</td>
<td>0.76014966</td>
<td>0.00035034</td>
</tr>
<tr>
<td>5</td>
<td>0.0646</td>
<td>0.7660</td>
<td>0.75935032</td>
<td>0.0004528</td>
</tr>
<tr>
<td>6</td>
<td>0.1185</td>
<td>0.7590</td>
<td>0.75804472</td>
<td>0.0005528</td>
</tr>
<tr>
<td>7</td>
<td>0.1678</td>
<td>0.7570</td>
<td>0.75709510</td>
<td>0.00059510</td>
</tr>
<tr>
<td>8</td>
<td>0.2132</td>
<td>0.7570</td>
<td>0.75520951</td>
<td>0.0014495</td>
</tr>
<tr>
<td>9</td>
<td>0.2545</td>
<td>0.7555</td>
<td>0.75508177</td>
<td>0.00048177</td>
</tr>
<tr>
<td>10</td>
<td>0.2924</td>
<td>0.7540</td>
<td>0.75230288</td>
<td>0.00020288</td>
</tr>
<tr>
<td>11</td>
<td>0.3269</td>
<td>0.7505</td>
<td>0.74935473</td>
<td>0.00085473</td>
</tr>
<tr>
<td>12</td>
<td>0.3585</td>
<td>0.7505</td>
<td>0.75014966</td>
<td>0.00035034</td>
</tr>
<tr>
<td>13</td>
<td>0.3873</td>
<td>0.7535</td>
<td>0.75230288</td>
<td>0.0014230</td>
</tr>
<tr>
<td>14</td>
<td>0.4137</td>
<td>0.7560</td>
<td>0.75280447</td>
<td>0.00032471</td>
</tr>
<tr>
<td>15</td>
<td>0.4373</td>
<td>0.7560</td>
<td>0.75459510</td>
<td>0.0010451</td>
</tr>
<tr>
<td>16</td>
<td>0.4590</td>
<td>0.7560</td>
<td>0.75320951</td>
<td>0.00049510</td>
</tr>
<tr>
<td>17</td>
<td>0.5265</td>
<td>0.7560</td>
<td>0.75135473</td>
<td>0.00085473</td>
</tr>
<tr>
<td>18</td>
<td>0.5398</td>
<td>0.7535</td>
<td>0.75124731</td>
<td>0.00024731</td>
</tr>
<tr>
<td>19</td>
<td>0.5523</td>
<td>0.7535</td>
<td>0.75014966</td>
<td>0.00035034</td>
</tr>
<tr>
<td>20</td>
<td>0.5755</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>21</td>
<td>0.5990</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>22</td>
<td>0.6225</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>23</td>
<td>0.6458</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>24</td>
<td>0.6692</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>25</td>
<td>0.6927</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
<tr>
<td>26</td>
<td>0.7162</td>
<td>0.75135473</td>
<td>0.75132473</td>
<td>0.00012473</td>
</tr>
</tbody>
</table>

5.2. Results on double diode model

In this model, seven parameters need to be extracted. The results of Rcr-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 Rcr-IJADE are reported in Table 6. Additionally, the I-V characteristic of Rcr-IJADE is plotted in Figure 1(b). Similar to the results in the single diode model, in this case, our proposed Rcr-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 Rcr-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.

<table>
<thead>
<tr>
<th>Item</th>
<th>V_L (V)</th>
<th>I_L measured (A)</th>
<th>I_L calculated (A)</th>
<th>IAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.2057</td>
<td>0.7640</td>
<td>0.76409268</td>
<td>0.00009268</td>
</tr>
<tr>
<td>2</td>
<td>-0.1291</td>
<td>0.7620</td>
<td>0.76265394</td>
<td>0.00065394</td>
</tr>
<tr>
<td>3</td>
<td>-0.0588</td>
<td>0.7605</td>
<td>0.76135758</td>
<td>0.00035758</td>
</tr>
<tr>
<td>4</td>
<td>0.0057</td>
<td>0.7605</td>
<td>0.76016253</td>
<td>0.00031253</td>
</tr>
<tr>
<td>5</td>
<td>0.0646</td>
<td>0.7600</td>
<td>0.75902000</td>
<td>0.00094000</td>
</tr>
<tr>
<td>6</td>
<td>0.1185</td>
<td>0.7590</td>
<td>0.75803065</td>
<td>0.00094935</td>
</tr>
<tr>
<td>7</td>
<td>0.1678</td>
<td>0.7570</td>
<td>0.75709635</td>
<td>0.00009635</td>
</tr>
<tr>
<td>8</td>
<td>0.2132</td>
<td>0.7570</td>
<td>0.75614465</td>
<td>0.00085535</td>
</tr>
<tr>
<td>9</td>
<td>0.2545</td>
<td>0.7555</td>
<td>0.75508115</td>
<td>0.00041885</td>
</tr>
<tr>
<td>10</td>
<td>0.2924</td>
<td>0.7540</td>
<td>0.75366874</td>
<td>0.00033126</td>
</tr>
<tr>
<td>11</td>
<td>0.3269</td>
<td>0.7505</td>
<td>0.75139511</td>
<td>0.00089511</td>
</tr>
<tr>
<td>12</td>
<td>0.3585</td>
<td>0.7465</td>
<td>0.74734939</td>
<td>0.00084939</td>
</tr>
<tr>
<td>13</td>
<td>0.3873</td>
<td>0.7385</td>
<td>0.74010214</td>
<td>0.00160214</td>
</tr>
<tr>
<td>14</td>
<td>0.4137</td>
<td>0.7280</td>
<td>0.72738784</td>
<td>0.00081216</td>
</tr>
<tr>
<td>15</td>
<td>0.4373</td>
<td>0.7065</td>
<td>0.70695162</td>
<td>0.00045162</td>
</tr>
<tr>
<td>16</td>
<td>0.4590</td>
<td>0.6755</td>
<td>0.67530112</td>
<td>0.00019888</td>
</tr>
<tr>
<td>17</td>
<td>0.4784</td>
<td>0.6320</td>
<td>0.63088766</td>
<td>0.00111233</td>
</tr>
<tr>
<td>18</td>
<td>0.4960</td>
<td>0.5730</td>
<td>0.57207477</td>
<td>0.00092523</td>
</tr>
<tr>
<td>19</td>
<td>0.5119</td>
<td>0.4990</td>
<td>0.49949417</td>
<td>0.00049417</td>
</tr>
<tr>
<td>20</td>
<td>0.5265</td>
<td>0.4130</td>
<td>0.41349125</td>
<td>0.00049125</td>
</tr>
<tr>
<td>21</td>
<td>0.5398</td>
<td>-0.0100</td>
<td>-0.00924461</td>
<td>0.00075539</td>
</tr>
<tr>
<td>22</td>
<td>0.5521</td>
<td>-0.1230</td>
<td>-0.12437667</td>
<td>0.00137667</td>
</tr>
</tbody>
</table>

Sum of IAE 0.01770933

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

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>I_p (A)</td>
<td>1.0318</td>
<td>1.0339</td>
<td>1.0286</td>
<td>1.0310</td>
<td>1.0310</td>
<td>1.0313</td>
<td>1.030514</td>
</tr>
<tr>
<td>I_{sd} (µA)</td>
<td>3.2875</td>
<td>3.0760</td>
<td>8.3010</td>
<td>3.8236</td>
<td>3.1756</td>
<td>3.6642</td>
<td>3.482263</td>
</tr>
<tr>
<td>R_s (Ω)</td>
<td>1.2057</td>
<td>1.2030</td>
<td>1.0920</td>
<td>1.0420</td>
<td>1.0420</td>
<td>1.0420</td>
<td>1.0420</td>
</tr>
<tr>
<td>R_sh (Ω)</td>
<td>555.5556</td>
<td>555.5556</td>
<td>1850.1000</td>
<td>689.6600</td>
<td>714.2577</td>
<td>833.3333</td>
<td>981.982240</td>
</tr>
<tr>
<td>a</td>
<td>48.4500</td>
<td>48.1862</td>
<td>52.2430</td>
<td>48.9300</td>
<td>48.2889</td>
<td>48.8211</td>
<td>48.642535</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.7805</td>
<td>0.6130</td>
<td>0.0035</td>
<td>0.0102</td>
<td>0.0118</td>
<td>0.0027</td>
<td>0.002425</td>
</tr>
</tbody>
</table>

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 MaxNFEs 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., N = 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++.
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 9: Comparison on different performance criteria among different DE variants for the single diode model. "NA" means not available.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Min</th>
<th>Median</th>
<th>Mean</th>
<th>Std</th>
<th>sig</th>
</tr>
</thead>
<tbody>
<tr>
<td>JADE</td>
<td>1.01467E-03</td>
<td>1.221579E-03</td>
<td>1.674985E-03</td>
<td>1.26149E-03</td>
<td>1.48E-04</td>
</tr>
<tr>
<td>CoDE</td>
<td>9.972754E-04</td>
<td>1.140593E-03</td>
<td>1.864976E-03</td>
<td>1.17580E-03</td>
<td>9.26E-05</td>
</tr>
<tr>
<td>DEGL</td>
<td>1.101999E-03</td>
<td>1.974311E-03</td>
<td>2.167376E-03</td>
<td>1.94597E-03</td>
<td>1.93E-04</td>
</tr>
<tr>
<td>jDE</td>
<td>1.66225E-03</td>
<td>2.08725</td>
<td>2.075882</td>
<td>2.001429</td>
<td></td>
</tr>
<tr>
<td>DEGL</td>
<td>9.860219E-04</td>
<td>1.084227E-03</td>
<td>1.872544E-03</td>
<td>1.162104E-03</td>
<td>2.06E-04</td>
</tr>
<tr>
<td>JADE</td>
<td>1.48088E-04</td>
<td>0.2000</td>
<td>0.2000</td>
<td>0.2000</td>
<td></td>
</tr>
</tbody>
</table>

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:** The NFEs is used to record the number of function evaluations in each run for finding a solution \( x \) satisfying \( F(x) - F(x^*) \leq \varepsilon \), where \( F(x^*) = 0 \) is the optimal objective function value in this work, and \( \varepsilon \) 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 \( \text{Max}_{\text{NFEs}} \) the algorithm finds a solution \( x \) satisfying \( F(x) - F(x^*) \leq \varepsilon \).

• **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 \( R_c-IJADE \) is significantly better than its competitor according to the Wilcoxon signed-rank test at \( \alpha = 0.05 \). With respect to the NFEs 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 \( R_c-IJADE \) consistently gets the overall best results among different DE variants in all cases. According to the Wilcoxon’s test, \( R_c-IJADE \) significantly surpasses other DE variants for different solar cell models. In addition, the standard deviation values of \( R_c-IJADE \) are the smallest, which means that \( R_c-IJADE \) is the most robust method compared with other five DE variants.

- For the NFEs performance, for the single diode model and PV module, DEGL requires the smallest NFEs values to reach the \( \varepsilon \), followed by \( R_c-IJADE \). However, when the extracted parameters increase, for the double diode model, \( R_c-IJADE \) is the best one, followed by DEGL. This phenomenon indicates that \( R_c-IJADE \) has better scalability than DEGL.

- Considering the success rate \( S_r \), it is clear that only \( R_c-IJADE \) 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 \( R_c-IJADE \).

- With respect to the convergence speed, Figure 2 shows that in the early stage DEGL converges the fastest, followed by \( R_c-IJADE \), JADE, SaDE, jDE, and CoDE. However, DEGL stagnates quickly, which makes DEGL not obtain highly accurate final solutions. \( R_c-IJADE \) is capable of successively converging toward the optimal solutions during the whole evolutionary process.

6. Conclusions

In this paper, an improved DE variant, \( R_c-IJADE \), is proposed to fast and accurately extract the parameters of solar cell models. \( R_c-IJADE \) 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 \( R_c-IJADE \), this is only one algorithmic parameter (\( \mu \)) that needs to be pre-given by the user, which makes it be ease of use for real-world problems. \( R_c-IJADE \) 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 \( R_c-IJADE \) 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 \( R_c-IJADE \) method is able to extract the parameters of different solar cell models fast and accurately. Additionally, compared with other methods, \( R_c-IJADE \) 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


