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, PR China

Received 24 January 2013; received in revised form 4 May 2013; accepted 11 May 2013

Available online 10 June 2013

Communicated by: Associate Editor Arturo Morales-Acevedo

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 $R_c$-IJADE, which is an improved version of JADE. In $R_c$-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 $R_c$-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 $R_c$-IJADE in terms of the quality of final solutions, success rate, and convergence speed. In addition, the simulated data with the extracted parameters of $R_c$-IJADE are in very good agreement with the experimental data in all cases.

© 2013 Elsevier Ltd. All rights reserved.

Keywords: 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 (Askarzadeh and Rezzazadeh, 2013). For PV systems, it is very important to select a model to closely emulate the characteristics of PV cells (Ishaque et al., 2011). Several models have been introduced to describe the current–voltage ($I-V$) relationship in solar cells over recent years (Wolf et al., 1977; Villalva et al., 2009; Ishaque et al., 2011; Bühler and Krenzinger, in press). In practice, two main equivalent circuit models have been widely used: single and double diode models (AlHajri et al., 2012). 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 (Chan and Phang, 1987; Ortiz-Conde et al., 2006; Saleem and Karmalkar, 2009; Bühler and Krenzinger, in press; Orioli and Gangi, 2013) and (ii) numerical approaches (Jervase et al., 2001; Chegaar et al., 2004; Chen et al., 2011). 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 (Mellit and Kalogirou, 2008; Baños et al., 2011), such as genetic algorithms (GAs) (Jervase et al., 2001; Zagrouba et al., 2010), particle swarm optimization (PSO) (Ye et al., 2009; Sandrolini et al., 2010; Huang et al., 2011; Macabbe et al., 2011; Soon and Low, 2012), differential evolution (DE) (da Costa et al., 2010; Ishaque and Salam, 2011; Ishaque et al., 2011; Ishaque et al., 2012), pattern search (PS) (AlRashidi et al., 2011; AlHajri et al., 2012), simulated annealing (SA) (El-Naggar et al., 2012), harmony search (HS) (Askarzadeh and Reza zadeh, 2012), and artificial bee swarm optimization (ABSO) (Askarzadeh and Reza zadeh, 2013).

Differential evolution (DE), proposed by Storn and Price in 1997 Storn and Price (1997), 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, and engineering design (Price et al., 2005). More applications can be found in the survey paper (Das and Suganthan, 2011), 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 (Brest et al., 2006), SaDE (Qin et al., 2009), JADE (Zhang and Sanderson, 2009), DEGL (Das et al., 2009), and CoDE (Wang et al., 2011). The advanced DE variants obtain better performance than the original DE algorithm through benchmark functions. As mentioned above, in da Costa et al. (2010), Ishaque and Salam (2011), Ishaque et al. (2011), and Ishaque et al. (2012), 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 (Zhang and Sanderson, 2009), 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 Gong and Cai (in press) 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 (Wolf et al., 1977; AlRashidi et al., 2011):

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

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:

\[
I_{d1} = I_{sd1}
\left( \exp \left( \frac{V_L + I_{ph} R_s}{a_1 V_f} \right) - 1 \right)
\]

\[
I_{d2} = I_{sd2}
\left( \exp \left( \frac{V_L + I_{ph} R_s}{a_2 V_f} \right) - 1 \right)
\]
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 $V_i$ is the junction thermal voltage as

$$V_i = \frac{kT}{q}$$  \hspace{1cm} (4)

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

The shunt resistor current $I_{sh}$ is formulated as

$$I_{sh} = \frac{V_L + I_L R_s}{R_{sh}}$$  \hspace{1cm} (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:

$$I_L = I_{ph} - I_{sd} \left( \exp \frac{V_L + I_L R_s}{a V_i} - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}}$$  \hspace{1cm} (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 (Soon and Low, 2012):

$$I_L = I_{ph} - I_{sd} \left( \exp \frac{V_L + I_L R_s}{a N_s V_i} - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}}$$  \hspace{1cm} (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 Ye et al. (2009), Ishaque et al. (2012), Askarzadeh and Rezazadeh (2013), the root mean square error (RMSE) is used as the objective function, which is described as

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

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

$$f(V_L, I_L, x) = I_{ph} - I_{sd1} \left( \exp \frac{V_L + I_L R_s}{a_1 V_i} - 1 \right) - I_{sd2} \left( \exp \frac{V_L + I_L R_s}{a_2 V_i} - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}} - I_L$$  \hspace{1cm} (9)

$$x = \{I_{ph}, I_{sd1}, I_{sd2}, R_s, R_{sh}, a_1, a_2\}$$  \hspace{1cm} (10)

For the single diode model

$$f(V_L, I_L, x) = I_{ph} - I_{sd} \left( \exp \frac{V_L + I_L R_s}{a V_i} - 1 \right) - \frac{V_L + I_L R_s}{R_{sh}} - I_L$$  \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 Askarzadeh and Rezazadeh (2013). 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 Eq. (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 Storn and Price (1997), 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:

<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 1

Ranges of the parameters of the double and single diode models.
\[ x_{ij} = L_j + \text{rndreal}(0, 1) \cdot (U_j - L_j) \]  

(13)

where \( L_i \) and \( U_i \) are respectively the lower bound and upper bound of \( x_i \), i.e., \( x_i \in [L_i, U_i] \), \( i = 1, \ldots, \mu, j = 1, \ldots, D \). \( D \) 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 \( x_i \), 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:

\[ v_i = x_i + F \cdot (x_{\text{best}} - x_i) + F \cdot (x_j - x_i) \]  

(14)

where \( F \) is the mutation scaling factor, \( r_1, r_2, r_3 \in \{1, \ldots, \mu\} \) are mutually different integers randomly generated, and \( r_1 \neq r_2 \neq r_3 \neq i \). \( x_{\text{best}} \) 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:

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

(15)

where \( Cr \) is the crossover rate and \( j_{\text{rand}} \) 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 (Liu and Lampinen, 2005; Brest et al., 2006) 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 Zhang and Sanderson (2009), 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:

\[ Cr_i = \text{rndc}((\mu_C), 0.1) \]  

(17)

and truncated to the interval \([0, 1]\). Where \( \mu_C \) is the mean value to generate \( Cr_i \). It is updated as follows:

\[ \mu_C = (1 - c) \cdot \mu_C + c \cdot \text{mean}(S_{Cr}) \]  

(18)

where \( c \) is a constant in \([0, 1]\); \( \text{mean}(\cdot) \) is the 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:

\[ F_i = \text{rndc}((\mu_F), 0.1) \]  

(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:

\[ \mu_F = (1 - c) \cdot \mu_F + c \cdot \text{mean}(S_F) \]  

(20)

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

\[ \text{mean}_L(S_F) = \frac{\sum_{i=1}^{n}(S_i)^{2}}{\sum_{i=1}^{n}S_i} \]  

(21)

### 4. Repaired JADE: \( R_{Cr}-\text{IJADE} \)

In order to fast and accurately extract the parameters of different solar cell models, we propose the improved JADE method, i.e., \( R_{Cr}-\text{IJADE} \). In \( R_{Cr}-\text{IJADE} \), 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 Eq. (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:
\[
b_{i,j} = \begin{cases} 1, & \text{if } (\text{rand}(0,1) < Cr \text{ or } j = j_{\text{rand}}) \\ 0, & \text{otherwise} \end{cases}
\]  

Therefore, the binomial crossover of DE in Eq. (15) can be reformulated as
\[
u_{i,j} = b_{i,j} \cdot v_{i,j} + (1 - b_{i,j}) \cdot x_{i,j}
\]
where \( i = 1, \ldots, \mu \) and \( j = 1, \ldots, D \). According to Eqs. (22) and (23), we can see that the binary string \( b_{i,j} \) is stochastically related to \( Cr \); however, the trial vector \( u_{i,j} \) is directly related to its binary string \( b_{i,j} \), 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
\[
Cr_{i} = \frac{\sum_{j=1}^{D} b_{i,j}}{D}
\]
where \( b_{i,j} \) is the binary string calculated in Eq. (22). The crossover rate is repaired after its binary string is generated by Eq. (22) based on \( Cri \). If the trial vector \( u_{i} \) is a successful vector, \( Cr_{i} \) will be stored in \( SCr \), instead of storing \( Cri \).

**Algorithm 1.** Ranking-based vector selection

**Input:** The target vector index \( i \)
**Output:** The selected vector index \( r_{1}, r_{2}, r_{3} \)
while \( \text{rand}(0,1) > \text{pr}_{1} \) or \( r_{1} == i \) do
- Randomly select \( r_{1} \in [1, \mu] \);
Randomly select \( r_{2} \in [1, \mu] \);
while \( \text{rand}(0,1) > \text{pr}_{2} \) or \( r_{2} == i \) or \( r_{2} == j_{\text{rand}} \) 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 \( r_{1} == i \) do
- Randomly select \( r_{3} \in [1, \mu] \);

\[
v_{i} = x_{r_{1}} + F_{i} \cdot (x_{\text{best}}^{p} - x_{r_{1}}) + F_{i} \cdot (x_{r_{2}} - x_{r_{3}})
\]

where \( x_{\text{best}}^{p} \) refers to the \( p \)-th solution, which is randomly selected from the top \( 100\% \) solutions, with \( p \in (0, 1) \). \( r_{1}, r_{2}, r_{3} \in \{1, \ldots, \mu\} \) and \( r_{1} \neq r_{2} \neq r_{3} \neq i \).

In Eq. (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 (Gong and Cai, in press) 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
\[
R_{i} = \frac{\mu}{i}
\]
After that, the selection probability of each vector is calculated as follows:
\[
p_{vi} = \left( \frac{R_{i}}{\mu} \right)^{2}
\]
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-best/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:
\[
x_{j} = L_{j} + \text{rand}(0,1) \cdot (U_{j} - L_{j})
\]
Actually, it is the **reinitialization** method as mentioned in Arabas et al. (2010).

**Algorithm 2.** The pseudo-code of the \( R_{cr}\)-IJADE algorithm
4.4. The R cr-IJADE algorithm

Combining the parameter adaptation techniques presented in JADE (Zhang and Sanderson, 2009) 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

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

<table>
<thead>
<tr>
<th>Item</th>
<th>GA (AlRashidi et al., 2011)</th>
<th>CPSO (Huang et al., 2011)</th>
<th>PS (AlHajri et al., 2012)</th>
<th>SA (El-Naggar et al., 2012)</th>
<th>IGHS (Askarzadeh and Rezazadeh, 2012)</th>
<th>ABSO (Askarzadeh and Rezazadeh, 2013)</th>
<th>R cr-IJADE</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{ph}$ (A)</td>
<td>0.7619</td>
<td>0.7607</td>
<td>0.7617</td>
<td>0.7620</td>
<td>0.7608</td>
<td>0.7608</td>
<td>0.760776</td>
</tr>
<tr>
<td>$I_{sd}$ (μA)</td>
<td>0.8087</td>
<td>0.4000</td>
<td>0.9980</td>
<td>0.4798</td>
<td>0.3435</td>
<td>0.3062</td>
<td>0.323021</td>
</tr>
<tr>
<td>$R_{s}$ (Ω)</td>
<td>0.0299</td>
<td>0.0354</td>
<td>0.0313</td>
<td>0.0345</td>
<td>0.0361</td>
<td>0.0366</td>
<td>0.036377</td>
</tr>
<tr>
<td>$R_{sh}$ (X)</td>
<td>42.3729</td>
<td>59.0120</td>
<td>64.1026</td>
<td>43.1034</td>
<td>53.2845</td>
<td>52.2903</td>
<td>53.718526</td>
</tr>
<tr>
<td>$a$</td>
<td>1.5751</td>
<td>1.5033</td>
<td>1.6000</td>
<td>1.5172</td>
<td>1.4874</td>
<td>1.4758</td>
<td>1.481184</td>
</tr>
<tr>
<td>RMSE</td>
<td>0.01908</td>
<td>0.00139</td>
<td>0.01494</td>
<td>0.01900</td>
<td>9.9306E−04</td>
<td>9.9124E−04</td>
<td>9.8602E−04</td>
</tr>
</tbody>
</table>

Fig. 1. Comparison on the I−V characteristics between the experimental data and simulated data obtained by R cr-IJADE for (a): the single diode model, (b): the double diode model, and (c): the PV module.
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., μc, μr, c, p) are kept the default values as used in JADE (Zhang and Sanderson, 2009). This makes Rcr-IJADE be easy to be applied to real-world applications.

5. Experimental results and analysis

In this section, the performance of Rcr-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 Easwarakhanthan et al. (1986), 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 μ = 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. Rcr-IJADE is coded in standard C++.

In order to show the superior performance of Rcr-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, Rcr-IJADE is directly compared with other advanced DE variants.
Table 7
Comparison among different parameter extraction techniques for the PV module.

<table>
<thead>
<tr>
<th>Item</th>
<th>Newton (Easwarakhanthan et al., 1986)</th>
<th>Method in (Bouzidi et al., 2007)</th>
<th>CPSO (Huang et al., 2011)</th>
<th>Method in (Chegara et al., 2008)</th>
<th>PS (AlHajri et al., 2012)</th>
<th>SA (El-Naggar et al., 2012)</th>
<th>Rcr-IJADE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iph (A)</td>
<td>1.0318</td>
<td>1.0339</td>
<td>1.0286</td>
<td>1.0310</td>
<td>1.0313</td>
<td>1.0313</td>
<td>1.030514</td>
</tr>
<tr>
<td>Isd (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>Rl (Ω)</td>
<td>1.2057</td>
<td>1.2030</td>
<td>1.0755</td>
<td>1.0920</td>
<td>1.2053</td>
<td>1.1989</td>
<td>1.201271</td>
</tr>
<tr>
<td>Rsh (Ω) 555.5556</td>
<td>555.5556</td>
<td>1850.1000</td>
<td>689.6600</td>
<td>714.2857</td>
<td>833.3333</td>
<td>981.98240</td>
<td></td>
</tr>
<tr>
<td>μ (cm²/Vs)</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>48.642835</td>
</tr>
<tr>
<td>RMSE 0.7805</td>
<td>0.6130</td>
<td>0.0035</td>
<td>0.0102</td>
<td>0.0118</td>
<td>0.0627</td>
<td>0.002425</td>
<td></td>
</tr>
</tbody>
</table>

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 Rcr-IJADE are compared with those of GA (AlRashidi et al., 2011), CPSO (Huang et al., 2011), PS (AlHajri et al., 2012), SA (El-Naggar et al., 2012), IGHS (Askarzadeh and Rezazadeh, 2012), and ABSO (Askarzadeh and Rezazadeh, 2013). 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 **bold face** and *italic face*, respectively.

From Table 3, it can be observed that Rcr-IJADE provides the best RMSE value among all compared methods, followed by IGHS, CPSO, PS, SA, and GA. Although the RMSE values of ABSO and IGHS are very close to that of Rcr-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 in improvement in the knowledge about the real values of the parameters (AlRashidi et al., 2011). 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) (AlHajri et al., 2012) between the experimental data and simulated data are shown in Fig. 1a 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.

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 (AlHajri et al., 2012), SA (El-Naggar et al., 2012), IGHS (Askarzadeh and Rezazadeh, 2012), and ABSO (Askarzadeh and Rezazadeh, 2013). 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 Fig. 1b. 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 Fig. 1b.

5.3. Results on PV module

For the PV module, the experimental results are reported in Table 7, where Rcr-IJADE is compared with Newton (Easwarakhanthan et al., 1986), method in Bouzidi et al. (2007), CPSO (Huang et al., 2011), method in Chegara et al. (2008), PS (AlHajri et al., 2012), and SA

Table 8
Curve fitting results of Rcr-IJADE for the PV module.

<table>
<thead>
<tr>
<th>Item</th>
<th>Vl (V)</th>
<th>Il measured (A)</th>
<th>Il calculated (A)</th>
<th>IAE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1248</td>
<td>1.0315</td>
<td>1.02912049</td>
<td>0.00237951</td>
</tr>
<tr>
<td>2</td>
<td>1.8093</td>
<td>1.0300</td>
<td>1.02738564</td>
<td>0.00261436</td>
</tr>
<tr>
<td>3</td>
<td>3.3511</td>
<td>1.0260</td>
<td>1.02573499</td>
<td>0.00026501</td>
</tr>
<tr>
<td>4</td>
<td>4.7622</td>
<td>1.0220</td>
<td>1.02490557</td>
<td>0.00209557</td>
</tr>
<tr>
<td>5</td>
<td>6.0538</td>
<td>1.0180</td>
<td>1.02227575</td>
<td>0.00427575</td>
</tr>
<tr>
<td>6</td>
<td>7.2364</td>
<td>1.0155</td>
<td>1.01991719</td>
<td>0.00441719</td>
</tr>
<tr>
<td>7</td>
<td>8.3189</td>
<td>1.0140</td>
<td>1.01635389</td>
<td>0.00235389</td>
</tr>
<tr>
<td>8</td>
<td>9.3097</td>
<td>1.0100</td>
<td>1.01048191</td>
<td>0.00048191</td>
</tr>
<tr>
<td>9</td>
<td>10.2163</td>
<td>1.0035</td>
<td>1.00068707</td>
<td>0.00281293</td>
</tr>
<tr>
<td>10</td>
<td>11.0449</td>
<td>0.9880</td>
<td>0.98465514</td>
<td>0.00334846</td>
</tr>
<tr>
<td>11</td>
<td>11.8018</td>
<td>0.9630</td>
<td>0.95964925</td>
<td>0.00330575</td>
</tr>
<tr>
<td>12</td>
<td>12.4929</td>
<td>0.9255</td>
<td>0.92305160</td>
<td>0.00244840</td>
</tr>
<tr>
<td>13</td>
<td>13.1231</td>
<td>0.8725</td>
<td>0.87588829</td>
<td>0.00008829</td>
</tr>
<tr>
<td>14</td>
<td>13.6983</td>
<td>0.8075</td>
<td>0.80731392</td>
<td>0.00018608</td>
</tr>
<tr>
<td>15</td>
<td>14.2221</td>
<td>0.7265</td>
<td>0.72796294</td>
<td>0.00146294</td>
</tr>
<tr>
<td>16</td>
<td>14.6995</td>
<td>0.6345</td>
<td>0.63464364</td>
<td>0.00196347</td>
</tr>
<tr>
<td>17</td>
<td>15.1346</td>
<td>0.5345</td>
<td>0.53569189</td>
<td>0.00119189</td>
</tr>
<tr>
<td>18</td>
<td>15.5311</td>
<td>0.4275</td>
<td>0.42882216</td>
<td>0.00132216</td>
</tr>
<tr>
<td>19</td>
<td>15.8929</td>
<td>0.3185</td>
<td>0.31867170</td>
<td>0.00017170</td>
</tr>
<tr>
<td>20</td>
<td>16.2229</td>
<td>0.2085</td>
<td>0.20785189</td>
<td>0.00064811</td>
</tr>
<tr>
<td>21</td>
<td>16.5241</td>
<td>0.1010</td>
<td>0.09835838</td>
<td>0.00264162</td>
</tr>
<tr>
<td>22</td>
<td>16.7987</td>
<td>−0.0080</td>
<td>−0.00817367</td>
<td>0.00017367</td>
</tr>
<tr>
<td>23</td>
<td>17.0499</td>
<td>−0.1110</td>
<td>−0.11096908</td>
<td>0.00003092</td>
</tr>
<tr>
<td>24</td>
<td>17.2793</td>
<td>−0.2090</td>
<td>−0.20912100</td>
<td>0.00012100</td>
</tr>
<tr>
<td>25</td>
<td>17.4885</td>
<td>−0.3030</td>
<td>−0.30202427</td>
<td>0.00097573</td>
</tr>
</tbody>
</table>

Sum of IAE 0.04177271
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>RMSE</th>
<th>NFEs (ε = 0.001)</th>
<th>Sr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Median</td>
<td>Max</td>
</tr>
<tr>
<td>jDE</td>
<td>1.011467E−03</td>
<td>1.221579E−03</td>
<td>1.674285E−03</td>
</tr>
<tr>
<td>SaDE</td>
<td>9.977254E−04</td>
<td>1.146093E−03</td>
<td>1.386479E−03</td>
</tr>
<tr>
<td>CoDE</td>
<td>1.101993E−03</td>
<td>1.494410E−03</td>
<td>2.081250E−03</td>
</tr>
<tr>
<td>DEGL</td>
<td>9.860219E−04</td>
<td>1.084221E−03</td>
<td>1.827544E−03</td>
</tr>
<tr>
<td>JADE</td>
<td>9.860219E−04</td>
<td>9.889961E−04</td>
<td>1.142034E−03</td>
</tr>
</tbody>
</table>

“+” indicates R_cr-IJADE is significantly better than its competitor according to the Wilcoxon signed-rank test at α = 0.05, hereinafter.

Comparison on different performance criteria among different DE variants for the double diode model. “NA” means not available.

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMSE</th>
<th>NFEs (ε = 0.001)</th>
<th>Sr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Median</td>
<td>Max</td>
</tr>
<tr>
<td>jDE</td>
<td>1.070336E−03</td>
<td>1.470803E−03</td>
<td>2.114012E−03</td>
</tr>
<tr>
<td>SaDE</td>
<td>9.917107E−04</td>
<td>1.460147E−03</td>
<td>2.221314E−03</td>
</tr>
<tr>
<td>CoDE</td>
<td>1.225893E−03</td>
<td>1.820441E−03</td>
<td>2.560516E−03</td>
</tr>
<tr>
<td>DEGL</td>
<td>9.825195E−04</td>
<td>9.858991E−04</td>
<td>1.583493E−03</td>
</tr>
<tr>
<td>JADE</td>
<td>9.826437E−04</td>
<td>1.241540E−03</td>
<td>2.125234E−03</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>RMSE</th>
<th>NFEs (ε = 0.01)</th>
<th>Sr</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Min</td>
<td>Median</td>
<td>Max</td>
</tr>
<tr>
<td>jDE</td>
<td>2.425084E−03</td>
<td>2.428733E−03</td>
<td>2.605655E−03</td>
</tr>
<tr>
<td>SaDE</td>
<td>2.425075E−03</td>
<td>2.425405E−03</td>
<td>2.470221E−03</td>
</tr>
<tr>
<td>CoDE</td>
<td>2.437571E−03</td>
<td>2.631745E−03</td>
<td>3.499591E−03</td>
</tr>
<tr>
<td>DEGL</td>
<td>2.425075E−03</td>
<td>2.425120E−03</td>
<td>3.307934E−03</td>
</tr>
<tr>
<td>JADE</td>
<td>2.425075E−03</td>
<td>2.425075E−03</td>
<td>2.639902E−03</td>
</tr>
<tr>
<td>R_cr-IJADE</td>
<td>2.425075E−03</td>
<td>2.425075E−03</td>
<td>2.425075E−03</td>
</tr>
</tbody>
</table>

(El-Naggar et al., 2012). Like the previous cases, R_cr-IJADE also gets the best RMSE value among all compared methods in the PV module. Moreover, according to the curve fitting results shown in Table 8 and the I−V characteristic plotted in Fig. 1c, 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 (Brest et al., 2006), SaDE (Qin et al., 2009), CoDE (Wang et al., 2011), DEGL (Das et al., 2009), and JADE (Zhang and Sanderson, 2009)) 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., μ = 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++.

5.4.1. Performance criteria

To compare the performance between different algorithms, the following performance criteria are used.
As described in Eq. (8), it is used to measure the quality of the fit between the simulated data and experimental data obtained by the algorithm.

The NFEs is used to record the number of function evaluations in each run for finding a solution $x$ satisfying $F(x) - F(x^*) \leq \epsilon$, where $F(x^*) = 0$ is the optimal objective function value in this work, and $\epsilon$ 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 $x$ satisfying $F(x) - F(x^*) \leq \epsilon$.

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 9–11 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 $z = 0.05$. With respect to the NFEs, performance, the mean and standard deviation values are recorded. The convergence graphs of different DE variants are plotted in Fig. 2.

![Convergence graphs](image-url)

Fig. 2. Convergence graphs of different DE variants for a: the single diode model, b: the double diode model, and c: the PV module.
Based on the results shown in Tables 9–11 and Fig. 2, it can be observed that:

- In terms of the RMSE values, the proposed \( R_{cr-}\text{IJADE} \) consistently gets the overall best results among different DE variants in all cases. According to the Wilcoxon’s test, \( R_{cr-}\text{IJADE} \) significantly surpasses other DE variants for different solar cell models. In addition, the standard deviation values of \( R_{cr-}\text{IJADE} \) are the smallest, which means that \( R_{cr-}\text{IJADE} \) 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 \( c \), followed by \( R_{cr-}\text{IJADE} \). However, when the extracted parameters increase, for the double diode model, \( R_{cr-}\text{IJADE} \) is the best one, followed by DEGL. This phenomenon indicates that \( R_{cr-}\text{IJADE} \) has better scalability than DEGL.

- Considering the success rate \( S_r \), it is clear that only \( R_{cr-}\text{IJADE} \) can consistently provide \( S_r = 1.0 \) for different solar cell models, while other DE methods cannot 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 \( R_{cr-}\text{IJADE} \).

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

6. Conclusions

In this paper, an improved DE variant, \( R_{cr-}\text{IJADE} \), is proposed to fast and accurately extract the parameters of solar cell models. \( R_{cr-}\text{IJADE} \) 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 \( R_{cr-}\text{IJADE} \), 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. \( R_{cr-}\text{IJADE} \) 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 \( R_{cr-}\text{IJADE} \) 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 \( R_{cr-}\text{IJADE} \) method is able to extract the parameters of different solar cell models fast and accurately. Additionally, compared with other methods, \( R_{cr-}\text{IJADE} \) can provide more precise and robust solutions, obtain higher success rate, and converge faster. Therefore, \( R_{cr-}\text{IJADE} \) can be an efficient and reliable alternative for other complex optimization problems of solar cell models.

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

References


