Applied Energy 112 (2013) 185–193

Contents lists available at SciVerse ScienceDirect

Applied Energy

journal homepage: www.elsevier.com/locate/apenergy

Parameter estimation of solar cells and modules using an improved adaptive differential evolution algorithm

Lian Lian Jiang [a,*], Douglas L. Maskell [a], Jagdish C. Patra [b]

[a] School of Computer Engineering, Nanyang Technological University, Singapore
[b] Faculty of Engineering & Industrial Sciences, Swinburne University of Technology, Australia

HIGHLIGHTS

- An improved adaptive DE (IADE) based technique for parameter estimation in solar cells and modules is proposed.
- The proposed IADE algorithm is based on the feedback of fitness value in the evolutionary process.
- It features simple structure, small storage requirement and good accuracy.
- It eliminates the need for users to manually tune the control parameters of DE algorithm.
- It provides better performance than many other popular optimization methods.

ARTICLE INFO

Article history:
Received 12 March 2013
Received in revised form 29 May 2013
Accepted 5 June 2013
Available online 3 July 2013

Keywords:
Parameter estimation
Solar cells and modules
Improved adaptive differential evolution technique
Single diode model
Optimization

ABSTRACT

A number of different analytical and numerical methods have previously been proposed to estimate solar cell/module parameters. For a single diode model, the parameters include the photocurrent, the saturation current, the parasitic series and shunt resistances, and the ideality factor. Among the proposed optimization techniques, the differential evolution (DE) based method besides its computational advantage, provides better accuracy compared to other methods. However, it is not an easy task to determine the control parameters for the DE algorithm. This paper presents an improved adaptive DE (IADE) based optimization technique. New formulas for the scaling factor and crossover rate are proposed. The proposed IADE algorithm uses a simple structure based on the feedback of fitness value in the evolutionary process. It provides better performance for estimation of the solar cell and module parameter values than other popular optimization methods such as particle swarm optimization, genetic algorithm, conventional DE, simulated annealing (SA), and a recently proposed analytical method. Comparisons with existing methods for solar cell and module parameter estimation are presented using both synthetic and experimental data under various environmental conditions. The proposed scheme offers higher accuracy, and also eliminates the need for users to manually tune the control parameters of DE algorithm.

1. Introduction

Solar energy has become one of the most promising renewable sources due to its almost zero-emission, decreasing cost, the abundance of the energy source and advancements in semiconductor and power electronic devices. Many strategies have been proposed to improve the efficiency of the solar energy systems, such as: using different composite materials [1], creating effective solar cell structures [2], designing more efficient maximum power point tracking (MPPT) and electrical converters for power harvesting [2–7], and new architectures for photovoltaic (PV) arrays [4,8]. Among all these studies, accurate parameter extraction of the PV cells is highly significant in solar energy related research as the parameters extracted from the current–voltage (*I–V*) curves of the PV cell can be used in many applications. For example, it can be used for analyzing performance of more complex grid-connected PV systems, or for investigating the performance of the MPPT system under different irradiance conditions. Moreover, the insight provided by the parameters, which are closely associated with the known physical processes, can be used in quality control during development of the devices and fabrication process optimization [9]. Parameter extraction refers to the process of extracting the electrical parameters of the solar cells/modules from the measured *I–V* curves. These *I–V* curves can be under dark [10] or illuminated [9]conditions. There are two widely used non-linear lumped-parameter equivalent circuit models for solar cells: the

* Corresponding author. Tel.: +65 83589341.
E-mail addresses: ljiang2@e.ntu.edu.sg (L.L. Jiang), asdouglas@ntu.edu.sg (D.L. Maskell), JPatra@swin.edu.au (J.C. Patra).

0306-2619/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.apenergy.2013.06.004

single diode model [11] and the double-diode model [11]. The single diode model offers lower accuracy but is much simpler than the double-diode model. For a single diode model, the extracted parameters include the photocurrent (I_{ph}), the saturation current (I_{s1}), the parasitic series and shunt resistance (R_s and R_p), and the ideality factor (n). For the double diode solar cell model, two additional parameters: the saturation current (I_{s2}) and ideality factor (n_2) of the second diode are included.

During the last two decades, many methods for parameter estimation of solar cells have been proposed. Generally, the parameter estimation methods can be categorized into analytical methods [11] and numerical methods [12–22]. The analytical method for parameter estimation of solar cells requires some key points from the *I–V* characteristic curve, such as the optimal power point, the short circuit current, the open circuit voltage, and the slopes at the axis intersections [11]. This method allows for a relatively simple and fast calculation of the parameter values. However, the accuracy of the extracted parameters heavily depends on the accuracy of these key points. Unfortunately, the measured data usually contains noise due to device inaccuracy and other electrical disturbances. In addition, when using the values from the datasheet provided by the manufacturer, significant errors can be introduced because the *I–V* characteristics in the datasheet are measured under standard test conditions, and are usually not the same as those measured under real weather conditions. Since the parameters vary with environmental conditions, it is unrealistic to use the key data points from the datasheet to calculate the model parameters.

To overcome the uncertainties in the analytical method, many researchers have explored numerical methods, such as the Newton Raphson (NR) method [12], the genetic algorithm (GA) [13], particle swarm optimization (PSO) [14], simulated annealing (SA) [15], harmony search (HS) [16], pattern search (PS) [17], differential evolution (DE) [18], and artificial neural network (ANN) [19–21]. For the traditional NR method, the final results are heavily dependent upon the initial choice of the parameters. Additionally, the NR method is computationally intensive as the Jacobian matrix needs to be calculated at each iteration. Even worse, when the Jacobian matrix is very close to singular, the NR method may fail to converge. Compared to the conventional NR method, the GA method exhibits better results with no initial condition dependency. However, GA requires a much longer time to determine the parameters [18]. Ye et al. [14] proposed using PSO to extract the solar cell parameters from the illuminated *I–V* characteristics. PSO gave higher parameter precision and better computational efficiency than GA, however, the consistency of the results is affected by the randomly generated initial values. In order to solve this problem, Sandrolini et al. [22] used cluster analysis to determine whether the parameter mean values adequately represented the PV module. Among the different clusters obtained from analysis, the one with the highest density is chosen from which the parameters are extracted. However, this procedure requires a large amount of memory, and when applied to a real-time system, a high enough speed for simulation is not guaranteed. For SA, the initial temperature and the cooling scheme significantly influence the results. Similarly, HS and PS have several shortcomings, including a low search speed and a lack of accuracy. ANN based technology is suitable for the system providing sufficient data for network training. Other than the shortcomings identified the specific characteristics of process for parameter estimation should also be considered. For example, the parameter estimation method should be reliable, accurate and fast, for a range of different types of cells estimation. Of the parameter estimation techniques mentioned above, conventional DE seems to be a good choice estimation as it exhibits good accuracy and high convergence speed [18]. However, the results depend on the control parameters of the DE algorithm and an inappropriate choice of these parameters may result in DE failing to converge or in a very slow convergence to the global optimal point.

In this paper, we propose an improved adaptive DE (IADE) based parameter estimation method. IADE has a simple structure and automatically adjusts the two control parameters for the DE algorithm. Compared to the standard DE optimization algorithm, our proposed IADE algorithm reduces the number of control parameters to be tuned manually. It not only ensures the accuracy of the extracted parameters but also simplifies and accelerates the optimization process. The feasibility of the proposed method is tested with both synthetic data and experimental *I–V* data. The proposed algorithm can also be applied to other methods which take advantage of an auxiliary function [23–25]. In these cases, an auxiliary equation, $f(I, V)$, can be used to transform the original variable space into another space so that the relationship between I or V and the auxiliary function can be represented as simple polynomial equations. This method makes the process of parameter estimation for solar cells/modules easier as the parameters can then be extracted using the relation between the original parameters and newly formed parameters in the auxiliary function.

The remainder of this paper is organized as follows. Section 2 describes the solar cell models and introduces the parameter estimation process. Section 3 briefly presents the main principles of the DE algorithm. Section 4 introduces our proposed IADE algorithm. Section 5 provides parameter estimation results using both synthetic and experimental data. Performance comparisons with other methods are also presented in this section. Finally, Section 6 concludes this paper.

2. Problem formulation

In order to illustrate the procedure of parameter estimation, the mathematical model of the solar cell/module and the objective function of the optimization process for parameter estimation are introduced in this section.

2.1. Solar cell/module model

In our work, the single diode model is used to describe the electrical characteristics of a solar cell/module. The equivalent circuit of the single diode model for a solar cell is shown in Fig. 1.

Usually, multiple solar cells are connected in series or parallel to satisfy the higher voltage and/or current requirements of the load. Assemblies of solar cells form modules and for large-scale PV systems arrays of modules are usually connected in series and/or parallel. The single diode lumped parameter equivalent electrical circuit of solar cell is described as [26]:

$$I = I_{ph} - I_0\left[\exp\left(\frac{V + IR_s}{nV_t}\right) - 1\right] - \frac{V + IR_s}{R_p} \tag{1}$$

where I is the output current, V is the output voltage, I_{ph} is the photocurrent generated by the solar cell, R_s and R_p are the series and

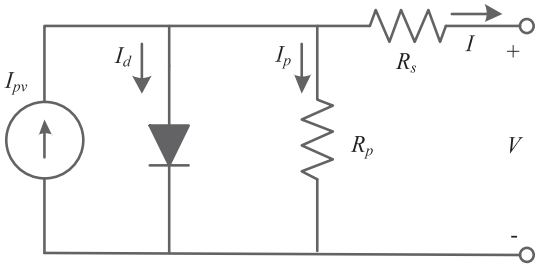

Fig. 1. Equivalent circuit for the single diode model.

parallel resistances, respectively, V_t is the thermal voltage of the diode ($V_t = kT/q$), k is the Boltzmann constant ($k = 1.3806503 \times 10^{-23}$ J/K), q is the electron charge, T is the temperature of the solar cell in Kelvin, and n is the ideality factor of the diode, representing the diffusion and recombination current components. This equivalent circuit expression can also be used for solar modules. However, for the PV module, it should be noted that though the extracted parameters show good association with the experimental data, they cannot be exactly related to the physical phenomena due to the differences between the solar cells connected to form the modules [12].

2.2. Parameter estimation process

Since the single diode model for a solar cell/module is used in this paper, the values of the five unknown parameters, I_{ph}, I_o, R_s, R_p, and n are estimated. From its definition, it shows that Eq. (1) represents a non-linear transcendental function. A fitness function [14,18] can then be defined as the root mean square error (RMSE) between the simulated current value and the measured current value. For N measured data points (the combination of current and voltage values), the fitness (or objective) function in the optimization process is defined as:

$$fitness(\theta) = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\varepsilon(V_i, I_i, \theta)^2} \tag{2}$$

where

$$\varepsilon(V_i, I_i, \theta) = I_i - I_{pv} + I_0\left[\exp\left(\frac{V_i + I_i R_s}{nV_t}\right) - 1\right] + \frac{V_i + I_i R_s}{R_p} \tag{3}$$

where V_i and I_i are the experimental voltage and current points, and θ is the vector of parameters to be extracted [I_{ph}, I_o, R_s, R_p, n]. The purpose of the optimization process is to determine the optimal parameters by minimizing the fitness function. A small value for the fitness function indicates a small error between the computed current and the measured current.

3. Overview of the differential evolution algorithm

DE is a population based stochastic function optimization algorithm which optimizes a problem by iteratively improving the solution according to predefined performance criteria. This continuous function optimizer was initially developed by Storn and Price [27] to solve the Chebychev polynomial fitting problem. The critical idea behind the DE algorithm is the mechanism for generating trial parameter vectors. DE adds the weighted difference between two population vectors to a third mutated vector. No separate probability distribution or gradient of the problem needs to be used, making the scheme fully self-organizing. The basic steps of the standard DE algorithm are described as follows [27].

Step 1: *Initialization*. Set the initial conditions of the DE algorithm, such as the generation label ($g = 1$), the population size (N_p), maximum generations (G_{max}), the crossover rate (CR), and the scaling factor (F). Randomly generate an initial (first generation) population size of N_p D-dimensional vectors with a uniform probability distribution, within the predetermined limits (X_{Lower}, X_{Upper}) of the parameter space. The initial population can be given by:

$$X^1 = \left[X_1^1, X_2^1, \ldots, X_i^1, \ldots, X_{NP}^1\right], \quad (i = 1, 2, \ldots, N_p) \tag{4}$$

$$X_i^1 = X_{Lower} + (X_{Upper} - X_{Lower})RM_i^1, \quad X_i^1 = \left[x_{i,1}^1, x_{i,2}^1, \ldots, x_{i,D}^1\right] \tag{5}$$

where each variable $\left(X_i^1\right)$ is a D-dimensional vector, and $RM_i^1 = \left[rm_{i,1}^1, rm_{i,2}^1, \ldots, rm_{i,D}^1\right]$, a randomly generated vector within the range of $(0, 1)$ according to the uniform distribution.

Step 2: *Mutation*. For each individual in the gth generation, DE generates new solution candidate (mutant vector) by adding the weighted difference between randomly selected individuals to another vector (target vector). These randomly selected individuals are different from the target vector. In this work, we use the mutation strategy of "DE/best/1/bin" (a nomenclature scheme to reference different DE variants), where "DE" denotes differential evolution, "best" indicates that the target vector selected to compute the mutation vectors is chosen to be the individual with the best fitness value from the current population in the gth generation, "1" represents the number of difference vectors adopted, and "bin" means binomial recombination. The mutation strategy is described by:

$$V_i^g = X_{best}^g + F\left(X_{r_1}^g - X_{r_2}^g\right) \tag{6}$$

where $r_1 \neq r_2 = \{1, 2, \ldots, N_p\}$ are randomly generated integers respectively, X_{r1}^g, X_{r2}^g are the randomly selected vectors among the population, X_{best}^g is the vector with the best fitness value among the individuals, $F \in (0, 2]$ is a scaling factor which controls the amplification of the differential variation and $V_i^g = \left[v_{i,1}^g, v_{i,2}^g, \ldots, v_{i,D}^g\right]$. Other mutation strategies, described in [28], are also widely used in DE algorithms.

Step 3: *Crossover*. In order to increase the diversity of the population, a crossover process is introduced. Crossover is realized by accepting a component of the mutant vector according to a stochastic probability. If a newly generated random value, with uniform probability distribution, is not greater than CR ($0 \leqslant CR \leqslant 1$), the jth component of the mutant vector is accepted. Otherwise, it is rejected. As a result, after crossover operation it forms a vector called trial vector, which is given by:

$$u_{i,j}^g = \begin{cases} v_{i,j}^g, & \text{if } \left(rm_{i,j}^g \leqslant CR\right) \text{ or } j = rm_{i,j}^g \\ x_{i,j}^g, & \text{if } \left(rm_{i,j}^g > CR\right) \text{ and } j \neq rm_{i,j}^g \end{cases}, \quad (i = 1, 2, \ldots, N_p, \quad j = 1, 2, \ldots, D) \tag{7}$$

where $u_{i,j}^g$ is a component of $U_i^g = \left[u_{i,1}^g, u_{i,2}^g, \ldots, u_{i,D}^g\right]$.

Here, boundary constrains are used to avoid the violation of specified limits. Any particle which violates the predetermined boundary of variable space is constrained by the following equations:

$$U_i^g = \begin{cases} U_i^g + (X_H - X_L) \times rm_i, & \text{if } U_i^g \leqslant X_L \\ U_i^g - (X_H - X_L) \times rm_i, & \text{if } U_i^g \geqslant X_H \end{cases} \tag{8}$$

Step 4: *Selection*. After the crossover process, the generated trial vector may be chosen to be a member of the next generation (the $(g + 1)$th generation) based on the selection criteria, which is given by:

$$X_i^{g+1} = \begin{cases} U_i^g, & \text{if } fitness(U_i^g) < fitness(X_i^g) \\ X_i^g, & \text{otherwise} \end{cases} \tag{9}$$

For a minimization problem, if the fitness function value, $fitness(U_i^g)$, of the trial vector is smaller than that of the original vector, $fitness(X_i^g)$, the newly generated trail vector will be considered as the successful individual in the next generation. Otherwise, it is rejected and the original vector is retained.

Step 5: *Iterate*. Increase generation step, $g = g + 1$, and repeat step 2 to step 4 until it satisfies the stop criteria which can be

the maximum number of generations (G_{max}) or fitness function values reaches a predefined minimum limit (ε_p).

Step 6: *Output*. Extract the best individual with minimum fitness function in the last generation when predefined minimum limit or maximum number of generation is reached. Output the minimum fitness value and optimal solution (X_{opt}).

4. Proposed improved adaptive DE algorithm

In the conventional DE, the scaling factor (F) and crossover rate (CR) are usually fixed. However, for a particular multidimensional problem, it is not an easy task to properly set the control parameters for the DE algorithm. When initialized with inappropriate control parameters, the DE algorithm may require a long run time or even fail to converge to an optimal value. Therefore, a trial-and-error method is usually used for tuning the control parameters, however this is obviously not satisfactory nor optimal and usually results in the need for many tedious optimization trials.

Different adaptive or self-adaptive DEs have been proposed to dynamically adjust the control parameters throughout the evolutionary search process. In the past few years, adaptive DEs have been successfully used in many different applications [29,30]. Popular adaptive DEs include SaDE [31], FADE [32], jDE [33], JADE [34], EPSDE [35], and CoDE [36], etc. In [31], Qin et al. proposed SaDE, in which a trial vector is generated from four candidate pools based on the success rate of each mutation strategy within a certain number of generations. The values of scaling factor F and crossover CR are generated randomly on a normal distribution within a predefined range respectively. In [32], Liu et al. presented a fuzzy adaptive DE (FADE), it uses a fuzzy knowledge-based system to dynamically adjust the control parameters F and CR. In [33], Brest et al. proposed jDE, in which the parameters F and CR are encoded into the individuals $X_i^g = \left[x_{i,1}^g, x_{i,2}^g, \ldots, x_{i,Np}^g, F_i^g, CR_i^g\right]$. The values of these two parameters are updated with other components in the population vector. From the experimental results, it was found that jDE performs better than the classic DE algorithm and FADE. In [34], Zhang et al. proposed JADE, in which a novel mutation strategy and an external archive are applied into the evolutionary process. In the new mutation strategy, a certain number of explored inferior solutions which are stored in an archive are also considered as the candidates to be chosen for generating the mutation vector. In addition, instead of using the best vector of the whole population, any of the top 100p% ($0 < p < 1$) individuals in the current population can be randomly selected as a component to contribute in generating the mutation vector $\left(X_{best,g}^p\right)$. Based on the idea of SaDE, Mallipeddi et al. proposed the EPSDE [35], in which trial vectors along with a pool of discrete F and CR compete among three mutation strategies to produce potential solutions, which are stored when the target vector survives. In similar way, Wang et al. [36] presented a composite DE which has a systematic framework for mutation strategies, each of which has fixed combination values of F and CR.

The experimental results of each method applied to different benchmark functions shows that they provide accurate enough global optimal results. However, the common characteristics of these algorithms are that they have a complex procedure and that a large system memory is generally required to store the previous states in order to predict the progress direction. They are usually suitable for highly dimensional problems and an off-line calculation process. However, when they are used for online optimization, these adaptive DEs cannot satisfy the response requirements. In our study, the parameters extracted from the PV cells/modules are used in real-time modeling of the complete PV system, and as such, an optimization algorithm which has a simple structure, a small memory footprint and control parameters which can be automatically tuned will better contribute to the parameter estimation process.

Therefore, in our work we present an adaptive DE algorithm with a simple structure, called IADE, which can automatically adjust the control parameters according to the fitness values during the optimization process. A better defined search range can accelerate the search process. The value ranges of F and CR vary with different optimization problems and there is no strict rule when choosing F and CR. In order to make sure the algorithm can choose F and CR wisely, the fitness values of the subsequent previous steps, $(i-1)$ and (i), are an important factor affecting the strategy for choosing parameters F and CR. The ranges of CR and F in [18,37] are recommended between 0.4 and 1, however, after trying different fixed values of F and CR manually, we found the algorithm leads to less error in the extracted parameters when the range of these two parameters is restricted to between 0.5 and 1. We use an exponential function to map F and CR in the range from [0,1] to [0.5,1], rather than a linear function, as an exponential has a smooth characteristic with arbitrary order derivatives. Since the range variation of F and CR is the same, the expressions for choosing F and CR are similar. The formulas of the scaling factor (F) and crossover rate (CR) are given by:

$$A = \frac{fitness\_best(i)}{fitness\_best(i-1)} \tag{10}$$

$$F = b \cdot \exp(a \cdot A \cdot rm)) \tag{11}$$

$$CR = b \cdot \exp(a \cdot A \cdot rm)) \tag{12}$$

Table 1
Pseudo code of the IADE algorithm.

```
Begin
Set parameters ranges [X_Lower, X_Upper], population size (N_p), maximum
    generation (G_max), current–voltage dataset of solar cell (V_i, I_i);
Create a random initial population X_i^1 = X_Lower + (X_Upper − X_Lower) rm_i^1;
Get the vector with best fitness value in the first generation
    X_best^1 = X(min(fitness(X_i^1)));
Set g = 1;
fitness_best_last1 = min(fitness(X_i^1))
fitness_best_last2 = fitness_best_last1
While (g ⩽ G_max) and (tol > ε_p)
  For i = 1 to N_p
    Randomly choose X_r1 and X_r2 (X_r1 ≠ X_r2) from current population;
    Generate scaling factor and crossover rate;
      A = fitness_best_last1/fitness_best_last2
      F = b · exp(a · A · rm), CR = b · exp(a · A · rm)
    Generate trial vector V_i^g = X_best^g + F(X_r1^g − X_r2^g);
    If V_i^g ⩽ X_L or V_i^g ⩾ X_H
      V_i^g = X_L + (X_H − X_L) × rm_i^g
    End if
    For j = 1 to D
      If rm_ij^g ⩽ CR or j = rm_ij^g
        u_ij^g = v_ij^g
      Else if rm_ij^g > CR or j≠rm_ij^g
        u_ij^g = x_ij^g
      End if
    End for
    If fitness (U_i^g) < fitness(X_i^g)
      X_i^g = U_i^g
    Else X_i^g = X_i^g
    End if
  End for
  Get vector with best fitness value X_best^g = X(min (fitness(X_i^g)));
  fitness_best_last2 = fitness_best_last1
  fitness_best_last1 = min (fitness(X_i^g))
  g = g + 1;
End while
```

where *fitness_best* (i) and *fitness_best* $(i-1)$ are the fitness value for the ith and $(i-1)$th iteration, respectively; A is the factor used to connect the current fitness value with the strategy for choosing parameters, F and CR. In order to generate the values of F and CR randomly, the random number rm, with a uniform distribution ($rm \in [0,1]$) is used in (11) and (12). a and b are constant parameters calculated based on the predefined search ranges of F and CR. By mapping the range $[0,1]$ to $[0.5,1]$, when the value of rm equals to 0 or 1, the corresponding value of F (or CR) should be 0.5 or 1, respectively. As a result, the values of these constant parameters are computed as: $a = \ln 2$, and $b = 0.5$. After substituting the values of a and b into (11) and (12), we can see that when there is no improvement in fitness value between two subsequent generations, the value of A is equal to 1. Thus, the F and CR parameters are random values in the range $[0.5,1]$. For the case when there is an improved fitness value, the value of A is between 0 and 1. As a result, F and CR take random values from $[0.5, 0.5 \cdot \exp \cdot (\ln 2 \cdot A)]$, which has a relatively smaller search range than $[0.5,1]$. This choice of ranges for F and CR, in the event of no improvement in fitness value, provides a diverse range from which to choose a new value. Thus, it adds more diversity to the population in the search process, which leads to more promising progress directions and hence avoids premature convergence and thus provides a higher quality final solution. The pseudo code of the proposed IADE method is presented in Table 1.

5. Results and discussions

In order to verify the effectiveness of our proposed improved adaptive DE, both synthetic PV data and experimental PV cell (and module) data under different irradiance conditions are tested. Comparisons with other optimization algorithms for parameter estimation are also presented for the synthetic data, which is generated using the single diode solar cell/module model. Since we cannot practically guarantee optimal parameters for the experimental solar cell/module, we consider the results with the minimum RMSE, defined in Eqs. (2) and (3), to be the optimal solution. All the algorithms are implemented in MATLAB on a standard PC with a 3.17 GHz duo core Intel CPU and 8G RAM operating under the Windows 7 operating system.

5.1. Estimation results for synthetic data

The synthetic data is generated from the single diode solar cell model with following parameter values: $I_{ph} = 0.7608$ A, $I_s = 3.223$ - $\times 10^{-7}$ A, $R_s = 0.0364\ \Omega$, $R_p = 53.76\ \Omega$, $n = 1.4837$ under the condition of $G = 1000\ \mathrm{W/m^2}$ and $T = 33\ ^\circ\mathrm{C}$. These parameter values were chosen as they are widely used in literatures [12,38]. Before testing the proposed adaptive DE algorithm, the difficulty in selecting the proper control parameters for the DE algorithm is examined. The fitness value of the extracted parameters for different combinations of F and CR, are shown in Fig. 2. The values of F and CR change from 0.1 to 2 and 0.1 to 1, respectively, with a step of 0.1. Thus, there are 200 combinations of F and CR. In order to better show the error distribution, 10 points with fitness value larger than 0.025 are not shown in Fig. 2. From Fig. 2, we can see that there are a number of F and CR pairs which display a large fitness value, that is, they provide a poor estimate. If we consider that the extracted parameters with a fitness value less than 1×10^{-4} constitute a successful result, the success rate (defined as a ratio of the number of successful results and the total number of combinations (200 in this example)) is only 13.5%. The minimum, maximum and average fitness values for the 200 combinations of F and CR are 5.2979×10^{-11}, 0.2091and 0.0071, respectively. This means that even though the minimum fitness value is small, the average and the maximum fitness value are relatively large.

Next, we examine the estimation results using different optimization methods for synthetic PV data. Since the range of each parameter influences the accuracy of both the results and the search speed, we choose the search range to be ±100% of the parameter values, similar to [14]. The maximum number of generations, (G_{max}), was set to 8000. The extracted parameters are computed from the synthetic data, using a number of different optimization methods. 30 runs of each optimization method were executed and the best results from the 30 runs are presented in Table 2. From Table 2, we can see that PSO, GA, SA, and the analytical method presented in [39] all show higher RMSE than our proposed IADE method. Even though the traditional P-DE algorithm gives good accuracy, it requires careful initialization of the control parameter values. If unsuitable control parameters are set, like ($F = 0.7$, $CR = 1$) in Table 2, the algorithm shows poor accuracy. However, our proposed method is able to effectively adapt the control parameters and thus can always estimate the parameters with good accuracy.

In much of the previous work described in literature, the extracted parameters from the proposed methods are usually the best solution after trying different combinations of control parameters. However, the distribution of the fitness values within a certain number of runs is also an important indicator for evaluating the optimization algorithm. In order to evaluate the consistency of each optimization algorithm for parameter estimation, we calculate the minimum ($fitness_{min}$), maximum ($fitness_{max}$) and average ($fitness_{avg}$) of the fitness function after 30 runs for each algorithm. These results are shown in Table 3. From Table 3, we can see that even though PSO gives better accuracy (i.e. a better minimum fitness value ($fitness_{min}$)) compared to the other non-DE optimization algorithms, PSO has a large distribution of results. This means that PSO algorithm cannot guarantee consistency in

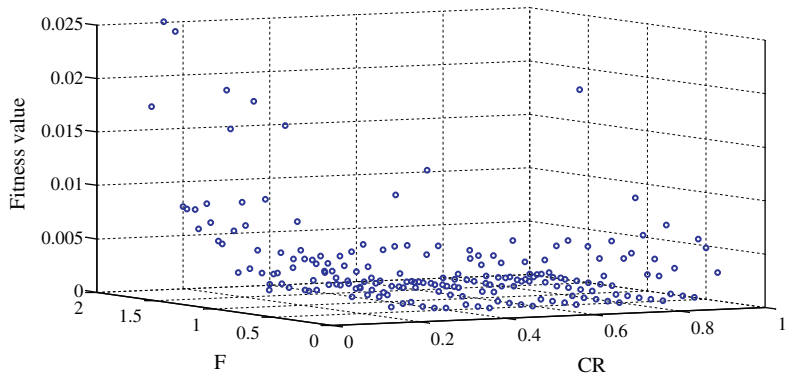

Fig. 2. Influence of control parameters on fitness value for different combinations of F and CR.

Table 2
Comparison of different methods for parameter estimation with synthetic data ($G = 1000\ W/m^2$ and $T = 33\ ^\circ C$).

Parameters	Synthetic values	PSO	GA	SA	P-DE ($F = 0.7$, $CR = 1$)	P-DE ($F = 0.8$, $CR = 1$)	Analytical method [39]	Proposed method (IADE)
I_{ph}	0.7608	0.7609	0.7583	0.7286	0.7602	0.7608	0.7608	0.7608
I_s	3.2230×10^{-7}	3.0800×10^{-7}	9.5342×10^{-7}	8.9263×10^{-7}	3.8686×10^{-7}	3.2272×10^{-7}	3.2228×10^{-7}	3.2272×10^{-7}
R_s	0.0364	0.0366	0.0321	0.0297	0.0357	0.0364	0.0364	0.0364
R_{sh}	53.7600	52.4725	348.8490	290.0109	64.0848	53.7965	53.76275	53.7965
n	1.4837	1.4792	1.6017	3.1979	1.5023	1.4838	1.4837	1.4838
RMSE	–	5.402×10^{-5}	1.900×10^{-3}	1.185×10^{-1}	3.229×10^{-4}	3.583×10^{-11}	5.808×10^{-5}	4.382×10^{-11}

Table 3
Performance of different optimization algorithms for parameter estimation with synthetic data within 30 runs, $G = 1000\ W/m^2$ and $T = 33\ ^\circ C$.

Fitness value	PSO	GA	SA	P-DE ($F = 0.7$, $CR = 1$)	P-DE ($F = 0.8$, $CR = 1$)	Proposed method (IADE)
$Fitness_{min}$	5.402×10^{-5}	1.900×10^{-3}	2.620×10^{-2}	3.229×10^{-4}	3.583×10^{-11}	4.437×10^{-11}
$Fitness_{max}$	4.736×10^{-1}	1.132×10^{-1}	1.133×10^{-1}	4.600×10^{-3}	9.942×10^{-11}	9.975×10^{-11}
$Fitness_{avg}$	1.050×10^{-1}	2.120×10^{-2}	5.700×10^{-2}	2.300×10^{-3}	7.449×10^{-11}	8.397×10^{-11}

the results generated. The traditional DE with control parameter ($F = 0.8$, $CR = 1$) and our IADE method both give very good accuracy and consistency of results. However, the traditional DE method requires a number of trial runs, for the particular dataset, to determine the best control parameters. On the contrary, our proposed IADE method does not require exact values for the control parameters to the DE algorithm, but instead only requires a predefined range and gives a better result consistency than other methods. It can automatically tune the values of F and CR, based on the fitness values in each generation step. As a result, it simplifies the time-consuming process of choosing proper control parameters for DE.

5.2. Estimation results for experimental data

In order to illustrate the effectiveness of our proposed method, we verify our IADE algorithm using experimental I–V data for a solar cell/module under different irradiance and temperature condition. For the solar cell, four different irradiance conditions ($G = 1003.76, 884.21, 712.03, 601.50\ W/m^2$) with constant temperature ($T = 31\ ^\circ C$) are tested. For the solar module, four different irradiance conditions ($G = 999.0,\ 400.7,\ 400.6,\ 999.9\ W/m^2$) but with different temperatures ($T = 24.6, 24.6, 45, 45\ ^\circ C$) are tested.

5.2.1. Parameter estimation for solar cell

For the solar cell case, four I–V datasets are measured from a multi-crystalline cell by using a solar simulator and a current and voltage recorder. Each dataset contains 100 points, representing pairs of current and voltage values. The irradiance values related to these four datasets are $1003.76\ W/m^2$, $884.21\ W/m^2$, $712.03\ W/m^2$, and $601.50\ W/m^2$ respectively. A thermocouple was adhered to the backside of the cell to measure the temperature. The temperature was maintained at approximately 31 °C using air-based cooling. The search range of each parameter to be extracted from the I–V curves are defined as follows:

$$I_{pv} \in (0, 5), \quad I_s \in (1 \times 10^{-12}, 1 \times 10^{-5}), \quad R_s \in (0, 2),$$
$$R_{sh} \in (50, 5000), \quad n \in (0.5, 10) \tag{13}$$

After applying our proposed IADE method to each I–V dataset, the extracted parameter values and the RMSE are tabulated in Table 4. In addition, I–V curves generated using the parameters extracted by our proposed method (IADE) are plotted against the experimentally measured data for the multi-crystalline solar cell under the four different irradiance conditions. They are shown in

Table 4
The cell parameters extracted for the experimental solar cell using the proposed method (IADE), with different irradiance conditions and a constant temperature $T = 31\ ^\circ C$.

Parameters	$1003.76\ W/m^2$	$884.21\ W/m^2$	$712.03\ W/m^2$	$601.50\ W/m^2$
I_{ph}	0.343	0.300	0.240	0.204
I_s	2.153×10^{-8}	1.921×10^{-8}	1.859×10^{-8}	3.322×10^{-8}
R_s	0.408	0.408	0.403	0.374
R_{sh}	4968.266	4733.853	4898.204	4976.464
n	1.362	1.343	1.342	1.391
RMSE	3.940×10^{-4}	3.900×10^{-4}	2.920×10^{-4}	3.698×10^{-4}

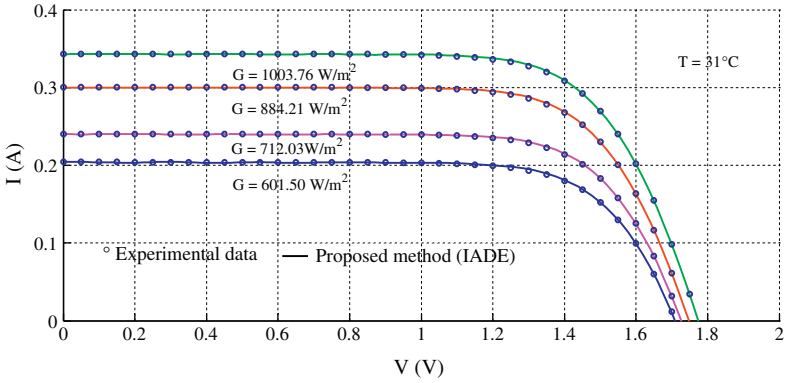

Fig. 3. A comparison of the I–V curves generated using the extracted parameters from our proposed IADE method (solid line) and the measured data for a multi-crystalline solar cell (dots) under four different irradiance conditions, $G = 1003.76\ W/m^2$, $884.21\ W/m^2$, $712.03\ W/m^2$, and $601.50\ W/m^2$, respectively.

Table 5
Extracted parameters for the SL80CE-36M solar module under different irradiance and temperature conditions.

Parameters	T (°C)	G (W/m^2)	T (°C)	G (W/m^2)	T (°C)	G (W/m^2)	T (°C)	G (W/m^2)
	24.6	999.0	24.6	400.7	45	400.6	45	999.9
I_{ph}	2.352		0.945		0.955		2.373	
I_s	6.548×10^{-11}		1.386×10^{-9}		2.5594×10^{-8}		1.9551×10^{-9}	
R_s	0.922		0.972		1.0973		0.9665	
R_{sh}	370.468		883.518		1180.500		442.140	
n	11.561		13.186		7.7395		6.8035	
RMSE	0.0115		0.0060		0.0071		0.0154	

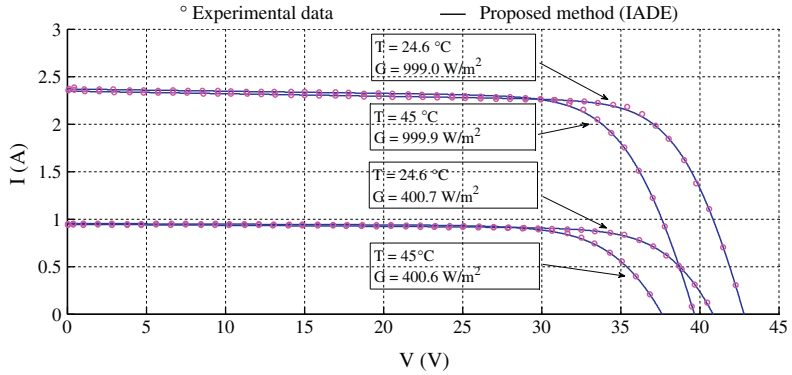

Fig. 4. A comparison of I–V curves generated using the extracted parameters from our proposed IADE method (solid line) and the measured data for a multi-crystalline solar module (dots) under different irradiance and temperature conditions.

Fig. 3. For the sake of clarity, we have only marked selected experimental data points, uniformly distributed within the I–V range, for each curve in Fig. 3. From Fig. 3, we can see that the calculated current values fit very well with the measured data, which means that the parameters extracted by our proposed method can represent the intrinsic parameters of the cell efficiently.

5.2.2. Parameter estimation for solar module

In this section, we use a commercial solar module (SL80CE-36M) to verify the effectiveness of our proposed method for parameter estimation. There are 72 solar cells connected in series in the solar module. The commercial parameters of the module from the manufacture's datasheet are, maximum power: 80 W, optimum power voltage (V_{mp}): 35.1 V, optimum power current (I_{mp}): 2.28 A, open circuit voltage (V_{oc}): 43.2 V, short circuit current (I_{sc}): 2.44 A, short circuit current coefficient (K_i): 0.976 mA/°C, and open circuit voltage coefficient (K_v): −164.16 mV/°C. Similar to the previous solar cell experiment, four datasets corresponding to the I–V curves under different irradiance and temperature conditions ((G, T) = (24.6 °C, 999.0 W/m^2), (24.6 °C, 400.7 W/m^2), (45 °C, 400.6 W/m^2), (45 °C, 999.9 W/m^2)) are obtained. There are 800 points for each dataset. The search ranges of each parameter are defined as:

$$I_{pv} \in (0, 10), \quad I_s \in (1 \times 10^{-12}, 1 \times 10^{-5}), \quad R_s \in (0, 5), \quad R_{sh} \in (50, 5000), \quad n \in (0.5, 100) \tag{14}$$

The extracted parameters and the RMSE after applying our proposed IADE method to these four I–V curves are given in Table 5. Fig. 4 shows the calculated I–V curves using the extracted parameters along with the experimentally measured I–V curves. Again, as in Fig. 3, for clarity purposes we have only marked selected experimental data points, uniformly distributed within the I–V range. From Fig. 4, we can see that the I–V values of the curves obtained using the extracted parameters fit reasonably well over the whole range of the experimental data, except for some points near the maximum power point which have a relatively larger error.

In addition, we have also tested our proposed IADE method using the same experimental I–V data for the multi-crystalline cell/module as in [40] and [12]. The search range for each parameter in solar cell experiment is set as:

$$I_{pv} \in (0, 2), \quad I_s \in (1 \times 10^{-12}, 1 \times 10^{-5}), \quad R_s \in (0, 2), \quad R_{sh} \in (10, 500), \quad n \in (0.5, 4) \tag{15}$$

For the solar module experiment, the search range for each parameter is set as:

$$I_{pv} \in (0, 2), \quad I_s \in (1 \times 10^{-12}, 1 \times 10^{-5}), \quad R_s \in (0, 2), \quad R_{sh} \in (50, 5000), \quad n \in (0.5, 100) \tag{16}$$

The resultant parameter values and the evaluation values for the solar cell with the experimental dataset given by [40] are shown in Table 6. From Table 6, we can see that the RMSE computed by the proposed IADE method is much lower than the method used in [40]. Similarly, we also tested our proposed method with the experimental datasets of solar cell and module provided by [12]. Table 7 summarizes the estimated parameters computed using the proposed IADE method and by the method in [12]. As can be seen, for both the solar cell and the module the RMSE values are much lower than that from [12]. In fact, the RMSE for the solar module parameters derived from our method is about 233 times smaller than that from [12].

Table 6
A comparison of the extracted parameters obtained from our proposed IADE method and those from [40] using the experimental data for the solar cell from [40].

Parameters	Proposed IADE method	From [40]
I_{ph} (A)	0.5810	0.5754
I_s (A)	9.1248×10^{-8}	9.4586×10^{-8}
R_s (Ω)	0.1647	0.1576
R_p (Ω)	32.2752	65.1712
n	1.4872	1.5675
RMSE	6.0057×10^{-4}	3.7000×10^{-3}

Table 7
A comparison of the extracted parameters obtained from our proposed method and those from [12] using the experimental data for the solar cell/module from [12].

Parameters	Solar cell		Solar module	
	Proposed method (IADE)	From [12]	Proposed method (IADE)	From [12]
I_{ph} (A)	0.7607	0.7608	1.0320	1.0318
I_s (A)	3.3613×10^{-7}	3.223×10^{-7}	3.886×10^{-6}	3.2876×10^{-6}
R_s (Ω)	0.03621	0.0364	1.189	1.2057
R_p (Ω)	54.7643	53.7634	921.850	549.450
n	1.4852	1.4837	49.068	48.450
RMSE	9.8900×10^{-4}	9.7000×10^{-3}	2.4000×10^{-3}	5.6010×10^{-1}

The smaller RMSE values in the above comparisons means that our proposed IADE method provides better accuracy. Moreover, the analytical method proposed in [40] is highly dependent on the initial values of the extracted parameters. The advantage is that our proposed DE algorithm requires only a course search-range for parameter determination of solar cells/modules, thus simplifying the optimization procedure [12], unlike algorithms which require very close estimates or guesses to produce a reasonable solution. Moreover, our proposed adaptive DE algorithm has a simple structure and requires no manual tuning; instead it can automatically tune the control parameters using intelligent strategies, which contributes to making it more feasible and effective, when applied to parameter estimation for solar cells/modules.

6. Conclusion

This paper addresses parameter estimation of solar PV cells and modules. Among the many different heuristic optimization algorithms, DE provides a fast convergence speed and better accuracy in finding the global optimal point. However, the control parameters in the traditional DE algorithm significantly influence the estimation results. It is a time-consuming and complex process to choose control parameters. Additionally, the existing adaptive DE methods are usually efficient in highly dimensional problems and require large memory and a complex procedure. In order to overcome these shortcomings, we propose an improved adaptive DE algorithm with a simple structure which can automatically adjust the control parameters in search process. Our proposed IADE method has good accuracy when applied to the problem of extracting the parameters from solar cells and modules. Comparisons with other optimization methods are also presented. These comparisons show that our proposed IADE method provides better accuracy without the need to manually choose the control parameters. The effectiveness of the proposed method is verified using both synthetic and experimental data from our test setup and reference datasets from the related literature.

References

[1] Asim N, Sopian K, Ahmadi S, Saeedfar K, Alghoul MA, Saadatian O, et al. A review on the role of materials science in solar cells. Renew Sust Energy Rev 2012;16:5834–47.
[2] Wright M, Uddin A. Organic–inorganic hybrid solar cells: a comparative review. Sol Energy Mater Sol Cells 2012;107:87–111.
[3] Esram T, Chapman PL. Comparison of photovoltaic array maximum power point tracking techniques. IEEE Trans Energy Convers 2007;22:439–49.
[4] Jiang LL, Maskell DL, Patra JC. A novel ant colony optimization-based maximum power point tracking for photovoltaic systems under partially shaded conditions. Energy Build 2013;58:227–36.
[5] Jiang LL, Maskell DL, Patra JC. A FLANN – based controller for maximum power point tracking in PV systems under rapidly changing conditions. In: IEEE int conf on acoustics, speech, and signal processing Kyoto, Japan; 2012. p. 2141–4.
[6] Hohm DP, Ropp ME. Comparative study of maximum power point tracking algorithms using an experimental, programmable, maximum power point tracking test bed. In: IEEE int conf on photovoltaic specialists conference; 2000. p. 1699–702.
[7] Patel H, Agarwal V. Maximum power point tracking scheme for PV systems operating under partially shaded conditions. IEEE Trans Ind Electron 2008;55:1689–98.
[8] Blaabjerg F, Zhe C, Kjaer SB. Power electronics as efficient interface in dispersed power generation systems. IEEE Trans Power Electron 2004;19:1184–94.
[9] Chegaar M, Nehaoua N, Bouhemadou A. Organic and inorganic solar cells parameters evaluation from single *I–V* plot. Energy Convers Manage 2008;49:1376–9.
[10] Bouzidi K, Chegaar M, Aillerie M. Solar cells parameters evaluation from dark *I–V* characteristics. Energy Proc 2012;18:1601–10.
[11] Chan DSH, Phang JCH. Analytical methods for the extraction of solar-cell single- and double-diode model parameters from *I–V* characteristics. IEEE Trans Electron Devices 1987;34:286–93.
[12] Easwarakhanthan T, Bottin J, Bouhouch I, Boutrit C. Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers. Int J Sol Energy 1986;4:1–12.
[13] Zagrouba M, Sellami A, Bouaïcha M, Ksouri M. Identification of PV solar cells and modules parameters using the genetic algorithms: application to maximum power extraction. Sol Energy 2010;84:860–6.
[14] Ye MY, Wang XD, Xu YS. Parameter extraction of solar cells using particle swarm optimization. J Appl Phys 2009;105. 094502–094502-8.
[15] El-Naggar KM, AlRashidi MR, AlHajri MF, Al-Othman AK. Simulated annealing algorithm for photovoltaic parameters identification. Sol Energy 2012;86:266–74.
[16] Askarzadeh A, Rezazadeh A. Parameter identification for solar cell models using harmony search-based algorithms. Sol Energy 2012;86:3241–9.
[17] AlHajri MF, El-Naggar KM, AlRashidi MR, Al-Othman AK. Optimal extraction of solar cell parameters using pattern search. Appl Energy 2012;44:238–45.
[18] Ishaque K, Salam Z, Mekhilef S, Shamsudin A. Parameter extraction of solar photovoltaic modules using penalty-based differential evolution. Appl Energy 2012;99:297–308.
[19] Patra JC. Chebyshev neural network-based model for dual junction solar cells. IEEE Trans Energy Convers 2011;26:132–9.
[20] Patra JC. Neural network-based model for dual-junction solar cells. Prog Photovoltaics Res Appl 2011;19:33–44.
[21] Patra JC, Maskell DL. Modeling of multi-junction solar cells for estimation of EQE under influence of charged particles using artificial neural networks. Renew Energy 2012;44:7–16.
[22] Sandrolini L, Artioli M, Reggiani U. Numerical method for the extraction of photovoltaic module double-diode model parameters through cluster analysis. Appl Energy 2010;87:442–51.
[23] Ortiz-Conde A, García Sánchez FJ, Muci J. New method to extract the model parameters of solar cells from the explicit analytic solutions of their illuminated *I–V* characteristics. Sol Energy Mater Sol Cells 2006;90:352–61.
[24] Chegaar M, Azzouzi G, Mialhe P. Simple parameter extraction method for illuminated solar cells. Solid-State Electron 2006;50:1234–7.
[25] Haouari-Merbah M, Belhamel M, Tobías I, Ruiz JM. Extraction and analysis of solar cell parameters from the illuminated current–voltage curve. Sol Energy Mater Sol Cells 2005;87:225–33.
[26] Villalva MG, Gazoli JR, Filho ER. Comprehensive approach to modeling and simulation of photovoltaic arrays. IEEE Trans Power Electron 2009;24:1198–208.
[27] Storn R, Price K. Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. J Glob Optim 1997;11:341–59.
[28] Epitropakis MG, Tasoulis DK, Pavlidis NG, Plagianakos VP, Vrahatis MN. Enhancing differential evolution utilizing proximity-based mutation operators. IEEE Trans Evol Comput 2011;15:99–119.
[29] Ghosh A, Datta A, Ghosh S. Self-adaptive differential evolution for feature selection in hyperspectral image data. Appl Softw Comput 2013;13:1969–77.
[30] Al-Anzi FS, Allahverdi A. A self-adaptive differential evolution heuristic for two-stage assembly scheduling problem to minimize maximum lateness with setup times. Eur J Oper Res 2007;182:80–94.
[31] Qin AK, Suganthan PN. Self-adaptive differential evolution algorithm for numerical optimization. In: IEEE cong on evol comput; 2005. p. 1785–91.
[32] Liu J, Lampinen J. A fuzzy adaptive differential evolution algorithm. Softw Comput 2005;9:448–62.
[33] Brest J, Greiner S, Boskovic B, Mernik M, Zumer V. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. IEEE Trans Evol Comput 2006;10:646–57.

[34] Zhang J, Sanderson AC. JADE: adaptive differential evolution with optional external archive. IEEE Trans Evol Comput 2009;13:945–58.
[35] Mallipeddi R, Suganthan PN, Pan QK, Tasgetiren MF. Differential evolution algorithm with ensemble of parameters and mutation strategies. Appl Softw Comput 2011;11:1679–96.
[36] Wang Y, Cai Z, Zhang Q. Differential evolution with composite trial vector generation strategies and control parameters. IEEE Trans Evol Comput 2011;15:55–66.
[37] Gamperle R, Muller DS, Koumoutsakos P. A parameter study for differential evolution. In: Int conf on advances in intelligent systems, fuzzy systems, evolutionary computation; 2002. p. 293–8.
[38] Ye MY, Wang XD, Xu YS. Parameter extraction of solar cells using particle swarm optimization. J Appl Phys 2009;105.
[39] Chen Y, Wang K, Li D, Hong R, Shen H. Parameters extraction from commercial solar cells *I–V* characteristics and shunt analysis. Appl Energy 2011;88:2239–44.
[40] Ghani F, Duke M, Carson J. Numerical calculation of series and shunt resistances and diode quality factor of a photovoltaic cell using the Lambert W-function. Sol Energy 2013;91:422–31.