Parameter extraction of solar cells using particle swarm optimization

Meiying Ye, Xiaodong Wang, and Yousheng Xu

Citation: Journal of Applied Physics **105**, 094502 (2009); doi: 10.1063/1.3122082
View online: http://dx.doi.org/10.1063/1.3122082
View Table of Contents: http://scitation.aip.org/content/aip/journal/jap/105/9?ver=pdfcov
Published by the AIP Publishing

Parameter extraction of solar cells using particle swarm optimization

Meiying Ye,[1,a)] Xiaodong Wang,[2] and Yousheng Xu[1]
[1]*Department of Physics, Zhejiang Normal University, Jinhua 321004, People's Republic of China*
[2]*Department of Electronic Engineering, Zhejiang Normal University, Jinhua 321004, People's Republic of China*

(Received 29 October 2008; accepted 24 March 2009; published online 4 May 2009)

In this article, particle swarm optimization (PSO) was applied to extract the solar cell parameters from illuminated current-voltage characteristics. The performance of the PSO was compared with the genetic algorithms (GAs) for the single and double diode models. Based on synthetic and experimental current-voltage data, it has been confirmed that the proposed method can obtain higher parameter precision with better computational efficiency than the GA method. Compared with conventional gradient-based methods, even without a good initial guess, the PSO method can obtain the parameters of solar cells as close as possible to the practical parameters only based on a broad range specified for each of the parameters.
[DOI: 10.1063/1.3122082]

I. INTRODUCTION

The solar cell is a very useful device for clean electric power generation, and its performance has been continuously improved through intensive research in this field.[1] It is convenient to use a lumped parameter equivalent circuit model to simulate solar cell performance, whose parameters are closely related to the internal physical mechanisms of the solar cell. The knowledge of model parameters is important, not only for cell array and system simulation, but also as an analysis tool to gain an understanding of the processes involved.[2] For example, the series resistance in the model has a significant effect on both the fill factor and the conversion efficiency. The shunt resistance is crucial to photovoltaic system performance, especially at reduced irradiance levels.[3] During the past decades, a number of methods have been proposed to extract the solar cell model parameters from the measured illuminated current-voltage (I-V) data. Among these methods, the direct approaches are based on the use of the I-V curve features, such as the axis intercepts and the gradients at selected points, to determine some cell parameters. The accuracy of these techniques is therefore limited by the measured I-V data, whose errors are introduced by the numerical differentiation and the simplified formulas used in parameter extraction as well. Besides, several different conventional nonlinear fitting algorithms, such as the quasi-Newton method and its variations, have been proposed to solve above solar cell parameter extraction problem.[4–7] However, nonlinear optimization problem also introduces a difficulty that the extracting process cannot guarantee accurately results for the global convergence if it starts from an arbitrarily chosen initial guess. So, by using conventional gradient-based methods, we still cannot characterize the nonlinear behavior of the solar cell very well.

Recently, the methods based on evolutionary computation algorithms have attracted much attention in the area of the solar cell parameter extraction. For example, in literature,[8–10] a technique based on genetic algorithms (GAs) is proposed to improve the accuracy of the solar cell parameter extraction. The performance of this technique also surpasses the quasi-Newton method, a gradient-based search and optimization algorithm. Although GAs have been widely used in parameter estimation and many others, recent research has identified some deficiencies in GA performance.[11] Especially, the degradation in efficiency is apparent in applications with highly epistatic objective functions, i.e., where the parameters being optimized are highly correlated. In this case, the crossover and mutation operations cannot ensure better fitness of offspring because chromosomes in the population have similar structures and their average fitness is high toward the end of the evolutionary process.

As an alternative to GAs, particle swarm optimization (PSO) (Ref. 12) is a recently invented high-performance computation method. The PSO algorithm simulates the behavior of swarm as a simplified social system. Like bird flocking, the social behavior of such organisms can be treated as an optimization procedure. The PSO algorithm differs from traditional optimization algorithms in that a population of potential solutions is used in the search. The direct fitness information, instead of function derivatives or related knowledge, is used to guide the search. Compared with GAs, PSO has some attractive characteristics. First, PSO has memory, that is, the knowledge of good solutions is retained by all particles, whereas in GAs, the previous knowledge of the problem is destroyed once the population changes. Second, PSO has constructive cooperation between particles, that is, particles in the swarm share their information.[13] Due to the simple concept, easy implementation, robustness to control variables, and computational efficiency, when compared with other heuristic optimization techniques, nowadays PSO has gained much attention and wide applications in different fields.

The problem we will solve in this paper is the extraction of a series of solar cell parameters through fitting given ex-

[a)]Author to whom correspondence should be addressed. Tel.: +86 579 82298904. Electronic mail: ymy@zjnu.cn.

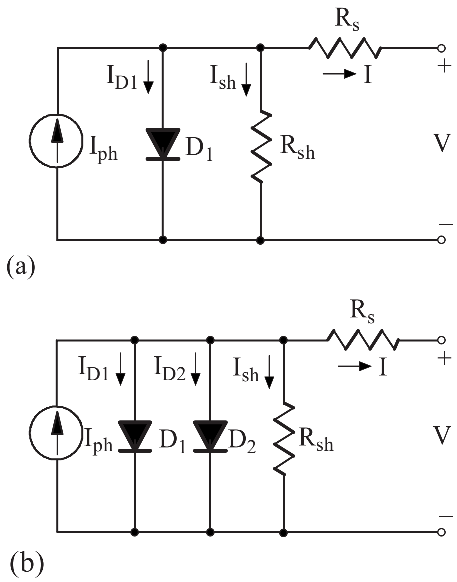

FIG. 1. The (a) single and (b) double diode equivalent circuit models of a solar cell.

perimental *I-V* data using the PSO method. The results are also compared to those obtained by the GA method.

II. EQUIVALENT CIRCUIT MODEL OF A SOLAR CELL

In previous papers, many equivalent circuit models have been presented,[14–20] but there exist two main circuit models in practice to describe solar cells. The first one is the single diode model described by a modified Shockley diode equation incorporating a diode quality factor to account for the effect of recombination in the space-charge region. Under illumination and normal operating conditions, the single diode model is the most popular model for solar cells. However, the single diode model is particularly inaccurate in describing cell behavior at low illuminations.[21] The second model, commonly referred to as the double diode model, is used to simulate the space-charge recombination effect by incorporating a separate current component with its own exponential voltage dependence. With a double diode model we can get insight into the relation between physical phenomena that take place in a solar cell and its parameters.[22] Moreover, the double diode model has been shown to be a more accurate representation of solar cell behavior than the single diode model in some cases.[19] The GA-based parameter extraction of the double diode model was recently reported.[9]

The solar cell models are shown in Fig. 1. For the single diode model, with reference to Fig. 1(a), the *I-V* relationship may be expressed as

$$I = I_{\mathrm{ph}} - I_{D1} - I_{\mathrm{sh}} = I_{\mathrm{ph}} - I_{SD1}\left[\exp\left(\frac{q(V+IR_s)}{n_1 k_B T}\right) - 1\right] - \frac{V+IR_s}{R_{\mathrm{sh}}}, \tag{1}$$

where I is the terminal current, V is the terminal voltage, I_{ph} is the cell-generated photocurrent, I_{D1} is the diode current, I_{sh} is the shunt current, R_s and R_{sh} are the series and shunt resistances, respectively, I_{SD1} is the diode saturation current, and n_1 is the diode ideality factors. The other constants: k_B is Boltzmann constant, q is the electronic charge, and T is the temperature in kelvin.

For the double diode model, with reference to Fig. 1(b), the current-voltage relationship may be altered as follows:

$$I = I_{\mathrm{ph}} - I_{D1} - I_{D2} - I_{\mathrm{sh}} = I_{\mathrm{ph}} - I_{SD1}\left[\exp\left(\frac{q(V+IR_s)}{n_1 k_B T}\right) - 1\right] - I_{SD2}\left[\exp\left(\frac{q(V+IR_s)}{n_2 k_B T}\right) - 1\right] - \frac{V+IR_s}{R_{\mathrm{sh}}}, \tag{2}$$

where I_{SD1} and I_{SD2} describe the diffusion and recombination characteristics of the charge carriers in the material itself and in the space-charge zone, respectively, and n_1 and n_2 are the diffusion and recombination diode ideality factors, respectively. Other variables and constants used in the double diode model are the same as those in the single diode model.

From Eqs. (1) and (2), it is shown that the solar cell parameter extraction problem reduces to determination of the five parameters (R_s, R_{sh}, I_{ph}, I_{SD1}, and n_1) in the single diode model or the seven parameters (R_s, R_{sh}, I_{ph}, I_{SD1}, I_{SD2}, n_1, and n_2) in double diode model with a set of experimental *I-V* data.

After the description of the equivalent circuit models of a solar cell, we start to extract the model parameters and the approach adopted will be detailed in Secs. III–V.

III. PARAMETER EXTRACTION OF THE SOLAR CELL

The solar cell parameters are mostly extracted within the following approach: Given a set of experimental *I-V* data of solar cell, then we can apply an optimization algorithm to tune the parameters until the experimental data are in accord with the relation of Eq. (1) or (2).

Before proceeding with the optimization operations, a performance criterion or an objective function should be first defined, because the value of an objective function will deeply influence on how to perform the calculation of the solar cell parameters. Based on Eqs. (1) and (2) as well as following the approach recently proposed in Ref. 8, we may construct a function expressed as

$$y(I,V,\Theta) = \begin{cases} I - I_{\mathrm{ph}} + I_{D1} + I_{\mathrm{sh}} = f(I,V,\Theta) & \text{for the single diode model} \\ I - I_{\mathrm{ph}} + I_{D1} + I_{D2} + I_{\mathrm{sh}} = g(I,V,\Theta) & \text{for the double diode model,} \end{cases} \tag{3}$$

where $\Theta=[R_s,R_{sh},I_{ph},I_{SD},n_1]$ for the single diode model or $\Theta=[R_s,R_{sh},I_{ph},I_{SD1},I_{SD2},n_1,n_2]$ for the double diode model is a set of parameters; other variables are as described in Eqs. (1) and (2). Note that Eq. (3) implies the error between experimental-measured and calculated currents. Therefore, the objective function based on the root mean square error can be given by

$$F = \sqrt{\frac{1}{N}\sum_{k=1}^{N} y(I_k,V_k,\Theta)^2}, \tag{4}$$

where I_k and V_k, $k=1,2,\ldots,N$, are the experimental illuminated I-V data at the kth point, N is the number of the experimental data, and Θ are unknown model parameters for practical systems, to be determined as accurately as possible. Thus, the problem consists to minimize the objective function F with respect to the set of parameters Θ. The smaller the objective function, the smaller the difference between the experimental-measured and calculated currents. Theoretically, the objective function should be zero for any experimental I-V data when the exact value has been determined for each parameter. However, we actually expect to obtain a small value as possible due to the presence of measuring noise and numerical calculating errors. It is worth noting that Eqs. (1) and (2) are implicit and not solvable analytically. With the help of the objective function described by Eq. (4), the optimization procedure has the advantage of no requirement of solving numerically the implicit function.

Notice that F is a nonlinear function. This means that the objective function is not quadratic, possessing a single global minimum. This will result in local minima, which in turn might attract the solution into one of them depending on the starting position. Nonlinearity also makes it necessary to use iterative schemes, which collect local gradient information or a Hessian matrix. However, in cases of discontinuous types of nonlinearity and/or unobservable states, the estimation of the gradients or the Hessian matrix is not feasible. So a different approach should be sought that does not require any derivative calculations. These methods should only rely on the objective function evaluations. In the past, there are many such conventional methods available such as in the quasi-Newton method or its variation; however, these methods perform well locally but are trapped in local minima easily. Much attention has also been paid recently to heuristic methods such as GAs, which are designed to work on a discrete parameter space. In this case, any attempt to discretize parameters can only increase the complexity of the problem.[23]

In this article, we try to apply the PSO algorithm, as discussed below, to adjust the parameters Θ in Eq. (4). Our objective is that the value of the objective function F denoted by Eq. (4) is minimized, and then approaching zero as much as possible. In general, the PSO algorithm only needs to evaluate the objective function to guide its search and does not require for derivatives about the system. Obviously, parameter extraction essentially is a multidimensional numerical optimization problem, where decision variables are parameters.

IV. PARTICLE SWARM OPTIMIZATION ALGORITHM

The development of PSO is based on observations of the social behavior of animals such as bird flocking, fish schooling, and swarm theory. PSO is initialized with a population of random solutions. In a PSO algorithm, the individuals, called particles, are flown or swum through hyperspace. Each particle tries to search the best position (state) with time in a multidimensional space. During flight or swim, each particle adjusts its position in light of its own experience and the experiences of neighbors, including the current velocity and position and the best previous position experienced by itself and its neighbors. For more information, we refer reader to Ref. 24, a standard textbook on PSO, treating both the social and computational paradigms.

The PSO algorithm is initialized with a group of random particles (solutions) and then searches for optima by updating generations. Particles profit from the discoveries and previous experience of other particles during the exploration and search for better objective function values. Let i indicate a particle's index in the swarm. Each particle flies through the d-dimensional search space R^d with a velocity v_i, which is dynamically adjusted according to its own previous best solution $p\text{best}_i$ and the previous best solution $g\text{best}$ of the entire swarm. In other words, PSO utilizes $p\text{best}_i$ and $g\text{best}$ to modify the current search point to avoid the particle "flying" in the same direction. The velocity updates are calculated as a linear combination of position and velocity vectors. The particles interact and move according to the following equations:

$$v_i(j+1) = w(j)v_i(j) + c_1 r_1(j)[p\text{best}_i(j) - x_i(j)] + c_2 r_2(j) \times [g\text{best}(j) - x_i(j)], \tag{5}$$

$$x_i(j+1) = v_i(j+1) + x_i(j), \tag{6}$$

where each particle's position x_i (corresponding to a set of solar cell parameter values) represents a possible solution point in the problem search space, and r_1 and r_2 are random numbers between 0 and 1. c_1 and c_2 are learning factors, usually about $c_1=c_2=2$. And w is an inertia weight, which plays an important role in balancing the global search and local search. A large inertia weight facilitates a global search while a small inertia weight facilitates a local search. It can be a positive constant or a positive decreasing linear function of iteration index j. In this investigation, the inertia weight is the following decreasing linear function:

$$w(j) = w_{\max} - (w_{\max} - w_{\min})\frac{j}{j_{\max}}, \tag{7}$$

where $w_{\max}$ and $w_{\min}$ are the final weight and the initial weight, respectively, and $j_{\max}$ is the maximum iteration times. Using the above equation, the diversification characteristic is gradually decreased and a certain velocity, which gradually moves the current searching point close to $p\text{best}_i$ and $g\text{best}$, can be calculated. Through empirical studies, Shi and Eberhart[25] observed that the optimal solution can be improved by varying the value from 0.9 at the beginning of the search to 0.4 at the end of the search for most problems. By using a linearly decreasing inertia weight, the performance of the PSO algorithm can be improved greatly and have better results than that of the PSO, whose inertia weight is a positive constant.

For the solar cell parameter extraction problem, the fitness function is equal to the objective function as described in Eq. (4). Namely, the smaller the objective function, the better the fitness of an individual.

The PSO algorithm is described as follows.

(1) Set the iteration index j to zero. Initialize randomly the swarm S_w of N_s particles so that the position $x_i(0)$ and velocity $v_i(0)$ of each particle meet the prescribed conditions. Also, initialize the inertia weight $w(0)$. Then, a set of values for the I-V characteristics serves as the input data for the PSO algorithm.

(2) Based on Eq. (4), evaluate the fitness of each particle $F[x_i(j)]$.

(3) Compare the personal best of each particle to its current fitness, and set $p\text{best}_i(j)$ to the better performance, namely,

$$p\text{best}_i(j) = \begin{cases} p\text{best}_i(j-1) & \text{if } F(x_i(j)) \geq F(p\text{best}_i(j-1)) \\ x_i(j) & \text{if } F(x_i(j)) < F(p\text{best}_i(j-1)). \end{cases} \tag{8}$$

(4) Set the global best $g\text{best}(j)$ to the position of the particle with the best fitness within the swarm, namely,

$$g\text{best}(j) \in \{p\text{best}_1(j), p\text{best}_2(j), \ldots, p\text{best}_{N_s}(j)\} | F(g\text{best}(j)) = \min\{F(p\text{best}_1(j)), F(p\text{best}_2(j)), \ldots, F(p\text{best}_{N_s}(j))\}. \tag{9}$$

(5) Change the velocity vector for each particle according to Eq. (5), then the velocity of each particle is updated according to the following relation:

$$v_i(j+1) = \begin{cases} V_{\max} & \text{if } v_i(j+1) > V_{\max} \\ -V_{\max} & \text{if } v_i(j+1) < -V_{\max} \\ v_i(j+1) & \text{otherwise}, \end{cases} \tag{10}$$

where $V_{\max}$ is a constant in order to clamp the excessive roaming of particles. Generally, the choice of a $V_{\max}$ equals to the maximum allowable excursion of any particle in that dimension.[26]

(6) Move each particle to its new position, according to Eq. (6).

(7) Update the inertia weight according to Eq. (7).

(8) Let $j=j+1$.

(9) Go to step (2), and repeat until meets the stop criteria. The stop criteria can be that the maximum iteration number $j_{\max}$ is reached or the minimum fitness function $F_{\min}$ condition is satisfied.

It can be easily seen that there are two key steps when applying the PSO algorithm to optimization problems: the representation of the solution and the fitness function. One of the desirable merits of the PSO algorithm is that it takes real numbers as particles. It is not like GAs, where transformation of binary encoding and special genetic operators are needed. Also, the PSO is free from the complex computation in GAs (e.g., selection, crossover, and mutation) The complete application of PSO, as well as the method to do parameter extraction of solar cells, is discussed in Sec. V.

V. EXTRACTION RESULTS AND DISCUSSION

To verify the proposed method in the parameter extraction of solar cell models, both synthetic and real world experimental I-V data have been used as an evaluation of the extracting power of the PSO method, which has been implemented by MATLAB. The calculation was processed in a 1.2 GHz Mobile Intel processor, under Windows XP. The effectiveness of the parameter extraction process will be assessed in terms of the shape of the obtained I-V characteristics, fitness function value, computational efficiency, and relative error of extracted parameters.

In the following experiments, we considered that the controlling variables used in PSO operations were given by learning factors $c_1=c_2=2$, population size $N_s=60$, and maximal iteration number $j_{\max}=40\,000$. The inertia weight $w(j)$ was taken as a decreasing linear function in iteration index j from 0.9 to 0.4, i.e., $w_{\max}=0.9$ and $w_{\min}=0.4$, which are as the same as those recommended by other papers[25–27] and the values do not depend on the problems.

In our investigation, in order to compare the performance of the GA with PSO methods, we also implemented the parameter extraction of solar cell models by using a real-value GA method with the same fitness function, individual definition, maximal iteration index, population size, and search range as the PSO method. The following real-value GA controlling parameters have been used: The crossover rate was 0.8, and the mutation rate was 0.2. The elite strategy was used, where the best individual of each iteration was copied into the succeeding iteration in order to speed convergence. The GA method was carried out by the function "ga" available in the genetic method and direct search toolbox of the MATLAB software.

A. Parameter extraction with synthetic *I-V* data

First, we consider parameter extraction of solar cell with synthetic I-V data in order to validate the accuracy of the PSO method. For the sake of comparison, the synthetic I-V data were calculated by using the single diode model with parameters $R_s=0.0364\ \Omega$, $R_{\text{sh}}=53.76\ \Omega$, $I_{\text{ph}}=0.7608$ A, $I_{SD1}=3.223\times10^{-7}$ A, and $n_1=1.4837$ at 306 K as reported in several literatures,[4–7,28,29] as well as the double diode model parameter values of $R_s=0.031\,17\ \Omega$, $R_{\text{sh}}=19.92\ \Omega$, $I_{\text{ph}}=0.9072$ A, $I_{SD1}=2.831\times10^{-5}$ A, $I_{SD2}=2.466\times10^{-9}$ A, $n_1=2$, and $n_2=1$ at 323 K as reported in a previous literature.[21] Before invoking the PSO and GA algorithms, a search range has to be specified for all solar cell parameters. In this subsection, the search range was set to $\pm100\%$ of the parameter values specified above.

The evolving processes of different methods with synthetic I-V data for both models are illustrated in Fig. 2. Although the GAs converge rapidly, there still is a difficulty in converging to the global optimal solution. This drawback of the GA method can lead to poor performance regarding the accuracy of the extracted parameters. It is clear that the PSO

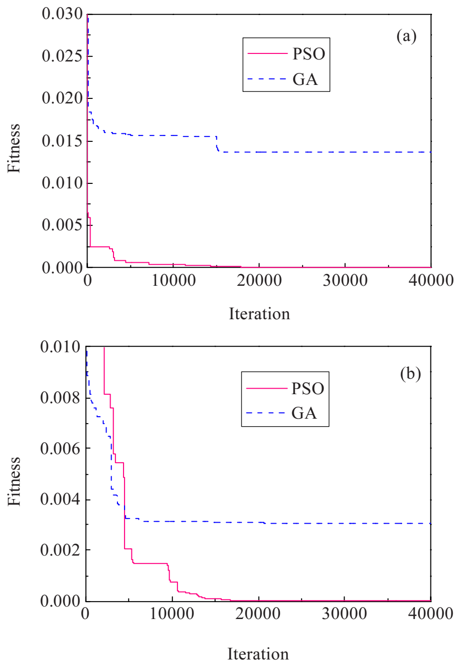

FIG. 2. (Color online) Evolving processes of different methods with synthetic *I-V* data for both (a) the single and (b) the double diode models.

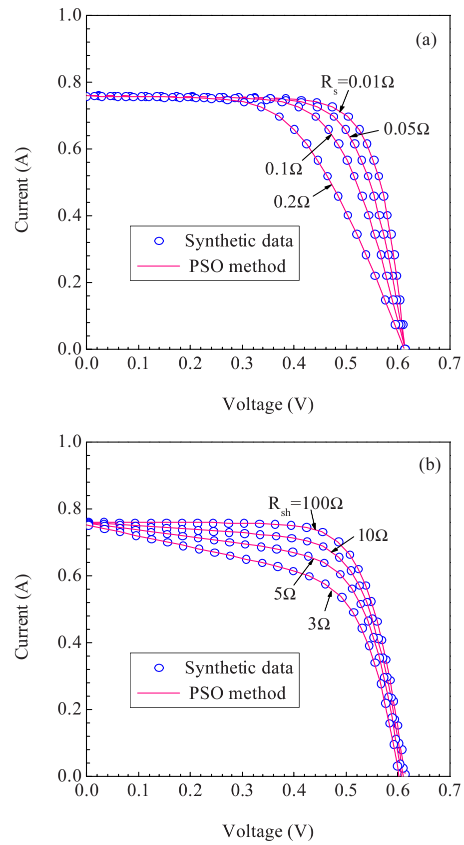

FIG. 3. (Color online) Synthetic data and the *I-V* characteristics obtained by the PSO method for the single diode model with (a) different series resistances (R_{sh} fixed at 53.76 Ω) and (b) different shunt resistances (R_s fixed at 0.0364 Ω).

method has much lower fitness value than the GA method if the number of iteration times is large enough, so the PSO method can generate higher quality solutions than the GA method.

The resultant parameter values of the solar cell are summarized in Table I. Also presented are the parameters extracted by the GA method. Additionally, the minimum fitness function values and computational efficiency (i.e., less time consumption) are listed in the last two row of Table I. As can be seen, for both the single and the double diode models, the values of solar cell parameters extracted by the PSO method are very close to the real parameters. Based on the proposed method, the maximum relative errors of all extracted parameters are less than 1.68%, which corresponds to good parameter extraction. Especially, in the case of double diode model, relative errors of less than 0.14% correspond to nearly perfect parameter extraction. However, with the GA method, the relative errors of all extracted parameters are much more than those obtained by the PSO method for the same search range. Moreover, as shown in Table I, it is observed that the minimum fitness function values F_{min} of the PSO method are far less than that of the GA method. The proposed method requires less time consumption than the GA method when the exact values have been determined for each parameter.

Furthermore, Figs. 3 and 4 show a comparison between the synthetic *I-V* data of the solar cells and the *I-V* characteristics derived from the parameters extracted by the PSO method for the single and double diode models, respectively. It should be noted that the different series and shunt resistances are considered. For both models, when the series resistance and shunt resistance are varied from 0.01 to 0.2 Ω and from 3 to 100 Ω, respectively, the *I-V* characteristics obtained by the PSO method are also in good agreement with synthetic data over its whole range.

TABLE I. Results of parameter extraction for both the single and double diode models with synthetic *I-V* data.

	Single diode model (T=306 K)			Double diode model (T=323 K)		
	Real values	PSO method	GA method	Real values	PSO method	GA method
R_s (Ω)	0.0364	0.036 394 0	0.069 709 3	0.031 17	0.031 172 9	0.021 081 4
R_{sh} (Ω)	53.76	53.796 5	10.084 3	19.92	19.889 1	31.693 3
I_{ph} (A)	0.7608	0.760 798	0.775 626	0.907 2	0.907 209	0.903 942
I_{SD1} (A)	3.223×10^{-7}	$3.227\ 21\times10^{-7}$	$5.549\ 08\times10^{-7}$	2.831×10^{-5}	$2.788\ 66\times10^{-5}$	$1.462\ 59\times10^{-5}$
I_{SD2} (A)	⋯	⋯	⋯	2.466×10^{-9}	$2.424\ 7\times10^{-9}$	$4.431\ 47\times10^{-9}$
n_1	1.4837	1.483 82	0.805 345	2	1.996 17	1.763 14
n_2	⋯	⋯	⋯	1	0.999 20	1.059 71
F_{min}	⋯	1.68×10^{-6}	1.37×10^{-2}	⋯	2.73×10^{-6}	3.07×10^{-3}
Time (s)	⋯	242	321	⋯	346	378

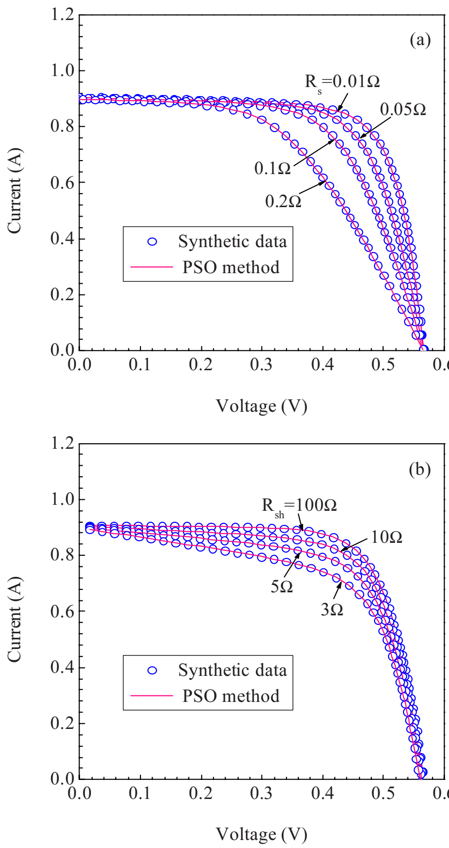

FIG. 4. (Color online) Synthetic data and the *I*-*V* characteristics obtained by the PSO method for the double diode model with (a) different series resistances (R_{sh} fixed at 19.92 Ω) and (b) different shunt resistances (R_s fixed at 0.03117 Ω).

These results show that the use of the PSO method for solar cell parameter extraction significantly decreases the errors in the extracted values, and hence improves the accuracy of the determined parameters. This may be attributable to the fact that PSO has memory and cooperates to move toward a region containing the global or a near-optimal solution.[30]

B. Parameter extraction with experimental *I-V* data

Second, to further validate the efficiency of the present extraction method, it is again applied but now to experimental *I*-*V* data of a commercial solar cell at 328 K. The *I*-*V* characteristics calculated by using extracted parameter values were also offered so as to produce results, which is comparable to the experimental data.

Figure 5 shows the evolving processes of different methods with experimental *I*-*V* data for both the single and the double diode models. Obviously, in this case, the PSO method also has a lower fitness values than the GA method, when the maximum iteration number is reached.

In the case of experimental *I*-*V* data, the results of the parameter extraction are listed in Table II, where the search ranges specified for each parameter are also listed. Due to using the experimental data, there is no way to know how good the results are obtained. For this reason, any advance in achieving a best value of the fitness function is very important as it leads to improvement in the knowledge of the real

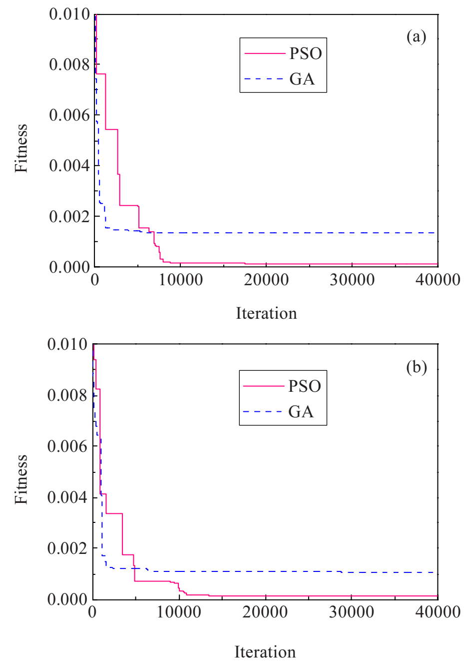

FIG. 5. (Color online) Evolving processes of different methods with experimental *I*-*V* data for both (a) the single and (b) the double diode models.

values of the extracted parameters. The results of the GA method are also given in Table II. Through the examination of the table, it can be seen that the results obtained by the PSO method outperform those obtained by the GA method, whether the minimum fitness function values $F_{\min}$ and the computation time displayed the last two row of Table II in both models. A high-quality solution within a relative short computation time can be achieved by the PSO method, when the maximum iteration number is reached. The $F_{\min}$ obtained by the PSO method are lower one order of magnitude than those obtained by the GA method. As listed in Table II, we can observe that the PSO does not particularly necessitate initial guesses as close as possible to the solutions in the parameters' extraction of the solar cell, while only requires a broad search range specified for each parameter.

Figure 6 shows the comparisons between the experimental *I*-*V* data of a commercial solar cell and the *I*-*V* characteristics obtained by the PSO method for both models. As shown in the figures, the *I*-*V* characteristics almost approach the experimental data, which attest the quality of the parameter extraction.

To investigate the effect of population size on the performances of PSO, experiments were carried out under different population sizes but other controlling variables fixed. Figure 7 illustrates the effect of population size on the minimum fitness function values for both models. As shown in the figure, the minimum fitness function values obtained by the PSO method are much lower than those obtained by the GA method in all population size cases in both of the single diode mode or the double diode model. Generally, as population size increases, the results obtained by the PSO method will become better at the cost of more computation time. However, in this investigation, the fitness function values only appear a little fluctuation with different population sizes unless the population size is too small. If the population size

TABLE II. Results of parameter extraction for both the single and double diode models with experimental *I-V* data.

		Single diode model		Double diode model	
	Search ranges	PSO method	GA method	PSO method	GA method
R_s (Ω)	[0,0.05]	0.023 273 6	0.021 928 8	0.023 354 5	0.023 098 0
R_{sh} (Ω)	[0,250]	112.099	171.998	112.751	166.993
I_{ph} (A)	[1,2]	1.540 03	1.540 60	1.540 03	1.540 59
I_{SD1} (A)	$[0,1\times10^{-4}]$	$5.759\ 39\times10^{-5}$	$7.833\ 83\times10^{-5}$	$5.801\ 07\times10^{-5}$	$8.028\ 65\times10^{-5}$
I_{SD2} (A)	$[0,5\times10^{-8}]$	⋯	⋯	$4.325\ 76\times10^{-9}$	$4.220\ 08\times10^{-9}$
n_1	[0,3.5]	1.696 17	1.743 60	1.698 60	1.762 59
n_2	[0,2]	⋯	⋯	1.190 04	1.187 68
F_{min}	⋯	1.45×10^{-4}	1.37×10^{-3}	1.40×10^{-4}	1.08×10^{-3}
Time (s)	⋯	243	290	318	337

is too small, the results will be poor because the solution space cannot be explored enough. As a consequence, considering both the searching quality and computation time, it is recommended to choose population size between 30 and 90 in the parameter extraction of solar cells.

It should be noted that the parameter precision can obviously be improved by using more data points because the number of data points has a varying influence.

VI. CONCLUSIONS

This paper has presented a new technique of parameter extraction of silicon solar cells using the PSO method. The validity of the proposed method has been confirmed by applying it to both experimental and synthetic *I-V* data. It is clear from the results that the proposed method outperforms the GA method in the shape of the fitted *I-V* characteristics, fitness function value, parameter precision, and computation time for both the single and the double diode models. In addition, compared with conventional gradient-based methods, the PSO method does not particularly necessitate initial guesses as close as possible to the solutions, while only required a broad range specified for each of the parameters. So the proposed technique is accurate, fast, and easily applicable for the parameter extraction of solar cells from illuminated *I-V* characteristics.

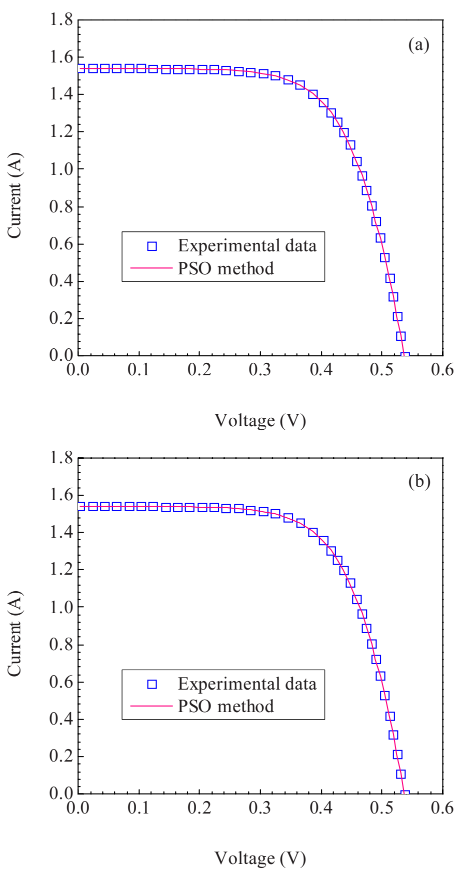

FIG. 6. (Color online) Experimental data of a commercial solar cell and the *I-V* characteristics obtained by the PSO method for both (a) the single and (b) the double diode models.

ACKNOWLEDGMENTS

This work has been financially supported by the National Natural Science Foundation of China (Grant No. 10672147) and Zhejiang Provincial Natural Science Foundation of

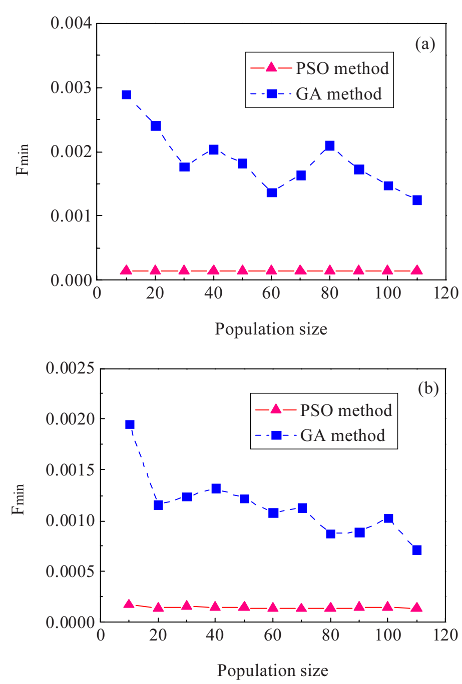

FIG. 7. (Color online) Minimum fitness function values obtained by the PSO and GA methods with different population sizes for both (a) the single and (b) the double diode models.

China (Grant No. Y106786).

[1]K. I. Ishibashi, Y. Kimura, and M. Niwano, J. Appl. Phys. **103**, 094507 (2008).
[2]A. Ortiz-Conde, F. J. García Sánchez, and J. Muci, Sol. Energy Mater. Sol. Cells **90**, 352 (2006).
[3]Priyanka, M. Lal, and S. N. Singh, Sol. Energy Mater. Sol. Cells **91**, 137 (2007).
[4]M. Chegaar, Z. Ouennoughi, and A. Hoffmann, Solid-State Electron. **45**, 293 (2001).
[5]K. Bouzidi, M. Chegaar, and A. Bouhemadou, Sol. Energy Mater. Sol. Cells **91**, 1647 (2007).
[6]T. Easwarakhanthan, J. Bottin, I. Bouhouch, and C. Boutrit, Int. J. Sol. Energy **4**, 1 (1986).
[7]M. Chegaar, Z. Ouennoughi, and F. Guechi, Vacuum **75**, 367 (2004).
[8]J. A. Jervase, H. Bourdoucen, and A. Al-Lawati, Meas. Sci. Technol. **12**, 1922 (2001).
[9]A. Sellai and Z. Ouennoughi, Int. J. Mod. Phys. C **16**, 1043 (2005).
[10]A. Sellami, M. Zagrouba, M. Bouaïcha, and B. Bessaïs, Meas. Sci. Technol. **18**, 1472 (2007).
[11]Z.-L. Gaing, IEEE Trans. Power Syst. **18**, 1187 (2003).
[12]J. Kennedy and R. Eberhart, in *Conference on Neural Networks* (IEEE, Perth, Aust, 1995), Vol. 4, pp. 1942–1948.
[13]L. L. Li, L. Wang, and L. H. Liu, Appl. Math. Comput. **179**, 135 (2006).
[14]K. Nishioka, N. Sakitani, Y. Uraoka, and T. Fuyuki, Sol. Energy Mater. Sol. Cells **91**, 1222 (2007).
[15]J. Pallarès, R. Cabré, L. F. Marsal, and R. E. I. Schropp, J. Appl. Phys. **100**, 084513 (2006).
[16]S. Yoo, B. Domercq, and B. Kippelen, J. Appl. Phys. **97**, 103706 (2005).
[17]V. Yong, S. T. Ho, and R. P. H. Chang, Appl. Phys. Lett. **92**, 143506 (2008).
[18]K. Araki and M. Yamaguchi, Sol. Energy Mater. Sol. Cells **75**, 457 (2003).
[19]M. Wolf, G. T. Noel, and R. J. Stirn, IEEE Trans. Electron Devices **ED-24**, 419 (1977).
[20]B. Mazhari, Sol. Energy Mater. Sol. Cells **90**, 1021 (2006).
[21]D. S. H. Chan and J. C. H. Phang, IEEE Trans. Electron Devices **ED-34**, 286 (1987).
[22]C. L. Garrido-Alzar, Renewable Energy **10**, 125 (1997).
[23]A. Kyprianou, K. Worden, and M. Panet, J. Sound Vib. **248**, 289 (2001).
[24]J. Kennedy, R. C. Eberhart, and Y. Shi, *Swarm Intelligence* (Morgan Kaufmann, San Francisco, CA, 2001).
[25]Y. H. Shi and R. C. Eberhart, in *Congress on Evolutionary Computation* (IEEE, Seoul, South Korea, 2001), pp. 101–106.
[26]Y. Shi and R. Eberhart, in *Conference on Evolutionary Programming* (Springer-Verlag, New York, 1998), pp. 591–600.
[27]H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Nakanishi, IEEE Trans. Power Syst. **15**, 1232 (2000).
[28]M. Chegaar, G. Azzouzi, and P. Mialhe, Solid-State Electron. **50**, 1234 (2006).
[29]Z. Ouennoughi and M. Chegaar, Solid-State Electron. **43**, 1985 (1999).
[30]J. R. Pérez and J. Basterrechea, IEEE Trans. Antennas Propag. **55**, 549 (2007).