10.1049/iet-rpg.2016.0473

IET Renewable Power Generation

Special Issue: Active Power Control of Renewable Energy Generation Systems

ISSN 1752-1416
Received on 3rd June 2016
Revised 1st September 2016
Accepted on 23rd November 2016
E-First on 31st January 2017
doi: 10.1049/iet-rpg.2016.0473
www.ietdl.org

Simplified swarm optimisation for the solar cell models parameter estimation problem

Wei-Chang Yeh[1], Peijie Lin[2] ✉, Chia-Ling Huang[3]

[1]*Department of Industrial Engineering and Engineering Management, National Tsing Hua University, P.O. Box 24-60, Hsinchu 300, Taiwan*
[2]*School of Physics and Information Engineering, Fuzhou University, 2 Xue Yuan Road, University Town, Fuzhou, Fujian, 350108, People's Republic of China*
[3]*Department of Logistics and Shipping Management, Kainan University, No. 1, Kainan Road, Taoyuan 33857, Taiwan*
✉ *E-mail: linpeijie@fzu.edu.cn*

Abstract: Solar energy applications and research are becoming increasingly popular, and photovoltaics (PVs) are among the most significant solar energy applications. To simulate and optimise PV system performance, the optimal parameters of the solar cell models should be estimated exactly. In this study, improved simplified swarm optimisation (iSSO), a recently introduced soft computing method based on simplified swarm optimisation, is proposed to minimise the least square error between the extracted and the measured data for the solar cell models parameter estimation of the single- and double-diode model problems. Based on the new all-variable difference update mechanism and survival of the fittest policy, the proposed algorithm is able to find an improved approximation for estimating the parameters of single- and double-diode solar cell models. As evidence of the utility of the proposed iSSO, the authors present extensive computational results for two benchmark problems. The comparison of the computational results supports the proposed iSSO algorithm outperforms the previously developed algorithms for all of the experiments in the literature.

1 Introduction

The global installed capacity of photovoltaic (PV) systems has noticeably increased and reaches 227 GW at the end of 2015 [1]. To comprehend the characteristics and then optimise the design of PV systems, an accurate PV simulator is essential [2, 3]. As the principally component of a PV system, the model of solar cell should be set up accurately. The main work for establishing solar cell models is to extract the parameters of the equivalent circuit model, which contains two main steps: the mathematical model formulation and the subsequent accurate estimation of their parameter values [4]. The mathematical model of the solar cell is a nonlinear current versus voltage (I–V) characteristic curve. To describe this characteristic curve of the solar cell, various equivalent electronic circuit models have been developed over the past several decades, from the single-diode model (SDM) to more complex model, such as the double-diode model (DDM), the modified-double diode (MDDM) and the three-diode model (TDM). The SDM is the simplest one and as a trade-off between accuracy and simplicity [5]. In DDM, two diodes are considered to reflect the effect of recombination current loss in the depletion region [6]. To take into account the effect of grain boundary region, an additional resistance RS_2 is added in series with the second diode in the MDDM [7]. In the TDM, the influence of grain boundaries and large leakage current are taken into consideration [8]. In practical application, the SDM and DDM are the most common models, which have five and seven unknown parameters to be extracted, respectively[9, 10]. The SDM and DDM are applied in this paper.

In recent years, various methods have been proposed to solve the solar cell model parameter estimation problem [11]. These methods can be categorised into three types: analytical approach, numerical approach, and soft computing. The analytical approach is generally formulated by applying elementary functions [12–14] or the Lambert W-function [15, 16] that only needs some particular point values of the I–V solar cell curve, i.e.short-circuit current (I_{SC}), open-circuit voltage (V_{OC}), the current and voltage (I_{MPP} and V_{MPP}) at the maximum power point (MPP), and the slopes of the I–V curve at the short-circuit point and open-circuit point. All of the analytical approaches heavily depend on the precision of the selected points, and any improperly selected points may lead to erroneous results [2]. Furthermore, analytical approaches often need to make some approximations during computation, which may also cause a significant difference between the simulated data and experimental data. The solar cell model parameter extraction is an NP-hard problem with a non-linear, multi-variable, multi-modal problem with numerous local optima [17]. Therefore, it is very difficult and unreasonable to use the analytical approach to obtain the exact solution.

The goal of the numerical approach is to find a good approximated solution for the solar cell model parameter extraction. The numerical approach usually needs to fit all of the points on the I–V curve using certain mathematical algorithms, such as the Newton–Raphson method [18, 19] and the Levenberg–Marquardt algorithm [20, 21]. One of the major drawbacks of most numerical approaches is that the accuracy of the results relies on the selection of the method type and the initial values. Basically, the numerical approach is similar to a local search method and is able to find an approximation only if it is not trapped in the local optimiser for the solar cell model parameter extraction.

To overcome the two obstacles of the exponential runtime for the analytical approach and the high possibility of becoming trapped in the local optimiser, more and more scholars have implemented soft computing to find a good approximation for this problem within a reason time due to the powerful global search capability, genetic algorithm (GA) [22, 23] is proposed to identify the electrical parameters of single-diode photovoltaic solar cells and modules, which obtains better result than conventional numerical approaches. Particle swarm optimisation (PSO) [24, 25] is applied to extract the parameter of silicon solar cell, the validity of the proposed method has been confirmed by applying it to both experimental and synthetic I–V data. Artificial bee colony (ABC) is developed to accurately identify the solar cells parameters. In comparison with other evolutionary algorithms, ABC exhibits a better search capacity to face multimodal objective functions [26]. Flower pollination algorithm (FPA) [4] shows an unnoticed deviation between the estimated and experimental data and has the highest speed of conversion to the optimal solutions with the

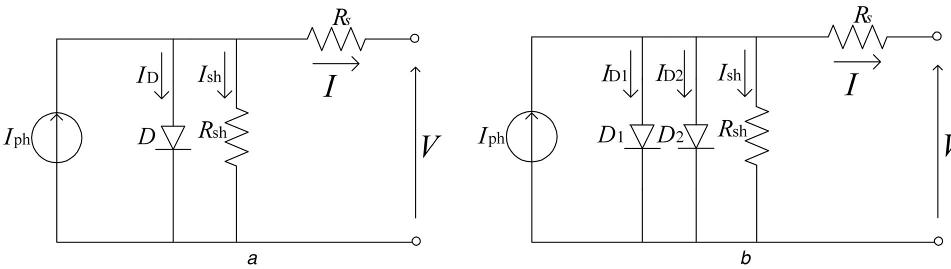

Fig. 1 *Two equivalent solar cell models*
(a) SDM, ***(b)*** DDM

shortest convergence time. In addition, simulated annealing [27], pattern search [28], harmony search [9], cuckoo search [29], artificial bee swarm optimisation [30], and mutative-scale parallel chaos optimisation algorithm [31] are proposed to identify the parameters of solar cell models.

Although soft computing achieves better results than the analytical approaches and the numerical approaches, the complex update mechanisms (UMs) of some evolutionary algorithms may cause efficiency loss. For instance, GA requires genetic operations for crossover and mutation, the PSO algorithm must calculate both the velocity and position functions, each onlooker in ABC may calculate more than once fitness function in each generation, and the Lévy flights in the Rule 1 of FPA may be time consuming. Thus, there is always a need for more simplicity, efficiency, and flexibility in soft computing for the solar cell model parameter estimation problem.

Simplified swarm optimisation (SSO) is an emerging population-based stochastic optimisation method and belongs to both swarm intelligence and evolutionary computation [32–36]. The UM of SSO is much simpler than that of the other major soft computing techniques. Simulation results from SSO demonstrate that it has excellent convergence to quality solutions. In addition, to address many optimisation problems, SSO has advantages of simplicity, efficiency, and flexibility in exploring large and complex spaces [32–38].

Therefore, based on the new all-variable difference UM and survival of the fittest policy, an alternative method using the recently introduced improved SSO (iSSO) for identifying the parameters of solar cells is proposed. Moreover, the FPA-based solar cell models parameter extraction approach has the most outstanding performance among the aforementioned algorithms according to the literature [4]. Therefore, FPA is selected for an in-detail compared to the proposed method.

2 Solar cell models problem formulation

The SDM and DDM are the most popular equivalent models in practice, which are illustrated in Fig. 1.

2.1 Single-diode model

A solar cell is fundamental a semiconductor diode and its p–n junction exposes to light [39]. Ideally, a solar cell can be modelled as a current source (I_{ph}) which is shunted with a diode (D). However, to be realistic, a series resistance (R_s) that presents the contact resistance and the resistance in the material and of the electrodes needs to be considered [29]. Moreover, a shunt resistance (R_{sh}) that lead to the leakage current should also be added to the model. Hence, the equivalent circuit of the single-diode solar cell model is shown in Fig. 1*a*. According to Kirchhoff's current law (KCL), the solar cell terminal current I_t is calculated as follows:

$$I_t = I_{ph} - I_D - I_{sh} \tag{1}$$

where I_{ph} is the photo generated current, I_D is the diode current as calculated by (2) using the Shockley equation, and I_{sh} is the shunt current, which is calculated as shown in (3)

$$I_D = I_{sd}\left[\exp\left(\frac{q(V_t + R_s \cdot I_t)}{n \cdot k \cdot T}\right) - 1\right] \tag{2}$$

$$I_{sh} = \frac{V_t + R_s \cdot I_t}{R_{sh}} \tag{3}$$

where I_{sd} is the reverse saturation current, V_t is the solar cell terminal voltage, R_s and R_{sh} are the series and shunt resistances, respectively, n is the ideality factor, k is Boltzmann's constant (1.380×10^{-23} J/K), q is the electronic charge (1.602×10^{-19} C) and T is the cell absolute temperature. By substituting (2) and (3) into (1), the solar cell terminal current can be rewritten as follows:

$$I_t = I_{ph} - I_{sd}\left[\exp\left(\frac{q(V_t + R_s \cdot I_t)}{n \cdot k \cdot T}\right) - 1\right] - \frac{V_t + R_s \cdot I_t}{R_{sh}} \tag{4}$$

Therefore, the five unknown parameters to be extracted in SDM are I_{ph}, I_{sd}, n, R_s, and R_{sh}. This problem can be solved using the optimisation algorithm based on a given measured set of $I_t - V_t$ data for the solar cell.

2.2 Double-diode model

In DDM, to consider the effect of recombination current loss in the depletion region, two diodes are considered and they are in parallel with the current source [6]. The equivalent circuit of the DDM is shown in Fig. 1*b*. Similar to the SDM, by applying KCL and the Shockley equation, the terminal current of the DDM can be expressed as follows:

$$\begin{aligned} I_t = I_{ph} &- I_{sd1}\left[\exp\left(\frac{q(V_t + R_s \cdot I_t)}{n_1 \cdot k \cdot T}\right) - 1\right] \\ &- I_{sd2}\left[\exp\left(\frac{q(V_t + R_s \cdot I_t)}{n_2 \cdot k \cdot T}\right) - 1\right] - \frac{V_t + R_s \cdot I_t}{R_{sh}} \end{aligned} \tag{5}$$

where I_{sd1} and I_{sd2} are the diffusion and saturation currents, respectively, and n_1 and n_2 are the two diode ideality factors, respectively. The other terms are introduced in Section 2.1. There are seven parameters to be extracted in DDM, namely, I_{ph}, I_{sd1}, I_{sd2}, n_1, n_2, R_s, and R_{sh}.

2.3 Optimisation process for parameter extraction of solar cell models

Solar cell parameters extraction using the optimisation algorithm usually consists of the following steps [31]. A set of measured $I_t - V_t$ data of solar cell is first prepared, followed by defining an appropriate objective. Then, the proposed optimisation algorithm is applied to minimise the objective function value until the stopping criterion is reached. The solution of the optimal objective function is the extracted optimal parameter values.

In this work, the objective function is defined as the root mean square error (RMSE) between the experimental data and estimated values, which is described by the following equation:

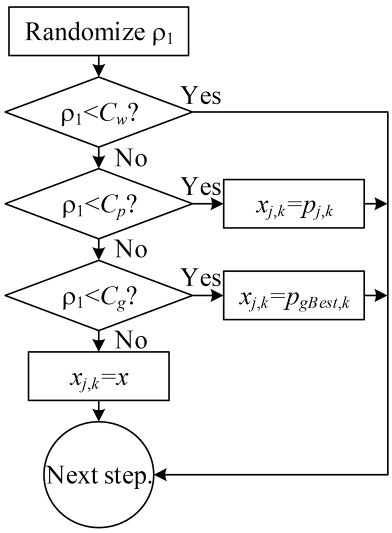

Fig. 2 *Block diagram of SSO in updating* $x_{j,k}$

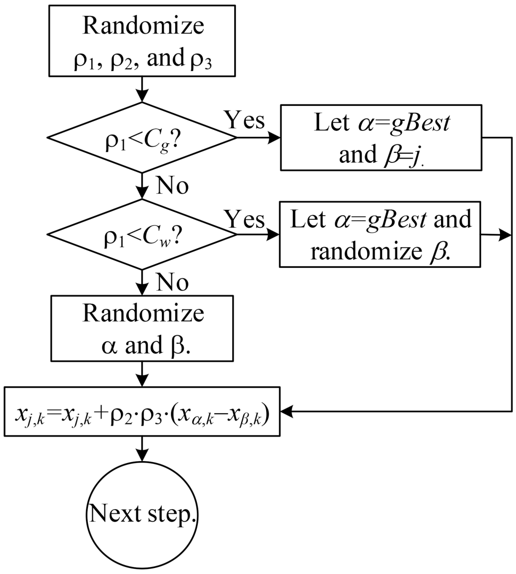

Fig. 3 *Block diagram for the proposed iSSO in updating* $x_{j,k}$

$$F(\boldsymbol{X}) = \sqrt{\frac{\sum_{t=1}^{N} \Delta(V_t, I_t, \boldsymbol{X})^2}{N}} \tag{6}$$

where N is the number of the experimental data and $\Delta(V_t, I_t, \boldsymbol{X})$ is the error function, which can be derived using (7) and (8) for the SDM and DDM, respectively,

$$\Delta(V_t, I_t, \boldsymbol{X}) = I_t - I_t(I_{ph}, I_{sd}, n, R_s, R_{sh}) \tag{7}$$

$$\Delta(V_t, I_t, \boldsymbol{X}) = I_t - I_t(I_{ph}, I_{sd1}, I_{sd2}, n_1, n_2, R_s, R_{sh}) \tag{8}$$

In the above function, V_t and I_t are the measured values of the voltage and current of the solar cell, respectively, and $\boldsymbol{X}$ is the vector of the parameters, where $\boldsymbol{X} = (I_{ph}, I_{sd}, n, R_s, R_{sh})$ for SDM and $\boldsymbol{X} = (I_{ph}, I_{sd1}, I_{sd2}, n_1, n_2, R_s, R_{sh})$ for DDM. $I_t(I_{ph}, I_{sd}, n, R_s, R_{sh})$ and $I_t(I_{ph}, I_{sd1}, I_{sd2}, n_1, n_2, R_s, R_{sh})$ are the tth solar cell output currents of SDM and DDM, respectively; these values are calculated by substituting the tth solution $\boldsymbol{X}$ of the optimisation algorithm into (4) and (5), respectively. The proposed optimisation algorithm extracts the parameters by minimising the objective function.

3 SSO and proposed iSSO

This section discusses two cores of the proposed iSSO: the novel all-variable difference UM and the survival of the fittest policy together with a discussion of the complete iSSO pseudocode. Furthermore, an overview of SSO is also given in this section.

3.1 Single-diode model

As proposed originally by Yeh, SSO is a fresh population-based stochastic optimisation method that belongs to both swarm intelligence and evolutionary computing, two popular research topics in soft computing. From various optimisation applications, SSO had proved its ability to produce good-quality solutions to numerous large-scale NP-hard optimisation problems within a reasonable time [32–38, 40].

Let Nsol be the number of solutions and Nvar be the number of variables, $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,N\text{var}})$ be the jth solution with a fitness value $F(X_j)$, *pBest* $P_j = (p_{j,1}, p_{j,2}, \ldots, p_{j,N\text{var}})$ be the best jth solution in the evaluation history, and *gBest* $P_{gBest} = (p_{gBest,1}, p_{gBest,2}, \ldots, p_{gBest,N\text{var}})$ be the best solution among all P_j, where $j = 1, 2, \ldots, N$sol and $gBest \in \{1, 2, \ldots, N\text{sol}\}$. As is true for all of the soft computing techniques, SSO initialises all solutions randomly and updates each solution from generation to generation to search for the optimal solution.

In each generation, SSO updates each variable in each solution, say $x_{j,k}$, based on the following equation as shown in Fig. 2:

$$x_{j,k} = \begin{cases} x_{j,k} & \text{if } \rho_1 \in [0, C_w) \\ p_{j,k} & \text{if } \rho_1 \in [C_w, C_p) \\ p_{gBest,k} & \text{if } \rho_1 \in [C_p, C_g) \\ x & \text{if } \rho_1 \in [C_g, 1] \end{cases}, \tag{9}$$

where ρ_1 is a random variable generated within [0,1], $j = 1, 2, \ldots, N$sol; $k = 1, 2, \ldots, N$var; and C_w, $C_p - C_w$, $C_g - C_p$, and $1 - C_g$ are the predefined probabilities that the current variable will be updated to the same value (i.e., no change), the value of the same position of the variable in its *pBest*, the value of the same position of *gBest*, and a new regenerated random feasible value, respectively.

From (9), it is observed that SSO is flexible and can be integrated with any other UM and is efficient without needing tedious or complex calculations, which makes it suitable for larger problems [32–38, 40]. Therefore, SSO has also been shown to be successful at exploring large and complex spaces for a number of problems [32–38, 40].

3.2 All-variable difference UM

The UM is the core of each soft computing method, and there is always a need to slightly revise the UM based on its original concept to fit different applications in various situations. The SSO UM listed in (9) is more appropriate for limited data values, e.g. discrete data. Hence, (9) is redefined for floating-point data for the solar cell model parameter estimation problem as follows and shown in Fig. 3:

$$x_{j,k} = x_{j,k} + \rho_2 \cdot \rho_3 \cdot \begin{cases} (x_{gBest,k} - x_{j,k}) & \text{if } \rho_1 \in [0, C_g) \\ (x_{gBest,k} - x_{\beta,k}) & \text{if } \rho_1 \in [C_g, C_w), \\ (x_{\alpha,k} - x_{\beta,k}) & \text{otherwise} \end{cases} \tag{10}$$

where ρ_1, ρ_2, and ρ_3 are random numbers generated from the uniform distribution within [0,1], $C_g = 0.4$, $C_w = 0.6$, and all *pBest* are removed from (9).

For example, let $X_2 = (3.1, 5.3, 7.6, 9.8)$, $X_5 = (7.3, 6.7, 4.5, 8.9)$, $X_7 = (6.2, 9.9, 4.9, 7.7)$, and $X_9 = (2.3, 1.7, 5.5, 8.5)$; the second variable (i.e. $c_{2,2}$) in X_2 can be selected randomly to update, and $\rho_2 = 0.2$ and $\rho_3 = 0.7$ can be generated randomly, where $X_{gBest} = X_5$ is the *gBest* and $X_\alpha = X_9$ and $X_\beta = X_7$ are the two randomly selected solutions. The new update X_2 for three different cases for values of ρ_1 are shown as follows:

Case 1: $\rho_1 = 0.21$, original $x_{2,2} = 5.3$, updated $x_{2,2} = 5.3 + 0.2 \cdot 0.7 \cdot (6.7 - 5.3) = 5.496$, updated $X_2 = (3.1, 5.496, 7.6, 9.8)$

Case 2: $\rho_1 = 0.54$, original $x_{2,2} = 5.3$, updated $x_{2,2} = 5.3 + 0.2 \cdot 0.7 \cdot (6.7 - 9.9) = 4.852$, updated $X_2 = (3.1, 4.852, 7.6, 9.8)$

Case 3: $\rho_1 = 0.89$, original $x_{2,2} = 5.3$, updated $x_{2,2} = 5.3 + 0.2 \cdot 0.7 \cdot (1.7 - 9.9) = 4.152$, updated $X_2 = (3.1, 4.152, 7.6, 9.8)$

3.3 Survival of the fittest policy

The survival of the fittest policy, inspired by nature to select the most fit and eliminated unfit options, is adopted in the proposed iSSO and is applied to all of the updated solutions to reduce the evolution time. The complete pseudocode steps of the proposed iSSO are described as follows:

Procedure iSSO

Step 0: Generate $X_j = (x_{j,1}, x_{j,2}, \ldots, x_{j,N\text{var}})$ randomly, calculate the fitness $F(X_j)$ based on (6)–(8) as discussed in Section 2, and find $gBest \in \{1, 2, \ldots, N\text{sol}\}$ such that $F(X_{gBest}) \leq F(X_j)$, where $j = 1, 2, \ldots, N\text{sol}$.

Step 1: Let $j = 1$.

Step 2: Let $X^* = X_j$ and $F^* = F(X_j)$, and generate a random number ρ_1 from the uniform distribution between [0, 1].

Step 3: If $\rho_1 < C_g$, then let $\alpha = gBest$ and $\beta = j$, and go to Step 6.

Step 4: If $\rho_1 < C_w$, then let $\alpha = gBest$, select β randomly from $\{1, 2, \ldots, N\text{sol}\}$, and go to Step 6.

Step 5: Select two integers α and β randomly from $\{1, 2, \ldots, N\text{sol}\}$.

Step 6: Let $x_{j,k} = x_{j,k} + \rho_2 \cdot \rho_3 + (x_{\alpha,k} - x_{\beta,,k})$ for all $k = 1, 2, \ldots, N\text{var}$, and calculate its fitness.

Step 7: If $F(X_j) > F^*$, then let $X_j = X^*$ and $F(X_j) = F^*$ and go to Step 9.

Step 8: If $F(X_j) < F(X_{gBest})$, then let $gBest = j$.

Step 9: If the runtime is greater than or equal to the predefined time limit τ, then X_{gBest} is the final solution and the algorithm is stopped.

Step 10: If $j < N\text{sol}$, let $j = j + 1$ and go to Step 3. Otherwise, go to Step 1.

4 Experimental results

The proposed iSSO algorithm is tested on two benchmarks of the commonly used equivalent electronic circuits of solar cells, i.e. the SDM and DDM. In addition, the performance of the proposed iSSO is compared that of SSO and the best-known algorithm FPA proposed in [4]; FPA was selected as its to achieve a global optimum outperforms that of other evolutionary algorithms [4].

Table 1 Lower and upper bounds of the solar cell parameters

Models	Parameters	Lower bound	Upper bound
SDM	I_{ph}, A	0	1
	I_{sd}, μA	0	1
	n	1	2
	R_s, Ω	0	0.5
	R_{sh}, Ω	0	100
DDM	I_{ph}, A	0	1
	I_{sd1}, μA	0	1
	I_{sd2}, μA	0	1
	n_1	1	2
	n_2	1	2
	R_s, Ω	0	0.5
	R_{sh}, Ω	0	100

4.1 Experimental environment

Both SDM and DDM benchmarks are adopted from the commercial (R.T.C., France) silicon cell with an I–V characteristic curve of a 57-mm, tested under 1 sun (1000 W/m^2) at 33°C [18], and share the same 26 voltages (v) and currents (c), i.e. $N = 26$, represented by (v, c) as follows:

(−0.2057, 0.764), (−0.1291, 0.762), (−0.0588, 0.7605), (0.0057, 0.7605), (0.0646, 0.76),
(0.1185, 0.759), (0.1678, 0.757), (0.2132, 0.757), (0.2545, 0.7555), (0.2924, 0.754),
(0.3269, 0.7505), (0.3585, 0.7465), (0.3873, 0.7385), (0.4137, 0.728), (0.4373, 0.7065),
(0.459, 0.6755), (0.4784, 0.632), (0.496, 0.573), (0.5119, 0.499), (0.5265, 0.413),
(0.5398, 0.3165), (0.5521, 0.212), (0.5633, 0.1035), (0.5736, −0.01), (0.5833, −0.123), and
(0.59, −0.21)

The bounds of the solar cell parameters in both SDM and DDM benchmarks as provided by the literature survey [4, 22–31] are as shown in Table 1.

All algorithms in the experiments are coded using C programming language and implemented on an Intel Core i7-5960X 3.00 GHz CPU with 16 GB of RAM and a Windows 10 64-bit operating system. The stopping criteria are $T = 5$, 10, 15, 20, 25, 30, 35, 40, 45, and 50 in CPU seconds as the runtime unit. The parameters for iSSO, SSO, and FPA are listed below:

iSSO: $c_g = 0.2$, $c_w = 0.65$, $N\text{sol} = 25$;

SSO: $c_g = 0.5$, $c_w = 0.4$, $N\text{sol} = 100$; and

FPA: the probability switch $P_s = 0.8$, $\beta = 3/2$, sig = 0.6966, $N\text{sol} = 25$.

The above parameters for FPA are adopted from [4] for a fair comparison.

In each comparison, there are 55 individual runs for each test, and only the top 50 are recorded to remove outliers. The average, minimal, maximal, and standard deviation of the fitness values are calculated and denoted as F_{avg}, $F_{\min}$, $F_{\max}$, and F_{stdev}, and the best value written in bold.

The average of the corresponding fitness calculation number (N_{avg}), the average of the fitness calculation number that finds the final *gBest* (n_{avg}), the ratio of $n_{\text{avg}}/N_{\text{avg}}$, and the number of *gBest*s defined by N_{gBest} obtained from the related algorithm that outperform the existing solutions are recorded.

In addition, appropriate statistical tests such as the Tukey test of the one-variable analysis of variance (ANOVA), and multivariate ANOVA (MANOVA) are run to investigate the performance of the proposed iSSO.

4.2 Single diode benchmark

The values of F_{avg}, $F_{\min}$, $F_{\max}$, and F_{stdev} obtained from FPA, iSSO, and SSO are recorded in Tables 2–4 for the comparisons of the solution quality and robustness; the values of N_{gBest} are listed in Table 5 to compare the successful rate among FPA, iSSO, and SSO in obtaining a better solution than the existing known solution; the values of N_{avg}, n_{avg}, and $n_{\text{avg}}/N_{\text{avg}}$ are given in Table 6 to demonstrate the efficacy of the proposed iSSO when compared to FPA.

4.2.1 Compare F_{avg}: In Table 3, the proposed iSSO is superior to that of FPA and SSO in F_{avg} for each T. Based on the Tukey test of ANOVA at a significance level of 0.05, there is always a significant difference between both iSSO and SSO, and FPA and SSO, for all T, with similar trends seen in iSSO and FPA except $T = 20$, 25, and 30 in Table 3. Hence, based on solutions, we can determine that iSSO ≤ FPA < SSO (the smaller the better) from the values of F_{avg}.

Table 2 The fitness values of the final *gBests* obtained the proposed iSSO for the **SDM** benchmark

T	F_{avg}	F_{min}	F_{max}	F_{stdev}
5	**0.000986021880377620**	0.0009860218778920	**0.0009860219059790**	$\mathbf{6.9400 \times 10^{-12}}$
10	**0.000986021877893800**	0.0009860218778920	**0.0009860218779050**	$\mathbf{3.4934 \times 10^{-15}}$
15	**0.000986021877892159**	0.0009860218778920	**0.0009860218778940**	$\mathbf{4.6770 \times 10^{-16}}$
20	**0.000986021877891999**	**0.0009860218778910**	**0.0009860218778930**	$\mathbf{2.0202 \times 10^{-16}}$
25	**0.000986021877891939**	**0.0009860218778910**	0.0009860218778920	2.3992×10^{-16}
30	**0.000986021877891959**	**0.0009860218778910**	0.0009860218778920	1.9796×10^{-16}
35	**0.000986021877891859**	**0.0009860218778910**	0.0009860218778920	3.5053×10^{-16}
40	**0.000986021877891859**	**0.0009860218778910**	0.0009860218778920	3.5053×10^{-16}
45	**0.000986021877891879**	**0.0009860218778910**	0.0009860218778920	3.2828×10^{-16}
50	**0.000986021877891839**	0.0009860218778910	0.0009860218778920	3.7035×10^{-16}

Table 3 The comparisons of F_{avg} among iSSO, SSO, and FPA for the SDM benchmark

T	iSSO−SSO	iSSO−FPA	FPA−SSO
5	-1.2033×10^{-4} [a]	-9.4536×10^{-9} [a]	-9.4536×10^{-9} [a]
10	-6.1265×10^{-5} [a]	-7.4089×10^{-11} [a]	-7.4089×10^{-11} [a]
15	-5.1844×10^{-5} [a]	-1.3054×10^{-13} [a]	-1.3054×10^{-13} [a]
20	-5.1768×10^{-5} [a]	-4.6000×10^{-16} [a]	-4.5992×10^{-16} [a]
25	-4.3630×10^{-5} [a]	-6.0000×10^{-17}	-6.0065×10^{-17} [a]
30	-3.2744×10^{-5} [a]	-4.0000×10^{-17}	-4.0115×10^{-17} [a]
35	-2.5466×10^{-5} [a]	-1.2000×10^{-16} [a]	-1.2013×10^{-16} [a]
40	-2.6089×10^{-5} [a]	-1.4000×10^{-16} [a]	-1.4008×10^{-16} [a]
45	-2.1315×10^{-5} [a]	-1.2000×10^{-16} [a]	-1.1991×10^{-16} [a]
50	-2.5950×10^{-5} [a]	-1.4000×10^{-16} [a]	-1.4008×10^{-16} [a]

[a]The average difference is significant at significance level of 0.05 from the Tukey test of ANOVA.

Table 4 The comparisons of F_{min}, F_{max}, F_{stdev} among iSSO, SSO, and FPA for the SDM benchmark

T	iSSO–SSO			iSSO–FPA		
	F_{min}	F_{max}	F_{stdev}	F_{min}	F_{max}	F_{stdev}
5	-7.3628×10^{-6}	-3.4207×10^{-4}	-8.6103×10^{-5}	$0.0000 \times 10^{+0}$	-1.5293×10^{-7}	-2.6110×10^{-8}
10	-1.6381×10^{-6}	-2.8318×10^{-4}	-6.2451×10^{-5}	$0.0000 \times 10^{+0}$	-9.3502×10^{-10}	-1.9843×10^{-10}
15	-2.6648×10^{-6}	-1.7876×10^{-4}	-4.5891×10^{-5}	$0.0000 \times 10^{+0}$	-2.3710×10^{-12}	-4.4230×10^{-13}
20	-1.5571×10^{-6}	-3.3783×10^{-4}	-6.8191×10^{-5}	-1.0001×10^{-15}	-1.0001×10^{-15}	-5.0388×10^{-16}
25	-8.7478×10^{-7}	-1.6602×10^{-4}	-4.0217×10^{-5}	-1.0001×10^{-15}	$0.0000 \times 10^{+0}$	2.3926×10^{-16}
30	-7.0756×10^{-7}	-1.4348×10^{-4}	-3.8414×10^{-5}	-1.0001×10^{-15}	$0.0000 \times 10^{+0}$	1.9731×10^{-16}
35	-4.2741×10^{-7}	-1.1544×10^{-4}	-3.1247×10^{-5}	$0.0000 \times 10^{+0}$	$0.0000 \times 10^{+0}$	2.0910×10^{-16}
40	-5.8939×10^{-7}	-9.7229×10^{-5}	-2.7270×10^{-5}	-1.0001×10^{-15}	$0.0000 \times 10^{+0}$	3.4988×10^{-16}
45	-5.6175×10^{-7}	-1.2086×10^{-4}	-2.7034×10^{-5}	-1.0001×10^{-15}	$0.0000 \times 10^{+0}$	3.2763×10^{-16}
50	-4.8064×10^{-7}	-1.5725×10^{-4}	-3.6283×10^{-5}	$0.0000 \times 10^{+0}$	$0.0000 \times 10^{+0}$	2.2892×10^{-16}

Table 5 The number of N_{gBest}

T	SSO	iSSO	FPA	Tie
5	0	**23**	**23**	9
10	0	**25**	4	28
15	0	**24**	7	31
20	0	**23**	6	32
25	0	**4**	9	51
30	0	7	**10**	48
35	0	6	**9**	49
40	0	**9**	0	46
45	0	**10**	0	45
50	0	**9**	0	46
sum	0	**140**	25	385

4.2.2 Compare F_{min}, F_{max}, and F_{stdev}: From Table 4, the F_{min} of the proposed iSSO is better than that of FPA for $T = 5$, 10, 15, and 50, and equal to that of FPA for the other values of T and the F_{max} of the proposed iSSO is better than that of FPA for $T = 5$, 10, 15, and 20, and equal to that of FPA for the other values of T. iSSO however, shows better performance in F_{stdev} also for $T = 5$, 10, 15, and 20 but FPA has better F_{stdev} for the other values of T. Therefore, iSSO ≤ FPA < SSO (the smaller the better) in terms of overall accuracy and robustness performance.

4.2.3 Compare N_{gBest}: It can be seen that total N_{gBest} for FPA, iSSO, and SSO are 25, 140, and 0 from Table 5. Also, the number of N_{gBest} for iSSO is better than that of FPA and SSO for almost each T. Hence, the proposed iSSO has a higher chance of obtaining a better solution than FPA and SSO, i.e. the proposed iSSO is better than the SSO and FPA in obtaining *gBests* that are better than the existing solutions, i.e. iSSO < FPA < SSO (the smaller the better) in $\boldsymbol{N}_{\text{gBest}}$.

4.2.4 Compare N_{avg}, n_{avg}, and n_{avg}/N_{avg}: From Tables 2–5, the proposed iSSO and FPA are better than SSO. Therefore, we will only focus on comparing the proposed iSSO and FPA algorithms in N_{avg}, n_{avg}, and $n_{\text{avg}}/N_{\text{avg}}$ in this subsection.

The n_{avg} achieved by the proposed iSSO is better than that achieved by FPA for $T = 5$, 10, 15, 20, and 25; FPA has better n_{avg} for $T = 30$, 35, 40, 45, and 50 from Table 6. Furthermore, from the Tukey test of ANOVA at significance level of 0.05 in Table 6, there is a significant difference between iSSO and FPA in N_{avg} for each T and in n_{avg} for $T = 5$, 10, 15, and 20.

Also from Table 6, the values of both N_{avg} and $n_{\text{avg}}/N_{\text{avg}}$ obtained by iSSO are always larger than those obtained by FPA for each T. Note larger number of N_{avg} and n_{avg} equates to the model efficiently running more numbers in the same duration of time; in addition, a smaller ratio of $n_{\text{avg}}/N_{\text{avg}}$ also indicates more efficiency. Therefore, the longer the runtime is, the better the iSSO results are compared with FPA in the fitness calculation number and $n_{\text{avg}}/N_{\text{avg}}$. On the whole, iSSO is more efficient than FPA.

From Tables 2–6, the proposed iSSO algorithm outperforms the FPA and SSO in the SDM benchmark.

4.3 Double diode benchmark

For the double diode case, the solutions, fitness calculation numbers, and number of times improved existing solutions are obtained from the proposed iSSO and FPA algorithms are as shown in Tables 7–9 and discussed as follows.

4.3.1 Compare F_{avg}, F_{min}, F_{max}, and F_{stdev}: The solution quality (i.e., F_{avg}, F_{min}, and F_{max}) and the solution robustness (i.e., F_{stdev}) of the proposed iSSO algorithm is superior to that of the FPA algorithm because all of average, minimal (the best), maximal (the worst), and standard deviations of the fitness values for each T obtained by iSSO are better than those obtained by FPA, as shown in Table 8. Also, the Tukey test of ANOVA at a significance level of 0.05 shows that there is always a significant difference between iSSO and FPA in F_{avg} as shown in the second column of Table 6. Hence, the proposed iSSO outperforms FPA effectively.

4.3.2 Compare N_{gBest}: The number of *gBest* that are better than the existing solutions, i.e. N_{gBest}, for iSSO is 549 which show that the proposed iSSO is almost able to improve the existing solutions in each test for each T. On the contrary, $N_{\text{gBest}} = 0$ for each T except $N_{\text{gBest}} = 1$ for $T = 35$. Hence, the proposed iSSO has a higher chance of obtaining a better solution than FPA and this observation is also found in the SDM benchmark.

4.3.3 Compare N_{avg}, n_{avg}, and n_{avg}/N_{avg}: From Table 9, the proposed iSSO always has a better value in N_{avg}, n_{avg}, and $n_{\text{avg}}/N_{\text{avg}}$ than that of FPA. Also, based on the statistical test, both N_{avg} and n_{avg} obtained from the proposed iSSO are significantly better than that of FPA at a significance level of 0.05 from the MANOVA test as shown in Table 9. Hence, the efficiency of the proposed iSSO is superior to that of FPA.

From Tables 7–9, the performance of the proposed iSSO algorithm is much better than that of FPA with more accuracy, efficiency, and robustness for the DDM benchmark.

Table 6 The number of N_{avg}, n_{avg}, and $n_{\text{avg}}/N_{\text{avg}}$

T	iSSO			iSSO–FPA		
	N_{avg}	n_{avg}	n/N_{avg}	N_{avg}	n_{avg}	n/N_{avg}
5	**3325518.90**	**3173124.72**	**0.95**	918176.62[a]	844424.66[a]	−0.02
10	**6643362.74**	**5632796.54**	**0.85**	1839251.72[a]	1148665.34[a]	−0.08
15	**9957496.98**	**7751917.68**	**0.78**	2758892.40[a]	1082722.10[a]	−0.15
20	**13272889.94**	**9813480.92**	**0.74**	3676406.20[a]	1198085.08[a]	−0.16
25	**16586784.52**	**11547902.88**	**0.70**	4593668.62[a]	823991.88	−0.19
30	**19900980.56**	12690608.78	**0.64**	5511860.82[a]	−171298.66	−0.25
35	**23214663.08**	13751832.76	**0.59**	6428569.54[a]	−217367.36	−0.24
40	**26528197.96**	14495115.20	**0.55**	7345632.74[a]	−2097131.90	−0.31
45	**29841840.28**	18194183.96	**0.61**	8263569.78[a]	−1593836.46	−0.31
50	**33155762.14**	19591887.18	**0.59**	9182215.84[a]	−656948.30	−0.25

[a]The average difference is significant at significance level of 0.05 from the Tukey test of MANOVA.

Table 7 Descriptive statistics of the proposed iSSO and FPA for the **DDM** benchmark

T	F_{avg}	F_{min}	F_{max}	F_{stdev}
5	**0.000984245079678920**	**0.0009824931442180**	**0.0009860219274770**	$\mathbf{1.1043 \times 10^{-6}}$
10	**0.000983464119822360**	**0.0009824872198090**	**0.0009859477000040**	$\mathbf{8.7465 \times 10^{-7}}$
15	**0.000983399517044620**	**0.0009824834928690**	**0.0009855223046410**	$\mathbf{8.0096 \times 10^{-7}}$
20	**0.000983210205132080**	**0.0009824847245010**	**0.0009843696769690**	$\mathbf{5.9532 \times 10^{-7}}$
25	**0.000982936341426980**	**0.0009824833179640**	**0.0009842797252360**	$\mathbf{5.0168 \times 10^{-7}}$
30	**0.000982798740381660**	**0.0009824834379240**	**0.0009833317961100**	$\mathbf{2.4004 \times 10^{-7}}$
35	**0.000982765419051640**	**0.0009824842556020**	**0.0009835644240890**	$\mathbf{2.9939 \times 10^{-7}}$
40	**0.000982734209150680**	**0.0009824832054510**	**0.0009832855883340**	$\mathbf{2.4221 \times 10^{-7}}$
45	**0.000982743859456140**	**0.0009824833656480**	**0.0009834562474810**	$\mathbf{2.7640 \times 10^{-7}}$
50	**0.000982665203364340**	**0.0009824831731220**	**0.0009833574398800**	$\mathbf{2.0933 \times 10^{-7}}$

Table 8 The values of the difference fitness in iSSO−FPA for the DDM benchmark

Time	F_{avg}	F_{min}	F_{max}	F_{stdev}
5	-5.672×10^{-6}[a]	-1.901×10^{-7}	-2.687×10^{-5}	-5.048×10^{-6}
10	-1.943×10^{-6}[a]	-2.480×10^{-7}	-3.753×10^{-6}	-5.125×10^{-7}
15	-1.847×10^{-6}[a]	-3.598×10^{-9}	-1.709×10^{-6}	-4.301×10^{-7}
20	-1.702×10^{-6}[a]	-1.220×10^{-7}	-2.390×10^{-6}	-6.071×10^{-7}
25	-1.847×10^{-6}[a]	-4.690×10^{-8}	-1.757×10^{-6}	-4.937×10^{-7}
30	-2.022×10^{-6}[a]	-1.527×10^{-9}	-3.053×10^{-6}	-9.202×10^{-7}
35	-1.659×10^{-6}[a]	-2.133×10^{-8}	-2.492×10^{-6}	-7.040×10^{-7}
40	-1.933×10^{-6}[a]	-9.975×10^{-9}	-2.822×10^{-6}	-9.758×10^{-7}
45	-1.778×10^{-6}[a]	-2.985×10^{-9}	-2.659×10^{-6}	-9.546×10^{-7}
50	-1.758×10^{-6}[a]	-2.536×10^{-9}	-2.677×10^{-6}	-9.590×10^{-7}

[a]The average difference is significant at a significance level of 0.05 from the Tukey test of MANOVA.

Table 9 The N_{avg}, n_{avg}, and n_{avg}/N_{avg} obtained from iSSO and FPA for the DDM benchmark

Time	iSSO			iSSO-FPA		
	N_{avg}	n_{avg}	n/N_{avg}	N_{avg}	n_{avg}	n/N_{avg}
5	**1886533.36**	**1510782.76**	**0.80**	440308.24[a]	72856.94	−0.19
10	**3758687.52**	**2944210.34**	**0.78**	875488.40[a]	96837.90	−0.21
15	**5630053.02**	**4800050.88**	**0.85**	1302406.68[a]	501406.86[a]	−0.14
20	**7493549.70**	**6466333.96**	**0.86**	1736859.94[a]	749614.70[a]	−0.13
25	**9352190.80**	**8179507.96**	**0.87**	2153945.40[a]	1015707.86[a]	−0.13
30	**11220982.22**	**10291013.56**	**0.92**	2582847.04[a]	1667818.04[a]	−0.08
35	**13091855.26**	**11359389.94**	**0.87**	3039236.52[a]	1365231.98[a]	−0.12
40	**14969219.70**	**13389713.68**	**0.89**	3455286.88[a]	1984330.10[a]	−0.1
45	**16836151.14**	**15449998.18**	**0.92**	3902397.78[a]	2570879.70[a]	−0.08
50	**18683558.34**	**17057804.90**	**0.91**	4305415.40[a]	2702061.70[a]	−0.09

[a]The average difference is significant at a significance level of 0.05 from the Tukey test of MANOVA.

5 Conclusion

In this study, an improved SSO algorithm, known as iSSO, is proposed to optimise the newest solar energy problem with a comprehensive comparative study of its performance for two well-known benchmarks, i.e. the solar cell SDM benchmark and the solar cell DDM benchmark.

From the experiments in the SDM benchmark problem of Section 4, the solutions obtained from the proposed iSSO is at least 10^{-05}significantly better than that of SSO (from Table 3). In addition, it is also slightly better than FPA with a 140/25 = 560% (from Table 5) chance of obtaining a better solution than FPA and is 918176.62 and 844424.66 less in N_{avg} and n_{avg}, respectively, (from Table 6). The advantage of iSSO is more evident and stark in the DDM benchmark problem of which the proposed iSSO is at least 10^{-06}significantly better than FPA in F_{avg} (from Table 8), 54,900% higher in N_{gBest}, 440308.24 and 72856.94 less in N_{avg} and n_{avg}, respectively, (from Table 9) than those of FPA. Note that FPA outperforms other evolutionary algorithms in both the SDM and DDM benchmark problems [4, 9, 18, 22, 23, 26, 27, 29–31, 38].

Hence, the overall efficiency of the proposed iSSO is much better than that of FPA with greater accuracy and robustness for both benchmarks. Thus, the proposed iSSO algorithm is the best existing algorithm to optimise the newest solar energy problems. Given the great outcome of iSSO, the proposed algorithm proposed iSSO algorithm should be extended to further applications and different optimisation problems with more variables or larger-scale benchmarks in future studies.

6 Acknowledgments

The authors wish to thank the anonymous editors and the reviewers for their constructive comments and recommendations, which have significantly improved the presentation of this paper. This research was supported in part by the National Science Council of Taiwan, R.O.C. under grant NSC101-2221- E-007-079- MY3 and NSC 102-2221-E-007-086-MY3.

7 References

[1] International Energy Agency, Photovoltaic Power Systems Programme.: *'2015 snapshot of global photovoltaic markets'*. Report IEA PVPS T1-29, 2016

[2] Ishaque, K., Salam, Z., Mekhilef, S., *et al.*: 'Parameter extraction of solar photovoltaic modules using penalty-based differential evolution', *Appl. Energy*, 2012, **99**, pp. 297–308

[3] Ishaque, K., Salam, Z.: 'An improved modeling method to determine the model parameters of photovoltaic (PV) modules using differential evolution (DE)', *Sol. Energy*, 2011, **85**, (9), pp. 2349–2359

[4] Alam, D.F., Yousri, D.A., Eteiba, M.B.: 'Flower pollination algorithm based solar PV parameter estimation', *Energy Convers. Manage.*, 2015, **101**, pp. 410–422

[5] Mares, O., Paulescu, M., Badescu, V.: 'A simple but accurate procedure for solving the five-parameter model', *Energy Convers. Manage.*, 2015, **105**, pp. 139–148

[6] Wolf, M., Noel, G.T., Stirn, R.J.: 'Investigation of the double exponential in the current–voltage characteristics of silicon solar cells', *IEEE Trans. Electron Devices*, 1977, **24**, (4), pp. 419–428

[7] Kassis, A., Saad, M.: 'Analysis of multi-crystalline silicon solar cells at low illumination levels using a modified two-diode model', *Sol. Energy Mater. Sol. Cells*, 2010, **94**, (12), pp. 2108–2112

[8] Nishioka, K., Sakitani, N., Uraoka, Y., *et al.*: 'Analysis of multicrystalline silicon solar cells by modified 3-diode equivalent circuit model taking leakage current through periphery into consideration', *Sol. Energy Mater. Sol. Cells*, 2007, **91**, (13), pp. 1222–1227

[9] Askarzadeh, A., Rezazadeh, A.: 'Parameter identification for solar cell models using harmony search-based algorithms', *Sol. Energy*, 2012, **86**, (11), pp. 3241–3249

[10] Renno, C., Petito, F., Gatto, A.: 'Artificial neural network models for predicting the solar radiation as input of a concentrating photovoltaic system', *Energy Convers. Manage.*, 2015, **106**, pp. 999–1012

[11] Chin, V.J., Salam, Z., Ishaque, K.: 'Cell modelling and model parameters estimation techniques for photovoltaic simulator application: A review', *Appl. Energy*, 2015, **154**, pp. 500–519

[12] Chan, D.S.H., Phang, J.C.H.: 'Analytical methods for the extraction of solar-cell single-and double-diode model parameters from IV characteristics', *IEEE Trans. Electron Devices*, 2014, **86**, pp. 3241–3249

[13] Saleem, H., Karmalkar, S.: 'An analytical method to extract the physical parameters of a solar cell from four points on the illuminated curve', *IEEE Electron Device Lett.*, 2009, **30**, pp. 349–352

[14] Lun, S., Du, C., Yang, G., *et al.*: 'An explicit approximate I–V characteristic model of a solar cell based on pad approximants', *Sol. Energy*, 2013, **92**, pp. 147–159

[15] Jain, A., Kapoor, A.: 'Exact analytical solutions of the parameters of real solar cells using Lambert W-function', *Sol. Energy Mater. Sol. Cells*, 2004, **81**, pp. 269–277

[16] Jain, A., Sharma, S., Kapoor, A.: 'Solar cell array parameters using Lambert W-function', *Sol. Energy Mater. Sol. Cells*, 2006, **90**, pp. 25–31

[17] Kim, W., Choi, W.: 'A novel parameter extraction method for the one-diode solar cell model', *Sol. Energy*, 2010, **84**, (6), pp. 1008–1019

[18] Easwarakhanthan, T., Bottin, J., Bouhouch, I., *et al.*: 'Nonlinear minimization algorithm for determining the solar cell parameters with microcomputers', *Int. J. Sol. Energy* , 1986, **4**, (1), pp. 1–12

[19] Elbaset, A.A., Ali, H., Abd-El Sattar, M.: 'Novel seven-parameter model for photovoltaic modules', *Sol. Energy Mater. Sol. Cell*, 2014, **130**, pp. 442–455

[20] Ma, T., Yang, H., Lu, L.: 'Development of a model to simulate the performance characteristics of crystalline silicon photovoltaic modules/strings/arrays', *Sol. Energy*, 2014, **100**, pp. 31–41

[21] Tossa, A.K., Soro, Y.M., Azoumah, Y., *et al.*: 'A new approach to estimate the performance and energy productivity of photovoltaic modules in real operating conditions', *Sol. Energy*, 2014, **110**, pp. 543–560

[22] Zagrouba, M., Sellami, A., Bouaïcha, M., *et al.*: 'Identification of PV solar cells and modules parameters using the genetic algorithms: application to maximum power extraction', *Sol. Energy*, 2010, **84**, (5), pp. 860–866

[23] Jervase, J.A., Bourdoucen, H., Al-Lawati, A.: 'Solar cell parameter extraction using genetic algorithms', *Meas. Sci. Technol.*, 2001, **12**, (11), p. 1922

[24] Ye, M., Wang, X., Xu, Y.: 'Parameter extraction of solar cells using particle swarm optimization', *J. Appl. Phys.*, 2009, **105**, (9), pp. 094502–094508

[25] Wei, H., Cong, J., Lingyun, X., *et al.*: 'Extracting solar cell model parameters based on chaos particle swarm algorithm'. Int. Conf. on Electric Information and Control Engineering (ICEICE), 2011, 2011, pp. 398–402

[26] Oliva, D., Cuevas, E., Pajares, G.: 'Parameter identification of solar cells using artificial bee colony optimization', *Energy*, 2014, **72**, pp. 93–102

[27] El-Naggar, K.M., AlRashidi, M.R., AlHajri, M.F., *et al.*: 'Simulated annealing algorithm for photovoltaic parameters identification', *Sol. Energy*, 2012, **86**, (1), pp. 266–274

[28] AlRashidi, M.R., AlHajri, M.F., El-Naggar, K.M., *et al.*: 'A new estimation approach for determining the *I–V* characteristics of solar cells', *Sol. Energy*, 2011, **85**, (7), pp. 1543–1550

[29] Ma, J., Ting, T.O., Man, K.L., *et al.*: 'Parameter estimation of photovoltaic models via cuckoo search', *J. Appl. Math.*, 2013, **85**, pp. 1543–1550

[30] Askarzadeh, A., Rezazadeh, A.: 'Artificial bee swarm optimization algorithm for parameters identification of solar cell models', *Appl. Energy*, 2013, **102**, pp. 943–949

[31] Yuan, X., Xiang, Y., He, Y.: 'Parameter extraction of solar cell models using mutative-scale parallel chaos optimization algorithm', *Sol. Energy*, 2014, **108**, pp. 238–251

[32] http://integrationandcollaboration.org/SSO.html

[33] Yeh, W.C.: 'Novel swarm optimization for mining classification rules on thyroid gland data', *Inf. Sci.*, 2012, **197**, pp. 65–76

[34] Yeh, W.C.: 'Optimization of the disassembly sequencing problem on the basis of self-adaptive simplified swarm optimization', *IEEE Trans. Syst. Man Cybern. A, Syst. Humans*, 2012, **42**, (1), pp. 250–261

[35] Yeh, W.C.: 'Simplified swarm optimization in disassembly sequencing problems with learning effects', *Comput. Oper. Res.*, 2012, **39**, (9), pp. 2168–2177

[36] Yeh, W.C.: 'New parameter-free simplified swarm optimization for artificial neural network training and its application in the prediction of time series', *IEEE Trans. Neural Netw. Learn. Syst.*, 2013, **24**, (4), pp. 661–665

[37] Azizipanah-Abarghooee, R., Niknam, T., Gharibzadeh, M., *et al.*: 'Robust, fast and optimal solution of practical economic dispatch by a new enhanced gradient-based simplified swarm optimisation algorithm', *IET Gener. Transm. Distrib.*, 2013, **7**, (6), pp. 620–635

[38] Azizipanah-Abarghooee, R.: 'A new hybrid bacterial foraging and simplified swarm optimization algorithm for practical optimal dynamic load dispatch', *Int. J. Electr. Power Energy Syst.*, 2013, **49**, pp. 414–429

[39] Villalva, M.G., Gazoli, J.R.: 'Comprehensive approach to modeling and simulation of photovoltaic arrays', *IEEE Trans. Power Electron.*, 2009, **24**, (5), pp. 1198–1208

[40] Yeh, W.C.: 'An improved simplified swarm optimization', *Knowl.-Based Syst.*, 2015, **82**, pp. 60–69